
Documentação - MAME

Release 0.260

Tradução e Revisão Wellington T. Uemura

26 abr. 2024

1	O QUE É O MAME?	3
1.1	I. Objetivo	3
1.2	II. Custo	4
1.3	III. Software em formato de imagem	4
1.4	IV. Obras derivativas	4
1.5	V. Mais informações e contato legal	4
2	ALERTA SOBRE A SAÚDE	5
2.1	Fotossensibilidade / Ataques Epiléticos	5
3	CONFIGURANDO O MAME PARA SER USADO PELA PRIMEIRA VEZ	7
3.1	Os sistemas emulados	7
3.2	A compatibilidade dos sistemas operacionais	7
3.3	Requisitos do sistema	8
3.4	Instalando	9
3.5	Compilando o MAME	10
3.6	Compilação cruzada	53
3.7	Configuração inicial	58
4	NOÇÕES GERAIS DE USO E CONFIGURAÇÃO	69
4.1	Primeiros passos	70
4.2	A interface do usuário	84
4.3	Teclas já predefinidas	95
4.4	Cardápio de opções	112
4.5	Como o MAME localiza os seus arquivos?	158
4.6	Front-ends	166
4.7	Sobre as ROMs e os seus conjuntos	167
4.8	Problemas comuns e algumas perguntas frequentes	174
4.9	Perguntas frequentes sobre alguns sistemas	183
5	CONFIGURAÇÕES E OPÇÕES PARA A LINHA DE COMANDO	251
5.1	Opções Universais de linha de comando	251
5.2	Configurações específicas para o Windows	333
5.3	Configurações específicas para as versões SDL	335
5.4	Índice das opções para a linha de comando	341
6	PLUG-INS	353

6.1	Introdução	353
6.2	Usando os plug-ins	353
6.3	Plug-ins inclusos	354
7	CONFIGURAÇÕES AVANÇADAS	369
7.1	Diversos arquivos de configuração	369
7.2	Lidando com o caminho dos arquivos	376
7.3	Desativando o interruptor de câmbio	377
7.4	Efeitos BGFX para (quase) todo mundo	378
7.5	Os efeitos HLSL para Windows	384
7.6	Efeitos GLSL para *nix, OS X e Windows	393
7.7	Criando e usando um LUT 3D no MAME	396
7.8	Arquivos de configuração para os controles	409
7.9	ID fixo para os controles	415
7.10	A arma de luz	418
7.11	MAME em câmera lenta	433
7.12	Truques para tornar a vida mais fácil	435
8	O DEPURADOR	489
8.1	Introdução	489
8.2	Os comandos do depurador	490
8.3	Determinando os dispositivos e as faixas de endereço	540
8.4	A sintaxe das expressões do depurador	542
9	INTERFACE PARA SCRIPTS LUA	547
9.1	Introdução	547
9.2	Características	548
9.3	Referência da API	549
9.4	Tutorial do console interativo Lua	648
10	FERRAMENTAS ADICIONAIS	651
10.1	Chdman	651
10.2	Imgtool	662
10.3	Castool	690
10.4	Floptool	692
10.5	Outras ferramentas incluídas com o MAME	694
10.6	Ferramentas voltadas ao desenvolvimento	694
11	CONTRIBUINDO COM O MAME	697
11.1	Testando e relatando bugs	697
12	DEFINIÇÕES TÉCNICAS	715
12.1	As convenções das nomenclaturas usadas pelo MAME	716
12.2	Os arquivos layout	719
12.3	Usando scripts no layout do MAME	813
12.4	Os localizadores de objetos	830
12.5	Entrada do sistema	850
12.6	O dispositivo da interface de memória	860
12.7	O dispositivo da interface da ROM	863
12.8	O device_disasm_interface e os desmontadores	865
12.9	A memória emulada do sistema e o gerenciamento dos espaços de endereçamento	870
12.10	O novo subsistema de disquete	887
12.11	O novo subsistema SCSI	899
12.12	A implementação da nova família 6502	904
12.13	Software de renderização 3D no MAME	913

13 O MAME E A PREOCUPAÇÃO COM A SEGURANÇA	939
14 LICENÇA	941
15 LICENCIAMENTO DESTE DOCUMENTO E SUA TRADUÇÃO	943
16 CÓDIGO DE CONDUTA PARA COLABORADORES	945
16.1 Nossa promessa	945
16.2 Nossos padrões	945
16.3 Nossas responsabilidades	946
16.4 Escopo	946
16.5 Aplicação	946
16.6 Atribuição	946



Nota: Esta publicação é o resultado de um trabalho em contante evolução. Você pode acompanhar se há alguma atualização dos tópicos e assuntos aqui tratados acessando o [issue tracker](#). Veja como você pode contribuir na seção [contribute](#) no site do GitHub.

CAPÍTULO 1

O QUE É O MAME?

O MAME é uma estrutura multiúso de trabalho voltado para a emulação, com o tempo (originalmente significava *Multiple Arcade Machine Emulator* ou em tradução livre, *Multi Emulador de Máquinas Arcade*) absorveu o projeto irmão MESS (Multi Emulator Super System) e o AGEMAME (Arcade Gambling Extensions for MAME). Agora também documenta uma grande variedade de computadores, principalmente aqueles bem mais antigos, consoles de videogame, calculadoras, indo muito além do seu foco inicial.

1.1 I. Objetivo

O principal objetivo do MAME é preservar décadas de história, impedindo com que a evolução da tecnologia este importante software "vintage" não seja esquecido e se perca ao longo tempo. Isso se torna possível usando o próprio código-fonte do MAME para documentar o funcionamento do equipamento físico original. Com a evolução do desenvolvimento do MAME, programas e jogos que foram originalmente desenvolvidos para rodarem apenas no seu equipamento físico original, agora através da emulação é possível rodá-los novamente usando um PC moderno, sendo referência ao funcionamento interno dos sistemas emulados e um efeito colateral bem-vindo, este, porém não é o foco principal do projeto.

Não é nossa intenção infringir quaisquer direitos autorais assim como as patentes dos softwares ou sistemas. O MAME e todo o seu código-fonte é de criação autoral, disponível gratuitamente. O emulador requer se utiliza das ROMs originais, CD, disco rígido e outras imagens em mídia usadas originalmente pelos sistemas ou pelos drivers para operar e que devem ser providenciados pelo usuário. Nenhuma parte do código-fonte original do sistema ou de qualquer outro software estão inclusos no executável do MAME.

1.2 II. Custo

O MAME é disponibilizado de forma gratuita assim como o seu código fonte.

O projeto como um todo é distribuído através da Licença Pública Geral GNU, versão 2 ou mais recente (GPL-2.0+), porém, a maior parte do código, incluindo a funcionalidade principal, também estão disponíveis através da cláusula Nº 3 da licença BSD (3-Clause BSD License).

1.3 III. Software em formato de imagem

Os formatos de imagem de mídias como ROM, CD, CD-ROM, disco rígido dentre outros, são materiais protegidos por direitos autorais. Eles **não podem ser distribuídos** sem uma permissão **explícita** dos seus respectivos responsáveis ou detentores da propriedade intelectual destes direitos, tão pouco são "abandonware"¹ e qualquer dos programas compatíveis com o MAME jamais perdem os seus direitos autorais.

O MAME NÃO FOI CRIADO E TÃO POUCO SE DESTINA A SER USADO COMO UMA FERRAMENTA DE PIRATARIA OU DE VIOLAÇÃO DE DIREITOS AUTORAIS EM MASSA. PORTANTO, SOMOS VEEMENTEMENTE CONTRA QUE O MAME SEJA COMERCIALIZADO, ANUNCIADO OU VINCULADO A QUALQUER TIPO DE RECURSOS QUE FORNEÇAM CÓPIAS ILEGAIS DE ROM, CD, CD-ROM, DISCO RÍGIDO OU OUTRAS IMAGENS DE MÍDIA DE QUALQUER TIPO.

1.4 IV. Obras derivativas

MAME é uma propriedade intelectual com marca registrada, caso queira usar o nome **MAME** como parte do seu trabalho ou derivado dele, **É OBRIGATÓRIO** um pedido prévio de permissão de uso.

A versão do MAME deve refletir o número da versão usada no momento em qualquer trabalho que seja derivado dele.

1.5 V. Mais informações e contato legal

Para questões relacionadas à licença do MAME, marca comercial ou qualquer outra utilização, acesse:

<https://www.mamedev.org/legal>

MAME ®

Copyright © 1997-2024 Nicola Salmoria and the MAME team

MAME é uma marca registrada de Gregory Ember

¹ Abandonware é um "software" que foi descontinuado que não é mais desenvolvido, mantido ou comercializado, seja porque ficou obsoleto, a empresa não existe mais ou qualquer outro motivo que ao olhar do usuário o "software" tenha sido abandonado, daí o termo *abandonware* ou **abandoned software**.

ALERTA SOBRE A SAÚDE

2.1 Fotossensibilidade / Ataques Epiléticos

Um percentual muito baixo de pessoas podem sofrer ataques epiléticos ou desmaios quando expostos a determinados padrões, ou "flashes" de luz. A exposição a certos padrões, fundos de tela numa televisão, tela de computador, jogando videogames, pode provocar ataques epiléticos, ou desmaios nestes indivíduos.

Estas condições podem desencadear sintomas epiléticos não detectados previamente em pessoas sem histórico de epilepsia ou convulsão. Podem ser inclusas nestas condições a precisão ou não de uma emulação, o desempenho de um computador durante a execução do MAME, os controladores da placa de vídeo, o seu monitor e diversos outros fatores. Caso você ou qualquer outra pessoa da sua família, tenha epilepsia ou já tenha sofrido convulsões de qualquer tipo, consulte um médico antes de voltar a usar o MAME.

Se durante a utilização do MAME você apresentar alguns dos sintomas abaixo **INTERROMPA O USO IMEDIATAMENTE**:

- Vertigem
- Visão alterada
- Espasmos musculares ou oculares
- Perda de consciência
- Desorientação
- Qualquer movimento involuntário ou convulsivo
- Convulsões

CONFIGURANDO O MAME PARA SER USADO PELA PRIMEIRA VEZ

3.1 Os sistemas emulados

O ProjectMESS contém uma lista completa dos sistemas atualmente emulados. Note que ter um sistema emulado, não significa que a emulação dele está perfeita. Caso queira:

1. Verificar a condição geral da emulação nas páginas wiki de cada sistema, acessível a partir da página de drivers (por exemplo, para o Apple Macintosh, olhe o arquivo de driver `mac.cpp`, é possível também ter acesso aos drivers do **macplus** e **macse**).
2. Assim como ler também os registros correspondentes no arquivo **sysinfo.dat** para entender melhor quais problemas é possível encontrar durante a execução de um sistema no MAME. (para o Apple Macintosh Plus, também é necessário verificar esse arquivo).

Como alternativa, é possível ver essa condição por conta própria, caso haja, prestando atenção na tela de aviso vermelha ou bege que aparece antes do início da emulação. Observe que, caso tenha informações que possam ajudar a melhorar a emulação de um sistema emulado ou se você puder contribuir com correções, ou novas adições ao código-fonte atual, siga as instruções na página de contato ou poste uma mensagem no Fórum do MAME em <https://forum.mamedev.org/>

Grande parte do MAME é programado em C/C++ e outras linguagens auxiliares. Atualmente o MAME consegue emular mais de 3200 sistemas independentes das últimas 5 décadas.

3.2 A compatibilidade dos sistemas operacionais

O código-fonte atual pode ser compilada diretamente nos principais sistemas operacionais: Microsoft Windows (ambos com suporte nativo para DirectX/BGFX ou com suporte SDL), Linux, FreeBSD e Max OS X. Há ainda a compatibilidade para ambas as versões de 32 e de 64 bits, saiba que a versão de 64 bits mostra uma melhora considerável no desempenho em comparação com a sua variante de 32 bits.

3.3 Requisitos do sistema

O desenvolvimento do MAME gira em torno das linguagens C/C++ e já foi portado para diferentes plataformas. Com o passar do tempo, à medida que o equipamento físico do computador vai evoluindo, o código do MAME evolui junto para aproveitar melhor o maior poder de processamento e os novos recursos.

Os binários oficiais do MAME são compilados e projetados para serem executados rodar em qualquer sistema Windows. Os requisitos mínimos são:

- Processador Intel Core ou equivalente com pelo menos 2.0 GHz.
- Sistema Operacional com 64-bit (Windows 7 SP1 ou mais recente, macOS 10.14 ou mais recente).
- 4 GB de RAM.
- Uma placa gráfica compatível com Direct3D, OpenGL ou Vulkan.
- Qualquer placa de som compatível com DirectSound.

Claro, os requisitos mínimos são apenas um pequeno exemplo. Nem sempre é possível obter o melhor desempenho usando a configuração acima, porém, o MAME deverá rodar sem maiores problemas. As versões mais recentes do MAME tendem a exigir mais recursos do equipamento do que as suas versões anteriores, assim, as versões mais antigas do MAME poderão ter um desempenho melhor num PC mais fraco ao custo de perder as melhorias feitas nos sistemas existentes, dos novos sistemas que foram adicionados e das correções posteriores a versão do MAME que estiver usando.

O MAME tira vantagem dos recursos 3D do equipamento físico para a exibição das ilustrações assim como o redimensionamento do software ou do jogo na tela inteira. Para fazer uso destes benefícios, é obrigatório ter uma placa de vídeo mais recente capaz de lidar com Direct3D 11 e no mínimo 2 GiB de memória RAM.

Os filtros especiais HLSL ou GLSL assim como o efeito de simulação de uma tela de tubo CRT, causam uma sobrecarga extra na emulação, especialmente em resoluções mais altas. Exigindo uma placa de vídeo moderna o bastante para aguentar o tranco, pois, a carga do processamento sobe exponencialmente à medida que se aumenta também a resolução. Caso o HLSL ou o GLSL fiquem muito pesado, tente reduzir o tamanho da resolução do vídeo do sistema emulado.

Tenha sempre em mente que, mesmo usando os computadores mais rápidos disponíveis hoje, o MAME ainda é incapaz de rodar alguns sistemas em sua velocidade nativa. O principal objetivo do projeto não é fazer com que todos os sistemas emulados rodem na sua velocidade nativa, seja no seu computador ou, seja lá onde o MAME esteja sendo rodado; o interesse do projeto é documentar o hardware e reproduzir o seu comportamento original tão fielmente quanto for possível.

3.3.1 As extrações da BIOS e dos programas

Para que o MAME consiga emular a maioria destes sistemas, o conteúdo dos circuitos integrados originais destes aparelhos precisam ser extraídos. Isso pode ser feito extraíndo estes dados do aparelho original você mesmo, ou procurando por eles na internet por sua conta e risco.

O MAME não fornece, disponibiliza ou vem acompanhado com nenhum deles justamente pelo fato destes conteúdos estarem protegidos por leis de direitos autorais. Caso tenha interesse em encontrar algum software que rode numa dos sistema já emulados, lembre-se, o Google e outros sites de pesquisa são os seus melhores amigos.

3.4 Instalando

3.4.1 Microsoft Windows

Baixe a versão mais recente disponível em www.mamedev.org e extraia o seu conteúdo. Dentro da pasta onde ele foi extraído haverá vários arquivos e pastas (abaixo mostraremos para que servem alguns deles), dentre eles o arquivo mais importante de todos que é o **mame.exe**, o emulador em si. Ele é um programa que funciona na linha de comando.

O processo de instalação termina aqui, fácil não?

3.4.2 Outros sistemas operacionais

Neste caso procure por uma versão pré compilada do executável do (SDL) MAME que pode ser encontrado em alguns repositórios da distribuição Linux da sua preferência, ou, compile o MAME você mesmo, baixando e descompactando o arquivo do código-fonte em alguma pasta qualquer.

Caso queira compilar o MAME você mesmo, consulte a seção [Compilando o MAME](#) para mais detalhes.

3.5 Compilando o MAME

- *Primeiros passos*
 - *Acelerando uma compilação*
 - *Exemplos práticos para todas as plataformas*
- *Microsoft Windows*
 - *Configurando o pacote MSYS2 já pronto*
 - *Preparando a instalação do MSYS2 manualmente*
 - *Resolvendo possíveis problemas com o MSYS2*
 - *Compilando com o Microsoft Visual Studio*
 - *Algumas observações sobre o ambiente MSYS2*
- *Linux*
 - *Fedora Linux*
 - *Debian e Ubuntu (incluindo dispositivos Raspberry Pi e ODROID)*
 - *Arch Linux*
- *Apple macOS*
- *Javascript Emscripten e HTML*
- *Opções gerais para a compilação*
 - *Usando ferramentas de compilação alternativas*
 - *Incluindo os subconjuntos dos sistemas suportados*
 - *Recursos opcionais*
 - *Sede das bibliotecas e framework*
- *Problemas conhecidos*
 - *Problemas relacionados com versões específicas do compilador*
 - *Recursos do código-fonte fortify da biblioteca GNU C*
 - *Problemas que afetam o Microsoft Visual Studio*
 - *Ponto de entrada não encontrado*
- *Configurações para compilações não ortodoxas*
 - *Lincando com o LLVM linker*
 - *Usando libc++ no Linux*
 - *Usando uma instalação do GNU GCC libstdc++ que esteja fora do local tradicional no Linux*
- *O MAME travou, o que fazer?*
 - *Investigando e lidando com problemas comuns*
 - *A tela de travamento*
 - *Usando o gdb*

– *Usando o AddressSanitizer*

3.5.1 Primeiros passos

Caso nunca tenha compilado o MAME antes, não sabe por onde começar e está completamente perdido, este documento irá te ajudar com o mínimo necessário para que tenha sucesso na sua primeira compilação do MAME.

Antes, alguns pontos importantes:

- É necessário um compilador C++17 e suas respectivas bibliotecas, para o GCC, a versão 10.3 é mínima aceitável ou então versões mais recentes. Já o Clang é necessário a versão 11 ou mais recente. O executável do MAME rodará com a biblioteca GNU libstdc++ versão 10.3 ou mais recente, assim como, libc++ versão 11 ou mais recente. A versão inicial de qualquer grande lançamento de versão do GCC deve ser evitada. Por exemplo, caso queira compilar o MAME com o GCC 12, é preferível que você utilize a versão 12.1 ou mais recente.
- A versão nativa do MAME roda no Windows, então uma grande ênfase será dada a compilação neste sistema operacional.
- O **git** é a melhor maneira para baixar o código-fonte do MAME e mantê-lo atualizado. Antigamente era preciso baixar o código-fonte do site oficial, aplicar dezenas de correções (patch) para que o mesmo fosse atualizado e corrigido, quando eram aplicados fora da ordem, geravam outros problemas.

Hoje o processo é automatizado com o git, com um simples comando ele vai baixar todo o código-fonte assim como aplicar todas as alterações feitas ao código de maneira organizada e bem controlada.

Cada nova modificação que é enviada ao código-fonte do MAME e autorizada pelos administradores, o servidor conta como sendo **1 commit** ou 1 envio.

- **make** é o comando que faz o papel do *mestre de cerimônias*, é ele quem vai repassar todos comandos ou opções que vierem depois dele para o compilador e outras ferramentas.

Digite *make* no diretório raiz onde se encontra o código-fonte do MAME para que ele leia as instruções contidas num arquivo chamado **Makefile** para que uma versão do MAME seja compilada, observe que é possível usar outras opções fazendo com que a compilação do MAME seja customizada e atenda as suas necessidades, como, por exemplo, se beneficiar das propriedades do seu processador dentre outras opções que será mostrada mais adiante.

As opções devem **sempre** ser adicionadas depois do comando **make** como mostra o exemplo abaixo:

```
make VERBOSE=1
```

Várias outras opções podem ser adicionadas desde que estejam separadas por espaço, abra o arquivo **makefile** num editor de texto e veja quais são aquelas que estão disponíveis.

- Algumas vezes o processo de compilação é interrompido antes de chegar ao fim, os motivos são os mais diversos, pode ser a falta de alguma biblioteca, um erro de configuração em algum lugar, uma atualização do código-fonte onde algum desenvolvedor deixou passar algo despercebido, enfim, estes são problemas comuns encontrados durante a compilação do MAME.

Caso o processo tenha parado, nos terminais linux e no MSYS2 clique na tecla cima do teclado para repetir o comando anterior seguido de **Enter**. Geralmente a compilação continua sem maiores problemas, porém, caso pare novamente no mesmo lugar, pode haver algum outro problema que exija a intervenção de um desenvolvedor.

Se a versão que esteja tentando compilar seja a GIT, aguarde algumas horas ou, um dia inteiro até que os desenvolvedores resolvam o problema. Nestes casos não é necessário reportar o erro, pois o código fonte do MAME no GIT é atualizado a todo instante.

- Para que o código-fonte do MAME possa ser compilado, há toda uma estrutura que precisa estar configurada no momento que o comando **make** é executado, incluindo diversos outros parâmetros de compilação. Sempre que um novo parâmetro for adicionado ou removido, quando o código-fonte de um driver for adicionado, atualizado, renomeado, removido e assim por diante, todos os arquivos do projeto responsáveis pela compilação precisam ser atualizados através da opção **REGENIE=1**.
- Durante o processo de compilação são gerados arquivos objetos *.o, arquivos de arquivamento *.a dentre vários outros, é importante que seja feito um **make clean** sempre após uma atualização do código fonte do MAME com o comando `git pull` quando for fazer uma *compilação cruzada* ou quando for personalizar uma compilação.

Esta opção faz com que todo o diretório **build** seja apagado, este diretório nada mais é do que um espaço auxiliar usado pelo processo de compilação.

É possível atualizar o código-fonte com o comando `git pull` seguido de um `make REGENIE=1` para compilar apenas os novos códigos fontes que foram adicionados, atualizados, etc; e aproveitar os arquivos já compilados, porém, algumas vezes isso pode causar erros de compilação. É uma boa prática fazer um **make clean** antes do **make** para evitar qualquer residual das compilações anteriores.

- Use dois comandos em sequência com **&&** como é mostrado abaixo:

```
make clean && make <opções>
```

Assim faz com que o segundo comando apenas seja executado quando o primeiro terminar. Caso a compilação pare por algum erro, tente repetir apenas o comando **make**.

- As opções usada pelo make podem ser adicionadas num arquivo **useroptions.mak**. Muito útil em casos onde a lista de opções para a compilação são grandes e repetitivas, dentro do arquivo as opções se organizam da seguinte maneira:

```
OPÇÃ01=X
OPÇÃ02=Y
OPÇÃ03=Z
```

Onde X, Y ou Z são os valores das opções usadas independente para cada tipo de opção, como por exemplo `SSE2=1` que irá se beneficiar das propriedades do seu processador caso ele seja compatível com as extensões **SSE2** e assim por diante.

- O MAME acompanha algumas ferramentas adicionais que poderão ser úteis em algum momento, caso queira que tais ferramentas também sejam compiladas junto com o MAME, adicione a opção `TOOLS=1`. Para mais informações sobre cada uma dessas ferramentas e de como usá-las, veja *FERRAMENTAS ADICIONAIS*.
- Nas versões compiladas do git (versão GIT), a versão do MAME acompanha um identificador único depois da versão, por exemplo:

```
./mame -help
MAME v0.205 (mame0205-540-gc8e4dab20c)
```

Onde:

mame0205 - É a versão atual do MAME.

540 - Indica a quantidade de **commits** ou a quantidade de atualizações aplicadas ao código-fonte desde a última mudança de versão.

gc8e4dab20c - São os primeiros 10 dígitos do último **commit**.

- O git mantém um controle de todos os arquivos do código-fonte, qualquer alteração que não tenha sido feita pelos administradores a versão do seu MAME incluirá um identificador **dirty** no final:

```
./mame -help
MAME v0.205 (mame0205-540-gc8e4dab20c-dirty)
```

O problema ocorre também caso exista algum residual antigo vindo de outra compilação, de não fazer um `make clean` antes de uma nova compilação, **arquivos não rastreados** dentro do diretório de trabalho do código-fonte ou até mesmo quando há arquivos alterados que por algum motivo não foram aplicados, exemplo:

```
git status --short

M bgfx/shaders/essl/chains/crt-geom/fs_crt-geom-deluxe.bin
M bgfx/shaders/essl/chains/crt-geom/fs_crt-geom.bin
...
?? language/Afrikaans/strings.mo
?? language/Albanian/strings.mo
...
```

A letra **M** indica que o arquivo foi alterado, já **??** indica os novos arquivos que foram criados. Independente do que tenha acontecido, execute `git commit -a` para aplicar estas alterações.

Agora ao pedir o status do git, ele deve retornar informando que está tudo limpo:

```
git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
```

Caso não funcione, execute a opção abaixo com todos os arquivos que vierem a aparecer ao fazer um **git status**:

```
git checkout 3rdparty/winpcap/Lib/libpacket.a 3rdparty/winpcap/Lib/libwpcap.a
```

Caso nenhum dos comando acima funcione e depois de ter absoluta certeza de que nada tenha sido alterado, experimente o comando `git clean -d -x -f`, note que o comando vai apagar tudo o que não seja relacionado com o código-fonte do MAME, isso inclui o seu **useroptions.mak** ou qualquer outro arquivo que ali esteja. Portanto, faça um **backup** antes de executar o comando!

Vamos supor que o arquivo abaixo tenha sido alterado por qualquer motivo:

```
git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git checkout -- <file>..." to discard changes in working directory)

    modified:   scripts/src/osd/sdl_cfg.lua

no changes added to commit (use "git add" and/or "git commit -a")
```


Execute o comando abaixo para restaurá-lo ao seu estado original:

```
git checkout master -- scripts/src/osd/sdl_cfg.lua
```

Acelerando uma compilação

Compilar todo o código-fonte do MAME é um processo demorado e que consome muitos recursos de processamento, memória e principalmente energia elétrica. É possível acelerar todo este processo usando o **ccache**, este programa armazena uma cópia da sua compilação, fazendo com que apenas o código-fonte que foi atualizado seja compilado, todo o resto vem do armazenamento que o **ccache** fazendo com que a compilação termine num tempo muito menor, estamos falando em compilar todo o código-fonte do MAME em segundos com o **ccache**, sem ele, uma compilação pode levar horas.

Para sistemas **Ubuntu** e **Debian Linux** o comando para instalar o **ccache** é `sudo apt-get install ccache`, para **Arch Linux** e **MSYS2** o comando é `pacman -s ccache`, veja qual é a opção para o seu sistema operacional.

A configuração é muito simples, basta usá-lo antes dos compiladores, é mais fácil adicionar essas opções no arquivo **useroptions.mak** assim não é necessário usar uma linha muito grande de configuração, para o Linux a configuração ficaria assim:

```
# Escolha apenas uma opção para OVERRIDE_CC e OVERRIDE_CXX
# Remova o # da frente da opção que deseja usar.
#
# Compila com ccache Linux
OVERRIDE_CC=/usr/bin/ccache gcc
OVERRIDE_CXX=/usr/bin/ccache g++
#
# Compila com ccache Linux (Clang)
# CCACHE_CPP2=yes
# OVERRIDE_CC=/usr/bin/ccache /usr/bin/clang
# OVERRIDE_CXX=/usr/bin/ccache /usr/bin/clang++
```

A configuração para Windows no MSYS2 fica assim:

```
# Compila com ccache MSYS2 (Windows) 32-Bit
# OVERRIDE_CC=/mingw32/bin/ccache /mingw32/bin/gcc
# OVERRIDE_CXX=/mingw32/bin/ccache /mingw32/bin/g++
#
# Compila com ccache MSYS2 (Windows) 64-Bit
# OVERRIDE_CC=/mingw64/bin/ccache /mingw64/bin/gcc
# OVERRIDE_CXX=/mingw64/bin/ccache /mingw64/bin/g++
#
# Compila com ccache MSYS2 (Windows) 64-Bit (Clang)
# OVERRIDE_CC=/mingw64/bin/ccache /mingw64/bin/clang
# OVERRIDE_CXX=/mingw64/bin/ccache /mingw64/bin/clang++
```

Para ver a condição do armazenamento cache faça `ccache -s`:

cache directory		/home/mame/.ccache
primary config		/home/mame/.ccache/ccache.conf
secondary config	(readonly)	/etc/ccache.conf
Cacheable calls:	18137 / 18247	(99.40%)
Hits:	108 / 18137	(0.60%)
Direct:	98 / 108	(90.74%)
Preprocessed:	10 / 108	(9.26%)
Misses:	18029 / 18137	(99.40%)
Uncacheable calls:	110 / 18247	(0.60%)
Local storage:		
Cache size (GiB):	1.2 / 5.0	(24.71%)
Hits:	108 / 18137	(0.60%)
Misses:	18029 / 18137	(99.40%)

Antes de usar tenha certeza que a variável de ambiente `USE_CCACHE` exista e seja igual à **1**, caso não exista, defina com `export USE_CCACHE=1` antes da compilação ou salve no arquivo `~/.bashrc` como já foi descrito em [Preparando a instalação do MSYS2 manualmente](#).

Para montar a sua cache basta fazer uma compilação limpa do código-fonte do MAME com `rm -rf build/*` && `make -j7`, no final em **cache size** deve aparecer o quanto foi armazenado em cache. Para aumentar o **max cache size** edite o arquivo `/home/mame/.ccache/ccache.conf`.

Para que o **ccache** funcione é **obrigatório** manter exatamente a mesma configuração usada para gerar o cache, caso contrário o **ccache** vai gerar um novo cache para essa nova configuração e assim por diante.

Veja todas as opções do **ccache** com o comando `ccache -h`.

Caso precise usar uma nova opção para a compilação, elimine o cache antigo com o comando `ccache -C` e faça uma nova compilação limpa com todas as suas novas opções, por experiência, isso tende a manter a rapidez da compilação, alternar opções a todo o momento tende a inflar o cache e deixar as coisas mais lentas.

Exemplos práticos para todas as plataformas

O código-fonte do MAME já vem preparado de forma que seja possível compilar toda a estrutura ou apenas uma parte dela como arcades por exemplo, consoles, portáteis ou até mesmo um sistema em específico como Neo Geo, CPS1, CPS2; assim como consoles como Megadrive/Genesis, Super Nintendo, Playstation e assim por diante.

Isso é útil quando temos que lidar com limitações de tamanho do arquivo final do MAME ou caso queira apenas uma versão do MAME bem específica.

Para compilar a versão completa do MAME faça o comando:

make

Caso o seu processador tenha 5 núcleos, é possível usar os núcleos extras do seu processador para ajudar a reduzir o tempo de compilação com a opção `-j`. Observe que a quantidade máxima de núcleos disponíveis fica limitado a quantidade de núcleos que o seu processador tiver mais um.

Usando valores acima da quantidade de núcleos do seu processador não faz com que a compilação fique mais rápida, além disso, a sobrecarga extra de processamento pode fazer com que seu processador superaqueça, seu computador pode ficar mais lento, pare de responder, etc. No caso específico de compilação no Windows, a sobrecarga tira todos os benefícios da compilação em paralelo, nos testes realizados com Windows 10 64-bit o valor ideal foi a quantidade de núcleos `-1` ou seja, num processador com 8 núcleos o valor ideal é 7.

make -j7

Para compilar o MAME junto com as *ferramentas*, use a opção abaixo:

make TOOLS=1 -j7

Para incluir os símbolos de depuração na compilação, use a opção **SYMBOLS=1**, opção útil caso o MAME trave por algum motivo. Para mais informações veja *SYMBOLS*. É importante também adicionar o nível destes símbolos, para mais informações veja *SYMLEVEL*. Seja qual for a versão do MAME que esteja compilando, é uma boa prática manter ambas as opções em todas elas. Observe que ao compilar a versão completa do MAME com os símbolos embutidos no próprio executável extrapola o tamanho máximo do executável permitido pelo Windows, motivo este que os símbolos precisam ser extraídos do executável.

make TOOLS=1 SYMBOLS=1 SYMLEVEL=1 -j7

Para compilar uma versão de depuração do MAME use o comando abaixo, para mais informações veja *DEBUG*:

make TOOLS=1 SYMBOLS=1 SYMLEVEL=1 DEBUG=1 -j7

É possível customizar a sua compilação escolhendo um driver em específico usando a opção *SOURCES=<sistema>*, lembrando que é obrigatório usar a opção **REGENIE=1** para regenerar os arquivos do projeto caso exista uma compilação anterior, a opção **REGENIE=1** não é necessário caso você faça um *make clean* antes. Caso queira compilar uma versão customizada do MAME que só rode o jogo **Pac Man**, use o comando abaixo:

make SOURCES=src/mame/pacman/pacman.cpp REGENIE=1 -j7

O MAME também permite de maneira prática que seja possível compilar uma versão só com sistemas ARCADE, nessa versão os portáteis, consoles, computadores, dentre outras ficam de fora. Caso queira uma versão arcade do MAME use o comando abaixo:

make SUBTARGET=arcade SYMBOLS=1 SYMLEVEL=1 -j7

Para compilar uma versão do MAME só com consoles, use o comando abaixo:

make SUBTARGET=mess SYMBOLS=1 SYMLEVEL=1 -j7

Para compilar uma versão do MAME chamada **meumame** apenas com a família de sistemas inclusas em *Pac-Man* e *Galaxian* incluindo as ferramentas:

make SUBTARGET=meumame SOURCES=src/mame/pacman/pacman.cpp,src/mame/galaxian/galaxian.cpp TOOLS=1 REGENIE=1 -j7

Caso encontre erros de linkagem dos arquivos estáticos da compilação após a alteração das fontes, exclua estes arquivos do diretório utilizado para a compilação do seu *subtarget*. No exemplo acima, seria o diretório *build/mingw-gcc/bin/x64/Release/meumame*.

Para compilar uma versão do Apple II compilando até seis arquivos fonte em paralelo faça:

make SUBTARGET=appulator SOURCES=apple/apple2.cpp,apple/apple2e.cpp,apple/apple2gs.cpp REGENIE=1 -j7

Para compilar uma versão do MAME que tire proveito da extensão SSE2 do seu processador melhorando o desempenho, use o comando abaixo. Para mais informações veja *SSE2*:

make TOOLS=1 SYMBOLS=1 SYMLEVEL=1 SSE2=1 -j7

É possível compilar o MAME usando todas as extensões disponíveis do seu processador e não apenas a SSE2 desde que seja também compatível com o compilador que estiver usando, use a opção **ARCHOPTS** com **-march=native** no seu comando de compilação. Ao ativar estas opções pode ou não tirar o máximo de desempenho possível do seu processador, assim como o MAME pode ou não se beneficiar de todas elas. O comando completo então ficaria assim, note que a opção **SSE2=1** foi removida:

make SYMBOLS=1 SYMLEVEL=1 ARCHOPTS=-march=native -j7

O ponto negativo é que os binários gerados com essa opção só irão funcionar em processadores iguais ao seu, caso compile uma versão num processador i3 da Intel, essa versão não vai funcionar em qualquer outro processador i7 por exemplo, o mesmo vale para os processadores da AMD. Assim como ao ativar estas extensões o seu MAME pode apresentar algum problema que não existe na versão oficial, logo, a sua sorte com o uso dela pode variar bastante. Por isso saiba que oficialmente os desenvolvedores do MAME **não apoiam** o uso dessa opção.

Execute o comando abaixo para saber quais as extensões serão ativadas com a opção **-march=native**:

```
gcc -march=native -Q --help=target|grep enabled
```

Dependendo do modelo do processador o comando retornará mais ou menos extensões disponíveis, num processador AMD FX(tm)-8350 com 8 núcleos o **-march=native** vai usar estas extensões do seu processador:

-m64	[enabled]
-m80387	[enabled]
-m96bit-long-double	[enabled]
-mabm	[enabled]
-maes	[enabled]
-malign-stringops	[enabled]
-mavx	[enabled]
-mbmi	[enabled]
-mcx16	[enabled]
-mf16c	[enabled]
-mfancy-math-387	[enabled]
-mfentry	[enabled]
-mfma	[enabled]
-mfma4	[enabled]
-mfp-ret-in-387	[enabled]
-mfxsr	[enabled]
-mglibc	[enabled]
-mhard-float	[enabled]
-mieee-fp	[enabled]
-mlong-double-80	[enabled]
-mlwp	[enabled]
-mlzcnt	[enabled]
-mmmx	[enabled]
-mpclmul	[enabled]
-mpopcnt	[enabled]
-mprfchw	[enabled]
-mpush-args	[enabled]
-mred-zone	[enabled]
-msahf	[enabled]
-msse	[enabled]
-msse2	[enabled]
-msse3	[enabled]
-msse4	[enabled]
-msse4.1	[enabled]
-msse4.2	[enabled]
-msse4a	[enabled]
-mssse3	[enabled]
-mstackrealign	[enabled]
-mtbm	[enabled]

(continua na próxima página)

(continuação da página anterior)

<code>-mtls-direct-seg-refs</code>	<code>[enabled]</code>
<code>-mxop</code>	<code>[enabled]</code>
<code>-mxsave</code>	<code>[enabled]</code>

Apesar de ter todas essas extensões ativadas, incluindo outras variantes do SSE como a SSE3, SSE4 e assim por diante, não espere que o desempenho do MAME aumente de forma considerável, há sistemas onde não se nota nada de diferente, muito pelo contrário, há perda no desempenho, já outras podem lhe dar um desempenho considerável.

Em alguns testes a melhor média foi obtida usando apenas as opções **SSE3=3 OPTIMIZE=03** e mais nada, apesar do padrão do MAME ser **SSE2=1**. Novamente, essa é uma questão muito subjetiva pois depende muitas variáveis como a configuração do seu hardware por exemplo, logo a sua sorte pode variar bastante. É muito difícil saber com precisão se haverá uma melhora no desempenho ou não pois o MAME depende muito do desempenho do hardware onde ele é executado (quanto mais potente, melhor) e do sistema operacional, dos sistemas, etc.

Podemos fazer um teste prático compilando duas versões do MAME para rodar apenas o **pacman** usando opções diferentes:

```
Opção 1
make SOURCES=src/mame/pacman/pacman.cpp SUBTARGET=pacman SSE3=1 OPTIMIZE=3

Opção 2
make SOURCES=src/mame/pacman/pacman.cpp SUBTARGET=pacman ARCHOPTS=-march=native OPTIMIZE=3
```

Rodamos o nosso MAME por 90 segundos num AMD FX(tm)-8350 4 Ghz (8 núcleos), 16 GiB de memória DDR3 1866 Mhz, AMD R7 250E 1 GiB, Windows 10 x64 usando a opção *bench*:

```
pacman64.exe pacman -bench 90
```

Para a **opção 1** ele retorna:

```
Average speed: 6337.43% (89 seconds)
```

Para a **opção 2** nós temos:

```
Average speed: 6742.91% (89 seconds)
```

Agora compilando o MAME para rodar num Linux Debian 9.7 x64, usando as mesmas opções, o mesmo driver, o mesmo código-fonte e usando exatamente o mesmo hardware, nós temos um resultado bem diferente:

Para a **opção 1** nós temos:

```
Average speed: 8438.88% (89 seconds)
```

Já a **opção 2**:

```
Average speed: 8332.99% (89 seconds)
```

Ambas as versões foram compiladas usando a mesma versão do GCC **6.3.0** do Debian, uma versão foi compilada nativamente e a outra usando *compilação cruzada*. Como é possível ver nestes exemplos a questão de otimização do MAME não é uma ciência exata, apesar da versão do Linux ter levado a melhor, há casos onde dependendo do sistema escolhido, a versão do Windows leva a melhor, assim como também há casos onde há um empate técnico, nenhum dos dois levam vantagens significativas.

Para aqueles que se interessarem por benchmarks, [aqui tem um site](#) interessante que publica de tempos em tempos e inclusive uma versão diária do GIT, uma comparação com diferentes sistemas e diferentes versões do MAME.

Use estas opções em conjunto com o comando make ou definindo-as como variáveis de ambiente ou ainda adicionando-as ao seu **useroptions.mak**. Note que o GENie não reconstrói automaticamente os arquivos afetados por modificações posteriormente usadas.

Com o tempo e experiência, cada um irá adaptar as opções de compilação para a sua própria necessidade, no exemplo abaixo tem um template para o seu **useroptions.mak**:

```
# Template de configuração do usuário para a compilação do MAME.
# Altere as opções conforme a sua necessidade. Remova o # da frente
# da opção que deseja usar.
#
# Para compilações que usem o Clang
# <- Clang ->
#OVERRIDE_CC=/usr/bin/clang
#OVERRIDE_CXX=/usr/bin/clang++
#
# Só use em ÚLTIMO CASO! Para depuração apenas!
#~SANITIZE=address
#<- Clang ->
#
# Para compilar o MAME com apenas uma maquina em especifico.
#SOURCES=src/mame/neogeo
#
# Para incluir símbolos de depuração (obrigatório)
SYMBOLS=1
SYMLEVEL=1
#
# <- Compilação cruzada ->
# Para compilar o MAME para o Windows usando o Linux por exemplo.
#TARGETOS=windows
#STRIP_SYMBOLS=1
# Use a opção abaixo para compilar uma versão 64-bit do MAME, não
# precisa ser definido para compilações normais.
#PTR64=1
#
# <- Compilação cruzada ->
#
# Caso queira compilar uma versão tiny apenas para teste.
#SUBTARGET=tiny
#
# Caso queira uma versão ARCADE do MAME
#SUBTARGET=arcade
#
# <- Opções Relacionados com a CPU ->
# SSE2
SSE2=1
#
# SSE3
#SSE3=1
#
# Nível de otimização.
# 0 Desativa a otimização favorecendo a depuração.
# 1 Otimização simples sem impacto direto no tamanho final do executável.
# 2 Ativa a maioria das otimizações visando desempenho e tamanho reduzido.
```

(continua na próxima página)

(continuação da página anterior)

```
# 3 Máxima otimização ao custo de um tamanho final maior. (padrão)
# s Ativa apenas as otimizações que não impactem no tamanho final.
OPTIMIZE=3
#
# Essa opção ativa todas as extensões do seu processador, se for usar
# não use as opções SSE2 e SSE3.
#ARCHOPTS=-march=native
# <- Opções Relacionados com a CPU ->
#
# Nova configuração
EMULATOR=1
```

Com o arquivo acima configurado e com as opções definidas, execute o comando `make -j7` que o seu MAME será compilado levando as suas opções em consideração. Uma nova configuração foi [adicionada em 31/12/2023](#) consulte [EMULATOR](#).

A próxima seção resume algumas das opções úteis reconhecidas pelo makefile.

3.5.2 Microsoft Windows

O MAME para Windows é compilado usando o ambiente MSYS2. Será necessário o Windows 7 ou mais recente e uma instalação atualizada do MSYS2. Recomendamos veementemente que o MAME seja compilado num sistema 64-bit, talvez seja necessário fazer ajustes para que a compilação funcione com sistemas 32-bit.

Configurando o pacote MSYS2 já pronto

- Baixe o pacote de instalação do MSYS2 já pronto contendo todas as ferramentas necessárias para a compilação do MAME em [MAME Build Tools](#).
- Descompacte em algum lugar, entre no diretório, abra o shell do MSYS2 (**mingw64.exe**) e aguarde ele terminar a sua configuração.

Apesar da recomendação para atualizar as ferramentas na [documentação oficial](#) a experiência mostra que algumas vezes essa atualização acaba quebrando a compilação do MAME de alguma maneira, veja por exemplo [este exemplo](#), portanto, prefira manter a ferramenta oficial sem atualizações a não ser que seja extremamente necessário.

Caso encontre algum problema veja [Resolvendo possíveis problemas com o MSYS2](#). Ao final do processo, execute a sequência de comandos abaixo:

1. `git config --global core.autocrlf true`
2. `mkdir /src`
3. `cd /src`
4. `git clone https://github.com/mamedev/mame.git`

O último comando irá baixar todo o código-fonte do MAME para um diretório chamado **mame**, o caminho completo é `/src/mame`.

- Por predefinição o MAME será compilado usando interfaces nativas do Windows como gerenciamento de janelas, saída de áudio e vídeo, renderizador de fontes, etc. Em vez disso, caso queira compilar o MAME usando o SDL (Simple DirectMedia Layer), adicione a opção `OSD=sd1` nas opções de compilação do make. É necessário que seja instalado os pacotes de desenvolvimento do SDL 2 no MSYS2 da versão **2.0.14** ou mais recente.

Caso queira compilar uma versão SDL (Simple DirectMedia Layer) do MAME para Windows em vez da versão nativa, instale os pacotes SDL com o comando:

Para versões **x64**

```
pacman -S mingw-w64-x86_64-SDL2 mingw-w64-x86_64-SDL2_ttf
```

Para versões **x32**

```
pacman -S mingw-w64-i686-SDL2 mingw-w64-i686-SDL2_ttf
```

- Por predefinição o MAME incluirá a versão nativa do depurador para Windows, para que também seja incluída a versão Qt do depurador, é necessário instalar os pacotes de desenvolvimento do Qt versão 5 no MSYS2 e depois usar `QTDEBUG=1` nas opções de compilação do make.

Preparando a instalação do MSYS2 manualmente

A versão nativa do MAME para Windows é compilada usando o ambiente de desenvolvimento MSYS2, é necessário ter o Windows 7 ou mais recente assim como uma versão atualizada do MSYS2. É aconselhável compilar o MAME num sistema operacional de 64-bit, para sistemas 32-bit é necessário fazer algumas alterações. Baixe e instale o ambiente de desenvolvimento MSYS2 direto da página do [MSYS2](#).

Por fim é necessário definir as variáveis MINGW32 e MINGW64, instale o editor de texto nano com o comando `pacman -S nano`, após a instalação faça `nano ~/.bashrc` e adicione a linha abaixo no final do arquivo:

```
export MINGW32=/mingw32 MINGW64=/mingw64
```

Salve o arquivo com **CTRL+O** seguido de **ENTER** e faça **CTRL+X** para sair do editor, essas variáveis de ambiente permitem a compilação das versões 32-bit e 64-bit do MAME. Feche e abra o terminal novamente para que essas configurações sejam aplicadas.

Caso ocorra algum erro do tipo **GPGME error**, veja [Resolvendo possíveis problemas com o MSYS2](#). Ao final, **feche a janela** e reinicie o `mingw64.exe`.

- Instale os primeiros pacotes necessários para compilar o MAME com o comando.

```
pacman -S bash git make
```

- Para as versões **64-bit** do MAME é necessário instalar os pacotes:

```
pacman -S mingw-w64-x86_64-gcc mingw-w64-x86_64-python
```

- Para as versões **32-bit** do MAME é necessário instalar os pacotes:

```
pacman -S mingw-w64-i686-gcc mingw-w64-i686-python
```

- Para linkar usando o LLVM linker (é geralmente mais rápido que a versão do GNU linker), instale o pacote `mingw-w64-x86_64-llvm` e o `mingw-w64-x86_64-llvm-libs` para as versões 64-bit ou o pacote `mingw-w64-i686-llvm` e o `mingw-w64-i686-llvm-libs` para as versões 32-bit. Para mais informações consulte [Linkando com o LLVM linker](#).

- Para compilar usando as interfaces portáteis do SDL **64-bit** é necessário instalar os pacotes:

```
pacman -S mingw-w64-x86_64-sdl mingw-w64-x86_64-sdl-ttf
```

- Para compilar usando as interfaces portáteis do SDL **32-bit** é necessário instalar os pacotes:

```
pacman -S mingw-w64-i686-sdl mingw-w64-i686-sdl-ttf
```

- Para compilar o MAME com o depurador Qt **64-bit** é preciso instalar o pacote:

```
pacman -S mingw-w64-x86_64-qt5
```

- Para compilar o MAME com o depurador Qt **32-bit** é preciso instalar o pacote:

```
pacman -S mingw-w64-i686-qt5
```

Nota: Utilize `QTDEBUG=1` nas opções de compilação do `make` para compilar a interface QT do depurador.

- Para gerar a documentação API do código-fonte é preciso instalar o pacote **doxygen**.
- Para fazer a depuração do MAME é necessário instalar o **gdb**. Para mais informações sobre o `gdb` veja [Usando o gdb](#).

É possível também utilizar estes comandos para garantir que todos os pacotes necessários para compilar o MAME estejam disponíveis no seu sistema, omita aqueles cuja configuração você não planeja utilizar para compilar ou combine diversos comandos **pacman** para instalar mais de um pacote de uma vez:

```
pacman -Syu
pacman -S curl git make
pacman -S mingw-w64-x86_64-gcc mingw-w64-x86_64-libc++ mingw-w64-x86_64-llvmlibc mingw-w64-x86_64-  
↳python
pacman -S mingw-w64-x86_64-SDL2 mingw-w64-x86_64-SDL2_ttf
pacman -S mingw-w64-x86_64-qt5
pacman -S mingw-w64-i686-gcc mingw-w64-i686-libc++ mingw-w64-i686-llvmlibc mingw-w64-i686-python
pacman -S mingw-w64-i686-SDL2 mingw-w64-i686-SDL2_ttf
pacman -S mingw-w64-i686-qt5
```

Resolvendo possíveis problemas com o MSYS2

Em caso de erro do tipo **error: GPGME error: Invalid crypto engine** que faz com que a atualização pare, verá que na internet há diversos tópicos em centenas de diferentes fóruns sobre o assunto e praticamente nenhuma solução na prática, então aqui vai a dica para este erro em específico, caso apareçam outros, este documento será atualizado.

Edite o arquivo `/etc/pacman.conf` e mude **SigLevel = Required DatabaseOptional** para **SigLevel = Never** e salve, mantenha a tela do seu editor aberto. Vá até o diretório `/etc/pacman.d` e apague o diretório **gnupg**.

Abra o shell do MSYS2 (**mingw64.exe**) e digite os comandos abaixo nesta sequência:

1. `pacman-key --init`
2. `pacman-key --populate msys2`
3. `pacman-key --refresh-keys`

A atualização agora pode prosseguir com o comando `pacman -Syu`, caso os passos acima tenham sido seguidos corretamente, haverá um retorno semelhante ao que é mostrado abaixo:

```
$ pacman -Syu
:: Sincronizando a base de dados de pacotes...
mingw32 está atualizado
mingw64 está atualizado
msys está atualizado
mame está atualizado
:: Starting core system upgrade...
não há nada a fazer
:: Iniciando atualização completa do sistema...
resolvendo dependências...
procurando por pacotes conflitantes...

Pacotes (69) bash-completion-2.8-2 brotli-1.0.7-1 bsdcpio-3.3.3-3
      bsdtar-3.3.3-3 ca-certificates-20180409-1 coreutils-8.30-1
      curl-7.63.0-1 dash-0.5.10.2-1 dtc-1.4.7-1 file-5.35-1
      gawk-4.2.1-2 gcc-libs-7.4.0-1 glib2-2.54.3-1 gnupg-2.2.12-1
      grep-3.0-2 heimdal-libs-7.5.0-3 icu-62.1-1 info-6.5-2
      less-530-1 libarchive-3.3.3-3 libargp-20110921-2
      libassuan-2.5.2-1 libcrypt-2.1-2 libcurl-7.63.0-1
      libexpat-2.2.6-1 libffi-3.2.1-3 libgcrypt-1.8.4-1
      libgnutls-3.6.5-1 libgpg-error-1.33-1 libgpgme-1.12.0-1
      libhogweed-3.4.1-1 libidn2-2.0.5-1 libksba-1.3.5-1
      liblz4-1.8.3-1 liblzma-5.2.4-1 liblz0-2.10-2 libnettle-3.4.1-1
      libnghttp2-1.35.1-1 libnpt-1.6-1 libopenssl-1.1.1.a-1
      libp11-kit-0.23.14-1 libpcre-8.42-1 libpcre16-8.42-1
      libpcre2_8-10.32-1 libpcre32-8.42-1 libpcrecpp-8.42-1
      libpcreposix-8.42-1 libpsl-0.20.2-1 libreadline-7.0.005-1
      libsqlite-3.21.0-4 libssh2-1.8.0-2 libunistring-0.9.10-1
      libutil-linux-2.32.1-1 libxml2-2.9.8-1 m4-1.4.18-2
      ncurses-6.1.20180908-1 nettle-3.4.1-1 openssl-1.1.1.a-1
      p11-kit-0.23.14-1 pcre-8.42-1 pinentry-1.1.0-2 pkgfile-19-1
      rebase-4.4.4-1 sed-4.7-1 time-1.9-1 ttyrec-1.0.8-2
      util-linux-2.32.1-1 wget-1.20-2 xz-5.2.4-1

Tamanho total download:      36,91 MiB
```

(continua na próxima página)

(continuação da página anterior)

```
Tamanho total instalado: 206,90 MiB
Alteração no tamanho:    61,49 MiB
```

```
Continuar a instalação? [S/n]
```

Pressione "Enter" e aguarde, no final do processo é importante que siga as instruções, não saia do terminal, feche a janela e abra-a novamente. Retorne ao seu editor de texto e mude novamente **SigLevel = Never** para **SigLevel = Required DatabaseOptional**, salve o arquivo e feche o editor.

Para ter certeza de que não há nenhum erro execute o comando `pacman -Syu` novamente:

```
$ pacman -Syu
:: Sincronizando a base de dados de pacotes...
mingw32 está atualizado
mingw64 está atualizado
msys está atualizado
mame está atualizado
:: Starting core system upgrade...
não há nada a fazer
:: Iniciando atualização completa do sistema...
não há nada a fazer
```

Caso tenha recebido um retorno diferente ou tenha qualquer outro problema que o impeça de fazer a atualização, verifique se não há qualquer um [destes programas](#) instalados em seu computador, caso haja, veja se é possível desativá-los, adicionar uma regra de exclusão do diretório do MSYS2 (`c:\mysys64` ou `c:\mysys32`) ou até mesmo removê-los até que você consiga montar o seu ambiente sem problemas.

Uma outra alternativa interessante seria usar um sistema virtual para compilar o MAME ou para montar o ambiente sem qualquer erro.

Nota: A mesma dica acima serve também para resolver outro erro relacionado "chave PGP inválida" (*invalid or corrupted package (PGP signature)*). a solução foi apresentada por mim na parte de [issues do MSYS2](#) Outros usuários também comprovaram que a solução funciona.

Nota: Caso atualize a ferramenta fornecida pelo MAMEDev (incluindo o GCC e Clang), pode aparecer alguns erros sobre arquivos PCH inválidos ou não encontrados, se for o caso, use a opção `PRECOMPILE=0` na linha de comando ou adicione-o no seu `useroptions.mak`. Faça um **make clean && make -j6** para limpar a compilação antiga e compilar tudo desde o início. Caso ainda tenha problemas, aguarde mais um tempo pela atualização do repositório pois os desenvolvedores podem estar resolvendo esta questão internamente até fazerem uma nova publicação.

Compilando com o Microsoft Visual Studio

- É possível gerar projetos compatíveis com o Visual Studio 2019 usando o comando **make vs2019**. É predefinido que a solução e o projeto serão criados no diretório `build/projects/windows/mame/vs2019`. O nome do diretório **build** pode ser alterado modificando a opção `BUILDDIR`.

O comando sempre regenera as configurações, logo a opção **REGENIE=1** não é necessário.

- Usando a opção **MSBUILD=1** será construído a solução usando o *Microsoft Build Engine* após a criação dos arquivos do projeto. Observe que é necessário que o ambiente e os caminhos estejam corretamente configurados para que o Visual Studio possa encontrá-los.
- Consulte [Usando o conjunto de ferramentas Microsoft C++ na linha de comando](#). Pode ser que você ache mais fácil carregar o projeto direto na interface do Visual Studio do que usar **MSBUILD=1**.
- Ainda que o Visual Studio seja usado é necessário ter também o ambiente `MSYS2` para gerar os arquivos do projeto, converter os layouts internos, compilar as traduções da interface, etc.

Algumas observações sobre o ambiente MSYS2

O MSYS2 utiliza a ferramenta pacman do gerenciador de pacotes do Arch Linux. Existe uma página no wiki do [Arch Linux](#) com informações relevantes e que ensinam como usar a ferramenta pacman.

O ambiente MSYS2 inclui dois tipos de ferramentas: As ferramentas MSYS2 desenvolvidas para trabalhar num ambiente semelhante ao UNIX no Windows e as ferramentas MinGW que foram desenvolvidas para trabalhar em um ambiente Windows. As ferramentas do MSYS2 são instaladas no `/usr/bin` enquanto as ferramentas do MinGW são instaladas no `/mingw64/bin` ou `/mingw32/bin` sempre relativo ao diretório de instalação do MSYS2. As ferramentas do MSYS2 trabalham melhor num terminal do MSYS2 enquanto as ferramentas do MinGW trabalham melhor com o prompt de comando do Windows.

É possível notar sintomas óbvios quando você roda as ferramentas certas nos terminais errados quando não há a interatividade dos programas com as teclas direcionais por exemplo. Caso rode o MinGW gdb ou python a partir da janela do terminal do MSYS2 por exemplo, o histórico dos comandos não funcionam e é bem provável que interrompa o funcionamento dos programas anexados com o gdb. De forma similar, pode ser bem difícil editar os arquivos com o vim do MSYS2 no prompt de comandos do Windows.

O MAME é compilado usando o compiladores do MinGW, logo, os diretórios do MinGW são inclusos anteriormente no ambiente de compilação através do PATH. Caso queira utilizar um programa interativo do MSYS2 a partir de um shell MSYS2, pode ser que seja necessário informar os caminhos completo para evitar a utilização das ferramentas equivalentes do MinGW.

O gdb do MSYS2 podem ter problemas para depurar programas MinGW como o MAME. É possível obter melhores resultados ao instalar a versão do gdb do MinGW e rodá-lo a partir do prompt de comandos do Windows para depurar o MAME.

O GNU make é compatível com shells de ambos os estilos POSIX (como o bash por exemplo) e o `cmd.exe` da Microsoft. Há um problema a ser levado em consideração ao utilizar o `cmd.exe` da Microsoft pois comando `copy` não verbaliza nada muito útil durante a condição da sua ação, assim as operação de cópia são geralmente silenciosas. Prefira o uso de ferramentas como o [robocopy](#) que garante a integridade do arquivo do destino e gera um relatório completo.

Não é possível realizar a compilação cruzada de uma versão 32-bit do MAME utilizando ferramentas 64-bit do MinGW no Windows pois causa problemas devido ao tamanho do MAME, portanto, as ferramentas 32-bit do MinGW devem ser utilizadas. Não é possível linkar uma versão completa do MAME 32-bit incluindo as versões SDL e o depurador Qt. Ambos os GNU `ld` e o `ldd` ficarão sem memória gerando um arquivo final que não funciona. Também não é possível compilar uma versão 32-bit com todos os símbolos. O GCC pode ficar sem memória e certos arquivos de código-fonte podem extrapolar o limite de **32.768** seções impostas pelo formato PE/COFF do objeto.

3.5.3 Linux

Fedora Linux

Alguns pré-requisitos precisam ser atendidos na sua distro antes de continuar. As versões anteriores ao SDL 2 versão **2.0.14** não possuem a funcionalidade necessária, certifique-se que a versão mais recente esteja instalada:

```
sudo dnf install gcc gcc-c++ make python SDL2-devel SDL2_ttf-devel libXi-devel libXinerama-
↳devel qt5-qtbase-devel qt5-qttools expat-devel fontconfig-devel alsa-lib-devel pulseaudio-
↳libs-devel
```

Caso queira utilizar ferramentas LLVM mais eficientes para arquivar as bibliotecas estáticas e linkagem, você precisará instalar os seguintes pacotes:

```
sudo dnf install lld llvm
```

A compilação é exatamente como descrito em [Exemplos práticos para todas as plataformas](#).

Debian e Ubuntu (incluindo dispositivos Raspberry Pi e ODROID)

Alguns pré-requisitos precisam ser atendidos na sua distro antes de continuar. As versões anteriores ao SDL 2 versão **2.0.14** não possuem a funcionalidade necessária, certifique-se que a versão mais recente esteja instalada:

```
sudo apt-get install git git-lfs build-essential python3 libxi-dev libsdl2-dev libsdl2-ttf-
↳dev libfontconfig-dev libpulse-dev qtbase5-dev qtchooser qt5-qmake qtbase5-dev-tools
```

Caso queira utilizar ferramentas LLVM mais eficientes para arquivar as bibliotecas estáticas e linkagem, você precisará instalar os seguintes pacotes:

```
sudo apt-get install lld llvm
```

A compilação é exatamente como descrito em [Exemplos práticos para todas as plataformas](#)

Arch Linux

Alguns pré-requisitos precisam ser atendidos na sua distro antes de continuar.

```
sudo pacman -S base-devel git sdl2_ttf python libxinerama libpulse alsa-lib qt5-base libxi_
↳libpulse
```

A compilação é exatamente como descrito em [Exemplos práticos para todas as plataformas](#)

3.5.4 Apple macOS

Alguns pré-requisitos são necessários. Certifique-se que você esteja usando o macOS 11.0 Big Sur ou posterior. Você precisará do SDL 2 versão **2.0.14** ou mais recente. Também será necessário instalar o Python 3, atualmente, ele está incluído nas ferramentas de linha de comando do Xcode, mas você também pode instalar uma versão autônoma ou obtê-la através do gerenciador de pacotes Homebrew.

- Instale o **Xcode** encontrado no Mac App Store ou o [ADC](#) (é preciso ter o AppleID).
- Para localizar a última versão do Xcode correspondente para a versão do seu macOS visite [xcodereleases.com](#).
- Inicie o programa **Xcode**.
- Será feito o download de alguns pré-requisitos adicionais. Deixe rodando antes de continuar.
- Ao terminar saia do **Xcode** e abra uma janela do **Terminal**
- Digite o comando `xcode-select --install` para instalar o kit obrigatório de ferramentas para compilar o MAME (também disponível como pacote no ADC).

Em seguida, é preciso baixar e instalar o SDL 2.

- Vá para [este site](#) e baixe o arquivo .dmg para o macOS.
- Caso o arquivo .dmg não abra sozinho de forma automática, execute-o manualmente.
- Clique no 'Macintosh HD' (o HD do seu Mac), no painel esquerdo onde está localizado a janela do **Finder**, abra a pasta **Biblioteca** e arraste o arquivo **SDL2.framework** na pasta **Frameworks**. Será preciso se autenticado com a senha do seu usuário.

Caso ainda não tenha, obtenha o Python 3 e configure:

- Vá até o site oficial do Python, navegue nos [downloads para o macOS](#), clique no link para fazer o download do instalador para a última versão estável (até o momento da atualização deste texto, seria o [Python 3.10.4](#)).
- Role para baixo até a seção "Files" e baixe a versão do macOS (chamado "macOS 64-bit universal2 installer" ou similar).
- Depois de baixado, execute e siga as instruções de instalação.

Use o Terminal para iniciar a compilação, navegue até onde está o código-fonte do MAME (comando `cd`) e siga as instruções normais de compilação acima para [todas as plataformas](#).

3.5.5 Javascript Emscripten e HTML

Primeiro, baixe e instale o **Emscripten 2.0.25** ou mais recente segundo as instruções no [site oficial](#).

Depois de instalar o *Emscripten*, será possível compilar o MAME direto, usando a ferramenta '**emmake**'. O MAME completo é muito grande para ser carregado numa página web de uma só vez, então é preferível que seja compilado versões menores e separadas do MAME através do parâmetro **SOURCES**, por exemplo, faça o comando abaixo no mesmo diretório do MAME:

```
emmake make SUBTARGET=pacman SOURCES=src/mame/pacman
```

O parâmetro *SOURCES* deve apontar para pelo menos um arquivo *driver *.cpp*. O comando *make* tentará localizar e reunir todas as dependências para compilar o executável do MAME junto com o *driver* definido. No entanto porém, caso ocorra algum erro e o processo não encontre algum arquivo, é necessário declarar manualmente um ou mais arquivos que faltam (separados por vírgula). Por exemplo:

```
emmake make SUBTARGET=apple2e SOURCES=src/mame/apple2e,src/devices/machine/applefdc.cpp
```

O valor do parâmetro *SUBTARGET* serve apenas para se diferenciar dentre as várias compilações existente e não precisa ser definido caso não seja necessário.

O *Emscripten* oferece suporte à compilação do *WebAssembly* com um *loader* de *JavaScript* em vez do *JavaScript* inteiro, esse é o padrão nas versões mais recentes. Para impor a ativação ou não do *WebAssembly*, adicione *WEBASSEMBLY=1* ou *WEBASSEMBLY=0* ao comando *make*, respectivamente.

Outros parâmetros para o *make* também poderão ser usados assim como foi o *-j* para fazer o uso da compilação em *multithread*.

Nota: Ao pé da letra, *thread*, significa cordão ou linha. Na computação uma *thread* são diversas tarefas realizadas dentro de um processo, por exemplo, ao rodar o MAME você inicia um processo, dentro deste processo várias "linhas" (*threads*) são criadas onde cada uma delas serão lidas e processadas pelo processador.

Então *multithread* é a capacidade do processador e do sistema operacional de organizar e processar diferentes processos de forma independente e ao mesmo tempo.

Quando a compilação atinge a fase da *emcc*, será exibido uma certa quantidade de mensagens de aviso do tipo "*unresolved symbol*". Até o presente momento, isso é esperado para funções relacionadas com o OpenGL como a função "*glPointSize*". Outros podem também indicar que um arquivo de dependência adicional precisa ser especificado na lista *SOURCES*. Infelizmente, este processo ainda não é automatizado sendo necessário localizar e informar o arquivo de código-fonte, assim como, os arquivos que contém os símbolos que estão faltando. Pode ser que ignorar os avisos e dar sequência na compilação funcione, desde que os códigos ausentes não sejam usados no momento da execução.

Se tudo correr bem, um arquivo. *js* será criado no diretório. Este arquivo não pode ser executado sozinho, ele precisa de um loader HTML para que ele possa ser exibido e que seja possível também passar os parâmetros de linha de comando para o executável.

O [Projeto Emularity](#) oferece tal loader.

Existem amostras de arquivos .html nesse repositório que pode ser editado para refletir as suas configurações pessoais e apontar o caminho do seu arquivo js recém compilado do MAME. Para usar o MAME num servidor web, os arquivos abaixo são necessários:

- O arquivo .js compilado do MAME.
- O arquivo .wasm do MAME caso tenha compilado o WebAssembly.
- Os arquivos .js do pacote Emularity (loader.js, browserfs.js, etc.).
- Um arquivo .zip com as ROMs do driver que serão rodados, caso existam.
- Qualquer outro programa que queira rodar com o driver do MAME.
- Um loader do Emularity .html customizado para utilizar todos os itens acima.

Devido a restrição de segurança dos navegadores atuais, é necessário utilizar um servidor web em vez de tentar rodá-los localmente.

Caso algo dê errado e não funcione, abra o console Web do seu navegador principal e veja qual o erro que ele retorna (por exemplo, faltando alguma coisa, algum arquivo de ROM incorreto, etc.). Um erro do tipo "**ReferenceError: foo is not defined**" pode indicar que provavelmente faltou informar um arquivo de código-fonte na lista da opção **SOURCES**.

3.5.6 Opções gerais para a compilação

PREFIX_MAKEFILE

Define um arquivo *make* que será incluído no processo de compilação que tenha opções adicionais e que terá prioridade caso o mesmo seja encontrado (o nome predefinido é **useroptions.mak**). Pode ser útil caso queira alternar entre diferentes configurações de compilação de forma simples e rápida.

BUILDDIR

Define diretório usado para a compilação de todos os arquivos do projeto, códigos fonte auxiliares que são gerados ao longo da configuração, arquivos objeto e bibliotecas intermediárias. Por predefinição, o nome deste diretório é **build**.

REGENIE

Caso seja definido como **1**, faz com que toda a estrutura de instrução para a compilação do projeto seja regenerada, especialmente para o caso onde uma compilação tenha sido feita anteriormente e seja necessário alterar as configurações predefinidas anteriormente.

VERBOSE

Caso seja definido como **1**, ativa o modo loquaz, isso faz com que todos os comandos usados pela ferramenta make durante a compilação apareçam. Essa opção é aplicada instantaneamente e não precisa do comando **REGENIE**.

IGNORE_GIT

Caso seja definido como **1**, ignora o escaneamento da árvore de trabalho e não embute a revisão descritiva do git no campo da versão do executável.

TARGETOS

Define o Sistema Operacional de destino, é importante deixar claro que essa opção é desnecessária caso esteja compilando o MAME nativamente, os valores válidos são:

- android (Android)
- asmjs (Emscripten/asm.js)
- freebsd (FreeBSD)
- netbsd (NetBSD)
- openbsd (OpenBSD)
- pnacl (Native Client - PNaCl)
- linux (Linux)
- ios (iOS)
- macosx (OSX)
- windows (Windows)
- haiku (Haiku)
- solaris (Solaris SunOS)
- steamlink (Steam Link)
- rpi (Raspberry Pi)
- ci20 (Creator-Ci20)

SSE2

Double Precision Streaming SIMD Extensions, em resumo, são instruções que otimizam o desempenho em processadores compatíveis. Se definido como **1** o MAME terá um melhor desempenho segundo a [nota publicada](#) no site do MAME.

PTR64

Se definido como **1** define o tamanho do ponteiro em bit, assim sendo, gera uma versão 64-bit do executável do MAME ou 32-bit quando não for definido. Caso não haja nenhum problema durante o processo de compilação, haverá um executável do MAME chamado **mame.exe** para a versão *64-bit* ou **mame.exe** caso você tenha compilado uma versão para *32-bit*.

Usando ferramentas de compilação alternativas

OVERRIDE_CC

Define o compilador C/Objective-C avulso ou para um compilador voltado para um sistema em específico.

OVERRIDE_CXX

Define o compilador C++/Objective-C++ avulso ou para um compilador voltado para um sistema em específico.

OVERRIDE_LD

Define o comando para o lincador, caso o seu ambiente esteja corretamente configurado não é necessário lidar com ele, mesmo em compilação cruzada.

PYTHON_EXECUTABLE

Define o interpretador Python. Para compilar o MAME é necessário ter o Python versão 2.7, Python 3 ou mais recente.

CROSS_BUILD

Defina como **1** para que o lincador e o compilador fiquem isolados do sistema hospedeiro, opção obrigatória ao realizar uma *Compilação cruzada*.

OPENMP

Se definido como **1**, faz uso da *paralelização implícita* com o *OpenMP*. No MAME segundo o *FAQ oficial*, são adicionadas novas threads para aceleração de loop, trazendo um aumento de desempenho. Para fazer uso desta opção é necessário a instalação do libomp-devel ou libomp-dev dependendo da sua distribuição.

Incluindo os subconjuntos dos sistemas suportados

SUBTARGET

Define a compilação alternativa do compilador. Algumas predefinições estão disponíveis em `scripts/target/mame` ou usando os filtros dos drivers do sistema em `src/mame`. As versões alternativas criadas pelo usuário podem ser criadas usando **SOURCES** ou **SOURCEFILTER**. Os valores já predefinidos são:

- **dummy**: Compila uma versão bem simplificada do mame com apenas o driver da Coleco.
- **mame**: Compila uma versão do MAME com arcade, mess e virtual.
- **tiny**: Compila uma versão simples do MAME com alguns poucos drivers usado para testar a compilação do MAME, muito útil pois evita a obrigação de se compilar todo o código-fonte do MAME para testar apenas uma modificação feita na interface por exemplo.

O valor do parâmetro *SUBTARGET* serve também para se diferenciar dentre as várias compilações existente e não precisa ser definido sem necessidade. No exemplo do comando abaixo:

```
make REGENIE=1 SUBTARGET=neogeo SOURCES=neogeo/neogeo.cpp -j7
```

Será criado um binário do MAME com o nome **neogeo** no Linux ou **neogeo.exe** no Windows.

SOURCES

Define o arquivo com o código-fonte do driver que serão inclusos na compilação. Geralmente são usados em conjunto com a opção **SUBTARGET**. Os diferentes arquivos/pastas são separados com vírgulas.

SOURCEFILTER

Define um arquivo de filtro do driver do sistema. É usado normalmente em conjunto com a opção **SUBTARGET**. O arquivo de filtro pode definir os arquivos de origem na inclusão dos drivers do sistema e para escolher quais os drivers individuais do sistema que serão inclusos ou excluídos. Há alguns exemplos de arquivos filtro dos drivers de sistema na pasta `src/mame`.

Recursos opcionais

EMULATOR

Caso seja definido como **1**, todo o código fonte será compilado e o executável do MAME será gerado. Se for definido como **0**, apenas uma parte do código fonte será compilada sem o executável do MAME.

TOOLS

Caso seja definido como **1**, além do executável do MAME, serão compiladas ferramentas adicionais que trabalham em conjunto com o emulador como unidasm, chdman, romcmp, srcclean dentre outras. Utilize `TOOLS=1 EMULATOR=0` para compilar apenas as ferramentas.

NO_USE_PORTAUDIO

Caso seja definido como **1**, desativa a construção do módulo de saída de áudio PortAudio e sua biblioteca.

NO_USE_PULSEAUDIO

Caso seja definido como **1**, desativa a construção do módulo de saída de áudio PortAudio e sua biblioteca no Linux.

USE_WAYLAND

Caso seja definido como **1**, inclui suporte do MAME ao [Wayland](#) à saída de vídeo bgfx.

Nota: Para compilar o MAME com esta opção, é preciso instalar o pacote `libwayland-egl-backend-dev` ou equivalente no seu sistema operacional.

USE_TAPTUN

Caso seja definido como **1**, inclui o módulo de rede tap/tun, use **0** para desativar. O módulo de rede tap/tun está incluso por padrão no Windows e no Linux.

USE_PCAP

Caso seja definido como **1**, inclui o módulo de rede pcap, use **0** para desativar. O módulo de rede pcap está incluso por padrão no macOS e no NetBSD.

USE_QTDEBUG

Caso seja definido como **1**, será incluso o depurador com a interface Qt em plataformas onde a mesma não vem previamente embutida como MacOS e Windows por exemplo, defina como **0** para desativar. É obrigatório a instalação das bibliotecas de desenvolvimento Qt assim como suas ferramentas para a compilação do depurador. Todo este processo varia de plataforma para plataforma.

NOWERROR

Defina como **1** para desativar o tratamento das mensagens de aviso do compilador como erro. Talvez seja necessário em configurações minimamente compatíveis.

DEPRECATED

Defina como **0** para desativar as mensagens de aviso menos importantes/relevantes (repare que as mensagens de avisos não são tratadas como erro).

DEBUG

Defina como **1** para ativar as rotinas de verificações adicionais e diagnósticos ativando o modo de depuração. É importante que saiba que essa opção tem impacto direto no desempenho do emulador e só tem utilidade para desenvolvedores, não compile o MAME com esta opção sem saber o que está fazendo. Veja também [A tela de travamento](#).

OPTIMIZE

Define o nível de otimização. O valor predefinido é **3** onde o foco é desempenho ao custo de um executável maior no final da compilação. Há também as seguintes opções:

- **0**: Caso queira desativar a otimização e favorecendo a depuração.
- **1**: Otimização simples sem impacto direto no tamanho final do executável nem no tempo de compilação.
- **2**: Ativa a maioria das otimizações visando desempenho e tamanho reduzido.
- **3**: Este é o valor predefinido, em favor do desempenho ao custo de um executável maior.
- **s**: Ativa apenas as otimizações que não impactem no tamanho final do executável.

A compatibilidade destes valores dependem do compilador que esteja sendo usado.

SYMBOLS

Defina como **1** para ativar a inclusão de símbolos adicionais de depuração para a plataforma que o executável está sendo compilado, além dos já inclusos (muitas plataformas por predefinição já incluem estes símbolos e os nomes das respectivas funções).

SYMLEVEL

Valor numérico que controla a quantidade de detalhes nos símbolos de depuração, valores maiores facilitam a depuração ao custo do tempo de compilação e do tamanho final do executável. **SYMLEVEL=1** é usado na versão oficial do MAME e a mínima recomendada. A compatibilidade destes valores dependem do compilador que esteja sendo usado, no caso do GNU GCC e similares, estes valores são:

- **1**: Inclui tabelas numéricas e variáveis externas.
- **2**: Incluindo os itens descritos em **1**, inclui também as variáveis locais.
- **3**: Inclui também definições macros.

STRIP_SYMBOLS

Defina como **1** para que os símbolos de depuração em vez de ficarem embutidos no MAME, sejam armazenado num arquivo externo com extensão **".sym"**, este arquivo é extraído na versão do Windows. Esta opção é útil para aliviar o tamanho final do MAME já que **SYMLEVEL** com valores maiores que **1** geram uma grande quantidade de símbolos de depuração, impactando muito no tamanho final do executável.

ARCHOPTS

Opções adicionais que serão passadas ao compilador e ao linkador. Útil para a geração de códigos adicionais ou opções de interface binária de aplicação¹ como por exemplo a ativação de recursos opcionais do processador.

ARCHOPTS_C

Opções adicionais que serão passadas ao compilador durante a compilação dos arquivos de código-fonte em linguagem C.

ARCHOPTS_CXX

Opções adicionais que serão passadas ao compilador durante a compilação dos arquivos de código-fonte em linguagem C++.

ARCHOPTS_OBJC

Opções adicionais que serão passadas ao compilador durante a compilação dos arquivos de código-fonte Objective-C.

ARCHOPTS_OBJCXX

Opções adicionais que serão passadas ao compilador durante a compilação dos arquivos de código-fonte Objective-C++.

Sede das bibliotecas e framework**SDL_INSTALL_ROOT**

Diretório raiz onde se encontra a instalação dos arquivos de desenvolvimento SDL.

SDL_FRAMEWORK_PATH

Caminho onde se encontra o SDL framework.

USE_LIBSDL

Defina como **1** para usar a biblioteca SDL no destino onde o framework for predefinido.

USE_SYSTEM_LIB_ASIO

Defina como **1** caso prefira usar a biblioteca I/O assíncrona Asio C++ do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_EXPAT

Defina como **1** caso prefira usar o analisador sintático Expat XML do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_ZLIB

Defina como **1** caso prefira usar a biblioteca de compressão zlib instalada no seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_JPEG

Defina como **1** caso prefira usar a biblioteca de compressão de imagem libjpeg em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_FLAC

Defina como **1** caso prefira usar a biblioteca de compressão de áudio libFLAC em vez de usar a versão fornecida pelo MAME.

¹ No Inglês ABI ou [Application Binary Interface](#). (Nota do tradutor)

USE_SYSTEM_LIB_LUA

Defina como **1** caso prefira usar a biblioteca do interpretador Lua instalado no seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_SQLITE3

Defina como **1** caso prefira usar a biblioteca do motor de pesquisa SQLITE do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_PORTMIDI

Defina como **1** caso prefira usar a biblioteca PortMidi instalada no seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_PORTAUDIO

Defina como **1** caso prefira usar a biblioteca PortAudio do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_UTF8PROC

Defina como **1** caso prefira usar a biblioteca Julia utf8proc instalada no seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_GLM

Defina como **1** caso prefira usar a biblioteca GLM OpenGL Mathematics do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_RAPIDJSON

Defina como **1** caso prefira usar a biblioteca Tencent RapidJSON do seu sistema em vez de usar a versão fornecida pelo MAME.

USE_SYSTEM_LIB_PUGIXML

Defina como **1** caso prefira usar a biblioteca pugixml do seu sistema em vez de usar a versão fornecida pelo MAME.

3.5.7 Problemas conhecidos

Problemas relacionados com versões específicas do compilador

- O GCC 7 32-bit para Windows x86 gera erros esporádicos com alertas de acesso fora dos limites.² Use **NOWERROR=1** nas suas opções de compilação para remediar o problema e não tratar avisos como se fossem erros.
- Versões iniciais do GNU libstdc++ 6 contém uma implementação `std::unique_ptr` quebrada. Caso encontre qualquer mensagem de erro relacionado com `std::unique_ptr` é necessário a atualização do seu libstdc++ para uma versão mais recente.

Recursos do código-fonte fortify da biblioteca GNU C

A biblioteca GNU C possui opções para realizar verificações durante a compilação e verificações durante a execução, use `_FORTIFY_SOURCE` como 1 para ativar o recurso. Essa opção visa melhorar a segurança ao custo de uma pequena sobrecarga no executável. O MAME não é um programa seguro e nós não recomendamos que o MAME seja compilado com essa opção definida.

Algumas distribuições Linux como Gentoo e Ubuntu possuem versões modificadas do GNU GCC que já vem com o `_FORTIFY_SOURCE` ativado com 1. Isso gera problemas para a maioria dos projetos e não apenas para o MAME, pois afeta diretamente o desempenho do emulador, dificulta que essas verificações adicionais sejam desativadas, assim como torna difícil definir outros valores para `_FORTIFY_SOURCE` como 2 por exemplo, que ativa verificações ainda mais restritas.

Neste caso, você deve realmente pegar no pé dos mantenedores da sua distribuição preferida, deixando claro que você não quer que o GNU GCC tenha um comportamento fora do padrão.

Seria melhor que essas distribuições predefinissem essa opção em seu próprio ambiente de desenvolvimento de pacotes caso eles acreditem que de fato, tal opção seja realmente importante, em vez de obrigar a todos a usarem em todo e qualquer código-fonte que seja compilado no sistema sem necessidade.

A distribuição Red Hat faz da seguinte maneira, a opção `_FORTIFY_SOURCE` é definida apenas dentro do ambiente de compilação dos pacotes RPM e ao invés de distribuir uma versão modificada do GNU GCC.

Caso encontre erros relacionados com `bits/string_fortified.h`, verifique e tenha certeza se `_FORTIFY_SOURCE` já está definido no ambiente ou junto com **CFLAGS** ou **CXXFLAGS** por exemplo. É possível verificar o seu ambiente com o comando abaixo:

```
gcc -dM -E - < /dev/null | grep _FORTIFY_SOURCE
```

Caso `_FORTIFY_SOURCE` já esteja predefinido com um valor diferente de zero, é possível usar uma solução paliativa com `-U_FORTIFY_SOURCE`. Use em suas opções de compilação **ARCHOPTS** ou redefinindo as suas variáveis de ambiente **CFLAGS** e **CXXFLAGS**.

² Out-of-bounds access. (Nota do tradutor)

Problemas que afetam o Microsoft Visual Studio

A Microsoft introduziu uma nova versão do **XAudio2** com o Windows 8 que é incompatível com a versão incluída com o **DirectX** para as versões anteriores do Windows no nível de API. As novas versões do *Microsoft Windows SDK* incluem cabeçalhos e bibliotecas para a nova versão do XAudio2. É predefinido que a versão alvo do Windows seja definida para o Windows Vista (6.0) durante a compilação do MAME, o que impede o uso desta versão dos cabeçalhos e bibliotecas do XAudio2. Para construir o MAME com suporte ao XAudio2 usando o Microsoft Windows SDK, você deve fazer uma das seguintes ações:

- Adicione a opção `MODERN_WIN_API=1` ao make ao gerar os arquivos do projeto do Visual Studio. Isso definirá a versão do Windows para Windows 8 (6.2). Os binários resultantes desta compilação, poderão não rodar nas versões anteriores do Windows.
- Instale o DirectX SDL e configure o projeto `osd_windows` para realizar a procura dos caminhos dos cabeçalhos e das bibliotecas do DirectX antes de procurar os caminhos do Microsoft Windows SDK.

O compilador MSVC produz avisos espúrios sobre as variáveis locais que estejam não-inicializadas potencialmente. Atualmente é preciso adicionar `NOWERROR=1` às opções do make para gerar os arquivos do projeto do Visual Studio. Isto impede que os avisos sejam tratados como erros. (O MSVC parece não ter opções para controlar quais advertências específicas serão tratadas como erro, coisa que os outros compiladores suportam).

Ponto de entrada não encontrado

Caso o seu **sdlmame.exe** mostre um erro como este ou algo parecido:

Não foi possível localizar o ponto de entrada do procedimento
_ZNSt7__cxx1118basic_stringstreamIcSt11char_traitsIcESaIcEEC1Ev na
biblioteca de vínculo dinâmico D:\MAME\sdlmame.exe.

Devido a alteração feita [neste commit](#) o executável **sdlmame.exe** é compilado de maneira que ele busque as bibliotecas que ele precisa para funcionar no sistema em vez de tê-las embutidas em si.

Assim o executável **sdlmame.exe** busca pelas seguintes bibliotecas, `libgcc_s_seh-1.dll`, `libstdc++-6.dll`, `libwinpthread-1.dll`, `libgomp-1.dll` e `SDL2.dll`, todas elas estão dentro do diretório de instalação do seu MSYS2 (exemplo `C:\msys64\mingw64\bin`), é possível adicionar este caminho nas variáveis de ambiente do Windows:

1. Pressione a tecla com a bandeira do Windows (ela é chamada WINKEY) junto com a tecla Pause.
2. Clique na opção chamada Configurações Avançadas do Sistema.
3. Vá em *Avançado* → *Variáveis de Ambiente*.
4. Selecione Path e clique em Editar.
5. Clique em Novo e adicione o caminho onde está instalado o seu MSYS2.
6. No nosso exemplo seria `C:\msys64\mingw64\bin`, clique em Ok para finalizar e feche todas as janelas.

E aqui começa toda a confusão, caso você tenha baixado a [ferramenta de compilação oficial do MAME](#), ela já vem com uma versão do arquivo **libstdc++-6.dll**, porém caso você compile o seu SDL MAME com ela e tempos depois atualize o seu MSYS2, a versão do seu **libstdc++-6.dll** será diferente daquela que você compilou o seu SDL MAME, ocorrendo assim o problema.

Para solucionar o problema basta que você compile uma nova versão do MAME que fará com que este utilize a versão atualizada do arquivo **libstdc++-6.dll**. Caso não queira lidar com variáveis de ambiente, é possível também copiar as bibliotecas acima listadas para o diretório onde se encontra o seu SDL MAME.

Outra maneira de corrigir o problema sem ter que alterar as variáveis de ambiente do Windows é copiar as seguintes DLLs para a mesma pasta do seu **sdlmame.exe**:

```
libgcc_s_seh-1.dll  
libgomp-1.dll  
libstdc++-6.dll  
libwinpthread-1.dll  
SDL2.dll  
SDL2_ttf.dll
```

Note que até a presente versão deste texto, estas são as DLLs que a a versão SDL do MAME pede, pode ser que num determinado momento o MAME possa pedir outras Dlls, e se for o caso de estar faltando alguma dll, o próprio Windows vai mostrar uma nova mensagem de erro dizendo qual a dll que está faltando ao rodar o **sdlmame.exe**, neste caso, vá até a pasta C:\msys64\mingw64\bin e copie a dll que falta para dentro da pasta do MAME.

3.5.8 Configurações para compilações não ortodoxas

Lincando com o LLVM linker

Geralmente o LLVM linker é mais rápido que o GNU linker utilizado pelo GCC. Isso fica mais evidente em sistemas com uma elevada sobrecarga de operações dos arquivos do sistema (como o Microsoft Windows ou ao compilar num disco compartilhado na rede por exemplo). Para utilizar o LLVM linker com o GCC, tenha certeza de tê-lo instalado no seu sistema e utilize `-fuse-ld=lld` nas opções do compilador, seja através da variável de ambiente **LDFLAGS**, através da opção **LDOPTS** ou configurando o **LDOPTS** no arquivo **useroptions.mak**), exemplo:

```
LDOPTS=-fuse-ld=lld
```

Nota: Até a presente versão deste documento a opção ainda *não funciona* caso o MAME seja compilado com o clang. No entanto funciona bem com o gcc fazendo com que todo o processo de lincagem leve apenas **segundos** para ser concluído se comparado com o GNU linker.

Usando libc++ no Linux

O MAME pode ser compilado usando a biblioteca padrão C++ "libc++" do projeto LLVM. Os pré-requisitos são uma instalação funcional do clang/LLVM no seu sistema e a biblioteca de desenvolvimento libc++. No Linux Fedora os pacotes necessários são **libcxx**, **libcxx-devel**, **libcxxabi** e **libcxxabi-devel**. No Debian os pacotes são **libc++1**, **libc++-dev** e **libc++abi-dev**. Defina os compiladores clang C e C++ assim como o **-stdlib=libc++** nas opções do compilador C++ e do seu lincador. O comando completo ficaria assim:

```
env LDFLAGS=-stdlib=libc++ make OVERRIDE_CC=clang OVERRIDE_CXX=clang++ ARCHOPTS_CXX=-  
↳stdlib=libc++ ARCHOPTS_OBJCXX=-stdlib=libc++
```

Ou em caso de erro, tente:

```
env LDFLAGS=-stdlib=libc++ make OVERRIDE_CC=clang OVERRIDE_CXX=clang++ ARCHOPTS_OBJCXX=-  
↳stdlib=libc++ LDOPTS=-fuse-ld=lld -stdlib=libc++
```

As opções depois do comando make podem ser armazenadas num makefile customizado como descrito em [PREFIX_MAKEFILE](#).

Usando uma instalação do GNU GCC libstdc++ que esteja fora do local tradicional no Linux

O GNU GCC pode ter sido compilado e instalado num local diferente caso o mantenedor do mesmo utilize a opção `--prefix=` junto com o comando `configure`. Isso pode ser útil caso queira compilar o MAME numa distribuição Linux que ainda use a versão do GNU libstdc++ que anteceda o C++17. Caso queira compilar o MAME com uma versão alternativa do GNU GCC que esteja instalada em seu sistema, defina o caminho completo dos compiladores C (gcc) e C++ (g++), assim como, adicione o caminho completo da biblioteca do seu sistema. Supondo que tenha o GNU GCC instalado em `/opt/local/gcc72`, use o comando de compilação como mostrado abaixo:

```
make OVERRIDE_CC=/opt/local/gcc72/bin/gcc OVERRIDE_CXX=/opt/local/gcc72/bin/g++ ARCHOPTS=-Wl,  
↳-R,/opt/local/gcc72/lib64
```

Essas configurações podem ser armazenadas num makefile customizado como descrito em [PREFIX_MAKEFILE](#) caso pretenda utilizá-las regularmente.

3.5.9 O MAME travou, o que fazer?

Investigando e lidando com problemas comuns

A princípio é preciso saber se a causa do problema tem origem no MAME, se é algum bug interno ou se vem de alguma configuração externa. A primeira coisa a se fazer é ir eliminando possíveis *culpados*, caso tenha alterado algum tipo de configuração comece renomeando o seu `mame.ini` para `_mame.ini`, isso faz com que o MAME não encontre mais o seu arquivo de configuração use as suas configurações predefinidas internamente.

Caso o MAME não apresente o problema com o exemplo acima, crie um novo `mame.ini` com o comando `mame.exe -createconfig` e usando o editor de texto de sua preferência vá adicionando as suas configurações uma a uma, sempre testando com o MAME cada alteração adicionada até identificar o problema.

Supondo que o problema não tenha sido com o arquivo de configuração, verifique se o conteúdo dos diretórios **bgfx**, **hlsl** e **hash** foram atualizados. É comum para aqueles que compilam a sua versão do MAME e se esquecem de atualizar o conteúdo destes diretórios no dispositivo que estão usando ou até mesmo um outro lugar onde o MAME esteja sendo executado. Isso porém não acontece com quem baixa a versão já compilada do MAME do site oficial.

Experimente apagar o arquivo de configuração da último sistema que foi rodado, fica no diretório **cfg**, apague também o arquivo de memória que fica do diretório **nvram**. Em ambos os diretórios o nome do arquivo ou diretório será o mesmo que o nome do sistema usado, supondo que teve problemas com o sistema **Street Fighter Alpha**, no diretório **nvram** apague o diretório **sfa**, no diretório **cfg**, apague o arquivo **sfa.cfg**. Verifique se não existe nenhuma configuração customizada dentro do diretório **ini**, caso exista, experimente mover este arquivo para um outro lugar.

É provável que depois de uma atualização da versão GIT o MAME tenha se "*quebrado*", ao acompanhar o [desenvolvimento do MAME diariamente](#), verá que durante todo o dia, vários desenvolvedores estão enviando coisas novas e melhorando aquelas que já existem. Esse é o risco de se utilizar a versão GIT pois é uma versão instável que a qualquer momento algo pode deixar de funcionar.

O driver de vídeo algumas vezes pode causar problemas, alguma incompatibilidade com o Direct3D, os casos variam muito. A melhor maneira de descartar isso é testando o MAME usando uma outra opção de vídeo, caso esteja usando `-video d3d` (Windows) ou `-video opengl` (Linux e macOS) tente com `-video soft`. Para outras opções veja [-video](#).

A tela de travamento

Junto aos binários do MAME existe um arquivo de símbolos, para a versão *64-bit* será criado o arquivo **mame.sym** ou **mame.sym** para a versão *32-bit*. Estes arquivos já vem com a versão oficial assim como *já foi explicado* como criá-los durante a compilação.

Estes arquivos devem **sempre** estar junto ao executável do MAME, esse arquivo ".sym" é usado para traduzir as referências usadas no código-fonte junto com os códigos de erro, para a maioria não significa muito porém é útil para os desenvolvedores. Aqui um exemplo de um erro que causou a parada do MAME:

```
Exception at EIP=00000000 (something_state::something()+0x0000): ACCESS VIOLATION
While attempting to read memory at 00000000

-----
EAX=00000000 EBX=0fffffff ECX=0fffffff EDX=00000000
ESI=00000000 EDI=00000000 EBP=00000000 ESP=00000000
-----

Stack crawl:
0012abcd: 00123456 (something_state::something()+0x0000)
0034ef01: 00789abc (something_state::something()+0x0000)
E a listagem continua
...
```

Sem o arquivo de símbolos o `something_state::something` apareceria como um código hexadecimal sem sentido, com os símbolos esses códigos são traduzidos para algo legível e compreensível para os desenvolvedores. Caso o MAME trave durante a emulação, uma tela semelhante irá aparecer, copie e reporte³ o erro no fórum [MAME testers](#).

Usando o gdb

A ideia não é oferecer um manual completo de como usar o gdb, apenas o mínimo necessário para se obter um *stack trace* válido. No exemplo abaixo estou usando uma versão 64-bit do MAME para Linux, porém o procedimento é o mesmo em qualquer outra plataforma.

- Carregue o mame no gdb com o comando `gdb mame`, irá aparecer algo semelhante com a tela abaixo:

```
gdb mame
GNU gdb (Debian 7.12-6) 7.12.0.20161007-git
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from mame...done.
(gdb)
```

³ Pedimos a gentileza de relatar os problemas encontrados em Inglês. (Nota do tradutor)

Para executar o sistema com problema execute run seguido pelos comandos do MAME, exemplo:

```
(gdb) run kof99
Starting program: /home/mame/mame kof99
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[New Thread 0x7fffe4f6c700 (LWP 21026)]
[New Thread 0x7fffe4531700 (LWP 21027)]
[New Thread 0x7fffe3d30700 (LWP 21028)]
[New Thread 0x7fffe352f700 (LWP 21029)]
[New Thread 0x7fffe2d2e700 (LWP 21030)]
[New Thread 0x7fffe9ab5700 (LWP 21031)]
[New Thread 0x7fffe9a74700 (LWP 21032)]
```

O exemplo dado foi com **kof99** porém, pode ser com qualquer outro sistema que tenha dado problema, use a sistema até que o MAME trave, será exibida uma tela como no exemplo abaixo

```
Thread 1 "mame" received signal SIGSEGV, Segmentation fault.
_int_malloc (av=av@entry=0x7ffff459fb00 <main_arena>,
bytes=bytes@entry=67108864) at malloc.c:3650
3650 malloc.c: File or directory not found.
```

Faça o comando where para que o gdb liste as possíveis causas:

```
(gdb) where
#0 _int_malloc (av=av@entry=0x7ffff459fb00 <main_arena>,
bytes=bytes@entry=67108864) at malloc.c:3650
#1 0x00007ffff4280f64 in __GI___libc_malloc (bytes=67108864) at malloc.c:2928
#2 0x00007ffff4d7c7a8 in operator new(unsigned long) ()
from /usr/lib/x86_64-linux-gnu/libstdc++.so.6
#3 0x00005555cd4f0f3 in __gnu_cxx::new_allocator<unsigned char>::allocate ()
at /usr/include/c++/6/ext/new_allocator.h:104
#4 std::allocator_traits<std::allocator<unsigned char> >::allocate ()
at /usr/include/c++/6/bits/alloc_traits.h:436
#5 std::_Vector_base<unsigned char, std::allocator<unsigned char> >::_M_allocate () at /usr/
↳include/c++/6/bits/stl_vector.h:170
#6 std::_Vector_base<unsigned char, std::allocator<unsigned char> >::_M_create_storage ()_
↳at /usr/include/c++/6/bits/stl_vector.h:185
#7 std::_Vector_base<unsigned char, std::allocator<unsigned char> >::_Vector_base () at /
↳usr/include/c++/6/bits/stl_vector.h:136
...
#25 0x00005555591df406 in main () at ../../../../src/osd/sdl/sdlmain.cpp:217
```

O comando acima é suficiente para gerar informações que podem ser copiadas e enviadas para os desenvolvedores, no exemplo acima ele foi cortado entre o item #7 e #25. Para versões do MAME baixados do site oficial, envie essas informações para [MAME testers](#). Já no caso de versões GIT, as informações devem ser enviadas para o mamedev no [github](#). Lembrando que é obrigatório que os relatórios sejam feitos em Inglês. Para interromper o processo basta teclar **c** seguido da tecla **ENTER**, a tela do MAME será fechada, para sair do gdb digite **quit**.

Uma outra opção para o gdb é a utilização de interfaces que ajudam a organizar a saída do gdb como a [GDB Dashboard](#), com ela a saída do gdb além de ficar colorida, fica mais organizada, já são exibidos todos os valores mais relevantes dos registros, código-fonte, etc.

```

— Output/messages —
— Assembly —
0x000055555e33409 ? mov    %rsi,-0x70(%rbp)
0x000055555e3340d ? mov    %edx,-0x74(%rbp)
0x000055555e33410 ? mov    %ecx,-0x78(%rbp)
0x000055555e33413 ? lea     -0x39(%rbp),%rax
0x000055555e33417 ? mov    %rax,%rdi
0x000055555e3341a ? callq  0x5555558de53e <std::allocator<unsigned char>::allocator(>
0x000055555e3341f ? mov    -0x74(%rbp),%ecx

— Expressions —
— History —
— Memory —
— Registers —
rax 0x0000555558945ef0    rbx 0x000055555e45bbd    rcx 0x0000000000000000
rdx 0x0000000004000000    rsi 0x00007fff9ffff010    rdi 0x0000555558945ef0
rbp 0x00007fffffff5900    rsp 0x00007fffffff5870    r8 0x0000000000000001
r9 0x0000000000000001    r10 0x0000000000000000    r11 0x0000000000000000
r12 0x0000000000080000    r13 0x0000000000080000    r14 0x00007fffd10e7010
r15 0x0000000001000000    rip 0x000055555e33413    eflags [ IF ]
cs 0x00000033            ss 0x0000002b            ds 0x00000000
es 0x00000000            fs 0x00000000            gs 0x00000000

— Source —
499
500
501 void cmc_prot_device::gfx_decrypt(uint8_t* rom, uint32_t rom_size, int extra_xor)
502 {
503     int rpos;
504     std::vector<uint8_t> buf(rom_size);
505
506     // Data xor
507     for (rpos = 0; rpos < rom_size/4; rpos++)
508     {
509         decrypt(&buf[4*rpos+0], &buf[4*rpos+3], rom[4*rpos+0], rom[4*rpos+3], type0_t03,
↳ type0_t12, type1_t03, rpos, (rpos>>8) & 1);
— Stack —
[0] from 0x000055555e33413 in cmc_prot_device::gfx_decrypt at ../../../../src/devices/
↳ bus/neogeo/prot_cmc.cpp:504
(no arguments)
[1] from 0x000055555e3392c in cmc_prot_device::cmc42_gfx_decrypt at ../../../../src/
↳ devices/bus/neogeo/prot_cmc.cpp:566
(no arguments)
[+]
— Threads —
[1] id 2509 name neogeo64 from 0x000055555e33413 in cmc_prot_device::gfx_decrypt at ../../..
↳ ../../src/devices/bus/neogeo/prot_cmc.cpp:504

```

A instalação é simples, basta salvar o .gbdinit no seu home. Para que a informação do código-fonte (source) apareça como no exemplo acima, é necessário que o caminho completo onde o MAME foi compilado ainda exista, ou seja, depois de compilar o MAME não faça um make clean, deixe-o como está assim o gdb encontrará o que precisa.

A GDB Dashboard é customizável, oferece plug-ins e outras configurações que atendam as suas necessidades caso queira se envolver com desenvolvimento ou outras funções do gdb que não serão abordadas aqui.

Caso nada apareça no **Source Code** em conjunto com uma mensagem de erro:

```
2509 prot_cmc.cpp File or directry not found.
```

Ainda dentro do gdb indique o caminho completo para o código-fonte do MAME com o comando `directory`:

```
directory /home/mame/src/mame
```

Da próxima vez que ocorrer algum problema o gdb saberá onde pesquisar pelos arquivos fonte, use o comando `list` para testar.

Usando o AddressSanitizer

Quando tudo parece perdido chega a hora de praticamente chutar o balde, há momentos onde não há um *stack trace* ou se tem ele não é informativo o suficiente para que os desenvolvedores tenham informações úteis.

Apesar da opção estar disponível nas configurações, ela não é publicada e tão pouco seu uso é incentivado. Talvez a explicação seja mais simples do que parece, ao ativar o [AddressSanitizer](#) o MAME **rodará bem mais lento** que o normal pois o *AddressSanitizer* é um detector de erros de memória para C/C++.

Usaremos o Debian 9.97 (stretch) como referência, serve de base para outras distribuições e versões, adapte as configurações aqui mostradas para as versões mais recentes caso seja necessário. Atualmente o clang já está na versão 7, porém a versão 5 continua bem estável e é suficiente para o nosso exemplo, caso queira testar versões mais novas, é por sua conta e risco pois talvez hajam questões de conflitos que precisam ser resolvidos e que não serão abordados aqui.

Como administrador crie o arquivo **clang.list**:

```
sudo touch /etc/apt/sources.list.d/clang.list
```

Adicione as linha abaixo ao arquivo clang.list, o exemplo foi feito com a versão **5.0** porém ajuste para versões mais recentes ou que sejam compatíveis com a sua distribuição:

```
# 5.0
deb http://apt.llvm.org/stretch/ llvm-toolchain-stretch-5.0 main
```

Depois de um apt-get update instale com o comando:

```
sudo apt-get install clang-5.0 libclang-common-5.0-dev libclang1-5.0 liblldb-5.0 lldb-5.0_
↳python-lldb-5.0 libllvm5.0 llvm-5.0 llvm-5.0-runtime
```

Adicione as linhas abaixo ao arquivo ~/.bashrc do seu home:

```
echo "export ASAN_OPTIONS=symbolize=1" >> ~/.bashrc
echo "export ASAN_SYMBOLIZER_PATH=/usr/lib/llvm-5.0/bin/llvm-symbolizer" >> ~/.bashrc
```

Caso a sua distribuição seja diferente, faça o comando `locate llvm-symbolizer` para saber o caminho completo do seu **llvm-symbolizer** e adicione ao **ASAN_SYMBOLIZER_PATH**.

Recarregue as configurações do seu terminal com o comando `. .bashrc` (ponto, espaço, ponto bashrc) ou encerre a seção e faça login novamente.

Compile o MAME como mostra o exemplo abaixo:

```
make clean && make OVERRIDE_CC=/usr/bin/clang OVERRIDE_CXX=/usr/bin/clang++ OPTIMIZE=0_
↳SYMBOLS=1 SYMLEVEL=1 SANITIZE=address -j7
```

Ao rodar o MAME com o sistema com problema, você terá um retorno semelhante ao exemplo abaixo:

```

==2227==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x7f6b8d4a6800 at pc_
↳ 0x0000019a963e bp 0x7ffd4dd2d450 sp 0x7ffd4dd2d448
READ of size 2 at 0x7f6b8d4a6800 thread T0
    #0 0x19a963d in sma_prot_device::kof99_decrypt_68k(unsigned char*) /home/mame/build/
↳ projects/sdl/mame/gmake-linux-clang/../../../../src/devices/bus/neogeo/prot_sma.
↳ cpp:426:24
    #1 0x15e7b10 in neogeo_sma_kof99_cart_device::decrypt_all(unsigned char*, unsigned_
↳ int, unsigned char*, unsigned int, unsigned char*, unsigned int, unsigned char*, unsigned_
↳ int, unsigned char*, unsigned int, unsigned char*, unsigned int, unsigned char*, unsigned_
↳ int) /home/mame/build/projects/sdl/mame/gmake-linux-clang/../../../../src/devices/bus/
↳ neogeo/sma.cpp:75:14
    #2 0x15e7cd5 in non-virtual thunk to neogeo_sma_kof99_cart_device::decrypt_
↳ all(unsigned char*, unsigned int, unsigned char*, unsigned int, unsigned char*, unsigned_
↳ int, unsigned char*, unsigned int, unsigned char*, unsigned int, unsigned char*, unsigned_
↳ int, unsigned char*, unsigned int) /home/mame/build/projects/sdl/mame/gmake-linux-clang/..
↳ ../../../../../../src/devices/bus/neogeo/sma.cpp
    #3 0x6232c5 in neogeo_cart_slot_device::late_decrypt_all() /home/mame/build/projects/
↳ sdl/mame/gmake-linux-clang/../../../../src/devices/bus/neogeo/slot.h:327:48
...
SUMMARY: AddressSanitizer: heap-buffer-overflow /home/mame/build/projects/sdl/mame/gmake-
↳ linux-clang/../../../../src/devices/bus/neogeo/prot_sma.cpp:426:24 in sma_prot_
↳ device::kof99_decrypt_68k(unsigned char*)
Shadow bytes around the buggy address:
0x0fedf1a8ccb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0fedf1a8ccc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0fedf1a8ccd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0fedf1a8cce0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0fedf1a8ccf0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0fedf1a8cd00:[fa]fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0fedf1a8cd10: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0fedf1a8cd20: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0fedf1a8cd30: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0fedf1a8cd40: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0fedf1a8cd50: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
Addressable:          00
Partially addressable: 01 02 03 04 05 06 07
Heap left redzone:    fa
Freed heap region:    fd
Stack left redzone:   f1
Stack mid redzone:    f2
Stack right redzone:  f3
Stack after return:   f5
Stack use after scope: f8
Global redzone:       f9
Global init order:    f6
Poisoned by user:     f7
Container overflow:    fc
Array cookie:          ac
Intra object redzone: bb
ASan internal:         fe

```

(continua na próxima página)

(continuação da página anterior)

```
Left alloca redzone:    ca
Right alloca redzone:   cb
==2227==ABORTING
```

3.6 Compilação cruzada

3.6.1 Definição

Compilação cruzada¹ é o processo de poder compilar um executável numa plataforma diferente da qual ela se destina. Como usar o ambiente Linux para compilar um programa que rode no Windows, Mac ou qualquer outra plataforma compatível com o MAME. Nas instruções a seguir iremos configurar um ambiente de compilação cruzada numa plataforma Linux para compilar uma versão do MAME voltada para o Microsoft Windows, apesar do processo abaixo ser voltado para Windows, ele pode servir também de modelo para compilar o MAME em outras plataformas compatíveis além do Windows.

3.6.2 Vantagens

Dentre as várias vantagens é possível citar as mais relevantes:

- Transformar o código-fonte em linguagem de máquina consome muitos recursos e em geral a plataforma de destino pode não ter todos os recursos disponíveis em comparação com computador que está sendo usado para compilar, como por exemplo, poder de processamento, memória, etc.
- Ainda que seja utilizado o mesmo computador com dois sistemas operacionais instalados como o Linux e o Windows, o tempo gasto para compilar uma versão do MAME para o Linux é muito menor do que compilar a mesma versão no Windows. Compilar uma versão do MAME para Linux leva em torno de 30 minutos para mais ou para menos dependendo do poder de processamento do seu computador; compilando o mesmo código-fonte no mesmo sistema, com o Windows, usando a mesma versão do gcc e g++, a tarefa pode levar algumas horas² ainda que tenha um computador mais recente.
- Ao utilizar o processo de compilação cruzada, o tempo final de compilação leva aproximadamente o mesmo tempo que a versão nativa do Linux, ganhando tempo e economizando recursos, afinal de contas, manter o processador a 100% compilando o código-fonte por cerca de 30 minutos é uma coisa, fazer exatamente a mesma coisa gastando algumas horas além de ser uma perda de tempo, a sua conta de energia pode ficar um pouco mais cara no final do mês.
- Assim, o motivo principal para adotar a compilação cruzada é a economia de tempo e recursos.

3.6.3 Preparando o ambiente

A plataforma usada neste exemplo foi o *Debian 9*, porém pode ser qualquer outro, a vantagem do Debian é que ela é uma distribuição muito estável do sistema operacional Linux, por causa disso, o Debian não utiliza a última versão de nenhum software como o gcc por exemplo. Geralmente ela fica algumas versões para trás do último lançamento encontrado na internet pois o foco é a estabilidade ao invés de empacotar a última versão do que quer que seja.

Precisamos instalar os pacotes abaixo para compilar binários voltados ao sistema Windows, para outros sistemas operacionais ou dispositivos, o procedimento será semelhante contanto que seja escolhido o conjunto de pacotes apropriados para a plataforma desejada.

¹ Cross compiling no Inglês. (Nota do tradutor)

² Todo o processo no meu computador leva cerca de 4 horas, AMD FX tm-8350, 16GiB de memória DDR3. (Nota do tradutor)

Debian e Ubuntu

O comando abaixo vai instalar ferramentas adicionais além das quais já foram descritas na seção *Debian e Ubuntu (incluindo dispositivos Raspberry Pi e ODROID)*, note que o comando abaixo é formado por uma linha só:

```
64 bit
sudo aptitude install g++-mingw-w64-x86-64-posix gcc-mingw-w64-x86-64-posix gcc-mingw-w64-
↳x86-64-posix-runtime gobjc++-mingw-w64-x86-64-posix gobjc-mingw-w64-x86-64-posix binutils-
↳mingw-w64-x86-64 mingw-w64-common mingw-w64-tools mingw-w64-x86-64-dev win-iconv-mingw-w64-
↳dev

32 bit
sudo aptitude install g++-mingw-w64-i686-posix gcc-mingw-w64-i686-posix gcc-mingw-w64-i686-
↳posix-runtime gobjc++-mingw-w64-i686-posix gobjc-mingw-w64-i686-posix binutils-mingw-w64-
↳i686
```

Nota: No Debian 11 (e talvez mais recentes) não é preciso instalar o pacote win-iconv-mingw-w64-dev.

Como estamos fazendo uma compilação entre plataformas é necessário usar a versão POSIX para o **gcc**, **ar** e **g++**, o POSIX vem de *Interface Portável entre Sistemas Operacionais* que é regida pela norma [IEEE 1003](#)³. Para configurar os atalhos do **gcc**, **ar** e **g++** voltado para a criação de binários para a plataforma **64-bit** faça os comandos abaixo no terminal, note que **cada** comando *sudo* é formado por uma linha só:

```
sudo ln -s /usr/bin/x86_64-w64-mingw32-g++-posix /usr/x86_64-w64-mingw32/bin/x86_64-w64-
↳mingw32-g++
sudo ln -s /usr/bin/x86_64-w64-mingw32-gcc-ar-posix /usr/x86_64-w64-mingw32/bin/x86_64-w64-
↳mingw32-gcc-ar
sudo ln -s /usr/bin/x86_64-w64-mingw32-gcc-posix /usr/x86_64-w64-mingw32/bin/x86_64-w64-
↳mingw32-gcc
```

Já para a plataforma **32-bit** faremos estes comandos, note que **cada** comando *sudo* é formado por uma linha só:

```
sudo ln -s /usr/bin/i686-w64-mingw32-g++-posix /usr/i686-w64-mingw32/bin/i686-w64-mingw32-g++
sudo ln -s /usr/bin/i686-w64-mingw32-gcc-ar-posix /usr/i686-w64-mingw32/bin/i686-w64-mingw32-
↳gcc-ar
sudo ln -s /usr/bin/i686-w64-mingw32-gcc-VERSÃO-posix /usr/i686-w64-mingw32/bin/i686-w64-
↳mingw32-gcc
```

Fedora Linux

Instale os seguintes pacotes:

```
sudo dnf install mingw64-binutils mingw64-cpp mingw64-gcc mingw64-gcc-c++ mingw64-gcc-objc_
↳mingw64-gcc-objc++ mingw64-fontconfig mingw64-win-iconv mingw64-winpthreads mingw64-
↳winpthreads-static
```

Seguido dos comandos abaixo:

³ IEEE é conhecido no Brasil como Instituto de Engenheiros Eletricistas e Eletrônicos. (Nota do tradutor)


```
sudo ln -s /usr/bin/x86_64-w64-mingw32-g++ /usr/x86_64-w64-mingw32/bin/x86_64-w64-mingw32-g++
sudo ln -s /usr/bin/x86_64-w64-mingw32-gcc-ar /usr/x86_64-w64-mingw32/bin/x86_64-w64-mingw32-
↪gcc-ar
sudo ln -s /usr/bin/x86_64-w64-mingw32-gcc /usr/x86_64-w64-mingw32/bin/x86_64-w64-mingw32-gcc
```

Configurando as variáveis de ambiente

As variáveis **MINGW64** e **MINGW32** são necessárias para que os scripts usados para a compilação do MAME saibam onde encontrá-los. **Não use sudo** para o comando abaixo pois queremos aplicá-las nas variáveis de ambiente da nossa conta comum e não numa conta com poderes administrativos:

```
echo "export MINGW64=\"/usr/x86_64-w64-mingw32\"" >> ~/.bashrc
echo "export MINGW32=\"/usr/i686-w64-mingw32\"" >> ~/.bashrc
```

Recarregue as configurações do seu terminal com o comando `. ~/.bashrc` (ponto, espaço, ponto bashrc) ou saia e retorne à sua conta. É necessário aferir a configuração para que se tenha certeza de que as variáveis estão definidas no ambiente corretamente fazendo o comando abaixo:

```
$ echo $MINGW64 && echo $MINGW32
/usr/x86_64-w64-mingw32
/usr/i686-w64-mingw32
```

Caso o seu ambiente não tenha retornado nada, tenha certeza de que as instruções acima foram seguidas corretamente, se a sua distribuição Linux - ou outra distribuição - utiliza o arquivo `.bashrc`, caso não utilize, verifique no manual da sua distribuição qual arquivo de configuração ela utiliza para armazenar as variáveis do ambiente e onde ele se localiza.

3.6.4 Compilando o MAME para Windows no Linux

Para compilar uma versão *64-bit* do MAME para o **Windows**, execute o comando abaixo, lembrando que o comando deve ser executado de dentro da pasta raiz⁴ do código-fonte do MAME:

```
make clean && make TARGETOS=windows CROSS_BUILD=1 SYMBOLS=1 SYMLEVEL=1 STRIP_SYMBOLS=1_
↳SSE2=1 PTR64=1
```

Para compilar uma versão *32-bit* do MAME faça o comando abaixo:

```
make clean && make TARGETOS=windows CROSS_BUILD=1 SYMBOLS=1 SYMLEVEL=1 STRIP_SYMBOLS=1 SSE2=1
```

Assim como na compilação nativa, é possível adicionar a opção `-j` no final do comando visando acelerar o processo de compilação usando os núcleos do seu processador como já foi explicado com mais detalhes no capítulo [Compilando o MAME](#):

```
make clean && make TARGETOS=windows CROSS_BUILD=1 SYMBOLS=1 SYMLEVEL=1 STRIP_SYMBOLS=1_
↳SSE2=1 PTR64=1 -j5
```

⁴ Fica no mesmo diretório onde existe um arquivo chamado **makefile**. (Nota do tradutor)

3.6.5 Compilando o MAME SDL para Windows no Linux

Caso queira compilar uma versão SDL do MAME instale as estes pacotes mingw do SDL2, SDL2-static, SDL2_ttf e SDL2.

Debian e Ubuntu

Infelizmente será necessário compilar estes pacotes manualmente e no momento não iremos cobrir este assunto aqui porém lembre-se que é possível compilar esta versão do MAME SDL usando o *MINGW no Windows!*

Basta compilar usando a opção OSD=sdl na sua linha de comando, exemplo:

```
make clean && make OSD=sdl SYMBOLS=1 SYMLEVEL=1 STRIP_SYMBOLS=1 SSE2=1 -j5
```

Ao final da compilação será gerado um arquivo **sdlmame.exe**.

Fedora Linux

```
sudo dnf install mingw64-SDL2_ttf mingw64-SDL2 mingw64-SDL2-static
```

Agora use a opção OSD=sdl como mostra o exemplo abaixo para versões 32-bit:

```
make clean && make TARGETOS=windows CROSS_BUILD=1 OSD=sdl SYMBOLS=1 SYMLEVEL=1 STRIP_
↳SYMBOLS=1 SSE2=1 -j5
```

Para versões 64-bit:

```
make clean && make TARGETOS=windows CROSS_BUILD=1 OSD=sdl SYMBOLS=1 SYMLEVEL=1 STRIP_
↳SYMBOLS=1 SSE2=1 PTR64=1 -j5
```

Ao final da compilação será gerado um arquivo **sdlmame.exe**.

3.7 Configuração inicial

- *Configuração básica*
- *O caminho das ROMs*
 - *Configuração tradicional*
 - *Configuração avançada*
- *Configurando os gráficos*
- *Configurando o áudio*
- *Configurando os controles*

3.7.1 Configuração básica

Inicialmente o MAME precisa de poucas configurações para funcionar, num primeiro momento o MAME precisa criar alguns arquivos de configuração e depois é preciso configurar o caminho onde as suas ROMs estão.

Para os usuários do Windows, baixe o binário do [site oficial](#) e descompacte-o em algum lugar. Para usuários de outras versões como Linux ou macOS, será preciso compilar ou ver a disponibilidade de versões já compiladas para o seu sistema operacional.

No Windows, vá até a pasta onde o MAME foi descompactado, escolha um ponto vazio da pasta, clique com o botão direito do mouse e escolha **Novo --> Atalho**, quando aparecer a janela "*Criar atalho*" em "*Digite o local do item*" insira **cmd** e clique em **OK**. O ícone do prompt de comando deverá aparecer dentro da pasta do MAME, clique com o *botão direito* em cima dele e selecione **Propriedades**, selecione a aba **Atalho**, no campo **Iniciar em:** tem o caminho **C:\WINDOWS\system32** ou algo assim, **apague** o que estiver neste campo e clique em **OK**.

Clique duas vezes no ícone do prompt de comando e pronto, já estamos no diretório do MAME, para criar os arquivos de configuração inicial faça:

```
mame -createconfig ou mame -cc
```

O comando também serve para as versões do Linux e do macOS, este comando cria os seguintes arquivos:

- **name.ini**

É o arquivo que contém as principais configurações do MAME.

- **ui.ini**

Contém as configurações da interface, como a definição do tipo e o tamanho da fonte, da cor do texto e do fundo, etc.

- **plugins.ini**

Aqui fica a lista dos plug-ins disponíveis no MAME, depois do nome do plug-in, o **1** indica que ele está ativo e **0** que está desativado. Consulte [Plug-ins](#) para obter mais detalhes sobre os plug-ins que acompanham o MAME.

No macOS os arquivos ficam em ~/Application Support/mame, nos sistemas Linux eles ficam em /home/seu_usuario/.mame ou simplificando, no diretório ~/.mame.

Observe que o name.ini é basicamente um arquivo de texto simples, podendo ser editado com qualquer editor de texto (como o Notepad, o [Notepad++](#), o [Geany](#), o [Emacs](#) ou o [TextEdit](#) por exemplo) e alterar todas as opções conforme a sua necessidade.

Nota: Sempre mantenha um backup destes arquivos!

3.7.2 O caminho das ROMs

Configuração tradicional

A primeira configuração a ser feita é apontar o caminho completo onde as suas principais ROMs estão armazenadas. A opção `-rompath` permite a definição deste caminho, porém, é possível deixá-lo fixo, para isso abra o `mame.ini` e edite a opção `rompath`:

<code>rompath</code>	<code>roms</code>
----------------------	-------------------

Esta é a configuração original, ao definir `roms` o MAME vai procurar por uma pasta chamada `roms` no mesmo diretório do executável do MAME. Caso as suas ROMs fiquem em outro lugar, adicione o caminho completo separado por ponto e vírgula, exemplo:

<code>rompath</code>	<code>roms;D:\roms;E:\bkp\roms;F:\arcade\roms</code>
----------------------	--

Assim o MAME sempre irá buscar as ROMs nestes caminhos que foram definidos.

Porém dependendo da quantidade de ROMs que cada pasta tenha e até mesmo a quantidade de caminhos, esta talvez não seja a melhor maneira de configurar pois o MAME pode ficar um pouco mais lento até ele fazer a varredura de todos os diretórios ao executar a sua interface gráfica. Caso haja muitas e muitas ROMs na sua coleção a melhor opção possa ser a configuração avançada.

Configuração avançada

Uma outra maneira de se configurar o caminhos das ROMs é utilizando as configurações individuais parcialmente descrito em *-readconfig*. Primeiro é preciso identificar qual a configuração do seu inipath, para isso faça o comando no Windows:

```
mame -showconfig|findstr initpath
inipath                .;ini;ini/presets
```

No Linux e no macOS faça:

```
mame -showconfig|grep inipath
inipath                $HOME/.mame;.;ini
```

Em ambos os casos, o ponto final indica o mesmo lugar que o executável do MAME. Uma vez identificado onde a sua pasta ini está, dentro dela crie uma pasta chamada **sources**.

Vamos supor que queira jogar o **Sonic the Hedgehog** da SEGA no Genesis/Mega Drive, primeiro identifique o driver com o comando:

```
mame genesis -ls
genesis                sega/mdconsole.cpp
```

O driver é mdconsole.cpp, então crie o arquivo mdconsole.ini dentro da pasta ini\sources, edite o arquivo mdconsole.ini e adicione o caminho completo onde estão armazenados as suas ROMs do Genesis/Mega Drive, exemplo:

```
rompath                D:\genesis\roms
```

No Linux ou macOS:

```
rompath                /media/genesis/roms
```

Isso pode ser feito com qualquer outro sistema, basta identificar o driver, criar um arquivo *.ini com o nome do driver (sem o .cpp) dentro da pasta ini\sources. Lembrando que caso o sistema que será executado utilize algum tipo de BIOS como o Neo Geo por exemplo, é preciso apontar o caminho completo onde se encontra a BIOS do sistema e o caminho completo das ROMS do sistema.

Se for o caso do Neo Geo, crie o arquivo ini\sources\neogeo.ini, dentro do arquivo neogeo.ini adicione:

```
rompath                D:\roms;D:\bios
```

Um dos motivos de se deixar as ROMs dos sistemas separadas em vez de se deixar tudo junto, depende muito da quantidade de ROMs existente na pasta roms. Caso tenha TODAS AS ROMS do MAME numa mesma pasta, o gerenciamento delas pode ficar muito comprometido pois dependendo do hardware usado, uma simples navegação dentro desta pasta pode ficar extremamente lento dada a imensa quantidade de arquivos.

3.7.3 Configurando os gráficos

A maneira mais simples de fazer a configuração gráfica do MAME é clicando duas vezes em seu executável ou o invocando através do terminal ou do prompt de comando sem qualquer opção, quando a interface aparecer, com o mouse dê um clique duplo em *Definições gerais* e depois em *Opções do vídeo* e escolha uma das opções disponíveis em *Modo do vídeo*, para mais detalhes consulte [-video](#).

Se for possível prefira a opção *bgfx*, caso contrário *opengl* e em último caso **d3d**. O MAME ainda usa o Direct3D versão 9 porém do Windows 7 em diante o Direct 3D 9 não é mais compatível com as versões mais novas do Windows.

Note que apesar de ser possível selecionar o modo do vídeo, para tirar proveito da aceleração da sua placa de vídeo, prefira usar a versão do Direct3D do *bgfx*, neste caso, abra o seu *mame.ini*, localize a opção vídeo e troque auto por *bgfx*, exemplo:

video	bgfx
-------	------

Agora, procure a opção *bgfx_backend* e troque auto por *d3d11* ou *d3d12* caso a sua placa de vídeo seja compatível:

bgfx_backend	d3d12
--------------	-------

Também é possível fazer a configuração **parcial** do vídeo usando a própria interface do MAME, na linha de comando ou do prompt execute o *mame* sem nenhuma opção.

As configurações que já vem definidas são suficientes para a grande maioria dos usuários, caso não queira que o MAME inicie ocupando a tela inteira por exemplo, altere a opção *Modo janela* para *Ligado*, depois faça um duplo clique em *Retorne ao menu anterior* e novamente em *Salve a configuração* para armazenar as alterações no **mame.ini**.

Tenha ciência que nada que seja feito aqui vai "quebrar" o MAME, contudo, pode ser que por alguma questão de incompatibilidade, o MAME não abra mais a interface, rode sem som, com tela preta ou algum outro tipo de problema. Neste caso, faça o backup dos seus arquivos *mame.ini*, *ui.ini* e *plugins.ini*, crie uma nova configuração com o comando *mame -cc* e tente novos ajustes, evite aqueles que causaram problemas. Para mais informações consulte [-createconfig](#).

Para mais detalhes sobre todas as opções da parte da configuração do vídeo, consulte [Opções do vídeo](#), [Efeitos BGFX para \(quase\) todo mundo](#) para saber mais.

3.7.4 Configurando o áudio

Assim como na configuração de vídeo, num primeiro momento, o MAME detecta e configura o áudio de forma automática para que funcione com o hardware existente. Porém é possível obter uma menor latência (atraso) no áudio e assim melhorar o seu desempenho, os sistemas como a **Dance Dance Revolution** e outras do tipo, podem se beneficiar bastante de uma latência menor.

Para identificar a sua placa de som no MAME execute o comando *mame -v -sound portaudio*, ele deverá retornar uma lista com as opções disponíveis seja para Windows, Linux ou macOS. Para mais informações, primeiro consulte a opção [-pa_api](#) para identificar corretamente o seu hardware antes de continuar.

Abra o seu arquivo *mame.ini* e adicione estas opções:

# Audio	
sound	portaudio
lowlatency	1
samplerate	44100

(continua na próxima página)

(continuação da página anterior)

```
pa_api          ALSA
pa_device       "Xonar STX: Multichannel (hw:0,0)"
```

Remova a opção `pa_device` caso queira deixar que a seleção seja automática. Salve o arquivo e rode novamente no terminal o comando `mame ddrmax2 -v`, dentre as várias informações você deverá ver algo como mostra o exemplo abaixo para o meu computador:

```
PortAudio: API ALSA has 13 devices
PortAudio: ALSA: "Xonar STX: Multichannel (hw:0,0)" (default)
PortAudio: ALSA: "Xonar STX: Digital (hw:0,1)"
PortAudio: ALSA: "HDA ATI HDMI: 0 (hw:1,3)"
PortAudio: ALSA: "HDA ATI HDMI: 1 (hw:1,7)"
PortAudio: ALSA: "HDA ATI HDMI: 2 (hw:1,8)"
PortAudio: ALSA: "HDA ATI HDMI: 3 (hw:1,9)"
PortAudio: ALSA: "HDA ATI HDMI: 4 (hw:1,10)"
PortAudio: ALSA: "HDA ATI HDMI: 5 (hw:1,11)"
PortAudio: ALSA: "sysdefault"
PortAudio: ALSA: "iec958"
PortAudio: ALSA: "spdif"
PortAudio: ALSA: "pulse"
PortAudio: ALSA: "a52"
PortAudio: API OSS has 0 devices
PortAudio: Using device "Xonar STX: Multichannel (hw:0,0)" on API "ALSA"
PortAudio: Sample rate is 44100 Hz, device output latency is 8.67 ms
PortAudio: Allowed additional buffering latency is 30.00 ms/1440 frames
```

Experimente jogar uma partida e repare que houve uma melhora considerável no sincronismo do som com a ação na tela. Para obter o benefício de uma latência menor, o uso da placa de som se torna exclusiva para o MAME, ou seja, caso você goste de usar o MAME enquanto escuta música de fundo ou ouvir o som do Youtube, Spotify, Tidal ou qualquer outro site ou programa que use a placa de som, o áudio pode **não funcionar**.

Neste caso altere a configuração do arquivo `mame.ini` para:

```
# Audio
sound          sdl
samplerate      44100
```

Para manter o benefício do áudio com baixa latência para todos os sistemas referentes ao driver **ksys573** como a **Dance Dance Revolution**, entre no diretório **ini** e crie o diretório **source**, dentro dele crie o arquivo `ksys573.ini` com as configurações de áudio usadas anteriormente:

```
# Audio
sound          portaudio
lowlatency      1
pa_api          ALSA
pa_device       "Xonar STX: Multichannel (hw:0,0)"
```

Todos os sistemas que estão dentro da categoria "Arcade" agora passam a usar a interface comum de áudio e que funciona junto com quaisquer outros programas ou serviços de áudio e apenas os sistemas do driver **ksys573** passam a usar a configuração com baixa latência. O mesmo pode ser feito com outros drivers como a **djmain** que é responsável pelos sistemas da série **Beatmania** e **Pop'n Music**, lembrando que você pode identificar o nome do driver com o comando `mame nome_da_rom -ls`, para mais informações consulte o comando `-listsources`.

3.7.5 Configurando os controles

O MAME aceita dois tipos de configuração para os controles, a primeira é a configuração feita através da interface, depois de iniciar um sistema qualquer como o "Street Fighter II" da Capcom mame sf2, pressione Tab e selecione *Configurações da entrada -> Atribuições da entrada (este sistema)*:



Aqui temos a configuração que já vem predefinida para este sistema, o **P1** são as definições para o **jogador 1** e assim por diante. Selecione a configuração para **Up (cima)** e pressione Enter no teclado e pressione **cima** no seu controle, manche ou joystick e faça o mesmo para as outras definições.

A sequência dos botões de soco para este sistema estão organizados como:

- *Jab Punch* (soco fraco)
- *Strong Punch* (soco médio)
- *Fierce Punch* (soco forte)

Para o chute nós temos:

- *Short Kick* (chute fraco)
- *Forward Kick* (chute médio)
- *Roundhouse Kick* (chute forte)

No final, o nome para cada tipo de controle pode ficar um pouco diferente, isso pode variar muito dependendo do modelo e do adaptador usado. No exemplo da foto abaixo e usando um adaptador USB para

Playstation 2 nós fizemos a seguinte configuração, **quadrado (soco fraco)**, **L1 (soco médio)**, **triângulo (soco forte)**, **R1 (chute fraco)**, **xis (chute médio)**, **círculo (chute forte)**.

Isso nos deixa com a seguinte configuração:



A escrita mais apagada do lado direito do comando indica que a configuração está customizada ou usando uma configuração diferente da configuração predefinida. Pressione a tecla **ESQ** até encerrar a emulação ou simplesmente feche a janela. Ao encerrar a emulação o MAME cria uma configuração com o nome desta ROM sf2.cfg na pasta cfg:

```
<?xml version="1.0"?>
<!-- This file is autogenerated; comments and unknown tags will be stripped -->
<mameconfig version="10">
  <system name="sf2">
    <input>
      <port tag=":IN1" type="P1_JOYSTICK_RIGHT" mask="1" defvalue="1">
        <newseq type="standard">
          JOYCODE_1_AXIS_RIGHT_SWITCH
        </newseq>
      </port>
      <port tag=":IN1" type="P1_JOYSTICK_LEFT" mask="2" defvalue="2">
        <newseq type="standard">
          JOYCODE_1_AXIS_LEFT_SWITCH
        </newseq>
      </port>
      <port tag=":IN1" type="P1_JOYSTICK_DOWN" mask="4" defvalue="4">
```

(continua na próxima página)

(continuação da página anterior)

```

        <newseq type="standard">
            JOYCODE_1_YAXIS_DOWN_SWITCH
        </newseq>
    </port>
    <port tag=":IN1" type="P1_JOYSTICK_UP" mask="8" defvalue="8">
        <newseq type="standard">
            JOYCODE_1_YAXIS_UP_SWITCH
        </newseq>
    </port>
    <port tag=":IN1" type="P1_BUTTON1" mask="16" defvalue="16">
        <newseq type="standard">
            JOYCODE_1_BUTTON4
        </newseq>
    </port>
    <port tag=":IN1" type="P1_BUTTON2" mask="32" defvalue="32">
        <newseq type="standard">
            JOYCODE_1_BUTTON7
        </newseq>
    </port>
    <port tag=":IN1" type="P1_BUTTON3" mask="64" defvalue="64">
        <newseq type="standard">
            JOYCODE_1_BUTTON1
        </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON4" mask="1" defvalue="1">
        <newseq type="standard">
            JOYCODE_1_BUTTON8
        </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON5" mask="2" defvalue="2">
        <newseq type="standard">
            JOYCODE_1_BUTTON3
        </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON6" mask="4" defvalue="4">
        <newseq type="standard">
            JOYCODE_1_BUTTON2
        </newseq>
    </port>
</input>
</system>
</mameconfig>

```

O MAME sempre vai procurar pela configuração sf2.cfg sempre que a ROM sf2 for carregada.

Porém, existem diferentes versões deste sistema em diferentes drivers como o CPS-1, CPS-2, CPS-3 e várias outras que usam um esquema semelhante de botões. Para evitar o trabalho de se criar uma configuração destas para cada sistema individualmente, é possível aplicar esta exata configuração **por sistema**, ou seja, todas os sistemas do driver CPS-1 por exemplo, podem usar uma só configuração.

Para isso, copie o arquivo sf2.cfg da pasta cfg para a pasta ctrlr, esta pasta fica junto com o executável do MAME, caso o seu MAME venha de uma distribuição Linux ou macOS em particular com algum tipo de instalação, faça o comando:

```
mame -showconfig | grep ctrlrpath
ctrlrpath                /usr/share/ctrlr;etc
```

Depois de copiar o arquivo sf2.cfg para esta pasta, o renomeie para algo como street.cfg (pode ser o nome que quiser). Abra o arquivo street.cfg (ou o nome do arquivo que usou) num editor de texto e troque o **sf2** da linha <system name="sf2"> para **default** ou <system name="default"> e salve o arquivo.

Para aplicar a configuração para **TODOS** os sistemas do driver CPS-1, crie uma pasta chamada sources dentro da pasta ini, dentro da pasta sources crie um arquivo chamado cps1.ini, abra-o no editor de texto e adicione a opção:

```
ctrlr                street
```

Note que street se refere ao street.cfg que criamos **sem o .cfg**. Apague o seu arquivo sf2.cfg da pasta cfg e inicie novamente o sistema, pressione Tab e selecione *Configurações da entrada -> Atribuições da entrada (este sistema)*, repare que o MAME carregou as configurações do seu controle. Encerre a emulação novamente, tente outro sistema como a sf2ce, repare que este sistema também vai estar usando as configurações que você definiu para o seu controle, o mesmo vai acontecer para todos os outros sistemas deste driver.

Neste driver também há sistemas de tiro, pancadaria, dentre outros. Nestes casos é preciso criar uma configuração por sistema. Por exemplo, o "Carrier Air Wing (cawing)" possui 3 botões, é possível usar o mesmo tipo de configuração já ensinado anteriormente com Tab -> *Configurações da entrada -> Atribuições da entrada (este sistema)* em cada sistema ou pegar a configuração que for criada, neste caso seria cfg\cawing.cfg, alterar o **system name** para **default** como também já foi explicado anteriormente, porém agora, salve este arquivo como 3-botoes.cfg dentro da pasta ctrlr.

Vá até a pasta ini e crie um ini com o nome da ROM ou cawing.ini, abra o arquivo num editor, adicione a opção e salve:

```
ctrlr                3-botões
```

Assim os outros sistemas usam a configuração street para jogos com 6 botões enquanto a cawing usa a configuração com 3 botões. O mesmo princípio pode ser utilizado com sistemas que usam 2 botões e assim por diante.

Outras configurações também podem ser feitas, neste caso, consulte o capítulo *Truques para tornar a vida mais fácil*.

CAPÍTULO 4

NOÇÕES GERAIS DE USO E CONFIGURAÇÃO

4.1 Primeiros passos

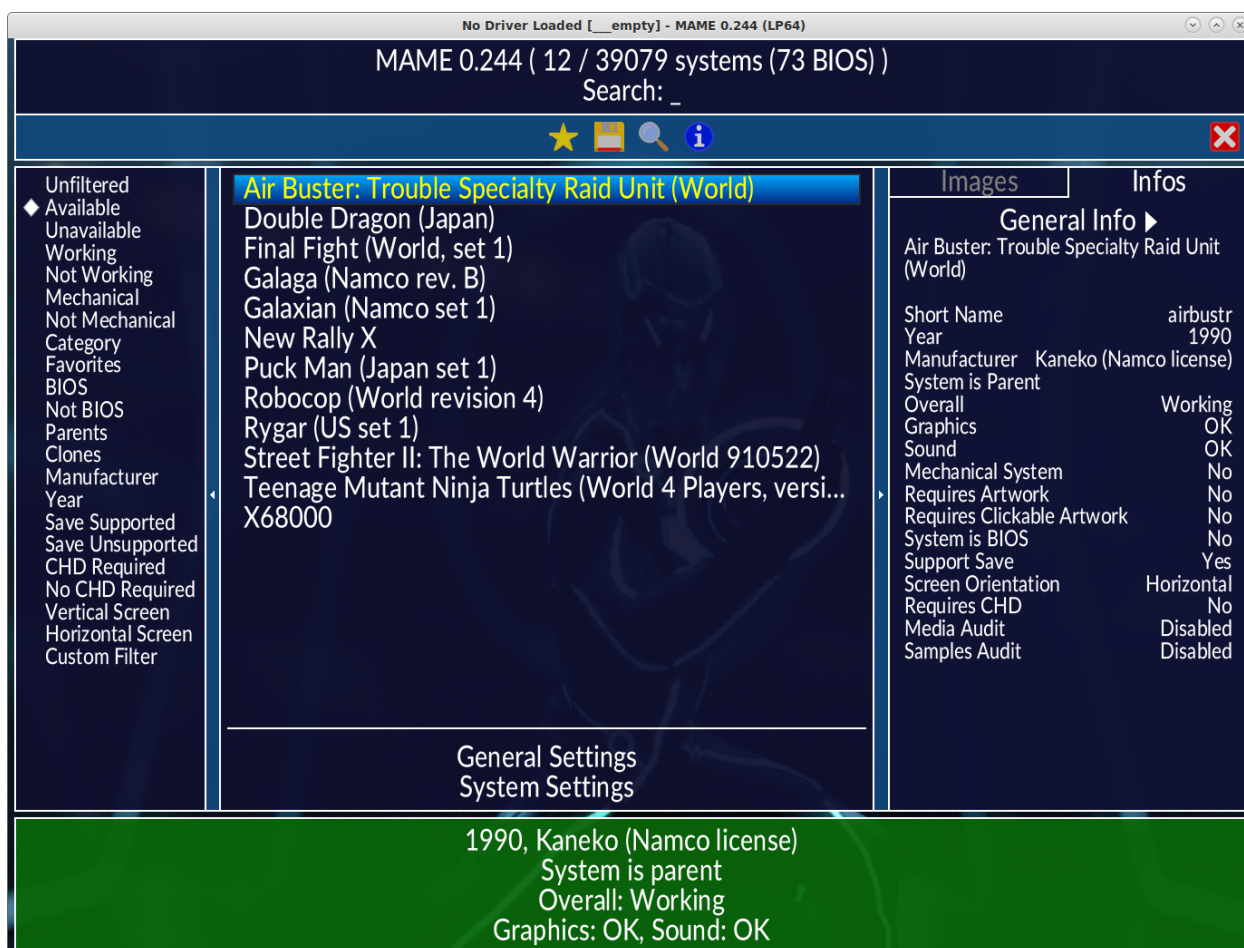
- *Interface gráfica*
- *Linha de comando*
- *Arquivos CHD*

4.1.1 Interface gráfica

Para iniciar a interface gráfica basta iniciar o executável clicando duas vezes no **mame** caso esteja usando o Windows, caso esteja no terminal do Linux e do macOS e esteja no mesmo diretório onde se encontra o binário do MAME, você o executa com o comando `./mame`. Independente do Sistema Operacional, caso o binário esteja no PATH do seu sistema, ele pode ser executado simplesmente digitando `mame`.

Para os marinheiros de primeira viagem o MAME pode parecer um pouco complexo, porém é bem mais simples do que parece.

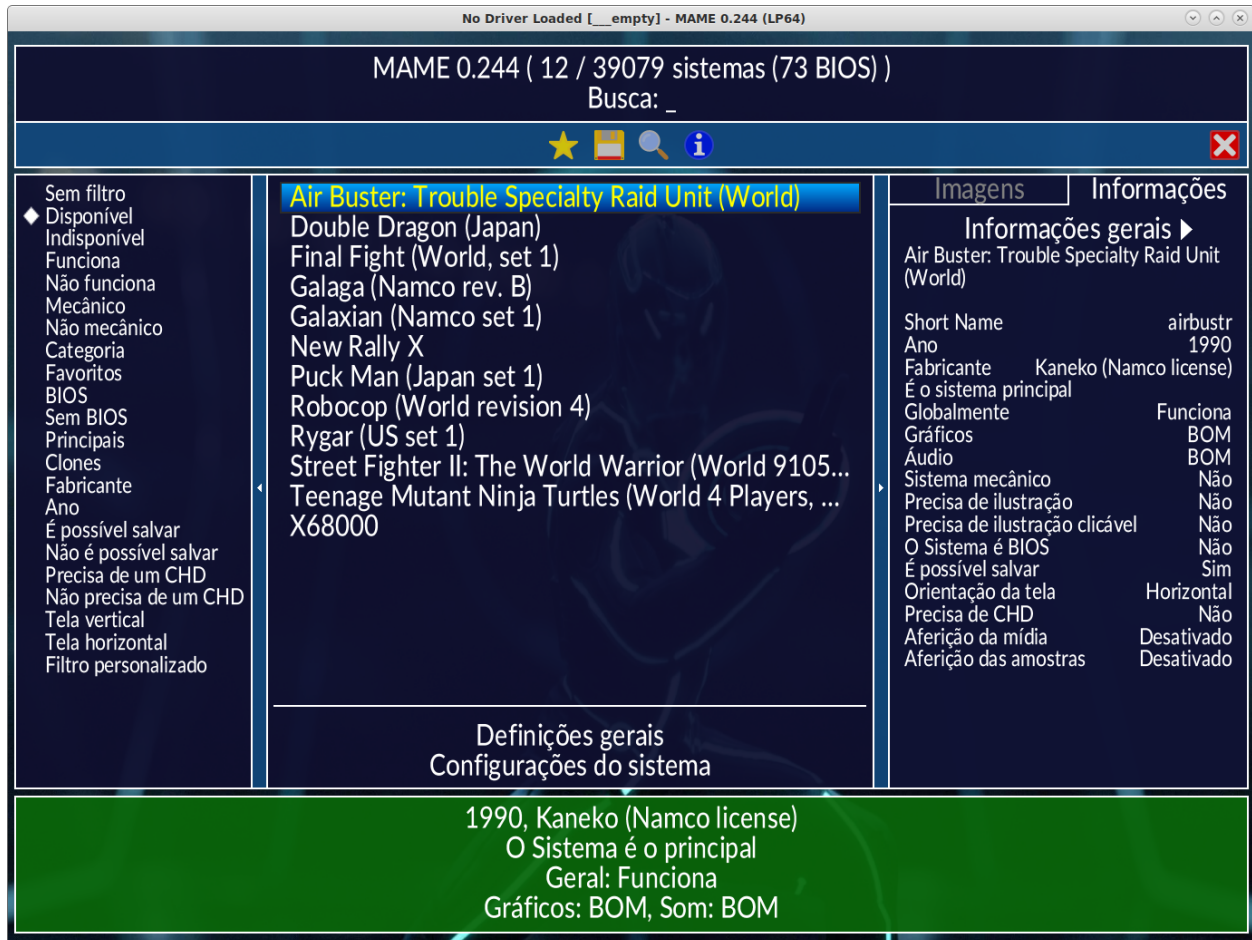
Inicie o MAME clicando duas vezes no executável:



Boa parte da interface já foi traduzida, para trocar o idioma da interface gráfica, vá em *General Settings --> Customize UI*, clique duas vezes em *Language*, role para baixo e selecione *Portuguese (Brazil)*. Retorne e faça um clique duplo em *Salva as definições*.

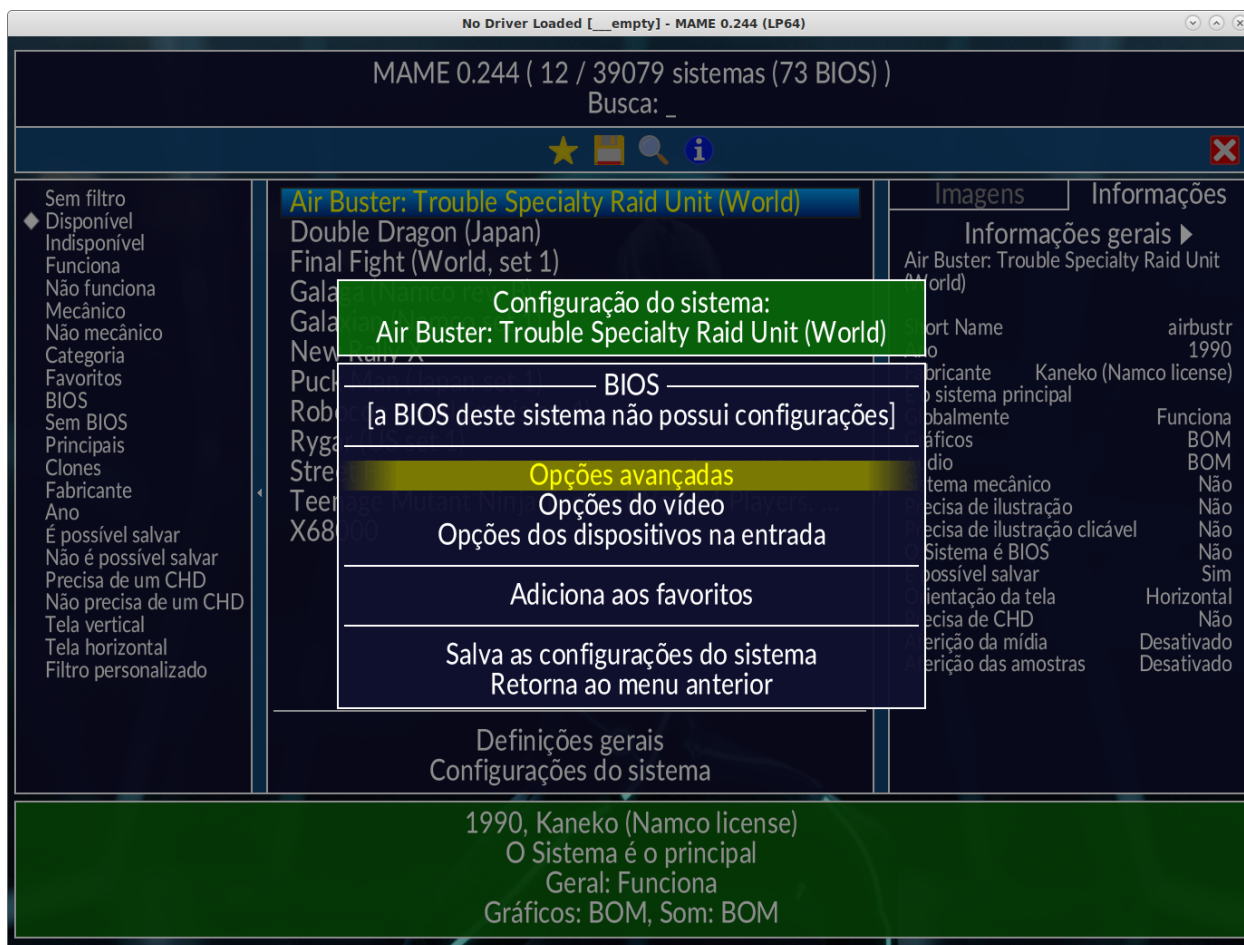
A versão da tradução compatível com este documento pode ser encontrado no site do [mamedoc](http://mamedoc.org) ou mais precisamente [aqui](#).

Baixe o arquivo `strings.mo` e copie-o para o diretório `language\Portuguese_Brazil`, substituindo o que já está lá.



Ao clicar com o botão direito do mouse no nome do sistema, é possível configurar o sistema individualmente caso seja necessário. Tais configurações ficam armazenadas em arquivos `<nome_do_sistema>.ini`, no Windows fica no mesmo diretório do executável do MAME e no Linux/macOS fica em `~/ .mame`.

Apesar de ser possível fazer algumas definições através da interface, nem todas as *configurações do MAME* estão acessíveis pela interface, configurações mais específicas ainda precisam ser feitas através da linha de comando ou por arquivos `*.ini` específicos. Consulte *Diversos arquivos de configuração* para obter mais informações.



Ainda é possível fazer personalizações visuais como adicionar uma *imagem de fundo*, adicionar ícones ao lado dos nomes dos sistemas:



Tais ícones são mantidos fora do projeto MAME, eles são mantidos pelo [progetto-SNAPS](#). A instalação é simples, baixe o arquivo mais recente e extraia o diretório **icons** e **folders** dentro do diretório do MAME, caso apareça algum erro de arquivo já existente basta sobrescrever. Agora ao iniciar o MAME o nomes dos sistemas mostram um ícone.

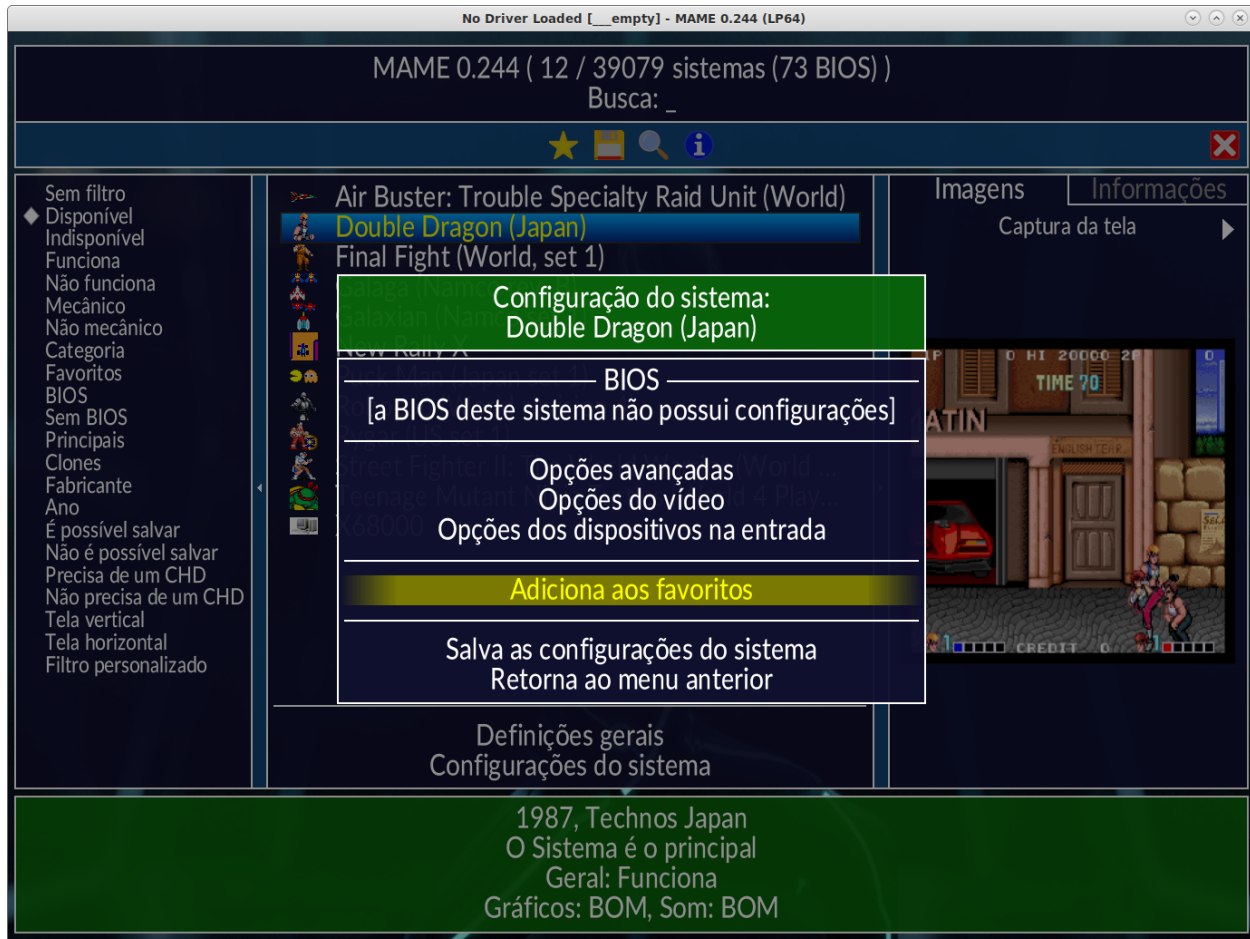
Note porém que quanto mais "coisas" forem adicionadas, o MAME vai levar mais tempo para carregar tudo para exibi-los na interface principal, porém, isso não altera em nada o funcionamento e a performance do MAME quando for executado através da linha de comando ou por uma *fron-tend*.

Para iniciar um sistema na lista que aparece ao centro da tela clique 2x no nome do sistema desejado. Pressione F12 durante a execução do sistema para que seja salvo uma **captura da tela** na pasta **snap**. Ao encerrar a emulação a imagem da tela vai aparecer do lado direito da interface do MAME.



Ainda no site do **progetto-SNAPS** é possível encontrar imagens para completar todas as abas dentro de *Imagens* como *Gabinetes*, *Painéis de controle*, *PCI* e vários outros. Porém quanto mais itens visuais forem adicionados, mais e mais o MAME vai demorar para iniciar por causa dos arquivos extras, leve isso em consideração.

Os sistemas que você mais gosta de jogar podem ser adicionadas aos *Favoritos*, na interface durante a seleção do sistema, clique com o botão direito em cima do nome do sistema e selecione a opção *Adiciona aos favoritos*. Caso tenha iniciado o MAME pela linha de comando ou caso o sistema já esteja rodando, pressione Tab para abrir a tela de opções e escolha *Adiciona aos favoritos*.



Todos os sistemas favoritadas ficam armazenados num arquivo `favorites.ini` dentro do diretório `ui` que fica no mesmo diretório do executável do MAME.

4.1.2 Linha de comando

Com o MAME no \$PATH para usuários macOS/Linux ou %PATH% para usuários Windows, inicie a interface do MAME com o comando:

```
mame
```

Para iniciar um sistema faça:

```
mame <ROM>
```

Para iniciar um sistema específico junto com um software faça:

```
mame <sistema> <software>
```

Em **sistema**, escolha um sistema numa lista de centenas de sistemas emulados, já em **software** é o nome do sistema¹ que deseja rodar. O software nada mais é do que um nome de uma ROM ou a imagem de uma mídia. É possível também a utilização de um [*catálogo de programas*] (**softlists**), no catálogo tem a lista de programas compatíveis (ou não) no banco de dados interno do MAME com o sistema em questão.

Para rodar o **Sonic the Hedgehog** na versão do **Sega Genesis Americano**, por exemplo, execute o comando como mostrado abaixo:

```
mame genesis sonic
```

Alternativamente, também é possível iniciar MAME apenas com o sistema:

```
mame genesis
```

Selecione *cartridge (cart)*, depois [*catálogo de programas*] para ver uma lista de qual deseja iniciar. Para que essa lista funcione é necessário ter um conjunto ROMs compatíveis no diretório **roms** ou outro que você definir. Observe que muitas cópias de ROMs antigas, de fitas e de discos que funcionavam em versões anteriores, podem não mais serem reconhecidas pelas versões mais novas do MAME exigindo algum tipo de atualização. Consulte [As ROMs dos itens dos programas](#) para obter mais informações sobre o catálogo de programas.

A estrutura básica da linha de comando fica assim:

```
mame <sistema> <mídia> <software> <opções>
```

Onde:

- <systema> é o apelido ou o nome encurtado do sistema que deseja emular como o nes, snes, genesis, c64, etc.
- <mídia> é o seletor da mídia que deseja carregar, continue lendo para mais detalhes.
- <software> é o programa ou jogo que deseja carregar (também pode ser usado o caminho completo para o arquivo que será carregado ou como o nome abreviado do arquivo que esteja na sua lista de software).
- <opções> quaisquer opções adicionais usadas na linha de comando para os controles, para o vídeo, áudio, etc.

¹ Os desenvolvedores do MAME preferem usar o termo **sistemas** em vez de **jogos**, talvez visando evitar problemas legais? O termo "máquina" foi alterado [neste commit](#) para sistema.

Lembre-se que ao digitar um nome de um sistema <sis²tema> e este ainda não exista ou não seja emulado, o MAME irá sugerir algumas opções próximas ao que foi digitado. No caso do desconhecimento de quais <mídias> estão disponíveis, inicie a emulação como mostra o exemplo abaixo:

```
mame <sis2tema> -listmedia <opções>
```

Supondo que ainda seja o **Sega Genesis Americano**:

```
mame genesis -listmedia
```

O MAME vai retornar algo como:

SYSTEM	MEDIA NAME	(brief)	IMAGE FILE EXTENSIONS	SUPPORTED
genesis	cartridge	(cart)	.smd .bin .md .gen	

Assim somos informados que o sistema aceita cartucho (*cartridge*) através da opção `-cart` e os arquivos ROMs com extensões **.smd**, **.bin**, **.md** e **.gen**. Traduzindo isso para um comando nós teríamos algo como:

```
mame genesis -cart /caminho/completo/rom.bin
```

Ou usando o nome de uma ROM compatível listada no catálogo:

```
mame genesis -cart sonic
```

Além dos consoles o MAME consegue emular computadores como o MSX por exemplo, na época dos anos 80 alguns devem se lembrar do **Expert XP-800** da Gradiente (`expert10`) ou o **HotBit HB-8000** da Sharp (`hotbit11`), além destes o MAME consegue emular uma lista enorme de outros modelos MSX, para ver a lista completa dos modelos faça:

```
mame expert10 -lb
```

Source file:	Name:	Parent:
msx/msx1.cpp	ax150	
msx/msx1.cpp	ax170	
...		
msx/msx1.cpp	expert10	expert13
msx/msx1.cpp	expert11	expert13
msx/msx1.cpp	expert13	
msx/msx1.cpp	expert20	
msx/msx1.cpp	expert3i	
msx/msx1.cpp	expert3t	
msx/msx1.cpp	expertac	
msx/msx1.cpp	expertdp	
msx/msx1.cpp	expertdx	
msx/msx1.cpp	expertpl	
...		

Consulte o [driver MSX1](#) do MAME para obter mais detalhes como o ano de lançamento, nome completo, etc. Aqui o [driver MSX2](#).

² Existe uma diferença entre driver e sistema, o comando em questão funciona apenas com sistemas. Os arcades são considerados drivers como o CPS1, CP2, ZN, etc; os jogos dentro dos drivers são chamados de sistemas. O comando ao ser usado com um sistema irá retornar um erro "Unknown system". (Nota do tradutor)

Vamos iniciar o **Expert XP-800** da Gradiente (expert10), porém, verifique antes se a ROM existe:

```
mame -verifyroms expert10
romset expert10 [expert13] is good
1 romsets found, 1 were OK.
```

Caso apareça algo de diferente como:

```
romset "expert10" not found!
```

Será preciso localizar o arquivo na internet, e qual seria ele? Pergunte ao MAME que ele ajuda:

```
mame expert10 -lr
ROMs required for driver "expert10".
Name                                     Size Checksum
expbios.rom                             32768 CRC(07610d77)
SHA1(ef3e010eb57e4476700a3bbff9d2119ab3acdf62)
```

No caso, precisamos do arquivo **expbios.rom**, ele fica dentro do arquivo **expert10.zip** e este arquivo deve estar dentro da pasta **roms**.

Agora que está tudo certo e funcionando faça:

```
mame expert10
```

Para identificar quais são os formatos de mídia compatíveis com este sistema faça:

```
mame expert10 -lm
```

SYSTEM	MEDIA NAME	(brief)	IMAGE FILE EXTENSIONS SUPPORTED
expert10	printout	(prin)	.prn
expert10	cassette	(cass)	.wav .tap .cas
expert10	cartridge1	(cart1)	.mx1 .bin .rom
expert10	cartridge2	(cart2)	.mx1 .bin .rom

O MAME mostra que ele é compatível com as opções -prin, -cass, -cart1 e -cart2 e as suas respectivas extensões.

Para ver a lista de slots compatíveis faça:

```
mame expert10 -lslot
```

SYSTEM	SLOT NAME	SLOT OPTIONS	SLOT DEVICE NAME
expert10	centronics	ap2000	Epson ActionPrinter 2000
		chessmec	Tasc ChessMachine EC Interface
		covox	Covox Speech Thing
		covox_stereo	Covox (Stereo-in-1)
		ex800	Epson EX-800
		lx800	Epson LX-800
		lx8101	Espon LX-810L
		p72	NEC PinWriter P72
		pl80	COMX PL-80
		printer	Centronics Printer
		samdac	SAMDAC
		smartboard	Tasc SmartBoard SB30 Interface

(continua na próxima página)

(continuação da página anterior)

cartslot1	bm_012	MSX Cartridge - BM-012
	moonound	MSX Cartridge - MoonSound
cartslot2	bm_012	MSX Cartridge - BM-012
	moonound	MSX Cartridge - MoonSound

Para usar quaisquer itens da lista de slots como o **Covox Speech Thing** por exemplo, faça:

```
mame expert10 -centronics covox
```

O slot *cartslot1* e *cartslot2* são para os dois cartuchos, para usar o **covox** junto com o cartucho de **midi** e o **moonound** faça:

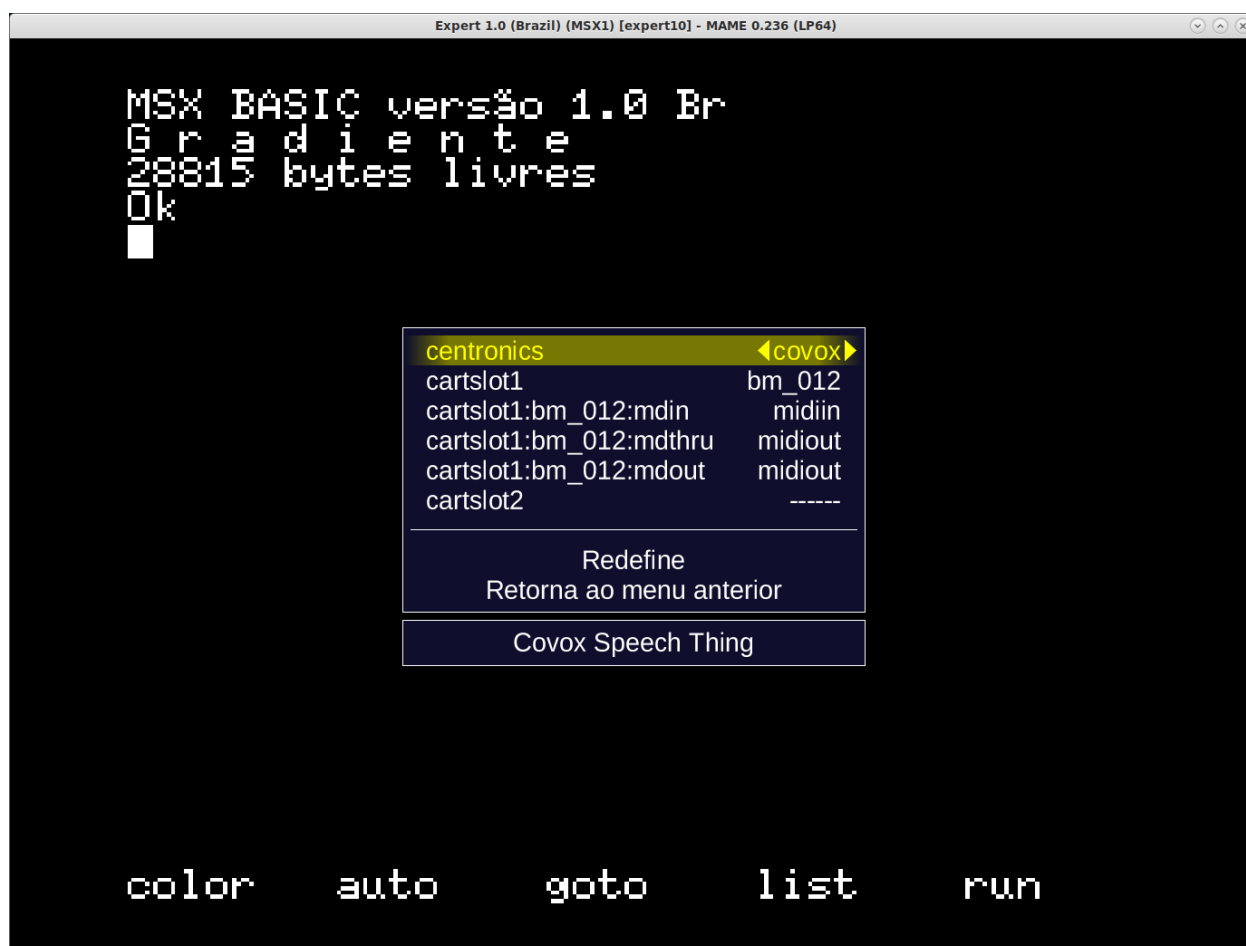
```
mame expert10 -centronics covox -cartslot1 bm_012 -cartslot2 moonound
```

Caso use um sintetizador de MIDI externo como o [VirtualMIDISynth](#) (Windows) ou qualquer outro que tenha no seu sistema (descubra qual usando a opção *-listmidi*), você o conecta com o comando:

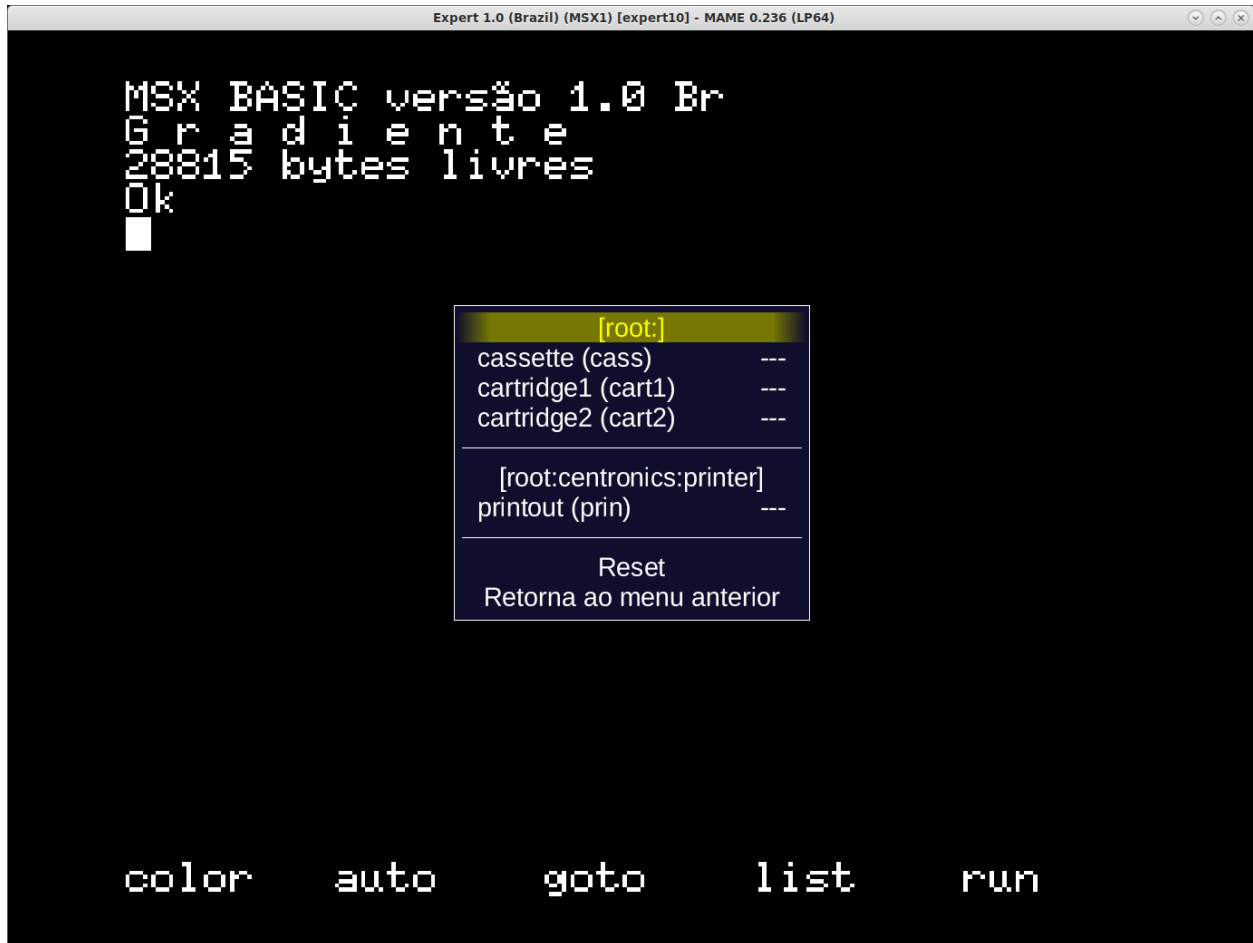
```
mame expert10 -centronics covox -cartslot1 bm_012 -midiout1 "VirtualMIDISynth #1" -cartslot2 ↵
↵moonound
```

Também é possível selecionar os mesmos itens através da interface do MAME, depois de iniciar o **expert10** pressione **ScrLk** para usar o teclado do seu computador, pressione **Tab** e vá em *Dispositivos de slot*:

Repare que todos os itens compatíveis com o sistema também estão disponíveis pela interface ao clicar nas setas esquerda e direita do teclado. Para surtir efeito é preciso selecionar os itens desejados e clicar duas vezes em *Redefine* para que o MAME seja reiniciado com estas opções definidas.



Através do *Gerenciador de arquivos* é possível carregar arquivos avulsos, talvez uma gravação de uma fita k7 feita por você, a imagem de algum cartucho que só você tenha, que tenha baixado, etc.

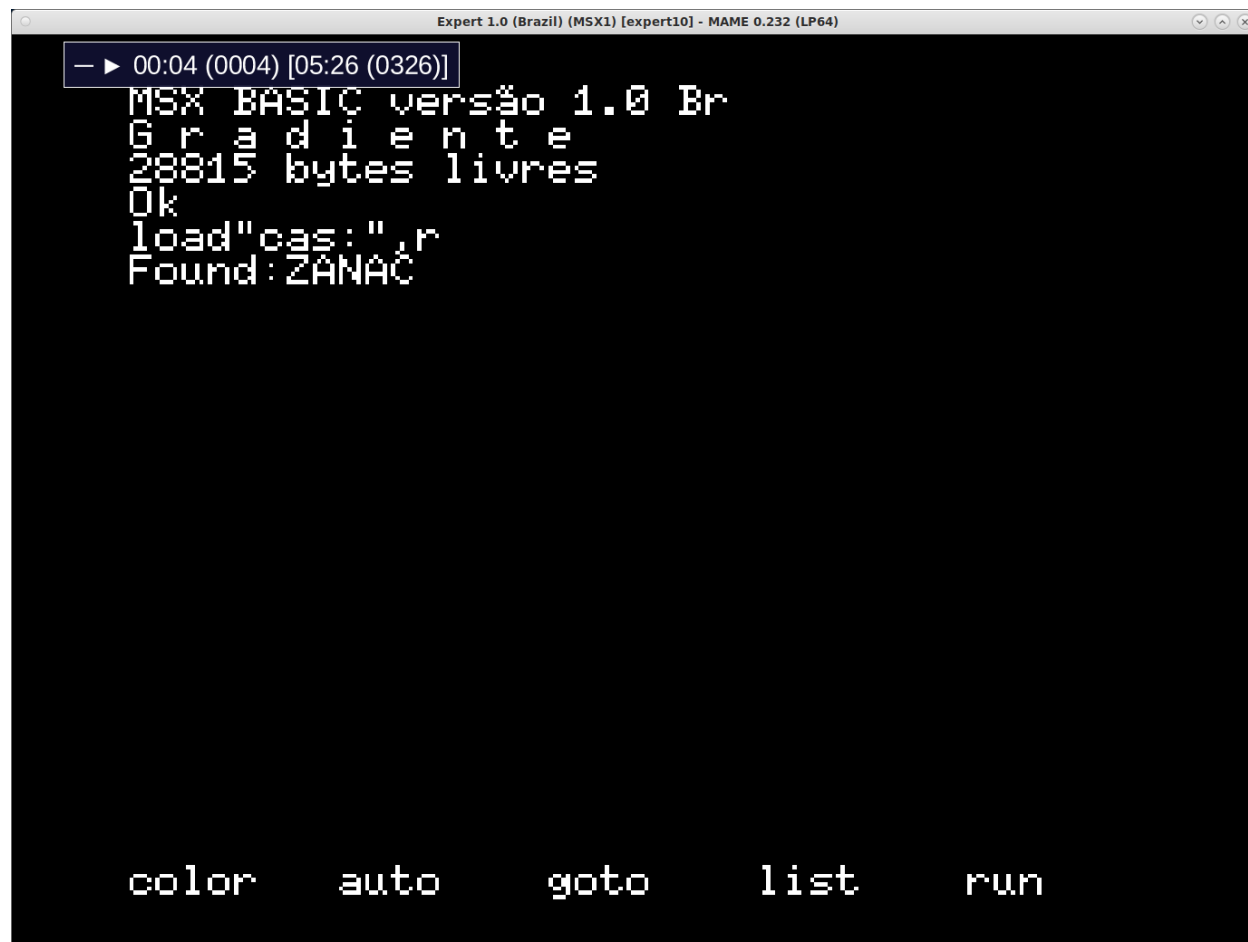


Para usar, clique duas vezes na opção desejada e navegue até o caminho onde a sua imagem está, os formatos compatíveis foram listados alguns parágrafos acima. Assim como foi feito anteriormente, é preciso reiniciar para que os itens estejam disponíveis no sistema emulado.

Para carregar uma imagem de uma fita k7 por exemplo, faça:

```
mame expert10 -cass msx/zanac.cas
```

Observe que `msx/zanac.cas` é uma pasta chamada **msx** dentro do diretório onde o executável do MAME se encontra, se os seus arquivos estão num outro lugar, informe o caminho completo para este arquivo. Já na interface do sistema carregue a imagem com o comando `load"cas:",r`, se imagem for compatível com o sistema emulado deverá aparecer **Found: ZANAC** ou qualquer outra imagem que esteja sendo carregada. Caso a mensagem **Found:** não apareça, significa que talvez a imagem seja incompatível com esta versão do MSX ou até mesmo a imagem está com algum problema, não foi capturada corretamente, etc.



No topo da tela há o contador e a leitura da imagem é feita em tempo real, assim como é feito no hardware físico. Para acelerar o processo, clique em `SrcLk` para usar o teclado do computador e pressione `PgDn` no Linux ou `:kbd`Insert`` no Windows ou as teclas `F8` e `F9` para acelerar a emulação e com isso acelerar o processo de leitura. Ao terminar, pressione `F9` até aparecer *auto (0/10) 100%* no topo da tela para que a emulação volte para a velocidade normal.

Este foi apenas um simples exemplo de centenas de outros sistemas que o MAME pode emular como o Macintosh, X68000, etc.

Precisando de ajuda?

mame -help

O comando exibe algumas opções básicas de uso, a versão do MAME e outras informações úteis.

mame -showusage

Mostra uma lista (bastante longa) das opções disponíveis na linha de comando. As opções principais são descritas na seção [Índice das opções para a linha de comando](#) deste manual.

mame -showconfig

Mostra uma lista (bastante longa) das opções de configuração que estão sendo usadas pelo MAME. Essas configurações sempre podem ser modificadas na linha de comando ou editadas diretamente no arquivo de configuração `mame.ini`, este é o arquivo primário de configuração do MAME. É possível encontrar uma descrição de algumas opções de configuração na seção [Índice das opções para a linha de comando](#) deste manual (na maioria dos casos, cada opção de configuração listada ali, possui uma versão equivalente para a linha de comando).

Também é possível utilizar a opção `-showconfig` quando o MAME já está devidamente configurado para listar uma configuração do `mame.ini` sem que você precise abri-lo, por exemplo, para listar o seu **rompath** no **Windows** faça:

```
mame -showconfig|findstr rompath
rompath                roms;D:\mame\roms
```

No Linux ou macOS faça:

```
mame -showconfig|grep rompath
rompath                roms;/mount/media/mame/roms;etc
```

Algumas vezes é bem mais prático fazer assim do que ter que acessar o arquivo `mame.ini` diretamente.

4.1.3 Arquivos CHD

Além das ROMs, alguns sistemas também precisam de arquivos **.CHD** para funcionarem. Estes arquivos são cópias das mídias originais (HDD, CD, DVD, etc.) compactadas num formato proprietário, para mais detalhes consulte [ROMs e CHDs](#).

O MAME procura por estes arquivos dentro do diretório **roms**, eles devem estar dentro de diretórios próprios com o mesmo nome da ROM a qual os arquivos CHD's estão associados, por exemplo, o sistema **Street Fighter III: New Generation (Euro 970204)** precisa que a ROM `sfiii.zip` esteja dentro da pasta **roms** e o arquivo **.CHD** `cap-sf3-3.chd` precisa estar dentro de um diretório com o mesmo nome da ROM, ou seja, uma pasta `sfiii` dentro da pasta **roms**:

```
roms
|
+-sfiii.zip
|
+-sfiii
  +-cap-sf3-3.chd
```

4.2 A interface do usuário

- *Introdução*
- *Navegando pelos menus*
 - *Usando um controle*
 - *Usando um mouse ou trackball*
 - *Usando uma tela sensível ao toque*
- *Configurando as entradas*
 - *Configurações da entrada digital*
 - *Configurações da entrada analógica*
- *O menu de seleção de programa e de sistema*
 - *Controles de navegação*
- *O menu de seleção simplificada do sistema*

4.2.1 Introdução

O MAME oferece uma simples interface ao usuário para que seja intuitivo selecionar, rodar um sistema ou um programa; assim como, alterar as configurações durante a emulação de um sistema. A interface foi projetada para ser utilizável com um teclado, com um controle ou com algo que seja possível apontar na tela (mouse, pistola, etc.), porém, um teclado é necessário para realizar as configurações iniciais.

As configurações predefinidas mais importantes da interface são as seguintes:

ScrLk (Scroll Lock) ou Forward Delete no macOS (IU Alterna)

Durante a emulação de sistemas com teclado, ativa ou desativa os controles da interface. O MAME iniciar com os controles da interface desativados nos sistemas que usam teclados, a não ser que a opção *ui_active* esteja ligada.

Tab (Menu de configuração)

Mostra ou esconde o menu durante a emulação.

Esc (UI Retorna)

Retorna para a seleção de sistemas ou encerra o MAME caso ele tenha sido iniciado com um determinado sistema (a partir da linha de comando ou caso esteja usando um *front-end* externo).

4.2.2 Navegando pelos menus

É predefinido que seja possível navegar pela interface do MAME usando os cursores do teclado. Todos os controles da interface podem ser alteradas na opção *Todas as entradas --> Interface do usuário*. Como padrão, o MAME utiliza o US ANSI QWERTY como mapeamento do teclado, consulte o capítulo [Comparativo entre os mapas de teclado](#) para ver as diferenças entre o teclado Americano e o ABNT-2 usado no Brasil.

As configurações correspondentes com o mapeamento padrão do teclado são:

Seta cima (IU Cima)

Destaca o item anterior ou o último caso o primeiro esteja destacado.

Seta baixo (IU Baixo)

Destaca o próximo item da lista ou o primeiro caso o último esteja destacado.

Seta esquerda (IU Esquerda)

Para itens do menu que podem ser ajustados, reduz o valor ou seleciona o valor anterior (tais configurações mostram setas indicativas).

Seta direita (IU Direita)

Para itens do menu que podem ser ajustados, aumenta o valor ou seleciona o valor posterior (tais configurações mostram setas indicativas).

Enter / Return e Enter do teclado numérico (UI Seleciona)

Seleciona o item em destaque.

Forward Delete ou Fn + Delete em alguns

teclados compactos (UI Limpa) Limpa a configuração ou redefine para o valor predefinido

Esc (UI Retorna)

Limpa o campo de busca, caso contrário, fecha o menu, retorna ao menu anterior ou retorna para a emulação no menu principal (também há um item na parte do menu que funciona igual).

Home (UI Home)

Destaca o primeiro item e rola para o topo do menu.

End (UI End)

Destaca o último item e rola para a parte debaixo do menu.

PgUp (UI Pág. cima)

Rola a página do menu para cima.

PgDn (UI Pág. baixo)

Rola a página do menu para baixo.

[(UI Grupo anterior)

Move os itens do grupo anterior (não é utilizado em todos os menus).

] (UI Próximo grupo)

Move o próximo item do grupo (não é utilizado em todos os menus).

Usando um controle

É possível navegar na interface do MAME usando um controle ou joystick. Por predefinição, apenas os controles mais importantes da interface têm atribuições do joystick:

- Mova o controle do joystick para cima ou para baixo no eixo y para destacar próximo item ou o item anterior.
- Mova o controle do joystick para esquerda ou para direita no eixo x para ajustar a configurações.
- Pressione o primeiro botão no primeiro joystick para selecionar o item destacado.
- Se o primeiro joystick tiver pelo menos três botões, aperte o segundo botão do primeiro joystick para fechar e retornar ao menu anterior, ou retornando ao sistema que estiver sendo emulado (geralmente tem um item na parte inferior do menu para o mesmo fim).

Para controles no estilo *gamepad*, o botão analógico esquerdo e o teclado direcional geralmente controlam a navegação da interface do usuário. Dependendo do controle, o botão analógico direito, os gatilhos e os botões adicionais podem ser atribuídos automaticamente às entradas da interface do usuário. Verifique o menu de atribuições da entrada da interface do usuário para ver como os controles estão atribuídos.

Para que seja possível usar o MAME com um controle joystick sem um teclado, é preciso definir os botões do joystick (ou a combinação dos seus botões) para estes controles também:

- **Mostra/esconde o menu / Tab**
Para mostrar ou dispensar o menu durante a emulação
- **IU retorna**
Para fechar os menus
- **IU cancela**
Para fechar os menus, retornar para a tela de seleção, para encerrar a emulação ou para fechar o MAME.
- **IU limpa**
Não é basicamente essencial para a emulação, porém é usado para limpar ou redefinir algumas configurações.
- **IU home, IU fim, IU sobe a página, IU desce a página,**
Grupo anterior da IU e Próximo grupo da IU Não são essenciais, contudo, tornam a navegação mais fácil em alguns menus.

Caso não esteja usando um front-end externo para rodar os sistemas no MAME, atribua os botões do joystick (ou combinações dos botões) nestes controles para fazer pleno uso dos menus de seleção do sistema:

- **IU Próx. foco, IU Foco ant.**
Para navegar entre os painéis.
- **IU Adiciona/Remove favoritos, IU Exporta lista e IU Audita mídia**
Caso queira acessar estes recursos sem um teclado ou mouse.

Usando um mouse ou trackball

MAME suporta a navegação através dos menus usando um mouse ou *trackball* que funciona como um dispositivo apontador no sistema:

- Clique nos itens do menu para destacá-los.
- Faça um clique duplo para selecioná-los
- Clique nas setas da esquerda/direita para ajustar as configurações.
- Para menus com muitos itens que não caibam na tela, use as setas cima/baixo para rolar as opções.
- Use gestos de rolagem vertical para rolar os menus ou as caixas de texto com muitos itens para que elas caibam na tela.
- Clique nos itens da barra de ferramentas para selecioná-los ou passe o mouse sobre eles para ver uma descrição.

Caso tenha botões adicionais no mouse, talvez queira atribuir combinações dos botões para o *Mostra/esconde o menu* / *Tab*, *Pausa* e/ou *Cancela* para que seja possível usar o MAME sem usar um teclado.

Usando uma tela sensível ao toque

O MAME tem um suporte básico para navegar pelos menus usando uma tela sensível ao toque:

- Toque nos itens do menu para destacá-los.
- Toque duplo nos itens do menu para selecioná-los.
- Deslize para a esquerda ou para a direita (horizontalmente) no item destacado do menu para ajustar a configuração, quando for aplicável.
- Deslize para cima ou para baixo (verticalmente) para rolar os menus ou caixas de texto com muitos itens que não cabem na tela.
- Para menus ou caixas de texto com muitos itens ou linhas que não cabem na tela, pressione o triângulo que aponta para cima ou para baixo na parte superior ou inferior para rolar para cima ou para baixo.

Note que na versão SDL do MAME, a opção de *tela sensível ao toque* deve ser ativada para esta função funcione.

4.2.3 Configurando as entradas

O MAME precisa de um sistema de entrada flexível para sustentar todos os mecanismos de controle da vasta gama de sistemas emulados por ele. Nas entradas que têm apenas dois estados distintos, *ligado* e *desligado* ou *ativo* e *inativo*, estas entradas são chamadas de digitais. Todas as outras entradas são chamadas de analógicas, mesmo que isso não seja estritamente verdadeiro.

Para atribuir os controles da interface do usuário do MAME ou as entradas predefinidas em todos os sistemas, selecione *Entrada (geral)* no menu principal durante a emulação ou selecione *Configurações* do menu de seleção do sistema, em seguida, selecione *Todas as entradas* e a partir daí, selecione uma categoria.

Para atribuir entradas no sistema em funcionamento, selecione no menu principal a opção *Entrada (este sistema)* durante a emulação. As entradas estão agrupadas por dispositivo e ordenadas por tipo. É possível mover entre os dispositivos com o próximo grupo e as teclas/botões do grupo anterior usando as teclas [e].

Os menus de atribuição da entrada mostram o nome da entrada emulada ou o controle da interface do usuário à esquerda, a entrada (ou combinação das entradas) à direita.

Para ajustar a sensibilidade, a velocidade da centralização automática, as configurações da inversão ou para ver como os controles analógicos emulados reagem, selecione *Controles analógicos* no menu principal durante a emulação. (Este item só aparece nos sistemas com controles analógicos).

Configurações da entrada digital

Cada entrada digital emulada tem uma única atribuição configurável. Visando uma maior flexibilidade, o MAME pode combinar as entradas do host (as teclas, os botões e os eixos do joystick) usando operações lógicas and, not e or. Isso fica melhor ilustrado com alguns exemplos:

Tecla 1

Neste simples exemplo, ao pressionar a tecla 1 no teclado, ativa a entrada emulada ou o controle da interface do usuário.

Tecla direcional Baixo ou baixo no direcional do joystick 1

Pressionando a seta para baixo no teclado ou no controle, ativa a entrada emulada ou o controle da interface do usuário.

Tecla P

Ao pressionar a tecla P do teclado ao mesmo tempo que **não** for pressionado Shift, a tecla ativa a entrada emulada ou o controle da interface do usuário. O MAME não mostra operações implícitas and.

Tecla P + Shift esquerdo ou P + Shift direito

Ao pressionar a tecla P do teclado ao mesmo tempo que pressiona as teclas Shift, ativa a entrada ou o controle da interface do usuário. Novamente, as operações implícitas and não são mostradas.

Tecnicamente, o MAME utiliza a soma booleana da lógica dos produtos para combinar as entradas.

Quando uma configuração para a entrada digital é destacada, o quadro abaixo do menu mostra se a seleção irá definir a atribuição ou anexar uma operação or a ela. Pressione IU Esquerda/Direita antes de selecionar se a configuração será para alternar entre a configuração ou se será para anexar uma operação or. Pressione Del ou forward delete para excluir a configuração ou restaurar a atribuição original.

Ao selecionar uma configuração de entrada digital, o MAME esperará que você digite uma entrada ou uma combinação das entradas para uma operação lógica and:

- Pressione uma tecla, um botão ou mova o controle analógico uma vez para adicionar uma operação and.
- Pressione uma tecla, um botão ou mova o controle analógico duas vezes para adicionar uma operação not na operação and. Ao pressionar a mesma tecla, botão ou movendo o mesmo controle analógico mais de uma vez, isso faz com que se ligue ou desligue a operação not várias vezes.
- Ao pressionar Esc **antes** restaura a configuração original.
- A nova configuração é mostrada abaixo do menu. Aguarde cerca de um segundo depois da ativação para que a nova configuração seja aceita.

Veja aqui como criar algumas configurações de exemplo:

Tecla 1

Pressione uma vez a tecla 1 no teclado, aguarde 1 segundo para aceitar a nova configuração.

Tecla F12 Shift Alt

Pressione uma vez a tecla F12 no teclado, pressione uma vez a tecla Shift esquerda, pressione uma vez a tecla Alt esquerda, aguarde cerca de um segundo para que a nova configuração seja aceita.

Tecla P

Pressione uma vez a tecla P, pressione duas vezes a tecla Shift esquerda, pressione duas vezes a tecla Shift direita, aguarde cerca de um segundo para que a nova configuração seja aceita.

Configurações da entrada analógica

Cada entrada analógica emulada possui três configurações de atribuição:

- Use a configuração do *axis setting* (ajuste do eixo) para atribuir um eixo analógico como controle de uma entrada analógica. As configurações do eixo utiliza o nome da entrada com o sufixo "Analog" (Analógico). O ajuste do eixo para o volante no sistema *Ridge Racer* por exemplo, é chamado de *Steering Wheel Analog*.
- Use o *increment setting* para atribuir à entrada (ou na combinação delas) o aumento do seu valor. A configuração para este incremento utiliza o nome *Analog Inc*. Por exemplo, a configuração de incremento no volante do sistema *Ridge Racer* se chama *Steering Wheel Analog Inc*. Esta é a entrada digital para este sistema, caso um eixo analógico seja atribuído a ele, o MAME não vai incrementar o valor emulado numa velocidade proporcional.
- Use o *decrement setting* para atribuir à entrada (ou na combinação delas) a redução do seu valor. A configuração para este incremento utiliza o nome *Analog Dec*. Por exemplo, a configuração de incremento no volante do sistema *Ridge Racer* se chama *Steering Wheel Analog Dec*. Esta é a entrada digital para este sistema, caso um eixo analógico seja atribuído a ele, o MAME não vai incrementar o valor emulado numa velocidade proporcional.

Os ajustes de aumento e de redução são muito mais úteis para controlar uma entrada analógica usando controles digitais (as teclas do teclado, os botões do joystick ou um teclado direcional por exemplo). Eles são configurados da mesma maneira que as entradas digitais (*ver acima*). **É de extrema importância que não se atribua o mesmo controle ao ajuste do eixo, assim como os ajustes para o aumento e/ou para a redução na mesma entrada ao mesmo tempo.** Por exemplo, caso atribua o ajuste analógico *Steering Wheel Analog* do *Ridge Racer* ao eixo X ao analógico esquerdo no seu controle, você não deve atribuir nem o ajuste analógico *Steering Wheel Analog Inc* nem o ajuste *Steering Wheel Analog Dec* ao eixo X do mesmo analógico.

É possível atribuir um ou mais eixos analógicos ao ajuste do eixo de uma entrada analógica emulada. Quando diversos eixos são atribuídos num ajuste do eixo, eles são adicionados juntos, porém os controles de posição absoluta anularão os controles de posição relativa. Por exemplo, suponha que para o sistema **Arkanoid** você atribua o ajuste do eixo guilabel:*Dial Analog* ao *Mouse X* ou *Joy 1 LSX* ou *Joy 1 RSX* ao mouse num controle estilo Xbox. Será possível controlar a palheta com o mouse ou com o eixo analógico, porém o mouse só terá efeito caso ambos os eixos analógicos estejam na posição neutra (centralizados) no eixo X. Se qualquer um dos eixos analógicos não estiver devidamente centralizado no eixo X, o mouse não surtirá qualquer efeito, pois um mouse é um controle de posição relativa, enquanto um joystick é um controle de posição absoluta.

Para os controles de posição absoluta como joysticks e pedais, O MAME permite a atribuição do alcance total de um eixo ou o alcance de um lado da posição neutra (*meio eixo*) ao ajuste de um eixo. A atribuição de um "meio eixo" é usada normalmente nos pedais ou nas outras entradas absolutas onde a posição neutra está numa extremidade do alcance da faixa de entrada. Vamos supor que para o sistema **Ridge Racer** você atribua o ajuste analógico do pedal de freio à parte de um eixo vertical do joystick abaixo da posição neutra. Caso controle esteja na posição neutra na vertical ou acima dela, o pedal do freio será liberado; já se o joystick estiver abaixo da posição neutra na vertical, o pedal do freio será aplicado proporcionalmente. Metade dos eixos são exibidos com o nome do eixo seguido por um sinal de mais ou menos (+ ou -). O sinal de **mais** se refere à parte do eixo abaixo ou à direita da posição neutra; o sinal de **menos** se refere à parte do eixo acima ou à esquerda da posição neutra. Para os controles dos pedais ou do gatilho analógico, a faixa ativa é tratada como estando acima da posição neutra (a metade do eixo indicada por um sinal de **menos**).

Quando as teclas ou os botões são atribuídos a um ajuste do eixo, eles ativam condicionalmente os controles analógicos que forem atribuídos ao ajuste. Isto pode ser usado em conjunto com um controle de posição absoluta para criar um controle "*pegadiço*".

Aqui estão alguns exemplos de algumas das atribuições possíveis para o ajuste dos eixos assumindo que seja usado um controle do Xbox e um mouse:

Joy 1 RSY

Faça um movimento vertical no analógico direito para atribuir essa entrada à emulação.

Mouse X ou Joy 1 LT ou Joy 1 RT Reverse

Faça um movimento horizontal, ou ative os gatilhos esquerda e direita para atribuir essa entrada à emulação. O gatilho direito é invertido, assim sendo, ele trabalha na direção inversa em relação ao gatilho esquerdo.

Joy 1 LB Joy 1 LSX

Faça um movimento horizontal no analógico esquerdo para atribuir essa entrada à emulação, porém *apenas* enquanto estiver mantendo pressionado o botão superior esquerdo. Caso o botão superior direito seja liberado enquanto o analógico esquerdo não estiver centralizado no eixo horizontal, a entrada da emulação manterá seu valor até o botão superior direito seja pressionado novamente (um controle "pegadiço").

não Joy 1 RB Joy 1 RSX ou Joy 1 RB Joy 1 RSX Inverso

Faça um movimento horizontal no analógico direito para atribuir essa entrada à emulação, mas inverta o controle caso o botão superior direito seja mantido pressionado.

Ao selecionar a configuração de um eixo, o MAME vai aguardar a sua resposta:

- Movimente um controle analógico para atribuí-lo ao ajuste de um eixo.
- Pressione uma tecla ou botão (ou uma combinação deles) *antes* ao mover um controle analógico para ativá-lo condicionalmente.
- Ao anexar a um ajuste, caso o último controle seja um controle com posição absoluta, mova o mesmo controle novamente para alternar entre a extensão total do eixo, a parte do eixo em ambos os lados da posição neutra e a faixa completa do eixo invertido.
- Ao anexar a um ajuste, caso o último controle atribuído seja um controle de posição relativa, mova o mesmo controle novamente para alternar a direção do controle entre ligado ou desligado.
- Ao anexar um ajuste, movimente um controle analógico desde que não seja o último controle atribuído ou pressione uma tecla ou botão para adicionar uma operação or.
- Ao pressionar Esc **antes** de ativar uma entrada do eixo, isso faz com que a configuração seja limpa, restaurando a sua atribuição original.
- Ao pressionar Esc **depois** de ativar uma entrada do eixo, isso mantém a sua atribuição original.
- A nova configuração é mostrada abaixo do menu. Espere um segundo depois de ativar uma entrada para que a nova configuração seja aceita.

Para realizar o ajuste da sensibilidade, da velocidade centralização automática, das configurações de inversão para entradas analógicas ou para ver como elas respondem às suas configurações, selecione a opção *Controles analógicos* no menu principal durante a emulação. A configuração das entradas estão agrupadas por dispositivo e ordenadas por tipo. É possível mover entre os dispositivos com o próximo grupo e as teclas/botões do grupo anterior usando as teclas [e]. O estado das entradas analógicas é mostrado abaixo do menu e elas reagem em tempo real. Pressione a tecla responsável pela *Visualização na tela* (a tecla ~ e ` num teclado US ANSI QWERTY e as teclas " e ' num teclado ABNT-2) para ocultar o menu principal, facilitando o teste sem alterar as configurações. Pressione novamente a mesma tecla ou botão para mostrar o menu completo novamente.

Cada entrada analógica possui quatro configurações no menu *Analog Controls*:

- Os controles de configuração *increment / decrement* (aumento / redução) controlam o quão rápido os valores da entrada aumenta ou reduz em resposta aos controles atribuídos nos ajustes de *increment / decrement*.

- A configuração *auto-centering speed* (velocidade autocentrante) controla o quão rápido o valor da entrada retorna ao estado neutro quando os controles atribuídos às configurações de *increment / decrement* são liberados.
- O ajuste *reverse* (inverso) permite inverter a direção da resposta recebida dos controles ao ser liberado. Ao definir como zero (0) faz com que o valor não retorne automaticamente para a posição neutro. Isso se aplica aos controles atribuídos ao ajuste do eixo e aos ajustes de *increment / decrement*.
- O ajuste *sensitivity* (sensibilidade) ajusta a resposta recebida do controle atribuído ao ajuste do eixo.

Use as teclas ou botões da esquerda e direita para ajustar a configuração em destaque. Ao selecionar uma configuração ou ao pressionar a tecla Del (Forward Delete) restabelece o seu valor inicial.

As unidades para as configurações da velocidade **increment/decrement**, **auto-centering speed** e **sensitivity** estão vinculadas à implementação do driver/dispositivo. As configurações da velocidade **increment/decrement**, **auto-centering speed** também são vinculadas à taxa dos quadros da tela principal do sistema. A resposta aos controles atribuídos às configurações de **increment/decrement** também será alterada caso o sistema altere a taxa de quadros desta tela.

4.2.4 O menu de seleção de programa e de sistema

Ao iniciar o MAME sem definir um sistema na linha de comando, será mostrado o menu para a seleção de sistemas (assumindo que a opção *-ui* esteja definido como cabinet). O menu para a seleção de sistemas também será exibido caso você escolha *Selecione um novo sistema* no menu principal durante a emulação. A seleção de um sistema que usa listas de programas mostra um menu de seleção semelhante.

O menu para a seleção de sistemas e de programas está dividido nestas partes:

- A área no topo mostra o nome e a versão do emulador, a quantidade dos sistemas ou dos itens do programa e o texto da pesquisa atual. O menu para a seleção do programa também mostra o nome do sistema selecionado.
- A barra de ferramentas abaixo da área do cabeçalho. Os botões mostrados na barra de ferramentas variam conforme o menu. Passe o ponteiro do mouse sobre um botão para ver uma descrição e clique para selecioná-lo.

Os botões da barra de ferramentas são *adiciona / remove* um sistema ou programa que estiver destacado nos favoritos (ícone de estrela), *Exporta a lista exibida para um arquivo* (ícone de disquete), *Audita a mídia* (ícone de lupa), *mostra as DATs* (ícone "i" num círculo azul), *Retornar ao menu anterior* e *Encerrar* (ícone de um "X" num quadrado vermelho).

- A lista dos sistemas ou dos programa ficam ao centro. O menu para a seleção dos sistemas, há opções para configuração abaixo da lista. Os clones são mostrados com um texto numa cor diferente (o padrão é cinza). É possível clicar com o botão direito do mouse no nome de um sistema ou programa que funciona como um atalho, mostrando as opções de configuração para o sistema.

Os sistemas ou os itens de programa são ordenados pelo seu nome completo ou descrição, mantendo os sistemas clonados logo abaixo dos sistemas principais. Pode parecer confuso num primeiro momento caso as suas configurações de filtro façam com que um sistema principal ou item de programa fique oculto enquanto um ou mais dos seus clones fiquem visíveis.

- O painel de informação na parte inferior, exibe informações resumidas sobre o sistema ou o programa em destaque. A cor de fundo muda dependendo da condição da emulação: verde para aqueles que funcionam, âmbar para emulação imperfeita ou caso tenha problemas conhecidos e vermelho no caso de problemas mais sérios como emulação incompleta.

Uma estrela amarela é exibida na parte superior esquerda do painel de informação caso o sistema ou programa destacado estiver na sua lista de favoritos.

- À esquerda há a lista dos filtros com as suas respectivas opções. Clique num filtro para aplicá-lo à lista de sistemas/programas. Alguns filtros mostram um menu com opções adicionais (informar o fabricante para o filtro do *Fabricante*, ou definir um arquivo e grupo para o filtro de *Categoria* por exemplo).

Clique em *Sem filtro* para exibir todos os sistemas disponíveis. Clique em *Filtro personalizado* para combinar diversos filtros. Clique no pilar entre a lista de filtros e a lista de sistemas/programa para mostrar ou ocultar a lista dos filtros. Esteja ciente que os filtros ainda permanecem aplicados mesmo que a lista dos filtros fique escondida.

- À direita há o visualizador de informações. Este possui duas abas para mostrar imagens e informações. Clique numa aba para alternar entre elas; clique nos triângulos à esquerda ou à direita ao lado do título da imagem/informação para alternar entre as imagens ou as fontes de informação.

As informações dos sistemas são mostradas automaticamente. As informações da lista de programas são mostradas para itens que forem relacionados com programas. Informações adicionais dos arquivos externos podem ser mostradas quando o *plug-in Data* estiver ativo.

É possível digitar algo na tela principal para iniciar pesquisa automática na lista exibida ao centro. Os sistemas são pesquisados pelo nome completo, fabricante e pelo nome abreviado. Caso esteja usando nomes de sistemas traduzidos, os nomes fonéticos também serão pesquisados caso estejam presentes. Os programas são pesquisados pela descrição, por títulos alternativos (elementos `alt_title` nas listas de programas) e pelo nome abreviado. A tecla Esc limpa a busca caso uma esteja sendo feita no momento.

Controles de navegação

Além dos *controles usuais de navegação*, o sistema e os menus para a seleção de programas têm controles configuráveis adicionais para navegar pelo layout dos vários painéis, fornecendo alternativas aos botões da barra de ferramentas caso não queira usar um dispositivo apontador. Como padrão, o MAME utiliza o US ANSI QWERTY como mapeamento do teclado, consulte o capítulo *Comparativo entre os mapas de teclado* para ver as diferenças entre o teclado Americano e o ABNT-2 usado no Brasil.

As configurações correspondentes com o mapeamento padrão do teclado são:

Tab (IU Próx. foco)

Foca a próxima região. A ordem é a lista de sistema/programa, configurações (caso esteja disponível), a lista dos filtros (caso esteja visível), abas de informação/imagem (caso estejam visíveis).

Shift + Tab (IU Foco ant.)

Move o foco para a região anterior.

Alt + D (IU Visualiza DAT externa)

Mostra o visualizador de informações em tela inteira.

Alt + F (IU Adiciona/remove favoritos)

Adiciona ou remove o sistema ou programa em destaque na lista de favoritos.

F1 (IU Afere mídia)

Realiza uma aferição das ROMs e das imagens dos discos dos sistemas. Os resultados são salvos e utilizados pelos filtros *Disponível* e *Indisponível*.

Quando o foco estiver na lista de filtros, é possível usar o controle de navegação do menu (cima, baixo, Home e End) para destacar um filtro e Enter / Return para selecioná-lo.

Quando o destaque estiver em qualquer região além das abas de *Informação / imagem*, é possível alterar a imagem ou a informação com as setas direcionais < / >. Quando o destaque estiver nas abas *imagens / Informações*, é possível rolar o texto com as informações a informação cima, baixo, PgUp, PgDn, Home e End.

É possível mover o foco para uma área clicando nela com o botão do meio do mouse.

4.2.5 O menu de seleção simplificada do sistema

Caso inicie o MAME sem especificar um sistema na linha de comando (ou caso escolha *Selecione um novo sistema* durante a emulação) com a opção *-ui* definida como *simple*, a seleção simplificada do sistema será mostrada. A seleção simplificada exibe 15 sistemas selecionados aleatoriamente desde que possuam as suas respectivas ROMs disponíveis dentro da pasta *ROM*. É possível digitar para realizar uma busca. Ao limpar a busca faz com que outros 15 sistemas aleatórios sejam exibidos novamente.

O painel de informação na parte debaixo mostra um resumo da informação do sistema que estiver destacado/selecionado. A cor de fundo muda conforme a condição da emulação daquele sistema: verde para aqueles que funcionam, âmbar para emulação imperfeita ou caso tenha problemas conhecidos e vermelho no caso de problemas mais sérios como emulação incompleta.

4.3 Teclas já predefinidas

- *Introdução aos controles*
- *Controles da interface*
 - *Menus para a seleção de programas e sistemas*
- *Controles predefinidos dos sistemas arcade*
- *Controles predefinidos para jogos arcade*
 - *Controles do jogador 1*
 - *Controles do jogador 2*
 - *Controles do jogador 3*
 - *Controles do jogador 4*
- *Teclas predefinidas para Mahjong e Hanafuda*
- *Teclas predefinidas para jogos de apostas*
 - *Teclas predefinidas para Blackjack*
 - *Teclas predefinidas para Poker*
 - *Teclas predefinidas para Slots*
- *Teclas predefinidas para computadores*
- *Outros sistemas*
- *Comparativo entre os mapas de teclado*
 - *QWERTY US (104 Teclas)*
 - *QWERTY ABNT-2 (107 Teclas)*

4.3.1 Introdução aos controles

O MAME é compatível com uma vasta gama de diferentes tipos de sistemas, e estes, possuem uma considerável quantidade de diferentes entradas entre elas. Isso significa que algumas teclas do teclado, alguns botões do mouse e alguns botões do joystick serão usados para diversas funções. Como resultado, o mapa dos controles mostrados abaixo são separados pelos tipos dos sistemas facilitando a busca.

Todos os controles abaixo são totalmente configuráveis através da interface do usuário. Os exemplos abaixo são as configurações predefinidas. As predefinições mostradas aqui são organizadas com base do no teclado **US ANSI QWERTY**. Caso tenha um layout de teclado diferente, as teclas poderão ser diferentes, consulte o capítulo [Comparativo entre os mapas de teclado](#) para ver as diferenças.

4.3.2 Controles da interface

Estes controles cobrem as teclas básicas de funcionamento do MAME como o cardápio de opções, a pausa do sistema, o salvamento e a leitura dos estados.

Tab

Chama o cardápio de opções.

~ - Tecla **til** no teclado *US ANSI QWERTY*.

` - Tecla **acento agudo** no teclado *US ANSI QWERTY*.

" - Tecla **aspas** no teclado *ABNT-2*.

' - Tecla **ápice** no teclado *ABNT-2*.

Mostra as opções na tela.

Caso esteja usando o MAME com a opção `-debug`, esta tecla funciona como um *'break'* na emulação.

É possível utilizar as teclas abaixo quando um controle deslizante estiver visível na tela:

- Cima - Seleciona o parâmetro anterior.
- Baixo - Seleciona o próximo parâmetro.
- Esquerda - Reduz o valor do parâmetro selecionado.
- Direita - Aumenta o valor do parâmetro selecionado.
- Enter - Redefine o valor do parâmetro para o seu valor predefinido.
- Control + Esquerda - Reduz o valor em passos de **10x**.
- Shift + Esquerda - Reduz o valor em passos de **1x**.
- Alt + Esquerda - Reduz todo o valor.
- Control + Direita - Aumenta o valor em passos de **10x**.
- Shift + Direita - Aumenta o valor em passos de **1x**.
- Alt + Direita - Aumenta o valor no máximo.
- End - Oculta temporariamente a opção visível na tela.
- Home - Traz de volta a opção na tela depois que for oculta.

Seta direcional cima

Destaca a opção anterior da lista da interface.

Seta direcional baixo

Destaca a opção posterior da lista da interface.

Seta direcional esquerda

Altera a opção com os valores disponível do lado esquerdo quando uma seta estiver presente.

Seta direcional direita

Altera a opção com os valores disponível do lado direito quando uma seta estiver presente.

Home / End

Destaca a primeira ou a última opção da lista.

[]

Move para o grupo anterior ou para o próximo grupo nos menus compatíveis da IU (por exemplo, move para as entradas do dispositivo anterior ou para o seguinte no menu *Atribuições da entrada (este sistema)*).

Enter / Botão 1 do controle 1

Seleciona a opção que estiver em destaque.

Espaço

Exibe um comentário na opção selecionada.

Delete

Limpa ou redefine o valor para a predefinição inicial nas configurações de entrada, trapaça e opções do plug-in.

F1

Liga o sistema nos sistema que tenham esta função de ligar e desligar.

F2

Desliga o sistema nos sistema que tenham esta função de ligar e desligar.

F3

Faz um reinício simples do sistema.

Shift esquerdo + F3

Realiza um reinício "pesado" onde tudo é zerado e carregado do início. Este é um processo mais completo do que quando feito com F3.

F4

Mostra a paleta do jogo, os gráficos decodificados, os gráficos em pedaços, os caracteres e qualquer outro *tilemap* disponível.

Use Enter para alternar entre os três modos (paleta, gráficos, e *tilemaps*).

Pressione F4 novamente para fechar essa tela. Os controles principais de cada modo variam um pouco:

Modo paleta/tabela de cores:

- [] - Alterna entre as paletas.
- Cima / Baixo - Sobe/desce uma linha por vez.
- PgUp / PgDn - Sobe/desce uma página por vez.
- Home / End - Vai para o topo/fim da lista.
- - / + - Aumenta/reduz a quantidade de cores por fila.
- 0 - Restaura a quantidade de cores por fila.
- Enter - Alterna para o visualizador gráfico.

Modo gráfico:

- [] - Alterna entre diferentes conjuntos gráficos.
- Cima / Baixo - Sobe/desce uma linha por vez.
- PgUp / PgDn - Sobe/desce uma página por vez.
- Home / End - Vai para o topo/fim da lista.
- Esquerda / Direita - Alterna a cor.
- R - Rotaciona os pedaços em 90° sentido horário.

- - / + - Aumenta/reduz a quantidade de blocos por fila (mantenha Shift pressionado para que valores integrais sejam usados ao aumentar/reduzir os blocos).
- 0 - Restaura a quantidade das blocos por fila (mantenha Shift pressionado para que valores integrais sejam usados ao aumentar/reduzir os blocos).
- Enter - Alterna para o visualizador *tilemap*.

Modo Tilemap:

- [] - Alterna entre diferentes *tilemaps*.
- Cima / Baixo / Esquerda / Direita - Desloca 8 pixels por vez.
- Shift + Cima / Baixo / Esquerda / Direita - Desloca 1 pixel por vez.
- Control + Cima / Baixo / Esquerda / Direita - Desloca 64 pixels por vez.
- R - Rotaciona os *tilemaps* em 90° sentido horário.
- - / + - Aumenta/reduz o fator de aproximação.
- 0 - Expande os pequenos *tilemaps* para preencher a tela.
- Enter - Alterna para o modo paleta/tabela de cores.

Observação: Nem todos os sistemas possuem gráficos decodificados e/ou *tilemaps*.

Shift esquerdo + F4

Enquanto estiver pausado, carrega o estado salvo mais recente para ser usado com o rebobinamento.

F5

Pausa a emulação.

Shift esquerdo + F5

Enquanto estiver pausado avança um quadro. Se o rebobinamento estiver ativo, um estado é salvo para ser usado pela função de rebobinamento.

F6

Salva um estado. É preciso pressionar uma tecla a mais para definir qual tecla será associada ao estado, semelhante à opção anterior. Caso já exista um estado salvo, ele também aparecerá na lista e também poderá ser substituído na lista.

Shift esquerdo + F6

Salva um estado de forma rápida.

F7

Carrega um estado. Você será solicitado a pressionar uma tecla no teclado para salvar a posição ou a escolher uma posição já salva no menu para ser carregada.

Observe que o recurso de salvar o estado ainda não é compatível com uma grande quantidade de drivers. Caso um determinado driver não funcione perfeitamente com este recurso, um aviso será mostrado alertando a incompatibilidade.

Shift esquerdo + F7

Faz o carregamento rápido de um estado já salvo.

F8

Reduz o salto de quadros em tempo real.

Shift esquerdo + F8

Liga ou desliga a trapaça. (caso a opção -cheat seja usada)

Alt esquerdo + F8

Reduz o prescaling. (*apenas nas versões SDL*)

F9

Aumenta o salto de quadros em tempo real.

Alt esquerdo + F9

Aumenta o prescaling. *(apenas nas versões SDL)*

F10

Liga ou desliga o supressor de velocidade.

Shift esquerdo + F10

Liga ou desliga o processamento HLSL. *(apenas na versão não SDL do Windows)*

Alt esquerdo + F10

Alterna o filtro. *(apenas nas versões SDL)*

F11

Liga ou desliga o indicador de velocidade no topo da tela.

Shift esquerdo + F11

Alterna a exibição do perfil interno (caso tenha sido compilado com ele).

Alt esquerdo + F11

Grava um vídeo com filtros HLSL.

F12

Faz uma captura da tela.

Shift esquerdo + F12

Inicia a gravação de um vídeo MNG.

Control esquerdo + Shift esquerdo + F12

Inicia a gravação de um vídeo AVI.

Alt esquerdo + F12

Faz uma captura da tela com um filtro HLSL.

Insert

Acelera a emulação. Enquanto a tecla estiver pressionada, a emulação é executada sem a supressão de velocidade e com o salto de quadros no máximo. *(apenas na versão não SDL do Windows)*

PgDn

Acelera a emulação. Enquanto a tecla estiver pressionada, a emulação é executada sem a supressão de velocidade e com o salto de quadros no máximo. *(apenas nas versões SDL)*

Alt esquerdo + Enter

Alterna entre tela inteira e modo janela.

ScrLk / FwdDel (Mac Desktop) / fn-Del (Mac Laptop)

Mapeamento predefinido para o **uimodekey**.

Essa tecla alterna entre a interface do MAME e a interface que está sendo emulada, quando a tecla é pressionada é possível usar a tecla Tab para alterar as opções do MAME e posteriormente retornar no sistema emulado. Consulte também a opção [-uimodekey](#)

Esc

Encerra a emulação, retorna para o menu anterior ou cancela uma opção na interface.

Menus para a seleção de programas e sistemas

Os menus de seleção do programa e do sistema usam controles adicionais.

Tab

Move o focus do teclado/controlador para o próximo painel da interface.

Shift + Tab

Move o focus do teclado/controlador para o painel anterior da interface.

Alt esquerda + F

Adiciona ou remove o sistema ou programa selecionado da lista de favoritos.

Alt esquerdo + E

Exporta a lista dos sistemas exibidos no momento.

Alt esquerdo + D

Mostra o visualizador de informações em tamanho real caso as informações estejam disponíveis para o sistema ou para o item selecionado da lista programas. (Mostra as informações carregadas pelo plug-in de dados dos arquivos externos, incluindo o **history.xml** e o **mameinfo.dat**.)

F1

Inicia a auditoria das ROMs e das imagens de disco.

4.3.3 Controles predefinidos dos sistemas arcade

Esta seção aborda os controles que são aplicáveis à maioria dos tipos dos sistemas de arcade. Observe que nem todas os sistemas terão todos estes controles disponíveis. Todos os controles abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

5 (fora do teclado numérico)

Ficha do slot 1

6 (fora do teclado numérico)

Ficha do slot 2

7 (fora do teclado numérico)

Ficha do slot 3

8 (fora do teclado numérico)

Ficha do slot 4

Backspace

Cédula 1 (Para sistemas que tenham um receptor/leitor de cédulas)

T

Tilt

Normalmente um interruptor de inclinação ou sensor de impacto que encerra o jogo atual, redefine os créditos e/ou reinicia o sistema caso ela seja derrubada com muita força ou seja deslocada. Mais comumente encontrado nos sistemas de pinball.

- (fora do teclado numérico)

Reduz o volume

Para sistemas que tenham um controle de volume.

= (fora do teclado numérico)

Aumenta o volume

Para sistemas que tenham um controle de volume.

F1

Redefine a memória

Isso redefine ou zera o(s) placar(es), os créditos/prêmios, as estatísticas ou as configurações do operador nos sistemas compatíveis.

F2

Modo de serviço

Este é um botão existente em alguns sistemas para acessar este modo, em outros sistemas pode ser uma chave ou uma chave DIP.

9 (fora do teclado numérico)

Serviço 1

Estes botões geralmente servem para dar créditos de graça, sem passar pela contabilidade do sistema, ou para navegar no menu de serviço do operador.

0 (fora do teclado numérico)

Serviço 2

- (fora do teclado numérico)

Serviço 3

= (fora do teclado numérico)

Serviço 4

4.3.4 Controles predefinidos para jogos arcade

Esta seção aborda os controles que são aplicáveis aos jogos arcades. Todos os controles abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

5 (fora do teclado numérico)

Ficha do slot 1

6 (fora do teclado numérico)

Ficha do slot 2

7 (fora do teclado numérico)

Ficha do slot 3

8 (fora do teclado numérico)

Ficha do slot 4

1 (fora do teclado numérico)

Inicia o jogador 1 ou o modo com 1 jogador

2 (fora do teclado numérico)

Inicia o jogador 2 ou o modo com 2 jogadores

3 (fora do teclado numérico)

Inicia o jogador 3 ou o modo com 3 jogadores

4 (fora do teclado numérico)

Inicia o jogador 4 ou o modo com 4 jogadores

Controles do jogador 1

Seta direcional cima

Jogador 1 cima

Seta direcional baixo

Jogador 1 baixo

Seta direcional esquerda

Jogador 1 esquerda

Seta direcional direita

Jogador 1 direita

E

Jogador 1 - Cima no controle esquerdo nos sistemas com dois controles (p.e. Robotron)

D

Jogador 1 - Baixo no controle esquerdo nos sistemas com dois controles (p.e. Robotron)

S

Jogador 1 - Esquerda no controle esquerdo nos sistemas com dois controles (p.e. Robotron)

F

Jogador 1 - Direita no controle esquerdo nos sistemas com dois controles (p.e. Robotron)

I

Jogador 1 - Cima no controle direito nos sistemas com dois controles (p.e. Robotron)

K

Jogador 1 - Baixo no controle direito nos sistemas com dois controles (p.e. Robotron)

J

Jogador 1 - Esquerdo no controle direito nos sistemas com dois controles (p.e. Robotron)

L

Jogador 1 - Direito no controle direito nos sistemas com dois controles (p.e. Robotron)

Ctrl esquerdo / Mouse B0 / Pistola 1 Botão 0

Jogador 1 botão 1

Alt esquerdo / Mouse B2 / Pistola 1 Botão 1

Jogador 1 botão 2

Espaço / Mouse B1 / Joystick 1 Botão 1 ou B.

Jogador 1 botão 3

Shift esquerdo

Jogador 1 botão 4

Z

Jogador 1 botão 5

X

Jogador 1 botão 6

C

Jogador 1 botão 7

V

Jogador 1 botão 8

B

Jogador 1 botão 9

N Jogador 1 botão 10

M Jogador 1 botão 11

, Jogador 1 botão 12

. Jogador 1 botão 13

/ Jogador 1 botão 14

Shift direito
Jogador 1 botão 15

Controles do jogador 2

R Jogador 2 cima

F Jogador 2 baixo

D Jogador 2 esquerda

G Jogador 2 direita

A Jogador 2 botão 1

S Jogador 2 botão 2

Q Jogador 2 botão 3

W Jogador 2 botão 4

E Jogador 2 botão 5

Controles do jogador 3

I Jogador 3 cima

K Jogador 3 baixo

J Jogador 3 esquerda

L Jogador 3 direita

Control direito

Jogador 3 botão 1

Shift direito

Jogador 3 botão 2

Enter (*fora do teclado numérico*)

Jogador 3 botão 3

Controles do jogador 4

8 (*no teclado numérico*)

Jogador 4 cima

2 (*no teclado numérico*)

Jogador 4 baixo

4 (*no teclado numérico*)

Jogador 4 esquerda

6 (*no teclado numérico*)

Jogador 4 direita

0 (*no teclado numérico*)

Jogador 4 botão 1

. (*no teclado numérico*)

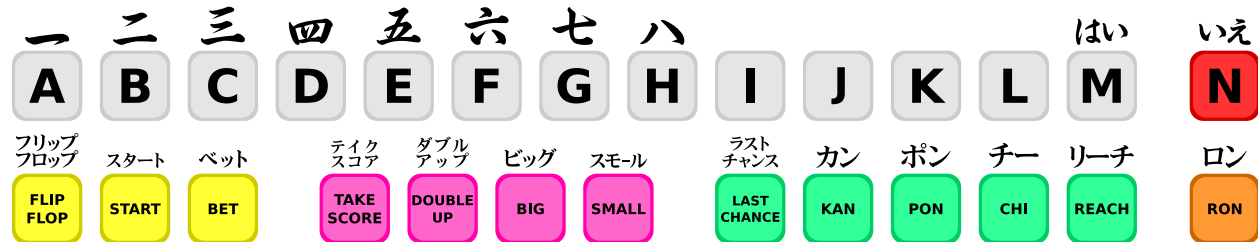
Jogador 4 botão 2

Enter (*no teclado numérico*)

Jogador 4 botão 3

4.3.5 Teclas predefinidas para Mahjong e Hanafuda

A maioria dos jogos de mahjong e hanafuda utilizam um layout padronizado de controle. Algumas teclas podem estar ausentes dependendo do tipo de jogo. Por exemplo, jogos sem o recurso de bônus podem não ter as teclas *Take Score*, *Double Up*, *Big* e *Small*. Jogos sem a característica de aposta também podem não ter a tecla *Bet*. Alguns jogos podem não usar todas as teclas presentes, alguns por exemplo não usam as teclas *Flip Flop* e *Last Chance*.



Por causa da grande quantidade de teclas, o MAME apenas oferece a configuração padrão para um único conjunto de controles por jogador. Para jogos multiplayer de mahjong/hanafuda ou jogos com múltiplas posições de jogador, é preciso fazer uma configuração manual das teclas. Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

5 (fora do teclado numérico)

Ficha do slot 1

6 (fora do teclado numérico)

Ficha do slot 2

7 (fora do teclado numérico)

Ficha do slot 3

8 (fora do teclado numérico)

Ficha do slot 4

Y

Jogador 1 Mahjong/Hanafuda Flip Flop

1 (fora do teclado numérico)

Inicia o jogador 1 ou o modo com 1 jogador

2 (fora do teclado numérico)

Jogador 2 start or 2 players mode

3 (fora do teclado numérico)

Jogador 3 start or 3 players mode

Mahjong Bet

4 (fora do teclado numérico)

Jogador 4 start or 4 players mode

Ctrl direito

Jogador 1 Mahjong/Hanafuda Take Score

Shift direito

Jogador 1 Mahjong/Hanafuda Double Up

Enter

Jogador 1 Mahjong/Hanafuda Big

Backspace

Jogador 1 Mahjong/Hanafuda Small

Alt direito

Jogador 1 Mahjong/Hanafuda Last Chance

Ctrl

Mahjong Kan

Alt

Mahjong Pon

Espaço

Mahjong Chi

Shift

Mahjong Reach

Z

Mahjong Ron

A

Jogador 1 Mahjong/Hanafuda A

B

Jogador 1 Mahjong/Hanafuda B

C

Jogador 1 Mahjong/Hanafuda C

D

Jogador 1 Mahjong/Hanafuda D

E

Jogador 1 Mahjong/Hanafuda E

F

Jogador 1 Mahjong/Hanafuda F

G

Jogador 1 Mahjong/Hanafuda G

H

Jogador 1 Mahjong/Hanafuda H

I

Jogador 1 Mahjong I

J

Jogador 1 Mahjong J

K

Jogador 1 Mahjong K

L

Jogador 1 Mahjong L

M

Jogador 1 Mahjong M

Jogador 1 Hanafuda Yes

N

Jogador 1 Mahjong N

Jogador 1 Hanafuda No

0 Jogador 1 Taiwanese Mahjong O
 ; Jogador 1 Taiwanese Mahjong P
 Q Jogador 1 Taiwanese Mahjong Q

4.3.6 Teclas predefinidas para jogos de apostas

Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

Observe que muitos jogos de apostas usam botões para diferentes funções. Por exemplo, em jogos do tipo slot o jogo pode usar o botão *Start* para parar todos os mostradores rotativos onde não há um botão dedicado para isso. Ou num jogo de poker os botões *hold* podem ser usados para controlar o *Double Up* do jogo e também não havendo uma tecla específica para os botões *Take Score*, *Double Up*, *High* e *Low*.

5 Ficha do slot 1
 6 Ficha do slot 2
 7 Ficha do slot 3
 8 Ficha do slot 4

Backspace

Cédula 1 (Para sistemas que tenham um receptor/leitor de cédulas)

I Payout
 Q Key In
 W Key Out

F1 Redefine a memória

9 (fora do teclado numérico)

Serviço 1

Estes botões geralmente servem para dar créditos de graça, sem passar pela contabilidade do sistema, ou para navegar no menu de serviço do operador.

0 (fora do teclado numérico)

Serviço 2 Book-Keeping (para sistemas com este recurso)

- (fora do teclado numérico)

Serviço 3

= (fora do teclado numérico)

Serviço 4

M

Bet

1 (*fora do teclado numérico*)

Inicia o jogador 1 ou o modo com 1 jogador

2 (*fora do teclado numérico*)

Deal

L

Stand

4 (*fora do teclado numérico*)

Take Score

Para jogos que permitem ganhar apostas num jogo de o dobro ou nada, este botão coleta as apostas do jogo.

3 (*fora do teclado numérico*)

Double Up

Para jogos que permitem ganhar apostas num jogo de o dobro ou nada, este botão aposta os prêmios do jogo.

D

Half Gamble

Usado em muitos jogos para apostar a metade dos prêmios.

A

High

S

Low

O

Door

Teclas predefinidas para Blackjack

Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

1

Inicia o jogador 1 ou o modo com 1 jogador

Usado para iniciar uma nova rodada em jogos que possuem botões separados para fazer uma nova rodada e obter uma carta a mais.

2

Deal (hit)

Usado para obter uma carta a mais e para iniciar uma nova rodada em jogos que não usam diferentes botões para fazer uma nova rodada e para obter uma carta a mais.

L

Stand

Teclas predefinidas para Poker

Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

- 1** Inicia o jogador 1 ou o modo com 1 jogador
Usado para iniciar uma nova rodada em jogos que possuem botões separados para fazer uma nova rodada e obter uma carta a mais.
- 2** Deal
Usado para substituir cartas e para iniciar uma nova rodada em jogos que não usam diferentes botões para fazer uma nova rodada e para substituir cartas.
- Z** Hold 1/discard 1 (Mantém/descarta 1)
- X** Hold 2/discard 2 (Mantém/descarta 2)
- C** Hold 3/discard 3 (Mantém/descarta 2)
- V** Hold 4/discard 4 (Mantém/descarta 4)
- B** Hold 5/discard 5 (Mantém/descarta 1)
- N** Cancela
Usado em alguns jogo para cancelar a seleção atual nas cartas que serão mantidas/descartadas.

Teclas predefinidas para Slots

Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

- 1** Inicia o jogador 1 ou o modo com 1 jogador
- X** Para o mostrador rotativo 1
- C** Para o mostrador rotativo 2
- V** Para o mostrador rotativo 3
- B** Para o mostrador rotativo 4
- Z** Para todos os mostradores rotativos

4.3.7 Teclas predefinidas para computadores

Todas as teclas abaixo são totalmente configuráveis na interface do usuário. Esta lista mostra a configuração predefinida do teclado.

Observe que os controles podem variar bastante de computador para computador, assim sendo, nem todas as teclas serão mostradas aqui. No MAME veja a opção *Atribuições da entrada (este sistema)* para obter mais detalhes do sistema que estiver usando.

Tab

Chama o menu de configuração.

ScrLk / FwdDel (Mac Desktop) / fn-Del (Mac Laptop)

Mapeamento predefinido para o **uimodekey**.

Essa tecla alterna entre a interface do MAME e a interface que está sendo emulada, quando a tecla é pressionada é possível usar a tecla Tab para alterar as opções do MAME e posteriormente retornar no sistema emulado. Consulte também a opção *-uimodekey*

F2

Inicia a fita dos sistemas que possuem leitores de fitas.

Shift + F2

Interrompe a leitura da fita.

Shift esquerdo + ScrLk

Cola a partir da área de transferência do sistema para o sistema emulado.

Teclas numéricas

Estas teclas são as mesmas usadas no sistema emulado.

4.3.8 Outros sistemas

Todas as teclas abaixo são totalmente configuráveis na interface do usuário.

Observe que os controles podem variar bastante de computador para computador, assim sendo, nem todas as teclas serão mostradas aqui. No MAME veja a opção *Atribuições da entrada (este sistema)* para obter mais detalhes do sistema que estiver usando.

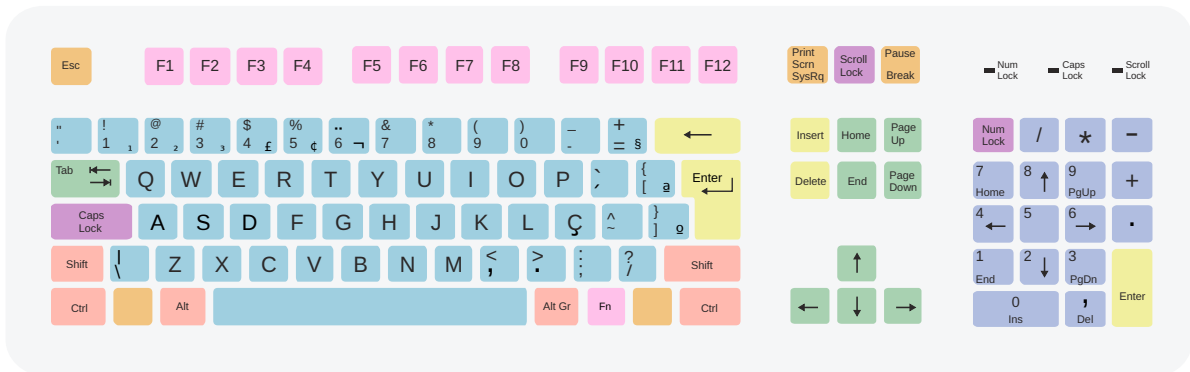
4.3.9 Comparativo entre os mapas de teclado

QWERTY US (104 Teclas)



- | | | | |
|---|--|--|---|
| Character keys | Enter and editing keys | Navigation keys | Numeric keypad |
| Modifier keys | System and GUI keys | Function keys | Lock keys |

QWERTY ABNT-2 (107 Teclas)



- | | | | |
|--|---|--|---|
| Teclas de função | Teclas modificadoras | Teclas do sistema e da interface | Teclas para comandos e edição |
| Teclas de bloqueio | Teclas numéricas | Teclas alfanuméricas | Teclas para navegação |

4.4 Cardápio de opções

- *Introdução*
- *Alterando os valores*
- *O menu principal*
- *Filtro*
- *Durante a emulação*
 - *Configurações da entrada*
 - *Atribuições da entrada (este sistema)*
 - *Ajustes da entrada analógica*
 - *Atribuições da entrada (geral)*
 - *Dispositivos de entrada*
 - *Chaves DIP*
 - *Contabilidade*
 - *Informação sobre o sistema*
 - *Alerta sobre o sistema*
 - *Informação sobre a imagem da mídia*
 - *Gerenciador de arquivos*
 - *Controle de fita*
 - *Pseudo Terminais*
 - *Seleção da BIOS*
 - *Dispositivos slot*
 - *Leitor de código de barras*
 - *Dispositivos de rede*
 - *Controles deslizantes*
 - *Opções do vídeo*
 - * *Tela #X*
 - * *Snapshot*
 - *Opções da mira*
 - *Trapaças*
 - *Opções dos plug-ins*
 - *Visualização da DAT externa*
 - *Adiciona/remove dos favoritos*
 - *Sobre o MAME*
 - *Selecione um novo sistema*

- Fecha o menu/Inicia o sistema
- Definições gerais
 - Personalize a Interface
 - Atribuição dos diretórios
 - Opções do vídeo
 - Opções do áudio
 - Opções diversas
 - Opções dos dispositivos na entrada
 - Atribuição das entradas (geral)
 - Controles do jogador 1 ~ 10
 - * Predefinições para o jogador 1
 - * Predefinições para o jogador 2
 - * Predefinições para o jogador 3
 - * Predefinições para o jogador 4
 - Opções avançadas
 - * Opções de desempenho
 - * Opções da Rotação da Tela
 - * Opções das ilustrações
 - * Opções do estado/reprodução
 - * Opções da entrada
- Plug-ins
- Dispositivos na entrada
 - Salva as definições
 - Retorna ao menu anterior
- Configurações do sistema

4.4.1 Introdução

Para exibir o *menu principal* durante a execução de um sistema emulado no MAME, pressione a tecla ou o botão para **Mostrar/Ocultar** o menu (a tecla predefinida é Tab). Se o sistema emulado tiver entradas de teclado, é necessário pressionar a tecla ou o botão **Toggle UI** para mostrar os controles (a tecla predefinida é "Scroll Lock", a tecla SRCLK ou "Forward Delete" no macOS) para que seja possível acessar os controles da interface de usuário do MAME. Você pode ignorar um menu pressionando a tecla ou o botão **UI Retorna** do usuário (a tecla predefinida é Esc). A desativação de um menu retornará ao seu menu principal ou ao sistema que estiver em execução, no caso do menu principal.

Pressione a tecla Tab para esconder o menu e retornar ao sistema principal, ao pressioná-la novamente, fará com que ela apareça novamente, isso é muito útil quando estiver testando ou configurando um sistema.

As entradas dos sistemas emulados são ignorados enquanto este menu estiver na tela. Contudo, ainda é possível pausar o sistema que estiver sendo executado ao pressionar a tecla para **Pausar** (a tecla predefinida é F5).

Ao iniciar o MAME sem especificar nenhum sistema na linha de comando, a interface com o menu de seleção do MAME será exibida (assumindo que a *opção da interface*) esteja definida como **cabinet**). Durante a execução de um sistema, este menu de seleção de sistema também aparece ao pressionar Tab e selecionar a opção *Selecione um novo sistema*.

Consulte a *seção relevante* para obter mais informações.

Caso rode o MAME sem nenhum parâmetro na linha de comando ou clicando duas vezes em seu ícone, você verá a interface do usuário e acessando as opções da parte debaixo da tela terá acesso ao cardápio de opções, entre eles a lista de seleção de jogos ao centro, os filtros do lado esquerdo, a aba de informações e imagens. No rodapé da tela tem um descritivo resumido do sistema, com o nome do sistema, nome do fabricante, ano, a condição som e imagem e se o sistema funciona ou não.

Dependendo da condição dos sistemas a interface do MAME exibirá diferentes cores de fundo indicando a sua condição.

- *Verde*: São sistemas com ROMs completos e que funcionam.
- *Vermelha*: São sistemas que não funcionam direito ou tem ROMs faltando.
- *Laranja*: São sistemas que funcionam mas estão imperfeitos na parte de som ou vídeo.

Abaixo da lista dos sistemas nós temos:

- *Definições gerais*: Exibe uma lista das configurações do MAME.
- *Configurações do sistema*: Exibe uma lista das opções de configuração do sistema selecionado.

Todos os itens exibidos nessa interface podem ser acessadas usando as setas do seu teclado (cima, baixo, esquerda, direita) e são selecionadas pressionando a tecla Enter do teclado. A interface também aceita o uso do mouse fazendo a seleção com um clique e um duplo clique para abrir a opção ou rodar um sistema.

4.4.2 Alterando os valores

A interface é bem intuitiva, os controles para modificar os valores predefinidos funcionam da seguinte maneira:

- **Mouse** - Move o cursor na tela, seleciona os itens, as teclas cima, baixo, esquerda e direita fazem o mesmo.
- **Clique duplo ou Enter** - Aguarda a entrada do usuário (controle, teclado, etc.).
- **Delete** - **1x** apaga o valor, **2x** retorna ao valor predefinido originalmente.

Os campos que possuam mais de uma opção de escolha podem ser abertos ao clicar duas vezes nele, como é o caso dos campos disponíveis em *Filtro*, por exemplo.

4.4.3 O menu principal

O menu principal é exibido quando se pressiona a tecla Tab ou o botão **Mostrar/Ocultar** o menu durante a execução de um sistema emulado ou enquanto a tela de informações do sistema é exibida. Ele fornece acesso a menus usados para alterar configurações, para controlar vários recursos e mostrar as informações sobre o sistema em execução e do próprio MAME.

Enquanto a tela de menu estiver sendo exibida o MAME não inicia a emulação do sistema até que este menu seja fechado (tecla Tab ou Esc). Isso pode ser útil para montar as imagens das mídias ou para alternar a posição das chaves DIP ou até mesmo para alterar as definições de algumas configurações da máquina antes de iniciar a emulação do sistema.

4.4.4 Filtro

Escolhe entre diferentes filtros pré configurados e um personalizado. Estes filtros ajudam o usuário a selecionar sistemas separados por categorias, caso queira encontrar um sistema que você não se lembra do nome porém se lembra do ano, é possível usar o filtro **Ano** para listar todos os sistemas conhecidos pelo MAME que foram lançados naquele ano.

Supondo que eu queira encontrar o sistema **Double Dragon**, faremos de conta que eu não me lembro, eu só lembro do ano 1987 e que o fabricante dela foi a *Technos Japan*. Vamos até o *Filtro Personalizado*, no primeiro filtro adicionamos um filtro para o Ano e colocamos 1987, adicionamos mais um filtro para o *Fabricante* e escolhemos *Technos Japan*, ao retornarmos ao menu anterior o MAME exibirá uma lista dos sistemas que atendam aos critérios definidos por nós. Neste exemplo então o MAME vai retornar 6 diferentes sistemas **Double Dragon**, **Super Dodge Ball** e **Nekketsu Koukou Dodgeball Bu**.

Os filtros disponíveis são:

- *Sem filtro*
Exibe toda a lista dos sistemas conhecidos e cadastrados no catálogo interno do MAME sem nenhum filtro.
- *Disponível*
Exibe a lista dos sistemas que o MAME identificou dentro do diretório roms.
- *Indisponível*
Exibe toda a lista dos sistemas conhecidos e cadastrados no catálogo interno do MAME que não estão disponíveis, ainda que a interface mostre a cor verde.

- *Funciona*

Exibe uma lista dos sistemas que funcionam e estão em condição verde e marrom, os sistemas na condição vermelha ou que ainda não funcionam ficam de fora da lista.

- *Não funciona*

Exibe apenas os sistemas que tenham condição vermelha e que não funcionam.

- *Mecânico*

Exibe toda a lista dos sistemas mecânicos conhecidos e cadastrados no catálogo interno do MAME como Pinball por exemplo.

- *Não mecânico*

Repete a lista [Não filtrado](#).

- *Categoria*

Este filtro usa de arquivos *.ini* para separar os sistemas em diversas categoria diferentes como por exemplo gabinetes com 2 jogadores, 4 jogadores, jogo de tiro, de corrida, de tabuleiro, corrida, etc. Em categorias onde a lista seja muito grande, clique duas vezes com o mouse em cima da lista para que uma nova tela seja exibida e fique mais fácil de escolher a opção desejada. Note que o uso destes arquivos pode fazer com que o MAME demore um pouco mais para iniciar.

O MAME não inclui nenhum arquivo de categoria, na internet é possível acessar o site [Progetto-Snaps](#) que oferece estes arquivos *.ini* para download [aqui](#). Depois que o arquivo for baixado e extraído o diretório **folders** deve ser copiado para o diretório raiz do MAME.

Até o presente momento não existe uma tradução dessas categorias para o Português Brasileiro. Abaixo estão as categorias existentes até o momento e que funcionam com o MAME, as categorias que não funcionam com o MAME foram criadas para serem usadas com o MAMEUI⁶ e não estão listadas aqui:

- *Cabinets*: Lista os sistemas **Arcade** do MAME que estão divididos em tipos de gabinetes.
- *Category*: Lista os sistema separados em categorias como corrida, tabuleiro, tiro, etc.
- *Driver*: Lista os sistemas por driver como *cps1.cpp*, *1943.cpp*, *3do.cpp*, etc.
- *FreePlay*: Lista os sistemas **Arcade** do MAME que possuem a opção de poder jogar de graça.
- *MonoChrome*: Lista os sistemas separado por cores.
- *Resolution*: Lista os sistemas separados pela sua resolução.

O site ainda oferece outros tipos de *.ini* como *version.ini* que separa os sistemas por versão em que eles apareceram pela primeira vez no MAME, note que estes arquivos extras não serão abordados neste documento porém já deve ter ficado fácil compreender a sua utilidade no MAME.

- *Favoritos*

Exibe uma lista dos sistemas que foram favoritados, para adicionar um sistema à lista de favoritos, pressione Tab, no menu que aparecer selecione *Adiciona aos favoritos*.

- *BIOS*

Exibe uma lista dos sistemas que precisam de uma BIOS para funcionar.

- *Sem BIOS*

Exibe uma lista dos sistemas que não precisam de uma BIOS para funcionar.

⁶ O [MAMEUI](#) é uma versão do MAME com uma interface gráfica diferente.

- *Principais*
Quando existirem sistemas derivados do sistema principal exibe uma lista dos sistemas que são originados desta matriz.
- *Clones*
Exibe uma lista dos sistemas que são considerados clones dos sistemas originais.
- *Fabricante*
Exibe uma lista com todos os fabricantes catalogados pelo MAME.
- *Ano*
Exibe uma lista dos sistemas separados por ano de lançamento.
- *Código-fonte*
Permite listar todos os sistemas existentes num arquivo de código-fonte (como capcom/cps2.cpp por exemplo).
- *É possível salvar*
Exibe uma lista dos sistemas onde existe a possibilidade de salvar o estado do sistema.
- *Não é possível salvar*
Exibe uma lista dos sistemas onde não é possível salvar o estado do sistema.
- *Precisa de CHD*
Exibe uma lista dos sistemas que precisam de uma imagem de disco para funcionar.
- *Não precisa de CHD*
Exibe uma lista dos sistemas que não precisam de uma imagem de disco para funcionar.
- *Tela vertical*
Exibe uma lista dos sistemas que usam orientação vertical da tela.
- *Tela horizontal*
Exibe uma lista dos sistemas que usam orientação horizontal da tela.
- *Filtro personalizado*
Todo o filtro criado será listado aqui.

4.4.5 Durante a emulação

Estas opções podem ser acessadas durante a emulação e estão acessíveis ao pressionar a tecla Tab e entrando na opção *Configurações da entrada*.

Configurações da entrada

Aqui ficam listadas todas as listas para as configurações das entradas do sistema que estiver sendo emulado no momento.

Atribuições da entrada (este sistema)

Aqui ficam as configurações do sistema que estiver em execução e por isso esta lista varia entre os sistemas, as configurações vão desde o botão para os créditos, os botões do controle dos jogadores, a definição para o botão do acesso ao modo de serviço do sistema (caso seja um arcade), teclas individuais de um teclado de um computador pessoal, etc.

Ajustes da entrada analógica

Aqui ficam todas as configurações para os ajustes dos controles analógicos, este menu só aparece quando o sistema possui controle analógico. As opções disponíveis com seus valores predefinidos também variam, abaixo temos um exemplo para o jogo **Virtua Racing**:

- *Jogador 1 pedal 1 aumenta/reduz a velocidade* 16
- *Jogador 1 pedal 1 velocidade da centralização automática* 16
- *Jogador 1 pedal 1 inverso* Desligado
- *Jogador 1 pedal 1 sensibilidade* 100
- *Jogador 1 pedal 2 aumenta/reduz a velocidade* 16
- *Jogador 1 pedal 2 aumenta/reduz a velocidade* 16
- *Jogador 1 pedal 2 inverso* Desligado
- *Jogador 1 pedal 2 sensibilidade* 100
- *Acionador aumenta/reduz a velocidade* 3
- *Acionador velocidade da centralização automática* 3
- *Acionador inverso* Desligado
- *Acionador sensibilidade* 100

Consulte o capítulo *Configurações da entrada analógica* para obter mais informações sobre os ajustes uma vez que a nomenclatura também muda conforme o sistema.

Atribuições da entrada (geral)

- *Interface do usuário*
Consulte *Atribuição das entradas (geral)*.
- *Controles do jogador [1~10]*
Consulte *Predefinições para o jogador 1*.
- *Outros controles*
Consulte *Outros controles*.

Dispositivos de entrada

Aqui serão listados todos os dispositivos que estão conectados ao sistema atual, como o teclado, o mouse, etc. A lista é dinâmica e varia dependendo do sistema e do que está sendo conectado nele. Ao fazer um clique duplo em um dos itens da lista um novo menu é mostrado e aqui é possível selecionar a opção *Copia a ID do dispositivo*, consulte o capítulo *ID fixo para os controles* para obter mais informações.

Chaves DIP

Aqui ficam as chaves DIP, elas servem para definir as configurações do sistema (quando for relevante), a quantidade de fichas necessárias para registrar 1 crédito, se a tela será invertida ou não, se a demonstração do sistema ficará em silêncio ou se reproduzirá qualquer tipo de som, etc.

Sempre que uma chave for alterada, sempre selecione **Reinicie** para que a alteração seja aplicada. Em algum sistema a ação já pode ser vista na tela, contudo, não é sempre o caso.

Contabilidade

É o registro interno do sistema que mostra o tempo total que ele ficou em execução e a quantidade de fichas que foram colocadas nela.

Informação sobre o sistema

Mostra um breve resumo do nome do sistema que está sendo emulado, o seu driver, o tipo do processador (vídeo, áudio e outros) e a resolução do vídeo.

Alerta sobre o sistema

Exibe informações de alerta sobre o sistema que está sendo emulado como o não funcionamento de áudio, vídeo, etc.

Informação sobre a imagem da mídia

Exibe informações sobre a mídia como o seu formato (imagem ROM, CD-ROM, etc.), nome do fabricante, ano de lançamento, etc. Este item só é mostrado se o sistema em execução tiver um ou mais dispositivos do tipo mídia (unidades de

Gerenciador de arquivos

Mostra o menu do gerenciador de arquivos onde é possível abrir arquivos ROM, montar ou desmontar imagens de mídia. Este item só é mostrado se o sistema em execução tiver um ou mais dispositivos de mídia (unidades de disquete, cartuchos ou slots de cartão de memória por exemplo).

Controle de fita

Mostra o menu de controle de fita, com ele é possível emular e controlar os mecanismos de uma fita k7. Este item só é mostrado se o sistema em execução usar uma fita k7.

Pseudo Terminais

Mostra a condição geral de qualquer dispositivo tipo "pseudoterminal" no sistema em execução (usado para conectar o sistema emulado aos pseudoterminais do host, como através de portas seriais por exemplo). Este item só é mostrado se o sistema em execução tiver dispositivos do tipo pseudoterminal.

Seleção da BIOS

Mostra o menu de seleção de BIOS, aqui é possível selecionar a BIOS para o *boot* ou a ROM do *firmware* para o sistema e as placas que o sistema tiver. Este item só é mostrado se o sistema em execução tiver disponibilidade de seleção ou opções de BIOS.

Dispositivos slot

Mostra o menu dos dispositivos slot, aqui é possível escolher entre os periféricos emulados. Este item só é mostrado se o sistema em execução tiver dispositivos no slot.

Leitor de código de barras

Mostra o menu do leitor de códigos de barras, aqui é possível simular a leitura de códigos de barras usando a emulação do leitor de código de barras. Este item só é mostrado se o sistema em execução tiver um leitor de código de barras.

Dispositivos de rede

Mostra o menu dos dispositivos de rede, aqui é possível configurar os adaptadores de rede emulados compatíveis com uma ponte numa rede para um host. Este item só é mostrado se o sistema em execução tiver adaptadores de rede que sejam compatíveis com *bridging*.

Controles deslizantes

Mostra o menu de controles deslizantes, aqui é possível ajustar diversas configurações incluindo ajustes de vídeo, níveis individuais dos canais de áudio dentre outros. As opções disponíveis aqui também dependem do tipo do sistema, outros ajustes podem aparecer porém os principais são estes:

- *Volume principal*

Faz o ajuste do volume do áudio principal do sistema que estiver sendo emulado.

- *Volume com xxx Ch.x*

Faz o ajuste individual de cada canal de áudio, sistema com áudio mono só tem o Ch. 0, já sistemas com canal estéreo possuem Ch. 0 (esquerdo) e Ch. 1 (direito) e assim por diante. A quantidade de canais disponíveis vai depender do sistema que está sendo emulado.

- *Brilho da tela*

Faz o controle do nível de preto da tela, consulte também *-brightness*.

O valor predefinido é 1.0.

- *Contraste da tela*

Faz o controle do nível de branco da tela, consulte também *-contrast*.

O valor predefinido é 1.0.

- *Gama da tela*

Faz o ajuste da escala de luminância da tela, consulte também *-gamma*.

O valor predefinido é 1.0.

- *Extensão horizontal da tela*

Estica a tela no eixo horizontal.

O valor predefinido é 1.0.

- *Posição horizontal da tela*

Desloca a tela no eixo horizontal

O valor predefinido é 0.0.

- *Extensão vertical da tela*

Estica a tela no eixo vertical.

O valor predefinido é 1.0.

- *Posição vertical da tela*

Desloca a tela no eixo vertical

O valor predefinido é 0.0.

Opções do vídeo

Mostra o menu das opções de vídeo, aqui é possível ajustar as configurações de vídeo, alterar as opções de visualização, selecionar as artes gráficas (quando estiver disponível) e as capturas de tela.

Tela #X

Caso o sistema possua mais de uma tela, todas elas serão listadas aqui, onde "X" indica o número da tela e cada uma delas com as opções mostradas abaixo. Aqui também vai aparecer qualquer tipo de ilustração do sistema emulado e as suas respectivas opções, quando houver.

- *Nome*

Caso esteja usando uma **artwork** e ela tiver um nome, ela será exibida aqui indicando que ela pode ser selecionada.

- *Tela 0 padrão (4:3)*

Faz com que a tela tenha uma proporção padrão de 4:3.

- *Tela 0 relação de aspecto (X:Y)*

Faz com que a tela use a proporção original (SAR) como 8:7, 12:7, etc.

- *Cocktail*

Faz com que a tela fique espelhada no eixo vertical da tela.

- *Rotação*

Rotaciona a tela, as opções disponíveis são:

- *CW 90°*: Rotaciona a tela no sentido horário em 90°.
- *180°*: Rotaciona a tela em 180°.
- *CCW 90°*: Rotaciona a tela no sentido anti-horário em 90°.

O valor predefinido é None.

- *Aproxime para a região da tela*

Quando o sistema estiver usando uma ilustração (artwork) onde exista o recorte de uma tela, somente esta região será aproximada.

O valor predefinido é Desligado.

- *Escale a tela com valores não inteiros*

Faz com que a tela possa ser expandida usando números fracionados em vez de números inteiros, isso causa efeitos "*aliasing*" (um efeito colateral de deformação dos pixels) indesejáveis na tela, para mais informações consulte [-unevenstretch](#).

As opções disponíveis são:

- *Apenas X*: Expande a tela apenas no eixo X.
- *Apenas Y*: Expande a tela apenas no eixo Y.
- *X ou Y (Auto)*: Expande a tela em ambos os eixos automaticamente.

O valor predefinido é Ligado.

- *Mantenha a relação de aspecto*

Mantém a proporção 4:3 da tela, independente do que as outras configurações façam, consulte também *-keepaspect*.

O valor predefinido é Ligado.

Snapshot

Faz uma captura da tela, caso esteja usando uma **ilustração** e ela tiver um nome, faça um clique duplo em cima do nome para que a captura da tela seja feito.

- *Tela O padrão (4:3)*

Faz um print da tela com esta proporção

- *Tela O relação de aspecto (X:Y)*

Faz uma captura da tela usando a proporção original (SAR) como 8:7, 12:7, etc.

- *Cocktail*

Faz uma captura da tela espelhada no eixo vertical.

- *Rotação*

Faz uma captura da tela com a tela rotacionada como demonstrado no exemplo anterior.

O valor predefinido é None.

- *Aproxime a área da tela*

Quando o sistema estiver usando uma ilustração onde exista um recorte de uma tela, a captura da tela será feita somente desta região.

O valor predefinido é Desligado.

Opções da mira

Mostra o menu das opções de mira, aqui é possível ajustar a aparência das miras usadas, para mostrar o local das miras e outras entradas absolutas de ponteira. Este item só é mostrado se o sistema em execução tiver entradas do tipo que usem algum tipo de apontamento. As seguintes opções estão disponíveis:

- *Visibilidade do P1*

Aqui é definido quando a mira deve estar visível na tela para o jogador 1 (player 1) com as seguintes opções:

- *Nunca*: Nunca mostra a mira.
- *Sempre*: Sempre mostra a mira.
- *Quando for movido*: Mostra apenas quando o mouse for movido.

O padrão é mostrar a mira sempre que ela for movida, no caso de mais jogadores, haverá uma opção para cada um deles (jogador 1, jogador 2, etc.).

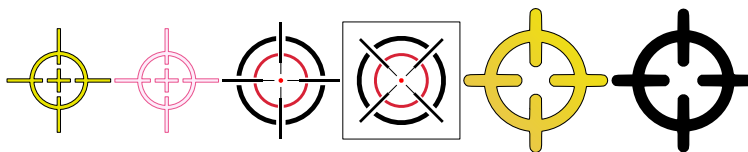
- *Mira P1*

Aqui é possível definir o tipo da mira que será usada pelo sistema, note que o MAME só aceita arquivos **.png** com transparência (sem fundo):

- *[Incorporado]*: Usa a mira padrão do MAME.

- *nome do arquivo mira*: Você tem a opção de selecionar os seus arquivos de mira pelo nome.

Note que é possível usar qualquer nome para o seu arquivo da mira para depois poderem ser selecionados aqui. Abaixo temos alguns exemplos:



Quem tiver interesse, é possível baixar estas miras já prontas para uso [neste endereço](#). Descompacte as imagens dentro da pasta **crosshair**.

Trapaças

Mostra o menu das trapaças, aqui é possível controlar o sistema interno de trapaças. Este item só é mostrado se o sistema em execução tiver trapaças ativadas. Note que que existe a opção de trapaça via plug-in que está acessível através do menu *Opções dos Plug-ins*.

Opções dos plug-ins

Mostra o menu das opções do plug-in, aqui é possível acessar as configurações para todos os plug-ins que estiverem ativos. Este item só é mostrado se o sistema em execução tiver plug-ins ativados ou caso o menu principal seja mostrado antes de iniciar a emulação de um sistema (ao pressionar a tecla Tab enquanto estiver aparecendo a tela de informação do sistema).

Quando os plug-ins forem ativados na configuração, eles serão listados aqui. Qualquer alteração que for feita ao ativar um plugin, ele é gravado no arquivo `plugin.ini` que fica na mesma pasta do MAME ou em `~/ .mame` em sistemas Linux e macOS.

Para mais informações consulte [Plug-ins](#).

Visualização da DAT externa

Mostra o menu de visualização da DAT externa, aqui é possível acessar as informações carregadas de diversos arquivos externos de suporte. Este item só é mostrado se o sistema em execução não tiver o [plugin de dados](#) ativado ou se o menu principal for mostrado antes de iniciar a emulação do sistema (ao pressionar a tecla Tab enquanto estiver aparecendo a tela de informação do sistema).

Esta opção estará disponível quando dois critérios forem atendidos, o plug-in **Data plugin** estiver ativo e os arquivos ***.dat** (`command.dat`, `gameinit.dat`, etc.) estiverem dentro do diretório **datas**.

Caso o nome do sistema exista dentro do `command.dat` por exemplo, será exibido uma lista de como jogar, dicas, a lista dos comandos do sistema na tela (em Inglês), etc.

Adiciona/remove dos favoritos

Adiciona o sistema em execução na lista de favoritos ou o remove caso ele já exista na lista. A lista de favoritos pode ser usada como um filtro no menu de seleção de sistemas.

Sobre o MAME

Mostra a versão do MAME, a sua revisão e as informações da licença de direitos autorais.

Selecione um novo sistema

Mostra o menu de seleção de sistemas, aqui é possível escolher um novo sistema e iniciar uma nova emulação. Este item só é mostrado se o sistema em execução mostrar o menu principal antes de iniciar a emulação do sistema (ao pressionar a tecla Tab enquanto estiver aparecendo a tela de informação do sistema).

Fecha o menu/Inicia o sistema

Fecha o menu principal retornando para o sistema em execução. Mostra *Inicia o sistema* antes de iniciar a emulação de um sistema (ao pressionar a tecla Tab enquanto estiver aparecendo a tela de informação do sistema).

4.4.6 Definições gerais

Personalize a Interface

Aqui é possível personalizar a interface do MAME, os valores numéricos podem ser alterados movendo o direcional para a esquerda e direita ou pressionando a tecla **Enter** e digitando o valor manualmente.

As opções disponíveis são:

- *Fontes*

Permite a customização da tipografia da interface, dentro desta opção temos:

- *Tipografia da interface*

Aqui é possível definir uma fonte para toda a interface do MAME.

O valor predefinido é Padrão.

- *Linhas*

Ajusta a dimensão do espaço e o tamanho da fonte, quanto maior o valor maior a dimensão da interface e menor o texto na tela.

O valor predefinido é 30.

- *Tamanho do texto das informações*

Ajusta o tamanho do texto nas caixas de texto na tela.

O valor predefinido é 0.75.

- *Cores*

Permite a customização completa das cores da interface do MAME, as opções disponíveis são:

- *Texto Normal*

Define a cor do texto de toda a interface.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 255.

- *Cor Seleccionada*

Define a cor do item que for seleccionado.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 0.

- *Fundo do texto normal*

Aparentemente não tem função alguma.

O valor predefinido é Opacidade: 239, Vermelho: 0, Verde: 0, Azul: 0.

- *Cor de fundo seleccionada*

Define a cor do item seleccionado.

O valor predefinido é Opacidade: 239, Vermelho: 128, Verde: 128, Azul: 0.

- *Cor de subitem*

Define a cor dos itens que estiverem abaixo do item principal.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 255.

- *Clone*

Define a cor do texto de segundo plano.

O valor predefinido é Opacidade: 255, Vermelho: 128, Verde: 128, Azul: 128.

– *Borda*

Define a cor das linhas da borda da tela.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 255.

– *Fundo*

Define a cor do fundo da tela e dos sistemas clonados.

O valor predefinido é Opacidade: 239, Vermelho: 16, Verde: 16, Azul: 48.

– *Chave DIP*

Define a cor das chaves DIP selecionadas nos sistemas que usam tal chaves.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 0.

– *Cor indisponível*

Aparentemente não tem função alguma.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 0.

– *Cor do controle deslizante*

Define a cor dos controles deslizantes.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 0.

– *Fundo do visualizador GFX*

Define a cor de fundo do visualizador GFX (tecla F4).

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 0.

– *Cor da sobreposição do mouse*

Define a cor que texto terá quando o mouse passar por cima de algum item selecionável.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 128.

– *Cor de fundo da sobreposição do mouse*

Define a cor de fundo do texto quando o mouse passar por cima de um item selecionável.

O valor predefinido é Opacidade: 112, Vermelho: 64, Verde: 64, Azul: 0.

– *Cor de subposição do mouse*

Aparentemente não tem função alguma.

O valor predefinido é Opacidade: 255, Vermelho: 255, Verde: 255, Azul: 128.

– *Cor debaixo do mouse*

Aparentemente não tem função alguma.

O valor predefinido é Opacidade: 176, Vermelho: 96, Verde: 96, Azul: 0.

• *Idioma*

Permite a seleção do Idioma da interface do MAME, faça um clique duplo no campo do idioma para abrir uma listagem com todos os idiomas disponíveis.

O valor predefinido é **English**

- *Os nomes dos sistemas*

No momento só existe a opção **incorporado**.

O valor predefinido é **incorporado**.

- *Mostra os painéis laterais*

Configura a exibição ou não dos painéis laterais da interface do MAME. As opções disponíveis são:

- *Mostra tudo*
- *Esconde os filtros*
- *Esconde info/imagem*
- *Esconde ambos*

Atribuição dos diretórios

Aqui é possível mudar as predefinições do locais onde os diretórios usados pelo MAME se encontram. As opções disponíveis são:

- *ROMs*

Define o caminho do diretório onde se encontram as ROMs. Veja também *-rompath*.

O valor predefinido é um diretório chamado **roms** no diretório raiz do MAME.

- *Software em mídia*

Define o caminho onde a imagem em mídia dos arquivos são armazenados como CD-ROM, floppy, fita K7 ou qualquer outro programa avulso.

O valor predefinido é um diretório chamado **software** no diretório raiz do MAME.

- *Amostras de áudio* Define o caminho do diretório onde se encontram os arquivos das amostras de áudio.

O valor predefinido é um diretório chamado **samples** no diretório raiz do MAME.

- *Ilustrações*

Define o caminho do diretório onde se encontram as ilustrações gráficas que fazem o preenchimento de fundo da tela dos sistemas. Veja mais em *-artpath*.

O valor predefinido é um diretório chamado **artwork** no diretório raiz do MAME.

- *Miras*

Define o caminho do diretório onde se encontram as imagens com uma arte gráfica de uma mira que podem ser usadas por jogos de tiro. Para mais informações consulte *opções da mira*.

O valor predefinido é um diretório chamado **crosshair** no diretório raiz do MAME.

- *Arquivos de trapaça*

Define o caminho do diretório onde se encontra o arquivo de trapaça. Este arquivo também pode ser deixado na pasta raiz do MAME.

O valor predefinido é um diretório chamado **cheats** no diretório raiz do MAME.¹

- *Plug-ins*

Define o caminho do diretório onde se encontram os plug-ins do MAME.

O valor predefinido é um diretório chamado **plugins** no diretório raiz do MAME.

¹ O site [Pugsy's Cheat](#) é um dos mais conhecidos que oferece um arquivo de trapaça para download.

- *Traduções da interface*

Define o caminho do diretório onde se encontram os arquivos de tradução da interface do MAME.

O valor predefinido é um diretório chamado **language** no diretório raiz do MAME.

- *INIs*

Define o caminho do diretório onde se encontram os arquivos *.ini*.

O valor predefinido é um diretório chamado **ini** no diretório raiz do MAME.

- *Configurações da interface*

Define o caminho do diretório onde se encontram os arquivos *.ini* da configuração da interface.

O valor predefinido é um diretório chamado **ui** no diretório raiz do MAME.

- *Dados do plug-in*

Define o caminho do diretório onde os dados do plug-in serão gravados.

O valor predefinido é o mesmo diretório raiz do MAME.

- *DATs*

Define o caminho do diretório onde se encontram os arquivos *.dat*.

O valor predefinido são os diretórios **dats**, **history** e no diretório raiz do MAME.

- *INIs com as categorias*

Define o caminho do diretório onde se encontram os arquivos *.ini* com descritivos de categoria.

O valor predefinido é um diretório chamado **folders** no diretório raiz do MAME.

- *Capturas da tela*

Define o caminho do diretório onde serão armazenadas as capturas da tela e a gravação de vídeo.

O valor predefinido é um diretório chamado **snapshots** no diretório raiz do MAME.

- *Ícones*

Define o caminho do diretório onde se encontram os arquivos *.ico* para serem usados como ícones que ficam ao lado do nome do sistema.²

- *Painéis de controle*

Define o caminho do diretório onde se encontram as imagens ou as fotos com a arte gráfica do painel onde se encontram os diferentes controles e botões do arcade.

O valor predefinido é um diretório chamado **cpanel** no diretório raiz do MAME.

- *Gabinetes*

Define o caminho do diretório onde se encontram as imagens dos gabinetes.

O valor predefinido são dois diretórios chamados **cabinets** e **cabdevs** no diretório raiz do MAME.

- *Marquises*

Define o caminho do diretório onde se encontram as imagens com a arte gráfica que ficavam na parte de cima do sistema.

O valor predefinido é um diretório chamado **marquees** no diretório raiz do MAME.

² O site do [MAMEICONS](#) e [Progetto Snaps](#) oferecem tais ícones e outras imagens para download.

O valor predefinido é um diretório chamado **icons** no diretório raiz do MAME.

- *PCIs*

Define o caminho do diretório onde se encontram fotos que mostram a placa de circuito impresso do sistema.

O valor predefinido é um diretório chamado **pcb** no diretório raiz do MAME.

- *Panfletos*

Define o caminho do diretório onde se encontram as imagens dos panfletos.

O valor predefinido é um diretório chamado **flyers** no diretório raiz do MAME.

- *Títulos das telas*

Define o caminho do diretório onde se encontram as imagens que mostram a tela de título do sistema.

O valor predefinido é um diretório chamado **titles** no diretório raiz do MAME.³

- *Finais dos jogos*

Define o caminho do diretório onde se encontram as imagens com a captura de tela mostrando o final do jogo.

O valor predefinido é um diretório chamado **ends** no diretório raiz do MAME.

- *Chefes*

Define o caminho do diretório onde se encontram as imagens com a captura de tela dos chefes de fase.⁴

O valor predefinido é um diretório chamado **bosses** no diretório raiz do MAME.

- *Amostras das ilustrações*

Define o caminho do diretório onde se encontram as imagens com as amostras das ilustrações, essas amostras tem um tamanho menor se comparadas com as ilustrações completas.

O valor predefinido é um diretório chamado **artwork preview** e **artpreiew** no diretório raiz do MAME.

- *Seleciona*

A ser concluído.

O valor predefinido é um diretório chamado **select** no diretório raiz do MAME.

- *Telas do game over*

Define o caminho do diretório onde se encontram as imagens que mostram a tela de **GAME OVER**.

O valor predefinido é um diretório chamado **gameover** no diretório raiz do MAME.

- *Como fazer*

Define o caminho do diretório onde se encontram as imagens ou fotos daqueles panfletos que mostravam as instruções de como jogar.

O valor predefinido é um diretório chamado **howto** no diretório raiz do MAME.

- *Logotipos*

Define o caminho do diretório onde se encontram as imagens ou ilustrações com a logomarca das empresas.

O valor predefinido é um diretório chamado **logos** no diretório raiz do MAME.

³ O site [MAME Channel](#) oferece diferentes telas de títulos para download.

⁴ É possível baixar essas imagens do site [EmuMovies](#).

- *Placares*

Define o caminho do diretório onde se encontram as imagens com a captura de tela mostrando as maiores pontuações.⁵

- *Versus*

Define o caminho do diretório onde se encontram as imagens com a captura de tela mostrando as maiores pontuações.

O valor predefinido é um diretório chamado **versus** no diretório raiz do MAME.

- *Capas*

Define o caminho do diretório onde se encontram as imagens com as capas dos jogos.

O valor predefinido é um diretório chamado **covers** no diretório raiz do MAME.

⁵ É possível baixar essas imagens do site [High-Score](#) e [Cubeman](#).
O valor predefinido é um diretório chamado **scores** no diretório raiz do MAME.

Opções do vídeo

Essas opções sempre serão carregadas na inicialização do MAME, lembrando que a linha de comando **SEMPRE** tem prioridade, independente do que seja definido aqui.

- *Modo do vídeo*

Para mais informações consulte [-video](#).

O valor predefinido é Auto.

- *– Quantidade de telas*

Predefine a quantidade das telas que serão usadas na emulação.

O valor predefinido é 1.

- *GLSL*

Ativa ou não os efeitos GLSL, para mais informações consulte [-gl_gsl_filter](#).

O valor predefinido é Desligado.

- *Filtragem bilinear*

Ativa ou não os filtros de tela para suavizar os gráficos, caso os gráficos fiquem muito borrados, experimente ativar também a opção *escala prévia do bitmap*.

O valor predefinido é Ligado.

- *Escala prévia do bitmap*

Opção útil quando os sistemas com baixa resolução são ampliadas para uma resolução maior, use essa opção para dar uma amenizada nessa aparência, essa opção geralmente é utilizada em conjunto com a opção **Filtragem bilinear**.

O valor predefinido é 1.

- *Modo janela*

Faz o MAME exibir a tela emulada numa janela ou numa tela inteira.

O valor predefinido é Desligado.

- *Mantenha a relação de aspecto*

Faz com que a proporção da tela seja sempre mantida.

O valor predefinido é Ligado.

- *Inicia com a tela expandida*

Faz o MAME exibir a tela emulada numa janela com o tamanho máximo do seu monitor, caso contrário exibe a tela emulada em sua resolução nativa.

O valor predefinido é Ligado.

- *Atualização síncrona dos quadros*

Consulte [-syncrefresh](#).

- *Aguarde a sincronização vertical*

Consulte [-waitvsync](#).

Opções do áudio

- *Áudio*
Ativa o áudio ou não, para mais informações consulte *-sound*.
O valor predefinido é `Ligado`.
- *Compressor*
Tenta manter o nível mais baixo e o mais alto do áudio no mesmo nível, atua também na redução do volume do volume do áudio caso seja muito alto.
O valor predefinido é `Ligado`.
- *Taxa da amostragem*
Define a taxa da amostragem do áudio que será usada em todas os sistemas.
O valor predefinido é `48000`.
- *Use amostras externas*
Consulte *-samples*.

Opções diversas

- *Ignore os avisos de emulação imperfeita*
Faz com que o MAME não exiba as telas de aviso dos sistemas com emulação imperfeita (tarja amarela).
O valor predefinido é `Desligado`.
- *Selecione novamente o último sistema já executado*
Faz com que o MAME se lembre do último sistema que foi jogado através da interface do MAME.
O valor predefinido é `Ligado`.
- *Aumenta as imagens no painel direito*
Aumenta o tamanho de qualquer uma das imagens exibidas no painel direito da interface do MAME, sempre mantendo a proporcionalidade da imagem.
O valor predefinido é `Ligado`.
- *Trapaças*
Ativa ou não o sistema de trapaças do MAME.
O valor predefinido é `Desligado`.
- *Mostra o ponteiro do mouse*
Ativa ou não a exibição do mouse na interface do MAME.
O valor predefinido é `Ligado`.
- *Confirma se deseja encerrar o sistema ou não*
Faz com que o MAME sempre te pergunte se quer realmente encerrar a emulação do sistema ou não.
O valor predefinido é `Desligado`.

- *Omite a tela de informações ao iniciar*

Não exibe a tela com informações sobre o sistema quando iniciar um sistema.

O valor predefinido é Desligado.

- *Mantenha o aspecto 4:3 nas capturas da tela*

Impõem uma proporção de 4:3 em todas as capturas da tela.

O valor predefinido é Ligado.

- *Usa uma imagem como plano de fundo*

Permite o uso de uma imagem como papel de parede na interface do MAME. Escolha uma imagem **.JPG** ou **.PNG** e a renomeie para **background.jpg** ou **background.png**. Para fazer uso dela coloque-a no diretório raiz do MAME (no mesmo diretório onde o executável do MAME se encontra).

O valor predefinido é Ligado.

- *Omite a tela da escolha da BIOS*

Faz com que o MAME inicie o sistema com a primeira BIOS disponível para o sistema ao em vez de usar uma lista.

O valor predefinido é Desligado.

- *Omite as partes do cardápio da seleção do programa*

Altera a maneira com que a lista do software é exibida, em vez de exibir a lista na ordem predefinida pelo MAME, exibe a lista na ordem listada no arquivo da respectiva lista.

O valor predefinido é Desligado.

- *Informação de aferição automática*

Exibe na aba de informações gerais do lado direito da interface do MAME informação quanto a condição da ROM selecionada se é **BOA** ou **RUIM**. Assim como também verifica se o sistema usa amostras ou não, aferindo se a condição delas seja **BOA** ou **RUIM**. Caso o sistema não use amostras, aparecerá a mensagem **Nenhuma Necessária**. Note que essa função deixa a interface do MAME um pouco mais lenta devido as aferições que são feitas em tempo real a cada seleção da ROM.

O valor predefinido é Desligado.

- *Esconde os sistemas sem ROMs da lista de disponíveis*

Esconde da lista de sistemas eletrônicos que não usam ROMs.

O valor predefinido é Ligado.

Opções dos dispositivos na entrada

- *Atribuição do dispositivo pistola de luz*

Caso exista um controlador para a pistola de luz, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo trackball*

Caso exista um controlador para o trackball, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo pedal*

Caso exista um controlador para pedais, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo do controle AD*

Caso exista um controlador para adstick, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo de acionamento*

Caso exista um controlador para acionamento, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do controle rotativo*

Caso exista um controle com um disco rotativo, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo posicional*

Caso exista um controlador de posição, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é keyboard.

- *Atribuição do dispositivo mouse*

Caso exista um controlador para mouse, os valores disponíveis são None, keyboard, mouse, lightgun e joystick.

O valor predefinido é mouse.

- *Provedor de entrada do teclado*

Determina o provedor responsável pela entrada do teclado, os valores disponíveis variam dependendo do sistema operacional auto, win32, dinput, rawinput, sdl e none.

O valor predefinido é auto.

- *Provedor de entrada do mouse*

Determina o provedor responsável pela entrada do teclado, os valores disponíveis variam dependendo do sistema operacional auto, win32, dinput, rawinput, sdl e none.

O valor predefinido é auto.

- *Provedor da pistola de luz*

Determina o provedor responsável pela entrada do teclado, os valores disponíveis variam dependendo do sistema operacional auto, win32, rawinput, X11 e none.

O valor predefinido é auto.

- *Provedor de entrada do joystick*

Determina o provedor responsável pela entrada do teclado, os valores disponíveis variam dependendo do sistema operacional auto, winhybrid, xinput, dinput, sdl e none.

O valor predefinido é auto.

Atribuição das entradas (geral)

- *interface do usuário*

Aqui estão os principais atalhos já predefinidos da interface do MAME, todos eles podem ser alterados conforme a necessidade. Para retornar ao valor original pressione Del duas vezes em cima da opção.

- *Visualização na tela*

Exibe um visor na parte inferior da tela durante a emulação para a realização de ajustes em tempo real.

A tecla predefinida é ~.

- *Entra no depurador*

Atalho para entrar no depurador durante a emulação, só funciona caso o MAME tenha sido compilado com ferramentas de depuração.

A tecla predefinida é ~.

- *Guia de configuração*

Chama o cardápio de opções do MAME.

A tecla predefinida é Tab.

- *Pausa*

Pausa a emulação.

A tecla predefinida é P.

- *Pausa - passo único*

Avança em passos de um quadro.

As teclas predefinidas são P + Shift Esq.

- *Rebobina - passo único*

Retrocede em passos de um quadro.

As teclas predefinidas são Til + Shift Esq.

- *Redefine o sistema*

Encerra a emulação e a reinicia do zero.

As teclas predefinidas são F3 + Shift Esq.

- *Redefinição rápida*

Reinicia sem encerrar a emulação.

A tecla predefinida é F3.

- *Mostra os gráficos decodificados*

Mostra a paleta GFX decodificada e os *tilemaps* dos jogos.

A tecla predefinida é F4.

- *Pula quadro dec*

Reduz o salto dos quadros de vídeo, os valores se alteram entre auto e entre 10~0. A predefinição é *auto*, ao pressionar a tecla a opção sai de auto para a velocidade mais rápida e vai diminuindo passo a passo até voltar para auto

A tecla predefinida é F8.

- *Pula quadro inc*

Aumenta o salto dos quadros de vídeo, os valores se alteram entre auto e entre 0~10. A predefinição é *auto*, ao pressionar a tecla, a opção sai de auto e aumenta a velocidade passo a passo até atingir auto.

A tecla predefinida é F9.

- *Supressor de velocidade*

Acelera a velocidade da emulação da nativa para o máximo possível.

A tecla predefinida é F10.

- *Avanço rápido*

Como o exemplo anterior porém faz a emulação rodar o mais rápido possível enquanto a tecla estiver pressionada.

- *Mostra FPS*

Exibe quantos quadros por segundo a emulação está rodando.

A tecla predefinida é PgDn em versões SDL do MAME e Insert no Windows.

- *Salva uma captura da tela*

Captura a tela emulada e a salva no diretório predefinido.

A tecla predefinida é F12.

- *Grava MNG*

Grava um vídeo em formato MNG sem áudio.

As teclas predefinidas são F12 + Shift Esq.

- *Grava AVI*

Grava um vídeo em formato AVI.

A teclas predefinidas são F12 + Shift Esq.

- *Liga/Desliga trapaça*

Ativa ou desativa a trapaça no jogo, só funciona quando usado a opção cheat na linha de comando.

A tecla predefinida é F6.

- *UI cima*

Move o cursor para cima.

A tecla predefinida é cima ou **Cima** no controle.

- *UI baixo*

Move o cursor para baixo.

A tecla predefinida é baixo ou **Baixo** no controle.

- *UI esquerda*
Move o cursor para a esquerda.
A tecla predefinida é esquerda ou **Esquerda** no controle.
- *UI direita*
Move o cursor para a direita.
A tecla predefinida é direita ou **Direita** no controle.
- *UI Home*
Move o cursor para o topo da lista.
A tecla predefinida é home.
- *UI Fim*
Move o cursor para o fim da lista.
A tecla predefinida é end.
- *UI Pág. cima*
Move o cursor para o topo da lista saltando 26 linhas por vez.
A tecla predefinida é PgUp.
- *UI Pág. baixo*
Move o cursor para o fim da lista saltando 26 linhas por vez.
A tecla predefinida é PgDn.
- *UI Próx. foco*
Faz com que foco do cursor passe para a próxima janela da interface.
A tecla predefinida é Tab.
- *UI Foco ant.*
Faz com que foco do cursor passe para a anterior anterior da interface.
A tecla predefinida é Tab + Shift Esq.
- *UI Selecciona*
Tecla de seleção para qualquer item selecionável.
As teclas predefinidas são Enter, **Botão 0 do controle** ou Enter do teclado numérico.
- *UI Cancela*
Tecla para cancelar qualquer ação.
A tecla predefinida é Esc.
- *UI Mostra comentário*
Tecla para exibir um comentário.
A tecla predefinida é barra de espaço.
- *UI Limpa*
Tecla para apagar/zerar uma opção.
A tecla predefinida é Del.

- *UI Aproxima*

Tecla para aproximar (dar zoom) na interface. Ainda não funciona em teclados ABNT, apenas em teclados tipo ANSI.

A tecla predefinida é =.

- *UI Recua* Tecla para sair do zoom da interface. Ainda não funciona em teclados ABNT, apenas em teclados tipo ANSI.

A tecla predefinida é -.

- *UI Aproximação predefinida* Retorna para a aproximação normal da tela.

A tecla predefinida é Ø.

- *UI Grupo anterior*

Faz a lista pular para o grupo anterior. Ainda não funciona em teclados ABNT, apenas em teclados tipo ANSI.

A tecla predefinida é [.

- *UI Próximo grupo*

Faz a lista pular para o próximo grupo. Ainda não funciona em teclados ABNT, apenas em teclados tipo ANSI.

A tecla predefinida é].

- *UI Rotaciona*

Gira a interface.

A tecla predefinida é R (não funciona).

- *Mostra o perfil*

Exibe o analisador de desempenho.

As teclas predefinidas são F11 + Shift Esq.

- *UI Alterna*

Alterna a interface do usuário.

A tecla predefinida é SrcLk.

- *UI Cola texto*

Cola texto na interface do usuário.

As teclas predefinidas são SrcLk + Shift Esq.

- *Salva o estado*

Salva o estado do sistema.

As teclas predefinidas são F7 + Shift Esq.

- *Carrega o estado*
Carrega o estado do sistema.
A tecla predefinida é F7.
- *(Primeira) UI inicia fita*
Inicia a fita na interface primária.
A tecla predefinida é F2.
- *(Primeira) UI para fita*
Para a fita na interface primária.
As teclas predefinidas são F2 + Shift Esq.
- *UI Visualiza DAT externa*
Exibe o DAT externo, para que a visualização do DAT seja possível é preciso ativar o plugin *Data Plugin* na interface ou editando o arquivo `plugin.ini`, o valor da linha **data** de 0 para 1.
As teclas predefinidas são Alt Esq + D.
- *UI Adiciona/remove favoritos*
Adiciona ou remove os sistemas da lista de favoritos.
As teclas predefinidas são Alt Esq + F.
- *UI exporta lista*
Exporta a lista dos sistemas no formato:
 - XML igual ao comando **-listxml**.
 - XML igual ao comando **-listxml** excluindo os dispositivos.
 - TXT igual ao comando **-listfull**.As teclas predefinidas são Alt Esq + E.
- *UI Afere mídia*
Realiza uma aferição das ROMs removendo as não disponíveis, o resultado é salvo no arquivo `mame_avail.ini` dentro do diretório **ui**.
A tecla predefinida é F1.
- *Alterna para tela inteira*
Alterna entre tela inteira e janela.
As teclas predefinidas são Enter + Alt Esq.
- *Liga/desliga o filtro*
Alterna entre usar ou não o filtro na tela.
As teclas predefinidas são F5 + Ctrl Esq.
- *Reduz a escala prévia*
Reduz a escala prévia dos pixels.
As teclas predefinidas são F6 + Ctrl Esq.

- *Aumenta a escala prévia*

Aumenta a escala prévia de dos pixels.

As teclas predefinidas são F7 + Ctrl Esq.

- *Grava um vídeo com efeitos*

Grava o vídeo usando todos os efeitos e filtros ativos na tela.

As teclas predefinidas são F12 + Ctrl + Alt Esq.

Controles do jogador 1 ~ 10

Definições para todos os botões e controles usados pelo sistema separado por jogador, entre o jogador 1 até o 10. Abaixo a lista das opções predefinidas para o jogador 1 que podem ser alteradas na própria interface do MAME.

Predefinições para o jogador 1

Controles do jogador 1	Predefinição
J1 cima	cima ou joy 1 cima
J1 baixo	baixo ou joy 1 baixo
J1 esquerda	esquerda ou joy 1 esquerda
J1 direita	direita ou joy 1 direita
J1 lado direito/cima	I ou joy 1 botão 1
J1 lado direito/baixo	K ou joy 1 botão 2
J1 lado direito/esquerdo	J ou joy 1 botão 0
J1 lado direito/direito	L ou joy 1 botão 3
J1 lado esquerdo/cima	E ou joy 1 cima
J1 lado esquerdo/baixo	D ou joy 1 baixo
J1 lado esquerdo/esquerdo	S ou joy 1 esquerda
J1 lado esquerdo/direito	F ou joy 1 direita
J1 botão 1	joy 1 botão 3
J1 botão 2	joy 1 botão 6
J1 botão 3	joy 1 botão 0
J1 botão 4	joy 1 botão 7
J1 botão 5	joy 1 botão 2
J1 botão 6	joy 1 botão 1
J1 botão 7	C ou joy 1 botão 6
J1 botão 8	V ou joy 1 botão 7
J1 botão 9	B ou joy 1 botão 8
J1 botão 10	N ou joy 1 botão 9
J1 botão 11	M ou joy 1 botão 10
J1 botão 12	vírgula ou joy 1 botão 11
J1 botão 13	Stop
J1 botão 14	Barra inc. direita
J1 botão 15	Shift Direito
J1 botão 16	n/a
J1 start	1
J1 select	5
J1 Mahjong A	A

continua na próxima página

Tabela 1 – continuação da página anterior

Controles do jogador 1	Predefinição
J1 Mahjong B	B
J1 Mahjong C	C
J1 Mahjong D	D
J1 Mahjong E	E
J1 Mahjong F	F
J1 Mahjong G	G
J1 Mahjong H	H
J1 Mahjong I	I
J1 Mahjong J	J
J1 Mahjong K	K
J1 Mahjong L	L
J1 Mahjong M	M
J1 Mahjong O	O
J1 Mahjong P	Dois pontos
J1 Mahjong Q	Q
J1 Mahjong Kan	Control esquerdo
J1 Mahjong Pon	Alt esquerdo
J1 Mahjong Chi	Espaço
J1 Mahjong Reach	Shift
J1 Mahjong Ron	Z
J1 Mahjong Bet	3
J1 Mahjong Last Chance	Alt direito
J1 Mahjong Score	Control direito
J1 Mahjong Double Up	Shift direito
J1 Mahjong Flip Flop	Y
J1 Mahjong Big	Return
J1 Mahjong Small	Backspace
J1 Hanafuda A/1	A
J1 Hanafuda B/2	B
J1 Hanafuda C/3	C
J1 Hanafuda D/4	D
J1 Hanafuda E/5	E
J1 Hanafuda F/6	F
J1 Hanafuda G/7	G
J1 Hanafuda H/8	H
J1 Hanafuda Sim	M
J1 Hanafuda Não	N
Chave dentro (in)	Q
Chave fora (out)	W
Serviço	9
Contabilidade	0
Porta	O
Prêmio	I
Aposta	M
Negocia	2
Mantém	L
Leva a pontuação	4
Dobra	3
Metade da aposta	D

continua na próxima página

Tabela 1 – continuação da página anterior

Controles do jogador 1	Predefinição
Alto	A
Baixo	S
Mantém 1	Z
Mantém 2	X
Mantém 3	C
Mantém 4	V
Mantém 5	B
Cancela	N
Interrompe o carretel 1	X
Interrompe o carretel 2	C
Interrompe o carretel 3	V
Interrompe o carretel 4	B
Interrompe todos os carretéis	Z
J1 pedal 1 analógico	...
J1 pedal 1 analógico inc	Control esq. ou joy 1 botão 0
J1 pedal 1 analógico dec	Nenhum
J1 pedal 2 analog	n/a
J1 pedal 2 analógico inc	Alt esq. ou joy 1 botão 1
J1 pedal 2 analógico dec	Nenhum
J1 pedal 3 analog	Nenhum
J1 pedal 3 analógico inc	Espaço ou joy 1 botão 2
J1 pedal 3 analógico dec	Nenhum
Acionador analógico	...
Acionador analógico inc	Direita
Acionador analógico dec	Esquerda
Acionador V analog	...
Acionador V analógico inc	Baixo
Acionador V analógico dec	Cima
Posicionamento analógico	...
Posicionamento analógico inc	Direita
Posicionamento analógico dec	Esquerda
Posicionamento V analog	...
Posicionamento V analógico inc	Baixo
Posicionamento V analógico dec	Cima
Rotativo analógico	...
Rotativo analógico inc	Baixo
Rotativo analógico dec	Cima
Rotativo V analógico	...
Rotativo V analógico inc	Baixo
Rotativo V analógico dec	Cima
Trilha X analógico	...
Trilha X analógico inc	Direita
Trilha X analógico dec	Esquerda
Trilha Y analógico	...
Trilha Y analógico inc	Baixo
Trilha Y analógico dec	Cima
Controle AD X analógico	...
Controle AD X analógico inc	Direita
Controle AD X analógico dec	Esquerda

continua na próxima página

Tabela 1 – continuação da página anterior

Controles do jogador 1	Predefinição
Controle AD Y analog	...
Controle AD Y analógico inc	Baixo
Controle AD Y analógico dec	Cima
AD stick Z analog	...
AD stick Z analógico inc	Z
AD stick Z analógico dec	A
Pistola de luz X analógico	...
Pistola de luz X analógico inc	Direita
Pistola de luz X analógico dec	Esquerda
Pistola de luz Y analógico	...
Pistola de luz Y analógico inc	Baixo
Pistola de luz Y analógico dec	Cima
Mouse X analógico	...
Mouse X analógico inc	Direita
Mouse X analógico dec	Esquerda
Mouse Y analógico	...
Mouse Y analógico inc	Baixo
Mouse Y analógico dec	Cima

Predefinições para o jogador 2

Controles do jogador 2	Predefinição
J2 cima	R
J2 baixo	F
J2 esquerda	D
J2 direita	G
J2 lado direito/cima	Nenhum
J2 lado direito/baixo	Nenhum
J2 lado direito/esquerdo	Nenhum
J2 lado direito/direito	Nenhum
J2 lado esquerdo/cima	Nenhum
J2 lado esquerdo/baixo	Nenhum
J2 lado esquerdo/esquerdo	Nenhum
J2 lado esquerdo/direito	Nenhum
J2 botão 1	A
J2 botão 2	S
J2 botão 3	Q
J2 botão 4	W
J2 botão 5	Nenhum
J2 botão 6	Nenhum
J2 botão 7	Nenhum
J2 botão 8	Nenhum
J2 botão 9	Nenhum
J2 botão 10	Nenhum
J2 botão 11	Nenhum
J2 botão 12	Nenhum
J2 botão 13	Nenhum

continua na próxima página

Tabela 2 – continuação da página anterior

Controles do jogador 2	Predefinição
J2 botão 14	Nenhum
J2 botão 15	Nenhum
J2 botão 16	Nenhum
J2 start	2
J2 select	6
J2 Mahjong A	Nenhum
J2 Mahjong B	Nenhum
J2 Mahjong C	Nenhum
J2 Mahjong D	Nenhum
J2 Mahjong E	Nenhum
J2 Mahjong F	Nenhum
J2 Mahjong G	Nenhum
J2 Mahjong H	Nenhum
J2 Mahjong I	Nenhum
J2 Mahjong J	Nenhum
J2 Mahjong K	Nenhum
J2 Mahjong L	Nenhum
J2 Mahjong M	Nenhum
J2 Mahjong O	Nenhum
J2 Mahjong P	Nenhum
J2 Mahjong Q	Nenhum
J2 Mahjong Kan	Nenhum
J2 Mahjong Pon	Nenhum
J2 Mahjong Chi	Nenhum
J2 Mahjong Reach	Nenhum
J2 Mahjong Ron	Nenhum
J2 Mahjong Bet	Nenhum
J2 Mahjong Last Chance	Nenhum
J2 Mahjong Score	Nenhum
J2 Mahjong Double Up	Nenhum
J2 Mahjong Flip Flop	Nenhum
J2 Mahjong Big	Nenhum
J2 Mahjong Small	Nenhum
J2 Hanafuda A/1	Nenhum
J2 Hanafuda B/2	Nenhum
J2 Hanafuda C/3	Nenhum
J2 Hanafuda D/4	Nenhum
J2 Hanafuda E/5	Nenhum
J2 Hanafuda F/6	Nenhum
J2 Hanafuda G/7	Nenhum
J2 Hanafuda H/8	Nenhum
J2 Hanafuda Sim	Nenhum
J2 Hanafuda Não	Nenhum
J2 pedal 1 analógico	n/a
J2 pedal 1 analógico inc	A
J2 pedal 1 analógico dec	Nenhum
J2 pedal 2 analógico	n/a
J2 pedal 2 analógico inc	S
J2 pedal 2 analógico dec	Nenhum

continua na próxima página

Tabela 2 – continuação da página anterior

Controles do jogador 2	Predefinição
J2 pedal 3 analógico	Nenhum
J2 pedal 3 analógico inc	Q
J2 pedal 3 analógico dec	Nenhum
Acionador 2 analógico	n/a
Acionador 2 analógico inc	G
Acionador 2 analógico dec	D
Acionador V 2 analógico	n/a
Acionador V 2 analógico inc	F
Acionador V 2 analógico dec	R
Posicionamento 2 analógico	n/a
Posicionamento 2 analógico inc	G
Posicionamento 2 analógico dec	D
Posicionamento V 2 analógico	n/a
Posicionamento V 2 analógico inc	F
Posicionamento V 2 analógico dec	R
Rotativo 2 analógico	n/a
Rotativo 2 analógico inc	G
Rotativo 2 analógico dec	D
Rotativo V 2 analógico	n/a
Rotativo V 2 analógico inc	F
Rotativo V 2 analógico dec	R
Trilha X 2 analógico	n/a
Trilha X 2 analógico inc	G
Trilha X 2 analógico dec	D
Trilha Y 2 analógico	n/a
Trilha Y 2 analógico inc	F
Trilha Y 2 analógico dec	R
Controle AD X 2 analógico	n/a
Controle AD X 2 analógico inc	G
Controle AD X 2 analógico dec	D
Controle AD Y 2 analógico	n/a
Controle AD Y 2 analógico inc	F
Controle AD Y 2 analógico dec	R
Pistola de luz X 2 analógico	n/a
Pistola de luz X 2 analógico inc	G
Pistola de luz X 2 analógico dec	D
Pistola de luz Y analógico	n/a
Pistola de luz Y analógico inc	F
Pistola de luz Y analógico dec	R
Mouse X 2 analógico	n/a
Mouse X 2 analógico inc	G
Mouse X 2 analógico dec	D
Mouse Y 2 analógico	n/a
Mouse Y 2 analógico inc	F
Mouse Y 2 analógico dec	R

Predefinições para o jogador 3

Controles do jogador 3	Predefinição
J3 cima	I
J3 baixo	K
J3 esquerda	J
J3 direita	L
J3 lado direito/cima	Nenhum
J3 lado direito/baixo	Nenhum
J3 lado direito/esquerdo	Nenhum
J3 lado direito/direito	Nenhum
J3 lado esquerdo/cima	Nenhum
J3 lado esquerdo/baixo	Nenhum
J3 lado esquerdo/esquerdo	Nenhum
J3 lado esquerdo/direito	Nenhum
J3 botão 1	Control direito
J3 botão 2	Shift direito
J3 botão 3	Return
J3 botão 4	W
J3 botão 5	n/a
J3 botão 6	n/a
J3 botão 7	n/a
J3 botão 8	n/a
J3 botão 9	n/a
J3 botão 10	n/a
J3 botão 11	n/a
J3 botão 12	n/a
J3 botão 13	n/a
J3 botão 14	n/a
J3 botão 15	n/a
J3 botão 16	n/a
J3 start	3
J3 select	7
J3 pedal 1 analógico	n/a
J3 pedal 1 analógico inc	Control direito
J3 pedal 1 analógico dec	Nenhum
J3 pedal 2 analógico	n/a
J3 pedal 2 analógico inc	Shift direito
J3 pedal 2 analógico dec	Nenhum
J3 pedal 3 analógico	Nenhum
J3 pedal 3 analógico inc	Return
J3 pedal 3 analógico dec	Nenhum
Acionador 3 analógico	n/a
Acionador 3 analógico inc	L
Acionador 3 analógico dec	J
Acionador V 3 analógico	n/a
Acionador V 3 analógico inc	K
Acionador V 3 analógico dec	I
Posicionamento 3 analógico	n/a
Posicionamento 3 analógico inc	L

continua na próxima página

Tabela 3 – continuação da página anterior

Controles do jogador 3	Predefinição
Posicionamento 3 analógico dec	J
Posicionamento V 3 analógico	n/a
Posicionamento V 3 analógico inc	K
Posicionamento V 3 analógico dec	I
Rotativo 3 analógico	n/a
Rotativo 3 analógico inc	L
Rotativo 3 analógico dec	J
Rotativo V 3 analógico	n/a
Rotativo V 3 analógico inc	K
Rotativo V 3 analógico dec	I
Trilha X 3 analógico	n/a
Trilha X 3 analógico inc	L
Trilha X 3 analógico dec	J
Trilha Y 3 analógico	n/a
Trilha Y 3 analógico inc	K
Trilha Y 3 analógico dec	I
Controle AD X 3 analógico	n/a
Controle AD X 3 analógico inc	L
Controle AD X 3 analógico dec	J
Controle AD Y 3 analógico	n/a
Controle AD Y 3 analógico inc	K
Controle AD Y 3 analógico dec	I
Pistola de luz X 3 analógico	n/a
Pistola de luz X 3 analógico inc	L
Pistola de luz X 3 analógico dec	J
Pistola de luz Y analógico	n/a
Pistola de luz Y analógico inc	K
Pistola de luz Y analógico dec	I
Mouse X 3 analógico	n/a
Mouse X 3 analógico inc	L
Mouse X 3 analógico dec	J
Mouse Y 3 analógico	n/a
Mouse Y 3 analógico inc	K
Mouse Y 3 analógico dec	I

Predefinições para o jogador 4

Controles do jogador 4	Predefinição
J4 cima	8 teclado numérico
J4 baixo	2 teclado numérico
J4 esquerda	4 teclado numérico
J4 direita	6 teclado numérico
J4 lado direito/cima	Nenhum
J4 lado direito/baixo	Nenhum
J4 lado direito/esquerdo	Nenhum
J4 lado direito/direito	Nenhum
J4 lado esquerdo/cima	Nenhum

continua na próxima página

Tabela 4 – continuação da página anterior

Controles do jogador 4	Predefinição
J4 lado esquerdo/baixo	Nenhum
J4 lado esquerdo/esquerdo	Nenhum
J4 lado esquerdo/direito	Nenhum
J4 botão 1	0 teclado numérico
J4 botão 2	Del teclado numérico
J4 botão 3	Return teclado numérico
J4 botão 4	W
J4 botão 5	n/a
J4 botão 6	n/a
J4 botão 7	n/a
J4 botão 8	n/a
J4 botão 9	n/a
J4 botão 10	n/a
J4 botão 11	n/a
J4 botão 12	n/a
J4 botão 13	n/a
J4 botão 14	n/a
J4 botão 15	n/a
J4 botão 16	n/a
J4 start	4
J4 select	8
J4 pedal 1 analógico	n/a
J4 pedal 1 analógico inc	0 teclado numérico
J4 pedal 1 analógico dec	Nenhum
J4 pedal 2 analógico	n/a
J4 pedal 2 analógico inc	Del teclado numérico
J4 pedal 2 analógico dec	Nenhum
J4 pedal 3 analógico	Nenhum
J4 pedal 3 analógico inc	Enter teclado numérico
J4 pedal 3 analógico dec	Nenhum
Acionador 4 analógico	n/a
Acionador 4 analógico inc	Nenhum
Acionador 4 analógico dec	Nenhum
Acionador V 4 analógico	n/a
Acionador V 4 analógico inc	Nenhum
Acionador V 4 analógico dec	Nenhum
Posicionamento 4 analógico	n/a
Posicionamento 4 analógico inc	Nenhum
Posicionamento 4 analógico dec	Nenhum
Posicionamento V 4 analógico	n/a
Posicionamento V 4 analógico inc	Nenhum
Posicionamento V 4 analógico dec	Nenhum
Rotativo 4 analógico	n/a
Rotativo 4 analógico inc	Nenhum
Rotativo 4 analógico dec	Nenhum
Rotativo V 4 analógico	n/a
Rotativo V 4 analógico inc	Nenhum
Rotativo V 4 analógico dec	Nenhum
Trilha X 4 analógico	n/a

continua na próxima página

Tabela 4 – continuação da página anterior

Controles do jogador 4	Predefinição
Trilha X 4 analógico inc	Nenhum
Trilha X 4 analógico dec	Nenhum
Trilha Y 4 analógico	n/a
Trilha Y 4 analógico inc	Nenhum
Trilha Y 4 analógico dec	Nenhum
Controle AD X 4 analógico	n/a
Controle AD X 4 analógico inc	Nenhum
Controle AD X 4 analógico dec	Nenhum
Controle AD Y 4 analógico	n/a
Controle AD Y 4 analógico inc	Nenhum
Controle AD Y 4 analógico dec	Nenhum
Pistola de luz X 4 analógico	n/a
Pistola de luz X 4 analógico inc	Nenhum
Pistola de luz X 4 analógico dec	Nenhum
Pistola de luz Y analógico	n/a
Pistola de luz Y analógico inc	Nenhum
Pistola de luz Y analógico dec	Nenhum
Mouse X 4 analógico	n/a
Mouse X 4 analógico inc	Nenhum
Mouse X 4 analógico dec	Nenhum
Mouse Y 4 analógico	n/a
Mouse Y 4 analógico inc	Nenhum
Mouse Y 4 analógico dec	Nenhum

As predefinições para o jogador 5 em diante estão vazias e podem ser customizadas conforme a necessidade.

- **Outros controles**

Muda a configuração dos botões usados para crédito, serviço, início de jogadores, etc. Abaixo a lista das opções predefinidas que podem ser alteradas na própria interface do MAME.

Inicia jogador 1	1
Inicia jogador 2	2
Inicia jogador 3	3
Inicia jogador 4	4
Inicia jogador 5	Nenhum
Inicia jogador 6	Nenhum
Inicia jogador 7	Nenhum
Inicia jogador 8	Nenhum
Ficha 1	5
Ficha 2	6
Ficha 3	7
Ficha 4	8
Ficha 5	Nenhum
Ficha 6	Nenhum
Ficha 7	Nenhum
Ficha 8	Nenhum
Ficha 9	Nenhum
Ficha 10	Nenhum
Ficha 11	Nenhum

continua na próxima página

Tabela 5 – continuação da página anterior

Ficha 12	Nenhum
Conta 1	Backspace
Serviço 1	9
Serviço 2	0
Serviço 3	Tecla menos
Serviço 4	Tecla igual
Tilt 1	T
Tilt 2	Nenhum
Tilt 3	Nenhum
Tilt 4	Nenhum
Liga	F1
Desliga	F2
Serviço	F2
Tilt	T
Trava interna da porta	Nenhum
Redefine a memória	F1
Abaixa o volume	Tecla menos
Aumenta o volume	Tecla igual
Teclado numérico	Nenhum
Teclado	None

Opções avançadas

Opções de desempenho

- *Salto automático dos quadros*
Ignora quadros de forma automática visando manter a velocidade da emulação.
O valor predefinido é Desligado.
- *Salto de quadro*
Define uma quantidade fixa de quadros a serem ignorados visando manter a velocidade da emulação.
O valor predefinido é 0
- *Supressão da velocidade*
Ativa a supressão de velocidade da emulação para que o sistema emulado rode na sua velocidade nativa em vez da velocidade do processador onde o sistema está sendo emulado.
O valor predefinido é Ligado.
- *Mute quando a supressão da velocidade estiver desligada*
Silencia o áudio quando a supressão de velocidade estiver desligado.
O valor predefinido é Desligado.
- *Dormir*
Reduz o consumo de processamento quando o MAME estiver parado sem fazer nada.
O valor predefinido é Ligado.
- *Velocidade*
Controla a velocidade do jogo com relação ao tempo de emulação.
O valor predefinido é 1
- *Ajuste a velocidade para bater com a taxa de atualização*
Controla a velocidade da emulação de forma automática mantendo a taxa de atualização de tela mais lenta em referência com a taxa de atualização de tela do computador que está rodando a emulação.
O valor predefinido é Desligado.
- *Baixa latência*
Reduz a latência (atraso) dos dispositivos de entrada como joysticks por exemplo. Para mais informações consulte *-[no]lowlatency*.

Opções da Rotação da Tela

- *Rotação*

Permite que a orientação da tela mude conforme a orientação de tela do jogo.

O valor predefinido é Ligado.

- *Rotacione à direita*

Rotacione a tela em 90 graus sentido horário.

O valor predefinido é Desligado.

- *Rotacione à esquerda*

Rotacione a tela em 90 graus sentido anti-horário.

O valor predefinido é Desligado.

- *Auto rotacione à direita*

Rotacione automaticamente a tela em 90 graus sentido horário caso a tela esteja orientada verticalmente.

O valor predefinido é Desligado.

- *Auto rotacione à esquerda*

Rotacione automaticamente a tela em 90 graus sentido anti-horário caso a tela esteja orientada verticalmente.

O valor predefinido é Desligado.

- *Vire o eixo X*

Inverte a tela da esquerda para a direita.

O valor predefinido é Desligado.

- *Vire o eixo Y*

Inverte a tela da direita para a esquerda.

O valor predefinido é Desligado.

Opções das ilustrações

- *Aproxime a área da tela*

Aproxima a região da tela emulada quando estiver numa ilustração.

O valor predefinido é Desligado.

Opções do estado/reprodução

- *Salve/restaure automático*
Em sistema compatíveis, carrega automaticamente o estado do sistema e a salva ao encerrar.
O valor predefinido é Desligado.
- *Permita o rebobinamento*
Permite o rebobinamento do estado do sistema.
O valor predefinido é Desligado.
- *Capacidade de rebobinamento*
Reserva uma quantidade em Megabytes da memória para rebobinamento.
O valor predefinido é 100
- *Filtro bilinear para as capturas da tela*
Define se os vídeos ou as capturas da tela terão o filtro aplicado.
O valor predefinido é Ligado.
- *Marca de queimado*
Cria uma captura da tela com marcas de fósforo queimado.
O valor predefinido é Desligado.

Opções da entrada

- *Trava da ficha*
Faz com que o sistema ignore a inserção de fichas em momentos que o sistema não está pronto para recebê-los.
O valor predefinido é Ligado.
- *Mouse*
Permite o uso de um mouse nos sistemas.
O valor predefinido é Desligado.
- *Controle*
Permite o uso de um controle nos sistemas.
O valor predefinido é Ligado.
- *Pistola de luz*
Ativa o uso de uma pistola de luz.
O valor predefinido é Desligado.
- *Mais de um teclado*
Permite o uso de mais de um teclado para cada entrada compatível.
O valor predefinido é Desligado.

- *Mais de um mouse*

Permite o uso de mais de um mouse para cada entrada compatível.

O valor predefinido é Desligado.

- *Steadykey*

Alguns sistemas exigem que dois ou mais botões sejam pressionados exatamente ao mesmo tempo para realizar movimentos ou comandos especiais. Devido a limitação do hardware do teclado, pode ser difícil ou até mesmo impossível de realizar usando um teclado comum. Essa opção seleciona diferentes modos de manuseio o que torna mais fácil registrar o pressionamento simultâneo das teclas, porém tem a desvantagem de deixar a sua capacidade de resposta mais lenta.

O valor predefinido é Desligado.

- *IU ativa*

Ativa a opção para que a interface do usuário se sobreponha a do teclado emulado caso esteja presente.

O valor predefinido é Desligado.

- *Recarga fora da tela*

Converte o botão 2 da pistola de luz como recarga fora da tela.

O valor predefinido é Desligado.

- *Zona morta do controle*

Permite fazer o ajuste fino do ponto morto do controle ou manche.

O valor predefinido é 0.3

- *Saturação do controle*

Faz o ajuste fino do eixo de fim de curso do controle.

O valor predefinido é 0.85

- *Teclado natural*

Ativa ou não o uso de um teclado natural.

O valor predefinido é Desligado.

- *Permite a entrada contraditória do comando*

Aceita comandos contraditórios e simultâneos no controle digital como esquerda e direita ou cima e baixo.

O valor predefinido é Desligado.

- *Impulso da ficha*

Define o tempo de impulso da ficha.

O valor predefinido é 0

4.4.7 Plug-ins

Consulte *PLUG-INS*.

4.4.8 Dispositivos na entrada

Aqui serão listados todos os dispositivos conectados ao sistema, como o teclado, o mouse, etc. A lista é dinâmica e varia dependendo do sistema e do que está sendo conectado nele.

Salva as definições

Salva todas as alterações feitas.

Retorna ao menu anterior

Retorna para a tela anterior.

4.4.9 Configurações do sistema

Permite que você configure individualmente cada sistema selecionado.

- *BIOS*
Informa se o sistema usa uma BIOS ou não, nos sistema que usam BIOS é possível escolher qual BIOS você quer que o sistema use.
- *Opções Avançadas*
Consulte *Opções avançadas*.
- *Opções do Vídeo*
Consulte *Opções do vídeo*.
- *Opções dos dispositivos na entrada*
Consulte *Opções dos dispositivos na entrada*.
- *Adiciona aos favoritos*
Adiciona o sistema selecionado aos seus favoritos.
- *Salva as configurações do sistema*
Salva a configuração apenas para o sistema selecionado.

4.5 Como o MAME localiza os seus arquivos?

- *Introdução*
 - *Terminologia*
- *Opções do caminho para fazer a busca*
- *Arquivos compactados*
- *Como o MAME procura por uma mídia?*
 - *As ROMs dos sistemas*
 - *ROMs do dispositivo*
 - *As ROMs dos itens dos programas*
 - *As imagens de disco no formato CHD*
 - *Programas avulsos*
 - *Diagnosticando uma mídia ausente*

4.5.1 Introdução

Ao contrário dos típicos programas de computador onde você navega pelo seu disco e seleciona um arquivo para abrir ou um local para salvar, o MAME possui configurações que indicam onde procurar os arquivos que ele usa. Você pode alterar estas configurações iniciando o MAME sem qualquer parâmetro, depois selecionando *Configurações* e, em seguida, selecionando *Configuração dos diretórios* (lembre-se de clicar em *Salva a configuração* para aplicar as suas alterações). Você também pode alterar as configurações editando diretamente os arquivos `mame.ini` e `ui.ini` ou usando as respectivas opções diretamente na linha de comando ou dos outros arquivos de configuração. Para obter mais informações sobre todas as opções disponíveis para controlar onde o MAME deve procurar pelos arquivos e os diferentes tipos de configurações, consulte os capítulos *Opções para a configuração dos principais diretórios* e o *Diversos arquivos de configuração*.

Terminologia

Precisamos compreender algumas terminologias utilizadas nas explicações mais adiante:

Sistema

Um sistema é um conjunto completo que pode ser emulado pelo MAME. Alguns sistemas executam um programa (*software*) fixo, enquanto outros podem carregar programas a partir de um catálogo de programas e/ou os arquivos de uma mídia.

Dispositivo

Um componente emulado que pode ser usado por vários sistemas ou por outros dispositivos. Alguns dispositivos precisam de uma cópia da imagem de uma ROM e alguns dispositivos permitem que um programa possa ser lido a partir de uma lista adicional de programas que serão usadas com um sistema específico.

Sistema principal

O MAME usa uma relação de principal/clone para agrupar os sistemas relacionados. Um sistema no grupo é escolhido para ser o *principal* e os outros são chamados de *clones*. (A escolha do sistema principal é um tanto arbitrária. Não é necessariamente a original ou a variante definitiva).

BIOS do sistema

Um sistema configurado sem programa. Isso é geralmente aplicável em sistemas do tipo arcade que usavam cartuchos com jogos intercambiáveis ou placas com as ROMs. Observe que isso *não* é o mesmo que as configurações da seleção da BIOS que permitem a escolha de uma ROMs para a inicialização do sistema (como os sistemas Neo Geo por exemplo) ou um *firmware* do dispositivo.

O item de um programa

Um pacote de um programa descrito num catálogo de programas. Os itens dos programas podem consistir em diversas *partes* que podem ser montadas de forma individual. Devido a grande variedade de mídia compatível com o MAME, as partes do programa podem usar diferentes *carregadores*. Neles se inclui os *carregadores de ROM*, tipicamente utilizado por cartuchos e o *carregador de arquivo de uma imagem*, usado por partes do programa que consistem numa única imagem de uma mídia (como a imagem de um disquete, de um CD-ROM, de uma fita cassete, etc.).

O item de um programa principal

Os itens dos programas relacionados são agrupados usando relacionamentos do tipo principal/clone, numa forma semelhante aos sistemas relacionados. Isso é geralmente utilizado para agrupar as diferentes versões ou os diferentes lançamentos de um mesmo tipo de programa. Caso o programa tenha um item principal, ele sempre estará no mesmo catálogo de programas.

Nome abreviado

O MAME utiliza *nomes abreviados* para identificar de forma única, os sistemas e os dispositivos pelo item de um programa dentro de um catálogo e para também, identificar de forma única as partes do programa dentro de um programa.

É possível ver o nome abreviado de um sistema destacando-o no menu de seleção do sistema, garantindo que o painel de informações esteja visível à direita e mostrando as *Informações gerais* na aba de *Informações*. Por exemplo, o nome abreviado para o **Nintendo Virtual Boy** é **vboy**. O nome abreviado do sistema e do dispositivo também podem ser vistos na saída das várias opções da linha de comando, incluindo as opções `-listxml`, `-listfull`, `-listroms` e `-listcrc`.

Você pode ver os nomes abreviados para um programa e em qual catálogo ele pertence, destacando-o no menu do catálogo de programas, garantindo que o painel de informações esteja visível à direita, fazendo com que seja mostrado as *Informações do catálogo de programas* na aba de *Informações*. Por exemplo, o nome abreviado para o sistema **Macintosh 6.0.3** é **sys603** e o nome abreviado para o catálogo de programas fica no arquivo `mac_flop.xml`. Os nomes abreviados do catálogo correspondem ao nome do arquivo (por exemplo, o catálogo de programas do cartucho do Sega Mega Drive/Genesis é um arquivo chamado de `megadriv.xml` e o seu nome abreviado é `megadriv`). Você também pode ver as listas abreviadas do catálogo de programas, os programas e suas partes ao procurar pelo atributo `name` nos arquivos XML do catálogo de programas que ficam dentro da pasta **hash**.

4.5.2 Opções do caminho para fazer a busca

A maioria das opções para definir os locais de busca, permitem que vários diretórios sejam adicionados desde que eles sejam separados por ponto e vírgula (;). As variáveis de ambiente são expandidas usando a sintaxe do prompt de comando no Windows ou no terminal nos sistemas UNIX.

Os caminhos relativos são interpretados em relação ao diretório atual de trabalho no momento do uso. No Windows, caso o executável do MAME seja iniciado com um duplo clique, o diretório de trabalho será este onde o executável do MAME estiver. Caso inicie o MAME clicando duas vezes nele através do *Finder* do macOS ou na maioria das distribuições Linux, o seu **home** será definido como o diretório de trabalho.

4.5.3 Arquivos compactados

O MAME pode carregar arquivos compactados do tipo PKZIP (ZIP) e LZMA (7-Zip) (estes devem ter as extensões .zip e .7z respectivamente). Uma série de extensões são compatíveis com o formato PKZIP, incluindo o Zip64 para arquivos grandes, registros de data e hora (*timestamps*) do NTFS e a compressão LZMA. Apenas o formato ASCII pode ser usado nos nomes dos arquivos ou UTF-8 nos arquivos do tipo PKZIP (os arquivos 7-Zip sempre usam nomes de arquivo com UTF-16).

Porém, o MAME **não carrega** arquivos que estejam agrupados e compactados num único arquivo, por exemplo, quando todas as ROMs do Master System estiverem dentro de um arquivo chamado `sms.zip` ou quando todas as ROMs do Mega Drive estiverem compactadas dentro de um arquivo chamado `megadriv.zip` e assim por diante. O MAME não possui compatibilidade com arquivos dividido em vários segmentos (arquivos divididos em partes como, .zip.001, .zip.002, *-part.001, *-part.002, etc.), com arquivos criptografados e com o método legado de compactação do tipo "implosão" nos arquivos PKZIP.

O desempenho do MAME pode ser reduzido ao utilizar arquivos compactados com uma grande quantidade de itens dentro deste arquivo. Os arquivos compactados que usam o algoritmo de compactação LZMA (7-zip) utilizam a CPU de forma mais intensa durante o processo de descompactação se comparado aos arquivos compactados usando algoritmos mais simples como o PKZIP. O layout do arquivo não é levado em consideração durante o carregamento dos arquivos, então o uso de "compactação sólida" geralmente resulta no MAME descompactando os mesmos dados repetidamente durante o carregamento da mídia.

4.5.4 Como o MAME procura por uma mídia?

Use a opção *rompath* para definir o caminho das pastas onde as imagens das ROMs, as imagens de disco e outras mídias devem ser procuradas. É predefinido que o MAME procure pela mídia numa pasta chamada **roms** dentro do diretório de trabalho. Para o propósito desta discussão, o disquete, a fita cassete, a fita de papel e as outras imagens de uma mídia que não sejam armazenados no formato CHD são tratados como cópias das ROMs.

Ao procurar por uma cópia (*dump*) do sistema, do dispositivo ou do programa, o MAME trata as pastas e arquiva dentro das pastas configuradas pela opção *rompath* como equivalente, mas lembre-se da limitação onde o MAME não pode carregar os arquivos de um arquivo contido em outro arquivo (um arquivo compactado que tenha outro arquivo compactado dentro). O MAME procura primeiro uma pasta, depois procura por um arquivo PKZIP e, finalmente, um arquivo 7-Zip. Ao procurar uma cópia da ROM num arquivo, o MAME primeiro procura um arquivo com um determinado nome e com um determinado CRC. Caso o arquivo correspondente não seja encontrado, o MAME procura por um arquivo com um determinado CRC ignorando o nome. Caso nenhum arquivo correspondente seja encontrado, o MAME finalmente procura por um arquivo com um determinado nome, ignorando o CRC.

Embora o MAME possa carregar imagens de disco no formato CHD de dentro dos arquivos, isso não é recomendado. Os arquivos CHD já possuem dados compactados, o formato em si já está compactado e os dados são armazenados de uma maneira permite o acesso aleatório dos arquivos. Caso uma imagem de disco no formato CHD esteja armazenada num arquivo PKZIP ou num 7-Zip, o MAME precisa primeiro,

carregar o arquivo inteiro na memória para poder usá-lo. Para casos como um disco rígido ou para imagens de um LaserDisc, isso provavelmente usará uma quantidade de espaço grande no arquivo de troca (*swap*), prejudicando o desempenho e possivelmente reduzindo a expectativa de vida útil dos seus discos ou SSDs. É melhor manter as imagens CHD sem compressão dentro das suas respectivas pastas.

As ROMs dos sistemas

Para cada pasta configurada em `rompath`, o MAME procura as ROMs do sistema nos seguintes locais:

- Numa pasta ou num arquivo correspondente ao nome do próprio sistema.
- Numa pasta ou num arquivo correspondente ao nome do sistema principal do sistema, caso seja aplicável.
- Numa pasta ou num arquivo correspondente ao nome da BIOS do sistema correspondente, caso seja aplicável.

Usando o sistema **Shiritsu Justice Gakuen** como exemplo, o MAME buscará pelas ROMs do sistema nesta ordem:

- O nome abreviado do sistema é `jgakuen`, então o MAME irá procurar por uma pasta chamada **jgakuen**, um arquivo PKZIP chamado `jgakuen.zip` ou um arquivo 7-Zip chamado `jgakuen.7z`.
- O sistema principal é a versão europeia do **Rival Schools**, que tem o nome abreviado `rvschool`, então o MAME irá procurar por uma pasta chamada **rvschool**, um arquivo PKZIP chamado `rvschool.zip` ou um arquivo 7-Zip chamado `rvschool.7z`.
- A BIOS do sistema correspondente é a placa Capcom ZN2, que tem o nome abreviado `coh3002c`, então o MAME irá procurar uma pasta chamada **coh3002c**, um arquivo PKZIP chamado `coh3002c.zip` ou um arquivo 7-Zip chamado `coh3002c.7z`.

ROMs do dispositivo

Para cada caminho configurado em `rompath`, o MAME procurará pelas ROMs dos dispositivos nos seguintes locais:

- Numa pasta ou num arquivo correspondente ao nome do dispositivo.
- Numa pasta ou num arquivo correspondente ao nome da ROM do dispositivo principal do dispositivo, caso seja aplicável.
- Numa pasta ou num arquivo correspondente ao nome do sistema.
- Numa pasta ou num arquivo correspondente ao nome do sistema principal do sistema, caso seja aplicável.
- Numa pasta ou num arquivo correspondente ao nome da BIOS do sistema correspondente, caso seja aplicável.

Usando o clone **Unitron 1024 Macintosh** com um teclado francês **Macintosh Plus** com um teclado numérico como exemplo, o MAME procurará a ROM do microcontrolador do teclado da seguinte maneira:

- O nome abreviado do teclado francês **Macintosh Plus** é `mackbd_m0110a_f`, então o MAME procurará uma pasta chamada **mackbd_m0110a_f**, um arquivo PKZIP chamado `mackbd_m0110a_f.zip` ou um arquivo 7-Zip chamado `mackbd_m0110a_f.7z`.
- A ROM principal do dispositivo é o teclado U.S. **Macintosh Plus** com um teclado numérico, que tem o nome abreviado `mackbd_m0110a`, então o MAME procurará por uma pasta chamada **mackbd_m0110a**, um arquivo PKZIP chamado `mackbd_m0110a.zip` ou um arquivo 7-Zip chamado `mackbd_m0110a.7z`.

- O nome abreviado do sistema **Unitron 1024** é `utr1024`, então o MAME procurará por uma pasta chamada **utr1024**, um arquivo PKZIP chamado `utr1024.zip` ou um arquivo 7-Zip chamado `utr1024.7z`.
- O sistema principal do **Unitron 1024** é o **Macintosh Plus**, que tem o nome abreviado `macplus`, então o MAME irá procurar por uma pasta chamada **macplus**, um arquivo PKZIP chamado `macplus.zip` ou um arquivo 7-Zip chamado `macplus.7z`.
- Caso não haja uma BIOS que corresponda ao sistema, então o MAME não fará mais buscas.

As ROMs dos itens dos programas

Para cada pasta configurada em `rompath`, o MAME procura pelos itens dos programas do sistema nos seguintes locais:

- Numa pasta ou num arquivo correspondente ao nome do item dos programas correspondente dentro de uma pasta correspondente ao nome abreviado do catálogo de programas (ou uma pasta correspondente ao nome abreviado do item do programa dentro de um arquivo correspondente ao nome do catálogo de programas).
- Numa pasta ou num arquivo correspondente ao nome do item do programa principal dentro de uma pasta que corresponda ao nome abreviado do catálogo de programas, caso seja aplicável (ou uma pasta correspondente ao nome abreviado do item do software principal num arquivo que corresponda ao nome do catálogo de programas).
- Numa pasta ou num arquivo correspondente ao nome do item do programa (assim é feito por questão de conveniência dos itens do programa que também são executados como sistemas autônomos com o mesmo nome abreviado, como nos jogos de Neo Geo).
- Numa pasta ou num arquivo correspondente ao nome do item do programa principal, caso seja aplicável. (Isso é utilizado para conveniência dos itens do programa que também são executados como sistemas autônomos com o mesmo nome abreviado, como nos jogos de Neo Geo).

Ao carregar a versão alemã do **Dune II** do cartucho do Mega Drive/Genesis a partir do catálogo de programas na versão PAL do Mega Drive, o MAME procurará a ROM do cartucho ROM da seguinte maneira:

- O nome abreviado do item do programa para a versão alemã do **Dune II** é `dune2g` e o nome abreviado da ROM do cartucho do Mega Drive/Genesis no catálogo de programas é `megadriv`, então o MAME procurará por uma pasta chamada **dune2g**, um arquivo PKZIP chamado `dune2g.zip` ou um arquivo 7-Zip chamado `dune2g.7z` dentro de uma pasta chamada **megadriv** (ou uma pasta chamada **dune2g** dentro de um arquivo PKZIP chamado `megadriv.zip` ou um arquivo 7-Zip chamado `megadriv.7z`).
- O item do programa principal é a versão PAL europeia genérica do **Dune II** no mesmo catálogo de programas, que tem o nome abreviado `dune2`, então o MAME procurará uma pasta chamada **dune2**, um arquivo PKZIP chamado `dune2.zip` ou um arquivo 7-Zip chamado `dune2.7z` dentro de uma pasta chamada **megadriv** (ou uma pasta chamada **dune2** dentro de um arquivo PKZIP chamado `megadriv.zip` ou um arquivo 7-Zip chamado `megadriv.7z`).
- O próximo MAME irá ignorar o nome abreviado do catálogo de programas e usará apenas o nome abreviado do item do programa, procurando por uma pasta chamada **dune2g**, um arquivo PKZIP chamado `dune2g.zip` ou um arquivo 7-Zip chamado `dune2g.7z`.
- Ainda ignorando o nome abreviado do catálogo de programas, o MAME usará apenas o nome abreviado do programa principal, procurando uma pasta chamada **dune2**, um arquivo PKZIP chamado `dune2.zip` ou um arquivo 7-Zip chamado `dune2.7z`.

As imagens de disco no formato CHD

O MAME procura pelos sistemas, pelos dispositivos e os itens do programa através das as imagens de disco no formato CHD, com algumas diferenças:

- Para os sistemas e para os itens de um programa, o MAME verificará o sistema ou programa principal se aplicável com os nomes alternativos para uma imagem de disco com o mesmo conteúdo. Isso permite que você mantenha uma única cópia de uma imagem de disco no formato CHD para um sistema principal, um item de um programa ou quaisquer clones que dependam de uma imagem de disco com o mesmo conteúdo, independentemente do nome esperado pelos clones.
- Para os itens de um programa, o MAME procurará imagens de disco no formato CHD numa pasta que corresponda ao nome abreviado no catálogo de programas. Isso é utilizado para conveniência quando todos os itens num catálogo de programas que tenham apenas uma única imagem de disco cada em formato CHD.
- Recomendamos que você **não comprima** as imagens de disco em formato CHD num arquivo PKZIP ou num arquivo 7-Zip. No entanto, caso isso seja feito, o MAME só conseguirá encontrar os discos em formato CHD dentro dos arquivos com um determinado nome. Isto ocorre porque o MAME utiliza o conteúdo do cabeçalho CHD e uma soma de verificação (*checksum*) do próprio arquivo CHD. A soma de verificação do próprio arquivo CHD pode variar dependendo das opções de compressão.

Visando a economia de espaço, o MAME permite que os arquivos delta CHD sejam usados por sistemas clones, dispositivo ROM principal e clones dos programas. O arquivo delta CHD depende de um CHD principal do sistema, ROM ou programa. A economia de espaço dependerá de quanto conteúdo poderá ser aproveitado do arquivo CHD principal. O MAME faz a busca nos mesmos locais que ele procuraria pelos arquivos CHD principais. Para obter mais informações consulte também no capítulo sobre [ROMs e CHDs](#).

Programas avulsos

Muitos sistemas são compatíveis com o carregamento da mídia a partir de um arquivo informando o caminho do arquivo na linha de comando usando uma das opções de uma mídia. Os caminhos relativos são interpretados em relação ao diretório de trabalho atual.

Você pode definir um caminho para um arquivo dentro de um arquivo PKZIP ou 7-Zip da mesma maneira que definir um caminho para um arquivo numa pasta (lembre-se de que você pode ter no máximo um único arquivo num caminho, pois o MAME não suporta o carregamento de arquivos de arquivos contidos em outros arquivos). Caso defina um caminho para um arquivo PKZIP ou um arquivo 7-Zip, o MAME usará o primeiro arquivo encontrado no arquivo (isso depende da ordem onde os arquivos são armazenados no arquivo, é mais útil para arquivos contendo um único arquivo).

Inicie o Nintendo Entertainment System/Famicom com o arquivo **amazon_diet_EN.nes** montado no slot do cartucho:

```
mame nes -cart amazon_diet_EN.nes
```

Inicie o sistema Osborne-1 com o primeiro arquivo do arquivo os1xutils.zip montado na primeira unidade de disquete:

```
mame osborne1 -flop1 os1xutils.zip
```

Inicie o sistema **Macintosh Plus** com o arquivo system tools.img no arquivo sys603.zip montado na primeira unidade de disquete:

```
mame macplus -flop1 "sys603.zip/system tools.img"
```

Diagnosticando uma mídia ausente

Ao iniciar um sistema a partir do menu de seleção do MAME ou a partir da seleção do menu do catálogo de programas, o MAME irá listar qualquer sistema, cópia da ROM do dispositivo (*dump*) ou as imagens ausentes do disco, desde que haja pelo menos uma cópia da imagem da ROM ou uma imagem de disco presente para o sistema. Para os clones do sistemas, pelo menos uma cópia da ROM deve estar presente ou uma imagem de disco *exclusiva para o clone*, para que o MAME possa listar as ROMs e as imagens de disco que estiverem ausentes.

Caso todas as imagens de disco, todas as ROMs do sistema e todas as ROMs do dispositivo estejam presentes e o sistema estiver sendo iniciado com um item de um programa, o MAME verificará se a ROM e a imagem do disco para o item do programa estão presentes. Se pelo menos uma ROM ou uma imagem de disco para o item de um programa estiver presente, o MAME listará todas as ROMs ou as imagens de disco que estiverem ausentes.

Por exemplo, se você tentar iniciar o sistema **Macintosh Plus** e a ROM do microcontrolador do teclado estiver ausente, o MAME exibe a seguinte mensagem de erro:

```
As imagens ROM/disco necessárias para o sistema selecionado ou estão  
faltando ou estão incorretas.
```

```
Adquira os arquivos corretos ou selecione um sistema diferente.
```

```
O arquivo 341-0332-a.bin (mackbd_m0110a) não foi encontrado
```

```
Pressione qualquer tecla para continuar.
```

O nome da ROM que falta é mostrado (341-0332-a.bin) bem como o nome abreviado do dispositivo ao qual ele pertence (mackbd_m0110a). Quando uma ROM está ausente ou a imagem do disco não for específico para o sistema selecionado, o nome abreviado do sistema ou do dispositivo ao qual ele pertence será mostrado.

Caso inicie um sistema no MAME a partir do prompt de comando ou do terminal, o MAME mostrará onde ele procurou pelas ROMs ou pelas imagens de disco que não foram encontradas.

Usando o exemplo de um clone do **Unitron 1024 Macintosh** com um teclado francês conectado, o MAME mostrará as seguintes mensagens de erro caso as ROMs não estejam presentes:

```
mame utr1024 -kbd frp  
342-0341-a.u6d NOT FOUND (tried in utr1024 macplus)  
342-0342-a.u8d NOT FOUND (tried in utr1024 macplus)  
341-0332-a.bin NOT FOUND (tried in mackbd_m0110a_f mackbd_m0110a utr1024 macplus)
```

O MAME utilizou o nome abreviado do sistema utr1024 e o nome abreviado do sistema principal macplus durante a procura pelas ROMs do sistema. Ao pesquisar pela ROM do microcontrolador do teclado, o MAME utilizou o nome abreviado do dispositivo mackbd_m0110a_f, o nome abreviado da ROM dispositivo principal mackbd_m0110a, o nome abreviado do sistema utr1024 e o nome abreviado do sistema principal macplus.

As partes do programa que utilizam um carregador de ROM (geralmente uma mídia em cartucho) mostra uma mensagem similar quando as ROMs não são encontradas. Usando o exemplo da versão alemã do **Dune II** num Mega Drive PAL, o MAME mostrará as seguintes mensagens de erro caso nenhuma ROM esteja presente:

```
mame megadriv dune2g
mpr-16838-f.u1 NOT FOUND (tried in megadriv\dune2g megadriv\dune2 dune2g dune2 megadriv_
↳genesis)
Fatal error: Required files are missing, the machine cannot be run.
```

O MAME procurou a ROM do cartucho usando:

- O nome abreviado do catálogo de programas do megadriv e o nome abreviado do programa dune2g.
- O nome abreviado do catálogo de programas megadriv e o nome abreviado do programa principal dune2.
- Apenas o nome abreviado do programa dune2g.
- Apenas o nome abreviado do programa principal dune2.
- Os locais que seriam procurados pelo sistema PAL do Mega Drive (o sistema megadriv com o nome abreviado e o nome abreviado do sistema principal genesis).

As partes do programa que usam o carregador de arquivos de imagem (incluindo disquetes e uma mídia em fita cassete) só verificam a mídia depois que as imagens da ROM forem carregadas e os arquivos da mídia forem mostrados de forma diferente. Usando o exemplo do **Macintosh System 6.0.3**, o MAME mostrará estas mensagens de erro caso o programa esteja faltando:

```
mame macplus -flop1 sys603:flop1
:fdc:0:35dd: error opening image file system tools.img: No such file or directory_
↳(generic:2) (tried in mac_flop\sys603 sys603 macplus)
Fatal error: Device Apple/Sony 3.5 DD (400/800K GCR) load (-floppydisk1 sys603:flop1)_
↳failed: No such file or directory
```

As mensagens de erro mostram onde o MAME fez a procura pelo arquivo de imagem. Neste caso, ele usou o nome abreviado do catálogo de programas mac_flop e o nome abreviado do programa sys603, apenas o nome abreviado do programa sys603 e os locais onde as ROMs do sistema seriam procuradas.

4.6 Front-ends

Existem várias ferramentas de terceiros desenvolvidas para o MAME que servem para tornar a vida do usuário mais agradável no que tange a seleção dos diferentes sistemas, programas, organização da sua lista de jogos preferidos dentre inúmeras funções extras.

Entre o usuário e o MAME é possível usar ferramentas chamadas *Front-ends*¹, elas funcionam como uma interface intermediária e há muitas delas disponíveis para listá-las todas aqui. Elas existem aos milhares e seria inviável tentar listá-las todas aqui.

Algumas delas são gratuitas porém nem todas elas são. Algumas *front-ends* mais antigas antecedem a fusão de MAME e MESS e não suportam as novas funcionalidades herdadas do MESS como consoles, portáteis, etc.

QMC2 (diversas plataformas)

Disponibiliza uma interface para a configuração de diversas configurações e características do MAME. Também inclui um gerenciador de ROMs e recursos de gerenciamento de mídia. É escrito em C++ usando o kit de ferramenta Qt, o [código-fonte está no SourceForge](#).

Negatron (diversas plataformas)

O *Negatron* enfatiza os recursos para a configuração de computadores e consoles. É escrito em Java, o [código-fonte está no GitHub](#).

BletchMAME (diversas plataformas)

O *BletchMAME* aproveita a interface de scripting Lua do MAME para se integrar de forma integrada e eficaz para substituir a interface de usuário interna do MAME. Possui muitas características úteis para emulação de computadores domésticos. É escrito em C++, o [código-fonte está no GitHub](#).

IV/Play (Microsoft Windows)

Um simples programa para o Windows para iniciar os sistemas no MAME. É escrito em C++, o [código-fonte está no GitHub](#).

Emu Loader (Microsoft Windows)

O *Emu loader* oferece uma interface Windows para iniciar sistemas e diversos emuladores, incluindo o MAME, Supermodel e o DEMUL. É escrito em Delphi Pascal, o [código-fonte está disponível na página de download do projeto](#).

Retrofire (Japanese, Microsoft Windows)

O *Retrofire* oferece uma interface gráfica em japonês para rodar sistemas ou programas no MAME.

A equipe MAME não oferece qualquer tipo de suporte para qualquer problema que você venha a ter com as sugestões dadas acima. Para receber suporte, sugerimos entrar em contato diretamente com o autor da interface ou buscar ajuda em qualquer um dos fóruns relacionados com o MAME espalhados pela Internet.

¹ Por ser um adjetivo, estamos seguindo [estas normas](#) e passamos a utilizar **front-end** e não mais **frontend**.

4.7 Sobre as ROMs e os seus conjuntos

O manuseio e a atualização das ROMs e dos seus respectivos conjuntos é provavelmente a questão que causa a maior confusão e problemas dentre os usuários do MAME, especialmente os marinheiros de primeira viagem.

Esta seção tem como objetivo esclarecer diversos tópicos sobre o assunto, para tirar muitas das dúvidas mais comuns sobre o assunto e para abordar detalhes que você precisa saber para poder usar o MAME de forma mais eficaz.

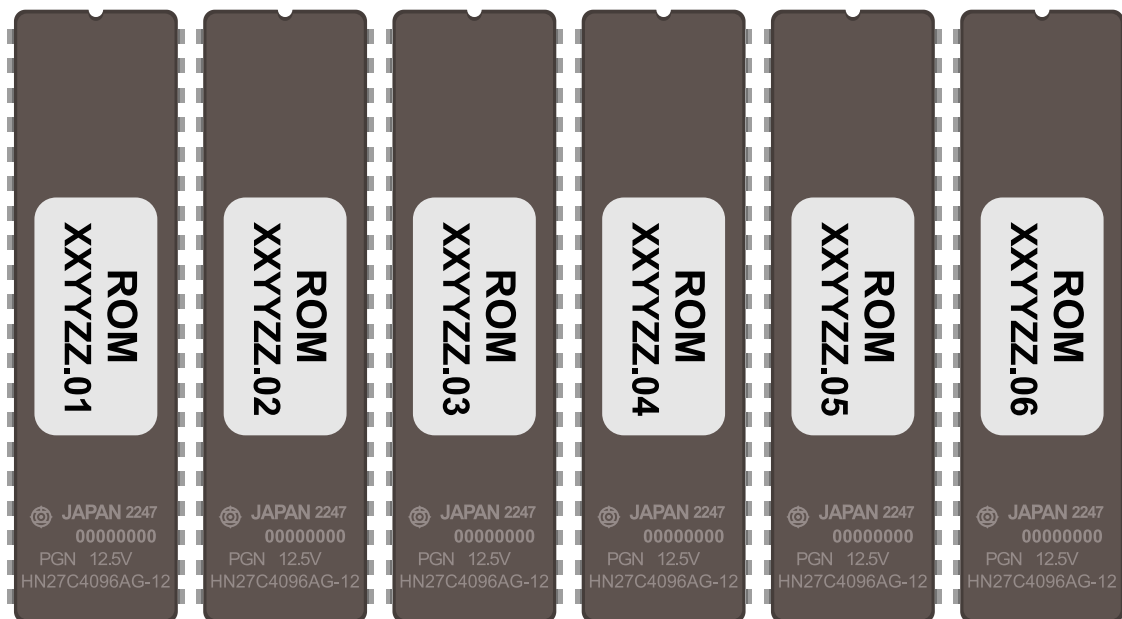
4.7.1 O que é uma imagem ROM?

A programação do jogo em si ficam armazenados dentro de circuitos integrados¹ na placa-mãe do arcade ou num outro tipo de dispositivo (CD-ROM, DVD-ROM, HDD, etc.). Nós chamamos estes arquivos de ROMs pois é um acrônimo de "*Read-Only Memory*" que significa memória de leitura apenas.

Na maioria dos consoles e portáteis, dentro dos cartuchos geralmente a programação do jogo vem num único CI (mas nem sempre). Já nos sistemas arcade a coisa é um pouco mais complicada devido ao seu design, o sistema em questão normalmente precisará ler todos os dados do jogo encontrado em diferentes CIs espalhados por toda a placa.

Alguns dos CIs usados para armazenar os dados não são regraváveis como um CI do tipo **PROM** por exemplo, uma vez gravados, os dados ficam armazenados de forma permanente (contanto que o CI não seja danificado ou caso envelheça até a pifar de vez!).

No exemplo abaixo podemos ver um exemplo de um conjunto de CIs do tipo **EPROM** que podem ser apagados através de uma luz ultravioleta e gravados novamente (as etiquetas são colocadas em cima destes CIs que contém uma abertura para realizar este apagamento, como proteção contra o apagamento involuntário ou acidental dos dados). Dada a questão da limitação da tecnologia e de espaço da época, os dados eram divididos em diferentes EPROMs.



¹ Estes circuitos integrados também são conhecidos pela abreviação "CI" (se fala CÊ-Í), assim como é chamado de "chip" em Inglês.

Posteriormente através de um leitor apropriado, os dados contidos dentro destas EPROMs são lidos e armazenado em forma de uma imagem binária. Este processo de se extrair a imagem de dentro destes CIs é chamado de "*dump*" ou *ROM dump*. As imagens depois de extraídas geralmente podem ser organizadas e armazenadas como `xxyyzz.01`, `xxyyzz.02`, `xxyyzz.03`, `xxyyzz.04`, `xxyyzz.05` e `xxyyzz.06` respectivamente. Após a conclusão de todo o processo e de todo um trabalho de aferição para saber se as imagens foram extraídas corretamente, os desenvolvedores registram o nome de cada imagem e os seus respectivos CRC² e SHA1³ dentro do código-fonte do respectivo driver para que o MAME saiba como carregá-los.

4.7.2 As ROMs e as suas "versões"

Há uma grande confusão com relação ao MAME e as "*versões*" das ROMs que o MAME utiliza, porém, vejamos o que o MAME leva em consideração ao aceitar uma ROM como válida. Usando a ROM **pacman** como exemplo, o banco de dados interno do MAME nos mostra o seguinte:

```
mame -lr pacman
ROMs required for driver "pacman".
Name                                     Size Checksum
pacman.6e                               4096 CRC(c1e6ab10) ↵
↳SHA1(e87e059c5be45753f7e9f33dff851f16d6751181)
pacman.6f                               4096 CRC(1a6fb2d4) ↵
↳SHA1(674d3a7f00d8be5e38b1fdc208ebef5a92d38329)
pacman.6h                               4096 CRC(bcdd1beb) ↵
↳SHA1(8e47e8c2c4d6117d174cdac150392042d3e0a881)
pacman.6j                               4096 CRC(817d94e3) ↵
↳SHA1(d4a70d56bb01d27d094d73db8667ffb00ca69cb9)
pacman.5e                               4096 CRC(0c944964) ↵
↳SHA1(06ef227747a440831c9a3a613b76693d52a2f0a9)
pacman.5f                               4096 CRC(958fedf9) ↵
↳SHA1(4a937ac02216ea8c96477d4a15522070507fb599)
82s123.7f                               32 CRC(2fc650bd) ↵
↳SHA1(8d0268dee78e47c712202b0ec4f1f51109b1f2a5)
82s126.4a                               256 CRC(3eb3a8e4) ↵
↳SHA1(19097b5f60d1030f8b82d9f1d3a241f93e5c75d6)
82s126.1m                               256 CRC(a9cc86bf) ↵
↳SHA1(bbcec0570aeceb582ff8238a4bc8546a23430081)
82s126.3m                               256 CRC(77245b66) ↵
↳SHA1(0c4d0bee858b97632411c440bea6948a74759746)
```

Para que a ROM **pacman** (do arquivo `pacman.zip`) seja considerada **válida**, várias características muito específicas devem bater com a lista acima como o nome de cada arquivo, o tamanho e o *checksum* (CRC + SHA1) de cada um deles.

Essa é a única informação que o MAME leva em consideração, **não existe** na lista acima qualquer informação de "*versão da ROM*" assim como o MAME não armazena tal informação em lugar nenhum, esta questão também já foi abordada com mais detalhes em [outro capítulo](#).

Para complicar a coisa ainda mais, temos mais duas questões, uma parte vem de **front-ends** e outros aplicativos que utilizam diferentes versões do MAME. A outra questão tem a ver com pessoas que compilam uma versão completa das ROMs conforme novas versões do MAME vão sendo lançadas visando "facilitar" todos aqueles que colecionam tais arquivos.

² Significa *Cyclic Redundancy Check* ou verificação cíclica de redundância, serve para aferir a integridade dos dados dos arquivos.

³ Significa *Secure Hash Algorithm* ou algoritmo de dispersão seguro, é uma função criptográfica que retorna um resultado com valor hexadecimal (hash) usado também para aferir a autenticidade dos dados dos arquivos.

Talvez por causa de desinformação, se espalhou pela internet um mito onde a ROM do **pacman**, por exemplo, "só vai funcionar" se a versão da ROM bater com a versão do MAME, como demonstrado acima, isso é uma mentira sem qualquer fundamento. Podemos ver, por exemplo, no site [Arcade Database](#) que nada foi alterado no jogo **pacman** entre as versões **0.196** -> **0.244** do MAME. Logo, contanto que os arquivos batam com a lista acima, o jogo **pacman** funcionará normalmente, independentemente da versão do MAME.

Portanto, para o usuário mais leigo, não é preciso atualizar todas as suas ROMs a cada nova versão do MAME, é muito provável que as ROMs que você mais usa e já tenha, continuarão a funcionar sem problemas. Caso os desenvolvedores do MAME atualizem alguma coisa no driver, no arquivo da ROM ou apareça um novo **ROM dump** (pois a versão anterior tinha algum tipo de problema), basta atualizar a ROM que parou de funcionar e não todo o conjunto de ROMs como muitos fazem. A mesma informação vale para aqueles que fazem coleção destes arquivos, não é preciso baixar todo um **ROMSET** para uma determinada versão do MAME, basta adicionar as novas ROMs e atualizar as já existentes (se for o caso), isso é facilmente gerenciável através dos [gerenciadores de ROMs](#).

4.7.3 A organização do conjunto das ROMs

Ao longo do desenvolvimento dos jogos arcade por exemplo, alguns deles passam por revisões, nestas revisões o código pode ter correções ou atualizações. Então muitas vezes se mantinha a placa original do arcade e trocava-se apenas a programação de um dos CIs relacionado com a atualização da programação do jogo. Há casos também onde um mesmo arcade será vendido em diferentes partes do mundo, assim a programação principal é mantida, porém, apenas algumas partes da programação é substituída, como o licenciamento, o nome da empresa licenciada, o idioma, o título, talvez os personagens e assim por diante.

Visando a economia de espaço, a estrutura interna do MAME foi organizada de maneira a utilizar um sistema hierárquico familiar de "parent" e "clone" ou podemos traduzir como **principal** e **clone**.

A última revisão corrigida de um determinado sistema será definido como a ROM **principal** desta família (*World*), mas nem sempre. Por exemplo, serão definidos como **clones** todos os conjuntos das ROMs que em geral usam exatamente os mesmos CIs, no entanto, caso haja dados que forem diferentes do conjunto principal em alguns deles (como a versão Japonesa do **Puckman** e a versão USA/World do **Pac Man**).

Ao rodar um jogo clone ou um dos seus conjuntos subsequentes sem antes ter o jogo principal disponível, o usuário será informado do problema. Usando o exemplo anterior, ao tentar jogar a versão Americana do **Pac Man** (pacman) sem antes ter a ROM principal puckman, aparecerá uma mensagem de erro informando quais são os arquivos que estão faltando.

Para fazer este teste nós podemos usar a opção **-verifyroms**, supondo que tenhamos ambas as ROMs puckman.zip e pacman.zip na nossa pasta **roms** podemos realizar o seguinte teste para verificar se puckman está com tudo em ordem:

```
mame -verifyroms puckman
romset puckman is good
1 romsets found, 1 were OK.
```

Podemos fazer o mesmo teste com pacman:

```
mame -verifyroms pacman
romset pacman [puckman] is good
1 romsets found, 1 were OK.
```

Repare que como a ROM **pacman** é um clone de **puckman**, o MAME destaca essa informação dentro de colchetes [], no caso [puckman], isso significa que temos ambas as ROMs.

Vamos supor que a gente não tenha a ROM **puckman**, este é o erro que o MAME apresentará ao realizar o mesmo teste com a ROM **pacman**:

```
mame -verifyroms pacman
pacman      : 82s123.7f (32 bytes) - NOT FOUND (puckman)
pacman      : 82s126.4a (256 bytes) - NOT FOUND (puckman)
pacman      : 82s126.1m (256 bytes) - NOT FOUND (puckman)
pacman      : 82s126.3m (256 bytes) - NOT FOUND (puckman)
romset pacman [puckman] is bad
1 romsets found, 0 were OK.
```

Aqui o MAME identifica quais são os arquivos que estão faltando (82s123.7f, 82s126.4a, 82s126.1m, 82s126.3m), o respectivo tamanho de cada um dos arquivos e o mais importante, o MAME informa qual o nome da ROM que está faltando (puckman), como **puckman** está entre colchetes, agora sabemos que **puckman** está faltando e que ela é necessária para que a ROM **pacman** funcione.

Podemos usar a opção *-listclones* / *-lc* para identificar qual é a ROM principal e qual é o clone, exemplo:

```
mame -lc pacman
Name:         Clone of:
pacman        puckman
```

O MAME também consegue apontar a falta de ROMs necessárias para o funcionamento de um sistema e que não seja necessariamente relacionado com a questão de ROMs principais e clones. Podemos usar o exemplo do jogo **The King of Fighters '94** (kof94), para que ele funcione, é preciso ter o arquivo com as *BIOS* do sistema, exemplo:

```
mame -verifyroms kof94
kof94        : sfix.sfix (131072 bytes) - NOT FOUND (neogeo)
kof94        : 000-lo.lo (131072 bytes) - NOT FOUND (neogeo)
kof94        : sp-s2.sp1 (131072 bytes) - NOT FOUND (neogeo)
kof94        : sp-s.sp1 (131072 bytes) - NOT FOUND (neogeo)
...
romset kof94 [neogeo] is bad
1 romsets found, 0 were OK.
```

Aqui o MAME está informando que a ROM neogeo (ou neogeo.zip na pasta **roms**) não existe, não foi encontrada na pasta **roms** ou que o caminho para onde ela exista não foi definido no *rompath*. Para corrigir o problema, basta baixar o neogeo.zip e colocá-lo na pasta **roms** e repetir o teste:

```
mame -verifyroms kof94
romset kof94 [neogeo] is good
1 romsets found, 1 were OK.
```

Dada a versatilidade do MAME de identificar o que ele precisa para funcionar, os conjuntos das ROMs são separados em 3 categorias:

- **non-merged** (não mesclado)
- **split** (dividido)
- **merged** (mesclado)

A sua organização é bem simples de se compreender conforme mostram as imagens⁴ abaixo:

Não mesclado

Principal	Clone
XXYYZZ.01	XXYYZZ.01
XXYYZZ.02	XXYYZZ.02
XXYYZZ.03	XXYYZZ.03
XXYYZZ.04	XXYYZZ.04
XXYYZZ.05	XXYYZZ.06

Nesta categoria, muito espaço é perdido pois há a duplicidade de arquivos. No nosso exemplo acima, nós temos a duplicidade das ROMs XXYYZZ de 1 a 4 e a única diferença é a rom 5 na ROM **principal** e a ROM 6 na ROM **clone**, como explicado anteriormente, a diferença entre elas pode ser um idioma diferente, o licenciamento diferente para um país diferente do original como do Japão para os EUA ou do Japão para o Reino Unido, etc. Dada a ineficiência de armazenamento, este é um modo não recomendado para armazenar as suas ROMs.

Dividido

Principal	Clone
XXYYZZ.01	XXYYZZ.06
XXYYZZ.02	
XXYYZZ.03	
XXYYZZ.04	
XXYYZZ.05	

Aqui nós temos todos os arquivos de ROMs principais num arquivo zip e apenas o arquivo da ROM que é diferente como **clone** em relação a ROM **principal** separado. A característica deste modo é a economia de espaço já que não há duplicidade de ROMs, porém, há quem prefira o modo **mesclado**. A desvantagem deste modo é a dependência externa de arquivos do tipo BIOS (neogeo.zip por exemplo) e DISPOSITIVOS (namco51.zip por exemplo), o sistema para de funcionar numa eventual perda ou exclusão acidental destes arquivos.

Mesclado

Principal+Clone

XXYYZZ.01
XXYYZZ.02
XXYYZZ.03
XXYYZZ.04
XXYYZZ.05
XXYYZZ.06

Neste caso nós temos todas as ROMs necessárias do sistema (incluindo os clones) num único arquivo. A vantagem deste modo é economizar ainda mais espaço que no modo dividido pois todos os arquivos clones que eventualmente acabam se repetindo em sistemas diferentes agora ficam num arquivo só, assim como, as ROMs do tipo BIOS e DISPOSITIVOS.

Estes são princípios básicos destes conjuntos de ROMs, porém, existem dois outros tipos de conjunto que serão usados no MAME de tempos em tempos.

⁴ Foi usado a imagem [deste link](#) como referência.

O primeiro, é o **conjunto de BIOS** (*BIOS set*). Alguns sistemas arcade compartilhavam uma plataforma de hardware em comum, como o hardware de arcade Neo Geo. Já que na placa principal havia todos os dados necessários para iniciar e realizar o seu próprio auto-teste do hardware antes de seguir inicializando um dos cartuchos de jogos. Aliás, não é apropriado colocar os dados do jogo para iniciar junto com a BIOS. Em vez disso, ele é armazenado separadamente como uma imagem BIOS para o próprio sistema (**neogeo.zip** para jogos Neo Geo por exemplo).

O segundo, o **conjunto de dispositivos** (*device set*). Visando a economia de tempo e de dinheiro, frequentemente os fabricantes de arcade reutilizavam várias partes dos seus projetos mais de uma vez afim de economizar tempo e dinheiro. Alguns destes circuitos menores reapareceriam nas placas mais novas desde que tivessem um mínimo em comum com o circuito das placas lançadas anteriormente, logo, não seria possível organizar os dados do circuito ou da própria ROM usando o contexto de **principal** e **clone**. Por causa disso, algumas ROMs são categorizados como "*Device*" (dispositivo) onde os dados são armazenados como um conjunto de dispositivos ou "*Device set*".

Por exemplo, a Namco utilizou um circuito integrado customizado de entrada e saída (I/O) *Namco 51xx* para lidar com os comandos do joystick e as chaves DIP para o jogo **Galaga** que também é utilizado por outros jogos, assim sendo, para que o jogo funcione no MAME será preciso a ROM de dispositivos armazenado no arquivo `namco51.zip` e também do `namco54.zip` para que a ROM `galaga.zip` funcione.

4.7.4 Solucionando problemas dos seus conjuntos de ROMs e um pouco de história

A frustração de muitos usuários do MAME podem estar relacionadas com as alterações julgadas como desnecessárias por muitos, porém, tais alterações são necessárias devido as alterações que os arquivos ROM sofrem ao longo do tempo e que na cabeça dos usuários, parece que fazemos isso para tornar a vida de vocês mais difícil e realmente não é o caso.

Compreender o motivo destas alterações e o porque delas serem necessárias lhe ajudará a evitar ser iludido por estas contradições sobre as *versões das ROMs*.

Uma grande quantidade de ROMs e com os seus respectivos conjuntos já existiam antes da emulação. Estes conjuntos iniciais foram criados pelos proprietários das casas de arcades e utilizados como um recurso de manutenção para as placas quebradas que já não funcionavam mais, assim como, para a substituição dos componentes/peças/CIs danificados. Infelizmente, alguns destes conjuntos já não continham mais todos os dados essenciais (o programa em si) para poder funcionar. Muitas das imagens que foram extraídas logo no começo tinham falhas e erros, seja por um procedimento errado na hora da extração ou pela falta da tecnologia para poder fazê-lo da forma mais eficiente possível, como a falta de informação responsável pela paleta de cores da tela por exemplo.

Os primeiros emuladores tentavam simular artificialmente estes dados das cores que faltavam da maneira mais próxima possível, porém, por mais que se tentasse ela nunca chegava próxima da original, havia erros. Até descobrirem os dados que faltavam em outros circuitos integrados. Isso faz com que seja necessário extrair estes dados e atualizar os conjuntos antigos com os novos arquivos conforme fosse necessário.

Não demoraria muito para descobrir que muitos dos conjuntos já existentes tinham dados ruins para um ou mais circuitos integrados conforme a emulação daquele sistema específico foi melhorando, entenda "dados ruins" como uma imagem extraída faltando partes, no formato errado, com partes corrompidas, etc. Algumas vezes os desenvolvedores precisavam criar burlar o funcionamento original de certos circuitos para que a emulação pudesse funcionar porque a imagem deste CI específico os desenvolvedores não tinham acesso ou porque ainda não era possível extrair o seu conteúdo. E quando a imagem deste CI era extraído e usado no emulador, a emulação do sistema já não funcionava mais.

Por essas e outras que, uma vez compreendido como o circuito funcionava, o driver e as ROMs daquele respectivo sistema precisavam ser atualizados. E conforme mais ROMs iam aparecendo, mais e mais conjuntos precisariam de revisões completas.

Ocasionalmente, seria descoberto que a documentação de alguns jogos estava errada ou foi feita de forma errada. Alguns jogos considerados originais eram na verdade, cópias piratas de fabricantes desconhecidos. Outros jogos que foram considerados como "piratas" eram na verdade a versão original do jogo e assim por diante. Os dados de alguns jogos estavam bagunçados de maneira que não se sabia exatamente de qual região determinada placa era (como jogos **World** misturado com **Japão** por exemplo) o que exigiu também que ajustes internos e a correção dos nomes fossem feitos.

Mesmo agora, acontecem achados ocasionais e "milagrosos" que alteram a nossa compreensão desses jogos. Como é fundamental que uma documentação seja precisa para registrar a história dos arcades, o MAME mudará o nome dos conjuntos sempre que for necessário, visando a precisão e mantendo as coisas da maneira mais correta possível, sempre no limite do conhecimento que a equipe têm a cada novo lançamento do MAME.

Isso resulta numa compatibilidade muito irregular para os conjuntos das ROMs que param de funcionar nas versões mais antigas do MAME. Alguns jogos podem não ter mudado muito entre 20 ou 30 novas versões do MAME, assim como outros podem ter mudado drasticamente entre as novas versões já lançadas.

Caso encontre problemas com um determinado conjunto de ROMs que não funcionam mais, há várias coisas a serem verificadas:

- Você está tentando rodar um conjunto de ROMs destinado à uma versão mais antiga do MAME?
- Você tem o conjunto de BIOS necessários ou a ROM dos dispositivos?
- Seria este um clone que precisaria ter também a ROM principal?

O MAME *sempre informará quais os arquivos estão faltando*, dentro de quais conjuntos e onde eles foram procurados.

4.7.5 ROMs e CHDs

Os dados do CI que contém a imagem da ROM, tendem a ser relativamente pequenos e são carregados sem maiores problemas na memória do sistema. Alguns jogos também usavam mídias adicionais de armazenamento, como discos rígidos, CD-ROMs, DVDs e Laserdiscs. Esses meios de armazenamento são, por questões técnicas diversas, inadequados para serem armazenados da mesma forma que os dados das ROMs e, em alguns casos, não caberão por inteiro na memória.

Assim, um novo formato foi criado para eles, sendo armazenados num arquivo CHD. **Compressed Hunks of Data** ou numa tradução literal seria **Pedaços Comprimidos de Dados** ou CHD para simplificar. Estes são projetados especificamente em torno das necessidades de armazenamento dessa mídia. Para rodar, alguns jogos de arcade, de consoles e de PCs precisarão de um arquivo CHD.

Como os CHDs já estão comprimidos, eles **NÃO DEVEM** ser comprimidos novamente num arquivo ZIP, 7Z ou qualquer outro.

Visando a economia de espaço na existência de diversas variantes de um sistema ou de um programa, o MAME oferece suporte a arquivos "*delta CHD*". Um arquivo delta CHD armazena apenas as partes de dados que forem diferentes dos dados do arquivo CHD *principal*, isso possibilita uma economia de espaço considerável quando houver um grande compartilhamento de dados entre eles. Os arquivos delta CHD só podem ser usados nos clones dos sistemas principais, em dispositivos com uma ROM principal e os clones dos programas. Para que seja possível usar um delta CHD, é obrigatório que exista um CHD principal para que o MAME consiga ler os dados compartilhados, seja para um sistema, para dispositivos ROM ou para programas.

4.8 Problemas comuns e algumas perguntas frequentes

Aviso: As informações a seguir não possuem qualquer fundamento jurídico e tão pouco se destinam para tanto.

1. *Por que as minhas ROMs antigas não funcionam mais com as versões mais novas do MAME?*
2. *Por que o meu jogo mostra uma tela de erro quando eu insiro moedas rapidamente?*
3. *Por que o meu pacote MAME não oficial (vindo do EmuCR por exemplo ou de qualquer outro lugar) não funciona direito? Por que a minha atualização oficial está quebrada?*
4. *Por que o MAME suporta jogos de console e terminais burros? Não seria mais rápido se o MAME suportasse apenas jogos de arcade? Não usaria menos memória RAM? Não faria com que o MAME ficasse mais rápido por causa de A, B ou C?*
5. *Por que a minha ROM de Neo-Geo não funcionam mais? Como eu faço para que o jogo Humble Bundle volte a funcionar?*
6. *Como posso usar a coleção para a Steam do Mega Drive Classics collection do Sega Genesis com o MAME?*
7. *Por que o MAME alega que "faltam arquivos" sendo que eu tenho estas ROMs?*
8. *Como posso ter certeza que tenho as ROMs certas?*
9. *Por que alguns jogos têm a versão Americana como a principal, outras têm a Japonesa e outros a versão Mundo (World)?*
10. *Como faço para obter legalmente as ROMs ou as imagens de disco para poder rodar no MAME?*
11. *A cópia legal das ROMs não esbarram num possível limiar jurídico?*
12. *As ROMs dos jogos não podem ser consideradas abandonadas com o tempo (abandonware)?*
13. *E aqueles gabinetes de arcade vendidos no Mercado Livre, OLX e outros lugares que vêm com todas as ROMs?*
14. *E aqueles caras que gravam DVDs com ROMs e cobram apenas o preço da mídia?*
15. *Mas não há uma isenção especial do DMCA que torne a cópia de uma ROM legal?*
16. *Há algum problema se eu baixar a ROM e "experimentar" por 24 horas?*
17. *E se eu comprar um gabinete com ROMs legalizadas, posso disponibilizá-lo num local público para que eu possa ganhar dinheiro?*
18. *Mas eu já vi gabinetes do Ultracade e Global VR Classics montados em lugares públicos? Por que eles podem?*
19. *AJUDA! Eu estou tendo tela preta ou uma mensagem de erro relacionada com o DirectX no Windows!*
20. *Eu tenho um controlador que não quer funcionar com a versão nativa do MAME no Windows, o que posso fazer?*
21. *O que aconteceu com o suporte do MAME para placas de som externas com o OPL2 integrado?*
22. *O que aconteceu com a opção de auto disparo? (turbo)*
23. *O MAME é compatível com o G-Sync ou FreeSync? Como configurar o MAME para utilizá-los?*

4.8.1 Por que as minhas ROMs antigas não funcionam mais com as versões mais novas do MAME?

Não há um único motivo porém podemos citar dois, por questões de melhoramentos da emulação feita pelo MAME.

A emulação em si não é algo simples e trivial, horas e mais horas de desenvolvimento são necessárias para descobrir como certos sistemas funcionam, para descobrir como quebrar certas proteções, para descobrir informações não disponíveis, para lidar com componentes eletrônicos que foram customizados pelas empresas e não há informações disponíveis ao público, etc. Logo, certos parâmetros para fazer determinado sistema funcionar são diferentes. Quando tais parâmetros mudam, certos arquivos das ROMs deixam de ser necessárias, algumas precisam passar por uma validação e outras precisam ser extraídas novamente o que acontece na grande maioria das vezes.

As ROMs antigas foram extraídas de forma precária no passado sem passar por validações, hoje com mais conhecimento e tecnologia estas mesmas ROMs agora podem ser extraídas de forma mais confiável, por isso que as ROMs são atualizadas e as antigas deixam de ser válidas, mesmo que elas funcionassem nas versões mais antigas do MAME.

Há outros tipos de ROMs onde só é possível a sua extração através de um processo chamado de [decapagem](#) onde o miolo do circuito integrado é exposto para que só então o seu conteúdo possa ser extraído à força, por isso que algumas ROMs que funcionavam antes agora precisam de mais um ou mais arquivos.

Em conjunto com o processo de decapagem, depois que o miolo é exposto há também casos onde alguns colaboradores e desenvolvedores do MAME precisam recriar a ROM [manualmente](#) se utilizando de fotos de alta resolução e este é um processo lento e muito demorado. Para conhecer mais sobre este processo consulte a página do [CAPSOFF](#) ou do [Guru](#).

Consulte também o capítulo *Sobre as ROMs e os seus conjuntos*.

4.8.2 Por que o meu jogo mostra uma tela de erro quando eu insiro moedas rapidamente?

Isso não é um bug do MAME. No hardware de arcade original, não era possível inserir moedas tão rápido como se faz apertando um botão. A única maneira que você pode obter crédito nesse ritmo é se o hardware do mecanismo de moedas estiver com defeito ou se você estivesse fisicamente tentando enganar o mecanismo de moeda.

Em ambos os casos, o jogo apresentaria um erro para que o responsável investigasse a situação, evitando que algum espertinho tirasse vantagem em cima daquele que trabalha duro para conquistar seu dinheiro. Mantenha um ritmo lento de inserção de moedas e para que este erro não ocorra.

4.8.3 Por que o meu pacote MAME não oficial (vindo do EmuCR por exemplo ou de qualquer outro lugar) não funciona direito? Por que a minha atualização oficial está quebrada?

Em muitos casos, as alterações de vários subsistemas tais como plug-ins Lua, HLSL ou BGFX vem como atualizações para diversos arquivos diferentes assim como o código-fonte principal do MAME. Infelizmente as versões que vem de terceiros podem vir como apenas um executável principal do MAME ou com arquivos externos desatualizados, que podem quebrar a relação entre estes arquivos externos e o código fonte principal do MAME. Apesar das repetidas tentativas de entrar em contato com alguns destes terceiros para alertá-los, estes insistem em distribuir um MAME quebrado e sem as atualizações.

Como não temos qualquer controle sobre como estes terceiros distribuem essas versões, tudo o que podemos fazer para sites como EmuCR é informar que não fornecemos suporte para programas que nós não compilamos. Compile o seu próprio MAME ou use um dos pacotes oficialmente distribuídos por nós.

Este problema também ocorre caso não tenha atualizado o conteúdo das pastas HLSL e BGFX com as últimas versões oficiais do MAME.

4.8.4 Por que o MAME suporta jogos de console e terminais burros? Não seria mais rápido se o MAME suportasse apenas jogos de arcade? Não usaria menos memória RAM? Não faria com que o MAME ficasse mais rápido por causa de A, B ou C?

Este é um equívoco comum. A velocidade da emulação não é regida pelo tamanho final MAME, apenas as partes mais ativamente usadas são carregadas na memória quando for necessário.

Para o MAME os dispositivos adicionais são uma coisa boa pois nos permite realizar testes de estresse em seções dos vários núcleos de CPU e outras partes da emulação que normalmente não veem uma utilização mais pesada. Enquanto um computador e um sistema de arcade podem usar exatamente a mesma CPU, a maneira como eles usam esta CPU pode diferir drasticamente.

Nenhuma parte do MAME é descartável, independente de qual seja. O princípio que o MAME defende que é a preservação e a documentação, sejam os sistemas de vídeo poker quanto os arcades, não importa. O MAME é um software de código aberto, muitas coisas já foram abordadas da melhor maneira possível, há sempre espaço para melhorias, todas elas são sempre bem vindas caso seja um programador habilidoso

4.8.5 Por que a minha ROM de Neo-Geo não funciona mais? Como eu faço para que o jogo Humble Bundle volte a funcionar?

Recentemente a BIOS do Neo-Geo foi atualizada para adicionar uma nova versão da Universe BIOS. Isso começou entre as versões 0.171 e 0.172 do MAME que resultou num erro ao tentar carregar qualquer jogo de Neo-Geo com um conjunto **neogeo.zip** desatualizado, isso também afeta o conjunto de pacote do jogo **Humble Bundle**.

Os jogos em si estão corretos e atualizados a partir da versão 0.173 do MAME (e provavelmente continuará assim) no entanto uma atualização manual destes arquivos que estão dentro dos pacotes .ZIP terá que ser feita. O conjunto de BIOS do Neo-Geo (**neogeo.zip**) incluído no pacote do jogo **Humble Bundle** estava incompleto até a versão 0.172 do MAME.

Sugerimos que entre em contato com o fornecedor dos seus jogos (**Humble Bundle** e **DotEmu**) e peça para eles atualizarem o jogo para a versão mais recente. Se muita gente pedir de forma gentil, pode ser que eles atualizem para você.

4.8.6 Como posso usar a coleção para a Steam do Mega Drive Classics collection do Sega Genesis com o MAME?

A partir da atualização de Abril de 2016, todas as imagens ROM incluídas no conjunto são agora 100% compatíveis com o MAME e outros emuladores *Genesis/Mega Drive*. As ROMs estão guardadas na pasta `steamapps\\Sega Classics\\uncompressed ROMs` como uma série de extensões em formatos de imagem do tipo `.68K` e `.SGD`, que podem ser carregadas diretamente no MAME. Os manuais em PDF para os jogos podem também serem encontrados na pasta `steamapps\\Sega Classics\\manuals`.

4.8.7 Por que o MAME alega que "faltam arquivos" sendo que eu tenho estas ROMs?

Pode ser causado por várias razões:

- Não é incomum as ROMs de um jogo mudarem entre as novas versões do MAME. Por que isso aconteceria?

Muitas vezes é feita uma extração melhor do CI que contém a ROM ou então foi feita uma extração mais completa hoje e que não foi possível na época, ou até mesmo foi feito uma nova extração para corrigir os erros detectados nas ROMs anteriores. As primeiras versões do MAME não eram tão chatas sobre esta questão, porém as versões mais recentes são.

Além disso, podem haver mais características de um jogo emulado numa versão posterior que não havia na versão anterior, o que exige a execução de mais códigos dentro do MAME para rodar essa nova ROM.

- Alguns jogos precisam de arquivos CHD.

Um arquivo CHD é uma representação comprimida de uma imagem de um jogo em disco rígido, CD-ROM ou laserdisc, geralmente não é incluído como parte das ROMs de um jogo.

No entanto, assim como na maioria dos casos, esses arquivos são necessários para rodar o jogo, e o MAME vai reclamar se eles não puderem ser encontrados.

- Alguns jogos como **Neo-Geo**, **Playchoice-10**, **Convertible Video System**, **Deco Cassette**, **MegaTech**, **MegaPlay**, **ST-V Titan** e outros, precisam das suas ROMs e do conjunto de BIOS.

As ROMs da BIOS geralmente contêm um código da ROM que é usado para inicializar o sistema, o código faz lista dos jogos em sistema multijogos e o código comum a todos os jogos no referido sistema.

As ROMs da BIOS devem estar nomeadas corretamente e comprimida em formato .zip dentro da pasta ROMs.

- Versões mais antigas do MAME precisavam de tabelas para descriptografar os arquivos criptografados, criado na época pela equipe **CPS2Shock** para que fosse possível emular jogos da **Capcom Play System 2**, também conhecido como jogos **CPS2**.
- Alguns jogos no MAME são considerados *Clones* de outros jogos. Isto é, o jogo em questão é simplesmente uma versão alternativa do mesmo jogo.

As versões alternativas de alguns jogos incluem as versões com texto em outros idiomas, com diferentes datas de direito autoral, versões posteriores ou atualizações, versões piratas, etc.

Os jogos "clonados" muitas vezes se sobrepõem algum código da ROM do jogo, como se fosse a versão original. Para verificar se possui ou não algum tipo de jogo "clonado" digite o comando:

```
mame.exe -listclones
```

Para rodar um "jogo clonado" basta colocar a ROM pai dentro da pasta ROMs (sempre zipada).

4.8.8 Como posso ter certeza que tenho as ROMs certas?

O MAME verifica a existência das ROMs corretas antes de iniciar a emulação. Caso veja alguma mensagem de erro, as suas ROMs não são aquelas testadas e que funcionam corretamente com o MAME. É necessário obter as ROMs corretas através de meios legais.

Caso tenha vários jogos e quiser verificar se eles são compatíveis com a versão atual do MAME, será possível usar a opção `-verifyroms`.

Por Exemplo:

```
mame.exe -verifyroms robby
```

Verifica as suas ROMs para o jogo **Robby Roto** e exibe os resultados na tela.

```
mame.exe -verifyroms \* >verify.txt
```

Verifica a autenticidade de TODAS as ROMs dentro do seu diretório ROMs e grava os resultados dentro de um arquivo de texto chamado *verify.txt*.

4.8.9 Por que alguns jogos têm a versão Americana como a principal, outras têm a Japonesa e outros a versão Mundo (World)?

Embora essa regra nem sempre seja verdadeira, normalmente é a maneira na qual estes conjuntos são organizados. A prioridade normal é usar o conjunto **Mundo**, caso esteja disponível, **Americana**, se não existir nenhum outro conjunto mundial em Inglês e **japonês** ou uma outra região qualquer.

As exceções são aplicadas quando os conjuntos Americanos e Mundo têm censuras ou alterações significativas da sua versão original. Por exemplo, o jogo Gals Panic (do conjunto **galsnew**) usa a versão Americana como pai porque têm recursos adicionais se comparado com a versão de exportação mundial (do conjunto **galsnewa**). Esses são recursos opcionais censurados, como uma opção de layout de controle adicional (que não usa nenhum botão) e clipes de voz no idioma Inglês.

Uma outra exceção seria para os jogos que foram licenciados por terceiros para que fossem exportados e lançados lá fora. O Pac Man, por exemplo, foi publicado pela Midway nos EUA, embora tenha sido criado pela Namco do Japão. Como resultado, o conjunto pai é o conjunto japonês **puckman**, que mantém os direitos autorais da Namco.

Por último, um desenvolvedor que adiciona um novo conjunto, este pode optar por usar qualquer esquema de hierarquia e de nomenclatura que deseje e não fica restrito às regras acima. No entanto, a maioria segue essas diretrizes.

4.8.10 Como faço para obter legalmente as ROMs ou as imagens de disco para poder rodar no MAME?

As principais opções são:

- É possível obter uma licença para eles, comprando uma através de um distribuidor ou fornecedor que tenha a devida autoridade para fazê-lo.
- É possível baixar um dos conjuntos de ROMs que foram disponibilizados gratuitamente para o público em geral e para o uso não comercial do mesmo.
- É possível comprar uma PCB de arcade e extrair as ROMs ou discos você mesmo e usá-las com o MAME.

No mais, você está por sua própria conta e risco.

4.8.11 A cópia legal das ROMs não esbarram num possível limiar jurídico?

Não, de forma alguma. Você não tem permissão para fazer cópias de software sem a permissão do proprietário que detém estes direitos. A questão é preto no branco, mais claro que isso, impossível.

4.8.12 As ROMs dos jogos não podem ser consideradas abandonadas com o tempo (abandonware)?

Não.

Até mesmo as empresas que faliram tiveram seus ativos comprados por alguém e esse alguém hoje é o detentor legal desses direitos autorais.

4.8.13 E aqueles gabinetes de arcade vendidos no Mercado Livre, OLX e outros lugares que vêm com todas as ROMs?

Ele poderá estar cometendo um crime caso o vendedor não tenha uma licença adequada ou permissão para fazer a venda, sem falar nas devidas permissões legais e licenças para vender um gabinete junto com essas ROMs. Ele só poderá vendê-las junto com o gabinete quando ele tiver uma licença ou permissão para vender as ROMs em seu nome, vindas de um distribuidor ou fornecedor licenciado para tanto. Caso contrário, estamos falando de pirataria de software.

E para incluir uma versão do MAME nestes gabinetes que eles estão vendendo junto com as ROMs, seria necessário também assinar um contrato conosco para obter uma versão licenciada do MAME para rodar apenas as ROMs que ele adquiriu de forma legal e mais nada.

4.8.14 E aqueles caras que gravam DVDs com ROMs e cobram apenas o preço da mídia?

O que eles fazem é tão ilegal quanto vender as ROMs de forma direta ou junto com os gabinetes. Enquanto alguém possuir os direitos autorais destes jogos, fazer cópias ilegais da maneira que for e disponibilizá-las para venda é crime e ponto final. Caso alguém vá para a internet vender cópias piratas do último álbum de um artista qualquer a preço de banana cobrando apenas o custo da mídia, você acha que eles conseguiriam sair impunes dessa?

Pior ainda, muitas dessas pessoas gostam de afirmar que elas estão ajudando o projeto. Para a equipe do MAME, essas pessoas só criam mais problemas. Nós não estamos associados a essas pessoas de forma alguma, independentemente de quão "oficiais" elas se achem. Ao comprar pirataria você está incentivando os criminosos a continuar lucrando com a venda de software pirata na qual eles não possuem direito algum.

Qualquer pessoa que use o nome do MAME e/ou seu logotipo para vender esses produtos, também está violando direitos autorais e a marca registrada do MAME.

4.8.15 Mas não há uma isenção especial do DMCA que torne a cópia de uma ROM legal?

Não.

Você entendeu essas isenções de forma errada. A isenção permite que as pessoas façam a engenharia reversa para quebrar a criptografia que protege a cópia de programas de computador obsoletos.

Ela permite que se faça isso para descobrir como esses programas obsoletos funcionavam, não sendo ilegal de acordo com a DMCA. Isso nada tem haver com legalidade de violar os direitos autorais dos programas de computador alheios, que é o que você faz caso faça cópias ilegais de ROMs.

O DMCA é uma lei Americana, é um acrônimo para **Digital Millennium Copyright Act** ou numa tradução literal ficaria "*Lei dos Direitos Autorais do Milênio Digital*".

No Brasil essa lei não tem validade alguma e tão pouco existe qualquer lei equivalente no Brasil.

4.8.16 Há algum problema se eu baixar a ROM e "experimentar" por 24 horas?

Esta é uma lenda urbana criada por pessoas que distribuem ROMs para download em seus sites, tentando justificar o fato deles estarem infringindo a lei. Não existe nada disso em qualquer lei de direitos autorais nos EUA e muito menos no Brasil ou em qualquer outro lugar.

4.8.17 E se eu comprar um gabinete com ROMs legalizadas, posso disponibilizá-lo num local público para que eu possa ganhar dinheiro?

Geralmente não.

Tais ROMs são licenciadas apenas para fins pessoais e de uso não comercial a não ser que você tenha adquirido uma licença que diga o contrário e permita tal uso.

4.8.18 Mas eu já vi gabinetes do Ultracade e Global VR Classics montados em lugares públicos? Por que eles podem?

O Ultracade tinha dois produtos distintos, o sistema **Ultracade** possuía **licenças comerciais** para o uso dos jogos com finalidade comercial, já o **Arcade Legends** possuía uma licença **exclusiva** voltada **apenas** para uso em ambiente particular e residencial.

Apenas os sistemas com licença comercial foram concebidas para serem colocadas em local público e gerar renda, assim como foram e ainda são os sistemas de arcade tradicionais.

Desde sua aquisição pela empresa Global VR eles só oferecem o gabinete **Global VR Classics**, que equivale ao produto Ultracade anterior.

4.8.19 AJUDA! Eu estou tendo tela preta ou uma mensagem de erro relacionada com o DirectX no Windows!

Possivelmente os arquivos Runtimes do DirectX, estejam faltando ou estão danificados. Você pode baixar a ferramenta do DirectX mais recente direto do site da Microsoft no endereço abaixo: <https://www.microsoft.com/pt-br/download/details.aspx?displaylang=en&id=35>

Informações adicionais para a solução de problemas podem ser encontradas na página da Microsoft em: <https://support.microsoft.com/pt-br/help/179113/how-to-install-the-latest-version-of-directx>

4.8.20 Eu tenho um controlador que não quer funcionar com a versão nativa do MAME no Windows, o que posso fazer?

O MAME predefine que lerá de forma direta os dados do(s) joystick(s), do mouse e do(s) teclado(s) no Windows. Isso funciona com a maioria dos dispositivos fornecendo resultados mais estáveis. No entanto, alguns dispositivos precisam da instalação de drivers especiais que podem não funcionar ou não ser compatível com o MAME.

Tente configurar as opções *keyboardprovider*, *mouseprovider* ou *joystickprovider* (dependendo de qual tipo de dispositivo de entrada ele seja) vindo da entrada direta para uma das outras opções como o *dinput* ou *win32*.

Consulte também o capítulo *Opções relacionadas ao que é exibido na tela (OSD)* para saber mais detalhes de outros provedores compatíveis.

4.8.21 O que aconteceu com o suporte do MAME para placas de som externas com o OPL2 integrado?

O MAME em vez de emular o **OPL2**¹, inicialmente adicionou o suporte para placas de som com o CI **YM3212** da Yamaha em sua versão 0.23.

Na versão nativa do MAME nunca houve um suporte adequado para essa funcionalidade e foi completamente eliminada na versão 0.60, assim a emulação do OPL2 tornou-se avançada o suficiente para ser a melhor solução para a maioria dos casos naquela época.

Atualmente as placas de som mais recentes e modernas, não vem mais com o **YM3212** embutido, tornando-se então a única solução.

As versões não oficiais do MAME podem também ter mantido esse suporte por um período de tempo maior.

¹ OPL é um acrônimo de "FM Operator Type-L" ou numa tradução livre, *Operador de Modulação em Frequência Tipo L*, o 2 é o número do modelo. (Nota do tradutor)

4.8.22 O que aconteceu com a opção de auto disparo? (turbo)

Um plug-in Lua mais aprimorado e com mais funcionalidades foi adicionado ao MAME 0.210, a opção que antes estava embutida no código-fonte foi removida no MAME 0.216. Este novo plug-in tem mais funcionalidades do que o recurso de auto disparo incorporado na versão anterior, por exemplo, você pode configurar um botão alternativo para o auto disparo com diferentes taxas de disparo.

Para ativar e configurar o novo sistema de auto disparo, siga as instruções abaixo:

- Inicie o MAME sem nenhum sistema.
- Selecione a opção *Configuração* na parte debaixo da tela
- Selecione a opção *Plug-ins*.
- Ative a opção *Autofire plugin* (utilize as teclas esquerda / direita para alternar entre ligado e desligado).

A configuração é salva no arquivo `plugin.ini` dentro do diretório raiz do MAME no Windows ou `~/.mame` em sistemas **nix* e compatíveis.

Consulte o capítulo [Autofire Plugin](#) para obter mais informações de como usar o turbo ou acesse [PLUG-INS](#) para saber como usar os plug-ins do MAME.

4.8.23 O MAME é compatível com o G-Sync ou FreeSync? Como configurar o MAME para utilizá-los?

O MAME já é compatível com ambos G-Sync e FreeSync nos sistemas Windows e Linux, contudo não há a compatibilidade com o macOS ainda.

- Tenha certeza que seu monitor suporta pelo menos 120 Hz G-Sync/FreeSync. Caso o seu monitor seja compatível apenas com o modo G-Sync/FreeSync em 60 hz, haverá problemas com drivers como o *Pac-Man* que roda a **60.60606 Hz**, assim como outros que rodam com uma velocidade semelhante porém não os 60 Hz exatos.
- Caso esteja jogando o MAME em modo janela ou utilizando o sistema de vídeo BGFX, será necessário ter certeza que tenha o G-Sync/FreeSync ativado para o driver de vídeo que estiver utilizando nas aplicações em modo janela, assim como no modo de tela inteira.
- Tenha certeza que a opção *triple buffering* esteja desligado.
- No geral, é aconselhável ligar a opção Vsync quando usar o G-Sync e FreeSync.
- A opção *-lowlatency* não afeta o desempenho do MAME com o G-Sync ou FreeSync.

Os efeitos do G-Sync e FreeSync é mais notável nos drivers que rodam com taxas de atualização que sejam bem diferentes do que o seu PC esteja executando. Como por exemplo, as primeiras três versões do **Mortal Kombat** roda a **54.706841 Hz**.

4.9 Perguntas frequentes sobre alguns sistemas

Esta é uma tradução feita do [FAQ Games](#) encontrado no site oficial do MAME. Alguns sistemas da lista oficial foram removidos porque ou a informação não está correta com base na versão mais recente do MAME, talvez porque houve uma atualização na emulação, houveram correções nas ROMs, o sistema já não funciona mais (independente do motivo), etc. Outras dicas foram alteradas seja porque a informação não corresponde mais com o que aparece na tela ou já foram corrigidas. Alguma notas foram adicionadas visando informar a condição atual de determinados sistemas.

4.9.1 Índice dos sistemas

0-9

2 on 2 Open Ice Challenge
3 Bags Full
720 Degrees

A

American Poker 95
American Poker II (and bootlegs)
Amuse
Animal Catch
Apple 10
Arm Champs II
Assalt
Asterix
Autumn Moon
Avengers in Galactic Storm

B

Beatmania III
Beatmania III Append Core Remix
Beatmania III Append 6th Mix
Beatmania III Append 7th Mix
Beatmania III The Final
Beast Busters
Bells & Whistles
Big Deal
Big Karnak
Black Rhino
Blaster
Bosconian
Bottle 10
Bubble Memories

Bubble Trouble
Bubbles
Bucky O'Hare

C

Captain Commando
CarnEvil
Championship Bowling
Champion Super 2
Complex X
Coral Riches II
Crazy Monkey
Critter Crush
Cuore 1
Cyber Cycles
Cyber Sled

D

Dark Horse (Bootleg)
Dark Horse Legend
Daytona USA
Defender (incluindo bootlegs)
Detana!! Twin Bee
Dirt Fox
Double Dragon
Dragoon Might
DrumMania
Dream Soccer '94
Dungeons & Dragons: Shadow over Mystara

E

Elephant Family
Enchanted Forest
Escape Kids

F

Fantastic Journey
Fantasy Fortune
Final Lap 2
Final Lap 3
Final Lap R
Zenkoku Seifuku Bishoujo Grand Prix Find Love
Five a Side Soccer
Food Fight
Fruit Cocktail
Funky Head Boxers

G

Gaiapolis
Garage
G.I. Joe
Gokujyou Parodius
Golden Canaries
Série Golden Tee Fore!
Golfing Greats 2
GTI Club

H

Hae Hae Ka Ka
Hang Pilot
Hard Drivin' / Race Drivin'
Hyper Sports
Il Pagliaccio

I

Inferno
Island / Island 2
Itazura Monkey

J

Jockey Club
Jolli Witch
Jolly Card
Jolly Joker
JoJo's Bizarre Adventure e sistemas CPS-3 em geral
Joust
Joust 2

K

K.G Bird
Karate Blazers
Keks
Keyboardmania
Keyboardheaven (Korea)
Keyboardmania 2nd Mix
Keyboardmania 3rd Mix
King of Fighters (Série)
King Pin / King Pin Multi-Game
Konami 80's AC Special
Konami's Open Golf Championship
Kram
Kyukyoku Sentai Dadandarn

L

Laser Ghost
Lethal Enforcers
Lethal Enforcers 2
Line of Fire
Lotto Fun
Lovely Poker
Lucky & Wild
Lucky Haunter

M

Magic's 10
Magic Card II
Magical Zunou Power
Magnavox Odyssey 2 / Philips Videopac G7000
Mahjong Pon Chin Kan
Mahjong Yuugi
Marchen Maze
Martial Champion
Metal Slug (Series)
Metamorphic Force
Microman Battle Charge
Millennium Nuovo 4000
Monster Maulers
Mortal Kombat II / Mortal Kombat 3 / Ultimate Mortal Kombat 3
Multi Fish
Mystic Marathon
Mystic Warriors

N

Name Club Ver.3
Narc
NBA Hangtime / NBA Maximum Hangtime

O

P

ParaParaParadise
ParaParaDancing
ParaParaParadise v1.1
ParaParaParadise 1st Mix Plus
Pac-Slot
Phantom Pays
Phrase Craze
Pirate
Pirate Ship
Player's Edge Plus (Set038) Set Chip
Player's Edge Plus games (peplus.cpp)
Poker (All Sets Version 50.02 ICB) - Greyhound Electronics
Four In One Poker
Poker De Win
Poker Master (Tony Poker) Hack
Pop'n Music 4
Pop'n Music 5

Pop'n Music 6
Pop'n Music 7
Pop'n Music 8
Pop'n Music Mickey Tunes
Pop'n Music Mickey Tunes!
Pop'n Music Animelo
Pop'n Music Animelo 2
Pontoon
Pool 10
Pot Game
Premier Soccer
Print Club 2
Pro Mahjong Kiwame
Pro Mahjong Kiwame S
Prop Cycle
Pull Tabs
Pye-nage Taikai

Q

Qix
Quiz Gakumon no Susume
Quiz Sangokushi
Quiz Tonosama no Yabou

R

Raiden Fighters (Todas as Regiões)
Raiden Fighters Jet (Todas as regiões)
Racing Force
Rampage World Tour
Red Earth e sistemas CPS-3 em geral
Resident
Revolution X
Rim Rockin' Basketball
Robotron
Rock Climber (rclimb_3)
Royal Card
Royal Vegas Joker Card
Run and Gun

S

Salamander 2
San Fransisco Rush
Scud Race
Sea Bass Fishing
Selection - Greyhound Electronics
Sengoku Mahjong [BET]
Shadow Dancer
Sigma Poker 2000
Sigma Poker
Simpsons Bowling
Sinistar
Slam Dunk
Slither
Snooker 10
Soccer Superstars
Solar Assault
Space Dungeon
Space Gun
Speed Racer
Splat!
Star Trek (Defender bootleg)
Stargate
Steel Gunner / Steel Gunner 2
Steep Slope Sliders
Street Fighter III (Série) e sistemas CPS-3 em geral
Sunset Riders
Super Gran Safari
Super High Impact
Super Pool 99
Super Real Mahjong Part 2
Sweet Life

T

Taihou de Doboan
Taisen Puzzle-dama
Taisen Tokkar-dama
Target Panic
Technical Bowling
Techno Drive
Teenage Mutant Ninja (Hero) Turtles 2 - Turtles in Time
Ten Balls
Terminator 2
The Couples
The Electric Yo-Yo

The Simpsons
Thrill Drive (JAE/JAA)
Thunder Cross II
Time Crisis
Tobikose! Jumpman
Tokimeki Memorial Taisen Puzzle-dama
Top Gear
Tortuga Family
Track & Field
Triple Hunt
Turkey Shoot
Twin Bee Yahhoo!

U

V

Vendetta
Videomat
Violent Storm
Virtual Mahjong
Virtual Mahjong 2 - My Fair Lady
Virtua Racing

W

Warzard e sistemas CPS-3 em geral
Water Match
Wild West C.O.W.-Boys of Moo Mesa
Witch
World Tennis
WWF Wrestlemania

X

X-Men

Xexex

Y

Z

Zero

Zero 2

Zero Team 2000 (zerotm2k)

Zombie Raid

Zoo Keeper

2 on 2 Open Ice Challenge

- **O som deste sistema é MUITO BAIXO!**

Inicialmente quando o jogo estiver rodando pressione F2 para chamar a tela de serviço e pressione F2 novamente para evitar ficar travado nesta. Rode o jogo novamente pressione F2, vá até a opção **Volume Adjust**, aumente o volume e pressione o botão 1 do **jogador 1**. É importante que você não ultrapasse 80-90% do volume pois podem ocorrer distorções e saturação do áudio. Ao concluir o ajuste selecione o menu **Exit to Game Over**.

3 Bags Full

- **Metering Error - Call Attendant**

Pressione F1 (**Jackpot Key**) e F2 (**Audit Key**), ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas Z + X + C. Se realizado corretamente o valor de **RF/AMT** será alterado de 0 para **200**, pressione F1, depois F2.

720 Degrees

- **Este jogo usa um controle rotativo?**

O controle original foi um joystick com restrições de movimento via hardware (Spinner Control) fazendo com ele se movesse apenas ao redor de um círculo¹ pré determinado. O mecanismo interno que realizava a leitura dos comandos era similar aos controles rotativos porém ele tinha um benefício adicional em fornecer em qual direção o controle estava apontando. Devido a este fator ser um importante aspecto do jogo, a atual emulação do controle é mapeado para um joystick analógico em vez de um controle rotativo.

American Poker 95

- **Init Machine**

Ligue o **Operator Key** pressionando 9. Pressione **Door Key** pressionando a letra 0, após alguns segundos a tela de contabilidade vai aparecer (Operator Mode), pressione 9 para iniciar o jogo ou F3 caso nada aconteça.

American Poker II (and bootlegs)

- **Init Machine**

Ligue o **Operator Key** pressionando 9. Pressione **Door Key** pressionando a letra 0, após alguns segundos a tela de contabilidade vai aparecer (Operator Mode), pressione 9 para iniciar o jogo ou F3 caso nada aconteça.

¹ https://www.arcade-museum.com/game_detail.php?game_id=6791

Amuse

- **ERROR**

Ao ver a palavra **ERROR** pressione F2 + F3 para iniciar o **SELF TEST**, quando terminar pressione F2.

Animal Catch

- **BACK UP RAM NG**

Mantenha F2 pressionado seguido de F3 para reiniciar o sistema e entrar na tela de serviço. Use F2 para selecionar a opção **3. Set Mode** seguido da tecla Ctrl. Use F2 para ir passando pelas opções até chegar em **RAM CLEAR**, mantenha Ctrl pressionado até aparecer **OK** na tela. Reinicie o sistema pressionando F3.

Apple 10

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Arm Champs II

- **WARNING**

Durante o **MEMORY TEST** vai aparecer a mensagem **MOTOR TEST**, quando essa mensagem aparecer mova o controle para a esquerda e direita por um segundo ou dois, logo depois o título deve aparecer.

Assalt

- **35 WARNING 00180040**

Pressione 1.

Asterix

- **EEPROM 17B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Autumn Moon

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Se realizado corretamente o valor de **RF/AMT** será alterado de **0** para **200**, pressione **F1**, depois **F2**.

Avengers in Galactic Storm

Pressione **F2** para entrar na tela de serviço, mova o controle uma vez para cima selecionando a opção **Game Mode**, pressione o botão **Ctrl** do lado esquerdo (Botão 1 do jogo) para retornar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Beatmania III

- **ERROR E160 BACKUPRAM ERROR**
- **ERROR E165 RTC ERROR**
- **SYSTEM RECOVERY MODE**

Pressione **F2**, acesse o [Firebeat RTC Recovery Password](#) e gere a sua senha. No MAME use a tecla **0** (zero) e **-** (sinal de menos) para alterar as letras, **9** para apagar e **F2** para confirmar.

Ao terminar de inserir a senha, pressione **F2** duas vezes, assista este procedimento no Youtube:

<https://www.youtube.com/watch?v=XXp9u7RKNlQ>

Beatmania III Append Core Remix

- **ERROR E150 BACKUPRAM ERROR**
- **ERROR E166 RTC ERROR**
- **SYSTEM RECOVERY MODE**

Consulte [Beatmania III](#).

Beatmania III Append 6th Mix

- **ERROR E150 BACKUPRAM ERROR**
- **ERROR E166 RTC ERROR**
- **SYSTEM RECOVERY MODE**

Consulte [Beatmania III](#).

Beatmania III Append 7th Mix

- **ERROR E150 BACKUPRAM ERROR**
- **ERROR E166 RTC ERROR**
- **SYSTEM RECOVERY MODE**

Consulte *Beatmania III*.

Beatmania III The Final

- **ERROR E150 BACKUPRAM ERROR**
- **ERROR E166 RTC ERROR**
- **SYSTEM RECOVERY MODE**

Consulte *Beatmania III*.

Beast Busters

- **Tela Preta**

É necessário calibrar TODAS as 3 pistolas para que este sistema funcione corretamente! Ao tentar calibrar apenas uma como manda o manual a EPROM os dados da pistola estarão inválidos e uma tela preta deve aparecer depois que o sistema for reiniciado. Entre no modo de serviço e atire nos pontos indicados com as 3 pistolas, só então uma EPROM correta será gerada.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Nota: Este problema já foi resolvido nas versões mais recentes do MAME. [MT07333](#)

Bells & Whistles

- **EEPROM 17A BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Big Deal

Nada de especial, apenas tenha paciência. A inicialização do sistema leva cerca de 40 segundos.

Big Karnak

Nada de especial, apenas tenha paciência. A inicialização do sistema leva cerca de 20 segundos.

Black Rhino

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Se realizado corretamente o valor de **RF/AMT** será alterado de 0 para **200**, pressione **F1**, depois **F2**.

Blaster

Seja paciente durante o processo de inicialização. Os testes internos deste sistema demoram e na tela aparece apenas uma tela cheia de estática.

Pressione **F2** ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Bosconian

Seja paciente durante o processo de inicialização. Os testes internos deste sistema demoram e na tela aparece apenas uma tela cheia de estática.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Bottle 10

Mantenha pressionada as teclas **9 + 0**, toque em **F3**. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Bubble Memories

- **BACKUP DATA FAILED**

Ao rodar o sistema pela primeira vez aparecerá a mensagem **...BACKUP DATA FAILED**. Pressione F2 para chamar a tela de serviço e selecione **Exit** e pressione 1 para reiniciar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Bubble Trouble

Mantenha pressionada as teclas de serviço 1 e 9, em seguida pressione F2 para manter a tela de ajustes aberta. Atire no alvo usando a mira, pressione as teclas de serviço 1 e 9 novamente para mudar o alvo. Quando terminar e retornar para a tela inicial, pressione F2.

Bubbles

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Bucky O'Hare

Ao rodar o sistema pela primeira vez o **EEPROM N2** aparece como **BAD** durante a verificação de **RAM/ROM**. Mantenha F2 pressionado e toque em F3 para reiniciar o sistema criando uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Captain Commando

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar esse sistema para 4 jogadores?**

A quantidade predefinida de jogadores para esta e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as chaves DIP ou entrando na tela de serviço com F2.

CarnEvil

- **Calibrando a Pistola**

Use F2 para entrar na tela de serviço, use as teclas - e = para navegar até a opção **Gun Calibration**, use F2 para selecionar. Siga as instruções da tela mirando e atirando nos pontos indicados, faça o mesmo com cada pistola. Ao completar navegue até a opção **Exit** e pressione F2 para concluir.

Championship Bowling

- **ERROR**

Ao rodar o sistema pela primeira vez a palavra **ERROR** aparece pois a memória NVRAM necessita de inicialização. Apenas pressione 1 (P1 Start).

Champion Super 2

- **INIZIALIZAZIONE PARZIALE!**

Inicialização parcial da NVRAM, pressione e mantenha pressionado as teclas 0 + 9 e toque em F3 para que a NVRAM seja reinicializada por completo.

Complex X

Pressione 9 para passar por todas as telas de configuração.

Coral Riches II

- **Memory Error - Watchdog Reset**

Aviso: Até a presente versão do MAME este sistema ainda não funciona.
--

Crazy Monkey

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Critter Crush

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Cuore 1

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Cyber Cycles

Este sistema precisa ser previamente calibrado para poder esterçar corretamente. Mantenha a tecla de serviço 9 pressionada e pressione F2 até ver a frase **INITIALIZING** ou **INITIALIZED** na tela, em seguida pressione F2 e encerre o sistema pressionando Esc para ter absoluta certeza que a calibração tenha sido gravada corretamente.

Cyber Sled

Pressione Tab, defina a chave **DSW2** como **Ligada** e reinicie, pressione 0 para chamar a tela de serviço, segure a tecla 9 e dê um toque em F3. Uma tela com a frase **Volume Adjust** deverá aparecer. Desligue a chave **DSW2**, pressione 0 e encerre o sistema pressionando Esc. Agora os controles devem estar alinhados automaticamente permitindo a movimentação correta no jogo e nos menus de serviço.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Dark Horse (Bootleg)

O sistema está configurado para funcionar apenas com 1 crédito. Pressione a tecla =, isso vai acender a opção **CONFIG** na parte de baixo da tela, toque em F3 para chamar a tela de configuração. Vá em **Configuration Setup** selecionando com a tecla Ctrl. Utilize 1 até chegar na opção **COIN SENSOR**, utilize Ctrl para mudar para **SINGLE**. Pressione = até aparecer a mensagem **NEED SYSTEM RESTART! POWER OFF THEN TURN ON AGAIN**. Segure Shift e pressione F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Dark Horse Legend

Esse sistema precisa passar por um processo especial de inicialização quando esse sistema for iniciado pela primeira vez. Deixe que todo o procedimento seja concluído antes de encerrar o MAME, caso contrário será necessário fazer tudo novamente!

Daytona USA

Esse sistema já vem predefinido como **MASTER** numa configuração de cabine dupla (TWIN). É necessário a configuração seja alterada para **SINGLE**.

Ao ver a mensagem **Network Checking** pressione F2, use X para mover o cursor até **Game System** e entre pressionando 1, vá até **LINK ID** e altere a opção para **SINGLE** pressionando 1. Mova o cursor até **EXIT** pressionando 1 para sair e repita até o menu principal. A configuração será salva e o sistema iniciará sem problemas.

As versões mais recentes do MAME iniciam sem problemas, ainda que elas sejam configuradas como **MASTER** ou **SLAVE**.

Defender (incluindo bootlegs)

Ao rodar o sistema pela primeira vez irá aparecer uma mensagem **01 0000 COINS LEFT** ou ficará parada numa tela com os dizeres **Factory settings restored**. Pressione F2 seguido de F3. Caso falhe, tente F3 ou F1 + F2.

Detana!! Twin Bee

- **EEPROM 17A BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Dirt Fox

Mantenha a tecla 9 pressionada e então pressione F2 para entrar na tela de serviço. Verifique se os valores dos controles estão zerados, pressione F2 para sair.

Double Dragon

- **Algumas vezes o jogo parece estar lento, será que eu tenho CPU suficiente?**

Apesar do contador de FPS exibir 100% ainda assim pode parecer que o sistema está rodando mais lento do que deveria ainda que o seu computador (ou seja lá onde o MAME esteja rodando) tenha muito poder computacional. Essa lentidão também ocorre no hardware original do sistema, logo tal lentidão apenas reflete o quão preciso é a sua emulação.

Pode ser que ao utilizar trapaças, essa lentidão possa ser eliminada ou atenuada de alguma maneira, trapaças como **CPU Overclocking**, **Disable projectile impact slowdown**² ou algo neste sentido, fazendo com que a emulação do processador rode mais rápido podendo ou não eliminar o problema ou eliminando lentidões impostas pelos programadores do sistema. Note porém que independente da solução que escolha usar, isso fará com que erros ocorram durante a emulação, logo faça por sua conta e risco! Depois não saia por aí perturbando as pessoas caso algum problema ocorra por causa da alteração que você fizer.

Dragoon Might

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

DrumMania

- **Please Call Attendant**

Veja a solução em detalhes [neste capítulo](#).

Dream Soccer '94

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar este sistema para 4 jogadores?**

A quantidade predefinida de jogadores para esta e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as chaves DIP ou entrando na tela de serviço com F2.

² Trapaça encontrada na série Street Fighter II por exemplo.

Dungeons & Dragons: Shadow over Mystara

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar esta sistema para 4 jogadores?**

A quantidade predefinida de jogadores para esta e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as chaves DIP ou entrando na tela de serviço com F2.

Elephant Family

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Enchanted Forest

- **Metering Error**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** fique pressionando A até voltar para a tela principal.

Escape Kids

- **EEPROM 13B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Fantastic Journey

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Fantasy Fortune

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Se realizado corretamente o valor de **RF/AMT** será alterado de 0 para **200**, pressione **F1**, depois **F2**.

Final Lap 2

Mantenha a tecla **9** pressionada e então pressione **F2** para entrar na tela de serviço. Verifique se os valores dos controles estão zerados, pressione **F2** para sair.

Final Lap 3

Mantenha a tecla **9** pressionada e então pressione **F2** para entrar na tela de serviço. Verifique se os valores dos controles estão zerados, pressione **F2** para sair.

Final Lap R

Mantenha a tecla **9** pressionada e pressione **F2**, caso não apareça uma mensagem dizendo **Initializing** seguido de uma tela de teste de entrada, repita o procedimento até conseguir. Encerre o MAME e reinicie o sistema.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Zenkoku Seifuku Bishoujo Grand Prix Find Love

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione **9 (Service)** depois **F2 (Test)** para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com **9** e pressione **F2** para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Five a Side Soccer

Para reinicializar a EEPROM, mantenha F2 pressionado e toque em F3 uma vez. Não solte F2 até que apareça a mensagem dizendo **Initializing EEPROM**.

Food Fight

- **NVRAM FAILED**

Ao rodar o sistema pela primeira vez aparecerá a mensagem **NVRAM FAILED:....** Pressione 1 para continuar. A memória NVRAM será inicializada automaticamente. Encerre pressionando F2.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Fruit Cocktail

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Funky Head Boxers

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para 1P (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Gaiapolis

- **EEPROM 28B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Garage

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

G.I. Joe

- **EEPROM 7D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Gokujyou Parodius

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Golden Canaries

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Se realizado corretamente o valor de **RF/AMT** será alterado de **0** para **200**, pressione **F1**, depois **F2**.

Série Golden Tee Fore!

- **Arrumando volume BAIXO ou NENHUM som**

Pressione **F2** para entrar na tela de serviço, selecione **Volume Adjustment** e aumente o nível de som conforme o seu gosto, pressione **F2** para sair da tela de serviço. Desde que o MAME seja encerrado da forma correta (pressionando **Esc**) as configurações de volume de som serão mantidas na memória **NVRAM** e será utilizada posteriormente.

Golfing Greats 2

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla **F2** e toque na tecla **F3** para reiniciar o sistema e criar uma nova **EEPROM**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Nota: Este jogo não funciona e talvez a informação esteja errada.

GTI Club

Para reinicializar a **EEPROM**, mantenha **F2** pressionado e toque em **F3** uma vez. Não solte **F2** até que apareça a mensagem dizendo **Initializing EEPROM**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Hae Hae Ka Ka

- **BACK UP RAM NG**

Mantenha F2 pressionado seguido de F3 para reiniciar o sistema e entrar na tela de serviço. Use F2 para selecionar a opção **3. Set Mode** seguido da tecla 9. Use F2 para ir passando pelas opções até chegar em **RAM CLEAR**, mantenha 9 pressionado até aparecer **OK** na tela. Reinicie o sistema com F3.

Hang Pilot

Para reinicializar a EEPROM, mantenha F2 pressionado e toque em F3 uma vez. Não solte F2 até que apareça a mensagem dizendo **Initializing EEPROM**. Reinicie com Shift + F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Hard Drivin' / Race Drivin'

Válido para os sistemas: `harddrivc`, `harddrivcg`, `harddrivcb`, `harddrivc1`

- **Algumas vezes o jogo parece estar lento, será que eu tenho CPU suficiente?**

Apesar do contador de FPS exibir 100% ainda assim pode parecer que o sistema está rodando mais lento do que deveria ainda que o seu computador (ou seja lá onde o MAME esteja rodando) tenha muito poder computacional. Essa lentidão também ocorre no hardware original do sistema, logo tal lentidão apenas reflete o quão preciso é a sua emulação.

Pode ser que ao utilizar trapaças, essa lentidão possa ser eliminada ou atenuada de alguma maneira, trapaças como **CPU Overclocking**, **Disable projectile impact slowdown**³ ou algo neste sentido, fazendo com que a emulação do processador rode mais rápido podendo ou não eliminar o problema ou eliminando lentidões impostas pelos programadores do sistema. Note porém que independente da solução que escolha usar, isso fará com que erros ocorram durante a emulação, logo faça por sua conta e risco! Depois não saia por aí perturbando as pessoas caso algum problema ocorra por causa da alteração que você fizer.

Caso tenha problemas de controlar o veículo ou caso o freio esteja sempre acionado, é necessário calibrar os controles!

Primeiramente vá até o diretório **NVRAM** e apague qualquer diretório com nome **hard-driv*** ou **racedriv***. O mapeamento dos controles no MAME não são muito intuitivos logo, cuidado para não se perder.

- **Steering wheel** (Volante): Player 1 Paddle Control, é predefinido para o mouse e as teclas esquerda/direita.
- **Gas pedal** (Pedal do acelerador): Pedal 1 Control, é predefinido para a tecla Ctrl esquerda. Ao pressioná-la aciona imediatamente o controle analógico e ao soltar ocorre o inverso.
- **Brake pedal** (Pedal de freio): Pedal 2 Control, é predefinido para a tecla Alt esquerda. Funciona exatamente como o **Gas Pedal**.
- **Clutch pedal** (Pedal da embreagem): Pedal 3 Control, é predefinido para a tecla Espaço. Funciona exatamente como os pedais anteriores.

³ Trapaça encontrada na série Street Fighter II por exemplo.

- **Analog Shifter** (Câmbio de marchas mecânico/analógico): Este tipo de câmbio é encontrado em algumas versões de gabinete não compactos. É mapeado para o joystick analógico do Player 2 no eixo X/Y que é predefinido nas teclas R, F, D, G.
- **Digital Shifter** (Câmbio de marchas digital): Este tipo de câmbio é encontrada nas versões compactas do gabinete. É mapeado para os botões 2, 3, 4 e 5 do Player 1. A primeira marcha é o botão 2 (padrão: Alt esquerdo), a segunda marcha é o botão 3 (padrão: Espaço), a terceira marcha é o botão 4 (padrão: Shift esquerdo), a quarta marcha é o botão 5 (padrão: tecla Z.).
- **Key** (Chave da ignição): É mapeada para o botão Start do Player 1 (padrão: tecla 1).
- **Aborto** (Abortar): É mapeada para o botão Start do Player 2 (padrão: tecla 2).

Segue abaixo o processo completo de ajustes. Note que ainda que seja usado um controle analógico conectado em qualquer um dos controles analógicos do sistema, será muito mais fácil realizar toda essa operação usando um teclado.

1. Não toque em nenhum dos controles e vire a ignição. Tenha absoluta certeza de não tocar em nenhum controle até aqui, pressione 1.
2. Vire o controle do volante em sentido anti-horário e vire a ignição pressionado 1.
3. Vire o controle do volante em sentido horário e vire a ignição pressionado 1.
4. Engate a primeira marcha, vire a ignição pressionando 1. Ainda com a primeira engatada como mostra o canto superior esquerdo, mantenha pressionado as teclas D e R por 5 segundos, ainda com as teclas pressionadas, pressione 1.
5. Engate a quarta marcha, vire a ignição pressionando 1. Como a quarta marcha está no canto inferior direito, mantenha pressionado as teclas F e G por 10 segundos, ainda com as teclas pressionadas, pressione 1.
6. Mova o assento todo para frente e pressione 1. O feedback do acento ainda não é emulado pelo MAME então pressione 1.
7. Mova o acento todo para trás e pressione 1.
8. Tire o pé do freio e pressione 1 para abortar. Agora essa é a parte complicada, ao ver esta tela, não faça nada. Aguarde até que apareça **New max 4064** e não pressione 1 antes do tempo ou será necessário refazer tudo novamente.
9. Pise firmemente uma vez no freio e solte, faça este procedimento pressionando e segurando por 3 segundos a tecla Alt Esquerdo, depois solte. Se tudo der certo, deverá aparecer **Range 4096** e em seguida o jogo deve continuar.

Não foi tão difícil assim, foi?

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Hyper Sports

Para ter a tabela de **High Score** devidamente inicializada, pressione Tab, vá para *Chaves DIP*, defina **World Records** como **Erase on Reset**, pressione F3. Redefina **World Records** como **Don't Erase**.

Il Pagliaccio

A mensagem **RICHIESTA INIZIALIZZAZIONE** aparece ao rodar o sistema pela primeira vez, mantenha pressionado 9 e toque em F3 para entrar na tela de serviço, pressione 1 para sair.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Inferno

Ao ver a mensagem **FACTORY SETTINGS RESTORED**, pressione F2.

Island / Island 2

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Itazura Monkey

- **BACK UP RAM NG**

Mantenha F2 pressionado seguido de F3 para reiniciar o sistema e entrar na tela de serviço. Use F2 para selecionar a opção **3. Set Mode** seguido da tecla Ctrl. Use F2 para ir passando pelas opções até chegar em **RAM CLEAR**, mantenha Ctrl pressionado até aparecer **OK** na tela. Reinicie o sistema com F3.

Nota: Este sistema responde muito mal aos controles, tenha um pouco de paciência.

Jockey Club

- **BACKUP MEMORY IS WRONG**

Pressione 9 até a luz **RESET** no canto inferior da tela acender, toque em F3 para reiniciar, depois toque em 9 novamente para apagar a luz **RESET**, o sistema começa o processo de inicialização.

Jolli Witch

Quando aparecer a mensagem **ACHTUNG FEHLER IM BLOCK...** pressione Z. para iniciar.

Jolly Card

Mantenha pressionado as teclas 0 e 9 e toque em F3, pressione **SHIFT+F3** para reiniciar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Jolly Joker

Mantenha pressionado as teclas 0 e 9 e toque em F3, pressione **SHIFT+F3** para reiniciar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

JoJo's Bizarre Adventure e sistemas CPS-3 em geral

O processo de inicialização da memória NVRAM destes sistemas é demorado, elas precisam criar uma memória NVRAM válida e completa. Uma maneira de acelerar um pouco este processo é utilizar a opção *-nothrottle* porém ainda assim todo o processo pode levar cerca de meia hora ou mais.

Joust

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Joust 2

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

K.G Bird

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas Z + X + C. Se realizado corretamente o valor de **RF/AMT** será alterado de 0 para **200**, pressione F1, depois F2.

Karate Blazers

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar este sistema para 4 jogadores?**

A quantidade predefinida de jogadores para esta e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as chaves DIP ou entrando na tela de serviço com F2.

Keks

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Keyboardmania

- **REAL TIME CLOCK BATTERY NG.**
- **BACKUP DATA ERROR**
- **E161 BACKUP BATTERY ERROR**

Pressione e segure a tecla F2 até entrar no modo de teste, mova o cursor para baixo com a tecla 2 e para cima com a tecla 1, selecione pressionando ambas as teclas 1 e 2. Vá até a opção **SOUND TEST**, selecione **SOUND FLASH INIT**, pressione a tecla 1 para iniciar e aguarde a conclusão, depois selecione **EXIT** e retorne para o menu principal. Agora selecione **BACKUP CLEAR** e confirme com 1.

Segure SHIFT e pressione F3 para concluir.

Keyboardheaven (Korea)

- **REAL TIME CLOCK BATTERY NG.**
- **BACKUP DATA ERROR**
- **SOUND FLASH MEMORY NG.**
- **E761 PCM DATA ERROR**

Consulte [Keyboardmania](#).

Keyboardmania 2nd Mix

- **E761 PCM DATA ERROR**

Pressione F2, mova o cursor para baixo com a tecla 2 e para cima com a tecla 1, selecione pressionando ambas as teclas 1 e 2. Vá até **SOUND CHECK**, selecione **INITIALIZE SOUND FLASH**, selecione **YES**, confirme e pressione 1 + 2, aguarde até que a tela retorne para o menu anterior, selecione **EXIT**.

Segure SHIFT e pressione F3 para concluir.

Keyboardmania 3rd Mix

- **E161 BACKUP BATTERY ERROR**
- **E150 BACKUP RAM ERROR**

Pressione F2, mova o cursor para baixo com a tecla 2 e para cima com a tecla 1, selecione pressionando ambas as teclas 1 e 2. Vá até **SOUND OPTIONS**, selecione **INITIALIZE SOUND FLASH**, selecione **YES**, confirme e pressione 1 + 2, aguarde até que a tela retorne para o menu anterior, selecione **EXIT**. Retorne ao menu principal, selecione **ALL FACTORY SETTINGS**, selecione **YES**, confirme e pressione 1 + 2.

Segure SHIFT e pressione F3 para concluir.

King of Fighters (Série)

- **Como ativar o sangue?**

Pressione F2, navegue até **SETTING UP THE SOFT DIP** e pressione o botão 1 do controle do **jogador 1** para entrar, navegue até **SLOT 1 (nome do jogo)**, novamente clique no botão 1, movimento o controle para baixo até ver **BLOOD COLOR** e altere para **ON**. Para sair dessa tela clique no botão 3 duas vezes, selecione **EXIT** e clique no botão 1.

King Pin / King Pin Multi-Game

Ao rodar o sistema pela primeira vez aparecerá a mensagem **PLEASE SETUP GAME** na tela. Pressione Tab, vá até *Chaves DIP*, defina como **ON** as seguintes chaves :1, 4, 5 e 7. Usando a configuração predefinida dos controles deste sistema, siga a seguinte sequência Espaço, Shift Esquerdo 4x para Kingpin, 7x para **Multi-Game**, para finalizar pressione a tecla 1. A mensagem **SET ALL SWITCHES OFF** deve aparecer na tela, retorne todas as chaves DIP para as suas posições originais e reinicie o MAME.

Konami 80's AC Special

Ao rodar o sistema pela primeira vez se iniciam vários processo de inicialização e irá parar numa tela de configuração do relógio. Navegue para baixo com o botão 2 do jogador 1, selecione **SAVE AND EXIT**, pressione 1 para continuar.

Konami's Open Golf Championship

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Kram

Pressione a tecla 9 para passar por todas as telas de configuração até chegar na tela do jogo.

Kyukyoku Sentai Dadandarn

- **EEPROM 28B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Laser Ghost

- **GUN ADJUSTMENT ERROR**

Clique com qualquer botão do mouse para iniciar a calibragem da pistola do jogador 1 (lado esquerdo), caso a mira não se mova inicie novamente o sistema com a opção `-mouse` ou `mouse 1` no seu `mame.ini`. Pressione A, atire nos círculos piscantes para calibrar a pistola do jogador 1, quando terminar pressione F2 para sair da tela de ajustes.

Lethal Enforcers

Ao rodar o sistema pela primeira vez deve aparecer alguma mensagem de erro na tela. Mantenha pressionado o F2 e toque no F3, mantenha F2 pressionado até aparecer a mensagem **EEPROM INITIALIZE COMPLETE**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Lethal Enforcers 2

Ao rodar o sistema pela primeira vez deve aparecer alguma mensagem de erro na tela. Mantenha pressionado o F2 e toque no F3, mantenha F2 pressionado até aparecer a mensagem **EEPROM INITIALIZE COMPLETE**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Line of Fire

- **GUN ADJUSTMENT ERROR**

Clique com qualquer botão do mouse para iniciar a calibragem da pistola do jogador 1 (lado esquerdo), caso a mira não se mova inicie novamente o sistema com a opção `-mouse` ou `mouse 1` no seu `mame.ini`. Pressione Ctrl Esquerdo para calibrar a pistola do jogador 1, atire nos círculos piscantes, pressione A para calibrar a pistola do jogador 2. Quando terminar pressione Alt Esquerdo + S para encerrar os ajustes.

Lotto Fun

- **MEMORY PROTECT SWITCH MUST BE DOWN!**

Ao ver a mensagem **MEMORY PROTECT SWITCH MUST BE DOWN!** pressione a tecla 9 para continuar.

Lovely Poker

- **BATTERY BACKUP CHECK - ERROR FOUND**

Ao ver a mensagem **BATTERY BACKUP CHECK - ERROR FOUND** pressione a tecla - para continuar.

Lucky & Wild

Pressione F2 seguido de 9 2x, atire nos alvos, pressione F2 quando terminar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Lucky Hunter

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Magic's 10

Ao iniciar o sistema pela primeira vez pressione Tab, vá em *Chaves DIP*, ligue a chave **Disable Free Play**, coloque 1 crédito e use o botão **Collect** para receber seu primeiro **Game Over**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Magic Card II

Ao rodar o sistema pela primeira vez aparece a mensagem **Elektronik Defekt**, pressione F3 para reiniciar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Magical Zunou Power

- **ONSEI BOARD ERROR**

Aviso: Atualmente o sistema não funciona e nada pode ser feito até que o drive seja atualizado ou a emulação seja corrigida.

Magnavox Odyssey 2 / Philips Videopac G7000

Para utilizar o sintetizador de voz, execute o comando abaixo:

```
mame odyssey2 -cart1 voice -cart2 caminho_completo_para_a_ROM\nome_da_rom.zip  
mame videopac -cart1 voice -cart2 caminho_completo_para_a_ROM\nome_da_rom.zip
```

Mahjong Pon Chin Kan

- **BACK UP ERR**

Ao ver a mensagem **BACK UP ERR** pressione F2 seguido de F3, quando aparecer a tela de serviço pressione F2 e depois F3 novamente.

Mahjong Yuugi

- **BACK UP ERR**

Ao ver a mensagem **BACK UP ERR** pressione F2 seguido de F3, quando aparecer a tela de serviço pressione F2 e depois F3 novamente.

Marchen Maze

- **TEST PROGRAM INIT...**

Ao ver a mensagem **TEST PROGRAM INIT...** pressione F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Martial Champion

- **VERSION ERROR**

Ao ver a mensagem **VERSION ERROR** mantenha pressionado F2 e toque em F3, solte as teclas depois que o sistema reinicie.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Metal Slug (Series)

- **Algumas vezes o jogo parece estar lento, será que eu tenho CPU suficiente?**

Apesar do contador de FPS exibir 100% ainda assim pode parecer que o sistema está rodando mais lento do que deveria ainda que o seu computador (ou seja lá onde o MAME esteja rodando) tenha muito poder computacional. Essa lentidão também ocorre no hardware original do sistema, logo tal lentidão apenas reflete o quão preciso é a sua emulação.

Pode ser que ao utilizar trapaças, essa lentidão possa ser eliminada ou atenuada de alguma maneira, trapaças como **CPU Overclocking**, **Disable projectile impact slowdown**⁴ ou algo neste sentido, fazendo com que a emulação do processador rode mais rápido podendo ou não eliminar o problema ou eliminando lentidões impostas pelos programadores do sistema. Note porém que independente da solução que escolha usar, isso fará com que erros ocorram durante a emulação, logo faça por sua conta e risco! Depois não saia por aí perturbando as pessoas caso algum problema ocorra por causa da alteração que você fizer.

Como ativar o sangue?

Pressione F2, navegue até **SETTING UP THE SOFT DIP** e pressione o botão 1 do controle do **jogador 1** para entrar, navegue até **SLOT 1 (nome do jogo)**, novamente clique no botão 1, movimente o controle para baixo até ver **BLOOD COLOR** e altere para **ON**. Para sair dessa tela clique no botão 3 duas vezes, selecione **EXIT** e clique no botão 1.

Metamorphic Force

- **VERSION ERROR**

Ao ver a mensagem **VERSION ERROR** mantenha pressionado F2 e toque em F3, solte as teclas depois que o sistema reiniciar.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

⁴ Trapaça encontrada na série Street Fighter II por exemplo.

Microman Battle Charge

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Millennium Nuovo 4000

Ao ver a mensagem **SCHEDA MANOMESSA DA AGENTE ESTERNO** pressione F2, pressione a tecla B para encerrar.

Monster Maulers

- **EEPROM 28B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Mortal Kombat II / Mortal Kombat 3 / Ultimate Mortal Kombat 3

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **O som deste sistema é MUITO BAIXO!**

Pressione F2, vá até a opção **Volume Adjust**, selecione com o botão 1 do **jogador 1**, aumente o volume e pressione o botão 1 do **jogador 1** para concluir. Não ultrapasse 80-90% do volume pois podem ocorrer distorções e saturação no áudio. Para finalizar selecione **Exit Test Menu**.

Quando aparecer **DIAGNOSTICS SWITCH IS CLOSED**, pressione F2.

Multi Fish

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Mystic Marathon

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Mystic Warriors

- **EEPROM 15B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Name Club Ver.3

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para 1P (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Narc

- **DEFAULT ADJUSTMENTS RESTORED**

Ao ver a mensagem **DEFAULT ADJUSTMENTS RESTORED**, pressione F3.

NBA Hangtime / NBA Maximum Hangtime

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **O som deste sistema é MUITO BAIXO!**

Pressione F2, vá até a opção **Volume Adjust**, selecione com o botão 1 do **jogador 1**, aumente o volume e pressione o botão 1 do **jogador 1** para concluir. Não ultrapasse 80-90% do volume pois podem ocorrer distorções e saturação no áudio. Para finalizar selecione **Exit to Game Over**.

ParaParaParadise

- **ERROR E 160 BACKUPRAM ERROR**
- **ERROR E 166 RTC UNFORMAT**
- **ERROR E 124 FLASH DATA ERROR**

Pressione F2, vá até **SOUND FLASH INIT** e pressione 1 no teclado para selecionar a opção, selecione **SOUND FLASH INIT** e pressione 1 novamente, quando acabar pressione 1 novamente. De volta ao menu principal selecione **ALL FACTORY SETTINGS** pressionando 1 no teclado, selecione **ALL FACTORY SETTINGS** novamente e pressione 1, escolha **YES** e confirme com 1, ao retornar ao menu principal segure Shift e pressione F3.

Assista este procedimento no Youtube:

<https://www.youtube.com/watch?v=0eKpoKAlFSw>

ParaParaDancing

- **ERROR E 160 BACKUPRAM ERROR**
- **ERROR E 166 RTC UNFORMAT**
- **ERROR E 124 FLASH DATA ERROR**

Consulte *ParaParaParadise*.

ParaParaParadise v1.1

- **ERROR E 160 BACKUPRAM ERROR**
- **ERROR E 166 RTC UNFORMAT**
- **ERROR E 124 FLASH DATA ERROR**

Consulte *ParaParaParadise*.

ParaParaParadise 1st Mix Plus

- **ERROR E 160 BACKUPRAM ERROR**
- **ERROR E 166 RTC UNFORMAT**
- **ERROR E 124 FLASH DATA ERROR**

Consulte [ParaParaParadise](#).

Pac-Slot

Reinicie o MAME 2x ao ver a mensagem **DIP SW READ ERROR** e **BACKUP MEMORY ERROR**

Phantom Pays

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Se realizado corretamente o valor de **RF/AMT** será alterado de 0 para **200**, pressione **F1**, depois **F2**.

Phrase Craze

Ao ver a mensagem **PUSH ANY SWITCH** pressione Espaço ou a tecla **C**.

Pirate

Para inicializar o sistema, ative o modo de serviço com **F2** e pressione **F3**. Mova o ponteiro até **INIT** com **C** e pressione **1**. Mantenha **Z (Cancel)** por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com **F2** e reinicie com **F3**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Pirate Ship

- **E510 BACK UP RAM ERROR**

Ao ver a mensagem **E510 BACK UP RAM ERROR** mantenha as teclas **F2** e **F3** pressionadas até aparecer uma lista com tudo **OK**.

Player's Edge Plus (Set038) Set Chip

Este sistema não é um jogo. É um utilitário para teste de tela para ser usado depois que um novo jogo for carregado.

Suponha que tenha o CI 'pepp0516' instalado e queira configurá-lo. Num sistema real talvez queira adicionar um validador de notas (bill validator), a única maneira de fazê-lo seria removendo o CI 'pepp0516', substituí-lo pelo 'peset038' e depois reiniciar o sistema. Então o programa deste CI roda para que você defina as configurações e depois o substitua pelo 'pepp0516' novamente.

A única maneira de simular isso num emulador é rodar a configuração do jogo 'pepp0516', encerrar a emulação, copiar o arquivo pepp0516.nv para peset038.nv e depois rodar o programa 'peset038'. Isso é necessário pois para funcionar você precisa ter a mesma EEPROM e os dados CMOS na mesma memória. Quando terminar com o programa peset038.nv você o renomeia de volta para pepp0516.nv.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Player's Edge Plus games (peplus.cpp)

• CHANGE REQUEST ... - CALLING ATTENDANT ...

Válido para os sistemas: pebe0014, pebe0014a, pebe1001, peke1001, peke1006, peke1012, peke1012a, peke1012b, peke1013, peke1013a, pekoc766, pekoc801, pekoc802, pekoc803, pekoc804, pekoc806, pekoc818, pekoc819, pekoc825, pebe0014, pepp0065, pepp0188, pepp0250, pepp0516, peps0014, peps0022, peps0043, peps0045, peps0308

Existem duas maneiras de se iniciar esses sistemas.

1. Pressione a tecla O.
2. **CMOS DATA** será exibido.
3. Pressione a tecla K.
4. **EEPROM DATA** será exibido.
5. Pressione a tecla O seguido da tecla K.

Válido para os sistemas: pepp0043, pepp0158, pepp0447, peps0615, peps0716, pex2069p, pexmp006, pexmp017, pexmp024, pexp0019, pexp0112, pexs0006

1. Pressione a tecla O.
2. **CMOS DATA** será exibido.
3. Mantenha a tecla K pressionada até ouvir o som de um "beep".
4. Pressione a tecla O.
5. Pressione a tecla L.
6. Pressione a tecla O.
7. **EEPROM DATA** será exibido.
8. Pressione a tecla K seguido da tecla O.

Poker (All Sets Version 50.02 ICB) - Greyhound Electronics

- **ERROR**

Ao ver a palavra **ERROR** pressione F2 + F3 para iniciar o **SELF TEST**, quando terminar pressione F2.

Four In One Poker

- **DATA ERROR**

Ao ver a mensagem **DATA ERROR** pressione F3.

Poker De Win

- **INIT MACHINE**

Ao ver a mensagem **INIT MACHINE** pressione a tecla 9 seguido da tecla 0. Pressione a tecla 9 para continuar.

Poker Master (Tony Poker) Hack

Para que o sistema funcione é necessário **desligar** todos os 5 bancos de chaves e depois ligá-las na seguinte configuração:

1. (DIP1) 1, 3, 4
2. (DIP2) 2, 7
3. (DIP3) 1, 2, 4, 5, 6, 7, 8
4. (DIP4) 5
5. (DIP5) 3, 6, 7, 8

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Pop'n Music 4

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Pressione F2, selecione **ALL FACTORY RESET** e confirme, segure Shift e pressione F3. Na segunda tela de erro **E165** pressione F2, acesse o [Firebeat RTC Recovery Password](#) e gere a sua senha. Ao terminar de inserir a senha, pressione F2, segure Shift e pressione F3.

Caso apareça o erro **E750: SPU DVDROM ERROR** depois deste procedimento, será preciso refazer o chd com o programa chdman, é importante usar a versão do chdman mais recente, neste exemplo estamos usando a versão que acompanha o **MAME 0.229**.

Antes de prosseguir, **FAÇA O BACKUP DOS ARQUIVOS QUE VOCÊ FOR CONVERTER!**

Pegue o **maior arquivo** chd do sistema, neste caso seria o gq986jaa02.chd e faça o comando:

```
chdman extractraw -i gq986jaa02.chd -o gq986jaa02.chd.iso
```

Renomeie o arquivo gq986jaa02.chd para algo como backup-gq986jaa02.chd e faça o comando:

```
chdman creatcd -i gq986jaa02.iso -o gq986jaa02.chd
```

Na próxima vez que a máquina iniciar não haverá mais o erro **E750: SPU DVDROM ERROR**. Caso já tenha tentado rodar o sistema antes e depois da conversão aparecer o erro:

```
gq986jaa02.chd DIFF CHD ERROR: invalid parent
```

Neste caso exclua o arquivo **diff\gq986jaa02.dif** e rode o sistema novamente, a dica serve para todas as outros sistema caso o mesmo erro ocorra.

Pop'n Music 5

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte [Pop'n Music 4](#)

Pop'n Music 6

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte [Pop'n Music 4](#)

Pop'n Music 7

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte [Pop'n Music 4](#)

Pop'n Music 8

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte *Pop'n Music 4*

Pop'n Music Mickey Tunes

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte *Pop'n Music 4*

Pop'n Music Mickey Tunes!

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte *Pop'n Music 4*

Pop'n Music Animelo

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte *Pop'n Music 4*

Pop'n Music Animelo 2

- **!!ERROR**
- **E161: BATTERY ERROR**
- **E165: RTC DATA ERROR**

Consulte *Pop'n Music 4*

Pontoon

- **JUST A MOMENT PLEASE...**

Ao ver a mensagem **JUST A MOMENT PLEASE...** pressione 0 para continuar.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Aviso: Até a presente versão do MAME este sistema ainda não funciona.

Pool 10

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Pot Game

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Premier Soccer

- **EEPROM 15A BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Print Club 2

- **Please Call Attendant**

Após a mensagem **Please Call Attendant** pressione F2 tela de serviço, pressione a tecla 9 para navegar até **SYSTEM ASSIGNMENTS** e pressione F2, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes. Para que o sistema funcione é necessário selecionar a BIOS US.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Aviso: Até a presente versão do MAME este sistema ainda não funciona.

Pro Mahjong Kiwame

- **BACKUP ERROR**

Pressione F3 ao ver a mensagem **BACKUP ERROR**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Pro Mahjong Kiwame S

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Configuração Extra

Após o boot pressione novamete F2, com a tecla 9 navegue até **Each Game Test / [KIWAME S]** e pressione F2, pressione F2 novamente na tela com as mesas e cadeiras, uma opção escrita em japonês começara a piscar te levando até a tela de configuração. Pressione 9 2x e mude a opção de **MANUAL** para **AUTO**, pressione 9 mais 10x para sair da tela de configuração ficando apenas uma tela de fundo com cadeiras e mesas. Pressione F2 para voltar a tela de serviço, selecione **EXIT** com a tecla 9 e pressione F2 para confirmar.

Prop Cycle

Para calibrar a direção mantenha pressionada a tecla 9 e pressione F2, encerre pressionando novamente F2.

Pull Tabs

- **DATA ERROR**

Pressione F3 ao ver a mensagem **DATA ERROR**.

Pye-nage Taikai

- **BACK UP RAM NG**

Mantenha F2 pressionado seguido de F3 para reiniciar o sistema e entrar na tela de serviço. Use F2 para selecionar a opção **3. Set Mode** seguido da tecla Ctrl. Use F2 para ir passando pelas opções até chegar em **RAM CLEAR**, mantenha Ctrl pressionado até aparecer **OK** na tela. Reinicie o sistema com F3.

Qix

Use a tecla 9 para passar por todas as telas de configuração até chegar na tela do título.

Quiz Gakumon no Susume

- **EEPROM 15B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Quiz Sangokushi

Pressione a tecla 2 quando aparecer a tela **MENU**.

Quiz Tonosama no Yabou

Pressione a tecla 2 quando aparecer a tela **MENU**.

Raiden Fighters (Todas as Regiões)

Esse sistema necessita de uma memória NVRAM válida para funcionar, seja paciente durante o processo de inicialização pois os testes internos deste sistema demoram bastante. Talvez a opção `-nothrottle` ajude a acelerar o processo.

Quando o contador chegar no **000** pressione Tab, vá em *Chaves DIP*, **Desligue** a chave **JP1**, encerre o MAME e inicie o sistema novamente.

Raiden Fighters Jet (Todas as regiões)

Esse sistema necessita de uma memória NVRAM válida para funcionar, seja paciente durante o processo de inicialização pois os testes internos deste sistema demoram bastante. Talvez a opção `-nothrottle` ajude a acelerar o processo.

Quando o contador chegar no **000** pressione Tab, vá em *Chaves DIP*, **Desligue** a chave **JP1**, encerre o MAME e inicie o sistema novamente.

Racing Force

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Aviso: Até a presente versão do MAME este sistema ainda não funciona.

Rampage World Tour

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **O som deste sistema é MUITO BAIXO!**

Pressione F2, vá até a opção **Volume Adjust**, selecione com o botão 1 do **jogador 1**, aumente o volume e pressione o botão 1 do **jogador 1** para concluir. Não ultrapasse 80-90% do volume pois podem ocorrer distorções e saturação no áudio. Para finalizar selecione **Exit To Game Over**.

Red Earth e sistemas CPS-3 em geral

O processo de inicialização da memória NVRAM destes sistemas é demorado, elas precisam criar uma memória NVRAM válida e completa. Uma maneira de acelerar um pouco este processo é utilizar a opção *-nothrottle* porém ainda assim todo o processo pode levar cerca de meia hora ou mais.

Resident

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Revolution X

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **CHECKING SCRATCH RAMS**

Para pular essa tela pressione 1.

- **O som deste sistema é MUITO BAIXO!**

Pressione F2, vá até a opção **Volume Adjust**, navegue com a tecla = e selecione com F2, aumente o volume com = ou reduza com -, para concluir pressione F2. Navegue até **Exit To Game Over** e pressione F2 para encerrar a configuração.

- **Como eu calibro as pistolas? Posso usar uma pistola de luz real?**

Depois de passar pelo processo de inicialização o sistema vai parar na tela de calibragem das pistolas, use a mira e atire nos pontos indicados.

Para aprender como utilizar e configurar as suas pistolas de luz leia o capítulo *Pistola de luz*.

Rim Rockin' Basketball

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar esse sistema para 4 jogadores?**

A quantidade predefinida de jogadores para esta e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as chaves DIP ou entrando na tela de serviço com F2.

Robotron

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Rock Climber (rclimb_3)

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Royal Card

Mantenha as teclas 9 e 0 pressionadas, toque em F3 até aparecer **RAM-Initializiert!**, reinicie se necessário.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Royal Vegas Joker Card

Mantenha as teclas 9 e 0 pressionadas, toque em F3, reinicie se necessário.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Aviso: Pode ser que a dica esteja errada, obsoleta ou já não funcione mais. A informação será mantida para futuras referências de versões mais antigas do MAME.

Run and Gun

- **EEP ROM 35E BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Salamander 2

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

San Fransisco Rush

Pressione o botão 1 no controle do **jogador 1**.

Scud Race

- **Cancelled. Network Board not present**

Este sistema ainda não possui a emulação da placa de rede, quando a mensagem **Cancelled. Network Board not present** aparecer pressione F2, navegue com Ctrl Esquerdo até a opção **GAME ASSIGNMENTS**, pressione F2 para selecionar, vá até o **LINK ID**, pressione F2 para alterar a opção para **Single**. Para encerrar, selecione a opção **EXIT** 2x.

<p>Aviso: Até a presente versão do MAME este sistema ainda não funciona.</p>

Sea Bass Fishing

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione F2 (Service) para entrar na tela de serviço. Navegue com a tecla 9 até a opção **SYSTEM ASSIGNMENTS**, pressione F2 para selecionar, com a tecla 9 navegue até **CABINET TYPE** e pressione F2 para selecionar **1P** encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Selection - Greyhound Electronics

- **ERROR**

Ao ver a palavra **ERROR** pressione F2 + F3 para iniciar o **SELF TEST**, quando terminar pressione F2.

Sengoku Mahjong [BET]

- **WARNING - DATA IN STOCK IS WRONG - CHECK RAM**

Pressione F3 ao ver a mensagem **WARNING - DATA IN STOCK IS WRONG - CHECK RAM**.

Shadow Dancer

- **Zerando a pontuação**

Pressione F2, navegue com o direcional até a opção **Backup Ram Clear**, Pressione F2 e confirme, selecione **EXIT** para encerrar a configuração.

Sigma Poker 2000

- **M. Error: Call Attendant!**

Pressione a tecla 0 e toque em F3, pressione B 3x até chegar em **Page 4**, navegue com C até **Clear All Informations**, mantenha pressionado a tecla Z até a tela piscar 1x. Encerre a configuração pressionando 0.

Sigma Poker

- **Init Machine**

Pressione a tecla 0, pressione 9. Após alguns segundos a tela de contabilidade vai aparecer (Operator Mode), pressione 0 depois 9, pressione F3 caso nada aconteça.

Simpsons Bowling

Para reinicializar a EEPROM, mantenha F2 pressionado e pressione F3 uma vez. Não solte F2 até que apareça a mensagem dizendo **Initializing EEPROM**.

Sinistar

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Slam Dunk

- **EEP ROM 35E BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Slither

- **CMOS INVALID**

Pressione 2 ao ver a mensagem **CMOS INVALID**.

Snooker 10

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Soccer Superstars

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

1 MONITOR SETTING!!! - DIP SW4=OFF

Para ativar a tela dividida para dois jogadores pressione Tab, vá em *Chaves DIP*, mude **SCREENS** para 2 e reinicie. A tela pode ficar melhor distribuída se o sistema rodar com uma tela para cada jogador. Rode o sistema com a opção `-numscreens 2`.

Solar Assault

- **EEPROM 24G BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Space Dungeon

Use a tecla 9 para passar por todas as telas de configuração até chegar na tela do título.

Space Gun

Para calibrar as pistolas pressione ambas as teclas F2 + F3 juntas para entrar na tela de serviço, pressione 2 até aparecer uma tela preta com um alvo piscando na parte superior esquerda da tela, atire nos alvos para calibrar a pistola para o jogador 1 e 2. Pressione a tecla 2 para continuar, deverá aparecer **EEPROM "WRITE" OK** e **EEPROM "READ" OK** na parte de cima da tela confirmando a operação.

Pressione ambas as teclas F2 + F3 para encerrar a configuração.

Speed Racer

Navegue para cima com X, para baixo com Y e selecione as opções com 9.

Para entrar na tela de ajustes dos controles pressione F2, navegue até **VOLUME ADJUST**, pressione a tecla Espaço na primeira opção, mova o direcional todo para a > e pressione 9, mova o direcional todo para a < e pressione 9, aperte o botão ou pedal do acelerador do **Jogador 1**. Pressione F3 para encerrar a configuração.

Splat!

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Star Trek (Defender bootleg)

Ao rodar o sistema pela primeira vez irá aparecer uma mensagem **01 0000 COINS LEFT** ou ficará parada numa tela com os dizeres **Factory settings restored**. Pressione F2 seguido de F3. Caso falhe, tente F3 ou F1 + F2.

Stargate

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Steel Gunner / Steel Gunner 2

Para calibrar as pistolas mantenha a tecla 9 pressionada e pressione F2, atire nos alvos com a pistola do jogador 1 e 2. Pressione F2 para encerrar a configuração.

Steep Slope Sliders

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Street Fighter III (Série) e sistemas CPS-3 em geral

O processo de inicialização da memória NVRAM destes sistemas é demorado, eles precisam criar uma memória NVRAM válida e completa. Uma maneira de acelerar um pouco este processo é utilizar a opção *-nothrottle* porém ainda assim todo o processo pode levar cerca de meia hora ou mais.

Sunset Riders

- **EEPROM 15B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Super Gran Safari

Durante a tela preta pressione V, X, B e 1.

Super High Impact

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

Super Pool 99

- **RAM BAD - ERRORE CHIAMARE ASSISTENZA**

Ao ver essa mensagem de erro mantenha pressionado as teclas 1 + 9 e toque em F3, mantenha essas teclas pressionadas até aparecer **RAM INIT**.

Super Real Mahjong Part 2

- **ERR BACK UP**

Pressione F2 seguido de F3, pressione F2 novamente e finalize com F3.

Sweet Life

Para inicializar o sistema, ative o modo de serviço com F2 e pressione F3. Mova o ponteiro até **INIT** com C e pressione 1. Mantenha Z (**Cancel**) por 5 segundos enquanto o contador regride. Ao concluir saia da tela de serviço com F2 e reinicie com F3.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Taihou de Doboön

- **BACK UP RAM NG**

Mantenha F2 pressionado seguido de F3 para reiniciar o sistema e entrar na tela de serviço. Use F2 para selecionar a opção **3. Set Mode** seguido da tecla 9. Use F2 para ir passando pelas opções até chegar em **RAM CLEAR**, mantenha 9 pressionado até aparecer **OK** na tela. Reinicie o sistema com F3.

Taisen Puzzle-dama

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Taisen Tokkar-dama

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Target Panic

O processo de inicialização da memória NVRAM destes sistemas é demorado, eles precisam criar uma memória NVRAM válida e completa. Uma maneira de acelerar um pouco este processo é utilizar a opção *-nothrottle* porém ainda assim todo o processo pode levar cerca de 7 minutos.

Technical Bowling

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para 1P (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Techno Drive

- **PRN PCB CONNECT : NG**

Pressione F2 ao ver a mensagem **PRN PCB CONNECT : NG**, navegue até **GAME OPTIONS**, pressione 9 para selecionar, navegue até **PRINTER** e mude a configuração para **OFF**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Calibrando os controles

Para calibrar os controles mantenha pressionado F2 e 9, mantenha pressionado o botão 1 do jogador 1 para configurar o acelerador (GAS), o botão 2 para o freio (BRAKE) e mova a direção ou direcional para a esquerda e direita. Pressione F2 para encerrar a configuração.

Teenage Mutant Ninja (Hero) Turtles 2 - Turtles in Time

- **EEPROM 15B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Ten Balls

Mantenha pressionada as teclas 9 + 0, toque em F3. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Terminator 2

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **Como eu calibro as pistolas? Posso usar uma pistola de luz real?**

Depois de passar pelo processo de inicialização, o sistema vai parar na tela de calibragem das pistolas, use a mira e atire nos pontos indicados.

Para aprender como utilizar e configurar as suas pistolas de luz leia o capítulo *Pistola de luz*.

The Couples

- **PUSH ANY SWITCH**

Mantenha o direcional **PARA BAIXO** pressionado ao ver a mensagem **PUSH ANY SWITCH**.

The Electric Yo-Yo

Use a tecla 9 para passar por todas as telas de configuração até chegar na tela do título.

The Simpsons

- **EEPROM 12C BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Thrill Drive (JAE/JAA)

- **RTC SELF CHECK ERROR**

Pressione F2 para continuar.

Thunder Cross II

- **EEPROM 17F BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Time Crisis

- **CALIBRANDO A MIRA**

Mantenha pressionado a tecla 9 e pressione F2, pressione Alt Esquerdo para selecionar a opção **GUN INITIALIZE [STANDARD]**, atire na tela e verifique se o tiro corresponde com o centro da mira, em caso afirmativo pressione 9 para encerrar a configuração.

Caso a mira esteja fora, pressione o botão de recarga (FOOT PEDAL) e alinhe o alvo com o centro da mira. pressione 9 para encerrar a configuração.

Caso a tela fique cheia de sombras das miras, pressione **SHIFT ESQUERDO** + F3.

Tobikose! Jumpman

- **DIP SW READ ERROR - BACKUP MEMORY ERROR**

Reinicie o MAME 2x.

Tokimeki Memorial Taisen Puzzle-dama

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Top Gear

- **Metering Error - Call Attendant**

Pressione **F1 (Jackpot Key)** e **F2 (Audit Key)**, ao entrar na tela de serviço chamada **Meter Set 1** pressione simultaneamente as teclas **Z + X + C**. Pressione **D** até aumentar o valor **CASHCADE MINIMUM** para **\$5**, pressione **F1** e depois **F2** para finalizar.

Tortuga Family

Mantenha pressionada as teclas **9 + 0**, toque em **F3**. Solte as teclas quando aparecer **RAM-INIZIALIZZATA!**.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Track & Field

Para ter a tabela de **High Score** devidamente inicializada, pressione **Tab**, vá para *Chaves DIP*, defina **World Records** como **Erase on Reset**, pressione **F3**. Redefina **World Records** como **Don't Erase**.

Triple Hunt

- **AIM GUN AT SQUARE AND FIRE**

Mire e atire no pequeno quadrado na parte superior esquerda e inferior direita da tela, ao ver a mensagem **END OF TEST - START GAME**, pressione Tab, vá até *Chaves DIP*, desligue a opção **Service Mode** e reinicie.

Turkey Shoot

Pressione F2 ao ver a mensagem **FACTORY SETTINGS RESTORED**.

Twin Bee Yahhoo!

- **EEPROM 22D BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Vendetta

- **EEPROM 13B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Videomat

- **Init Machine**

Ligue o **Operator Key** pressionando 9. Pressione **Door Key** pressionando a letra 0, após alguns segundos a tela de contabilidade vai aparecer (Operator Mode), pressione 9 para iniciar o jogo ou F3 caso nada aconteça.

Violent Storm

- **EEPROM 13C BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Virtual Mahjong

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Virtual Mahjong 2 - My Fair Lady

- **ERROR ON CARTRIDGE**

Após a mensagem **ERROR ON CARTRIDGE** pressione 9 (Service) depois F2 (Test) para entrar na tela de serviço. Selecione a opção **SYSTEM ASSIGNMENTS** com 9 e pressione F2 para prosseguir, defina **CABINET TYPE** para **1P** (um jogador), encerre selecionando **EXIT** duas vezes.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Virtual Racing

- **MULTIPLAYER E LIVE MONITOR**

Para que a configuração do seu MAME não fique bagunçada, crie uma pasta qualquer em algum lugar do seu desktop e dentro copie o executável do MAME, chama o prompt de comando caso esteja no Windows ou o terminal caso esteja no Linux ou macOS e execute o comando `mame -cc` para criar a sua configuração inicial.

Edite o seu arquivo **mame.ini** e altere as opções abaixo:

```
rompath roms;caminho-completo-das-suas-roms
skip_gameinfo 1
window 1
resolution 640x480
```

Faça o comando `mkdir nvram1 nvram2 nvram3` para criar os diretórios de configuração. O diretório **nvram1** será a configuração do sistema **master**, **nvram2** será a configuração sistema **slave** e o diretório **nvram3** será a configuração do nosso **live monitor**.

Para configurar o sistema **master** execute o comando abaixo:

```
mame vr -wdog 10 -nvram_directory nvram1
```

Pressione F2 para entrar na configuração do sistema, vá até **Game System**, defina **Link ID** para **Master**, aproveite que já está aí e altere a cor do veículo, caso contrário todos serão vermelhos. A opção `-wdog 10` serve para encerrar o MAME caso ele trave. Salve, ao retornar para a tela de configuração pressione Esc.

Para configurar o sistema **slave** (player 2) faça o mesmo, porém na linha de comando troque o **nvram1** por **nvram2**, em **Link ID** defina como **Slave** e não se esqueça de alterar a cor do carro. Salve, ao retornar para a tela de configuração pressione Esc.

Para configurar o sistema **live monitor** faça o mesmo, porém na linha de comando troque o **nvram2** por **nvram3**, em **Link ID** defina como **Live**. Salve, ao retornar para a tela de configuração pressione Esc.

No Windows, abra o seu editor de texto preferido e cole as linhas abaixo:

```
start /b mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram1 -comm_localport_
↪ 15112 -comm_remoteport 15114 -comm_framesync
start /b mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram2 -comm_localport_
↪ 15113 -comm_remoteport 15112 -comm_framesync
start /b mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram3 -comm_localport_
↪ 15114 -comm_remoteport 15113 -comm_framesync
```

No Linux ou macOS:

```
#!/bin/sh
./mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram1 -comm_localport 15112 -
↪ comm_remoteport 15114 -comm_framesync &
./mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram2 -comm_localport 15113 -
↪ comm_remoteport 15112 -comm_framesync &
./mame vr -window 640x480 -noka -wdog 10 -nvram_directory nvram3 -comm_localport 15114 -
↪ comm_remoteport 15113 -comm_framesync
```

Salve o arquivo como **vr-multi.bat** no Windows ou **vr-multi** no Linux ou macOS, para este último é necessário fazer o comando `chmod 755 vr-multi`. Agora ao executar estes arquivos deve aparecer três telas, uma para o jogador 1, para o jogador 2 e uma para o live monitor.

Este sistema suporta até 4 sistemas "Twin" conectados entre si totalizando 8 jogadores mais o live monitor. O segredo está em conectar um sistema na outro e o último (live monitor) conectando de volta para o sistema master. No exemplo acima, com a opção `-comm_remoteport` o sistema **nvram2** (**Player 2**) está apontando para a porta do sistema **nvram1** (**Player 1**), o sistema **nvram3** (**Player 3**) aponta para a porta da **nvram2** (**Player 2**) e assim sucessivamente.

A configuração acima serve apenas como um exemplo. É preferível que seja criada uma rede entre 8 ou 9 computadores para que eles possam conversar entre si, neste caso não é necessário criar diferentes

diretórios **nvr**am. No Windows, é necessário instalar o [OpenVPN](#) mais recente para que o MAME possa ver e usar os adaptadores de rede. Num exemplo de configuração de rede ela ficaria assim:

```
Computador MASTER Player 1
mame vr -wdog 10 -comm_localhost 192.168.1.1 -comm_localport 15111 -comm_remotehost 192.
↪168.1.10 -comm_remoteport 15110

Computador SLAVE Player 2
mame vr -wdog 10 -comm_localhost 192.168.1.2 -comm_localport 15112 -comm_remotehost 192.
↪168.1.1 -comm_remoteport 15111

Computador SLAVE Player 3
mame vr -wdog 10 -comm_localhost 192.168.1.3 -comm_localport 15113 -comm_remotehost 192.
↪168.1.2 -comm_remoteport 15112
...
Computador LIVE MONITOR
mame vr -wdog 10 -comm_localhost 192.168.1.10 -comm_localport 15110 -comm_remotehost_
↪192.168.1.1 -comm_remoteport 15111
```

Warzard e sistemas CPS-3 em geral

O processo de inicialização da memória NVRAM destes sistemas é demorado, eles precisam criar uma memória NVRAM válida e completa. Uma maneira de acelerar um pouco este processo é utilizar a opção *-nothrottle* porém ainda assim todo o processo pode levar cerca de meia hora ou mais.

Water Match

Os controles desse sistema são impossíveis! O que eu faço para poder jogar?

Segundo as instruções do sistema, os controles envolvem chacoalhar freneticamente dois controles feito um maluco enquanto pressiona um botão em determinados momentos do jogo. Veja mais informações sobre este sistema no [SEGA Retro](#) e no [Arcade Museum](#), caso queira se aventurar neste sistema pesquise pelo nome dele no YouTube.

- **FREESTYLE e BACKSTROKE**

Mova os controles para cima e para baixo de maneira que os controles sempre fiquem em posições opostas (enquanto um estiver para cima o outro deve ficar para baixo e vice-versa). Movimente os controles o mais rápido que conseguir.

- **BUTTERFLY**

Rode o controle esquerdo em sentido horário e o controle direito em sentido anti-horário.

- **BREASTSTROKE**

Mova os controles "para dentro", mova o controle esquerdo em sentido anti-horário e o controle direito em sentido horário. Para mergulhar na piscina o botão 1 do Jogador 1.

- **REMO**

Mova ambos os controles para cima e para baixo ao mesmo tempo.

- **CANOAGEM**

Mova ambos os controles de forma independente, quando a canoa estiver inclinando para a esquerda, mova o controle esquerdo, quando estiver inclinando para a direita mova o controle direito. Talvez seja necessário a interação do botão em determinados momentos, talvez quando a canoa virar.

Wild West C.O.W.-Boys of Moo Mesa

- **EEPROM N2 BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Witch

- **RAM ERROR - CALL ATTENDANT!!**

Pressione F1.

World Tennis

Pressione F3 para sair da tela de serviço.

WWF Wrestlemania

- **CMOS INVALID -- FACTORY SETTINGS RESTORED**

Pressione F3 para continuar.

- **O som deste sistema é MUITO BAIXO!**

Pressione F2, vá até a opção **Volume Adjust**, selecione com o botão 1 do **jogador 1**, aumente o volume e pressione o botão 1 do **jogador 1** para concluir. Não ultrapasse 80-90% do volume pois podem ocorrer distorções e saturação no áudio. Para finalizar selecione **Exit To Game Over**.

X-Men

- **EEPROM 13B/16B BAD**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Xexex

- **VERSION ERROR**

Caso este erro apareça ao rodar o sistema pela primeira vez, mantenha pressionado a tecla F2 e toque na tecla F3 para reiniciar o sistema e criar uma nova EEPROM.

Nota: Na última versão disponível do MAME este sistema não apresenta qualquer problema que necessite desta intervenção. Talvez o código-fonte do driver responsável por ele tenha sido atualizado ou a informação já esteja obsoleta. A informação será mantida para futuras referências de versões mais antigas do MAME.

Zero

Ao rodar o sistema pela primeira vez irá aparecer uma mensagem **01 0000 COINS LEFT** ou ficará parada numa tela com os dizeres **Factory settings restored**. Pressione F2 seguido de F3. Caso falhe, tente F3 ou F1 + F2.

Zero 2

Ao rodar o sistema pela primeira vez irá aparecer uma mensagem **01 0000 COINS LEFT** ou ficará parada numa tela com os dizeres **Factory settings restored**. Pressione F2 seguido de F3. Caso falhe, tente F3 ou F1 + F2.

Zero Team 2000 (zerotm2k)

- **Eu me lembro que jogava num sistema com 4 jogadores mas o jogo só permite 2. Como posso configurar este sistema para 4 jogadores?**

A quantidade predefinida de jogadores para este e outros sistemas é de 2 jogadores ainda que o programa tenha suporte para mais. Geralmente você pode ajustar a quantidade de jogadores usando a configuração com as *Chaves DIP* ou entrando na tela de serviço com F2.

Zombie Raid

- **Calibrando a Pistola**

Mantenha pressionado F2 e toque em F3, use Ctrl Esquerdo para navegar até **7. GUN ADJUSTMENT MODE**, pressione 1, selecione **GUN ADJUSTMENT**, mova a mira e atire nos alvos indicados na tela. Quando retornar para a tela principal mantenha pressionado F2 e pressione F3.

Zoo Keeper

Pressione 9 para passar por todas as telas de configuração.

CONFIGURAÇÕES E OPÇÕES PARA A LINHA DE COMANDO

5.1 Opções Universais de linha de comando

- *Comandos opções e parâmetros*
- *Conjunto de instruções*
- *Nome dos arquivos e a localização dos diretórios*
- *Opções de ajuda e verificação*
- *Opções de configuração*
- *Opções para listagem*
- *Opções relacionadas ao que é exibido na tela (OSD)*
- *Opções de MIDI e rede*
- *Opções de saída das notificações de tela*
- *Opções para a configuração*
- *Opções para a configuração dos principais diretórios*
- *Opções para a gravação e a reprodução do estado da emulação*
- *Opções para a gravação de áudio e vídeo*
- *Opções para a captura da tela*
- *Opções relacionadas ao desempenho e a velocidade da emulação*
- *Opções para a rotação da tela*
- *Opções para a configuração de vídeo*
- *Opções para a configuração da tela inteira*

- *Opções de vídeo para uso com janelas individuais*
- *Opções para uso com as ilustrações*
- *Opções para os ajustes de imagem da tela*
- *Opções para sistemas que usem gráficos vetoriais*
- *Opções das principais características do vídeo OpenGL*
- *Opções de vídeo OpenGL GLSL*
- *Opções para a configuração do áudio*
- *Opções para as configurações de diferentes entradas*
- *Opções das entadas principais ativadas automaticamente*
- *Opções voltadas para a depuração*
- *Opções para a configuração da rede*
- *Opções diversas*
- *Opções para uso com script*
- *Opções do servidor HTTP*

5.1.1 Comandos opções e parâmetros

Os comandos incluem o próprio executável do **mame**, bem como *as várias ferramentas* que estão incluídas na distribuição do MAME.

As opções (chamadas de "verbos" na documentação original) são ações que serão executadas em conjunto com o comando e qualquer outra ação auxiliar a opção (quando ela existir) nós a chamamos de **parâmetro** da opção. No Português Brasileiro, subentende-se que quando falamos **comando** estamos nos referindo ao comando junto com a opção (quando for o caso). Por exemplo, no comando `mame -validate pacman`, **mame** é o comando em si, **-validate** é a opção e **pacman** o parâmetro que desejamos usar com a opção **-validate**.

5.1.2 Conjunto de instruções

Muitos comandos são compatíveis com o uso de um *conjunto de instruções* ou um padrão¹, que pode ser um sistema ou um nome abreviado do dispositivo (como por exemplo, **a2600**, **zorba_kbd**) ou um conjunto de instruções globais que correspondam a um dos dois como **zorba_*** por exemplo.

Dependendo do comando que foi combinado com este conjunto de instruções, a correspondência dessas combinações podem equiparar um sistema (ou vários deles) assim como dispositivos. É aconselhável colocar aspas em torno dos parâmetros para evitar que o seu ambiente tente interpretá-los de forma independente em relação aos nomes dos arquivos que desejamos usar (`mame -validate "pac*"` por exemplo).

¹ **Pattern**, segundo o *Oxford Dictionary* significa arranjar algo de forma repetitiva, seguindo um padrão, uma padronagem. Tradicionalmente "*pattern*" é traduzido como "*padrão*" porém fica claro que não estamos falando de algo igual sendo repetido, mas de um conjunto de instruções ou um conjunto de comandos em cadência que está informando ao programa as opções que o usuário deseja usar. (Nota do tradutor)

5.1.3 Nome dos arquivos e a localização dos diretórios

Nativa em (O MAME aceita a utilização de mais de um caminho nas suas configurações, como por exemplo, as configurações que permitam a pesquisa das ROMs em diferentes locais, desde que tais configurações utilizem ; para separar cada caminho.

O MAME consegue também identificar o caminho dos locais dos diretórios usando as variáveis de ambiente já existente no seu sistema e sua sintaxe irá depender do sistema operacional a ser usado. No Windows por exemplo, caso a configuração `%APPDATA%\mame\cfg` seja definida, o MAME conseguirá ler a variável `%APPDATA%` e resolver o caminho completo para o diretório de dados de aplicativos do usuário.

Em sistemas estilo UNIX como macOS e Linux que utilizam o interpretador de comandos *Bourne Shell*, aconteceria o mesmo caso o caminho seja definido nas configurações como `/home/${USER}/.mame/cfg`. Assim como o sinal de porcentagem % é usado numa palavra para se definir uma variável no ambiente do *Windows*, o sinal de til ~ serve como um redirecionador para o diretório home do usuário, assim em vez de se digitar o caminho completo `/home/${USER}/.mame/cfg` é possível simplificar usando `~/.mame/cfg`.

Note porém que o MAME só aceita interpretações de variáveis simples, o MAME não reconhece expressões mais complexas compatíveis com Bash, ksh, ou zsh.

Os caminhos relativos são resolvidos a partir do diretório de trabalho atual. Caso inicie o MAME com um clique duplo, o diretório de trabalho atual será aquele onde estiver o executável do MAME. Caso faça o mesmo usando um macOS, será aberto uma janela de terminal onde o seu diretório home será o seu diretório de trabalho.

Caso queira um comportamento semelhante ao Windows Explorer no macOS, crie um script que contenha as linhas do exemplo abaixo no mesmo diretório onde se encontra o executável do MAME, no exemplo abaixo está sendo usada a versão 64-bit do MAME, mude conforme necessário.

```
#!/bin/sh
cd "`dirname "$0"`"
exec ./mame
```

Salve o script com um nome qualquer como **meumame** por exemplo no mesmo diretório do executável do MAME, abra o seu terminal e torne o script executável com o comando **chmod u+x meusame**, agora ao executar o script, uma janela do terminal se abrirá, o MAME será executado e o diretório de trabalho será definido no mesmo local do script.

5.1.4 Opções de ajuda e verificação

-help / -h / -?

Exibe a versão atual do MAME e o aviso de direitos autorais.

```
mame -help
```

-validate / -valid <palavra chave>

Executa uma validação interna num ou mais drivers e dispositivos no sistema. Execute isso antes de enviar qualquer alterações para nós visando garantir qualquer tipo de violação em nenhuma das regras do sistema principal.

Caso um padrão seja definido, ele validará a correspondência predefinida do sistema em questão, caso contrário, validará todos os sistemas e dispositivos. Note que caso um padrão seja definido, este será comparado apenas com sistemas e não com outros dispositivos, nenhum tipo de validação será realizada com dispositivos.

Exemplo:

```
mame -validate
Driver ace100 (file apple2.cpp): 1 errors, 0 warnings
Errors:
Software List device 'flop525_orig': apple2_flop_orig.xml: Errors parsing_
↳software list:
apple2_flop_orig.xml(126.2): Unknown tag: year
apple2_flop_orig.xml(126.8): Unexpected content
apple2_flop_orig.xml(127.2): Unknown tag: publisher
```


-verifyroms <palavra chave>

Verifica a condição dos arquivos da imagem ROM num determinado sistema. Serão verificados todos os sistemas e diretórios válidos que estejam dentro do rompath (caminho da rom):

Exemplo:

```
mame -verifyroms pacman
romset pacman [puckman] is good
1 romsets found, 1 were OK.
```

É possível usar um asterisco ao final do nome do sistema para que seja exibido uma lista com todos os outros sistemas relacionadas com o nome do sistema principal e a sua condição atual, exemplo:

Exemplo:

```
mame -verifyroms pacman*
romset pacman [puckman] is good
romset pacmanbl [puckman] is good
...
pacmanfm      : pm1-1.7f (32 bytes) - NEEDS REDUMP
pacmanfm      : pm1-4.4a (256 bytes) - NEEDS REDUMP
romset pacmanfm [puckman] is best available
...
romset pacmaniao [pacmania] is good
...
```

Todos os sistemas e os arquivos da imagem ROM serão verificadas caso nenhum nome seja informado.

-verifysamples <palavra chave>

Verifica a condição dos arquivos **samples** informado. Todos os arquivos samples ou diretórios válidos serão verificados desde que estejam configurados em samplepath:

Exemplo:

```
mame -verifysamples 005
sampleset 005 is good
1 samplesets found, 1 were OK.
```

É possível usar um asterisco ao final do nome do sample para que seja exibido uma lista com todos os outros samples relacionados com o nome do sample principal e a sua condição atual, exemplo:

Exemplo:

```
mame -verifysamples armora*
sampleset armora is good
sampleset armorap [armora] is good
sampleset armorar [armora] is good
3 samplesets found, 3 were OK.
```

Todas os samples serão listados caso nenhum nome seja informado.

-verifysoftware / -vsoft <palavra chave>

Verifica se há imagens ROM inválidas ou ausentes na lista de software. Por predefinição, todos os drivers que possuem arquivos .zip ou diretórios válidos no rompath (caminho da rom) serão verificados, no entanto, é possível limitar essa lista definindo um nome de driver específico ou combinações após o comando -verifysoftware.

Exemplo:

```
mame -vsoft x68000
romset x68k_flop:2069ad is good
romset x68k_flop:3takun is good
romset x68k_flop:38mankk is good
romset x68k_flop:4thunit is good
...
0000 romsets found in 1 software lists, 0000 romsets were OK.
```

-verifysoftlist / -vlist <nome da lista de programa>

Verifica ROMs ausentes com base numa lista de software predeterminado na pasta **hash**. É predefinido que a busca e a verificação será feita em todos os drivers e arquivos .zip em diretórios válidos no *rompath* (caminho da rom), no entanto, é possível filtrar essa lista usando uma palavra chave ou coringa em "softwarelistname" após o comando -verifysoftlist. As listas estão na pasta *hash* e devem ser informadas sem a extensão .xml.

O resultado é exatamente igual ao comando -verifysoftware, porém usando uma lista de software.

Exemplo:

```
mame -vsoft x68k_flop
romset x68k_flop:2069ad is good
romset x68k_flop:3takun is good
romset x68k_flop:38mankk is good
romset x68k_flop:4thunit is good
...
0000 romsets found in 1 software lists, 0000 romsets were OK.
```

5.1.5 Opções de configuração

-createconfig / -cc

Cria um arquivo `mame.ini` pré-configurado. Todas as opções de configuração (não verbos) descritos abaixo podem ser permanentemente alterados ao fazer a edição deste arquivo de configuração. Consulte também o capítulo [como criar apenas um mame.ini com as configurações de fábrica](#).

Exemplo:

```
mame -cc
```

-showconfig / -sc

Exibe as configurações atualmente usadas. É possível direcionar essa saída para um arquivo ou também é possível utilizá-lo como um arquivo `.ini`, como mostra o exemplo abaixo:

Exemplo:

```
mame -showconfig > mame.ini
```

É o mesmo que **-createconfig**.

-showusage / -su

Exibe todas as opções disponíveis no MAME que sejam compatíveis com o seu sistema operacional ou a versão do MAME que estiver usando, cada opção será acompanhada de um breve descritivo (em inglês).

As configurações nativas do Windows como `hls1` por exemplo, não estarão disponíveis, tão pouco serão listadas nas versões SDL do MAME que rodem em Linux, macOS e assim por diante.

Todas as opções aparecem comentadas.

Exemplo:

```
mame -su
Usage:  mame [machine] [media] [software] [options]

Options:

#
# CORE CONFIGURATION OPTIONS
#
-readconfig      enable loading of configuration files
-writeconfig     write configuration to (driver).ini on exit
#
# CORE SEARCH PATH OPTIONS
#
-homepath        path to base folder for plugin data (read/write)
-rompath         path to ROM sets and hard disk images
```

5.1.6 Opções para listagem

É predefinido que todos os comandos `-list` abaixo, exibam informações na saída predefinida do sistema, geralmente é a tela do terminal onde o comando foi digitado. Caso queira gravar a informação num arquivo texto, adicione o exemplo abaixo ao final do seu comando:

`> nome do arquivo`

Onde '*nome do arquivo*' é o nome do arquivo texto que será criado para registrar toda a saída do terminal (por exemplo, `lista.txt`). Note que qualquer conteúdo prévio que exista dentro deste arquivo será apagado sem qualquer aviso prévio. Exemplo:

Isso cria (ou sobrescreve se já existir) o arquivo `lista.txt` e completa o arquivo com os resultados de `-listcrc puckman`. Em outras palavras, a lista de cada ROM usada em *Puckman* e o CRC para essa ROM é gravada nesse arquivo.

`-listxml / -lx < dispositivo | driver | sistema >`

Gera uma lista detalhada e completa de toda a informação que o MAME mantém em seu banco de dados interno sobre os seus dispositivos, sistemas, drivers, nome do driver assim como muitas outras informações em formato XML. A sua saída pode ser limitada informando um nome de dispositivo (**`ym2203`** por exemplo), um driver (**`megadriv`** por exemplo) ou sistema (**`sf2`** por exemplo).

Geralmente a saída deste comando é usado para ser redirecionado em um arquivo texto que posteriormente é utilizado por outras ferramentas como *gerenciadores de ROMs* e interfaces intermediárias *front-ends*.

Caso utilize o MAME com o PowerShell da Microsoft, leia também *Redirecionamento com o PowerShell da Microsoft*.

Exemplo

```
mame -lx sf2
<?xml version="1.0"?>
<!DOCTYPE mame [
<!ELEMENT mame (machine+)>
<mame build="0.246 (mame0246-58-g3ba776d3e0b)" debug="no" mameconfig="10">
  <machine name="sf2" sourcefile="capcom/cps1.cpp">
    <description>Street Fighter II: The World Warrior (World_
↪910522)</description>
    <year>1991</year>
    <manufacturer>Capcom</manufacturer>
    ...
  </machine>
</mame>
```

-listfull / -ll <palavra chave>

Exibe uma lista com o nome do sistema pesquisado e a sua descrição:

Exemplo:

```
mame -ll pacman
Name:          Description:
pacman         "Pac-Man (Midway)"
```

É possível usar um asterisco ao final do nome do sistema para que seja exibido uma lista com todas as outros sistemas relacionadas com o nome do driver principal e as suas respectivas descrições, exemplo:

Exemplo:

```
mame -ll pacman*
Name:          Description:
pacman         "Pac-Man (Midway)"
pacmanbl       "Pac-Man (Galaxian hardware, set 1)"
pacmanbla      "Pac-Man (Galaxian hardware, set 2)"
pacmanblb      "Pac-Man (Moon Alien 'AL-10A1' hardware)"
...
```

É possível também listar a descrição de sistemas, infelizmente nem todos os sistemas possuem descrições disponíveis ainda, exemplo:

Exemplo:

```
mame -ll neogeo*
Name:          Description:
neogeo         "Neo-Geo MV-6F"
neogeo_cart_slot "Neo Geo Cartridge Slot"
...

mame -ll genesis*
Name:          Description:
genesis        "Genesis (USA, NTSC)"
genesis_tmss    "Genesis (USA, NTSC, with TMSS chip)"
genesisisp      "Genesis"
...

mame -ll snes*
Name:          Description:
snes           "Super Nintendo Entertainment System / Super Famicom (NTSC)"
snes4sl        "SNES 4 Slot arcade switcher"
snespal        "Super Nintendo Entertainment System (PAL)"
...
```

Todos os sistemas ou drivers serão listados caso nenhum nome seja informado.

-listsource / -ls <palavra chave>

Exibe uma lista de drivers/dispositivos dos sistemas e o nome dos seus respectivos arquivos fonte. Útil para identificar qual driver o sistema roda, muito útil para o relatório de bugs. É predefinido que todos os sistemas e os dispositivos sejam listados; contudo, é possível limitar a lista através de um nome ou texto qualquer após a opção **-listsource**.

Exemplo:

```
mame -ls pacman
pacman          pacman/pacman.cpp
```

É possível também utilizar um curinga (asterisco) ao final do nome do sistema para que seja exibido uma lista com todos os outros sistemas que estejam relacionadas com o nome do sistema principal, exemplo:

Exemplo:

```
mame -ls pacman*
pacman          pacman/pacman.cpp
pacmanbl        galaxian/galaxian.cpp
...
pacmania        namco/namcos1.cpp
```

Todos os sistemas serão listados caso nenhuma palavra chave seja informada.

-listclones / -lc <palavra chave>

Exibe uma lista de clones de um determinado sistema. O MAME irá listar todos os clones em seu banco de dados porém a lista pode ser filtrada com o uso de uma palavra chave após o comando. Exemplo:

Exemplo:

```
mame -lc rallyx
Name:          Clone of:
dngrtrck       rallyx
rallyxa        rallyx
rallyxm        rallyx
rallyxmr       rallyx
```

-listbrothers / -lb <palavra chave>

Exibe uma lista com o nome do driver, da ROM principal e parentes que compartilhem do mesmo driver do sistema pesquisado.

Exemplo:

```
mame -lb 005
Source file:      Name:          Parent:
segag80r.cpp      005
segag80r.cpp      astrob
segag80r.cpp      astrob1      astrob
segag80r.cpp      astrob2      astrob
segag80r.cpp      astrob2a     astrob
segag80r.cpp      astrob2b     astrob
```

-listcrc <palavra chave>

Exibe uma lista completa com CRCs de todas as imagens ROM que compõem um sistema, os nomes do sistema ou do dispositivo num formato simples que pode ser facilmente filtrado por comandos como `grep`, `awk` e `sed` no Linux e macOS ou `findstr` no Windows. Caso nenhuma palavra chave seja usada como filtro após o comando, o MAME irá listar *tudo* que estiver em seu banco de dados interno.

Exemplo:

```
mame -listcrc 005
8e68533e 1346b.cpu-u25          005          005
29e10a81 5092.prom-u1
...
1d298cb0 6331.sound-u8         005          005
```

-listroms / -lr <palavra chave>

Exibe uma lista com todos os arquivos ROM que fazem parte de um sistema ou dispositivo. A lista mostra o nome dos arquivos ROM, os valores CRC e SHA1, assim como mostra também se uma das ROMs contidas no arquivo estão sinalizadas como **BAD_DUMP**. Isso significa que o conteúdo extraído não é válido, pode conter erro, não foi extraído de forma correta ou de forma apropriada, por algum motivo não pode ser validada, etc. Caso nenhuma palavra chave seja usada como filtro após o comando, o MAME irá listar *tudo* que estiver em seu banco de dados interno.

Exemplo:

```
mame -lr 005
ROMs required for driver "005".
Name                                     Size Checksum
1346b.cpu-u25                          2048 CRC(8e68533e)
SHA1(a257c556d31691068ed5c991f1fb2b51da4826db)
5092.prom-u1                          2048 CRC(29e10a81)
SHA1(c4b4e6c75bcf276e53f39a456d8d633c83dcf485)
...
6331.sound-u8                          32 BAD CRC(1d298cb0)
SHA1(bb0bb62365402543e3154b9a77be9c75010e6abc) BAD_DUMP
```

-listsamples <palavra chave>

Exibe uma lista das amostras que fazem parte de um determinado sistema, dos nomes de um sistema ou os nomes dos dispositivos. Caso nenhum termo seja usado como filtro depois do comando, *todos* os resultados dos sistemas e dos dispositivos serão exibidos.

Exemplo:

```
mame -listsamples 005
Samples required for driver "005".
lexplode
sexplode
dropbomb
shoot
missile
helicopt
whistle
```

-romident <caminho\completo\para\o\rom\desconhecida>

Tenta identificar os arquivos ROM desconhecidos comparando-o com os arquivos cadastrados no banco de dados interno do MAME que sejam utilizados por apenas um sistema ou que também sejam compartilhados por mais de um arquivo .zip específico. Este comando também pode ser usado para tentar identificar conjuntos de ROM retirados de placas desconhecidas. A opção vai identificar os arquivos compactados ou não.

Exemplo:

```
mame -romident rom_desconhecida.zip
Identifying rom_desconhecida.zip...
pacman.6j          = pacman.6j          msheartb  Ms. Pac-Man Heart Burn
                  = pacman.6j          mspacman  Ms.
↳ Pac-Man
...
                  = pacman.5e          puckmod   _
↳ Puck Man (Japan set 2)
```

Ao finalizar, o comando retorna níveis de erro (errorlevel):

- 0: significa que todos os arquivos foram identificados
- 7: significa que todos os arquivos foram identificados, exceto um ou mais arquivos não qualificados como "não-ROM"
- 8: significa que alguns arquivos foram identificados
- 9: significa que nenhum arquivo foi identificado

Nota: Apesar do "errorlevel" constar na documentação oficial, o comando não retorna **nenhum** destes valores, pelo menos não é visível no terminal ou linha de comando. O comando retorna apenas a listagem mostrada no exemplo.

-listdevices / -ld <palavra chave>

Exibe as especificações técnicas e todos os dispositivos conhecidos e conectados no sistema. Caso os slots sejam populados por dispositivos, todos os slots adicionais que esses dispositivos fornecerem ficarão visíveis com -listdevices também.

Exemplo:

```
mame -ld x68000
Driver x68000 (X68000):
<root>                X68000
adpcm_outl            Volume Filter
adpcm_outr            Volume Filter
crtc                  IX0902/IX0903 VINAS CRTC @ 38.86 MHz
exp1                  Sharp X680x0 expansion slot
exp2                  Sharp X680x0 expansion slot
flop_list             Software List
gfxdecode             gfxdecode
gfxpalette            palette
hd63450               Hitachi HD63450 DMAC @ 10.00 MHz
...
```

-listslots / -lslot <sistema>

Exibe uma lista com todos os slots disponíveis para o sistema e suas respectivas opções, caso estejam disponíveis.

Exemplo:

mame -lslot x68000			
SYSTEM	SLOT NAME	SLOT OPTIONS	SLOT DEVICE NAME

x68000	keyboard	x68k	Sharp X68000 Keyboard
		upd72065:0	525hd 5.25" high
	density floppy drive		
		upd72065:1	525hd 5.25" high
	density floppy drive		
		upd72065:2	525hd 5.25" high
	density floppy drive		
		upd72065:3	525hd 5.25" high
	density floppy drive		
		exp1	cz6bs1 Sharp CZ-
	6BS1 SCSI-1		neptunex
	Neptune-X		x68k_midi
	X68000 MIDI Interface		
		exp2	cz6bs1 Sharp CZ-
	6BS1 SCSI-1		neptunex
	Neptune-X		x68k_midi
	X68000 MIDI Interface		

Nem todos os itens opcionais acima estão conectados quando o sistema é iniciado, sendo necessário que o item descrito em **SLOT NAME** seja utilizado em conjunto com o **SLOT OPTIONS**, como por exemplo, para utilizar o dispositivo MIDI do seu computador faça:

Exemplo:

```
mame x68000 -exp1 x68k_midi -midiout "o seu dispositivo MIDI"
```

Para saber qual o dispositivo MIDI disponível no seu sistema, consulte o comando *-listmidi*.

-listbios <palavra chave>

Exibe uma lista das BIOS para um sistemas (caso esteja disponível). As opções da BIOS podem estar disponíveis para o sistema ou quaisquer dispositivos selecionados como opções de slot.

Caso nenhum nome seja informado, a BIOS de *todos* os sistemas compatíveis serão listados.

Exemplo:

```
mame -listbios apple2 -sl2 grapplus -sl4 videoterm
BIOS options for system Apple II (apple2):
    default          Original Monitor
    autostart        Autostart Monitor

BIOS options for device Orange Micro Grappler+ Printer Interface (-sl2_
↳grapplus):
    v30              ROM 3.0
    v32              ROM 3.2

BIOS options for device Vindex Videoterm 80 Column Display (-sl4 videoterm):
    v24_60hz         Firmware v2.4 (60 Hz)
    v24_50hz         Firmware v2.4 (50 Hz)
```

-listmedia / -lm <sistema>

Exibe uma lista de mídias ou formatos compatíveis com o sistema como cartucho, cassete, disquete, etc. O comando também exibe as extensões compatíveis como cada sistema caso elas existam, na dúvida, execute o comando `mame -lm sistema` para saber quais os tipo de mídia o MAME aceita e quais delas são compatíveis com o sistema em questão. Exemplo:

Exemplo:

SYSTEM	MEDIA NAME	(brief)	IMAGE FILE EXTENSIONS SUPPORTED
psu	memcard1	(memc1)	.mc
psu	memcard2	(memc2)	.mc
psu	quickload	(quik)	.cpe .exe .psf .psx
psu	cdrom	(cdrm)	.chd .cue .toc .nrg .gdi .iso

Caso queira carregar uma ROM num sistema como o Megadrive por exemplo faça `mame genesis -cart caminho_para_a_rom`. Outros sistemas podem aceitar outros formatos, no caso dos sistemas que rodem CD-ROM por exemplo, a opção pode ser `-cdrom caminho_para_a_imagem` ou `-cdrm caminho_para_a_imagem`, caso o sistema também aceite cartões de memória (memory card), é possível combinar a opção `-cdrom caminho_para_a_imagem` com `-memc1 caminho_para_a_imagem`.

-listsoftware / -lsoft <sistema>

Exibe o conteúdo de todas as listas de software que podem ser utilizadas pelo sistema ou pelos sistemas (praticamente são todos os arquivos XML que estão dentro do diretório hash).

Exemplo:

```
mame -lsoft x68000
<?xml version="1.0"?>
<!DOCTYPE softwarelists [
<!ELEMENT softwarelists (softwarelist*)>
  <!ELEMENT softwarelist (software+)>
    <!ATTLIST softwarelist name CDATA #REQUIRED>
    ...
    <software name="sf2ce">
      <description>Street Fighter II' Champion Edition</
↳description>
      <year>1993</year>
      ...
    </software>
  </softwarelist>
</softwarelists>
```

-getsoftlist / -glist <lista de programa>

Exibe o conteúdo de uma lista de software em formato XML, exatamente mesma coisa que -listsoftware acima, porém em vez do sistema se utiliza o nome da lista de programa.

Exemplo:

```
mame -glist msx1_cass
<?xml version="1.0"?>
<!DOCTYPE softwarelists [
<!ELEMENT softwarelists (softwarelist*)>
  <!ELEMENT softwarelist (software+)>
    <!ATTLIST softwarelist name CDATA #REQUIRED>
    ...
    <software name="albatex1">
      <description>Albatross - Extended Course 1 (Jpn)</
↳description>
      <year>1986</year>
      ...
    </software>
  </softwarelist>
</softwarelists>
```

5.1.7 Opções relacionadas ao que é exibido na tela (OSD)

-uimodekey <tecla>

Tecla usada para ativar ou desativar os controles de teclado do MAME. A configuração predefinida é **SCRLOCK** no Windows, **Forward Delete** no macOS ou **SCRLOCK** em outros sistemas como Linux por exemplo. Use **FN-Delete** em computadores/notebooks Macintosh que usem teclados compactos.

Exemplo:

```
mame ibm5150 -uimodekey DEL
```

-controller_map / -ctrlmap <nome_do_arquivo>

O caminho para um arquivo de texto contendo os mapeamentos do controle, botões e do direcional no formato usado pelo SDL2 e pelo Steam ou none para usar apenas o mapeamento nativo do MAME. O arquivo deve usar um formato de texto compatível com ASCII e com terminações de linha nativas (CRLF no Windows por exemplo). Atualmente é compatível apenas ao usar a opção *-joystickprovider sdljoy*.

Uma [lista de controles mapeados](#) da comunidade pode ser encontrada no GitHub. Além de usar um editor de texto, várias ferramentas estão disponíveis para criar os mapeamentos dos controles, incluindo o [SDL2 Gamepad Mapper](#) e o [SDL2 ControllerMap](#) que são [fornecidos com o SDL](#). Também é possível configurar o seu controle no modo "Big Picture" do Steam e copiar os mapeamentos a partir das entradas do `SDL_GamepadBind` no arquivo **config.vdf** encontrado na pasta **config** dentro da pasta de instalação do Steam.

Exemplo:

```
mame -controller_map gamecontrollerdb.txt sf2ce
```

-[no]background_input

Define se a entrada será aceita ou se será ignorada quando o MAME não tiver o foco na interface. No Windows, o RawInput é atualmente compatível para entrada com mouse e teclado, com o DirectInput, é compatível com mouse, teclado e joystick, com o XInput é compatível apenas com joystick. No SDL, o XInput é compatível com controle de jogo e joystick. Essa configuração é ignorada enquanto o depurador estiver ativo.

O valor predefinido é desligado. (**-nobackground_input**).

Exemplo:

```
mame -background_input ssf2tb
```

-uifontprovider <módulo>

Define a fonte que será renderizada na Interface do Usuário.

O valor predefinido é auto.

Exemplo:

```
mame ajax -uifontprovider dwrite
```

Tabela 1: Provedores compatíveis da fonte para a IU²³ separado por plataforma

Microsoft Windows	win	dwrite	auto		sdl ⁹	none
macOS			auto	osx	sdl	none
Linux			auto		sdl	none

-keyboardprovider <módulo>

Escolhe como o MAME lidará com a entrada do teclado.

O valor predefinido é auto.

Exemplo:

```
mame c64 -keyboardprovider win32
```

Tabela 2: Provedores compatíveis com a entrada do teclado separado por plataforma

Microsoft Windows	auto ¹⁰	rawinput	dinput	win32	sdl ¹¹	none
SDL (macOS e Linux)	auto ¹²				sdl	none

Observações

Nota: Observe que as ferramentas de emulação de teclado do modo de usuário, como o joy2key, quase certamente exigirão o uso da opção `-keyboardprovider win32` nas máquinas Windows.

²³ Interface do Usuário.

⁹ O binário oficial do MAME para Windows não é compilado com SDL, sendo necessário compilar uma versão compatível para que a opção `sdl` funcione.

¹⁰ No Windows, `auto` tentará `rawinput`, caso contrário, usa o `dinput`.

¹¹ Para ter suporte SDL no Windows é preciso compilar o MAME com `OSD=sdl`. O binário oficial do MAME para Windows não é compilado com SDL. Para obter mais informações consulte o capítulo [Compilando o MAME](#).

¹² Nas versões SDL a opção `auto` será `sdl`.

-mouseprovider <módulo>

Escolhe como o MAME lidará com a entrada do mouse. No Windows, auto tentará o **rawinput**, caso contrário retornará para **dinput**. Nas versões SDL, o auto será predefinido como **sdl**. O binário oficial do MAME para Windows não é compilado com SDL, sendo necessário compilar uma versão compatível para que a opção **sdl** funcione.

O valor predefinido é auto.

Exemplo:

```
mame indy_4610 -mouseprovider win32
```

Tabela 3: Opções compatíveis com a entrada do mouse separado por plataforma

Microsoft Windows	auto ¹³	rawinput	dinput	win32	sdl ¹⁴	none
SDL (macOS and Linux)	auto ¹⁵				sdl	none

Observações**Example:**

```
mame indy_4610 -mouseprovider win32
```

-lightgunprovider <módulo>

Escolhe como o MAME lidará com a arma de luz (*light gun*).

O valor predefinido é auto.

Exemplo:

```
mame lethalen -lightgunprovider x11
```

Tabela 4: Opções compatíveis com a entrada para a arma de luz separado por plataforma

Microsoft Windows	auto ¹⁶	rawinput	win32	sdl ¹⁷		none
macOS	auto ¹⁸			sdl		none
Linux	auto ¹⁸			sdl	x11	none

¹³ No Windows, auto tentará o rawinput, caso contrário, usa o dinput.

¹⁴ Para ter suporte SDL no Windows é preciso compilar o MAME com OSD=sdl. O binário oficial do MAME para Windows não é compilado com SDL. Para obter mais informações consulte o capítulo [Compilando o MAME](#).

¹⁵ Nas versões SDL a opção auto será sdl.

¹⁶ No Windows, o auto tentará o rawinput, caso contrário, tentará win32 ou none caso não encontre nenhum.

¹⁷ Para ter suporte SDL no Windows é preciso compilar o MAME com OSD=sdl. O binário oficial do MAME para Windows não é compilado com SDL. Para obter mais informações consulte o capítulo [Compilando o MAME](#).

¹⁸ Nas versões SDL a opção auto será sdl.

Observações

-joystickprovider <módulo>

Escolhe como o MAME lidará com a entrada do joystick ou outro controle.

O valor predefinido é auto.

Exemplo:

```
mame mk2 -joystickprovider winhybrid
```

Tabela 5: Opções compatíveis com a entrada do joystick separado por plataforma

Microsoft Windows	auto ¹⁹	winhybrid	dinput	xinput	sdlgame ²⁰	sdljoy ²⁰	none
SDL	auto ²¹				sdlgame	sdljoy	none

Observações

¹⁹ No Windows, a predefinição é winhybrid.

²⁰ Para ter suporte SDL no Windows é preciso compilar o MAME com OSD=sdl. O binário oficial do MAME para Windows não é compilado com SDL. Para obter mais informações consulte o capítulo [Compilando o MAME](#).

²¹ Nas versões SDL a opção auto será sdlgame.

winhybrid

Usa o XInput para os controles compatíveis e retornando para DirectInput para outros controles. Geralmente, oferece uma melhor experiência no Windows.

dinput

Usa o DirectInput para todos os controles. Pode ser útil caso queira usar mais de quatro controles XInput ao mesmo tempo. Observe que os controles **LT** e **RT** são combinados com o uso de controles XInput via DirectInput.

xinput

É compatível com até quatro controles XInput.

sdlgame

Usa a API do controle SDL para controles com o mapeamento do botão/eixo disponíveis, retorna para a API do joystick SDL nos outros controles. Fornece uma atribuição consistente dos botões, dos eixos e dos nomes dos controles mais populares. Use a [opção *controller_map*](#) para fornecer mapeamentos para os controles adicionais ou substituir os mapeamentos já incluídos.

sdljoy

Usa a API dos joystick em todos os controles de jogos.

none

Ignora todos os controles de jogos.

5.1.8 Opções de MIDI e rede

-midiprovider <módulo>

Escolhe como MAME se comunicará com os dispositivos e as aplicações MIDI (teclados de música e sintetizadores por exemplo). As opções compatíveis são `pm` para utilizar a biblioteca *PortMidi* ou `none` para desativar a entrada e a saída MIDI (os arquivos MIDI ainda podem ser reproduzidos).

O padrão é `auto` (utilizará o *PortMidi* caso esteja disponível).

Exemplo:

```
mame -midiprovider none dx100 -midiin canyon.mid
```

-listmidi

Exibe uma lista dos nomes dos dispositivos MIDI disponíveis para serem utilizados durante a emulação.

Exemplo:

```
mame -listmidi
MIDI output ports:
Microsoft MIDI Mapper (default)
CoolSoft MIDIMapper
Microsoft GS Wavetable Synth
VirtualMIDISynth #1
```

-midiin <nome do dispositivo>

Informe manualmente o dispositivo MIDI de entrada da sua preferência caso o seu computador ou sistema utilize mais de um. O comando apenas funciona nos sistemas compatíveis e que estejam funcionando com uma entrada MIDI. Consulte também a opção `-listslot` para identificar o nome do slot. Caso o nome do dispositivo tenha espaço, use aspas.

Exemplo:

```
mame sistema -nome-do-slot -midiin "nome do dispositivo ou arquivo midi"
```

-midiout <nome do dispositivo>

Informe manualmente o dispositivo MIDI de saída da sua preferência caso o seu computador ou sistema utilize mais de um. O comando apenas funciona nos sistemas compatíveis e que estejam funcionando com uma entrada MIDI. Consulte também a opção `-listslot` para identificar o nome do slot. Caso o nome do dispositivo tenha espaço, use aspas.

Exemplo:

```
mame sistema -nome-do-slot -midiout "nome do dispositivo"
```

-listnetwork

Lista os adaptadores de redes que estiverem disponíveis para serem utilizados com a emulação.

Exemplo:

```
No Windows
mame -listnetwork
    Available network adapters:
    Conexão Local

No Linux
mame -listnetwork
    Available network adapters:
    TAP/TUN Device
```

Nota: No Windows, é necessário instalar o [OpenVPN](#) mais recente para que o MAME possa ver os adaptadores de rede.

-networkprovider <módulo>

Escolhe como o MAME oferecerá comunicação para as interfaces de rede emuladas orientadas a pacotes (placas Ethernet por exemplo). As opções suportadas são `taptun` para usar o "`TUN/TAP`", `TAP-Windows` ou similar, `pcap` para usar uma biblioteca `pcap` ou `none` para desativar a comunicação nas interfaces emuladas de rede. As opções disponíveis dependem do seu sistema operacional. No Windows e no Linux as opções disponíveis são `taptun` e `none`, no macOS as opções disponíveis são `pcap` e `none`.

O padrão é `auto` que usará a opção `taptun` caso esteja disponível ou retorna para `pcap`.

Exemplo:

```
mame -networkprovider pcap apple2ee -s13 uthernet
```

5.1.9 Opções de saída das notificações de tela

-output

Escolhe como o MAME lidará com o processamento das notificações da saída. É utilizado para conectar saídas externas como uma luz de LED dos botões iluminados de start para os jogadores 1 e 2 em determinados sistemas arcade, assim como qualquer outro tipo de iluminação externa caso esteja disponível.

Exemplo:

```
mame galaxian -output console  
lamp0 = 1  
lamp1 = 1  
lamp0 = 0  
lamp1 = 0
```

Tão logo um crédito seja inserido e se for o caso do botão do Jogador 1 (1P) começar a piscar os valores começaram a alternar na tela.

Aqui no caso do sistema "Breakers":

Exemplo:

```
mame breakers -output console  
digit1 = 63  
digit2 = 63  
digit3 = 63  
digit4 = 63
```

Cada sistema terá a sua própria característica.

É possível escolher entre: auto, none, console ou network.

O valor predefinido para a porta de rede é **8000**.

5.1.10 Opções para a configuração

-[no]readconfig / -[no]rc

Ativa ou não a leitura dos arquivos de configuração, é predefinido que todos os arquivos de configuração sejam lidos em sequência, como mostra a lista abaixo:

- **mame.ini**
- **<meumame>.ini**

Caso o arquivo binário do MAME seja renomeado para **mame060.exe**, então o MAME carregará o arquivo **mame060.ini**.

- **debug.ini**

Caso o depurador esteja ativado.

- **<driver>.ini**

Com base no nome do arquivo fonte ou driver.

- **vertical.ini**

Para sistemas com orientação vertical do monitor.

- **horizont.ini**

Para sistemas com orientação horizontal do monitor.

- **vector.ini**

Para sistemas com vetores apenas.

- **<parent>.ini**

Para clones apenas, poderá ser chamado de forma recursiva.

- **<nome-do-driver-do-sistema>.ini**

Para que as opções sejam aplicadas apenas no driver do sistema, para saber qual o nome do driver de um determinado sistema faça o comando:

Exemplo:

```
mame -ls sf2
sf2                capcom/cps1.cpp
```

O nome do driver é **cps1** (sem a extensão .cpp), logo, o arquivo deve ser nomeado como **cps1.ini**.

Veja mais em [Diversos arquivos de configuração](#) para mais detalhes.

Exemplo:

```
mame sf2ce -norc -ctrlr sf2
```

As configurações nos INIs posteriores substituem aquelas dos INIs anteriores. Então, por exemplo, caso queira desativar os efeitos de sobreposição nos sistemas vetoriais, é possível criar um arquivo **vector.ini** com a linha **effect none** nele, ele irá sobrescrever qualquer valor de efeito existente no seu **mame.ini**.

O valor predefinido é Ligado (**-readconfig**).

-[no]writeconfig / -[no]wc

Grava as configurações feitas no driver do sistema num arquivo (driver).ini ao encerrar da emulação. O valor predefinido é Desligado (-nowriteconfig).

Exemplo:

```
mame sf2ce -wc -ctrlr sf2
```

5.1.11 Opções para a configuração dos principais diretórios

-homepath <caminho>

Define o caminho para onde os **plugins** Lua armazenarão os dados. O valor predefinido é '.' (no diretório raiz do MAME).

Exemplo:

```
mame -homepath D:\mame\lua
```

-rompath / -rp / -biospath / -bp <caminho>

Define o caminho completo para encontrar imagens ROM, disco rígido, fita cassete, etc. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é roms (isto é, um diretório chamado **roms** no diretório raiz do MAME).

Exemplo:

```
mame -rompath D:\mame\roms;D:\MSX\floppy;D:\MSX\cass
```

-hashpath / -hash_directory / -hash <caminho>

Define o caminho completo para a pasta com os arquivos **hash** que é usado pela *lista de software* no gerenciador de arquivos. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é hash (isto é, um diretório chamado **hash** no diretório raiz do MAME).

Exemplo:

```
mame -hashpath D:\mame\hash;D:\roms\softlists
```

-samplepath / -sp <caminho>

Define o caminho completo para os arquivos de amostras (samples). Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é samples (isto é, um diretório chamado **samples** no diretório raiz do MAME).

Exemplo:

```
mame -samplepath D:\mame\samples;D:\roms\samples
```

-artpath <caminho>

Define o caminho completo para os arquivos com as ilustrações gráficas (*artworks*) dos sistemas. Essas ilustrações são imagens que cobrem o fundo da tela e oferecem alguns efeitos interessantes. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é artwork (isto é, um diretório chamado **artwork** no diretório raiz do MAME).

Exemplo:

```
mame -artpath D:\mame\artwork;D:\emu\shared-artwork
```

-ctrlrpath <caminho>

Define um ou mais caminhos para os arquivos de configuração dos controles. Mais de um caminho pode ser definido desde que estejam separados por ponto e vírgula. É usado em conjunto com a opção `-ctrlr`.

O valor predefinido é `ctrlr` (isto é, um diretório chamado **ctrlr** no diretório raiz do MAME).

Exemplo:

```
mame -ctrlrpath D:\mame\ctrlr;D:\emu\meus_controles
```

-inipath <caminho>

Define um ou mais caminhos onde os arquivos `.ini` possam ser encontrados. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula.

- No Windows a predefinição é `.;ini;ini/presets`, traduzindo, a primeira pesquisa é feita no diretório atual, a segunda no diretório **ini** e finalmente no diretório **presets** dentro do diretório **ini**.
- No macOS a predefinição é `$HOME/Library/Application Support/mame;$HOME/.mame;.;ini`, traduzindo, pesquisa no diretório **mame** dentro do diretório **Application Support** do usuário atual, depois no diretório **.mame** dentro do diretório **home** do usuário atual, depois no diretório raiz e então no diretório **ini**.
- Em outras plataformas onde se incluem o Linux, a predefinição é `$HOME/.mame;.;ini`, traduzindo, procura pelo diretório **.mame** no diretório **home** do usuário atual, seguido pelo diretório raiz e finalmente no diretório **ini**.

```
mame -inipath D:\mame\ini
```

-fontpath <caminho>

Define um ou mais caminhos onde os arquivos de fonte `.bdf` (*Adobe Glyph Bitmap Distribution Format*) possam ser encontrados. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é `.` (isto é, no diretório raiz do MAME).

Exemplo:

```
mame -fontpath D:\mame\;D:\emu\fontes
```

-cheatpath <caminho>

Define o caminho completo para os arquivos de trapaça em formato `.xml`. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é `cheat` (isto é, uma pasta chamada **cheat**, localizada no diretório raiz do MAME).

Exemplo:

```
mame -cheatpath D:\mame\cheat;D:\emu\trapaças
```

-crosshairpath <caminho>

Define um ou mais caminhos onde os arquivos de mira **crosshair** possam ser encontrados. Mais de um caminho podem ser definidos desde que estejam separados por ponto e vírgula. O valor predefinido é `crosshair` (isto é, um diretório chamado **crosshair** no diretório raiz do MAME). Caso uma mira seja definida no menu, o MAME procurará por `nomedosistema\cross#.png`, em seguida no **crosshairpath** especificado onde `#` é o número do jogador.

Caso nenhuma mira seja definida, o MAME usará a sua própria.

Exemplo:

```
mame -crosshairpath D:\mame\crsshair;D:\emu\miras
```

-pluginpath <caminho>

Define um ou mais caminhos onde possam ser encontrados os plug-ins do Lua para o MAME. O valor predefinido é `plugins` (isto é, um diretório chamado **plugins** no diretório raiz do MAME).

Exemplo:

```
mame -pluginpath D:\mame\plugins;D:\emu\lua
```

-languagepath <caminho>

Define um ou mais caminhos onde possam ser encontrados os arquivos de tradução que o MAME usa na Interface do Usuário. O valor predefinido é `language` (isto é, um diretório chamado **language** no diretório raiz do MAME).

Exemplo:

```
mame -languagepath D:\mame\language;D:\emu\idiomas
```

-swpath <caminho>

Define um ou mais caminhos onde possam ser encontrados arquivos avulsos dos programas (rom, iso, etc.). O valor predefinido é `software` (isto é, um diretório chamado **software** no diretório raiz do MAME).

Exemplo:

```
mame -swpath D:\mame\floppy;D:\emu\discos
```

-cfg_directory <caminho>

Define o diretório onde os arquivos de configuração são armazenados. Os arquivos de configuração armazenam as customizações feitas pelo usuário e são lidas na inicialização do MAME ou de um sistema emulado, depois quaisquer alterações são salvas ao encerrar o MAME.

Os arquivos de configuração preservam as configurações da ordem dos botões do seu controle ou joystick, configurações das chaves DIP, informações da contabilidade do sistema e a organização das janelas do depurador.

O valor predefinido é `cfg` (isto é, um diretório com o nome **cfg** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -cfg_directory D:\mame\cfg
```

-nvram_directory <caminho>

Define o diretório onde os arquivos **NVRAM** são armazenados. Os arquivos **NVRAM** armazenam o conteúdo da **EEPROM**, memória RAM não volátil (NVRAM) e informações de outros dispositivos programáveis que fazem uso deste tipo de memória. As informações são lidas no início da emulação e gravadas ao encerrar.

O valor predefinido é `nvram` (isto é, um diretório com nome "nvram" no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -nvram_directory D:\mame\nvram
```

-input_directory <caminho>

Define o diretório onde os arquivos de gravação da entrada são armazenados. As gravações da entrada são criadas através da opção **-record** e reproduzidas através da opção **-playback**. A opção grava todos os comando e acionamentos de botões que forem feitos durante a operação do sistema.

O valor predefinido é `inp` (ou seja, um diretório de nome **inp** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -input_directory D:\mame\inp
```

-state_directory <caminho>

Define o diretório onde os arquivos de gravação de estado são armazenados. Os arquivos de estado são lidos e gravados mediante a solicitação do usuário ou ao usar a opção **-autosave**.

O valor predefinido é `sta` (isto é, um diretório de nome **sta** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -state_directory D:\mame\sta
```

-snapshot_directory <caminho>

Define o diretório onde os arquivos de instantâneos da tela são armazenados quando solicitado pelo usuário.

O valor predefinido é `snap` (isto é, um diretório chamado **snap** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -snapshot_directory D:\mame\snap
```

-diff_directory <caminho>

Define o diretório onde os arquivos de diferencial do disco rígido são armazenados. Os arquivos de diferencial armazenam qualquer dado que é escrito de volta na imagem do disco, isso serve para preservar a imagem de disco original. Os arquivos são criados no início da emulação com uma imagem compactada do disco rígido.

O valor predefinido é diff (isto é, um diretório chamado **diff** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -diff_directory D:\mame\diff
```

-comment_directory <caminho>

Define o diretório onde os arquivos de comentário do depurador são armazenados. Os arquivos de comentário do depurador são escritos pelo depurador quando comentários são adicionados num sistema desmontado (disassembly).

O valor predefinido é comments (isto é, um diretório chamado **comments** no diretório raiz do MAME). Caso este diretório não exista, ele será criado automaticamente.

Exemplo:

```
mame -comment_directory D:\mame\comments
```

-share_directory <caminho>

Define o diretório que será compartilhado com o sistema ou o driver que está sendo emulado. Por exemplo, no caso de um sistema operacional compatível, os arquivos que forem colocados neste diretório será compartilhado com o host emulado.

Exemplo:

```
mame -share_directory D:\mame\share
```

5.1.12 Opções para a gravação e a reprodução do estado da emulação

-[no]rewind

Quando ativo e a emulação for paralisada, automaticamente é salvo o estado da condição da memória toda a vez que um quadro for avançado. O rebobinamento das condições de estado que foram salvas podem ser carregadas de forma consecutiva ao pressionar a tecla de atalho para rebobinar passo único (Shift Esquerdo + ~)².

O valor predefinido é Desligado (`-norewind`).

Caso o depurador esteja no estado *break*, a condição de estado atual é criada a cada *step in*, *step over* ou caso ocorra um *step out*. Nesse modo os estados salvos podem ser carregados e rebobinados executando o comando *rewind* ou *rw* no depurador.

Exemplo:

```
mame -norewind
```

-rewind_capacity <valor>

Define a capacidade de rebobinar em megabytes. É a quantidade total de memória que será usada para rebobinar os *savestates*. Quando a capacidade alcança o limite, os antigos *savestates* são apagados enquanto novos são capturados. Definindo uma capacidade menor do que o *savestate* atual, desativa o rebobinamento. Os valores negativos são automaticamente fixados em 0.

Exemplo:

```
mame -rewind_capacity 30
```

-statename <nome>

Descreve como o MAME deve armazenar os arquivos de estado salvos relativo ao caminho do *state_directory*. <nome> é uma *string* que fornece um modelo que será utilizado para gerar um nome de arquivo.

São disponibilizadas duas substituições simples: o caractere `/` representa o separador de caminho em qualquer plataforma de destino (até mesmo no Windows); a *string* `%g` representa o nome do driver do sistema atual.

O valor predefinido é `%g`, que cria uma pasta separada para cada sistema.

Em adição ao que foi dito acima, para os drivers que usem mídias diferentes, como cartões ou disquetes, é possível usar o indicador `%d_[media]`. Substitua `[media]` pelo comutador de mídia desejado.

Alguns exemplos:

- Caso use `mame robby -statename foo/%g%i` as capturas da tela serão salvos em **sta\foo\robby**.
- Caso use `mame nes -cart robby -statename %g/%d_cart` os instantâneos serão salvos em **sta\nes\robby**.
- Caso use `mame c64 -flop1 robby -statename %g/%d_flop1/%i` estes serão salvos como **sta\c64\robby\0000.png**.

² Até que o teclado **ABNT-2** seja mapeado pela equipe do MAMEDev, essa tecla fica do lado esquerdo da tecla 1, logo abaixo da tecla ESQ. (Nota do tradutor)

-state <slot>

Depois de iniciar um sistema determinado, fará com que o estado salvo no <slot> seja carregado imediatamente.

Exemplo:

```
mame -state 1
```

-[no]autosave

Quando ativado, cria automaticamente um arquivo com a condição atual do sistema ao encerrar o MAME e automaticamente tenta recarregá-lo caso o MAME inicie novamente com o mesmo sistema. A opção só funciona para os sistemas que sejam compatíveis com o salvamento do seu estado.

O valor predefinido é Desligado (-noautosave).

Exemplo:

```
mame -autosave
```

-playback / -pb <nome do arquivo>

Faz a reprodução de um arquivo de gravação. Esse recurso não funciona de maneira confiável com todos os sistemas, mas pode ser usado para assistir a uma sessão do jogo gravado anteriormente do início ao fim. Para tornar as coisas consistentes, apague os arquivos de configuração .cfg, NVRAM .nv e o cartão de memória. Consulte o comando *-record* para obter mais informações importantes.

O valor predefinido é NULO (sem reprodução).

Exemplo:

```
mame ssf2tu -playback perfect
```

Nota: Você pode ter problemas com a falta de sincronismo caso a configuração, a NVRAM, e o cartão de memória não coincidam com o original, inclusive caso seja utilizado uma versão do MAME muito diferente daquela usada na gravação. É recomendável que a configuração (.cfg), a NVRAM (.nv) ou o diretório com o nome do sistema dentro do diretório **nvr** sejam excluídos antes de iniciar uma gravação ou uma reprodução.

Aviso: Para que o playback funcione em alguns sistemas de alguns drivers, elas precisam da **NVRAM** como por exemplo a CPS1, a CPS2 e a CPS3, manter ou não o arquivo de configuração nestes casos não faz a menor diferença. Então caso você vá compartilhar a gravação com alguém, tenha certeza de enviar o arquivo **NVRAM** do sistema em questão.

Aviso: Em sistemas que não usam **NVRAM** como a pacman, mspacman e talvez outras, elas também perdem o sincronismo e algumas vezes criam anomalias (bugs) apenas durante a reprodução, neste caso apague o arquivo que mantém o registro do **high score** dentro do diretório **hi**. Caso você mantenha um registro de pontuações, faça um backup antes de apagar o arquivo.

-[no]exit_after_playback

O MAME encerra a emulação ao final do arquivo de reprodução caso seja usado em conjunto com a opção **-playback**. É predefinido que o MAME não encerre a emulação.

Exemplo:

```
mame ssf2tu -playback perfect -exit_after_playback
```

O valor predefinido é Desligado (**-noexit_after_playback**).

-record / -rec <nome do arquivo>

Faz a gravação de todos comandos feitos pelo usuários durante uma seção e define o nome do arquivo onde será registrado todos esses comandos durante uma seção. Esse recurso não funciona de forma confiável com todos os sistemas.

O valor predefinido é NULO (sem gravação).

Exemplo:

```
mame ssf2tu -rec perfect
```

Aviso: Em alguns sistemas como o **neogeo** por exemplo, é preciso excluir a **NVRAM** do sistema **ANTES** de iniciar uma gravação e **ANTES** de reproduzir com **-playback**, caso contrário, a reprodução pode iniciar num estágio diferente (na série *"The King of Fighters"* por exemplo) e fazendo com que a ação não bata com o que foi gravado ou até mesmo haja uma interrupção abrupta da reprodução muito antes do fim. Por exemplo, se for iniciar a gravação do sistema **kof2002**, dentro da pasta **NVRAM**, exclua a pasta **kof2002** ou uma pasta **kof2002_*** caso ela exista. Para obter mais informações consulte [Excluindo arquivos NVRAM \(script\)](#).

5.1.13 Opções para a gravação de áudio e vídeo

Há casos onde certos sistemas alternam a resolução da tela atrapalhando a gravação de vídeo, algumas gravações podem ficar com um tamanho de tela todo preto com um vídeo menor no meio ou em algum outro canto da tela, use essas duas opções caso isso aconteça, *-noswitchres* com *-snapsize*.

-mngwrite <nome do arquivo>.mng

Escreve cada quadro de vídeo num arquivo <nome do arquivo> no formato MNG, produzindo uma animação da sessão. Note que *-mngwrite* só grava quadros de vídeo, não grava qualquer áudio, use a opção *-wavwrite* para gravar o áudio e posteriormente use uma ferramenta de edição de áudio qualquer para unir os dois, ou use *-aviwrite* para gravar áudio e vídeo num único arquivo.

O valor predefinido é NULO (sem gravação).

Exemplo:

```
mame ssf2tu -mngwrite ssf2tu-video.mng
```

-aviwrite <nome do arquivo>.avi

Grava todos os dados de áudio e vídeo em formato AVI sem compressão, note que a taxa de quadros e a resolução são sempre fixas. Vídeos sem compressão ocupam muito espaço assim como, para que a gravação ocorra sem problemas é necessário um HDD rápido. Para alterar a resolução do arquivo que será gravado, consulte a opção *-snapsize*.

Talvez seja mais prático gravar os seus comandos com *-record* e depois fazer o vídeo com *-aviwrite* combinado com *-playback* e *-exit_after_playback*.

O valor predefinido é NULO (sem gravação).

Exemplo:

```
mame ssf2tu -pb perfect -exit_after_playback -aviwrite ssf2tu.avi
```

-wavwrite <nome do arquivo>.wav

Grava apenas o áudio da seção em formato PCM 16 bits. Para gravar com uma taxa de amostragem diferente da predefinida (**48000 Hz**), consulte a opção *-samplerate*.

O valor predefinido é NULO (sem gravação).

Exemplo:

```
mame ssf2tu -wavwrite audio.wav
```

5.1.14 Opções para a captura da tela

-snapname <nome>

Descreve como MAME deve nomear arquivos de instantâneos de tela. <nome> será o guia que o MAME usará para nomear o arquivo.

São disponibilizadas três substituições simples:

- O caractere /

Usado como separador de caminho em qualquer plataforma inclusive no Windows.

- Especificador de conversão %g

Converte %g para o nome do driver que for usado.

- Especificador de conversão %i

Cria arquivos iniciando com nome 0000 e os incrementa enquanto novos instantâneos forem sendo criados, O MAME incrementará o valor de %i para o próximo vazio, caso ele seja omitido, os instantâneos existentes com o mesmo nome serão gravados por cima.

O valor predefinido é %g/%i.

Para os drivers que usam mídias diferentes, como cartões ou disquetes, também é possível usar %d_[media]. Substitua [media] pelo dispositivo que deseja usar.

Alguns exemplos:

- Caso use `mame robby -snapname foo/%g%i` os instantâneos serão salvos como `snaps\foo\robby0000.png`, `snaps\foo\robby0001.png` e assim por diante.
- Caso use `mame nes -cart robby -snapname %g/%d_cart` os instantâneos serão salvos como `snaps\nes\robby.png`.
- No caso deste outro exemplo, `mame c64 -flop1 robby -snapname %g/%d_flop1/%i` estes serão salvos como `snaps\c64\robby\0000.png`.

-snapsize <largura> x <altura>

Define um tamanho fixo para os instantâneos e vídeos. É predefinido que o MAME criará instantâneos, assim como os vídeos, na resolução original do sistema em pixels brutos. Caso use esta opção, o MAME criará instantâneos e vídeos no tamanho determinado, com filtro bilinear (filtro de embaçamento de pixels) aplicado no resultado final. Observe que ao definir este tamanho a tela não gira automaticamente caso o sistema seja orientado verticalmente.

O valor predefinido é auto.

Exemplo:

```
mame ssf2tu -snapsize 640x480
```


-snapview <tipo>

Define a visualização que será utilizada nas capturas da tela e para gravar vídeos.

É predefinido que ambos utilizem a primeira visualização que estiver disponível ou somente da primeira tela. Ao usar esta opção é possível alterar o comportamento predefinido da exibição e selecionar apenas a visualização que será aplicada em todos os instantâneos e vídeos.

Observe que o <tipo> não precisa ser o nome exato, em vez disso, o MAME selecionará a primeira exibição cujo nome corresponda com o que for definido através do <tipo>, supondo que o nome seja **Cabine Animada** basta usar **Cabine** ou **cabine**.

Por exemplo, -snapview native irá casar a visualização *Nativa em (15:14)* ainda que o nome não combine perfeitamente. O <tipo> também pode ser "auto" onde será escolhida a primeira exibição de todas que existirem.

Nos casos onde você utiliza uma visualização com mais de uma opção ou que tenha nomes estranhos, nomes com caracteres não ASCII ou algo do tipo:

- XXYYZZ_01
- XXYYZZ_02
- XXYYZZ_03

Use -snapview XXYYZZ_03 para definir exatamente a visualização desejada na sua captura.

O valor predefinido é internal.

Exemplo:

```
mame ssf2tu -snapview pixel
```

-[no]snapbilinear

Especifique se o instantâneo ou vídeo deve ter filtragem bilinear aplicada, o filtro bilinear aplica um leve efeito de embaçamento ou suavização à tela, amenizando um pouco o serrilhado nos contornos gráficos e suavizando a tela do sistema. Desligar essa opção pode fazer a diferença melhorando o desempenho durante a gravação do vídeo.

O valor predefinido é Ligado (-snapbilinear).

Exemplo:

```
mame ssf2tu -nosnapbilinear
```

5.1.15 Opções relacionadas ao desempenho e a velocidade da emulação

-[no]autoframeskip / -[no]afs

Para que se mantenha a velocidade máxima de uma emulação, ajusta dinamicamente no sistema emulado a quantidade de quadros que serão pulados. Ativando esta opção ela se sobrepõem ao que for definido em **-frameskip** descrito logo abaixo.

O valor predefinido é Desligado (**-noautoframeskip**).

Exemplo:

```
mame gradius4 -autoframeskip
```

-frameskip / -fs <quantidade>

Determina a quantidade de quadros que são ignorados. Ela elimina cerca de 12 quadros enquanto estiver sendo executado. Caso seja definido **-frameskip 2** o MAME então exibirá 10 de cada 12 quadros por exemplo.

Ao ignorar estes quadros, pode ser que se atinja a velocidade nativa do sistema emulado sem que haja sobrecarga no seu computador ainda que ele não tenha um grande poder de processamento.

O valor predefinido é não ignorar nenhum quadro (**-frameskip 0**).

Exemplo:

```
mame gradius4 -frameskip 2
```

-seconds_to_run / -str <segundos>

Este comando pode ser usado para realizar um teste de velocidade de forma automatizada. O comando diz ao MAME para interromper a emulação depois de alguns segundos. Ao combinar com outras opções fixas de linha de comando é possível definir um ambiente para realizar testes de desempenho. Ao encerrar, a opção **-str** fará uma captura da tela com o nome determinado pela opção **-snapname**.

O comando diz ao MAME para interromper a emulação depois de um tempo determinado, o tempo em questão não é o tempo real e sim o tempo interno da emulação, assim, caso seja definido 30 segundos, pode ser que dependendo do sistema que esteja sendo emulado, a parada só venha a acontecer depois de algum tempo.

Este comando também é útil para a realização de benchmarks e testes de automação. Ao combinar esta opção com algumas outras, é possível construir uma estrutura de testes de desempenho do MAME. Adicionalmente a opção **-str**, faz também que ao final do tempo seja criado uma captura da tela determinado pela opção **-snapname** dentro da pasta dos *instantâneos*.

Exemplo:

```
mame ssf2tu -str 60
```

-[no]throttle

Ativa ou não a função de controle de velocidade do emulador⁴. Ao ativar esta opção, o MAME tenta manter o sistema rodando em sua velocidade nativa, com a opção desativada a emulação é executada na velocidade mais rápida possível. Dependendo das características do sistema emulado, o desempenho final pode limitada pelo seu processador, placa de vídeo ou até mesmo pelo desempenho final da sua memória.

O valor predefinido é Ligado (-throttle).

Exemplo:

```
mame pacman -nothrottle
```

-[no]sleep

Quando utilizada em conjunto com -throttle o MAME elimina os processos não utilizados durante a limitação de velocidade da emulação melhorando o rendimento de processamento. Em outras palavras, permite que outros programas tenham mais tempo de CPU assumindo que a emulação não esteja consumindo 100% dos recursos do processador. Esta opção pode causar uma certa intermitência no desempenho caso outros programas que também demandem processamento estejam rodando junto com o MAME.

O valor predefinido é Ligado (-sleep).

Exemplo:

```
mame ssf2tu -nosleep
```

-speed <fator>

Muda a maneira que o MAME controla a velocidade da emulação de maneira que seja possível que o sistema emulado rode em múltiplos da sua velocidade original.

Um <fator> 1.0 significa rodar o sistema em velocidade normal. Já um fator 0.5 significa rodar o sistema na metade da velocidade normal e um <fator> 2.0 significa rodar o sistema 2x acima da sua velocidade normal. Note que ao mudar este valor a velocidade de execução do áudio irá mudar proporcionalmente também.

A resolução interna da fração são dois pontos decimais, logo o valor 1.002 será arredondado para 1.0.

O valor predefinido é 1.0.

Exemplo:

```
mame ssf2tu -speed 1.25
```

Quando utilizado em conjunto com -rec é possível colocar o sistema em velocidade lenta como -speed 0.3 enquanto grava. Ao terminar, a reprodução com a opção -pb ocorrerá em velocidade normal, exemplo:

Exemplo:

```
mame ssf2tu -rec perfect -speed 0.3 -sound none
```

A opção -sound none serve para eliminar o áudio durante a gravação em câmera lenta. Para mais informações, consulte [slowmomame](#).

⁴ O termo *throttle* no Inglês significa *parar/interromper a respiração através da esganadura da garganta*. O termo então significa manter o controle do fluxo da velocidade. Em Inglês este termo também é usado para descrever o acelerador de um veículo, onde o *acelerador* faz o controle da velocidade do mesmo. (Nota do tradutor)

-[no]refreshspeed / -[no]rs

Permite ao MAME ajustar a velocidade da emulação para que a taxa de atualização da primeira tela emulada não exceda o menor valor da taxa de atualização da tela de qualquer um dos monitores do seu sistema. Visando evitar cortes no áudio ou efeitos colaterais indesejáveis, o MAME irá reduzir a velocidade da emulação para 99% em casos onde por exemplo, um monitor que funcione nativamente a 60 Hz e o sistema emulado rode a 60.6 Hz.

Utilize esta opção caso note pequenas travadas de tela durante cenas de movimentação horizontal ou vertical.

O valor predefinido é Desligado (`-norefreshspeed`).

Exemplo:

```
mame ssf2tu -refreshspeed
```

-numprocessors / -np <auto|valor>

Define a quantidade de núcleos do processador que serão utilizados. A opção auto usará a quantidade de núcleos informada pelo seu sistema ou pela variável de ambiente **OSDPROCESSORS**. Este valor é limitado internamente para quatro vezes o número dos processadores informado pelo seu sistema.

O valor predefinido é auto.

Exemplo:

```
mame ssf2tu -numprocessors 2
```

-bench <n>

Define a quantidade de segundos de emulação em <n> usado para teste de desempenho, o comando é um atalho com comando abaixo:

-str <n> -video none -sound none -nothrottle

Exemplo:

```
mame ssf2tu -bench 300
```

-[no]lowlatency

Diz ao MAME para desenhar um novo quadro antes de controlar a velocidade de emulação (*throttling*) visando reduzir o atraso (latência) de resposta da entrada. Esta opção é particularmente efetiva com telas com variação em sua taxa de atualização (Variable Refresh Rate).

Esta opção pode causar um efeito colateral de despassamento ou problemas com o sequenciamento dos quadros gerando instabilidades (especialmente em sistemas mais recentes com base 3D ou dependentes do 3D, assim como sistemas onde rodam um software similar ao sistema operacional).

O valor predefinido é `-nolowlatency`.

Exemplo:

```
mame bgaregga -lowlatency
```

5.1.16 Opções para a rotação da tela

-[no]rotate

Gira a tela para corresponder ao seu estado normal do sistema (horizontal / vertical). Isso garante que os sistemas vertical e horizontalmente orientados sejam exibidos corretamente sem que haja a necessidade de girar fisicamente a sua tela. Caso queira manter a disposição da tela como ela é no arcade original, mantenha esta opção **DESLIGADA**.

O valor predefinido é Ligado (-rotate).

Exemplo:

```
mame pacman -norotate
```

-[no]ror -[no]rol

Rotacione a tela do sistema para a direita -ror ou para a esquerda -rol em relação ao seu estado normal caso -rotate seja definido ou seu estado nativo caso -norotate seja definido.

O valor predefinido para ambas é Desligado (-noror** **-norol).

Exemplo:

```
mame pacman -ror  
mame pacman -rol
```

-[no]autoror -[no]autorol

Essas opções são projetadas para uso com telas giratórias que giram apenas numa única direção. Caso a tela gire somente no sentido horário, use o comando -autorol para garantir que o sistema encha a tela horizontalmente ou verticalmente numa das direções desejadas. Caso a sua tela gire somente no sentido anti-horário, use -autoror.

Exemplo:

```
mame pacman -autoror  
mame pacman -autorol
```

-[no]flipx -[no]flipy

Espelhe a tela do sistema horizontalmente -flipx ou verticalmente -flipy. As inversões são aplicadas depois que as opções de rotação -rotate e rolagem -ror/-rol forem aplicadas.

O valor predefinido para ambas as opções é Desligado (-noflipx -noflipy).

Exemplo:

```
mame pacman -flipx  
mame pacman -flipy
```

5.1.17 Opções para a configuração de vídeo

-video < bgfx | gdi | d3d | opengl | soft | accel | none >

Define qual tipo de saída de vídeo usar. As opções aqui descritas dependem do sistema operacional utilizado e se a versão do MAME é uma versão SDL ou não.

Opções geralmente disponíveis:

- **bgfx** (preferível)

Determina o novo renderizador acelerado por hardware, utilize esta opção caso a sua placa de vídeo seja compatível.

- **opengl**

Faz a renderização do vídeo usando [OpenGL](#), use em sistemas Windows compatíveis quando por algum motivo as outras opções causarem problemas.

Em sistemas não Windows, essa é a opção predefinida para a renderização da tela e para fazer a aceleração via hardware, caso seja compatível com o seu sistema operacional e pela sua placa de vídeo.

- **none**

Não exibe janelas e nem mostra nada na tela. É principalmente utilizado para realizar testes de desempenho (*benchmarks*) usando apenas a CPU.

No Windows:

- **gdi**

Diz ao MAME para renderizar o vídeo usando funções gráficas mais antigas do Windows. Em termos de desempenho é a opção mais lenta porém a mais compatível com as versões os sistemas Windows mais antigos.

- **d3d** (obsoleto)

Diz ao MAME para renderizar a tela com o **Direct3D**. Isso produz uma saída com uma melhor qualidade se comparada com a opção que o **gdi** assim como permite opções adicionais de renderização da tela e aceleração gráfica via hardware.

É recomendável ter uma placa de vídeo mediana (2002+) ou uma placa de vídeo Intel embutida modelo *HD3000* ou superior.

Nota: Esta opção já é obsoleta para um hardware mais moderno, prefira a opção **bgfx** usando o [bgfx_backend](#) com **d3d11** ou a versão mais recente. Caso a sua placa seja compatível, use **vulkan** para obter o melhor desempenho possível (isso também depende da compatibilidade relacionada ao desenvolvimento do MAME, do driver da sua placa de vídeo e do sistema operacional usado).

Em outras plataformas (incluindo o SDL no Windows):

- **accel**

Diz ao MAME para, se possível, processar o vídeo usando a aceleração 2D do SDL.

- **soft**

Faz com que a tela seja renderizada através de software. Por não usar nenhum tipo de aceleração de vídeo, o desempenho da emulação pode ser penalizada, porém favorecendo uma melhor compatibilidade em qualquer plataforma.

- **Predefinições até a versão 0.240:**

No Windows é d3d.

No macOS é opengl pois é quase certo que exista uma pilha OpenGL compatível.

No Linux é opengl.

O valor predefinido para todos os outros sistemas é soft.

Exemplo:

```
mame ssf2tu -video bgfx
```

- **Predefinições depois da versão 0.241:**

Todos os sistemas passam a utilizar bgfx²².

-numscreens <quantidade>

Diz ao MAME quantas telas devem ser criadas. Para a maioria dos sistemas só existe uma, porém alguns sistemas originalmente usavam mais de uma (*como os sistemas Darius e os sistemas Arcade PlayChoice-10 por exemplo*). Cada tela (até 4), possuem as suas próprias configurações, taxa de proporção de tela, resolução e exibição, que podem ser definidas usando as opções abaixo.

O valor predefinido é 1.

Exemplo:

```
mame darius -numscreens 3
mame pc_cntra -numscreens 2
```

-[no]window / -[no]w

Inicia a tela do MAME numa janela em vez da tela inteira.

O valor predefinido é Desligado (-nowindow).

Exemplo:

```
mame ssf2tu -window
```

²² <https://github.com/mamedev/mame/commit/eee7d7d155d527996890a90c952f9856675c965d>

-[no]maximize / -[no]max

Controla o tamanho inicial da janela. Caso esta opção seja ativada, durante a inicialização do MAME a janela será exibida com o maior tamanho possível. Com a opção desligada, a emulação terá início com o tamanho aproximado ao tamanho original do sistema, a sua escala será em apenas um eixo quando os pixels não quadrados estiverem em uso. Esta opção apenas surte efeito quando a opção **-window** é utilizada.

O valor predefinido é Ligado (**-maximize**).

Exemplo:

```
mame ssf2tu -window -maximize
```

-[no]keepaspect / -[no]ka

Faz com que a proporção de tela seja mantida. Quando essa opção está ativa, a taxa de proporção adequada da tela do sistema é aplicada, geralmente 4:3 ou 3:4 para monitores CRT dependendo da orientação, no entanto muitas outras proporções de tela já foram usadas como 3:2 (Nintendo Game Boy), 5:4 para algumas workstation assim como vários outros.

Caso a tela que estiver sendo emulada ou ilustração não preencher toda a tela por completo, a imagem será centralizada com barras pretas adicionadas as laterais conforme a necessidade para ocupar os espaços não utilizados, sejam eles em cima ou em baixo assim como na esquerda ou na direita.

Ao desativar essa opção a tela ou ilustração poderá ser esticada livremente para preencher os espaços vazios no modo janela. Em tela cheia a imagem ficará distorcida e fora das proporções.

Quando essa opção estiver ativa no Windows e o MAME estiver em modo janela, a proporção de tela será mantido mesmo que a janela seja redimensionada para diferente tamanhos, caso mantenha a tecla **Control** ou **Ctrl** pressionada durante redimensionamento da janela, a proporção será mantida.

O valor predefinido é Ligado (**-keepaspect**).

Exemplo:

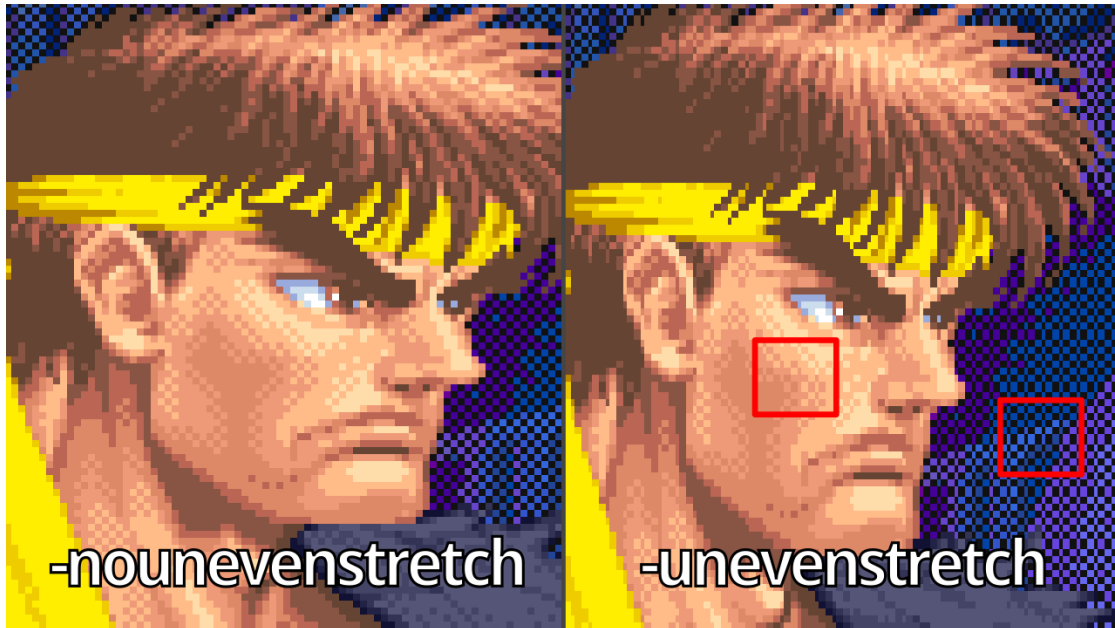
```
mame ssf2tu -ka
```

A equipe do MAME, sugere veementemente que se mantenha esta opção ativada. Esticando a tela do sistema além da proporção original vai causar distorções na aparência do sistema que vai muito além da capacidade de reparo dos filtros internos do MAME.

-[no]unevenstretch / -[no]ues

Permite que valores não inteiros possam ser usados para o redimensionamento da tela, isso faz com que a imagem possa ter uma forma mais distorcida ou esticada para que ela preencha toda a tela, porém há um preço a ser pago.

O uso de valores não inteiros geram uma interferência chamada **aliasing** nos pixels²⁴²⁵. Imagine o mapa de um jogo feito de linhas retas com 1 pixel de largura, quando ocorre o "aliasing" a linha que originalmente era feita com 1 pixel de largura passa a ter 2 pixels ou mais, essa interferência cria pixels aonde antes não existiam gerando distorções em **todos os pixels**. Abaixo um exemplo com destaque nas regiões marcadas com vermelho, porém nota-se que o problema afeta toda a imagem.



Atualmente as pessoas sentem a necessidade de preencher toda a tela de uma TV 16:9 com gráficos feitos para 4:3 ainda que isso gere distorções ao custo da desproporção dos gráficos.

Este é um assunto bem complexo pois, apesar de todos os pixels do lado esquerdo estarem com os quadrados perfeitos, o que significa uma proporção dos pixels de 1:1, também conhecido como **pixel perfect**, a imagem está com as suas proporções erradas, na época os gráficos não foram desenvolvidos com os pixels no formato de um quadrado perfeito e sim para terem 1 pixel mais alto visando as telas CRT 4:3 da época como mostra a imagem do lado direito.

Assim, apesar dos pixels estarem distorcidos na imagem da direita a proporção dos gráficos está correta! Ao mesmo tempo que apesar dos pixels estarem perfeitos do lado esquerdo a proporção do gráfico está errada.

Com esta opção é possível preencher a tela da sua TV 16:9 com gráficos desenvolvidos para uma tela 4:3 ao custo de distorções nos gráficos e isso fica pior ainda com textos.

Consulte também *-aspect*, *-keepaspect* e *-prescale*.

O valor predefinido é Ligado (-unevenstretch).

²⁴ <https://en.wikipedia.org/wiki/Aliasing>

²⁵ https://en.wikipedia.org/wiki/Spatial_anti-aliasing

Exemplo:

```
mame ssf2tu -nounevenstretch
```

-[no]unevenstretchx / -[no]uesx

Permite que a relação de aspecto da tela seja desigual e que a tela ou janela possa ser preenchida (esticada) apenas na horizontal.

O valor predefinido é Ligado (-unevenstretchx).

Exemplo:

```
mame ssf2tu -uesx
```

-[no]unevenstretchy / -[no]uesy

Permite que a relação de aspecto da tela seja desigual e que a tela ou janela possa ser preenchida (esticada) apenas na vertical.

O valor predefinido é Ligado (-unevenstretchy).

Exemplo:

```
mame ssf2tu -uesy
```

-[no]autostretchxy / -[no]asxy

Aplica a opção **-unevenstretchx/y** automaticamente com base na orientação nativa da fonte.

O valor predefinido é Desligado (-noautostretchxy).

Exemplo:

```
mame ssf2tu -asxy
```

-[no]intoverscan / -[no]ios

Permite que a imagem passe dos limites da tela (overscan) de alvos inteiros e dimensionáveis.

O valor predefinido é Desligado (-nointoverscan).

Exemplo:

```
mame ssf2tu -ios
```

-[no]intscalex / -[no]sx <fator>

Define o fator da escala para o preenchimento e a aproximação (zoom) da tela na horizontal, não causa aliasing nos pixels quando usado sozinho ou até o fator **4.0**. Causa aliasing mínimo nos pixels quando utilizado em conjunto com intscaley

O valor predefinido é 0.0 (-nointscalex 0.0).

Exemplo:

```
mame ssf2tu -sx 1.0  
mame ssf2tu -nowindow -ka -sx 5.0 -sy 5.0
```

-[no]intscaley / -[no]sy <fator>

Define o fator da escala para o preenchimento e a aproximação (zoom) da tela na vertical, não causa aliasing nos pixels quando usado sozinho ou até o fator **4.0**. Causa aliasing mínimo nos pixels quando utilizado em conjunto com intscalex.

O valor predefinido é 0.0 (-nointscaley 0.0).

Exemplo:

```
mame ssf2tu -sy 1.0
mame ssf2tu -nowindow -ka -sx 5.0 -sy 5.0
```

-[no]waitvsync

Aguarda acabar o período de atualização da tela do monitor do seu computador antes de começar a desenhar na tela. Caso esta opção esteja desligada, o MAME só irá desenhar na tela quando o quadro estiver pronto, mesmo que seja durante o processo de atualização de tela. Isso pode causar artefato de *screen tearing*⁵.

O efeito "tearing" não é perceptível em todos os sistemas, porém algumas pessoas acham o efeito desagradável, algumas mais do que as outras.

Os efeitos colaterais de se ativar a opção -waitvsync podem variar dependendo da combinação usada em diferentes sistemas operacionais e drivers de vídeo.

No **Windows**, -waitvsync será bloqueado até o próximo apagamento de vídeo, permitindo que o MAME desenhe o próximo quadro, sincronizando a taxa de quadros do sistema emulado com a taxa de quadros nativa do monitor que estiver sendo usado no Windows, apenas ative esta opção caso esteja utilizando o modo janela. Em tela inteira esta opção só é necessária caso a opção -triplebuffer não remova o indesejado efeito "*tearing*", neste caso, tente usar as duas opções juntas -notriplebuffer -waitvsync. Note que a opção -waitvsync não vai funcionar em conjunto com a opção -video gdi.

No **macOS**, -waitvsync não é bloqueado, contudo o quadro completamente desenhado será exibido no próximo apagamento de vídeo (vblank). Isso quer dizer que caso um sistema emulado tenha uma taxa de quadros maior do que a do seu sistema (ou do seu monitor), haverá uma queda periódica na velocidade dos quadros de vídeo emulados resultando em pequenos travamentos durante as cenas com movimentos.

O valor predefinido é Desligado (-nowaitvsync).

Exemplo:

```
mame ssf2tu -waitvsync
```

O **MAME SDL** funcionará com essa opção em modo janela caso haja compatibilidade com o seu sistema operacional, da sua placa de vídeo e respectivos drivers.

Rode o **MAME SDL** com a opção -video opengl para aumentar as suas chances de sucesso.

⁵ Faz com que a metade da parte de cima da tela saia de sincronismo com a outra metade da parte de baixo da tela, surgindo um efeito ou um "*defeito*" onde cada metade se deslocam horizontalmente para lados opostos. (Nota do tradutor)

-[no]syncrefresh

Ativa o controle de velocidade da taxa de atualização do seu monitor. Isso significa que a taxa de atualização usada pelo sistema é ignorada, porém, o código responsável pelo som tentará manter o sincronismo com a taxa de atualização usada pelo sistema, assim haverá problemas com o som.

Esta opção foi pensada naqueles que modificaram as configurações da sua placa de vídeo, combinando uma opção a mais com as de atualização da tela. Esta opção não funciona com a opção `-video gdi`.

O valor predefinido é Desligado (`-nosyncrefresh`).

Exemplo:

```
mame mk -syncrefresh
```

Nota: O `syncrefresh` pode ser útil para as pessoas com display compatível com G-Sync ou FreeSync.

-prescale <fator>

Controla a proporcionalidade da grandeza do redimensionamento do vídeo antes da aplicação de filtros ou shaders. No ajuste mínimo a tela é renderizada no seu tamanho original antes de ser dimensionada. Com valores maiores a tela é expandida pelo fator definido em `<fator>`. Isso gera imagens menos borradas com a opção `-video d3d` ao custo da perda de algum desempenho.

Experimente `-prescale 4` ou valores maiores para amenizar um pouco as distorções causadas pela opção `-unevenstretch` e para reduzir o efeito "*blur*" ao custo de um aumento de processamento.

Os valores válidos são 1 (mínimo) e 8 (máximo).

O valor predefinido é 1.

Funciona com todos os modos de vídeo no Windows (bgfx, d3d, etc.), nas outras plataformas funciona **APENAS** naquelas que forem compatíveis com o OpenGL. Não funciona com filtros *GLSL*.

Exemplo:

```
mame ssf2tu -video d3d -prescale 3
```

-[no]filter / -[no]d3dfilter / -[no]flt

Ativa o filtro bilinear, aplica um leve efeito de embaçamento ou suavização à tela, amenizando um pouco o serrilhado nos contornos gráficos e suavizando a tela do sistema.

Quando desativado terá uma imagem pura e com aparência mais serrilhada, esta opção também ocasiona artefatos na tela em caso de redimensionamento. Caso não goste da aparência amaciada da imagem, tente incrementar o valor da opção `-prescale` em vez de desativar todos os filtros. Consulte também a opção `-gl_gsl_filter`.

O valor predefinido é Ligado (`-filter`).

Exemplo:

```
mame ssf2tu -nofilter
```

No Windows funciona com todos os modos de vídeo (bgfx, d3d, etc.), nas outras plataformas **APENAS** aquelas compatíveis com OpenGL.

-[no]burnin

Monitora o brilho da tela durante a reprodução e no final da emulação, gera um PNG que pode ser usado para simular um efeito burn-in³ na tela. O PNG é criado de tal maneira que as áreas menos usadas da tela ficam totalmente brancas (pois as áreas a serem marcadas são escuras, todo o resto da tela deverá ficar um pouco mais iluminada).

A intenção é que este PNG possa ser carregado através de um arquivo de ilustração usando um valor alpha pequeno como valores entre 0.1 e 0.2 que se misturam bem com o resto da tela. Os arquivos PNG gerados são gravados no diretório snap dentro do systemname/burnin-<nome.da.tela>.png.

O valor predefinido é Desligado (-noburnin).

Exemplo:

```
mame neogeo -burnin
```

5.1.18 Opções para a configuração da tela inteira**-[no]switchres**

Permite ou não a comutação ou a troca da resolução durante a emulação. Esta opção é necessária para as opções -resolution evitando a troca das resoluções enquanto estiver no modo de tela inteira.

Em placas de vídeo modernas, há poucas razões para alternar as resoluções, a menos que esteja tentando alcançar as resoluções "exatas" dos pixels dos sistemas originais, o que exige ajustes significativos.

Útil também em monitores de LCD, uma vez que eles rodam com uma resolução fixa e as comutações da resolução algumas vezes são exageradas.

Essa opção não funciona com a opção -video gdi.

O valor predefinido é Desligado (-noswitchres).

Exemplo:

```
mame kof97 -switchres -resolution 978x720
```

³ Quando uma imagem ficava estática numa tela de tubo CRT durante muito tempo, a fina película de fósforo que fica por de trás da tela de vidro sofria uma leve **queima** nas regiões de maior intensidade ficando uma marca no lugar. Uma vez marcada, essa mancha ficava sobre a imagem como se fosse uma sombra e nem sempre era necessário que a tela estivesse ligada para que a mancha pudesse ser visualizada na tela. (Nota do tradutor)

5.1.19 Opções de vídeo para uso com janelas individuais

-screen[0-3] <display>

Define qual monitor físico do sistema usar em cada janela. Para usar várias janelas, o valor da opção **-numscreens** deverá ter sido aumentado. O nome de cada tela do seu sistema pode ser identificado executando o MAME com a opção **-verbose**. Os nomes de cada tela geralmente estão no formato: `\\.\DISPLAYn` no Windows e `screenN` no macOS e variantes do OpenGL como o Linux por exemplo, o **n** ou **N** é um número do monitor que estiver conectado.

O valor predefinido para estas opções é auto. O que significa que a primeira janela é colocada na primeira exibição, a segunda janela na segunda e assim por diante.

Exemplo:

```
Windows
mame pc_cntra -numscreens 2 -screen0 \\.\DISPLAY1 -screen1 \\.\DISPLAY2
mame darius -numscreens 3 -screen0 \\.\DISPLAY1 -screen1 \\.\DISPLAY3 -screen2_
↪ \\.\DISPLAY2

OpenGL (Mac, Linux, *nix)
mame pc_cntra -numscreens 2 -screen0 screen0 -screen1 screen1
mame darius -numscreens 3 -screen0 screen1 -screen1screen3 -screen2 screen2
```

Os parâmetros `-screen0`, `-screen1`, `-screen2`, `-screen3` são específicos para cada janela. Já o parâmetro `-screen` aplica a configuração à todas as janelas. As opções definidas para uma janela específica tem prioridade sobre às opções das outras janelas.

Nota: Utilize a opção **-verbose** para exibir quais os displays estão disponíveis no seu sistema e qual a sua resolução quando estiverem conectados.

Nota: A partir de agora a opção de várias telas simultâneas podem não funcionar corretamente em alguns computadores Mac.

-aspect[0-3] <largura:altura> / -screen_aspect <num:den>

Define a proporção física do monitor para cada janela. Para usar várias janelas, é necessário aumentar o valor da opção **-numscreens**. A proporção física pode ser determinada medindo a largura e a altura da imagem da tela visível e definindo-as separadas por dois pontos.

O valor predefinido para essas opções é auto.

Significa que o MAME assume que a proporção de tela é proporcional ao número de pixels no modo de vídeo da área de trabalho para cada monitor.

O parâmetro `-aspect0`, `-aspect1`, `-aspect2` e `-aspect3` são específicos para cada janela. O parâmetro `-aspect` se aplica à todas as janelas. As opções definidas para uma janela específica tem prioridade sobre às opções das outras janelas. Consulte [-unevenstretch](#).

Exemplo:

```
mame contra -aspect 16:9
mame pc_cntra -numscreens 2 -aspect0 16:9 -aspect1 5:4
```

-resolution[0-3] <[largura_x_altura]@[taxa de atualização]> / **-r[0-3]** <[largura_x_altura]@[taxa de atualização]>

Define a resolução exata a ser exibida. No modo de tela cheia o MAME tentará usar a resolução solicitada. A largura e a altura são obrigatórias, a taxa de atualização é opcional.

Caso seja omitido ou configurado para 0, o MAME determinará o modo automaticamente. Por exemplo, a opção `-resolution 640x480` forçará a resolução de 640x480 porém o MAME escolherá a taxa de atualização por conta própria.

Da mesma forma que `-resolution 0x0@60` obrigará que a taxa de atualização seja de 60 Hz, mas permite que o MAME escolha a resolução. O comando também funciona com `"auto"` e é equivalente a `0x0@0`.

No modo janela essa resolução é usada para determinar o tamanho máximo para a janela. Essa opção também requer que seja usada a opção `-switchres` para ativar a comutação de resolução em conjunto com a opção **-video d3d**.

O valor predefinido para essas opções é auto.

O parâmetro `-resolution0`, `-resolution1`, `-resolution2` e `-resolution3` se aplica a todas as janelas definidas. O parâmetro `-resolution` se aplica a todas as janelas. As opções específicas da janela substituem os valores das opções de todas as janelas.

Exemplo:

```
mame pc_cntra -numscreens 2 -resolution0 768x720 -resolution1 640x480
```

-view[0-3] <nome>

Define a configuração da visualização inicial de cada janela. Note que o nome de visualização <nome> não precisa ser uma combinação exata, em vez disso, será selecionado a primeira exibição cujo nome corresponde a todos os caracteres especificados por <nome>. Por exemplo, `-view native` representa uma visualização "Native (15:14)", ainda que não seja o nome exato, supondo que o nome seja **Cabine Animada** basta usar **Cabine** ou **cabine**. O campo <nome> também funciona com a opção `auto` fazendo com que um nome seja automaticamente escolhido.

O valor predefinido para estas opções é auto.

Os parâmetros `-view0`, `-view1`, `-view2` e `-view3` se aplicam a todas as janelas especificadas. O parâmetro `-view` se aplica a todas as janelas. As opções definidas para a janela substituem os valores das opções de todas as janelas.

Para identificar qual o nome correto para ser utilizado com a opção `-view`, ao iniciar o sistema pressione Tab e vá em *Opções de Vídeo*, escolha a visualização que mais lhe agrada e encerre a emulação. No diretório **cfg** haverá um arquivo de configuração com **nomedarom.cfg** (se usarmos o exemplo abaixo o nome do arquivo será **neobombe.cfg**), abra-o num editor de texto qualquer, no campo `view` veja qual a opção está sendo usada e use-a na linha de comando como mostra o exemplo abaixo:

Exemplo:

```
mame neobombe -view "Screen 0 Cropped (304x224)"
```

Supondo que esta seja a sua opção de visualização preferida e caso queira que ela seja aplicada a todos os sistemas deste driver, no nosso caso, basta fazer o comando `mame -ls neobombe` para identificá-lo. Com o nome do driver em mãos crie o arquivo **neogeo.ini** dentro do diretório **ini/source** e use a opção `view "Screen 0 Cropped (304x224)"`.

Salve o arquivo e rode o sistema novamente sem a opção `-view`, note que ele já começa com a opção selecionada de visualização.

5.1.20 Opções para uso com as ilustrações

-[no]artwork_crop / -[no]artcrop

Ativa o recorte de arte somente na área da tela do sistema. Significa que sistemas que tenham telas com orientação horizontal rodando em tela cheia possam exibir a sua ilustração do lado esquerdo e direito da tela.

Essa opção também está disponível através da interface gráfica na parte das opções de vídeo.

O valor predefinido é Desligado (-noartwork_crop).

Exemplo:

```
mame ssf2tu -artwork_crop
```

-fallback_artwork

Define uma ilustração alternativa caso nenhuma ilustração interna ou externa do layout seja definida. Caso a ilustração para o sistema esteja presente ou o seu layout esteja incluso no driver do sistema, então este terá precedência.

```
mame coco -fallback_artwork suprmrio
```

-override_artwork

Define uma ilustração para substituir a ilustração interna ou a ilustração externa do layout.

Exemplo:

```
mame galaga -override_artwork puckman
```


5.1.21 Opções para os ajustes de imagem da tela

-brightness <valor>

Controla o valor de brilho ou nível de preto da tela. Essa opção não afeta a arte ou outras partes da tela. Usando a interface interna do MAME, é possível configurar o brilho para cada tela do sistema e para todos os sistemas individualmente. Ao selecionar valores menores (não menor que 0.1) produzirá uma tela mais escura, enquanto valores maiores até 2.0 produzirão uma tela mais clara.

O valor predefinido é 1.0.

Exemplo:

```
mame ssf2tu -brightness 0.5
```

-contrast <valor>

Controla o contraste da tela ou o nível de branco da tela. Essa opção não afeta a arte ou outras partes da tela. Usando a interface interna do MAME, é possível configurar o brilho para cada tela do sistema e para todos os sistemas individualmente. Essa opção define o valor inicial de todas as telas visíveis de todos os sistemas. Selecionando valores (não menor que 0.1) produzirá uma tela mais apagada, enquanto valores maiores até 2.0 produzirão uma tela mais saturada.

O valor predefinido é 1.0.

Exemplo:

```
mame ssf2tu -contrast 0.5
```

-gamma <valor>

Controle de gamma, ajusta a escala de luminância da tela. Essa opção não afeta a arte ou outras partes da tela. Usando a interface interna do MAME, é possível configurar o gamma para cada tela do sistema e para todos os sistemas individualmente. Essa opção define o valor inicial de todas as telas visíveis de todos os sistemas. Essa configuração oferece um ajuste de luminância linear de preto para o branco. Ao selecionar valores menores (até 0.1) trará a luminância mais para o preto, enquanto valores maiores (até 3.0) empurrarão essa luminância para o branco.

O valor predefinido é 1.0.

Exemplo:

```
mame ssf2tu -gamma 0.8
```

-pause_brightness <valor>

Faz o controle do nível de brilho durante a pausa.

O valor predefinido é 0.65.

Exemplo:

```
mame ssf2tu -pause_brightness 0.33
```

-effect <nome do arquivo>

Define um único arquivo .png que será usado como sobreposição na tela de qualquer sistema. Presume-se que o arquivo .png esteja em um dos diretórios raiz do [artpath](#).

Ambas as combinações horizontais e verticais dentro do arquivo .png é repetido para cobrir toda a tela (mas nenhuma parte da arte externa). Ela é renderizada na resolução nativa do sistema.

É possível adicionar o efeito de forma automática para sistemas com orientação horizontal e vertical, basta criar os arquivos **vertical.ini** e **horizont.ini** dentro do diretório **ini**. No arquivo **vertical.ini** adicione a linha abaixo e salve ao terminar:

Exemplo:

effect	RealScanlinesV
--------	----------------

Para o arquivo **horizont.ini** adicione a linha abaixo e salve ao terminar:

Exemplo:

effect	RealScanlinesH
--------	----------------

Consulte [Opções para a configuração](#) para saber quais os arquivos .ini estão disponíveis.

Para os modos de vídeo **-video gdi** e **-video d3d** significa que um pixel dentro do .png será mapeado para um pixel da sua tela. Os valores RGB de cada pixel dentro do .png são multiplicados com os valores de RGB da tela de destino. Ambos os arquivos **RealScanlinesV** e **RealScanlinesH** estão disponíveis no [site do projeto](#).

O valor predefinido é **none** ou nenhum efeito.

Exemplo:

Para efeito Horizontal mame ssf2tu -effect RealScanlinesH Para efeito Vertical mame bgaregga -effect RealScanlinesV
--



Claro que os exemplos acima são apenas exemplos, existem diferentes outros tipos espalhados pela internet e também é uma questão de gosto, há quem prefira os **shaders** que dão uma aparência muito convincente de um monitor CRT. Para mais informações sobre estes efeitos consulte [BGFX](#), [HLSL](#), [GLSL](#) e a opção `-glsl_shader_mame`.

5.1.22 Opções para sistemas que usem gráficos vetoriais

-beam_width_min <largura>

Define a espessura mínima do feixe do vetor. Esta espessura varia entre mínimo e máximo à medida que a intensidade do traçado do vetor muda. Para desativar as alterações da largura do vetor com base na intensidade, defina o mesmo valor para `-beam_width_max` e `-beam_width_min`.

Exemplo:

```
mame asteroid -beam_width_min 0.1
```

-beam_width_max <largura>

Define a espessura máxima do feixe do vetor. Esta espessura varia entre mínimo e máximo à medida que a intensidade do traçado do vetor muda. Para desativar as alterações da largura do vetor com base na intensidade, defina o mesmo valor para `-beam_width_max` e `-beam_width_min`.

Exemplo:

```
mame asteroid -beam_width_max 2
```

-beam_dot_size <tamanho>

Fator de escala que será aplicado ao tamanho dos pontos nos jogos vetoriais com ponto único. Normalmente estes são renderizados de acordo com a largura do feixe computado, no entanto, é comum que isto produza pontos que sejam difíceis de se ver. Esta opção aplica um fator de escala no topo da largura do feixe para que estes fiquem mais visíveis.

Exemplo:

```
mame asteroid -beam_dot_size 2
```

-beam_intensity_weight <altura>

Define a intensidade do feixe do vetor. Este valor determina como a intensidade afeta a sua largura, o valor 0 (zero) cria um mapeamento linear a partir da intensidade da sua largura. Os valores negativos fazem com que os vetores com menor intensidade aumentem a sua largura máxima de forma mais rápida, enquanto valores positivos o farão de forma mais lenta.

Exemplo:

```
mame asteroid -beam_intensity_weight 0.5
```

-flicker <valor>

Simula um vetor de efeito de *tremulação* ou oscilação da tela semelhante aos monitores desregulados usados nos jogos vetoriais. Essa opção espera um valor flutuante (float) no intervalo entre 0.00 e 100.00 (0 = nenhum e 100 = máximo).

O valor predefinido é 0.

Exemplo:

```
mame asteroid -flicker 0.15
```

5.1.23 Opções das principais características do vídeo OpenGL

Essas são as opções compatíveis com `-video opengl`. Caso note artefatos renderizados na tela, poderá ser solicitado pelos desenvolvedores que você tente alterá-los, porém normalmente estes valores devem ser mantidos com seus valores originais para que se obtenha a melhor desempenho possível.

-[no]gl_forcepow2texture

Sempre utilize a potência de 2 para o tamanhos das texturas.

O valor predefinido é Desligado (`-nogl_forcepow2texture`).

-[no]gl_notexturerect

Não use o *OpenGL GL_ARB_texture_rectangle*

O valor predefinido é Ligado (`-gl_notexturerect`).

-[no]gl_vbo

Ative o *OpenGL VBO* (Vertex Buffer Objects) caso esteja disponível.

O valor predefinido é Ligado (`-gl_vbo`).

-[no]gl_pbo

Ativar o *OpenGL PBO* (Pixel Buffer Objects) caso esteja disponível.

O valor predefinido é Ligado (`-gl_pbo`).

5.1.24 Opções de vídeo OpenGL GLSL

-[no]gl_gsl

Ativar o *OpenGL GLSL* caso esteja disponível.

O valor predefinido é Desligado (-nogl_gsl).

Exemplo:

```
mame galaxian -gl_gsl
```

-gl_gsl_filter <valor>

Ativa a interpolação da imagem **OpenGL GLSL**, os valores válidos⁶ são:

- 0, Simples: Método de interpolação rápida e menos precisa que deixa os pixels de forma serrilhada pois utiliza a técnica de interpolação do [vizinho mais próximo](#).
- 1, Bilinear: Método de interpolação lenta e de qualidade mediana, suaviza a transição entre as cores dos pixels deixando a imagem mais suavizada como um todo. Veja também [-filter](#).
- 2, Bicúbico: Método de interpolação lenta e mais precisa, suaviza a transição entre as cores dos pixels próximos gerando uma gradação mais suave. Também suaviza a imagem porém nem tanto como o método bilinear.

O valor predefinido é 1 (-gl_gsl_filter 1).

Exemplo:

```
mame ssf2tu -gl_gsl -gl_gsl_filter 0
```

-gsl_shader_mame[0-9]

Define um efeito shader personalizado do OpenGL GLSL do MAME no slot fornecido entre (0-9). É possível aplicar um para a cada slot. O MAME não inclui nenhum shader por alguns podem ser encontrados online, como no [mameau](#) e no [mameworld](#).

Exemplo:

```
mame mpatrol -gl_gsl -gl_gsl_filter 0 -gsl_shader_mame0 gsl/osd/CRT-geom -
↳ resolution 992x756
```

-gsl_shader_screen[0-9]

Define um efeito shader personalizado do OpenGL GLSL do MAME que for escalada na tela para ser processada através da sua placa de vídeo no slot fornecido entre (0-9). Como no exemplo anterior, o MAME não inclui nenhum shader.

Exemplo:

```
mame suprmrio -gl_gsl -gsl_shader_screen0 gaussx -gsl_shader_screen1 gaussian
↳ -gsl_shader_screen2 CRT-geom-halation
```

⁶ <https://github.com/mamedev/mame/pull/2989/files>

5.1.25 Opções para a configuração do áudio

-samplerate <valor> / **-sr** <valor>

Define a taxa de amostragem do áudio. Valores menores como 11025 por exemplo, reduzem a qualidade da áudio porém o desempenho da emulação melhora. Valores maiores que 48000, aumentam a qualidade do áudio ao custo da perda do desempenho da emulação.

O valor predefinido é 48000 (`-samplerate 48000`).

Exemplo:

```
mame ssf2tu -samplerate 44100
```

-[no]samples

Usar arquivos de amostras caso estejam disponíveis. Esses arquivos são gravações de efeitos de áudio usados por alguns sistemas.

O valor predefinido é Ligado (`-samples`).

Exemplo:

```
mame qbert -nosamples
```

-[no]compressor

Ativa a compressão do áudio. Reduz temporariamente o volume total quando a saída do áudio for saturada.

O valor predefinido é Ligado (`-compressor`).

Exemplo:

```
mame popeye -nocompressor
```

-volume / **-vol** <valor>

Define o volume inicial. Pode ser alterado posteriormente usando a interface do usuário (consulte [Teclas já predefinidas](#)). O valor do volume está definido em decibéis (dB); por exemplo, `-volume -12` iniciará a emulação com uma atenuação de **-12 dB**. Observe que, caso o volume seja alterado na interface do usuário, ele será salvo no arquivo de configuração do sistema. O valor do arquivo de configuração para o sistema **tem prioridade** sobre as configurações de volume nos arquivos INI.

O valor predefinido é 0 (`-volume 0`, sem atenuação com volume no máximo).

Exemplo:

```
mame ssf2tu -volume -30
```

-speaker_report <valor>

Imprime um relatório na tela dizendo se houve ou não um ceifamento ou "*clipping*" da onda gerada pela saída de áudio no alto-falante. Este ceifamento causa distorções no áudio devido a saturação do sinal, tornando as curvas de uma onda senoide numa onda quadrada.

Os valores válidos são:

- **nível 1:** relate apenas se houve clipping.
- **nível 2:** relate a máxima geral, ainda que não haja qualquer clipping.

- **nível 3:** imprima uma lista detalhada de todos os momento que houveram clipping.
- **nível 4:** imprima uma lista detalhada de tudo.

O valor predefinido é Desligado.

Exemplo:

```
mame robocop -speaker_report 2
Speaker ":mono" - max = 51586 (gain *= 0.635) - clipped in 30/735 (4%) buckets
```

Nota: Este comando não tem utilidade prática para o usuário, serve apenas como uma ferramenta para os desenvolvedores do MAME.

-sound < auto | dsound | sdl | coreaudio | xaudio2 | portaudio | none >

Define qual o tipo de saída de áudio usar, Ao usar none desativa o áudio completamente porém o hardware de áudio continua sendo emulado. Abaixo as opções disponíveis para cada sistema operacional.

As versões especiais como o **SDLMAME** para Windows, pode usar a opção **sdl** e ter o **portaudio** desativado. O binário oficial do MAME para Windows não é compilado com SDL, sendo necessário compilar uma versão compatível para que a opção **sdl** funcione.

O valor predefinido é **dsound** no Windows, no Mac é **coreaudio** nas outras plataformas é **sdl**.

No Windows e no Linux a opção **portaudio** provavelmente dará uma menor latência possível, enquanto no Mac a opção **coreaudio** oferecerá os melhores resultados.

Exemplo:

```
mame sf2tu -sound portaudio
```

Tabela 6: Opções disponíveis para cada versão separado por plataforma

Microsoft Windows	auto	dsound	xaudio2	portaudio	none
macOS	auto	coreaudio	sdl	portaudio	none
SDL	auto	portaudio	sdl	none	

-audio_latency <valor>

Nesta opção, latência significa o tempo que o dispositivo de áudio demora para responder a um comando. Essa opção ajusta a quantidade dessa latência incorporada ao fluxo de dados de áudio.

O comportamento exato depende do módulo da saída de áudio selecionada. Os valores menores oferecem um menor atraso no áudio enquanto exige um desempenho maior do seu sistema. Valores maiores incrementam o atraso do áudio porém ajudam a evitar o esvaziamento da memória intermediária (buffer) e as interrupções do áudio.

Os valores válidos ficam entre 0 e 5, o valor predefinido é 2.

Exemplo:

```
mame galaga -audio_latency 1
```

Nota:

Para PortAudio, consulte *-pa_latency*.

O XAudio2 calcula a latência do áudio com passos de 10ms.

O DSound calcula a latência do áudio com passos de 10ms.

O CoreAudio calcula a latência do áudio com passos de 25ms.

O SDL calcula a latência do áudio com passos de Xms.

-pa_api <interface>

PortAudio é um novo recurso adicionado na versão **0.182** do MAME, o *PortAudio* é uma API, "Application Programming Interface" ou numa tradução livre "Interface de Programação para Aplicações". A API funciona como uma ponte conectando aplicações ao hardware de forma direta. Essa integração permite uma menor latência por haver uma redução no fluxo de dados e por estes dados de áudio serem direcionados diretamente ao dispositivo áudio, o desempenho é otimizado de maneira geral pois o que se salva em processamento no áudio pode ser aproveitado pelo MAME em outros setores da emulação.

Apesar do *PortAudio* ser o que há de mais moderno em comparação com o *DirectSound* ou *OpenGL Audio* e trazer muitos benefícios, há um ponto negativo, o *PortAudio* faz o uso exclusivo do dispositivo de áudio. Isso significa que não será possível por exemplo, escutar música ou qualquer outra coisa enquanto o MAME estiver rodando com *PortAudio*.

No Windows Vista ou mais recente nós temos essas interfaces:

- **MME:** É um acrônimo para *Multimedia Extension* criada pela Microsoft para um sistema operacional pouco conhecido na época chamado "Windows with MultiMedia Extensions 1.0" com base no **Windows 3.0**, é um dos primeiros API para comunicação direta com a placa de áudio. Essa interface já é obsoleta porém ainda muito usada por questões de compatibilidade.
- **Windows DirectSound:** É uma outra API introduzida pela Microsoft no **Windows 95** que adicionou uma camada de software entre a aplicação e o dispositivo de som. Com ele uma placa de som poderia ter dois canais ou mais, efeitos de som 3D foi uma novidade na época, aceleração de áudio via hardware, a placa de som poderia ser compartilhada entre diferentes aplicativos. Essa interface já é obsoleta porém ainda muito usada por questões de compatibilidade.
- **Windows WASAPI:** É um acrônimo para "Windows Audio Session API" ou numa tradução livre, "API da seção de áudio do Windows". Foi introduzido no **Windows Vista**, a grande vantagem do WASAPI é poder enviar os fluxos de dados de áudio direto para o dispositivo

de áudio sem ter que passar por nenhum tipo de *CODEC* (codificador / decodificador). Outra característica do *WASAPI* é ter o uso exclusivo do dispositivo de áudio melhorando a latência assim como a qualidade do áudio.

- **Windows WDM-KS:** É um acrônimo para "*Windows Driver Model*" também criado pela *Microsoft* e introduzido no **Windows 98** e no **Windows 2000**. O *KS* vem de "*Kernel Streaming*" uma maneira ainda mais rápida de acessar o dispositivo de áudio de forma direta através do cerne (kernel) do *Windows*. Apesar de também fazer uso exclusivo do dispositivo de áudio essa é a interface mais problemática pois ela é muito dependente da qualidade dos drivers usados, gera problemas com a hibernação do *Windows* quando há problemas com os drivers, a melhor opção é ficar com o *Windows WASAPI*.

Para escolher qual interface usar, inicie o mame como mostra o exemplo abaixo:

Exemplo:

```
mame -verbose -sound portaudio
```

No *Windows* dentre as várias informações que aparecerão no terminal as mais relevantes para nós serão estas:

Exemplo:

```
PortAudio: API Windows WASAPI has 10 devices
PortAudio: Windows WASAPI: "6 - SONY TV *01 (AMD High Definition Audio Device)
↳ (default)"
PortAudio: Windows WASAPI: "Alto-falantes (ASUS Xonar Essence STX Audio Device)
↳ "
PortAudio: Windows WASAPI: "S/PDIF Pass-through Device (ASUS Xonar Essence STX_
↳ Audio Device)"
PortAudio: Windows WASAPI: "Alto-falantes (2- Blackmagic Audio)"
PortAudio: Windows WASAPI: "Aux (ASUS Xonar Essence STX Audio Device)"
PortAudio: Windows WASAPI: "Entrada (2- Blackmagic Audio)"
PortAudio: Windows WASAPI: "Entrada (ASUS Xonar Essence STX Audio Device)"
PortAudio: Windows WASAPI: "Wave (ASUS Xonar Essence STX Audio Device)"
PortAudio: Windows WASAPI: "Stereo Mix (ASUS Xonar Essence STX Audio Device)"
PortAudio: Windows WASAPI: "Microfone (ASUS Xonar Essence STX Audio Device)"

PortAudio: API Windows WDM-KS has 12 devices
PortAudio: Windows WDM-KS: "Output (AMD HD Audio HDMI out #5)" (default)
PortAudio: Windows WDM-KS: "Input (ASUS Xonar Essence STX Audio)"
PortAudio: Windows WDM-KS: "Entrada (ASUS Xonar Essence STX Audio)"
PortAudio: Windows WDM-KS: "Aux (ASUS Xonar Essence STX Audio)"
PortAudio: Windows WDM-KS: "Microfone (ASUS Xonar Essence STX Audio)"
PortAudio: Windows WDM-KS: "Speakers (ASUS Xonar Essence STX Audio)"
PortAudio: Windows WDM-KS: "SPDIF Out (ASUS Xonar Essence STX Digital)"
PortAudio: Windows WDM-KS: "Wave (ASUS Xonar Essence STX Audio Wave In)"
PortAudio: Windows WDM-KS: "Input (Blackmagic WDM Capture)"
PortAudio: Windows WDM-KS: "Output ()"
PortAudio: Windows WDM-KS: "Speakers ()"
PortAudio: Windows WDM-KS: "Entrada ()"
```

No exemplo acima estão listados dois exemplos de interface, **Windows WASAPI** e **Windows WDM-KS**. O uso de qualquer uma destas interfaces depende do driver da sua placa de som. Para definir a interface use o nome dela entre aspas `-pa_api "Windows WASAPI"` ou `-pa_api "Windows WDM-KS"`.

Já no Linux nós temos uma lista um pouco diferente ainda que estejamos usando o mesmo hardware acima:

Exemplo:

```
PortAudio: API ALSA has 15 devices
PortAudio: ALSA: "Xonar STX: Multichannel (hw:0,0)"
PortAudio: ALSA: "Xonar STX: Digital (hw:0,1)"
PortAudio: ALSA: "HDA ATI HDMI: 0 (hw:1,3)"
PortAudio: ALSA: "HDA ATI HDMI: 1 (hw:1,7)"
PortAudio: ALSA: "HDA ATI HDMI: 2 (hw:1,8)"
PortAudio: ALSA: "HDA ATI HDMI: 3 (hw:1,9)"
PortAudio: ALSA: "HDA ATI HDMI: 4 (hw:1,10)"
PortAudio: ALSA: "HDA ATI HDMI: 5 (hw:1,11)"
PortAudio: ALSA: "sysdefault"
PortAudio: ALSA: "front"
PortAudio: ALSA: "iec958"
PortAudio: ALSA: "spdif"
PortAudio: ALSA: "pulse"
PortAudio: ALSA: "dmix"
PortAudio: ALSA: "default" (default)
PortAudio: API OSS has 0 devices
PortAudio: Unable to find specified API or device or none set, reverting to
↳ default
PortAudio: Using device "default" on API "ALSA"
PortAudio: Sample rate is 48000 Hz, device output latency is 8.67 ms
PortAudio: Allowed additional buffering latency is 30.00 ms/1440 frames
```

O valor predefinido é NULO (Nenhuma interface *PortAudio*).

-pa_device <dispositivo>

Define qual o dispositivo de áudio usar, assim como mostrado em *-pa_api*, escolha um dos dispositivos listados. O nome do dispositivo fica do lado direito da lista e entre aspas. Usando o exemplo para o Windows nós usaremos:

Exemplo:

```
mame -verbose -sound portaudio -pa_api "Windows WASAPI" -pa_device "6 - SONY
↳ TV *01 (AMD High Definition Audio Device)"
```

Já para Linux o comando também não é muito diferente para o mesmo dispositivo:

Exemplo:

```
./mame -verbose -sound portaudio -pa_api ALSA -pa_device "HDA ATI HDMI: 0
↳ (hw:1,3)"
```

Como resultado o MAME deverá exibir a mensagem abaixo mostrando que tanto a interface quanto o dispositivo foram aceitos:

Exemplo:

```
PortAudio: Using device "6 - SONY TV *01 (AMD High Definition Audio Device)"
↳ on API "Windows WASAPI"
```

Aqui o mesmo para o Linux:

Exemplo:

```
PortAudio: Using device "HDA ATI HDMI: 0 (hw:1,3)" on API "ALSA"  
PortAudio: Sample rate is 48000 Hz, device output latency is 8.00 ms
```

Caso nenhum seja definido o MAME escolherá o dispositivo padrão ou que estiver disponível.

O valor predefinido é NULO (Nenhum dispositivo *PortAudio*).

-pa_latency <segundos>

Escolha o tamanho em segundos da memória intermediária para a saída PortAudio. Números menores tem um menor atraso porém valores muito agressivos podem aumentar a quantidade de cortes no áudio. Os valores compatíveis são em pontos decimais. Tente começar com valores como **0.20** e vá aumentando ou reduzindo este valor até encontrar o valor apropriado que funcione melhor com o seu hardware e o seu sistema operacional.

O valor predefinido é 0.

Exemplo:

```
mame -verbose -sound portaudio -pa_api "Windows WASAPI" -pa_device "6 - SONY_  
↳TV *01 (AMD High Definition Audio Device)" -pa_latency 0.20
```

5.1.26 Opções para as configurações de diferentes entradas

-[no]coin_lockout / -[no]coinlock

Permite a simulação do recurso "bloqueio de ficha" implementado em vários PCBs de jogos de arcade. Cabia ao operador saber se as saídas de bloqueio da moeda estavam realmente conectadas aos mecanismos das moedas. Se esse recurso estiver ativado, as tentativas de inserir uma moeda enquanto o bloqueio estiver ativo falharão e exibirão uma mensagem na tela (no modo de depuração). Caso esta função esteja desativada, o sinal de bloqueio da moeda será ignorado.

O valor predefinido é Ligado (-coin_lockout).

Exemplo:

```
mame ssf2tu -coin_lockout
```

-ctrlr <controle>

Define um arquivo de configuração personalizada do controle, normalmente usado para definir novas atribuições das entradas predefinidas. Todos os diretórios definidos em `ctrlrpath` são pesquisados. Os arquivos de configuração do controle utilizam um formato similar ao `.cfg` que é utilizado para gravar as configurações do sistema. Estes arquivos são criados durante a configuração dos botões do controle de um sistema, estas configurações são gravadas no diretório `cfg` como `(nome_da_maquina).cfg`. Para mais informações, consulte o capítulo [Arquivos de configuração para os controles](#) e o capítulo [Configurando a ordem dos botões](#).

O valor predefinido é NULO (nenhum arquivo de configuração).

Exemplo:

```
mame ssf2tu -ctrlr 6-botoes
```

-[no]mouse

Controla se o MAME faz uso ou não dos controladores do mouse. Caso esta opção esteja ativa, o mouse ficará reservado para uso exclusivo do MAME até que a emulação seja pausada ou até que a emulação seja encerrada. A compatibilidade depende da configuração do seu *-mouseprovider*.

Observe que caso esta opção esteja desligada (-nomouse), a entrada do mouse ainda pode estar ativa dependendo da entrada presente no sistema emulado e a sua *definição automática de entrada*. Mais especialmente, o padrão é ativar a entrada do mouse quando o sistema emulado tiver entradas para o mouse (-mouse_device mouse), assim sendo, o ponteiro do seu mouse será capturado pelo MAME caso o sistema emulado tenha tal entrada, a menos que, você altere a configuração através da opção *-mouse_device*.

O valor predefinido é Desligado (-nomouse).

Exemplo:

```
mame centiped -mouse
```

-[no]joystick / -[no]joy

Controla se o MAME usa ou não os controles dos jogos (joystick, gamepad e outras simulações de controles). A compatibilidade com os controles dependem da configuração do seu *joystickprovider*. Caso esta opção esteja ativa, o MAME perguntará sobre quais os controles estão conectados atualmente no sistema.

Observe que caso esta opção esteja desligada (-nojoystick), a entrada do joystick ainda pode estar ativa dependendo da entrada presente no sistema emulado e a sua *definição automática de entrada*.

O valor predefinido é Desligado (-nojoystick).

Exemplo:

```
mame mappy -joystick
```

-[no]lightgun / -[no]gun

Controla se o MAME usa ou não os controles da arma de luz (lightgun). Observe que a maioria das armas de luz são mapeadas para o mouse, assim, ao se usar ambas as opções -lightgun e -mouse juntas, isso pode trazer resultados inesperados. O suporte aos controladores da arma de luz dependem das configurações do *lightgunprovider*. Para mais informações, consulte o capítulo *Funcionamento*.

Observe que caso esta opção esteja desligada (-nolightgun), a entrada para a arma de luz ainda pode estar ativa dependendo da entrada presente no sistema emulado e a sua *definição automática de entrada*.

O valor predefinido é Desligado (-nolightgun).

Exemplo:

```
mame lethalen -lightgun
```

-[no]multikeyboard / -[no]multikey

Determina se o MAME diferencia entre os vários teclados disponíveis. Alguns sistemas podem reportar mais de um teclado; por padrão, os dados de todos esses teclados são combinados para que pareçam um só. Ativando essa opção permitirá que o MAME retorne quais teclas foram pressionadas em diferentes teclados de maneira independente.

O valor predefinido é Desligado (-nomultikeyboard).

Exemplo:

```
mame ssf2tu -multikey
```

-[no]multimouse

Determina se o MAME diferencia entre os vários mouses disponíveis. Alguns sistemas podem reportar mais de um dispositivo de mouse; por padrão, os dados de todos esses mouses são combinados para que pareçam um só. Ativando esta opção fará com que o MAME relate o movimento e o pressionar de botões do mouse em diferentes mouses de maneira independente.

O valor predefinido é Desligado (-nomultimouse).

Exemplo:

```
mame warlords -multimouse
```

-[no]steadykey / -[no]steady

Alguns sistemas exigem que dois ou mais botões sejam pressionados exatamente ao mesmo tempo para realizar movimentos ou comandos especiais. Devido a limitação do hardware do teclado, pode ser difícil ou até mesmo impossível de realizar usando um teclado comum. Essa opção seleciona diferentes modos de manuseio o que torna mais fácil registrar o pressionamento simultâneo das teclas, porém tem a desvantagem de deixar a sua capacidade de resposta mais lenta.

O valor predefinido é Desligado (-nosteadykey).

Exemplo:

```
mame ssf2tu -steadykey
```

-[no]ui_active

Ativa a opção para que a interface do usuário se sobreponha a do teclado emulado caso esteja presente.

O valor predefinido é Desligado (-noui_active).

Exemplo:

```
mame apple2e -ui_active
```

-[no]offscreen_reload / -[no]reload

Controla se o MAME trata o segundo botão da arma de luz (lightgun) como um sinal para recarregar a arma. Neste caso, o MAME reportará a posição da arma como (0,MAX) com o gatilho pressionado, o que é o equivalente a uma recarga da arma com ela apontada para fora da tela. Isso só é necessário para jogos que precisam que o usuário atire para fora da tela para recarregar a arma e se também a sua arma não tiver essa funcionalidade.

O valor predefinido é Desligado (-nooffscreen_reload).

Exemplo:

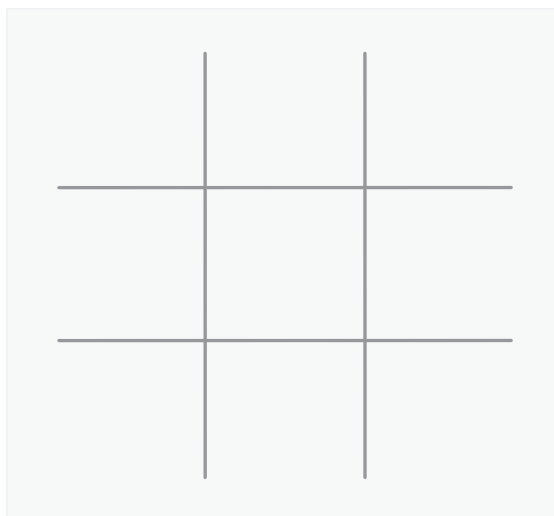
```
mame lethalen -offscreen_reload
```

-joystick_map / -joymap <mapa>

Controla como os valores do joystick analógico são mapeados para os controles num joystick digital.

Sistemas como o Pac-Man utilizam um controle digital com quatro posições (4-way joystick) que exibe um comportamento inapropriado quando a diagonal é acionada, o Pac-Man simplesmente para nos cruzamentos do mapa fazendo com que seja quase que impossível jogá-lo direito ou pelo menos, da maneira que ele foi desenvolvido para ser jogado. Outros sistemas arcade utilizam controles digitais do tipo 4-way ou 8-way (em vez de utilizarem controles analógicos), assim para controles analógicos como os utilizados em sistemas de voo com manches ou thumb sticks analógicos, estes precisam ser mapeados para valores correspondentes aos controles digitais com 4 ou 8 posições.

Para tanto o MAME divide o alcance do controle analógico numa grade com 9x9 muito semelhante ao desenho abaixo:



O MAME então toma a posição do eixo do joystick (apenas os valores nos eixos X e Y), mapeia para esta grade e então monitora a tradução a partir do mapa do joystick. Este parâmetro permite que você defina tal mapa.

Como por exemplo, um joystick com 8 posições geralmente se assemelha com a imagem abaixo:



Este mapeamento oferece uma grande margem de manobra para os ângulos aceitos numa determinada direção, assim sendo bem próximo da área da direção desejada. Sem isso, caso esteja um pouco fora do eixo central enquanto segura o controle para a esquerda, o sistema não vai reconhecer a ação de forma adequada.

O valor predefinido é auto, significa que o controle padrão com 4, 8 posições ou um mapa diagonal para um controle com 4 posições é selecionado de forma automática com base na informação da configuração da porta de entrada do sistema atual.

Caso queira configurar a opção `-joystick_map` de forma individual através do `<sistema>.ini` em vez da configuração do `mame.ini` para que tais configurações afetem apenas os sistemas que precisem destes mapeamentos, consulte a seção [Diversos arquivos de configuração](#) para mais detalhes.

Os mapeamentos são definidos como uma cadeia de caracteres e números. Uma vez que a grade possui 9x9, existem 81 caracteres no total para que seja possível definir um mapa completo. Abaixo é um exemplo de um mapa para um joystick com 8 posições que coincide com a imagem anterior.

7 7 7	8 8 8	9 9 9
7 7 7	8 8 8	9 9 9
7 7 7	8 8 8	9 9 9
4 4 4	5 5 5	6 6 6
4 4 4	5 5 5	6 6 6
4 4 4	5 5 5	6 6 6
1 1 1	2 2 2	3 3 3
1 1 1	2 2 2	3 3 3
1 1 1	2 2 2	3 3 3

Observe que os dígitos se assemelham com um teclado numérico. Então o "7" é mapeado como canto superior esquerdo, o "4" como esquerda, o "5" ao centro é mapeado como neutro, etc.

Em além dos valores numéricos, é possível especificar o caractere "s", que significa "sticky" ou "pegadiço". Neste caso, o valor do mapa é o mesmo que foi na última vez que um valor não pegadiço foi lido.

Para definir um mapa com estes parâmetros informe a cadeia de caracteres com as linhas separadas por um ponto '.' (que indica o final de uma linha):

Exemplo:

```
-joymap 777888999.777888999.777888999.444555666.444555666.444555666.111222333.
↪111222333.111222333
```

Contudo é possível reduzir o tamanho do comando utilizando-se de atalhos compatíveis com o parâmetro `<mapa>`. Caso falte uma informação sobre uma linha, se deduz então que qualquer dado que falte nas colunas 5-9 sejam simétricos com a posição esquerda/direita em relação com os dados das colunas 0-4; assim como, qualquer dado que falte nas colunas 0-4 é deduzido que sejam cópias dos dados anteriores. A mesma lógica se aplica para as linhas faltantes, exceto quando se assume a simetria cima/baixo.

Ao utilizar esta forma abreviada os 81 caracteres deste mapa podem ser simplificados para uma cadeia com 11 caracteres: `7778...4445` (significa que podemos utilizar o comando `-joymap 7778...4445`)

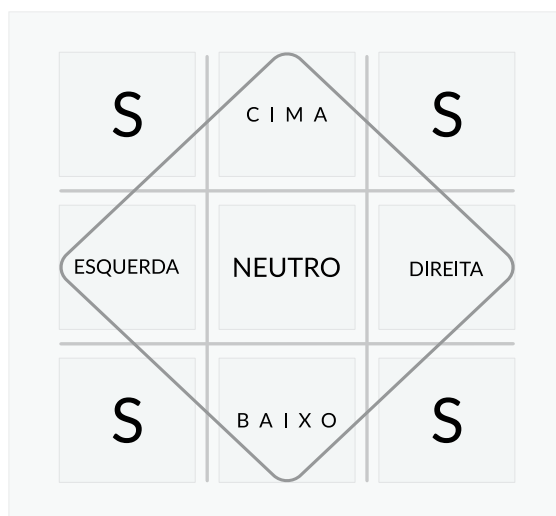
Olhando na primeira fileira, o `7778` tem o comprimento de apenas 4 caracteres. Não é possível utilizar a simetria na 5ª entrada, se deduz então que seja igual ao caractere anterior, '8'. O 6º caractere é simétrico com a posição esquerda/direita em conjunto com o 4º caractere, dando um

'8'. O 7º caractere é simétrico com a posição esquerda/direita em conjunto com o 3º caractere, dando um '9' (que é '7' com os lados esquerdo/direito invertido). Por fim, retorna o conjunto completo da linha **777888999**.

As carreiras dois e três estão faltando, se deduz então que sejam idênticos à 1ª linha. A 4ª carreira é decodificada de forma semelhante à 1ª gerando 444555666. A 5ª carreira está faltando, se deduz então que seja a mesma que a 4ª.

O resto das 3 carreiras também estão faltando, então deduz-se que sejam espelhos do sentido cima/baixo das 3 primeiras carreiras, gerando as carreiras finais com 111222333.

Em sistemas compatíveis com controles com 4 posições, o "sticky" (pegadiço) se torna importante para evitar problemas com as diagonais. Geralmente seria escolhido um mapa semelhante com este:



Significa que caso pressione esquerda e então role o controle para cima (sem passar pelo centro), o eixo do controle passará por aquela parte pegadiça do canto do controle fazendo com que o movimento "pegue" ou "raspe" o manche nas diagonais atrapalhando o movimento. Durante o movimento o MAME irá ler esta posição pegadiça como **esquerda** pois é a última posição não pegadiça que foi recebida. Até que o movimento chega até a posição **cima** do mapa até finalmente resultar no movimento para cima.

O mapa então ficaria muito parecido com isso:

s 8 8	8 8 8	8 8 s
4 s 8	8 8 8	8 s 6
4 4 s	8 8 8	s 6 6
4 4 4	5 5 5	6 6 6
4 4 4	5 5 5	6 6 6
4 4 4	5 5 5	6 6 6
4 4 s	2 2 2	s 6 6
4 s 2	2 2 2	2 s 6
s 2 2	2 2 2	2 2 s

Para este mapeamento, nós temos um alcance amplo para as direções cardeais 8, 4, 2 e 6.

Nós temos os pontos pegadiços marcado com "s" onde a direção apropriada não parece ser muito óbvia.

Para definir um mapa com estes parâmetros informe a cadeia de caracteres com as linhas separadas por um ponto '.' (que indica o final de uma linha):

Exemplo:

```
-joymap s8888888s.4s88888s6.44s888s66.444555666.444555666.444555666.44s222s66.
↪4s22222s6.s222222s
```

Assim como antes, por causa da simetria entre cima, baixo, esquerda e direita, podemos reduzir o comando para:

Exemplo:

```
-joymap s8.4s8.44s8.4445
```

-joystick_deadzone / -joy_deadzone / -jdz <valor>

Caso jogue com um joystick analógico ele poderá estar um pouco fora de centro. O `-joystick_deadzone` informa uma folga ao longo de um eixo que você deve mover antes que o eixo comece a mudar. Essa opção espera um valor flutuante (float) no intervalo entre 0.0 e 1.0. Onde 0 é o centro do joystick e 1 o limite externo.

Em ocasiões onde o joystick aparente ser muito sensível, tente alterar o valor do `joystick_deadzone` para 0 e o `joystick_saturation` para 1.

O valor predefinido é 0.3 (`-joystick_deadzone 0.3`).

Exemplo:

```
mame sinistar -joystick_deadzone 0.3
```

-joystick_saturation / joy_saturation / -jsat <valor>

Caso jogue com um joystick analógico as extremidades podem estar um pouco fora e podem não corresponder nas direções + /-. O `-joystick_saturation` define se uma folga no movimento do eixo será aceita até que se atinja o alcance máximo. Essa opção espera um valor flutuante (float) no intervalo entre 0.0 até 1.0 onde 0 é o centro do joystick e 1 é o limite externo.

O valor predefinido é 0.85 (`-joystick_saturation 0.85`).

Exemplo:

```
mame sinistar -joystick_saturation 1.0
```

-joystick_threshold <valor> / joy_threshold <valor> / -jthresh <valor>

Quando um eixo do *joystick* (ou outro eixo analógico absoluto) for atribuído a uma entrada digital, isso controla o quanto ele deve ser movido da posição neutra (ou centro) para ser considerado ativo ou ligado. Esta opção espera uma flutuação na faixa entre 0,0 a 1,0 onde 0 significa que qualquer movimento da posição neutra for considerado ativo e 1 significa que apenas os limites externos são considerados ativos. Este limite **não** é ajustado para a faixa entre a zona morta e o ponto de saturação. Observe que caso um *mapa do joystick* seja configurado, isso terá precedência sobre esta configuração quando os eixos principais X/Y de um joystick forem atribuídos às entradas digitais.

O valor predefinido é 0.3.

Exemplo:

```
mame raiden -joystick_threshold 0.2
```

-[no]natural

Permite que o usuário defina se deve ou não usar um teclado natural. Isso permite que o seu sistema num modo *nativo* dependendo da sua região, permitindo compatibilidade para teclados fora do padrão "QWERTY".

O valor predefinido é Desligado (-nonatural).

No modo de "teclado emulado" (predefinido) o MAME traduz o pressionamento/liberação de teclas/botões do host para pressionamentos emulados de tecla. Ao pressionar/soltar uma tecla/botão mapeado para uma tecla emulada, o MAME pressiona/libera a tecla emulada.

No modo "teclado natural", o MAME tenta traduzir os caracteres para as teclas digitadas. O sistema operacional traduz pressionamentos de tecla a caracteres (da mesma forma quando se digita num editor de texto) e o MAME tenta traduzir esses caracteres para pressionamentos de tecla emulados.

Existem várias limitações inevitáveis no modo "teclado natural":

- O driver do sistema emulado ou do dispositivo de teclado precisam ser compatíveis e haver suporte para eles.
- O teclado selecionado **deve** corresponder ao layout do teclado selecionado no sistema operacional emulado!
- As teclas que não produzam caracteres não podem ser traduzidas.
- Segurar uma tecla até que o caractere se repitam fará com que a tecla emulada seja pressionada repetidamente em vez de ser mantida pressionada.
- As sequências de chaves inativas na melhor das hipóteses, são complicadas de se usar.
- Não funcionará se a edição do **IME** estiver envolvida como Chinês/Japonês/Coreano por exemplo)

```
mame coco2 -natural
```

-[no]joystick_contradictory

Aceita a entrada de comandos contraditórios e simultâneos no controle digital como **Esquerda e Direita** ou **Cima e Baixo** ao mesmo tempo.

O valor predefinido é Desligado (-nojoystick_contradictory).

Exemplo:

```
mame ddr4m -joystick_contradictory
```

-coin_impulse [n]

Define o tempo de impulso da moeda com base em *n* (*n*<0 desativa, *n*=0 obedeça o driver, 0<*n* defina o tempo em *n*).

O valor predefinido é 0 (-coin_impulse 0).

Exemplo:

```
mame contra -coin_impulse 1
```

5.1.27 Opções das entadas principais ativadas automaticamente

-paddle_device / -paddle

Opções válidas none | keyboard | mouse | lightgun | joystick

-adstick_device / -adstick

Opções válidas none | keyboard | mouse | lightgun | joystick

-pedal_device / -pedal

Opções válidas none | keyboard | mouse | lightgun | joystick

-dial_device / -dial

Opções válidas none | keyboard | mouse | lightgun | joystick

-trackball_device / -trackball

Opções válidas none | keyboard | mouse | lightgun | joystick

-lightgun_device

Opções válidas none | keyboard | mouse | lightgun | joystick

-positional_device

Opções válidas none | keyboard | mouse | lightgun | joystick

-mouse_device

Opções válidas none | keyboard | mouse | lightgun | joystick

Cada uma destas opções de controle define se o mouse, o joystick ou o lightgun devem ser ativados ao executar a emulação de um sistema que usa uma classe das entradas analógicas em particular. Estas opções podem definir o *-mouse*, o *-joystick* ou o *-lightgun* dependendo do tipo das entradas disponíveis no sistema emulado. Observe que essas opções *não* substituirão as configurações usadas explicitamente na linha de comando de **-nomouse**, **-nojoystick** e/ou **-nolightgun** num nível de prioridade mais alto (num arquivo INI mais específico ou na linha de comando por exemplo).

Por exemplo, caso defina a opção `-paddle_device mouse`, então os controles do mouse serão automaticamente ativados ao executar um jogo que tenha controles do tipo "paddle" (como o *Super Breakout* por exemplo), ainda que você tenha usado a opção `-nomouse`.

O padrão é ativar os controles do mouse automaticamente ao rodar sistemas emulados com entradas do mouse (**-mouse_device mouse**).

Exemplo:

```
mame sbrkout -paddle_device mouse
```

Dica: Observe que estas configurações podem substituir as opções **-nomouse**, **-nojoystick** ou **-nolightgun**, dependendo das entradas presentes no sistema emulado.

5.1.28 Opções voltadas para a depuração

-[no]verbose / -[no]v

Este é o **modo loquaz**⁷, exibe todas as informações de diagnósticos disponíveis. Essas informações são úteis para apurar qualquer tipo de problemas com a sua configuração ou qualquer outra que possa aparecer. **IMPORTANTE:** favor rodar com `mame -verbose` e incluir a saída junto caso queira entrar em contato conosco para relatar um erro.

O valor predefinido é Desligado (`-noverbose`).

Exemplo:

```
mame ssf2tu -verbose
```

-[no]oslog

Escreve um arquivo `error.log` com mensagens de diagnósticos do sistema caso um esteja presente.

É predefinido que as mensagens de erro sejam enviadas para a saída padrão, geralmente é exibido no terminal, prompt de comando ou num arquivo de log de sistema. No *Windows*, caso um depurador esteja sendo usado como o depurador do *Visual Studio* ou *WinDbg*, as mensagens de erros serão enviadas para estes em vez de serem exibidas no terminal.

O valor predefinido é Desligado (`-nooslog`).

Exemplo:

```
mame mappy -oslog
```

-[no]log

Cria um arquivo chamado `error.log` que contém todos os registros de mensagens internas gerada pelo cerne do MAME e drivers de sistema. Isso pode ser usado ao mesmo tempo que `-oslog` para escrever os dados de saída de ambos ao mesmo tempo.

O valor predefinido é Desligado (`-nolog`).

Exemplo:

```
mame mappy -log
mame mappy -oslog -log
```

-[no]debug / -[no]d

Ativa o depurador embutido no MAME. É predefinido que o depurador entre em ação ao pressionar a tecla *til* ~⁸ e *acento grave* ` durante a emulação. Ele também entra direto no depurador depois que a emulação é redefinida, caso esta opção esteja ativa. Consulte *-debugger* para obter mais informações de como usar o depurador.

O valor predefinido é Desligado (`-nodebug`).

Exemplo:

```
mame indy_4610 -debug
```

⁷ Tagarela, que verbaliza muito, falador, barulhento. (Nota do tradutor)

⁸ Até que o teclado **ABNT-2** seja mapeado pela equipe do MAMEDev, essa tecla fica do lado esquerdo da tecla 1, logo abaixo da tecla ESQ. (Nota do tradutor)

-debugger <módulo>

Escolhe o módulo que será usado para depurar o sistema quando a opção *-debug* estiver ligada. A disponibilidade dos módulos de depuração dependem da plataforma e das opções de compilação.

Estes são os módulos compatíveis com esta opção:

windows

Depurador Win32 GUI (padrão no Windows). Compatível apenas no Windows.

qt

Depurador Qt GUI (padrão no Linux). É compatível no Windows, Linux e macOS, porém é incluído apenas no Linux. Defina a opção `USE_QTDEBUG=1` ao compilar o MAME para tornar disponível no Windows ou macOS.

osx

Depurador Cocoa GUI (padrão no macOS). Compatível apenas no macOS.

imgui

O depurador ImGui GUI é exibido na primeira janela do MAME. É preciso que a opção *-video* esteja definido como `bgfx`. É compatível com todas as plataformas que suportem a opção `BGFX`.

gdbstub

Funciona como um servidor de depuração remota para o depurador GNU GDB. Apenas um pequeno conjunto emulado das CPUs são compatíveis. Use a opção *-debugger_port* para definir a porta na interface de rede. É compatível com todas as plataformas com suporte a TCP/IP.

Exemplo:

```
mame ambush -debug -debugger qt
```

-debugscript <nome do arquivo>

Define um arquivo que vai conter a lista de comandos de depuração a serem executados no momento da inicialização.

O valor predefinido é `NULO` (nenhum comando).

Exemplo:

```
mame galaga -debugscript testscript.txt
```

-[no]update_in_pause

Ativa a atualização do bitmap inicial da tela enquanto o sistema estiver pausado. Isso significa que a opção de retorno **VIDEO_UPDATE** sempre será chamada durante a pausa, o que pode ser útil durante a depuração.

O valor predefinido é Desligado (`-nouupdate_in_pause`).

Exemplo:

```
mame indy_4610 -update_in_pause
```

-debugger_port <valor>

Define uma porta a ser usada pelo gdbstub debugger. Para usar, execute o gdb e faça o comando `target remote localhost:23946`.

A porta predefinida é 23946.

Exemplo:

```
mame indy_4610 -debugger_port 23999
```

-debugger_font <nome da fonte> / -dfont <nome da fonte>

Define o nome da fonte a ser usada nas janelas do depurador.

A fonte predefinida da janela é **Lucida Console**. A fonte predefinida do Mac (**Cocoa**) é o padrão de fonte de tamanho fixo do sistema (geralmente a fonte **Monaco**). A fonte padrão do Qt é **Courier New**.

Exemplo:

```
mame marble -debug -debugger_font "Comic Sans MS"
```

-debugger_font_size <pontos> / -dfontsize <pontos>

Define o tamanho da fonte a ser usada nas janelas do depurador em pontos.

O tamanho padrão da janela é de 9 pontos. O tamanho padrão do Qt é de 11 pontos. O tamanho padrão do Mac (**Cocoa**) é o tamanho padrão usado pelo sistema.

Exemplo:

```
mame marble -debug -debugger_font "Comic Sans MS" -debugger_font_size 16
```

-watchdog <tempo> / -wdog <tempo>

Ativa o temporizador watchdog interno que vai automaticamente matar o processo do MAME caso o tempo de duração definido em *<tempo>* passe caso não haja nenhuma atualização de quadro. Tenha em mente que alguns sistemas ficam parados por algum tempo durante o carregamento da tela, então *<duration>* deve ser grande o suficiente para levar esse tempo extra em consideração. Geralmente, um valor entre 10 e 30 segundos devem ser suficientes.

O valor predefinido é Nenhum.

Exemplo:

```
mame ibm_5150 -watchdog 30
```


5.1.29 Opções para a configuração da rede

-comm_localhost <endereço>

Definição para o endereço local. Este pode ser um endereço tradicional xxx.xxx.xxx.xxx ou um nome do host que possa ser resolvido

O valor predefinido é 0.0.0.0.

Exemplo:

```
mame arescue -comm_localhost 192.168.1.2
```

-comm_localport <porta>

Definição da porta local. Esta pode ser qualquer porta de comunicação tradicional como um valor inteiro (0-65535).

O valor predefinido é 15122.

Exemplo:

```
mame arescue -comm_localhost 192.168.1.2 -comm_localport 30100
```

-comm_remotehost <endereço>

Definição do endereço remoto. Este pode ser um endereço tradicional xxx.xxx.xxx.xxx ou um nome de host que possa ser resolvido.

O valor predefinido é 0.0.0.0.

Exemplo:

```
mame arescue -comm_remotehost 192.168.1.2
```

-comm_remoteport <porta>

Definição da porta remota. Esta pode ser qualquer porta de comunicação tradicional como um valor inteiro *non-signed* com 16-bit (0-65535).

O valor predefinido é 15122.

Exemplo:

```
mame arescue -comm_remotehost 192.168.1.2 -comm_remoteport 30100
```

-[no]comm_framesync

Sincroniza os quadros entre os hosts na rede.

O valor predefinido é Desligado (-nocomm_framesync).

Exemplo:

```
mame arescue -comm_remotehost 192.168.1.3 -comm_remoteport 30100 -comm_
↪framesync
```

5.1.30 Opções diversas

-[no]drc

Ativa o núcleo o DRC (recompilador dinâmico) da CPU visando uma velocidade máxima de emulação, caso esteja disponível.

Na recompilação dinâmica as instruções são traduzidas em tempo real e mais próxima possível do sistema que está sendo emulado com instruções reagrupadas em código de alto nível que então é compilado na linguagem nativa do sistema hospedeiro. A grande vantagem desta técnica é a melhor adequação do código gerado refletindo num melhor desempenho e eficiência durante a execução. Por outro lado, é preciso um poder de processamento muito maior para que isso seja possível.

O valor predefinido é Ligado (-drc).

Exemplo:

```
mame ironfort -nodrc
```

-[no]drc_use_c

Impõem o uso do DRC usando a infra-estrutura em código C.

O valor predefinido é Desligado (-nodrc_use_c).

Exemplo:

```
mame ironfort -drc_use_c
```

-[no]drc_log_uhl

Grave um registro descompilado DRC UHL num arquivo de registro (log).

O valor predefinido é (-nodrc_log_uhl).

Exemplo:

```
mame ironfort -drc_log_uhl
```

-[no]drc_log_native

Grave o DRC nativo e descompilado num registro de log em formato assembler.

O valor predefinido é Desligado (-nodrc_log_native).

Exemplo:

```
mame ironfort -drc_log_native
```

-bios <nome da bios>

Determina qual BIOS usar no sistema a ser emulado em sistemas que fazem uso de uma BIOS. A saída *-listbios* listará todos os nomes disponíveis das BIOS para o sistema, assim como a opção *-listxml*.

Não há valor predefinido (O MAME usará a primeira BIOS nativa do sistema que for encontrada, caso uma esteja disponível).

Exemplo:

```
mame mslug -bios unibios40
```

-[no]cheat / -[no]c

Ativa o cardápio de trapaças, exibindo uma lista de trapaças que ficam armazenadas num arquivo externo chamado `cheat.7z`. Essa opção também ativa as opções de turbo dos botões.

O valor predefinido é Desligado (`-nocheat`).

Exemplo:

```
mame kof97 -cheat
```

-[no]skip_gameinfo

Ao iniciar, faz com que o MAME não exiba a tela de informações do sistema iniciado.

O valor predefinido é Desligado (`-noskip_gameinfo`).

Exemplo:

```
mame samsho5 -skip_gameinfo
```

-uifont <nome da fonte>

Define o nome da fonte ou um nome do arquivo de fonte a ser usada na interface do usuário. Caso esta fonte não possa ser encontrada ou não puder ser carregada, o MAME usará a sua própria fonte embutida. Em algumas plataformas o *<nome da fonte>* (nome da fonte) pode ser um nome da fonte do sistema em vez de um arquivo fonte com extensão `.bdf`. Para o nome das fontes que tem espaço utilize o nome entre aspas (`-uifont "nome do arquivo da fonte"`).

O valor predefinido é `default` (O MAME usará a fonte nativa).

Exemplo:

```
mame -uifont "Comic Sans MS"
```

-ui <tipo>

Define o tipo de interface do usuário, as opções ficam entre `simple` a `mais simples` ou `cabinet` que é a mais completa.

O valor predefinido é `cabinet` (`-ui cabinet`).

Exemplo:

```
mame -ui simple
```

-ramsize / -ram <n>

Permite que seja alterado o tamanho padrão da RAM (caso exista suporte para tanto no driver).

Exemplo:

```
mame mac1c -ramsize 10M -hard1 mac761.chd
```

-[no]confirm_quit

Remove o aviso na tela *"Tem certeza que deseja encerrar?"* ao pressionar a tecla Esc.

O valor predefinido é Desligado (`-noconfirm_quit`).

Exemplo:

```
mame pacman -confirm_quit
```

-[no]ui_mouse

Exibe o ponteiro do mouse na interface do usuário do MAME.

O valor predefinido é Ligado (-ui_mouse).

Exemplo:

```
mame -ui_mouse
```

-language / -lang <idioma>

Especifique um idioma para ser usado na interface do usuário, os arquivos de tradução para cada idioma estão no caminho definido em **languagepath**.

Exemplo:

```
mame -language Portuguese_Brazil
```

-[no]nvram_save

Salva o conteúdo da NVRAM ao encerrar a emulação. Caso essa opção seja desligada, o conteúdo que foi gravado anteriormente não será apagado e qualquer alteração atual não será gravado. Ao desligar esta função, não será feito o salvamento dos dados nos arquivos .nv associados com alguns tipos de programas usados em cartuchos.

O valor predefinido é Ligado (-nvram_save).

Exemplo:

```
mame galaga88 -nonvram_save
```

5.1.31 Opções para uso com script

-autoboot_command / -ab "<comando>"

Uma cadeia de comandos que serão executados após a inicialização do sistema (entre aspas " "). Para emitir uma cotação para a emulação, use "" no comando. Usando \n irá criar uma nova linha, emitindo o que foi digitado antes como comando.

Exemplo:

```
mame c64 -autoboot_command "load ""$""",8,1\n"
```

-autoboot_delay <n>

Tempo de atraso (em segundos) para o **-autoboot_command**.

Exemplo:

```
mame c64 -autoboot_delay 5 -autoboot_command "load ""$""",8,1\n"
```

-autoboot_script / -script <nome_do_arquivo.lua>

Carrega e executa um scrip após a inicialização do sistema.

Exemplo:

```
mame ibm5150 -autoboot_script myscript.lua
```

-console

Ativa emulador do Console Lua.

O valor predefinido é Desligado (-noconsole).

Exemplo:

```
mame ibm5150 -console
```

-plugins

Ativa o uso de plug-ins Lua, para mais informações consulte o capítulo [PLUG-INS](#).

O valor predefinido é Ligado (-plugins).

Exemplo:

```
mame apple2e -plugins
```

-plugin <apelido do plugin>

Permite o uso de uma lista de plug-ins Lua separados por vírgula, para mais informações consulte o capítulo [PLUG-INS](#).

Exemplo:

```
mamealcon -plugin cheat,discord,autofire
```

-noplugin <apelido do plugin>

Permite desativar uma lista de plug-ins Lua separados por vírgula.

```
mamealcon -noplugin cheat
```

5.1.32 Opções do servidor HTTP

-http

Ativa o servidor de HTTP.

O valor predefinido é Desligado (-nohttp).

Exemplo:

```
mame -http
```

Nota: Até a última versão deste documento o comando não funciona.

-http_port <porta>

Define uma porta para o servidor HTTP.

O valor predefinido é 8080.

Exemplo:

```
mame apple2 -http -http_port 6502
```

Nota: Até a última versão deste documento o comando não funciona.

-http_root <diretório raíz>

Define a pasta raíz para os documentos do servidor HTTP.

O valor predefinido é web.

Exemplo:

```
mame apple2 -http -http_port 6502 -http_root c:\users\me\appleweb\root
```

Nota: Até a última versão deste documento o comando não funciona.

5.2 Configurações específicas para o Windows

Nesta seção descrevemos todas as opções de configuração disponível apenas para a versão nativa do MAME no Windows (não SDL).

5.2.1 Opções de desempenho

-priority <prioridade>

Define a prioridade de tarefas (*threads*) usadas pelo MAME. Não há nenhuma tarefa predefinida para que não haja interferência com outros aplicativos. Os valores válidos ficam entre -15 até 1 onde 1 é a maior prioridade.

O valor predefinido é 0 (**prioridade NORMAL**).

-profile [n]

Faz o uso de um perfil determinado por [n].

5.2.2 Configurações de tela inteira

-[no]triplebuffer / **-[no]tb**

Ativa ou não o "*triple buffering*", *buffering* é o nome da técnica ou função que faz o armazenamento prévio de dados numa memória preliminar, por analogia, funciona como uma caixa d'água para que o depois de cheia o fluxo na saída seja constante. Normalmente o MAME escreve "a seco" diretamente na tela sem fazer firulas com a memória preliminar. Porém com essa opção ativa o MAME cria e escreve seus ciclos intermediários e em ordem usando três memórias preliminares. Essa é uma maneira de tentar manter o fluxo contínuo de dados, evitando interrupções. Dentre as três, apenas a primeira é exibida, a segunda fica na espera sendo acumulada e a terceira fica sendo escrita constantemente. A opção **-triplebuffer** sobrescreve a opção **-waitvsync** caso a memória preliminar seja criada com sucesso.

Essa opção não funciona com **-video gdi**.

O valor predefinido é **Desligado (-notriplebuffer)**.

-full_screen_brightness <valor> / **-fsb** <valor>

Controla o brilho ou nível de preto da tela. Selecionando valores menores (até 0.1) produzirá uma tela mais escura, enquanto valores maiores (até 2.0) produzirão uma tela mais clara.

Note que nem todas as placa de vídeo são compatíveis com essa opção. Essa opção também não funciona com **-video gdi**.

O valor predefinido é 1.0.

-full_screen_contrast <valor> / **-fsc** <valor>

Controla o contraste ou nível de branco da tela. Selecionando valores menores (até 0.1) produzirá uma tela mais apagada, enquanto valores maiores (até 2.0) produzirão uma tela mais saturada.

Note que nem todas as placa de vídeo são compatíveis com essa opção. Essa opção também não funciona com **-video gdi**.

O valor predefinido é 1.0.

-full_screen_gamma <valor> / **-fsg** <valor>

Ajuste de gama da tela, faz o ajuste da escala de luminância da tela ajustando o contraste entre o claro e escuro. Essa opção não afeta a arte ou outras partes da tela.

Note que nem todas as placa de vídeo são compatíveis com essa opção. Essa opção não funciona com `-video gdi`.

O valor predefinido é 1.0.

5.2.3 Opções para o modo janela

-menu <valor>

Habilita uma barra de menu caso esteja disponível e implementado na interface do usuário.

O valor predefinido é 0.

-attach_window

Se anexa a uma janela arbitrária.

Não há valor predefinido.

5.2.4 Opções para a entrada de controle

-[no]dual_lightgun / **-[no]dual**

Controla se o MAME tenta ou não rastrear duas pistolas de luz que aparecem como um único mouse. Esta opção precisa que a opção *-lightgun* esteja ativada e a opção *-lightgunprovider* esteja definido como win32.

Esta opção suporta certas configurações mais antigas com duas pistolas de luz que funcionam ajustando a localização do ponteiro do mouse no momento em que um gatilho da pistola for ativado. Os gatilhos primário e secundário no primeiro revólver, correspondem ao primeiro e ao segundo botão do mouse, os gatilhos primário e secundário no segundo revólver, correspondem ao terceiro e ao quarto botão do mouse.

Caso tenha várias pistolas de luz conectadas, provavelmente será preciso ativar apenas a opção *-lightgun*, usar a opção padrão rawinput em *-lightgunprovider* e configurar cada pistola de luz individualmente.

O valor predefinido é **Desligado** (`-nodual_lightgun`).

5.3 Configurações específicas para as versões SDL

Nesta seção descreveremos as opções de configuração voltadas especificamente para qualquer versão MAME que for compatível com o SDL (incluindo o Windows caso o MAME tenha sido compilado com o SDL ao invés da sua forma nativa). Inicie o MAME com o comando `mame -v` para ver quais são os drivers que estão disponíveis para o seu sistema, abaixo um exemplo no Linux.

```
Available videodrivers: x11 wayland KMSDRM dummy
Current Videodriver: x11
  Display #0
    Renderdrivers:
      opengl          (0x0)
      opengles2       (0x0)
      software        (0x0)
Available audio drivers:
  pulseaudio
  alsa
  dsp
  jack
  disk
  dummy
...
Leave sdlwindow_init
Enter sdl_info::create
window: using renderer opengl
renderer: flag SDL_RENDERER_ACCELERATED
Leave renderer_sdl2::create
Audio: Start initialization
Audio: Driver is pulseaudio
Audio: frequency: 48000, channels: 2, samples: 256
```

Aqui temos o exemplo para o Windows das versões do MAME que forem compiladas com o SDL usando a opção `OSD=sdl`, para mais informações veja [Microsoft Windows](#).

```
Available videodrivers: windows dummy
Current Videodriver: windows
  Display #0
    Renderdrivers:
      direct3d         (0x0)
      direct3d11       (0x0)
      opengl           (0x0)
      opengles2        (0x0)
      software         (0x0)
Available audio drivers:
  wasapi
  directsound
  winmm
  disk
  dummy
...
Leave sdlwindow_init
Enter sdl_info::create
window: using renderer direct3d11
```

(continua na próxima página)

(continuação da página anterior)

```
renderer: flag SDL_RENDERER_ACCELERATED
Leave renderer_sd12::create
DirectSound: Primary buffer: 48000 Hz, 16 bits, 2 channels
```

Aqui temos as opções disponíveis para a customização em todas as versões SDL:

Hints:

SDL_FRAMEBUFFER_ACCELERATION	(NULL)
SDL_RENDER_DRIVER	(NULL)
SDL_RENDER_OPENGL_SHADERS	(NULL)
SDL_RENDER_SCALE_QUALITY	(NULL)
SDL_RENDER_VSYNC	(NULL)
SDL_VIDEO_X11_XVIDMODE	(NULL)
SDL_VIDEO_X11_XINERAMA	(NULL)
SDL_VIDEO_X11_XRANDR	(NULL)
SDL_GRAB_KEYBOARD	(NULL)
SDL_VIDEO_MINIMIZE_ON_FOCUS_LOSS	(NULL)
SDL_IOS_IDLE_TIMER_DISABLED	(NULL)
SDL_IOS_ORIENTATIONS	(NULL)
SDL_XINPUT_ENABLED	(NULL)
SDL_GAMECONTROLLERCONFIG	(NULL)
SDL_JOYSTICK_ALLOW_BACKGROUND_EVENTS	(NULL)
SDL_ALLOW_TOPMOST	(NULL)
SDL_TIMER_RESOLUTION	(NULL)
SDL_RENDER_DIRECT3D_THREADSAFE	(NULL)
SDL_VIDEO_ALLOW_SCREENSAVER	(NULL)
SDL_ACCELEROMETER_AS_JOYSTICK	(NULL)
SDL_MAC_CTRL_CLICK_EMULATE_RIGHT_CLICK	(NULL)
SDL_VIDEO_WIN_D3DCOMPILER	(NULL)
SDL_VIDEO_WINDOW_SHARE_PIXEL_FORMAT	(NULL)
SDL_VIDEO_MAC_FULLSCREEN_SPACES	(NULL)
SDL_MOUSE_RELATIVE_MODE_WARP	(NULL)
SDL_RENDER_DIRECT3D11_DEBUG	(NULL)
SDL_VIDEO_HIGHDPI_DISABLED	(NULL)
SDL_WINRT_PRIVACY_POLICY_URL	(NULL)
SDL_WINRT_PRIVACY_POLICY_LABEL	(NULL)
SDL_WINRT_HANDLE_BACK_BUTTON	(NULL)

Não há qualquer garantia que ao alterar qualquer uma destas opções traga alguma melhoria do desempenho, a sua sorte pode variar bastante dependendo do sistema operacional utilizado, da sua placa de vídeo e dos seus respectivos drivers.

No **Linux** e **macOS** você pode definir estes parâmetros como variáveis de ambiente no seu `~/ .bashrc` como por exemplo:

```
export SDL_FRAMEBUFFER_ACCELERATION=opengl
export SDL_RENDER_DRIVER=opengl
export SDL_RENDER_OPENGL_SHADERS=1
```

Antes do executável do MAME:

```
SDL_FRAMEBUFFER_ACCELERATION=opengl SDL_RENDER_DRIVER=opengl SDL_RENDER_OPENGL_SHADERS=1 ./
↪ mame
```

Ou então exportando estas opções para o ambiente, elas serão lidas durante a inicialização do MAME:

```
export SDL_FRAMEBUFFER_ACCELERATION=1 SDL_RENDER_DRIVER=opengl SDL_RENDER_OPENGL_SHADERS=1
```

Já para as versões do **Windows** você pode definir estas opções como variáveis do ambiente no prompt de comando antes de iniciar o MAME com os comandos:

```
set SDL_FRAMEBUFFER_ACCELERATION=1
set SDL_RENDER_DRIVER=direct3d11
set SDL_RENDER_OPENGL_SHADERS=1
```

Criar um arquivo **.BAT** com estas opções predefinidas dentro do diretório do MAME, exemplo de um `run.bat`:

```
@echo off
set SDL_FRAMEBUFFER_ACCELERATION=1
set SDL_RENDER_DRIVER=direct3d11
set SDL_RENDER_OPENGL_SHADERS=1
mame.exe
```

Ou então deixar isso disponível como variável do sistema, pressione as teclas WIN + R e execute o comando **sysdm.cpl**, siga para --> *Avançado* --> *Variáveis de Ambiente*, na parte de baixo da tela onde está escrito *Variáveis do sistema* clique em *Novo*, na próxima janela que aparecer adicione o *Nome da variável* que deseja definir, no campo *Valor* defina o valor apropriado. O valor para `SDL_RENDER_DRIVER` seria `direct3d11` e assim por diante, reinicie o computador ou encerre a sessão que estiver usando para que as alterações sejam aplicadas.

Novamente, não é garantia que ao definir estas opções você note alguma melhora no desempenho da emulação, tudo vai depender do hardware usado e seus respectivos drivers.

Para mais informações, consulte a página de [variáveis](#) do SDL.

5.3.1 Opções de desempenho

-[no]sdlvideofps

Ativa a saída de dados para benchmark no subsistema de vídeo SDL incluindo o driver de vídeo do seu sistema, o servidor X (caso seja aplicável) e stack OpenGL em modo `-video opengl`.

5.3.2 Opções de vídeo

-[no]centerh

Centraliza o eixo horizontal da tela.

O valor predefinido é Ligado (**-centerh**).

-[no]centerv

Centraliza o eixo vertical da tela.

O valor predefinido é Ligado (**-centerv**).

5.3.3 Configuração para tela inteira

-useallheads

Partilha a tela inteira com diferentes monitores.

O valor predefinido é none (nenhum).

5.3.4 Configurações específicas quando o driver de vídeo for software

-scalemode < none | hwblit | hwbest | yv12 | yv12x2 | yuy2 | yuy2x2 >

Modos de escala da família de espaços de cor, esta opção funciona apenas com **-video soft**.

O valor predefinido é none (nenhum).

5.3.5 Configurações para o mapeamento do teclado

-keymap

Permite que você ative o uso de um mapa de teclado personalizado.

O valor predefinido é Desligado (**-nokeymap**).

-keymap_file <arquivo>

Use em conjunto com **-keymap**, permite que você escolha um arquivo com um mapa de teclado customizado, atualmente o MAME já vem com um mapa de teclado para o teclado ABNT2 chamado **km_br_LINUX.map** no diretório **keymaps**. Um mapa é útil para que o mapeamento das teclas já predefinidas coincidam com o mapa de um teclado ABNT2 por exemplo, assim a tecla ~ (til) que fica acima da tecla Tab no teclado ANSI Americano pode ser remapeado para a tecla que fica do lado direito da tecla Ç (cê-cedilha) num teclado ABNT2.

O valor predefinido é `keymap.dat`.

5.3.6 Configurações de entrada SDL

-enable_touch

Ativa o suporte para entrada por toque. Quando esta opção for desativada, será usada a entrada do mouse simulada por dispositivos de toque.

O valor predefinido é Desligado (**-noenable_touch**).

-sixaxis

Use um tratamento especial para lidar com os controles *SixAxis* do PS3. Pode causar um comportamento indesejado com outros controladores. Apenas afeta quando usado com a opção *-joystickprovider sdljoy*.

O valor predefinido é Desligado (**-nosixaxis**).

-[no]dual_lightgun / -[no]dual

Controla se o MAME tenta ou não rastrear duas arma de luz para que apareçam como um único mouse. Para isso a opção *-lightgun* precisa estar ativada e a opção *-lightgunprovider* deve ser definida como `sdl`.

Esta opção oferece suporte a configurações de duas armas que funcionam definindo o local do ponteiro do mouse no momento que um acionador da arma for ativado. Os acionadores primário e secundário da primeira arma correspondem ao primeiro e segundo botões do mouse, já os acionadores primário e secundário do segundo revólver correspondem ao terceiro e quarto botões do mouse.

O valor predefinido é Desligado (**-nodual_lightgun**).

5.3.7 Mapeamento da pistola de luz SDL

```
-lightgun_index1 <nome>
-lightgun_index2 <nome>
...
-lightgun_index8 <nome>
```

Nome do dispositivo ou a ID de um determinado slot para a pistola de luz.

5.3.8 Opções para a configuração dos drivers

-videodriver < x11 | directfb | ... | auto >

Define um driver de vídeo SDL a ser usado, a disponibilidade de alguns destes drivers depende do sistema operacional.

O valor predefinido é auto

-renderdriver < opengl | directfb | ... | auto >

Define o driver de renderização SDL a ser usado, a disponibilidade de alguns destes drivers depende do sistema operacional.

O valor predefinido é auto

-audiodriver < pulseaudio | alsa | arts | ... | auto >

Define o driver de áudio SDL a ser usado, a disponibilidade de alguns destes drivers depende do sistema operacional.

O valor predefinido é auto

-gl_lib <driver>

Define o **libGL.so** alternativo a ser usado.

O valor predefinido para o sistema é auto

5.4 Índice das opções para a linha de comando

Este é o índice quase completo do MAME das opções para a linha de comando, muito conveniente para a localização rápida de comandos. O MAME possui ainda algumas opções disponíveis e não documentadas, a opção *-showusage* pode dar um resumo de algumas opções que não estejam aqui listadas.

5.4.1 Opções universais de comandos

Opções de ajuda e verificação

help
validate
verifyroms
verifysamples
verifysoftware
verifysoftlist

Opções de configuração

createconfig
showconfig
showusage

Opções para listagem

listxml
listfull
listsource
listclones
listbrothers
listcrc
listroms
listbios
listsamples
romident
listdevices
listslots
listmedia
listsoftware
getsoftlist

Opções relacionadas ao que é exibido na tela

uimodekey
controller_map
background_input
uifontprovider
keyboardprovider
mouseprovider
lightgunprovider
joystickprovider

Opções de MIDI e rede

midiprotector
listmidi
midin
midout
listnetwork
networkprovider

Opções de saída das notificações de tela

output

Opções para a configuração

noreadconfig

Opções para a configuração dos diretórios principais

homepath
rompath
hashpath
samplepath
artpath
ctrlrpath
inipath
fontpath
cheatpath
crosshairpath
pluginspath
languagepath
swpath
cfg_directory
nvram_directory
input_directory

state_directory
snapshot_directory
diff_directory
comment_directory
share_directory

Opções para a gravação e a reprodução do estado da emulação

[no]rewind / rewind
rewind_capacity
statename
state
[no]autosave
playback
[no]exit_after_playback
record

Opções para a gravação de áudio e vídeo

mngwrite
aviwrite
wavwrite

Opções para a captura da tela

snapname
snapsize
snapview
[no]snapbilinear

Opções relacionadas ao desempenho e a velocidade da emulação

[no]autoframeskip
frameskip
seconds_to_run
[no]throttle
[no]sleep
speed
[no]refreshspeed
numprocessors
bench
[no]lowlatency

Opções para a rotação da tela

[no]rotate
[no]ror
[no]rol
[no]autoror
[no]autorol
[no]flipx
[no]flipy

Opções para a configuração de vídeo

video
numscreens
[no]window
[no]maximize
[no]keepaspect
[no]unevenstretch
[no]unevenstretchx
[no]unevenstretchy
[no]autostretchxy
[no]intoverscan
[no]intscalex
[no]intscaley
[no]waitvsync
[no]syncrefresh
prescale
[no]filter
[no]burnin

Opções para a configuração da tela inteira

[no]switchres

Opções de vídeo para uso com janelas individuais

screen
aspect
resolution
view

Opções para uso com as ilustrações

[no]artwork_crop
fallback_artwork
override_artwork

Opções para os ajustes de imagem da tela

brightness
contrast
gamma
pause_brightness
effect

Opções para sistemas que usem gráficos vetoriais

beam_width_min
beam_width_max
beam_intensity_weight
beam_dot_size
flicker

Opções para a depuração de vídeo OpenGL

[no]gl_forcepow2texture
[no]gl_notexturerect
[no]gl_vbo
[no]gl_pbo

Opções de vídeo OpenGL GLSL

[no]gl_glsl
gl_glsl_filter
glsl_shader_mame[0-9]
glsl_shader_screen[0-9]

Opções para a configuração do áudio

samplerate
[no]samples
[no]compressor
volume
speaker_report
sound
audio_latency
pa_api
pa_device
pa_latency

Opções para as configurações de diferentes entradas

[no]coin_lockout
ctrlr
[no]mouse
[no]joystick
[no]lightgun
[no]multikeyboard
[no]multimouse
[no]steadykey
[no]ui_active
[no]offscreen_reload
joystick_map
joystick_deadzone
joystick_saturation
joystick_threshold
[no]natural
[no]joystick_contradictory
coin_impulse

Opções de entrada ativadas automaticamente

paddle_device
adstick_device
pedal_device
dial_device
trackball_device
lightgun_device
positional_device
mouse_device

Opções voltadas para a depuração

[no]verbose
[no]oslog
[no]log
[no]debug
debugger
debugscript
[no]update_in_pause
watchdog
debugger_port
debugger_font
debugger_font_size

Opções para a configuração da rede

comm_localhost
comm_localport
comm_remotehost
comm_remoteport
[no]comm_framesync

Opções diversas

[no]drc
[no]drc_use_c
[no]drc_log_uml
[no]drc_log_native
bios
[no]cheat
[no]skip_gameinfo
uifont
ui
ramsize
[no]confirm_quit
[no]ui_mouse
language
[no]nvram_save

Opções para uso com script

autoboot_command
autoboot_delay
autoboot_script
[no]console
[no]plugins
plugin
noplugin

Opções do servidor HTTP

http
http_port
http_root

Opções do PortAudio

pa_api
pa_device
pa_latency

5.4.2 Configurações específicas para o Windows

Esta seção descreve todas as opções de configuração compatíveis com o MAME do Windows (não SDL).

Opções de desempenho

priority
profile

Configurações de tela inteira para o Windows

[no]triplebuffer
full_screen_brightness
full_screen_contrast
full_screen_gamma

Opções para o modo janela

menu
attach_window

Opções para a entrada de controle para o Windows

[no]dual_lightgun

5.4.3 Configurações específicas para versões SDL

Esta seção contém opções de configuração que são específicas para as versões SDL compatíveis (incluindo versões Windows que foram compiladas com SDL em vez da sua versão nativa).

Opções relacionadas ao desempenho (SDL)

[no]sdlvideofps

Opções de vídeo (SDL)

[no]centerh

[no]centerv

Configuração para tela inteira (SDL)

useallheads

Configuração para diferentes tipos de espaços de cor para vídeo (SDL)

scalemode

Configurações para o mapeamento de teclado (SDL)

keymap

keymap_file

Configurações para o mapeamento da entrada (SDL)

[no]enable_touch

[no]sixaxis

[no]dual_lightgun

Mapeamento da pistola de luz (SDL)

lightgun_index

Opções para a configuração dos drivers (SDL)

videodriver
renderdriver
audiodriver
gl_lib

- *Introdução*
- *Usando os plug-ins*
- *Plug-ins inclusos*

6.1 Introdução

Os plug-ins externos oferecem funcionalidades adicionais ao MAME, eles foram escritos de maneira a se comunicarem com programas ou plataformas externas, jogar jogos de forma autônoma, exibir estruturas internas do jogo, como [hitbox](#)¹, fornecer interfaces alternativas ao usuário, e realizar testes automáticos da emulação. Consulte [INTERFACE PARA SCRIPTS LUA](#) para obter mais informações sobre a API Lua do MAME.

6.2 Usando os plug-ins

Para ativar os plug-ins é preciso ligar a opção [plugins](#) e confira se a opção [opção pluginspath](#) inclui o caminho a pasta onde os seus plug-ins estão. Também é possível configurar os plug-ins num arquivo INI ou através da linha de comando. Ao entrar em *Configurações --> Configuração dos diretórios --> plugins* é possível definir o caminho dos seus plug-ins.

Diversos plug-ins precisam armazenar configurações e/ou dados, a opção [homepath](#) define onde estes dados serão armazenados (é predefinido que seja armazenado na mesma pasta de trabalho do MAME). É possível alterar esta opção em *Configurações --> Configuração dos diretórios --> Dados do plug-in*.

Antes de tentar ligar ou desligar um plug-in específico, tenha certeza que ele está ativo, uma vez ativo talvez seja preciso encerrar e iniciar o MAME novamente para que o plug-in surta efeito, não é preciso fazer isso

¹ São regiões delimitadoras que quando ultrapassadas o jogo identifica um contato. Consulte [este exemplo](#) para mais informações e ilustrações sobre o assunto.

para o plug-in de **Trapaça**, porém, outros podem precisar dessa reinicialização. Também é possível usar a opção *plugin* através da linha de comando ou alterando as configurações no arquivo `plugi.ini`.

Caso algum plug-in ativo precise de uma configuração adicional ou precise mostrar algum tipo de informação uma **Opção do plug-in** deverá aparecer no menu principal (acessado através da tecla Tab durante a emulação).

6.3 Plug-ins inclusos

MAME inclui vários plug-ins que proporcionam uma funcionalidade útil e servem como exemplo de código que pode ser usado como um ponto de partida ao escrever os seus próprios plug-ins.

6.3.1 Autofire Plugin

- *Introdução*
- *Configurando os botões turbo*
- *Observações e possíveis imprevistos*

Introdução

O *autofire plugin* permite simular um botão turbo (disparo automático), pode ser útil para pessoas com alguma deficiência ou lesões em jogos de tiro ou onde é preciso pressionar o botão repetidas vezes, e pode também auxiliar na prevenção de lesões por esforço repetitivo por exemplo.

Para configurar o plug-in pressione Tab durante a emulação, selecione *Opções dos plug-ins --> Turbo*. O MAME não vem com nenhum botão turbo configurado, ao clicar duas vezes em *Adiciona um botão turbo*, você tem as opções para a configurações do botão, consulte *Configurando os botões turbo* para mais detalhes dessa configuração. Quando não quiser mais a configuração, destaque o botão que deseja excluir e clique em Del.

Todas as configurações do turbo são salvas na pasta **autofire** dentro da pasta designada para os dados do plug-in (consulte *homepath option*). Ali ficam armazenados arquivos de configuração para cada sistema, a nomenclatura usada é o nome da ROM com a extensão `.cfg`. Por exemplo, a configuração de um botão turbo para **Super-X** é salvo como `superx.cfg`, o conteúdo do arquivo tem o formato JSON.

Configurando os botões turbo

Ambas as opções para adicionar um novo botão turbo como para alterar as suas configurações são as mesmas.

Depois de entrar na opção *Adiciona um botão turbo*, clique duas vezes em *Entrada*, dessa lista, faça um clique duplo no botão que deseja ativar o turbo, em seguida escolha a *Tecla de atalho*, esta tecla pode ser uma tecla do teclado ou um botão qualquer do seu controle. Note porém que você não pode usar o mesmo botão já predefinido pelo MAME.

As opções *Quadros ligados* e *Quadros desligados* definem a quantidade dos quadros que o botão deve permanecer pressionado e quando deve ser liberado, o ajuste pode ser feito com as teclas esquerda / direita do teclado ou usando as setas ao lado da opção, para retornar aos valores originais, use a tecla Del. Ao concluir a configuração clique em *Crie* para criar o seu botão turbo.

Quanto menor o valor, mais rápido é o tubo, quanto maior o valor, mais lento ele fica. Observe que nem todos os sistemas podem aceitar valores muito baixos (rápido), para que o turbo surta efeito no sistema Alcon por exemplo, é preciso configurar os valores como 2 **quadros ligados** e 2 **quadros desligados**. Ao concluir a configuração, pressione Tab para retornar para a emulação. Experimente diferentes valores para obter os melhores resultados.

Ao adicionar um novo botão de turbo, há uma opção *Cancelar* que muda para *Criar* depois que você define a entrada e a tecla de atalho. Selecione *Criar* para concluir a criação do botão de turbo e retornar à lista dos botões de turbo. O novo botão de turbo será adicionado ao final da lista. Pressione a tecla *UI Back* (tecla :kbd`Esc`) ou selecione *Cancelar* antes de definir a entrada/tecla de atalho para retornar ao menu anterior sem criar nenhum botão turbo.

Se durante o teste o turbo não agir da forma desejada, retorne para a opção e clique duas vezes no botão para alterar as suas configurações.

Observações e possíveis imprevistos

Os botões turbo agem como se estivessem conectados em paralelo com as entradas convencionais do MAME. Isso significa que caso defina um atalho para o botão turbi no mesmo botão ou chave que também é atribuída diretamente a uma das entradas emuladas, você pode obter resultados inesperados. Usando Gradius como exemplo:

- Vamos supor que você atribuiu o **botão 1** do seu controle como tiro e também definiu ele como turbo. Ao manter o botão de tiro pressionado ele nunca será liberado e o turbo para de funcionar, o mesmo acontece caso defina um outro botão qualquer como turbo (**botão 3** por exemplo) e mantenha o **botão 1** e o **botão 3** pressionados ao mesmo tempo.
- Caso tenha definido o **botão 3** como turbo e ele também esteja definido como *powerup*, a ação vai sempre ativar o *powerup* pois internamente é como se este botão estivesse sempre pressionado.

Por isso que é referível escolher um outro botão qualquer e não aqueles que já estejam sendo usado pela emulação.

Usar um outro botão como turbo é útil, em jogos onde é preciso manter o botão de tiro pressionado para liberar um tiro com maior poder de fogo (*Raiden Fighters* por exemplo), assim, usar um outro botão com a função turbo acaba sendo uma necessidade¹.

6.3.2 Console Plugin

Este plug-in permite a interatividade com o console Lua do MAME, não é usado de forma direta. Consulte a opção *console* para ativar. Consulte também o capítulo *INTERFACE PARA SCRIPTS LUA* para obter mais informações.

¹ Nas versões mais antigas do MAME, o processo era muito mais simples, bastava ativar o turbo no botão que queria usar, porém a função foi removida no [MAME 0.216](#). Com a função removida, não dá mais para usar o botão que já funcionava como tiro, agora é preciso escolher um botão diferente. (o_0)!

6.3.3 Data Plugin

Este plug-in de dados carrega informações a partir de diferentes arquivos de suporte externo para que possam ser exibidos pelo MAME. Quando ativado, as informações são mostradas na aba *Informações* à direita nos menus de seleção do sistema e software. As informações pode ser mostradas clicando no botão da barra de ferramentas do sistema, software ou escolhendo *Visualização da DAT externa* no menu principal durante a emulação (este item não aparecerá caso o plugin de dados não esteja ativado ou caso não haja informações disponíveis para o sistema emulado no momento).

Para definir a pasta onde os plug-ins procuram por estes dados de suporte, escolha *Configurações --> Configurações dos diretórios --> Dats*. Também é possível definir o seu `historypath` no seu arquivo `ui.ini`.

Pode haver um atraso/demora na aparição da interface do MAME ao carregar grandes arquivos de dados tais como o **history.xml**. Então, seja paciente ao iniciar o MAME após atualizar ou adicionar novos arquivos de dados para serem carregados na inicialização.

Os seguintes arquivos são compatíveis com o MAME:

history.xml

Do [Gaming-History](#) (antigamente conhecido como Arcade-History)

mameinfo.dat

Do [MASH's MAMEINFO](#)

messinfo.dat

Do [progetto-SNAPS MESSINFO.dat](#)

gameinit.dat

Do [progetto-SNAPS GameInit.dat](#)

command.dat

Do [progetto-SNAPS Command.dat](#)

score3.htm

[Top Scores](#) da página [MAME Action Replay Page](#)

mameinfo.dat / command.dat Japonês

Do [MAME E2J](#)

sysinfo.dat

Do agora defunto site Progetto EMMA

story.dat

Do agora defunto site MAMESCORE

Caso instale o [hi2txt](#) também é possível exibir os maiores placares com o *data plugin* e salvar o placar da memória não volátil ou através do *hiscore support plugin* para jogos compatíveis.

Observe que não é possível, por exemplo, usar diferentes arquivos `mameinfo.dat` feitos para diferentes idiomas ao mesmo tempo.

O *data plugin* cria um arquivo chamado `history.db` na pasta **data** dentro da pasta `plugin` (consulte a opção [homepath](#)). Este arquivo armazena as informações dos arquivos de suporte em formato SQLite3 visando uma rápida leitura.

6.3.4 Discord Presence Plugin

O *Dicord presence plugin* funciona com o Discord para Windows, macOS ou Linux, ele serve para publicar a sua atividade no MAME. A atividade está definida como *In menu* (no menu) caso esteja usando o sistema ou escolhendo um programa no menu de seleção, *Playing* (jogando) caso uma emulação esteja em andamento ou *Paused* (parado) caso a emulação esteja parada.

A predefinição é para mostrar o nome do sistema e a descrição do software quando for possível.

6.3.5 Dummy Test Plugin

Este é um plug-in de exemplo que mostra como definir os metadados do plug-in, os retornos dos registros e mostra um menu simples. Exibe mensagens das condições gerais e adiciona uma entrada **Dummy** no menu *Opções do plug-in*.

6.3.6 GDB Stub Plugin

O *GDB stub plugin* age como um servidor de depuração remota para o depurador GNU (GDB). Permite que você se conecte ao MAME e faça a depuração nos sistemas compatíveis usando o GDB. O plug-in escuta na porta 2159 do localhost (127.0.0.1). Apenas a família 80386 (i386) é compatível com este plug-in.

Consulte a opção *debugger* para outras implementações com o depurador remoto GDB compatível com mais CPUs e configuração das portas.

6.3.7 Hiscore Support Plugin

Este plug-in salva e restaura o placar dos jogos que originalmente não salvavam tais pontuações na memória não volátil. Observe que este plug-in altera diretamente o conteúdo da memória sem qualquer coordenação com o software que está sendo emulado, portanto, altera o seu comportamento e pode ter efeitos indesejáveis, quebrando a jogabilidade ou fazendo com que o software que está sendo emulado no momento, trave.

O plugin inclui um arquivo `hiscore.dat` que contém as informações de como salvar e restaurar o placar nos sistemas compatíveis. Este arquivo deve ser mantido atualizado quando as definições do sistema mudarem no MAME.

O placar pode ser salvo automaticamente ao encerrar a emulação ou segundos depois que o placar for atualizado na memória. Para alterar a configuração pressione Tab durante a emulação e vá em *Opções do plugin* --> *Hiscore Support* e altere as opções do *Save scores*.

Um arquivo com o nome da ROM e a extensão `.hi` ficam gravados no diretório **hiscore** dentro da pasta de dados do plug-in, consulte a opção *homepath*). O conteúdo do arquivo tem o formato JSON.

6.3.8 Input Macro Plugin

- *Introdução*
- *Editando as macros*
- *Macros de exemplo*
 - *Turbo para Raiden*
 - *Trapaceando na corrida em Track & Field*
 - *Street Fighter II*
 - *Bad Dudes vs. Dragonninja*
 - *Top Ranking Stars*

Introdução

Este plug-in permite que uma série de comandos sejam executados ao pressionar um botão ou uma combinação deles. Isso pode ser útil para pessoas com algum tipo de deficiência ou lesões que não conseguem realizar algumas sequências. Também pode ser usado como uma forma trapacear nos jogos que exigem sequências rápidas como os eventos de corrida no *Track & Field* ou em minijogos como o *Daisu-Kiss*.

Para configurar o plug-in pressione Tab durante a emulação, selecione *Opções dos plug-ins --> Macros*. Caso existam macros para o sistema atual elas estarão listadas na tela,

Selecione uma macro para editá-la ou escolha *Add macro* para configurar uma nova entrada. Consulte [Editando as macros](#) para obter detalhes de como criar a sua. Para excluir uma macro da lista destaque-a no menu e pressionando a tecla Del.

As macros são armazenadas na pasta **inputmacro** na mesma pasta do executável do MAME ou o que estiver definido na opção *homepath option*. Um mesmo arquivo armazena diferentes macros para o mesmo sistema e mantém a mesma nomenclatura do nome da ROM seguido da extensão .cfg. Por exemplo, as macros de entrada para Daisu-Kiss serão salvas no arquivo daiskiss.cfg dentro da pasta **inputmacro**. O conteúdo do arquivo tem o formato JSON.

clique em *Nova macro* para criar a sua macro.

Editando as macros

Ambas as opções para adicionar uma nova macro como para alterar as suas configurações são as mesmas.

As macros são criadas numa sequência de *etapas*, opcionalmente cada *etapa* aguarda por um determinado tempo (quadros), depois ativa a próxima etapa e assim sucessivamente, dependendo do sistema, algumas etapas precisam ter um intervalo entre eles (atraso), caso contrário o sistema não consegue acompanhar o comando que foi passado. A macro depois concluída tem a opção de não fazer nada (libera os controles), pode ser continuada ou repetir a sequência novamente.

- *Nome*

Define o nome da macro. Pressione Enter no teclado ou o **botão 1** do controle para editar o nome que estiver ali ou Espaço para limpar antes de inserir um novo nome. Use as teclas direcionais para mover o cursor e a tecla Esc (*UI_Back*) para cancelar a edição.
- *Ativação*

Define uma tecla, um botão ou a combinação deles para ativar a macro. Tenha ciência que as entradas tradicionais continuam valendo, assim sendo, procure escolher os botões/teclas de atalho que não tenham nenhuma função dentro da emulação.

- *Ao soltar*

Define o que deve acontecer quando a ativação da sequência for iniciada antes da conclusão da macro. *Pare* interrompe a macro assim que ela terminar. *Conclua a macro* a macro será processada até o última etapa.

- *Enquanto estiver pressionado*

Use para definir o que deve acontecer caso o botão ou a tecla de ativação seja mantida pressionada depois que a macro for concluída.

- *Libera*

Executa a macro e não ativa novamente até que a ativação aconteça novamente.

- *Mantenha a etapa <n> ativa*

O **n** é o número do último passo da macro, neste caso, o passo final da macro permanece ativa até que a ativação seja liberada, por exemplo, caso o último passo seja um botão de tiro, ele vai funcionar como um turbo até que o botão seja liberado.

- *Repita até a etapa <n>*

o **n** é o número do passo que deseja que seja repetido, incluindo o atraso, caso a ativação se mantenha mantida após a conclusão do passo final.

Cada passo possui um atraso, duração e configuração da entrada:

- *Atraso (quadros)*

Define a quantidade de quadros que se deve aguardar antes que a ação seja feita, ou seja, durante o atraso nenhuma entrada é ativada. É possível redefinir o valor da configuração para zero ao pressionar a tecla Del.

- *Duração (quadros)*

Define a quantidade de tempo (quadros) que o botão ou direcional deve ser mantido pressionado antes de prosseguir para o próximo passo (ou completá-lo caso ele seja o último). Alguns jogos registram o comando logo nos primeiros quadros, já outros precisam de 3 ou mais quadros para registrar a ação. É possível redefinir o valor da configuração para zero ao pressionar a tecla Del.

- *Entrada*

Define a entrada que será ativada no passo, no momento apenas as entradas digitais são compatíveis. Clique em *Nova entrada* para escolher uma entrada de uma lista (esta opção aparece apenas depois de definir a primeira entrada). Caso o passo tenha diversas entradas, selecione uma delas use a tecla Del para excluir (todas as etapas devem ter pelo menos mais de uma entrada, caso contrário não é possível excluir a única entrada existente na etapa).

- *Excluí etapa*

Em macros com mais de uma etapa, use esta opção para excluí-la (esta opção não aparece se a macro possuir apenas uma etapa). Verifique antes se as configurações das opções *Ao soltar* e *Enquanto estiver pressionado* estão corretas **depois** de excluir uma etapa.

Para adicionar uma etapa, selecione *Adiciona uma etapa na posição* (depois dos passos já existentes), use as teclas direcionais Esquerda / Direita ou clique com o mouse nas setas para definir a posição desejada para a inserção do novo passo, pressione então Enter (ou clique duas vezes no item) para adicionar a nova etapa.

Será solicitado que você defina a primeira entrada para a nova etapa. Lembre-se de verificar as configurações *Ao soltar* e *Enquanto estiver pressionado* depois de adicionar as etapas. O item *Adiciona uma etapa na posição* só aparecerá depois que for definido a primeira entrada para a etapa inicialmente criada durante a criação de uma nova macro.

Ao criar uma nova macro há uma opção *Cancela* que muda para *Cria* depois de definir a ativação e a primeira entrada para a etapa inicial. Selecione *Cria* para finalizar a criação da macro e retornar à lista de entradas para a macro. A nova macro será adicionada no final da lista. Pressione a tecla Esc ou selecione *Cancela* antes de definir a ativação/entrada para retornar ao menu anterior sem criar a nova macro.

Ao editar uma macro já existente, selecione *Feito* ou pressione a tecla Esc para retornar à lista de macros de entrada, as alterações já entram em vigor imediatamente.

Macros de exemplo

Turbo para Raiden

Permite a funcionalidade de turbo ao jogador 1 usando a barra de espaço. O mesmo efeito pode ser obtido usando o *Autofire Plugin*, porém, o exemplo abaixo demonstra o uso de uma macro:

- **Nome:** *Turbo P1*
- **Ativação:** Tecla Espaço
- **Ao soltar:** *Pare*
- **Enquanto estiver pressionado:** *Repita até a etapa 2*
- **Etapa 1:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *2*
 - **Entrada 1:** *Jogador 1 botão 1*
- **Etapa 2:**
 - **Atraso (quadros):** *4*
 - **Duração (quadros):** *2*
 - **Entrada 1:** *Jogador 1 botão 1*

A primeira etapa não possui nenhum atraso para que o disparo comece assim que a barra de espaço seja pressionada. O segundo passo tem um atraso suficiente para garantir que o jogo reconheça o botão que está sendo pressionado e novamente liberado. O segundo passo também é repetido desde que a barra de espaço se mantenha pressionada.

Trapaceando na corrida em Track & Field

Isso permite que você corra segurando apenas um botão no *Track & Field* da Konami. Isso tira a maior parte da destreza (e da diversão) do jogo:

- **Nome:** *Corrida J1*
- **Ativação:** Tecla Shift
- **Ao soltar:** *Pare*
- **Enquanto estiver pressionado:** *Repita até a etapa 2*

- **Etapa 1:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 botão 1*
- **Etapa 2:**
 - **Atraso (quadros):** 1
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 botão 3*
- **Etapa 3:**
 - **Atraso (quadros):** 1
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 botão 1*

Esta macro alterna rapidamente entre os botões 1 e 3 do jogo fazendo com que você consiga correr no jogo.

Street Fighter II

Esta macro permite que você faça o *Shoryuken (Dragon punch)* ao pressionar a tecla M com o jogador 1 estando do lado esquerdo da tela, não se esqueça de clicar em *Feito* ao concluir:

Nota: A partir da versão **0.237** os direcionais e muitas outras opções já estão traduzidas para o nosso idioma, caso as opções dos direcionais estejam diferentes, atualize a sua tradução com a versão compatível com esta documentação baixando o arquivo [strings.mo](#) e substituindo o arquivo que está na pasta `language\Portuguese_Brazil`.

- **Nome:** *Jogador 1 Shoryuken SF*
- **Ativação:** Tecla M
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapa 1:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 direita*
- **Etapa 2:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 baixo*
- **Etapa 3:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 1

- **Entrada 1:** *Jogador 1 baixo*
- **Entrada 2:** *Jogador 1 direita*
- **Entrada 3:** *P1 Jab Punch*

A macro realiza o golpe e caso mantenha a tecla pressionada, nada acontece.

Esta é a macro para o *Hadouken* com *soco fraco* ao pressionar a tecla N com o jogador 1 estando do lado esquerdo da tela, não se esqueça de clicar em *Feito* ao concluir:

- **Nome:** *Jogador 1 Hadouken SF*
- **Ativação:** Tecla N
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Etapas:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *1*
 - **Entrada 1:** *Jogador 1 baixo*
 - **Etapas:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *1*
 - **Entrada 1:** *Jogador 1 baixo*
 - **Entrada 2:** *Jogador 1 direita*
 - **Etapas:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *1*
 - **Entrada 1:** *Jogador 1 direita*
 - **Entrada 2:** *P1 Strong Punch*

Esta macro é utilizada pelo personagem **Guile** para soltar o *Sonic Boom* com *soco médio* ao pressionar a tecla B com o jogador 1 estando do lado esquerdo da tela, não se esqueça de clicar em *Feito* ao concluir:

- **Nome:** *Jogador 1 Sonic Boom*
- **Ativação:** Tecla B
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Etapas:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *60*
 - **Entrada 1:** *Jogador 1 esquerda*
 - **Etapas:**
 - **Atraso (quadros):** *0*
 - **Duração (quadros):** *2*

- **Entrada 1:** *Jogador 1 direita*
- **Entrada 2:** *P1 Strong Punch*

Esta macro faz o tal "facão do Guile" (*Flash Kick*) com *chute fraco* ao pressionar a tecla V com o jogador 1 estando do lado esquerdo da tela, não se esqueça de clicar em *Feito* ao concluir:

- **Nome:** *Jogador 1 Flash Kick*
- **Ativação:** Tecla V
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 60
 - **Entrada 1:** *Jogador 1 baixo*
- **Etapas:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 2
 - **Entrada 1:** *Jogador 1 cima*
 - **Entrada 2:** *P1 Short Kick*

Esta macro é utilizada pelo personagem **Zanguief** para fazer o nosso conhecido "*Pilão Giratório*", também conhecido como *Spinning Piledriver* e *Screw Pile Driver* com *soco médio* ao pressionar a tecla C com o jogador 1 estando do lado esquerdo da tela, não se esqueça de clicar em *Feito* ao concluir:

- **Nome:** *Jogador 1 Screw Pile Driver*
- **Ativação:** Tecla C
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Atraso (quadros):** 0
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 direita*
- **Etapas:**
 - **Atraso (quadros):** 1
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 baixo*
- **Etapas:**
 - **Atraso (quadros):** 1
 - **Duração (quadros):** 1
 - **Entrada 1:** *Jogador 1 esquerda*
- **Etapas:**

- Atraso (quadros): 1
- Duração (quadros): 1
- Entrada 1: *Jogador 1 cima*
- Entrada 2: *P1 Strong Punch*

Bad Dudes vs. Dragonninja

Esta macro faz o personagem dar um chute giratório, escolha o botão de atalho que achar mais apropriado para o seu controle.

- **Nome:** *Giratória*
- **Ativação:** Tecla X
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Etapas 1:**
 - Atraso (quadros): 0
 - Duração (quadros): 4
 - Entrada 1: *P1 Jump*
 - **Etapas 2:**
 - Atraso (quadros): 0
 - Duração (quadros): 3
 - Entrada 1: *P1 Jump*
 - Entrada 2: *P1 Attack*

Top Ranking Stars

Esta macro é um exemplo de como ativar golpes especiais em jogos que precisam que os comandos sejam mantidos pressionados por mais tempo para que o comando seja corretamente identificado e executado. O comando é para o especial "Stardust" do personagem *Shouichi Kanou*. Escolha o botão de atalho que achar mais apropriado para o seu controle.

- **Nome:** *Stardust*
- **Ativação:** Tecla S
- **Ao soltar:** *Conclua a macro*
- **Enquanto estiver pressionado:** *Libera*
- **Etapas:**
 - **Etapas 1:**
 - Atraso (quadros): 1
 - Duração (quadros): 4
 - Entrada 1: *Jogador 1 baixo*
 - **Etapas 2:**
 - Atraso (quadros): 1

- Duração (quadros): 4
 - Entrada 1: *Jogador 1 baixo*
 - Entrada 2: *Jogador 1 direita*
- Etapa 3:
 - Atraso (quadros): 1
 - Duração (quadros): 4
 - Entrada 1: *Jogador 1 direita*
- Etapa 4:
 - Atraso (quadros): 0
 - Duração (quadros): 1
 - Entrada 1: *Jogador 1 botão 2*

A macro abaixo serve para o segundo especial do mesmo personagem.

- Nome: *Stardust 2*
- Ativação: Tecla A
- Ao soltar: *Conclua a macro*
- Enquanto estiver pressionado: *Libera*
- Etapa 1:
 - Atraso (quadros): 1
 - Duração (quadros): 4
 - Entrada 1: *Jogador 1 cima*
- Etapa 2:
 - Atraso (quadros): 1
 - Duração (quadros): 4
 - Entrada 1: *Jogador 1 cima*
 - Entrada 2: *Jogador 1 direita*
- Etapa 3:
 - Atraso (quadros): 1
 - Duração (quadros): 4
 - Entrada 1: *Jogador 1 direita*
- Etapa 4:
 - Atraso (quadros): 0
 - Duração (quadros): 1
 - Entrada 1: *Jogador 1 botão 3*

6.3.9 Layout Plugin

Quando ativado, o plugin de layout permite a execução de scripts Lua embutido nos arquivos de layout. As ilustrações embutidas para alguns sistemas e alguns pacotes de ilustrações externas podem usar scripts Lua para fornecer recursos aprimorados de interação. Consulte [Usando scripts no layout do MAME](#) para mais informações.

6.3.10 Timecode Recorder Plugin

O plug-in *timecode recorder* faz a gravação do tempo em código num arquivo texto em conjunto com a gravação dos comandos do jogador que serve na assistência das pessoas que gravam as partidas dos jogos. Este registro de tempo é feito *apenas* durante a gravação destes comandos. O arquivo de registro possui o mesmo nome que o arquivo de gravação dos comandos mas com a extensão `.timecode`. Utilize as opções `-record` e `-input_directory` para registrar os comandos e para definir o destino destes arquivos.

O atalho padrão para a gravação do código de tempo é F12. É possível alterar esta configuração no menu de configuração do plug-in (escolha *Opções dos plug-ins* a partir do menu principal durante a emulação e escolha *Timecode Recorder*).

As configurações do plug-in ficam armazenadas no arquivo `plugin.cfg` no formato JSON dentro da pasta **timecode** na pasta raiz do MAME (consulte a opção [-homepath](#)).

6.3.11 Game Play Timer Plugin

O *game play timer plugin* cronometra o tempo total gasto numa combinação entre sistema e um item da lista de programas, assim como a quantidade de vezes que cada combinação foi iniciada. Para ver as estatísticas, abra o menu principal (pressione Tab durante a emulação), escolha *Opções dos plug-ins* e então escolha *Cronômetro*.

Este plugin registra o tempo real (o tempo decorrido enquanto a emulação estiver acontecendo conforme o sistema operacional) assim como o tempo emulado. O tempo do relógio do sistema pode ser menor do que tempo decorrido pelo tempo da emulação caso desligue o [-\[no\]throttle](#), use o recurso de "avanço rápido" do MAME ou pode ser maior do que o tempo emulado decorrido caso pause a emulação de se a emulação.

As estatísticas são armazenadas num arquivo `timer.db` na pasta **timer** (consulte a opção [homepath option](#)).

O arquivo é um banco de dados SQLite3.

CONFIGURAÇÕES AVANÇADAS

7.1 Diversos arquivos de configuração

- *A ordem da leitura dos arquivos*
- *Exemplos da sequência de leitura dos arquivos*
- *Usando diferentes arquivos de configuração*
- *Criando uma configuração específica para o Super Street Fighter 2*
- *Criando uma configuração específica para um sistema/driver*
- *Aplicando efeitos na tela*
- *Criando configurações específicas para consoles*

O MAME tem um poderoso sistema de configuração que permite ajustar as opções de cada jogo, sistema ou até mesmo um tipo de monitor em específico de maneira individual, porém requer um cuidado especial na organização das configurações, para mais detalhes de como o MAME encontra as suas configurações, consulte também o capítulo *Como o MAME localiza os seus arquivos?*.

7.1.1 A ordem da leitura dos arquivos

O MAME faz a interpretação dos arquivos de configuração na ordem abaixo:

1. A linha de comando é interpretada primeiro e tem prioridade sobre qualquer outro arquivo **.ini**.
2. O arquivo `mame.ini` (ou qualquer outra plataforma que use um `.ini` como o `mess.ini` por exemplo) são interpretados duas vezes. Na primeira passada é feito a leitura de diferentes configurações, a segunda é feita para ver se existe outro arquivo de configuração com um outro tipo de configuração válida (caso exista, altera as configurações com as informações deste segundo arquivo).
3. O arquivo `debug.ini` caso esteja em modo de depuração. Este é um arquivo de configuração avançada, a maioria das pessoas não precisam sequer se preocupar com ele.
4. Os arquivos INI de orientação da tela (seja o `horizont.ini` ou o `vertical.ini`). Use o arquivo `vertical.ini` para aplicar as configurações apenas num sistema com tela vertical como **Pac-Man** por exemplo. Já para um sistema com tela horizontal, use o arquivo `horizont.ini` para aplicar as configurações apenas nos sistemas como **Street Fighter Alpha**.

Geralmente o arquivo `horizont.ini` será lido nos sistemas sem monitor, nos monitores com diversas orientações diferentes ou monitores conectados em dispositivos tipo slot.

5. Os arquivos INI voltados para diferentes tipos de tela como `vector.ini` para jogos vetoriais, `raster.ini` para jogos rasterizados, `lcd.ini` para jogos ou consoles que usem telas de cristal líquido (LCD), EL, plasma, etc. Ambas os sistemas como a **Pac-Man** e a **Street Fighter Alpha** são rasterizadas¹², assim o arquivo `raster.ini` é lido para obter e aplicar as configurações específicas deste tipo de sistema, já o sistema **Tempest** é um sistema com tela vetorial, assim o arquivo `vector.ini` é lido e as configurações vetoriais são aplicadas apenas nos sistemas desta categoria.

Para sistemas que tenham mais de um monitor como a **House Mannequin** (faça o teste com `mame housemnq -numscreens 3`) que usa um monitor *CRT raster* e um par de telas de cristal líquido, os arquivos `raster.ini` e `lcd.ini` são lidos e as suas respectivas configurações aplicadas nas suas respectivas telas.

6. Os arquivos INI voltados para os arquivos de código-fonte (driver). O MAME tentará ler `source/<sourcefile>.ini` onde `<sourcefile>` é o nome do arquivo de código-fonte onde o sistema estiver definido. O código-fonte de um driver pode ser encontrado usando o comando `mame -listsource <nome_da_rom>`, exemplo:

```
mame.exe -listsource sfa
sfa                cps2.cpp
```

A Banpresto **Sailor Moon**, a Atlus **Dodonpachi** e a Nihon System **Dangun Feveron** por exemplo, todos rodam num hardware semelhante e estão listados no arquivo de código-fonte chamado `cave.cpp` que chamamos de **driver**, assim sendo, todos eles usarão o arquivo `source/cave.ini` para obter as suas configurações.

7. Os arquivos INI para BIOS (caso seja aplicável). O sistema **The Last Soldier** por exemplo, usa a BIOS do **Neo-Geo MVS**, então o arquivo `neogeo.ini` será lido. Nenhum arquivo INI será lido nos sistemas que não usem uma BIOS.
8. Arquivo INI da mesma família. O **The Last Soldier** é um clone do **The Last Blade / Bakumatsu Roman - Gekka no Kenshi**, assim o arquivo `lastblad.ini` será lido. Nenhum arquivo INI da mesma família será lido.
9. Arquivo INI do sistema. Usando o exemplo anterior, o arquivo `lastsold.ini` será lido para o **The Last Soldier**.

¹ São imagens compostas por pixels.

² <https://pt.wikipedia.org/wiki/Raster>

7.1.2 Exemplos da sequência de leitura dos arquivos

- O Brix que é um clone de Zzyzzyxx. (mame brix)
 1. Linha de comando
 2. mame.ini (global)
 3. Caso o depurador não esteja ativado, nenhum arquivo INI extra será lido.
 4. vertical.ini (orientação da tela)
 5. raster.ini (tipo do monitor)
 6. source/jack.ini (configuração específica para o código fonte do driver)
 7. Nenhuma BIOS definida.
 8. zzyzzyxx.ini (sistema da mesma família)
 9. brix.ini (sistema)
- Super Street Fighter 2 Turbo (mame ssf2t)
 1. Linha de comando
 2. mame.ini (global)
 3. Caso o depurador não esteja ativado, nenhum arquivo INI extra será lido.
 4. horizont.ini (orientação de tela)
 5. raster.ini (tipo do monitor)
 6. source/cps2.ini (configuração específica para o driver)
 7. Nenhuma BIOS definida.
 8. Nenhum sistema da mesma família.
 9. ssf2t.ini (sistema)
- Final Arch (mame finlarch)
 1. Command line
 2. mame.ini (global)
 3. Caso o depurador não esteja ativado, nenhum arquivo INI extra será lido.
 4. horizont.ini (orientação de tela)
 5. raster.ini (tipo do monitor)
 6. source/stv.ini (configuração específica para o driver)
 7. stvbios.ini (BIOS definida)
 8. smleague.ini (sistema da mesma família)
 9. finlarch.ini (sistema)

Lembre-se que os parâmetros na linha de comando tem preferência sobre todos os outros arquivos!

7.1.3 Usando diferentes arquivos de configuração

O MAME oferece a possibilidade de criar diferentes tipos de configuração separada por algumas categorias ou pelo nome dos drivers em vez de concentrar todas as configurações num único arquivo como o `mame.ini`. E para que isso?

O MAME possui dezenas de opções disponíveis para configurar outra dezena de coisas como áudio, vídeo, controladores diversos, etc. A linha de comando pode ficar bem grande e complexa dependendo do sistema a ser emulado e variar de sistema para sistema. Ao criar diferentes configurações é possível individualizar diferentes definições criar diferentes **perfis** para diferentes tipos de sistemas, drivers, dispositivos, etc.

Podemos citar como exemplo a opção de vídeo, por predefinição, no Windows o MAME escolhe `d3d` como a melhor opção, contudo, em sistemas Windows mais recentes como o Windows 10/11, a melhor opção seria `bgfx` em conjunto com um `bgfx_backend` apropriado como o `d3d12` ou melhor ainda, o `vulkan` caso a sua placa de vídeo seja compatível. É possível definir essa configuração como um padrão para todos os sistemas dentro do arquivo `mame.ini`.

O arquivo `mame.ini` afeta a configuração de forma global, ou seja, tudo o que for configurado nele **vale para tudo** e isso pode causar diversos problemas. Por exemplo, o sistema **Arkanoid** usa um controle com um disco giratório (chamado de *spinner controller*), no MAME é possível usar um joystick, um controle tipo *gamepad* e o mouse.

Porém ao definir `mouse 1` no `mame.ini` o sistema **Arkanoid** funcionará perfeitamente com o mouse, mas como o `mame.ini` serve como um arquivo de configuração **global**, a opção `mouse 1` faz com que todas as outros sistemas, ainda que não usem o mouse, passem a usá-lo. Então ao iniciar o sistema **Street Fighter II** por exemplo, o seu mouse é sequestrado pelo MAME e assim ficará enquanto o MAME estiver sendo executado ou até que você pressione a tecla P para pausar a emulação e reaver o controle do mouse pelo seu sistema operacional.

No seu desktop talvez isso não seja um problema, contudo, imagine a mesma situação num sistema arcade rodando o MAME aonde você não tenha um acesso fácil ao teclado, ficar "sem mouse" no seu sistema e por não saber deste detalhe ficar quebrando a cabeça sem entender o que está acontecendo perdendo horas alterando as configurações.

É para casos como este que as configurações individuais são importantes e é por isso que é preciso personalizar certas definições em alguns casos. Usando o **Arkanoid** como exemplo, para que apenas este sistema use a opção `mouse 1`, crie o arquivo `ini\arkanoid.ini` e nele coloque a opção desejada, exemplo:

```
mouse 1
```

Ao salvar o arquivo e ao iniciar o sistema, repare que é possível usar o mouse como controle. Além desta configuração ser aplicada no sistema **Arkanoid**, a configuração também será aplicada em todas os sistema onde as suas ROMs comecem com **arkanoid**, exemplo:

```
arkanoid      "Arkanoid (World, older)"
arkanoidj     "Arkanoid (Japan, newer)"
arkanoidja    "Arkanoid (Japan, newer w/level select)"
arkanoidjb    "Arkanoid (Japan, older)"
arkanoidjbl   "Arkanoid (bootleg with MCU, set 1)"
arkanoidjbl2  "Arkanoid (bootleg with MCU, set 2)"
arkanoidu     "Arkanoid (US, newer)"
arkanoiduo    "Arkanoid (US, older)"
```

Fazendo assim nós também evitamos um conflito de configurações cruzadas onde a configuração de um sistema afete o outro e vice versa. Assim podemos ter configurações específicas para sistemas que usem vetores sem que estas configurações afete sistemas rasterizados ou sistemas que sequer usem telas por exemplo.

7.1.4 Criando uma configuração específica para o Super Street Fighter 2

Aqui uma sugestão de configuração onde vamos definir diferentes parâmetros, porém, sem alterar nada no `mame.ini`, tenha certeza que todas as ROMs estejam na pasta **roms**. Para o nosso exemplo usaremos a ROM `ssf2.zip` e `qsound_hle.zip`.

Super Street Fighter 2

- Crie um arquivo texto chamado `ssf2.ini` dentro do diretório **ini** e cole as configurações abaixo:

video	bgfx
bgfx_backend	vulkan
snabbilinear	0
refreshspeed	1
filter	0
bgfx_screen_chains	crt-geom
ctrlr	sf2

Observe que as opções sugeridas aqui são as mais genéricas possíveis para que funcionem com a maioria dos computadores mais recentes cobrindo o mínimo necessário.

7.1.5 Criando uma configuração específica para um sistema/driver

Aqui em vez de criar várias configurações individuais, nós criaremos uma única configuração que será aplicada a todos os sistemas listados dentro do driver `capcom/cps2.cpp` (use o comando `-ls` para indentificar o driver do sistema).

CPS-2

- Crie o arquivo texto `cps2.ini` dentro do diretório **ini\source**, com as seguintes opções:

video	bgfx
bgfx_backend	vulkan
snabbilinear	0
refreshspeed	1
filter	0
bgfx_screen_chains	crt-geom
ctrlr	sf2

Agora todos os sistemas dentro de `capcom/cps2.cpp` (veja quais são eles com o comando `mame ssf2 -lb`), usarão as configurações acima. Caso utilize esta configuração, você pode apagar o arquivo `ssf2.ini`, a não ser que você use ou queira usar uma configuração muito específica para ele, neste caso, não é preciso clonar a configuração, basta adicionar a opção que deseja. Então o seu arquivo `ssf2.ini` teria apenas a seguinte configuração:

samplerate	44100
window	1

Neste caso, somado a configuração anterior, o sistema e todos os relacionados a ele, serão configurados a usar uma taxa de amostragem de **44.1 kHz** e rodar em modo janela, enquanto todos os outros rodarão com tela inteira e com uma taxa de amostragem padrão de **48 kHz**.

7.1.6 Aplicando efeitos na tela

Jogos como **Street Fighter** e tantos outros na época utilizavam **raster graphics**, onde a imagem na tela é formada por pixels, no Brasil estes gráficos são também conhecidos como mapa de bits ou bitmap. Diferente de hoje, as imagens eram desenhadas em linhas de escaneamento para formar uma imagem nas antigas tela de tubo de raios catódicos ou CRT. Tal efeito assim como seus defeitos, podem ser reproduzidas. Abordaremos aqui apenas a configuração básica porém o assunto já foi abordado nos capítulos sobre [BGFX](#), [GLSL](#) e [HLSL](#).

Para aplicar um simples efeito de scanlines nos sistemas com **raster graphics**, crie o arquivo `raster.ini` dentro do diretório `ini`:

```
prescale          3
effect            scanlines
```

Agora ao rodar o sistema **Street Fighter II: The World Warrior** (`mame sf2`) note que há algumas linhas na tela, outros efeitos podem ser baixados do [MameWorld](#). Apesar dos efeitos de sobreposição darem apenas uma aparência muito simplificada de uma tela CRT, a sua vantagem é consumir poucos recursos, porém há muito tempo que não se utiliza mais a opção `effect` já que nos dias de hoje há opções mais modernas. Para quem tem um hardware um pouco mais sofisticado, use a simulação as opções do BGFX, GLSL e HLSL.

Baixe os shaders GLSL do [mameau](#), extraia o diretório `osd` no diretório raiz do MAME e experimente esta configuração no seu arquivo `ini\raster.ini`:

raster.ini

```
filter            0
gl_glsl           1
gl_glsl_filter    1
glsl_shader_mame0  osd\shader\glsl_plain
glsl_shader_mame1  osd\CRT-geom
```

Rode novamente o sistema **Street Fighter II: The World Warrior** e repare que a tela já possui curvatura, linhas de escaneamento, distorções e outras características semelhantes a uma tela CRT, incluindo os seus defeitos.

Particularmente prefiro apenas manter as características das linhas de escaneamento do CRT sem os defeitos, as distorções, a saturação, nada. Gosto de usar o `pix` que é um filtro que evita distorções de pixels (integer scaling) quando você aumenta a resolução da tela junto com o efeito [ApertureMRES](#):

Minha Configuração

```
filter            1
gl_glsl           1
gl_glsl_filter    1
glsl_shader_mame0 glsl\pix\pixellate
effect            ApertureMRES
```

Repare que ao usar o arquivo `raster.ini` para armazenar as configurações dos efeitos da tela, ela também será aplicada em qualquer outro sistema que seja definido como "raster" como consoles de video-game, computadores pessoais, etc.

7.1.7 Criando configurações específicas para consoles

Da mesma maneira que estas configurações funcionam com arcades é possível também fazer o mesmo para consoles como o **Sega Genesis/Mega Drive**, **Super Nintendo** dentre outros, aqui alguns exemplos.

Sega Genesis / Mega Drive

- Crie um arquivo `mdconsole.ini` dentro do diretório **ini\source** com as seguintes opções:

<code>romapath</code>	<code>mame_rom_dir;caminho_completo_roms</code>
<code>samplerate</code>	32000
<code>ctrlr</code>	<code>megadrive</code>

Super Nintendo

- Crie um arquivo `snes.ini` dentro do diretório **ini\source** com as seguintes opções:

<code>romapath</code>	<code>mame_rom_dir;caminho_completo_roms</code>
<code>samplerate</code>	32000
<code>ctrlr</code>	<code>snes</code>

Neo Geo

- Crie o arquivo `neogeo.ini` dentro do diretório **ini\source** com as seguintes opções:

<code>romapath</code>	<code>mame_rom_dir;caminho_completo_roms</code>
<code>bios</code>	<code>unibios40</code>
<code>ctrlr</code>	<code>neogeo</code>
<code>filter</code>	1
<code>prescale</code>	2
<code>keepaspect</code>	1

7.2 Lidando com o caminho dos arquivos

O MAME obedece uma sequência lógica quando verifica os arquivos dos usuários como ROMs e os arquivos de trapaça.

7.2.1 A sequência de leitura

Usando o exemplo de um arquivo de trapaça do Sega Genesis/Megadrive para o jogo **After Burner 2** (**aburner2** na lista de software para Megadrive), o caminho predefinido para o *cheatpath* é **cheat**. A sequência de pesquisa que o MAME utiliza para encontrar o arquivo de trapaça é demonstrado abaixo:

1. **cheat/megadriv/aburner2.xml**
Primeiro caminho a ser pesquisado.
2. **cheat/megadriv.zip → aburner2.xml**
Repare que primeiro é pesquisado um arquivo **.zip** e depois um arquivo **.7z**.
3. **cheat/megadriv.zip → <qualquer caminho>/aburner2.xml**
Caso exista, será procurado pelo primeiro arquivo **aburner2.xml** que for encontrado dentro daquele arquivo zip, independente de onde esteja.
4. **cheat.zip → megadriv/aburner2.xml**
A próxima pesquisa será feita por uma combinação de arquivo ou diretório, porém agora, dentro do arquivo de trapaça **cheat.zip**.
5. **cheat.zip → <qualquer caminho>/megadriv/aburner2.xml**
Caso exista, irá procurar em qualquer lugar menos dentro do primeiro **aburner2.xml** que fica dentro do diretório **megadriv** e que esteja dentro de um arquivo zip.
6. **cheat/megadriv.7z → aburner2.xml**
A próxima pesquisa é feita agora dentro de arquivos **.7z**¹.
7. **cheat/megadriv.7z → <qualquer caminho>/aburner2.xml**
Em sequência a pesquisa é feita em qualquer outro caminho próximo.
8. **cheat.7z → megadriv/aburner2.xml**
Similar aos arquivos **.zip**, porém agora com arquivos **.7z**.
9. **cheat.7z → <qualquer caminho>/megadriv/aburner2.xml**
Similar aos itens 7 e 8.

[a fazer: A leitura do conjunto de arquivos ROM é um pouco mais complicado, adicionar CRC. Documentar isso no próximo dia ou dois.

¹ 7zip <https://www.7-zip.org/> (Nota do tradutor).

7.3 Desativando o interruptor de câmbio

Este é um recurso avançado para lidar com o interruptor de certos sistemas arcade antigos como a *Spy Junter* e *Outrun* que usava uma chave de via dupla que funcionava como um câmbio de marchas. Por predefinição esse câmbio é tratado como um interruptor. Um toque na configuração do controle para que o câmbio alterne entre marcha alta e baixa, com outro toque ele volta para a posição anterior. Essa pode não ser a melhor opção se você tiver em mãos um câmbio que trabalhe como os câmbios antigos usados nos sistemas originais. O câmbio estará engrenado quando a chave for ligada, funcionando ao contrário quando estiver desligada.

Observe que este recurso *não* ajudará o controle dos usuários e tão pouco será de qualquer ajuda para jogos que tenham um câmbio com mais de dois estados (como jogos modernos com mais de uma marcha por exemplo).

Este recurso não é exibido através da tela do usuário pois é uma customização extrema feita apenas para pessoas que têm essa necessidade em específico e o conhecimento para fazer o uso dela da forma correta.

7.3.1 Alternando o interruptor de câmbio

O jogo Spy Hunter (do conjunto *spyhunt*) será usado como exemplo para explicar as alterações necessárias. Dentro da pasta CFG há um arquivo *.CFG* que você precisará editar como *spyhunt.cfg* por exemplo.

No MAME, comece rodando o jogo que estamos usando de exemplo, o *spyhunt*, assim:

mame spyhunt.

Configure os controles da maneira que faria em outros jogos, incluindo a configuração do câmbio. Saia do MAME e abra o arquivo *.cfg* do jogo no editor de texto de sua preferência.

Dentro do arquivo *spyhunt.cfg*, você deverá encontrar as seguintes linhas de texto de configuração para a entrada. O código de entrada exibido no meio da configuração pode variar dependendo da posição do controle que você tiver configurado:

```
<port tag=":ssio:IP0" type="P1_BUTTON2" mask="16" defvalue="16">
  <newseq type="standard">
    JOYCODE_1_RYAXIS_NEG_SWITCH OR JOYCODE_1_RYAXIS_POS_SWITCH
  </newseq>
</port>
```

Você precisa editar a linha da porta que definirá a entrada. Para o jogo Spy Hunter será *P1_BUTTON2*. Adicione *toggle="no"* no final da tag, como mostra o exemplo abaixo:

```
<port tag=":ssio:IP0" type="P1_BUTTON2" mask="16" defvalue="16" toggle="no">
  <newseq type="standard">
    JOYCODE_1_RYAXIS_NEG_SWITCH OR JOYCODE_1_RYAXIS_POS_SWITCH
  </newseq>
</port>
```

Salve e saia. Para desativar, simplesmente remova a opção **toggle="no"** de cada arquivo de *.CFG* que desejar.

7.4 Efeitos BGFX para (quase) todo mundo

- *Introdução*
- *Resolução e a relação de aspecto da tela*
- *Começando com o BGFX*
- *Alterando as configurações*
- *Customizando as configurações BGFX HLSL dentro do MAME*
- *Usando os filtros para adicionar pilares nos cantos da tela do vídeo*

Nota: Segundo [este post](#) feito pelo **MameHaze**¹ os efeitos BGFX causam lentidão na emulação e poderão ser removidos do MAME no futuro.

7.4.1 Introdução

Por padrão, o MAME gera uma versão idealizada de como estaria o vídeo ao ser enviado ao monitor do gabinete de arcade, com uma mínima alteração na saída (principalmente para esticar a imagem do jogo de volta à proporção original do monitor, geralmente 4:3). Isso funciona bem, contudo, perde um pouco do fator nostalgia. Os monitores arcade nunca foram ideais, ainda que estivesse em perfeitas condições, a natureza de um monitor CRT distorce essa imagem de maneira que altera a aparência da imagem original de forma significativa.

Os monitores de LCD modernos simplesmente não têm a mesma aparência e mesmo os monitores CRT de computador, não se igualam a aparência visual de um monitor de arcade sem uma ajuda extra.

E é aí que o novo renderizador **BGFX** com **HLSL** entra em cena.

O HLSL simula a maioria dos efeitos (e defeitos) que um monitor de arcade CRT geraria ao sinal de vídeo, tornando o resultado bem próximo a um monitor CRT de arcade, dando um ar muito mais autêntico ao visual. No entanto, o HLSL exige um certo esforço por parte do usuário: as configurações que você usa serão adaptadas às especificações do sistema do seu PC, especialmente, ao monitor que você está utilizando. Além disso, haviam centenas de milhares de diferentes tipos de monitores nos fliperamas. Cada um foi ajustado e mantido de forma diferente, o que significa que não há apenas uma aparência correta. As diretrizes básicas serão fornecidas aqui para ajudá-lo, mas você também pode pedir ajuda nos fóruns mais conhecidos sobre o MAME espalhados pela internet.

¹ **Citação** "O uso do BGFX adiciona um atraso em torno de 2 ou 3 quadros e no geral é considerado inadequado para jogos, ainda que se obtenha uma melhor aparência visual (Eu acredito que (o BGFX) acabará sendo removido já que há muitos problemas segundo os próprios usuários)".

7.4.2 Resolução e a relação de aspecto da tela

A resolução é um assunto muito importante para as configurações do hlsl. Você desejará que o MAME esteja usando a resolução nativa do seu monitor para evitar distorções e atrasos adicionais criados pelo seu monitor ao tentar preencher a imagem na tela.

Enquanto a maioria dos sistemas de arcade usava um monitor com proporção de tela no formato 4:3 (ou 3:4 se o monitor estivesse orientado verticalmente como é no caso do **Pac Man**), a essa altura do campeonato é difícil encontrar nos dias de hoje um monitor ou TV que tenha uma proporção de tela no formato 4:3.

A boa notícia é que esse espaço extra que sobra nas laterais não é desperdiçado. Muitos gabinetes de arcade na época utilizavam uma moldura com ilustrações ao redor da tela, caso você tenha esses arquivos o MAME também irá exibir essas ilustrações na tela. Para se obter um melhor resultado, ative o visualizador de ilustrações e selecione *Aproxime para a região da tela*, consulte [Opções do vídeo](#) para obter mais informações.

Alguns monitores de LCD mais antigos usavam uma resolução nativa de 1280x1024 onde tinham uma proporção de tela no formato 5:4. Neste exemplo, não há muito espaço extra suficiente para exibir a ilustração e você vai notar um leve esticamento vertical, porém os resultados ainda serão bons o suficiente, como se fossem um monitor com formato 4:3.

7.4.3 Começando com o BGFX

Antes de começar, você precisará seguir as instruções de configuração inicial do MAME encontrada em outra parte deste manual. As distribuições oficiais do MAME à partir da versão 0.172 já incluem o bgfx, então você não precisa baixar nenhum outro arquivo adicional.

Abra o seu `mame.ini` no seu editor de texto preferido como o bloco de notas por exemplo e verifique se as seguintes opções estão definidas corretamente:

- `video bgfx`

Agora tire um momento para ler as definições de configuração na seção abaixo para aprender como melhor configurar as opções do bgfx.

Como descrito em [Diversos arquivos de configuração](#), o MAME segue uma sequência na hora de processar os arquivos `.ini`. As configurações bgfx podem ser editadas diretamente no arquivo `mame.ini`, porém para tirar melhor proveito do poder dos arquivos de configuração do MAME, talvez seja melhor copiar as opções do **bgfx** do `mame.ini` para um outro arquivo de configuração e fazer as modificações lá.

Particularmente, você vai querer que as configurações `bgfx_screen_chains` sejam específicas e personalizáveis para cada sistema em vez de uma única configuração para todos.

Salve o arquivo `.ini` e já estamos prontos para começar.

7.4.4 Alterando as configurações

bgfx_path

Seus arquivos de sombreamento **bgfx** (*bgfx shader*) são armazenados aqui. Por definição, o nome desta pasta é **bgfx**, fica na pasta raiz do MAME.

bgfx_backend

Seleciona um tipo de infraestrutura de renderização para que o bgfx possa usar, o MAME faz a escolha da melhor opção compatível e disponível no seu sistema, caso queira defini-las manualmente, estas são as opções disponíveis:

- d3d9 Renderizador do Direct3D 9.0 (Requer o Windows XP ou uma versão mais nova do Windows).
- d3d11 Renderizador do Direct3D 11.0 (Requer o Windows Vista com o D3D11 atualizado, o Windows 7 ou uma versão mais nova do Windows).
- d3d12 Renderizador do Direct3D 12.0 (Requer o Windows 10 ou uma versão mais nova do Windows, assim como uma placa de vídeo com um driver compatível).
- opengl Renderizador OpenGL (Requer Drivers compatíveis com OpenGL, pode não funcionar bem com algumas placas de vídeo mais antigas ou mal projetadas). Compatível com Linux/macOS.
- metal Metal Apple Graphics API (Requer macOS 10.14 Mavericks ou mais recente).
- vulkan Renderizador Vulkan (via hardware), compatível atualmente com OpenGL ES 3.1, OpenGL 4.X ou versões mais recentes. Requer drivers compatíveis para as placas de vídeo mais atuais e pode não ser compatível com as placas de vídeo ou com drivers mais antigos.

O valor predefinido é auto.

bgfx_debug

Ativa as funcionalidades de depuração, voltado apenas para os desenvolvedores.

bgfx_screen_chains

Determina como manipular a renderização **BGFX** tela a tela. As opções disponíveis são:

- default Gera uma tela com filtro bilinear (padrão).
- unfiltered Gera uma tela sem filtro.
- hlsl Simula uma tela CRT usando shaders hlsl.
- lut Ativa o uso de LUTs (tabela de consulta de cores)
- crt-geom Simula uma tela CRT simples e leve.
- crt-geom-deluxe Simula uma tela CRT simples, porém, mais detalhada.
- lcd-grid Simula uma tela de display líquido.

Nós fazemos uma distinção entre os dispositivos de tela emuladas (que chamamos de **screen** ou **tela**) e a tela física (que chamaremos de **window** ou **janela**, configurável através da opção `-numscreens`). Nós usamos dois pontos : para separar as janelas e as vírgulas , para separar as telas definindo valores na opção `-bgfx_screen_chains`.

Numa combinação de só uma janela, no caso de jogos com uma única tela, como o **Pac Man** num monitor físico de PC, você pode definir a opção como:

```
bgfx_screen_chains hlsl
```

As coisas se complicam um pouco mais quando temos diversas telas e janelas.

Para usar uma só janela, num jogo com mais de uma tela, como é o caso do jogo **Darius** usando só um monitor físico de PC, defina as opções para cada uma dessas telas individualmente, como mostra o exemplo abaixo:

```
bgfx_screen_chains hlsl,hlsl,hlsl
```

Isso também funciona com jogos que usam uma única tela caso você queira espelhar a saída dela para vários outros monitores físicos. Por exemplo, você pode configurar o jogo **Pac Man** para ter uma saída não filtrada para ser usada numa transmissão de vídeo enquanto a saída para segunda tela é configurada para exibir uma tela com os efeitos hlsl.

Num jogo com múltiplas telas em várias janelas como o jogo **Darius** usando três monitores físicos, defina as opções como mostra abaixo de forma individual para cada janela:

```
bgfx_screen_chains hlsl:hlsl:hlsl
```

Outro exemplo seria o jogo **Taisen Hot Gimmick** que usa dois monitores CRT para cada jogador mostrando a mão de cada um. Caso esteja usando duas janelas com duas telas físicas, faça como o exemplo abaixo:

```
bgfx_screen_chains hlsl:hlsl
```

Outro caso especial, a Nichibutsu tinha uma sistema tipo coquetel de Mahjongg que usa uma tela CRT bem no meio do sistema, junto com outras duas telas de LCD individuais mostrando a mão para cada jogador. Nós gostaríamos que os LCDs não fossem tão filtrados como eram, enquanto o CRT seria melhorado através do uso do hlsl.

Como queremos dar a cada jogador sua própria tela cheia (dois monitores físicos) junto com o LCD, nós fazemos assim:

```
-numscreens 2 -view0 "Player 1" -view1 "Player 2" -video bgfx -bgfx_screen_chains_  
↪hlsl,unfiltered:hlsl,unfiltered
```

Isso configura a visualização de cada tela respectivamente, mantendo o efeito de tela CRT com HSL para cada janela física enquanto fica sem os filtros nas telas LCD.

Caso esteja usando apenas uma janela ou tela, tendo em mente que o jogo ainda tem três telas, nós faríamos assim:

```
bgfx_screen_chains hlsl,unfiltered,unfiltered
```

Observe que as vírgulas estão nas bordas externas e qualquer dois-pontos estão no meio.²

² Onde? (Nota do tradutor)

bgfx_shadow_mask

Especifica o arquivo PNG para ser usado como efeito da máscara. Por padrão é usado o arquivo `slot-mask.png`.

bgfx_lut

Use um arquivo LUT para aplicar diferentes efeitos de textura na tela.

O Valor predefinido é nenhum

bgfx_avi_name

Essa opção permite que você possa definir um nome de arquivo AVI para gravar o vídeo do sistema emulado com os efeitos `bgfx_avi_name` `pacman.avi` por exemplo.

O Valor predefinido é auto

7.4.5 Customizando as configurações BGFX HLSL dentro do MAME

Comece rodando o MAME com o jogo de sua preferência (**mame pacman** por exemplo).

Use a tecla til (~)³ para chamar a tela de opções que vai aparecer na parte de baixo da tela. Use as teclas cima e baixo para navegar dentre as várias opções, enquanto as teclas esquerda e direita irão permitir que você altere o valor dessas opções. Os resultados aparecerão em tempo real conforme elas forem sendo alteradas.

Observe que as configurações são individuais para cada tela.

As configurações dos controles deslizantes BGFX são registradas em arquivos .cfg individualmente por sistema. Caso a configuração bgfx_screen_chains tenha sido definida (num arquivo .ini ou através da linha de comando), será através deles que os efeitos iniciais serão definidos. Já se nenhuma configuração bgfx_screen_chains for definida, o MAME usará os efeitos que você escolheu na última vez que rodou o sistema.

7.4.6 Usando os filtros para adicionar pilares nos cantos da tela do vídeo

O MAME inclui exemplos de shaders BGFX e layouts para preencher o espaço não utilizado numa tela widescreen 16:9 com uma versão emulada da borda do vídeo. Todos os arquivos necessários estão inclusos, bastam ser ativados.

Para sistemas que usam monitores horizontais 4:3, experimente estas opções:

```
-override_artwork bgfx/border_blur -view Horizontal -bgfx_screen_chains crt-geom,pillarbox_
↳left_horizontal,pillarbox_right_horizontal
```

Para sistemas que usam monitores verticais 3:4, experimente estas opções:

```
-override_artwork bgfx/border_blur -view Vertical -bgfx_screen_chains crt-geom,pillarbox_
↳left_vertical,pillarbox_right_vertical
```

- Você pode usar uma configuração diferente no lugar do crt-geom para que o efeito seja aplicado à imagem da tela principal no centro (default, hlsl ou lcd-grid por exemplo).
- Caso tenha alterado a visualização do sistema no MAME anteriormente, a visualização correta dos pilares, por padrão, não será selecionada. Use o menu de opções do vídeo para selecionar a visualização correta.
- Você pode adicionar essas configurações a um arquivo INI para que elas sejam aplicados automaticamente em determinados sistemas (horizont.ini ou vertical.ini ou o arquivo INI de um determinado sistema por exemplo).

³ Até que o teclado **ABNT-2** seja mapeado pela equipe do MAMEDev, essa tecla fica do lado esquerdo da tecla 1, logo abaixo da tecla ESQ. (Nota do tradutor)

7.5 Os efeitos HLSL para Windows

Por predefinição, o MAME gera um sinal de vídeo puro, assim como seria também no hardware original do arcade até o sinal chegar aos circuitos que levam ao monitor CRT do arcade, com pequenas modificações na saída (em geral, esticar a imagem do jogo de volta à proporção que se teria num monitor CRT, geralmente na proporção 4:3), no geral isso funciona bem, mas perde-se um pouco do fator nostalgia. Os monitores de arcade, ainda que em perfeitas condições, nunca foram ideais pois devido a sua natureza o monitor CRT distorciam a imagem original de maneira a distorcerem significativamente a sua aparência final na tela.

Os monitores CRT dos arcades são uma experiência única na maneira que a imagem é formada e apresentada na tela, imagem essa que os monitores de LCD e até mesmo monitores CRT não possuem.

É aí então que o hlsl entra em cena.

O filtro hlsl simula a maioria dos efeitos de vídeo que um monitor CRT de arcade teria, fazendo com que o resultado visual seja muito mais realista. Porém, os filtros hlsl exigem um esforço extra dos recursos do seu computador e em especial do monitor que você estiver usando. Além disso, havia centenas de milhares de tipos monitores diferentes nos fliperamas. Cada um foi ajustado e mantido de forma diferente, o que significa que, não tem como escolher e definir entre todos eles, apenas um como referência. Diretrizes básicas serão fornecidas aqui para ajudá-lo, mas você também poderá pedir mais opiniões em qualquer um dos fóruns conhecidos sobre o MAME espalhados pela internet.

7.5.1 Resolução e relação de aspecto da tela

A resolução é um assunto muito importante para as configurações do hlsl. Você desejará que o MAME esteja usando a resolução nativa do seu monitor para evitar distorções e atrasos adicionais criados pelo seu monitor ao tentar preencher a imagem na tela.

Enquanto a maioria dos sistemas de arcade usava um monitor com proporção de tela no formato 4:3 (ou 3:4 se o monitor estivesse orientado verticalmente como é no caso do **Pac Man**), a essa altura do campeonato é difícil encontrar nos dias de hoje um monitor ou TV que tenha uma proporção de tela no formato 4:3.

A boa notícia é que esse espaço extra que sobra nas laterais não é desperdiçado. Muitos gabinetes de arcade na época utilizavam uma moldura com ilustrações ao redor da tela, caso você tenha esses arquivos o MAME também irá exibir essas ilustrações na tela. Para se obter um melhor resultado, ative o visualizador de ilustrações e selecione o modo recortado¹.

Alguns monitores de LCD mais antigos usavam uma resolução nativa de 1280x1024 onde tinham uma proporção de tela no formato 5:4. Neste exemplo, não há muito espaço extra suficiente para exibir a ilustração e você vai notar um leve esticamento vertical, porém os resultados ainda serão bons o suficiente, como se fossem um monitor com formato 4:3.

¹ Cropped do Inglês. (Nota do tradutor)

7.5.2 Introdução ao HLSL

Antes de começar, você precisará seguir as instruções de configuração inicial do MAME encontrada em outra parte deste manual. As distribuições oficiais do MAME já incluem o hlsl, então você não precisa baixar nenhum outro arquivo adicional.

Abra o seu `mame.ini` no seu editor de texto preferido como o bloco de notas por exemplo e verifique se as seguintes opções estão definidas corretamente:

- **video d3d**
- **filter 0**

O primeiro é necessário porque hlsl requer suporte do Direct3D. O último desliga filtro extras que possam interferir com a saída hlsl.

Por último, uma edição a mais para ativar o hlsl:

- **hlsl_enable 1**

Salve o arquivo `.ini` e você está pronto para começar.

Nota: Caso tenha problemas com o d3d (o efeito não funciona por exemplo), consulte o capítulo [Usando o HLSL no Windows 8/10/11 sem precisar instalar o DirectX 9](#).

Várias predefinições foram incluídas na pasta `ini` junto com o MAME, permitindo um bom ponto de partida para as configurações iniciais de tela para os consoles **Nintendo Game Boy**, **Nintendo Game Boy Advance**, **Rasterizado** e **Vetorizado**.

7.5.3 Customizando as configurações HLSL dentro MAME

Por vários motivos complicados de explicar, as configurações hlsl não são mais salvas quando você sai do MAME. Isso significa que apesar das configurações exigirem um pouco mais de trabalho de sua parte, os resultados sempre sairão conforme esperado.

Comece rodando o MAME com o jogo de sua preferência como por exemplo **mame pacman**.

Use a tecla til (~)² para chamar a tela de opções que vai aparecer na parte de baixo da tela. Use as teclas cima e baixo para navegar dentre as várias opções, enquanto as teclas esquerda e direita irão permitir que você altere o valor dessas opções. Os resultados aparecerão em tempo real conforme elas forem sendo alteradas.

Depois de encontrar as configurações desejadas, anote os números num bloco de notas e saia do MAME.

7.5.4 Alterando as configurações

Como descrito em [Diversos arquivos de configuração](#), o MAME segue uma sequência na hora de processar os arquivos `.ini`. As configurações hlsl podem ser editadas diretamente no arquivo `mame.ini`, porém para tirar melhor proveito do poder dos arquivos de configuração do MAME, talvez seja melhor copiar as opções do hlsl do `mame.ini` para um outro arquivo de configuração e fazer as modificações lá.

Por exemplo, uma vez que você encontrou configurações de hlsl que acha que são apropriadas para os jogos de Neo-Geo, você pode colocar essas configurações num arquivo `neogeo.ini` para que todos os jogos de Neo-Geo usem essas configurações sem que você tenha que adicioná-las manualmente uma a uma em diferentes arquivos `.ini` como o nome do jogo.

² Até que o teclado **ABNT-2** seja mapeado pela equipe do MAMEDev, essa tecla fica do lado esquerdo da tecla 1, logo abaixo da tecla ESQ. (Nota do tradutor)

7.5.5 Alterando as configurações

hlslpath

Seus arquivos de sombreado hlsl são armazenados aqui. Por definição o nome desta pasta é **hlsl** fica na pasta raiz do MAME.

hlsl_snap_width

Define a **largura** que as capturas de tela hlsl terão (Alt+F12).

hlsl_snap_height

Define a **altura** que as capturas de tela hlsl terão (Alt+F12).

shadow_mask_alpha

Define a intensidade que o efeito de sombra da máscara terá. O intervalo aceitável vai de **0** a **1**, onde **0** não exibe nenhum efeito de sombra da máscara, **1** a máscara será completamente opaca e **0.5** será **50%** transparente.

shadow_mask_tile_mode *(Máscara de Sombra em Modo Ladrilhado)*

Define se a máscara de sombra deve ser lado a lado com base na resolução de tela do seu monitor ou com base na resolução de origem do sistema emulado. Os valores válidos são entre **0** para modo de tela *Screen* e **1** para modo de origem *Source*.

shadow_mask_texture
shadow_mask_x_count (Quantidade X de Pixels Máscara de Sombra)
shadow_mask_y_count (Quantidade Y de Pixels Máscara de Sombra)
shadow_mask_usize (Tamanho U da Máscara de Sombra)
shadow_mask_vsize (Tamanho V da Máscara de Sombra)
shadow_mask_x_count (Deslocamento U da Máscara de Sombra)
shadow_mask_y_count (Deslocamento V da Máscara de Sombra)

Essas configurações devem estar em harmonia entre si. As regras **shadow_mask_texture** em particular, definem as regras de como você deve configurar as outras opções.

shadow_mask_texture

Configura a textura do efeito de máscara de sombra. O MAME vem com três máscaras de sombra:

- aperture-grille.png
- shadow-mask.png
- slot-mask.png

shadow_mask_usize e shadow_mask_vsize

definem o tamanho a ser usado pela textura do efeito **shadow_mask_texture** em valores de porcentagem, começando pelo canto superior esquerdo.

Isso significa que, para uma textura com o tamanho real com pixels de 24x24 e um tamanho de u/v com **0.5,0.5**, serão usados **12x12** pixels no canto superior esquerdo.

Lembre-se de definir um tamanho de u/v que possibilite organizar a textura lado a lado sem lacunas ou falhas. **0.5,0.5** é bom para qualquer uma das textura de máscara de sombra que estão inclusas no MAME.

shadow_mask_x_count e **shadow_mask_y_count**

Definem quantos pixels devem ser usados na tela para exibir o tamanho u/v da textura. Caso use o exemplo acima e configurar a quantidade textura x/y numa proporção de **12,12** pixels, ela será exibida com uma proporção **1:1** na tela.

Caso defina a quantidade da textura x/y em proporção **24,24** ela será exibida duas vezes maior.

Exemplos de configuração para shadow_mask.png:

```
shadow_mask_texture shadow-mask.png
shadow_mask_x_count 12
shadow_mask_y_count 6 ou 12
shadow_mask_usize 0.5
shadow_mask_vsize 0.5
```

Exemplos de configuração para slot-mask.png:

```
shadow_mask_texture slot-mask.png
shadow_mask_x_count 12
shadow_mask_y_count 8 ou 16
shadow_mask_usize 0.5
shadow_mask_vsize 0.5
```

Exemplos de configuração para aperture-grille:

```
shadow_mask_texture aperture-grille.png
shadow_mask_x_count 12
shadow_mask_y_count 12 ou outro qualquer
shadow_mask_usize 0.5
shadow_mask_vsize 0.5
```

shadow_mask_uoffset e shadow_mask_voffset

Podem ser usados para customizar o alcance do alinhamento final da máscara de sombreamento a nível de subpixel. O intervalo aceitável vai de **-1.00** até **1.00**, onde **0.5** move a máscara de sombreamento em **50%** com relação ao tamanho u/v da textura.

distortion

Define a intensidade da distorção quadrática da imagem na tela.

cubic_distortion

Define a intensidade da distorção cúbica da imagem na tela.

Os fatores de distorção em ambos podem ser negativos para que um seja compensado pelo outro, por exemplo, *distortion* **0.5** e *cubic_distortion* **-0.5**.

distort_corner

Define a intensidade de distorção dos cantos da tela, o que não afeta a distorção da imagem na tela em si.

round_corner

Define a intensidade de arredondamento dos cantos da tela.

smooth_border

Define a intensidade de suavização e desfoque das bordas da tela.

reflection (*Intensidade de Reflexo*)

Se configurado com um valor acima de **0**, cria um efeito de um reflexo em formato de mancha esbranquiçada na tela. É predefinido que a mancha seja colocada no canto superior direito da tela.

Editando o arquivo `POST.FX` na seção **GetSpotAddend**, você poderá alterar essa posição.

Os valores entre **0.00** até **1.00** ajustam a intensidade do efeito.

vignetting

Se configurado com um valor acima de **0**, incrementa o efeito vinheta nos cantos da tela com um pseudo efeito 3D.

Os valores entre **0.00** até **1.00** ajustam a intensidade do efeito.

scanline_alpha

Determina a intensidade do efeito de linhas de escaneamento dos monitores CRT na tela. O intervalo aceitável fica entre **0** e **1**, onde **0** não exibe nenhum efeito, **1** seria uma linha de escaneamento totalmente preta e **0.5** exibe 50% de transparência.

Observe que na tela dos monitores arcade as linhas de escaneamento não são completamente pretas.

scanline_size

Define o espaçamento total das linhas de escaneamento da tela. Se configurado como **1**, mostra uma consistente alternância de espaço entre as linhas da tela e as linhas de escaneamento.

scanline_height

Define o tamanho total de cada linha individual de escaneamento. Se configurando com um valor menor que **1**, faz com que as linhas fiquem mais finas, maiores que **1** as deixam mais grossas.

scanline_variation

Define a variação do tamanho de cada linha de escaneamento, dependendo do seu brilho. As linhas de escaneamento mais claras ficarão mais finas em comparação com as mais escuras.

Os valores ficam entre **0** e **2.0**, onde o valor predefinido é **1.0**. Se definido como **0.0**, todas as linhas de escaneamento ficam com o mesmo tamanho, independente do seu brilho.

scanline_bright_scale

Define a escala de brilho que a linha de escaneamento terá.

Valores maiores que **1** faz com que elas fiquem mais clara, valores menores as deixam mais escuras. Se definido como **0**, faz desaparecer todas as linhas de escaneamento.

scanline_bright_offset

Define o deslocamento do brilho/saturação das linhas de escaneamento, suavizando e deixando mais lisa a parte de cima e de baixo de cada linha de escaneamento.

scanline_jitter

Define a intensidade de oscilação ou tremulação das linhas de escaneamento na tela do monitor.

Alerta: Valores muitos altos podem irritar seus olhos.

hum_bar_alpha

Define a intensidade do efeito de interferência vertical.

defocus

Define a intensidade de desfoque na tela borrando os pixels individualmente como as bordas de um monitor velho. Especifique com valores *X,Y* (**defocus 1,1** por exemplo).

Os valores abaixo ajustam a convergência dos canais vermelho, verde e azul para uma determinada direção simulando um monitor velho, muitos monitores mal cuidados tem uma péssima convergência causando um efeito fantasma devido ao vazamento de cores que ficam fora do eixo do sprite.

- **converge_x** (*Convergência Linear X, RGB*)
- **converge_y** (*Convergência Linear Y, RGB*)
- **radial_converge_x** (*Convergência Radial X, RGB*)
- **radial_converge_y** (*Convergência Radial Y, RGB*)

Os valores abaixo definem a matriz 3x3 que será multiplicado junto com os sinais RGB para simular a proporção de interferência em cada canal de cor.

Por exemplo, o sinal verde com (**0.100, 1.000, 0.250**) é **10%** mais fraco que o sinal vermelho e **25%** mais forte no sinal azul.

- **red_ratio** (*Proporção de sinal RGB Vermelho*)
- **grn_ratio** (*Proporção de sinal RGB Verde*)
- **blu_ratio** (*Proporção de sinal RGB Azul*)

offset

Fortalece ou enfraquece a intensidade do deslocamento do sinal em uma determinada cor. Por exemplo, o sinal vermelho com um valor **0.5** com um desvio/deslocamento de **0.2** será intensificado para **0.7**.

scale

Aplica uma escala ao valor da cor do sinal atual. Por exemplo, o sinal vermelho com um valor de **0.5** com uma escala **1.1**, resultará num sinal de vermelho com **0.55**

power

Define um valor expoente da cor do sinal atual, também conhecido como gama. O gama é o valor relativo entre o claro e o escuro de uma imagem. Por exemplo, o sinal vermelho com um valor de **0.5** e com **power** valor **2** no vermelho, resulta um sinal de vermelho com **0.25**.

Em jogos com vetores, essa configuração também pode ser usada para ajudar a espessura dessas linhas.

floor

Define o valor do piso do sinal RGB, é o valor mínimo absoluto para um sinal de cor. Por exemplo, o sinal vermelho com um valor de **0.0** (ausência total do sinal vermelho) com o sinal vermelho com piso de **0.2**, resulta num sinal vermelho com valor **0.2**.

Normalmente usado em conjunto com a ilustração ativada para fazer a tela ter um brilho da trama mais fraca.

phosphor_life

Define o tempo de vida útil do fósforo das telas CRT, dando um efeito de envelhecimento na cor do sinal e de fantasma na tela.

O valor **0** não produz nenhum efeito fantasma, enquanto o valor **1** deixa um rastro para trás que só volta a ser alterado por sinal de cor de maior valor.

Isso também afeta bastante os jogos vetoriais.

saturation

Define a intensidade de saturação de cor.

bloom_blend_mode

Define a intensidade da mistura do efeito lume³. Os valores ficam entre **0** para um efeito mais *Claro* e **1** para um tipo mais *Escuro*, essa última só é útil com monitores do tipo STN LCD.

bloom_scale

Determina a escala da intensidade do efeito lume. Os monitores CRT dos arcades tem uma tendência a ter esse efeito naturalmente, onde as cores mais claras se misturam com os pixels que ficam ao redor. Este efeito utiliza mais recursos da sua placa de vídeo, deixe em **0** para desabilitar e economizar recursos de processamento da sua GPU.

bloom_overdrive

Determina o nível de saturação do branco do efeito lume, os valores RGB são separados por vírgula. Muito útil em jogos com tramas coloridas, LCD colorido ou jogos vetorizados coloridos.

- **bloom_lvl0_weight** (*Escala do Nível do Bloom 0*)
- **bloom_lvl1_weight** (*Escala do Nível do Bloom 1*)
-
- **bloom_lvl7_weight** (*Escala do Nível do Bloom 7*)
- **bloom_lvl8_weight** (*Escala do Nível do Bloom 8*)

Define o nível de intensidade do efeito lume. Os valores ficam entre **0.00** até **1.00**. Se for usado da maneira correta em conjunto com o **phosphor_life** o efeito de brilho/fantasma enquanto os objetos se movem na tela será aprimorado.

hlsl_write

Defina como **1** para habilitar a gravação dos efeitos hlsl junto com a *gravação do vídeo*.

O valor predefinido é desligado ou **0**.

Estes são as predefinições sugeridos para os jogos rasterizados:

bloom_lvl0_weight 1.00	Peso 0 do Nível Bloom	Tamanho Máximo.
bloom_lvl1_weight 0.64	Peso 1 do Nível Bloom	1/4 menor que o nível 0
bloom_lvl2_weight 0.32	Peso 2 do Nível Bloom	1/4 menor que o nível 1
bloom_lvl3_weight 0.16	Peso 3 do Nível Bloom	1/4 menor que o nível 2
bloom_lvl4_weight 0.08	Peso 4 do Nível Bloom	1/4 menor que o nível 3
bloom_lvl5_weight 0.06	Peso 1 do Nível Bloom	1/4 menor que o nível 4
bloom_lvl6_weight 0.04	Peso 1 do Nível Bloom	1/4 menor que o nível 5
bloom_lvl7_weight 0.02	Peso 1 do Nível Bloom	1/4 menor que o nível 6
bloom_lvl8_weight 0.01	Peso 1 do Nível Bloom	1/4 menor que o nível 7

³ Lume significa clarão de luz, luz forte, o efeito é muito semelhante a uma névoa ou neblina.

7.5.6 Jogos vetorizados

Os efeitos HLSL também podem ser usados com jogos vetorizados. Devido a uma grande variedade de opções para a configuração individual de jogos vetoriais, é altamente recomendável que você os adicione em arquivos INI individuais jogo a jogo (*tempest.ini* por exemplo).

As máscaras de sombreamento só estão disponíveis em jogos vetoriais e não devem ser usados em jogos vetoriais monocromáticos. Além disso, os jogos de vetoriais não usavam linhas de varredura, de modo que também devem ser desativados.

Abra o seu arquivo *.ini* no seu editor de texto preferido (o Bloco de notas por exemplo) e verifique se as seguintes opções estão configuradas corretamente:

- **video d3d**
- **filter 0**
- **hsl_enable 1**

Nas Opções Principais de Vetores:

- **beam_width_min 1.0** (*Feixe Com o Máximo de*)
- **beam_width_max 1.0** (*Feixe Com o Mínimo de*)
- **beam_intensity_weight 0.0** (*Altura da Intensidade do Feixe*)
- **flicker 0.0** (*Vector Flicker*)

Na Seção das Opções de Pós Processamento de Vetores:

- **vector_beam_smooth 0.0** (*Intensidade de Suavização do Feixe do Vetor*)
- **vector_length_scale 0.5** (*Atenuação Máxima do Vetor*)
- **vector_length_ratio 0.5** (*Extensão Mínima de Atenuação do Vetor*)

Valores sugeridos para jogos vetoriais:

- **bloom_scale** o valor deve ser maior em jogos vetoriais do que os jogos rasterizados. Para obter um melhor efeito, tente valores entre 0.4 e 1.0.
- **bloom_overdrive** só deve ser usado em com jogos vetoriais coloridos.
- **bloom_lvl_weights** deve ser configurado como mostrado abaixo:

bloom_lvl0_weight 1.00	Peso 0 do Nível Bloom	Tamanho Máximo.
bloom_lvl1_weight 0.48	Peso 1 do Nível Bloom	1/4 menor que o nível 0
bloom_lvl2_weight 0.32	Peso 2 do Nível Bloom	1/4 menor que o nível 1
bloom_lvl3_weight 0.24	Peso 3 do Nível Bloom	1/4 menor que o nível 2
bloom_lvl4_weight 0.16	Peso 4 do Nível Bloom	1/4 menor que o nível 3
bloom_lvl5_weight 0.24	Peso 1 do Nível Bloom	1/4 menor que o nível 4
bloom_lvl6_weight 0.32	Peso 1 do Nível Bloom	1/4 menor que o nível 5
bloom_lvl7_weight 0.48	Peso 1 do Nível Bloom	1/4 menor que o nível 6
bloom_lvl8_weight 0.64	Peso 1 do Nível Bloom	1/4 menor que o nível 7

7.6 Efeitos GLSL para *nix, OS X e Windows

Por predefinição, o MAME gera um sinal de vídeo puro, assim como seria também no hardware original do arcade até o sinal chegar aos circuitos que levam o sinal ao monitor CRT do arcade, com pequenas modificações na saída (em geral, esticar a imagem do jogo de volta à proporção que se teria num monitor CRT, geralmente na proporção 4:3), no geral isso funciona bem, mas perde-se um pouco do fator nostalgia. Os monitores de arcade, ainda que em perfeitas condições, nunca foram ideais pois devido a sua natureza o monitor CRT distorciam a imagem original de maneira a distorcerem significativamente a sua aparência final na tela.

Os monitores CRT dos arcades são uma experiência única na maneira que a imagem é formada e apresentada na tela, imagem essa que os monitores de LCD e até mesmo monitores CRT não possuem.

É aí então que o glsl entra em cena.

O filtro glsl simula a maioria dos efeitos de vídeo que um monitor CRT de arcade teria, fazendo com que o resultado visual seja muito mais realista. Porém, os filtros glsl exigem um esforço extra dos recursos do seu computador e em especial do monitor que você estiver usando. Além disso, havia centenas de milhares de tipos monitores diferentes nos fliperamas. Cada um foi ajustado e mantido de forma diferente, o que significa que, não tem como escolher e definir entre todos eles, apenas um como referência. Diretrizes básicas serão fornecidas aqui para ajudá-lo, mas você também poderá pedir mais opiniões em qualquer um dos fóruns conhecidos sobre o MAME espalhados pela internet.

7.6.1 Resolução e relação de aspecto da tela

A resolução é um assunto muito importante para as configurações do glsl. Você desejará que o MAME esteja usando a resolução nativa do seu monitor para evitar distorções e atrasos adicionais criados pelo seu monitor ao tentar preencher a imagem na tela.

Enquanto a maioria dos sistemas de arcade usava um monitor com proporção de tela no formato 4:3 (ou 3:4 se o monitor estivesse orientado verticalmente como é no caso do **Pac Man**), a essa altura do campeonato é difícil encontrar nos dias de hoje um monitor ou TV que tenha uma proporção de tela no formato 4:3.

A boa notícia é que esse espaço extra que sobra nas laterais não é desperdiçado. Muitos gabinetes de arcade na época utilizavam uma moldura com ilustrações ao redor da tela, caso você tenha esses arquivos o MAME também irá exibir essas ilustrações na tela. Para se obter um melhor resultado, ative o visualizador de ilustrações e selecione o modo recortado¹.

Alguns monitores de LCD mais antigos usavam uma resolução nativa de 1280x1024 onde tinham uma proporção de tela no formato 5:4. Neste exemplo, não há muito espaço extra suficiente para exibir a ilustração e você vai notar um leve esticamento vertical, porém os resultados ainda serão bons o suficiente, como se fossem um monitor com formato 4:3.

¹ Cropped do Inglês. (Nota do tradutor)

7.6.2 Introdução ao GLSL

Antes de começar, você precisará seguir as instruções de configuração inicial do MAME encontrada em outra parte deste manual. As distribuições oficiais do MAME já são compatíveis com o glsl, mas **NÃO** incluem os arquivos de sombreado glsl. Você precisará obter esses arquivos de sombreado através de um outro fornecedor qualquer pela internet.

Abra o seu `mame.ini` no seu editor de texto preferido como o bloco de notas por exemplo e verifique se as seguintes opções estejam definidas corretamente:

- `video opengl`
- `filter 0`

O primeiro é necessário pois o glsl requer suporte ao OpenGL. Já o último desliga os filtros extras que possam interferir com a saída glsl.

Por último, resta uma edição a mais para ativar o glsl:

- `gl_glsl 1`

Salve o arquivo `.ini` e já estamos pronto para começar.

7.6.3 Customizando as configurações GLSL de dentro do MAME

Por vários motivos complicados de explicar, as configurações glsl não são mais salvas quando você sai do MAME. Isso significa que apesar das configurações exigirem um pouco mais de trabalho de sua parte, os resultados sempre sairão conforme esperado.

Comece rodando o MAME com o jogo de sua preferência como por exemplo **mame pacman**.

Use a tecla til (`~`)² para chamar a tela de opções que vai aparecer na parte de baixo da tela. Use as teclas cima e baixo para navegar dentre as várias opções, enquanto as teclas esquerda e direita irão permitir que você altere o valor dessas opções. Os resultados aparecerão em tempo real conforme elas forem sendo alteradas.

Depois de encontrar as configurações desejadas, anote os números num bloco de notas e saia do MAME.

7.6.4 Alterando as configurações

Como descrito em *Diversos arquivos de configuração*, o MAME segue uma sequência na hora de processar os arquivos `.ini`. As configurações glsl podem ser editadas diretamente no arquivo `mame.ini`, porém para tirar melhor proveito do poder dos arquivos de configuração do MAME, talvez seja melhor copiar as opções do glsl do `mame.ini` para um outro arquivo de configuração e fazer as modificações lá.

Por exemplo, uma vez selecionada as opções glsl que deseja usar nos sistemas para Neo-Geo, coloque essas configurações no arquivo `neogeo.ini` dentro do diretório **ini/sources** para que essas configurações sejam compartilhadas com todos os sistemas Neo-Geo automaticamente. Caso contrário seria necessário ter que adicionar essas configurações uma a uma manualmente em diferentes arquivos `.ini` como o nome de cada sistema dentro do diretório **ini**.

² Até que o teclado **ABNT-2** seja mapeado pela equipe do MAMEDev, essa tecla fica do lado esquerdo da tecla 1, logo abaixo da tecla ESQ. (Nota do tradutor)

7.6.5 Opções disponíveis

`gl_glsl`

Caso seja igual à 1 ativa o glsl, desativa se for definido como 0.

O valor predefinido é 0.

`-gl_glsl_filter <valor>`

Habilita a interpolação da imagem **OpenGL GLSL**, os valores válidos³ são:

- 0, Simple: Método de interpolação rápida e menos precisa que deixa os pixels de forma serrilhada pois utiliza a técnica de interpolação do **vizinho mais próximo**.
- 1 Bilinear: Método de interpolação lenta e de qualidade mediana, suaviza a transição entre as cores dos pixels deixando a imagem mais suavizada como um todo. Veja também `-filter`.
- 2, Bicúbico: Método de interpolação lenta e mais precisa, suaviza a transição entre as cores dos pixels próximos gerando uma gradação mais suave. Também suaviza a imagem porém nem tanto como o método bilinear.

`glsl_shader_mame0`

...

`glsl_shader_mame9`

Especifica quais dos sombreadores usar, na ordem entre 0 a 9. Informe-se com o autor do seu pacote de sombreadores para saber em qual ordem rodar primeiro para que o efeito seja exibido de forma correta.

`glsl_shader_screen0`

...

`glsl_shader_screen9`

Determina em qual tela aplicar os efeitos.

³ <https://github.com/mamedev/mame/pull/2989/files>

7.7 Criando e usando um LUT 3D no MAME

- *Introdução*
- *Criando um HaldCLUT*
- *Criando um LUT 3D para usar no MAME*
- *Diferenças nas cores*
- *As paletas de cores*

7.7.1 Introdução

O LUT é um acrônimo de "Look Up Table", é um arquivo que contém informações usadas para aplicar correções de cores (LUT técnico¹) ou para fazer uma gradação de cores (alterar as cores) de uma maneira a aplicar certos efeitos ou para alterar o tom da imagem (LUT criativo) visando criar uma aparência diferente da original.

É muito comum a utilização de LUTs nas produções de vídeo e foto seja para dar uma certa aparência quente/fria/neutra a uma determinada cena ou para criar perfis personalizados para câmeras e para as lentes. Estes perfis servem para corrigir eventuais desvios ou até mesmo para fazer ajustes finos. Independentemente da marca ou do modelo do equipamento, é comum que eles apresentem desvios nas cores em determinadas condições de iluminação ou situações específicas, seja por causa do tipo da luz utilizada (natural ou artificial) ou por causa de fatores externos como a posição do sol em diferentes horários do dia, cenas com o tempo nublado, ambiente iluminado por velas e assim por diante.

Neste contexto, o LUT é utilizado para compensar ou para corrigir estas variações, garantindo que as cores de um determinado objeto, ou de uma pessoa, estejam consistentes em diferentes condições de luz. Por exemplo, um LUT pode ser criado para garantir que uma blusa com um tom específico de vermelho, apareça com a mesma tonalidade em diferentes condições de iluminação, proporcionando uma aparência visual uniforme durante todo o processo de produção.

Para aprender mais sobre LUT, consulte os links ao lado.²³⁴

A capacidade de aplicar LUT foi inserida no MAME em [Outubro de 2018](#), porém, mesmo após o seu lançamento no MAME 0.203, não houve nenhum grande anúncio ou qualquer manifestação conhecida da comunidade. Talvez porque a grande maioria das pessoas sequer sabem para que serve ou como se usa um LUT.

Neste capítulo vamos aprender a criar e aplicar LUTs de correção de cores no sistema *Nintendo Entertainment System* ou *NES* emulado pelo MAME.

¹ <https://www.avmakers.com.br/blog/l-u-t-o-que-e-e-como-utilizar>

² https://docs.darktable.org/usermanual/3.8/pt_br/module-reference/processing-modules/lut-3d/

³ <https://blog.emania.com.br/o-que-e-lut-look-up-table/>

⁴ <https://www.layerlemonade.com/colorizacao/as-diferencas-entre-1d-lut-e-3d-lut>

7.7.2 Criando um HaldCLUT

Para criar um CLUT que contenha todas as informações necessárias para corrigir as cores, é preciso seguir um processo que envolve o uso de uma imagem criada pelo MAME, que nos servirá como uma base e de uma outra imagem de referência, esta será usada como correção. Basicamente você está informando ao programa que o nosso alvo é corrigir as cores do MAME com as características da imagem de referência.

Se vamos criar um CLUT de correção de cores, antes de mais nada, nós precisamos ter uma referência que possamos confiar, assim sendo, eu criei uma tela com [barras de cores SMPTE](#) gerado no padrão [BT.601](#) com [luminância](#) no intervalo 16,235 ([nível de TV](#)). Este é um padrão de teste utilizado até os dias de hoje pela indústria e pelos profissionais na produção de vídeo, cinema, televisão e fotografia para aferir a precisão das cores em uma ampla variedade de equipamentos utilizados na indústria audiovisual. Ele foi escolhido pois era o padrão da época, logo, eu sei de onde estou partindo quanto ao tom e a intensidade das cores, contudo, isso **não garante** que a correção das cores será fidedigna em todos os jogos, pois o MAME usa uma paleta dinâmica [matematicamente calculada](#).

Para gerar a nossa imagem de referência, nós usamos o [AviSynth+](#) e criamos um arquivo `Colorbars.avs` com o seguinte conteúdo:

```
# Color bars SD
ColorBars (550,480)
ConvertToRGB32(matrix="Rec601")
```

Arrastando o arquivo `Colorbars.avs` no [Virtualdub2](#) nós obtemos a seguinte imagem:



Ela servirá como a referência de correção que será aplicada na imagem do MAME.

Nota: Para usuários de Linux e macOS, é possível obter tela semelhante usando o [ffmpeg](#), depois de instalar o `ffmpeg` abra o terminal e rode o comando: `ffmpeg -hide_banner -f lavfi -i smptebars=duration=10:size=550x480:rate=30 -pix_fmt yuv444p smptebars.mp4`

Baixe o [240p-test-mini](#) e salve na mesma pasta do MAME, ele será usado para gerar a tela com barras coloridas, usando o terminal ou prompt de comando faça:

```
mame nes -cart 240pee.nes -resolution 552x481 -noka -window -nofilter -video opengl
```



Pressione o botão "B" e selecione a opção **SMPTE color bars**, pressione Alt + PrtScn (*Print Screen*) para fazer uma captura da tela (no Windows ou Linux, para macOS use [estas instruções](#)), este é o resultado (já recortado):



Nota: Caso tenha alguma dificuldade, aproveite as imagens acima que já estão recortadas no tamanho apropriado.

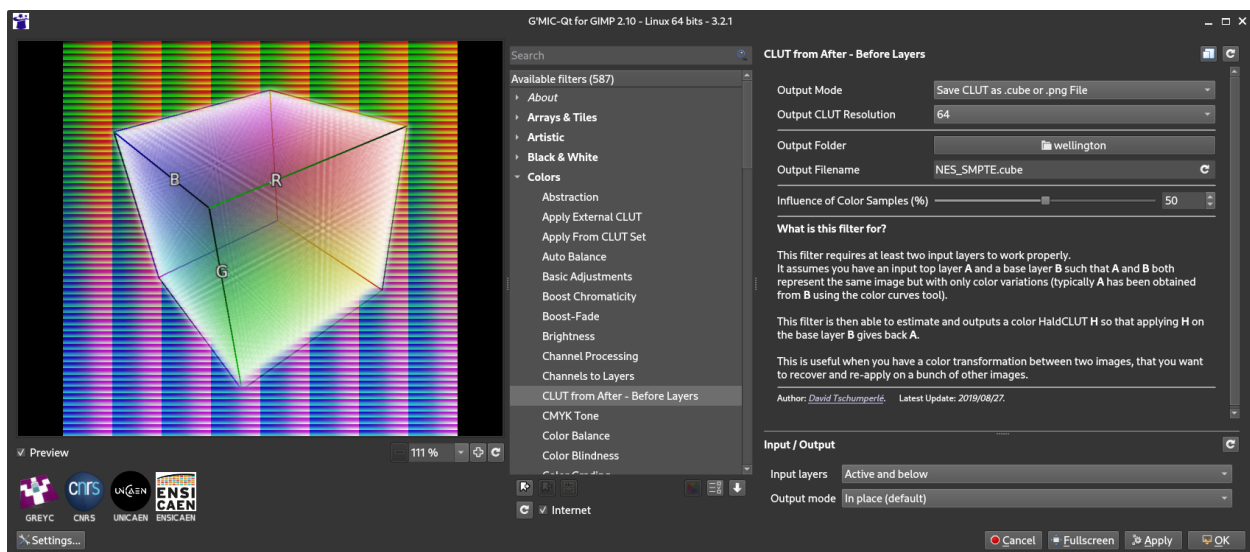
O que faremos agora é transferir as cores da imagem do MAME para a nossa imagem de referência. Isso é necessário pois, para criar um CLUT válido, ambas as imagens precisam ter o mesmo tamanho e as mesmas características, a única coisa que deve variar na imagem são as cores e mais nada (observe que a barra verde tem um tamanho diferente por exemplo), caso contrário, o resultado será diferente do esperado. Se não estiver interessado em aprender como criar um CLUT ou queira evitar todo este procedimento, baixe o arquivo NES_SMPTE.cube [deste link](#) e [pule para a próxima etapa](#).

Para esta tarefa usaremos o [Gimp](#) com o plug-in [G'MIC](#), abra a nossa imagem de referência no Gimp e duplique-a, nomeie esta primeira camada como "**MAME**" e a segunda como "**REFERÊNCIA**", é na camada "**MAME**" onde clonaremos as cores do MAME.

- Abra o imagem do MAME (será criado uma nova aba), com o "Color Picker" () clique na primeira barra à esquerda para capturar a sua cor.
- Volte para a primeira aba, selecione a ferramenta "Fuzzy Select Tool" (), clique na camada "**MAME**" para selecioná-la, clique na primeira barra à esquerda para criar uma seleção ao redor da cor.
- Em seguida escolha a ferramenta "Bucket Fill Tool" (Shift + B) e clique em cima da barra selecionada para preencher esta barra com as cores do MAME.
- Repita este procedimento com todas as outras barras.
- Selecione a camada "REFERÊNCIA", logo abaixo, clique na seta para cima para mover a camada para o topo. Assim você terá a primeira camada "**REFERÊNCIA**" e abaixo "**MAME**".

- Vá em *Filters > G'MIC-Qt...*, depois em *Available Filters*, clique duas vezes em *Color* e clique em *CLUT from After - Before Layers*.
- Em *Output Mode* selecione *Save CLUT as .cube or .png File*.
- Em *Output CLUT Resolution* deixe em 64.
- Em *Output Folder* defina uma pasta de destino para salvar o arquivo .cube.
- Em *Output Filename* defina o nome do arquivo com a extensão .cube, aqui usaremos NES_SMPTE.cube.
- Logo abaixo em **Input / Output** defina *Input Layers* como *Active and below*.
- Em *Output mode* selecione *In place (default)*.
- Clique em OK para criar o arquivo NES_SMPTE.cube.

A imagem abaixo mostra todas as configurações, clique nela para ampliar (HTML) ou aproxime a página (PDF).



Agora todas as informações necessárias para aplicar as correções de cores estão armazenadas no arquivo NES_SMPTE.cube. Acredite, o processo é muito mais fácil quando trabalhamos com fotos.

7.7.3 Criando um LUT 3D para usar no MAME

O tipo de LUT compatível com o MAME é o 3D no formato 64x64x64 com **8-bit** e **PNG**⁵. Ele é criado usando o **3D Lut Maker** que acompanha o **DisplayCAL**, ele é gratuito e está disponível para todos os sistemas operacionais, no **Debian** instale com o comando:


```
sudo apt install displaycal dispalcgui
```

No **Windows** é preciso baixar a versão para Windows no link acima e baixar também o **Argyll Color Management** (ArgyllCMS). Não é preciso baixar o instalador do DisplayCalc, baixe a versão ZIP e descompacte-o em algum lugar (menos em C:\), depois de baixar o ArgyllCMS, descompacte-o dentro da pasta do DisplayCalc. Quando iniciar o programa **DisplayCAL-3DLUT-maker**, ele vai te questionar para localizar a pasta onde se encontra os executáveis do **ArgyllCMS**, direcione o explorer para a pasta onde você descompactou o ArgyllCMS, entre na pasta **bin** e clique em "selecionar pasta" para que o programa inicie corretamente.


Aqui as configurações utilizadas no Linux:

⁵ https://www.reddit.com/r/MAME/comments/9tp2l1/using_3d_luts_with_mame/

⁶ <https://github.com/mamedev/mame/pull/4043/commits/372982391d04c24473ba6bab1b87a0f50066ddd>

 **Create 3D LUT** — ⌵ ✕

Source profile

SMPTE 240M ⌵ 

Tone curve


☐ Unmodified

☐ Apply black output offset (100%)


☒ Gamma 2.2 ⌵ Gamma 2.2 ⌵ Relative ⌵

Black output offset ⌵ 100 — + — %

☐ Abstract ("Look") profile

⌵ 

Destination profile

DisplayCAL calibration preset: sRGB ⌵  Current profile

☐ Apply calibration (vcgt)

Gamut mapping mode

☒ Inverse device-to-PCS

☐ PCS-to-device

Rendering intent

Absolute colorimetric with white point scaling ⌵

3D LUT file format

PNG (.png) ⌵

Input encoding

TV RGB 16-235 ⌵

Output encoding

TV RGB 16-235 ⌵

3D LUT resolution

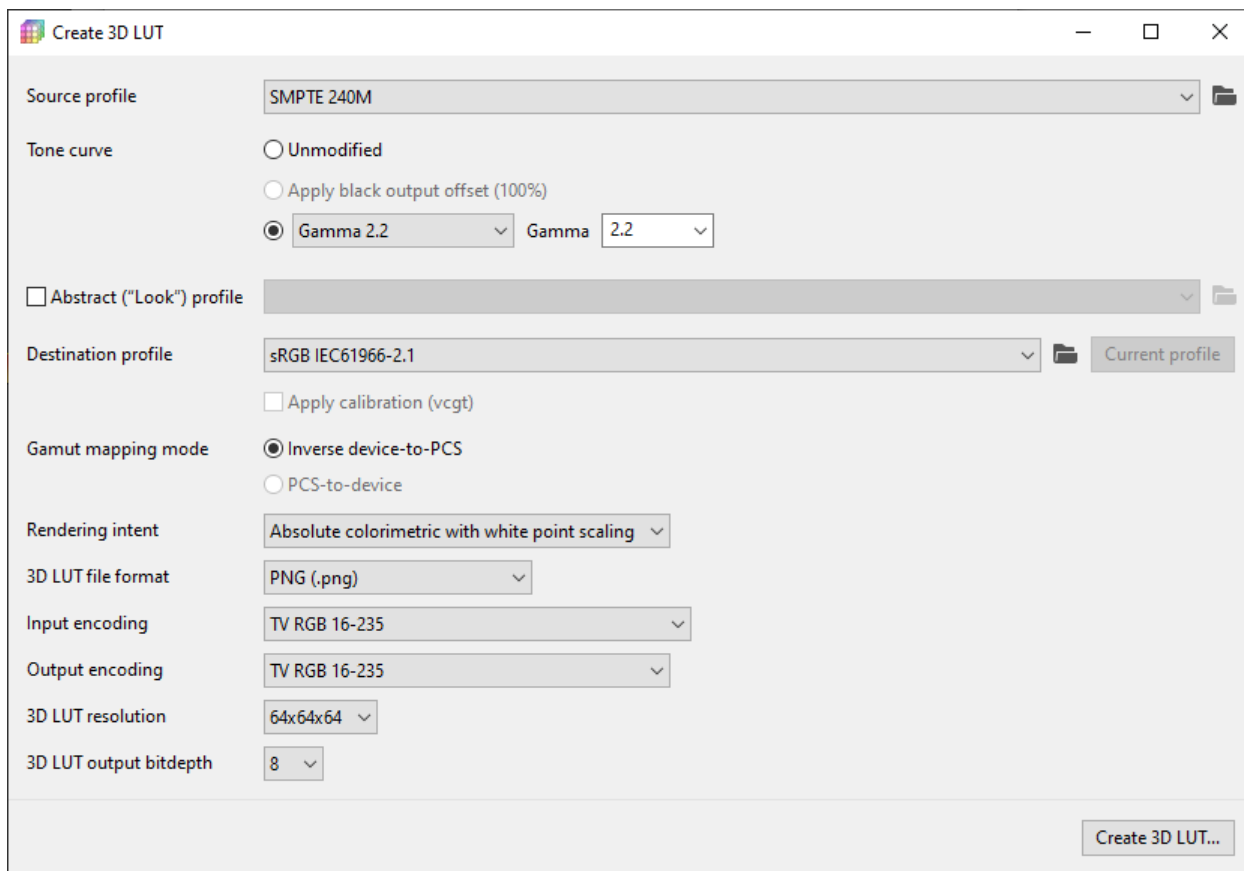
64x64x64 ⌵

3D LUT output bitdepth

8 ⌵

Create 3D LUT...

Aqui as configurações utilizadas no Windows:



Ao clicar em *Create 3D LUT* o programa cria diversos arquivos diferentes, porém, o que nos interessa é esta imagem aqui:



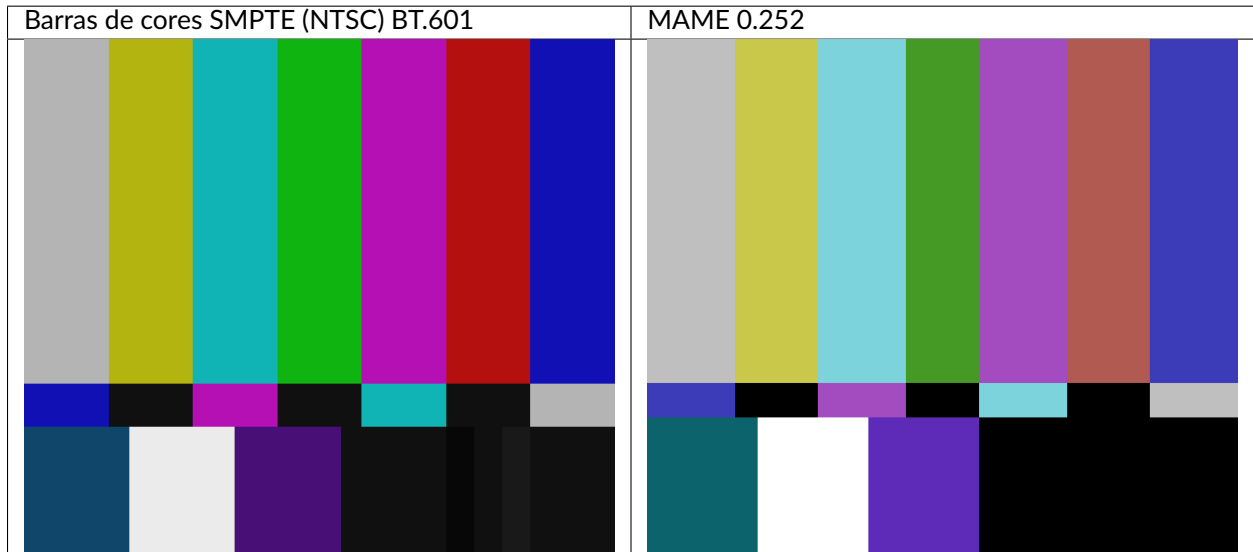
Esse é o nosso **LUT 3D base**, é nele que aplicaremos as correções de cores, e depois a partir dele, que exportaremos um novo LUT 3D com as correções que desejamos. As opções escolhidas foram as que me pareceram mais corretas. Faça você mesmo os seus próprios testes com outros parâmetros e veja como fica.

7.7.4 Diferenças nas cores

De todos os diferentes emuladores existentes que emulam o NES apresentam uma paleta de cores diferente entre si (incluindo o MAME) e não existe (até onde eu saiba) um consenso sobre qual deveria ser a paleta de cores "*mais correta*" para o NES pois a sua PPU *não gera sinais de vídeo em RGB* ficando a cargo do circuito interno da TV da pessoa interpretar as cores que chegavam nela, a partir daí começa toda essa confusão, pois os circuitos internos das TVs eram diferentes, assim sendo, as cores variavam de uma TV para outra. No MAME por exemplo, a paleta de cores é *matematicamente calculada*, já outros emuladores utilizam outros algoritmos ou paletas para apresentar as cores na tela.

Veja como fica ao colocarmos a nossa imagem de referência ao lado da imagem gerada pelo MAME, repare na diferença do tom e na intensidade das cores:

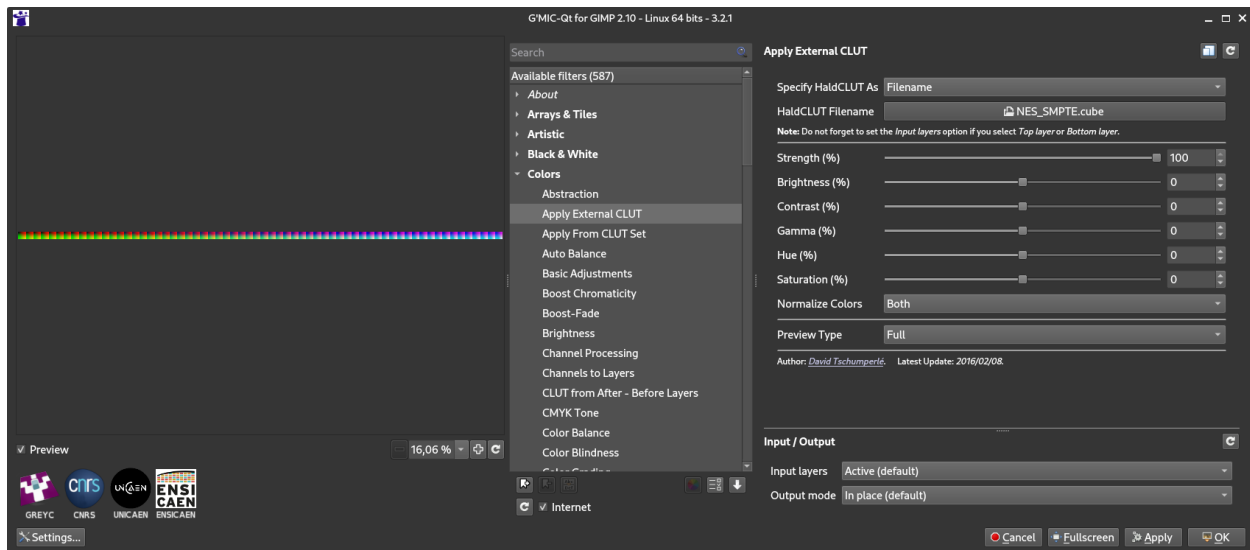
Tabela 1: Comparando a imagem de referência com o padrão gerado pelo MAME.



Com o nosso CLUT e o nosso LUT 3D base criados, nós já temos o nosso ponto de partida para dar início a criação do nosso LUT 3D de correção, o procedimento é o seguinte:

- Abra a imagem do **LUT 3D base** que criamos acima com o Gimp.
- Vá em *Filters > G'MIC-Qt...*, depois em *Available Filters*, clique duas vezes em *Color* e escolha *Apply External CLUT*.
- Em *Specify HaldCLUT As* escolha *Filename*.
- Em *HaldCLUT Filename* procure pelo arquivo `NES_SMPTE.cube`.
- Em *Normalize Colors* selecione *Both*.
- Em *Preview Type* selecione *Full*.
- Em *Input layers* deixe em *Active (default)*.
- Em *Output mode* deixe em *In place (default)*.
- Clique em *Ok* para aplicar as correções.

Veja na imagem abaixo como ficou as nossas opções:



Observe que não é necessário alterar qualquer outra configuração (brilho e contraste por exemplo), pois tudo o que precisamos já está no arquivo NES_SMPTE.cube. Mesmo quando trabalhamos com fotos ou geramos LUT para diferentes finalidades, todos os ajustes finos que precisamos fazer na imagem é feito antes de se criar o arquivo *.cube e não depois. Claro que é possível fazer alguns ajustes finos se for necessário, mas, para a nossa finalidade aqui com o MAME os ajustes que já foram feitos são suficientes.

Nota: Geralmente quando baixamos um LUT externo (geralmente um LUT criado por outra pessoa) é que estes controles se tornam úteis, servem como um ajuste fino no efeito que desejamos aplicar. Porém com o MAME mesmo pequenos ajustes podem extrapolar o efeito desejado.

Vá em *File > Export As...* (Shift + Ctrl + E) e salve o arquivo com um nome bacana como NES_SMPTE.png dentro da pasta **Artwork** do MAME.

Este é o LUT 3D NES_SMPTE.png que acabamos de criar:



Para ver se funcionou ou não, crie o arquivo ini\source\nes.ini com o seguinte conteúdo:

```
video bgfx
bgfx_backend d3d11
bgfx_screen_chains lut
bgfx_lut NES_SMPTE.png
```

Para quem usa HLSL no Windows:

```
video d3d
hls_enable 1
filter 0
lut_enable 1
lut_texture NES_SMPTE.png
saturation 1.36
```

Nota: Caso tenha problemas com o d3d (o efeito não funciona ou o LUT não é aplicado por exemplo), consulte o capítulo *Usando o HLSL no Windows 8/10/11 sem precisar instalar o DirectX 9*.

Para Linux/macOS tente:

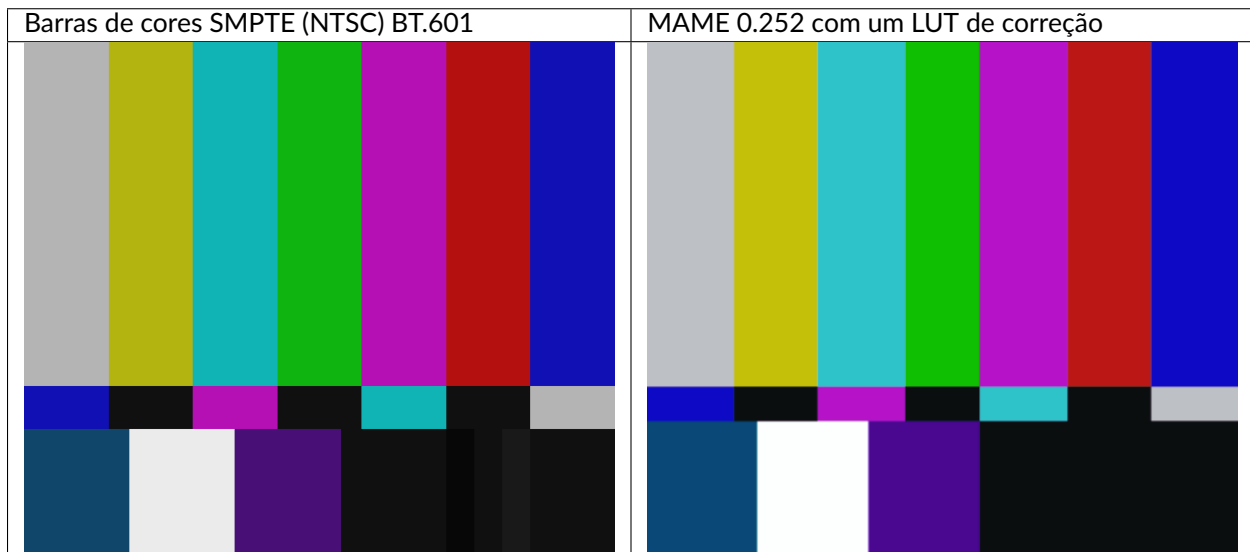
```
video bgfx
bgfx_backend vulkan # (tente opengl ou auto caso não funcione)
bgfx_screen_chains lut
bgfx_lut NES_SMPTE.png
prescale 5
```

Rode o comando abaixo no terminal ou no prompt de comando e veja o resultado:

```
mame nes -cart 240pee.nes -noka
```

Pressione o botão "B" e selecione a opção **SMPTE color bars**, veja na imagem abaixo como o próprio MAME é capaz de aplicar as devidas correções em tempo real:

Tabela 2: O próprio MAME usando um LUT 3D para aplicar as correções de cores.



7.7.5 As paletas de cores

Nota: Até a presente data em que escrevo este capítulo, não há suporte para o uso de paleta de cores de qualquer tipo na versão mais recente do MAME (0.252).

Se pesquisar na internet, é possível encontrar diferentes sites ([como este](#)) que disponibilizam uma grande variedade de paletas de cores (elas vem com a extensão .pal) que são utilizadas por outros emuladores como o [nestopia](#), [Mesen2](#), [FCEUX](#), dentre vários outros, para que estes emuladores consigam gerar cores na tela com um determinado padrão, ou obedece certos parâmetros de cores determinada pelos seus criadores. Para obter mais informações, consulte os links ao lado (todos em Inglês).⁷⁸⁹

Para compreender como funciona a geração dos gráficos no NES, consulte o link ao lado (em Inglês).¹⁰

⁷ https://www.nesdev.org/wiki/NTSC_video

⁸ <http://www.firebrandx.com/nepalette.html>

⁹ <https://forums.nesdev.org/viewtopic.php?t=9027>

¹⁰ https://www.youtube.com/watch?v=7Co_8dC2zb8

Dada a grande variedade de paletas e como não existe (até o presente momento) uma paleta definitiva para o NES, eu escolhi a paleta do [ROM Detectives](#) por uma simples questão de gosto, geralmente é assim que acontece, das várias disponíveis você usa aquelas que mais gosta.

Tabela 3: Paleta do ROM Detectives

ROM Detectives NES Palette

Aqui a mesma paleta convertida num arquivo chamado NES_NTSC.pal.

00000000	7c 7c 7c 00 00 fc 00 00	bc 44 28 bc 90 00 84 a8D(....
00000010	00 20 a8 10 00 88 14 00	50 30 00 00 78 00 00 68P0..x..h
00000020	00 00 58 00 00 40 58 00	00 00 00 00 00 00 00 00	..X..@X.....
00000030	bc bc bc 00 78 f8 00 58	f8 68 44 fc d8 00 cc e4x..X.hD....
00000040	00 58 f8 38 00 e4 5c 10	ac 7c 00 00 b8 00 00 a8	.X.8..\..
00000050	00 00 a8 44 00 88 88 00	00 00 00 00 00 00 00 00	...D.....
00000060	f8 f8 f8 3c bc fc 68 88	fc 98 78 f8 f8 78 f8 f8	...<..h...x..x..
00000070	58 98 f8 78 58 fc a0 44	f8 b8 00 b8 f8 18 58 d8	X..xX..D.....X..
00000080	54 58 f8 98 00 e8 d8 78	78 78 00 00 00 00 00 00	TX.....xxx.....
00000090	fc fc fc a4 e4 fc b8 b8	f8 d8 b8 f8 f8 b8 f8 f8
000000a0	a4 c0 f0 d0 b0 fc e0 a8	f8 d8 78 d8 f8 78 b8 f8x..x..
000000b0	b8 b8 f8 d8 00 fc fc f8	d8 f8 00 00 00 00 00 00

Novamente, até o momento, o MAME não é compatível com paletas externas, assim sendo, precisamos converter a paleta em LUT 3D.

Nota: O nosso **LUT 3D base** deve estar limpo e sem nenhum efeito ao aplicar uma nova correção. Então, caso a correção do processo anterior já esteja aplicado, faça um Ctrl + Z para desfazer a correção e só então, aplique uma nova. Na dúvida e sem salvar nada, feche a aba com o **LUT 3D base**, abra-a novamente, aplique as novas correções e só então exporte como **NES_NTSC.png**. Se não fizer isso, haverá duas correções diferentes no mesmo arquivo e o resultado final será muito diferente do desejado.

- Carregue a ROM **240p-test-mini** no **nestopia**, **Mesen2**, **FCEUX** ou qualquer outro emulador que aceite paleta de cores.
- Configure o vídeo para que ele tenha o dobro de tamanho do original.
- Pressione o botão "B" e selecione a opção **SMPTE color bars**.
- Carregue e aplique a paleta **NES_NTSC.pal**.
- Pressione **Alt + PrtScn** (*Print Screen*) para fazer uma captura da tela (no Windows ou Linux, para o macOS use [estas instruções](#)), encerre o emulador.
- Abra o **Gimp** e cole a imagem, dê o nome de **"CORREÇÃO"**.
- Use a [captura anterior](#)) e cole no Gimp, dê o nome de **"MAME"**.
- Faça os ajustes necessários para que ambas as imagens fiquem alinhadas.

- Faça um recorte (*crop*) para que ambas tenham exatamente o mesmo tamanho.
- Mova a camada "**MAME**" para baixo, deixando a camada "**CORREÇÃO**" em primeiro.
- Vá em *Filters > G'MIC-Qt...*, depois em *Available Filters*, clique duas vezes em *Color*, clique em *CLUT from After - Before Layers*.
- Em *Output Mode* selecione *Save CLUT as .cube or .png File*.
- Em *Output CLUT Resolution* deixe em 64.
- Em *Output Folder* defina uma pasta de destino para salvar o arquivo .cube.
- Em *Output Filename* defina o nome do arquivo com a extensão .cube, aqui usaremos NES_NTSC.cube.
- Logo abaixo em **Input / Output** defina *Input Layers* como *Active and below*.
- Em *Output mode* selecione *In place (default)*.
- Clique em OK para criar o arquivo NES_NTSC.cube.
- Faça o procedimento descrito em *Diferenças nas cores* para aplicar o CLUT NES_NTSC.cube *no LUT 3D base que criamos*.
- Vá em *File > Export As...* (Shift + Ctrl + E) e salve o arquivo como **NES_NTSC.png** dentro da pasta **Artwork** do MAME.

Ao concluir estas etapas, agora nós temos o LUT 3D NES_NTSC.png:



Para testar, edite o arquivo `ini\source\nes.ini` e substitua a opção `NES_SMPTE.png` por `NES_NTSC.png` e salve. Agora teste um jogo qualquer do NES e veja como fica, tente o *Batman* por exemplo:

mame nes batmanu

Usando a própria interface do MAME, é possível alternar entre os dois LUTs 3D que criamos (NES_NTSC e o NES_SMPTE):

- Pressione Tab.
- Seleccione *Controles deslizantes*.
- Mova o cursor até *Window 0, Screen 0 LUT Texture*.
- Seleccione esquerda ou direita para alternar entre os LUTs.

Para quem tiver interesse, aqui está a paleta **NES SMPTE**.pal.

Tabela 4: Paleta Mamedoc SMPTE



Mamedoc NES SMPTE Palette

A 4x14 grid of color swatches representing the Mamedoc NES SMPTE Palette. The colors are arranged in a gradient from dark to light, with a black background on the right side.

Row	Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10	Col 11	Col 12	Col 13	Col 14
1	Dark Grey	Blue	Blue	Purple	Magenta	Red	Red	Red	Brown	Green	Green	Green	Dark Blue	Black
2	Light Grey	Blue	Blue	Purple	Magenta	Red	Red	Red	Brown	Green	Green	Green	Dark Blue	Black
3		Cyan	Blue	Purple	Magenta	Red	Red	Red	Yellow	Yellow	Green	Green	Cyan	Grey
4		Cyan	Blue	Purple	Magenta	Red	Red	Red	Yellow	Yellow	Green	Green	Cyan	White

```

00000000 85 7a 7d 07 03 de 09 04 c0 3b 0e 9a a7 03 83 bc |.z}.....;.....|
00000010 02 15 bc 0c 01 9f 11 03 5b 37 03 02 9a 05 02 88 |.....[7.....|
00000020 07 02 74 07 06 3e 69 00 00 00 00 00 00 00 00 |..t..>i.....|
00000030 c8 c9 c9 03 73 e4 03 4b e0 70 2e d4 d2 01 ce da |...s..K.p.....|
00000040 01 4a e6 28 00 d7 4c 0c bb 7a 03 01 c6 01 01 bf |.J.(..L..z.....|
00000050 01 04 af 3b 05 7f 8f 00 00 00 00 00 00 00 00 |...;.....|
00000060 ff ff ff 1e ba d6 51 81 d8 a4 72 de e1 7b e4 de |.....Q...r...{..|
00000070 4a a2 dc 72 44 d6 a3 24 e3 c1 01 c0 dc 0d 4f cf |J..rD..$......0..|
00000080 49 3a d6 93 06 de d5 80 74 77 00 00 00 00 00 00 |I:.....tw.....|
00000090 ff ff ff 91 e4 ea c4 c8 e0 dd c7 e8 ea c4 eb dc |.....|
000000a0 b3 ca e2 d8 bd e8 e0 b1 e1 d6 6d d6 e1 6d c5 e0 |.....m..m..|
000000b0 c7 b9 eb e0 06 e8 e8 f3 e4 f3 00 00 00 00 00 00 |.....|

```

O processo de criação desta paleta é semelhante ao que já foi explicado:

- Abra a imagem da paleta do *ROM Detectives* no Gimp.
- Vá em *Filters > G'MIC-Qt...*, depois em *Available Filters*, clique duas vezes em *Color* e escolha *Apply Extrenal CLUT*.
- Em *Specify HaldCLUT As* escolha *Filename*.
- Em *HaldCLUT Filename* procure pelo arquivo *NES_SMPTE.cube*.
- Em *Normalize Colors* selecione *Both*.
- Em *Preview Type* selecione *Full*.
- Em *Input layers* deixe em *Active (default)*.
- Em *Output mode* deixe em *In place (default)*.
- Clique em *Ok* para aplicar as correções.

Extraia os valores em hex ("html notation" no Gimp) das cores e cole num editor hexadecimal e salve como *nome_da_paleta.pal*.

Eu compilei um conjunto de **37 LUTS**, incluindo os que criamos aqui neste documento, quem tiver interesse de baixar e experimentar, [baixe aqui](#) (nova *versão 2023*, as tabelas de cores foram atualizadas e corrigidas para o MAME 0.261 ou mais recente), descompacte os arquivos .png dentro da pasta **artwork**. Você pode ou configurar uma paleta específica em *ini\source\nes.ini* ou usar a própria interface do MAME para alternar entre elas como já foi explicado [anteriormente](#).

Nota: Agora estes LUTs também estão disponíveis através do site do MAMEDEV em [mame-goodies](#).

Nota: Para remover o efeito "*blur*" (borrado) nas versões OpenGL do MAME durante o uso de Luts com o **nes**, use a opção *prescale* no arquivo *ini\source\nes.ini*. O valor 5 (médio), remove todo o efeito.

7.8 Arquivos de configuração para os controles

- *Introdução*
- *Estrutura básica*
- *Remapeando os controles já predefinidos*
- *Substituindo os controles predefinidos pelo tipo da entrada*
- *Substituindo a configuração predefinida para uma tecla específica*
- *Atribuindo números aos dispositivos de entrada*
- *Definição das opções de entrada do ponteiro*

7.8.1 Introdução

Os arquivos de configuração para o controle podem ser usados para alterar as configurações da entrada predefinida do MAME. Tais arquivos de configuração podem ser usados com um dispositivo de entrada fornecendo padrões mais adequados, ou usados como perfis que possam ser selecionados para diferentes situações. O MAME inclui alguns exemplos de arquivos de configuração na pasta **ctrlr** desenvolvidos para fornecer padrões úteis para certos controles do tipo arcade.

Apesar da extensão `.cfg`, estes arquivos estão no formato XML. O MAME procura por tais arquivos nos diretórios definidos com a opção `ctrlrpath`. Um arquivo de configuração será selecionado quando a opção `ctrlr` seguido pelo nome do arquivo sem a extensão `.cfg`, exemplo:

```
ctrlr scorpionxg
```

Isso faz com que o MAME use o arquivo **ctrlr\scorpionxg.cfg**.

Haverá um erro caso o MAME identifique que o arquivo de configuração informado não exista ou caso o seu conteúdo não tenha nenhuma seção aplicável ao sistema que está sendo emulado.

A utilização dos símbolos ou dos *tokens* de entrada variam conforme a aplicação. A precisão dos valores disponíveis e os seus significados dependem da versão exata do MAME que está sendo usado, dos dispositivos conectados na entrada, dos módulos do provedor da entrada selecionada (as opções `keyboardprovider`, `mouseprovider`, `lightgunprovider` e `joystickprovider`) e possivelmente das outras configurações.

7.8.2 Estrutura básica

A sua estrutura segue um formato similar ao formato usado pela configuração do sistema que o MAME usa para salvar coisas como as configurações da entrada e os dados da contabilidade do sistema (criado na pasta que for definido usando a opção `cfg_directory`). Neste exemplo temos a estrutura geral de um arquivo de configuração de um controle:

```
<?xml version="1.0"?>
<nameconfig version="10">
  <system name="default">
    <input>
      <!-- as configurações que afetam todos os sistemas emulados vão aqui -->
    </input>
  </system>
  <system name="neogeo">
    <input>
      <!-- as configurações que afetam o neogeo e os seus clones vão aqui -->
    </input>
  </system>
  <system name="intellec4.cpp">
    <input>
      <!-- as configurações que afetam todos os sistemas definidos no driver intellec4.
      →cpp vão aqui -->
    </input>
  </system>
</nameconfig>
```

A base de um arquivo de configuração do controle deve ser um elemento `nameconfig`, com um atributo `version` que especifica a versão do formato desta configuração (atualmente 10 - O MAME não carregará um arquivo usando qualquer outra versão). O elemento `nameconfig` contém um ou mais elementos do sistema, cada qual com um nome do atributo especificando os sistemas aos quais eles se aplicam. Cada elemento `system` pode conter um elemento `input` que mantém o remap atual e os elementos de configuração da porta `port` que serão descritos mais adiante. Cada elemento `system` também pode conter um elemento `pointer_input` para definir as opções de entrada do ponteiro para sistemas com ilustração interativa.

Ao iniciar a emulação de um sistema o MAME aplicará a configuração dos elementos `system` onde o valor do atributo `name` atende a um dos seguintes critérios:

- Caso o atributo `name` tenha o valor `default`, ele sempre será aplicado (isso inclui os menus de seleção do sistema/software).
- Caso o atributo `name` tenha o mesmo nome do sistema, das suas variantes relacionadas ao sistema ou o nome encurtado da BIOS do sistema (caso seja aplicável).
- Caso o atributo `name` tenha o mesmo nome do arquivo fonte do driver.

Para o sistema *"DaeJeon! SanJeon SuJeon (AJTUE 990412 V1.000)"* por exemplo, os elementos `system` serão aplicados caso o seu atributo `name` tenha o valor `default` (se aplica a todos os sistemas), `sajeon` (um nome encurtado do próprio sistema), `sasissu` (um nome encurtado de uma outra versão do mesmo sistema), `stvbios` (um nome encurtado do nome da BIOS do sistema) ou `stv.cpp` (nome do arquivo fonte/driver onde o sistema foi definido).

Num outro exemplo, será aplicado aos sistemas *"The Invaders"*, *"Super Invader Attack"* (um bootleg do The Invaders) e *"Dodgem"* caso um elemento `system` onde o atributo `name` tenha o valor `zac2650.cpp`.

Os elementos `system` são aplicados na ordem em que aparecem no arquivo de configuração. As configurações dos elementos que aparecem ao final do arquivo podem modificar ou alterar as configurações dos elementos anteriores. Dentro de um elemento `system`, os elementos `remap` são aplicados antes dos elementos `port`.

7.8.3 Remapeando os controles já predefinidos

É possível usar o elemento `remap` para substituir uma entrada do host para um outro qualquer na configuração padrão do MAME. O exemplo abaixo substitui as teclas no teclado numérico para as teclas direcionais do cursor:

```
<input>
  <remap origcode="KEYCODE_UP" newcode="KEYCODE_8PAD" />
  <remap origcode="KEYCODE_DOWN" newcode="KEYCODE_2PAD" />
  <remap origcode="KEYCODE_LEFT" newcode="KEYCODE_4PAD" />
  <remap origcode="KEYCODE_RIGHT" newcode="KEYCODE_6PAD" />
</input>
```

O atributo `origcode` define o *token* para a entrada do host que será substituído, o atributo `newcode` define o *token* para a entrada do host que será substituído. Neste caso, são as atribuições que usa o cursor para cima, para baixo e as setas para a esquerda e para a direita, elas serão substituídas pelas teclas numéricas 8, 2, 4 e 6 do teclado numérico.

Observe que as substituições indicadas usando os elementos `remap` se aplicam apenas às entradas que usam a atribuição padrão do MAME para o tipo do controle. Ou seja, elas só se aplicam às atribuições padrão para os tipos de controle definidos no menu *Entradas (gerais)*. Eles não se aplicam às atribuições padrão das entradas definidas nas definições das portas de E/S do driver/dispositivo (usando a macro `PORT_CODE`).

O MAME aplica os elementos `remap` encontrados dentro de qualquer elemento `system` que seja aplicável.

7.8.4 Substituindo os controles predefinidos pelo tipo da entrada

Utilize os elementos `port` com os atributos `type` sem os atributos `tag` para substituir as definições de entrada padrão do host para os controles.

```
<input>
  <port type="UI_MENU">
    <newseq type="standard">KEYCODE_TAB OR KEYCODE_1 KEYCODE_5</newseq>
  </port>
  <port type="UI_CANCEL">
    <newseq type="standard">KEYCODE_ESC OR KEYCODE_2 KEYCODE_6</newseq>
  </port>
  <port type="P1_BUTTON1">
    <newseq type="standard">KEYCODE_C OR JOYCODE_1_BUTTON1</newseq>
  </port>
  <port type="P1_BUTTON2">
    <newseq type="standard">KEYCODE_LSHIFT OR JOYCODE_1_BUTTON2</newseq>
  </port>
  <port type="P1_BUTTON3">
    <newseq type="standard">KEYCODE_Z OR JOYCODE_1_BUTTON3</newseq>
  </port>
  <port type="P1_BUTTON4">
    <newseq type="standard">KEYCODE_X OR JOYCODE_1_BUTTON4</newseq>
  </port>
</input>
```

A configuração acima define as seguintes atribuições das entradas:

- **Mostra/esconde o menu** (Interface do usuário).
Tecla Tab, ou pressionando as teclas 1 e 2 simultaneamente.
- **UI Cancel** (Interface do usuário).
Tecla ESC, ou pressionando as teclas 2 e 6 simultaneamente.
- **P1 Button 1** (Controles do jogador 1).
Tecla C, ou o botão 1 do joystick 1.
- **P1 Button 2** (Controles do jogador 1).
Tecla Shift esquerda, ou o botão 2 do joystick 1.
- **P1 Button 3** (Controles do jogador 1).
Tecla Z, ou o botão 3 do joystick 1.
- **P1 Button 4** (Controles do jogador 1).
Tecla X, ou o botão 4 do joystick 1.

Repare que isto será aplicado somente às entradas do controle do MAME. Ou seja, os elementos port sem os atributos tag substituem apenas as atribuições predefinidas no menu "**Entradas (gerais)**". Eles não substituem as atribuições das entradas definidas nas definições das portas de E/S do driver/dispositivo (usando a macro PORT_CODE).

O MAME aplica os elementos port sem os atributos tag encontrados dentro de qualquer elemento system.

7.8.5 Substituindo a configuração predefinida para uma tecla específica

Utilize os elementos port com os atributos tag, type, mask e defvalue para substituir os valores predefinidos para controles específicos. Estes elementos port devem ser definidos dentro dos elementos system para que sejam apenas aplicados em determinados sistemas ou o código-fonte do driver (eles não devem existir dentro dos elementos system onde o atributo name tenha o valor default). A atribuição da entrada predefinida do host pode ser substituída assim como também é possível alternar as configurações dos controles digitais.

Os atributos tag, type, mask e o defvalue são usados para identificar a entrada em questão. É possível encontrar os valores usados para uma determinada entrada do host alternando a sua atribuição, encerrando o MAME e verificando os valores no arquivo de configuração do sistema. Observe que não há garantias que estes valores sejam os mesmos e podem variar entre as versões do MAME.

Abaixo um exemplo que substitui as entradas predefinidas para o **280-ZZZAP**:

```
<system name="280zzzap">
  <input>
    <port tag=":IN0" type="P1_BUTTON2" mask="16" defvalue="0" toggle="no" />
    <port tag=":IN1" type="P1_PADDLE" mask="255" defvalue="127">
      <newseq type="increment">KEYCODE_K</newseq>
      <newseq type="decrement">KEYCODE_J</newseq>
    </port>
  </input>
</system>
```

Esta configuração define as entradas para o estercamento esquerdo e direito para as teclas K e J respectivamente, desativando também as configurações do câmbio para a entrada relacionada com a troca de marchas.

7.8.6 Atribuindo números aos dispositivos de entrada

Use os elementos `mapdevice` em conjunto com os atributos `device` e `controller` para atribuir números fixos aos dispositivos de entrada, caso contrário, eles são alterados, consulte o capítulo *ID fixo para os controles* para obter mais informações. Observe que para que esta configuração funcione como esperado, é preciso que o dispositivo já esteja conectando quando o MAME for iniciado.

Defina o atributo `device` para a ID do dispositivo de entrada e defina o atributo `controller` para o token do dispositivo desejado (dispositivo, tipo e número).

Aqui está um exemplo ao enumerar duas pistolas de luz e dois controles de jogo do tipo `XInput`:

```
<system name="default">
  <input>
    <mapdevice device="VID_D209&PID_1601" controller="GUNCODE_1" />
    <mapdevice device="VID_D209&PID_1602" controller="GUNCODE_2" />
    <mapdevice device="XInput Player 1" controller="JOYCODE_1" />
    <mapdevice device="XInput Player 2" controller="JOYCODE_2" />
  </input>
</system>
```

O MAME apenas aplica os elementos `mapdevice` que forem encontrados dentro de qualquer elemento `system` quando for possível. Visando evitar confusão, é mais simples colocar o elemento `system` para que se aplique a todos os sistemas (o atributo `name` definido como `default`) logo na primeira linha do arquivo e fazer seu uso para atribuir os valores numéricos na entrada dos dispositivos.

7.8.7 Definição das opções de entrada do ponteiro

Um elemento `pointer_input` pode conter elementos `target` para definir as opções de entrada do ponteiro para cada tela ou janela final. Cada elemento `target` deve ter um atributo `index` que contém o índice da tela com base em zero à qual se aplica.

Cada elemento `target` pode ter um atributo `activity_timeout` para definir o tempo de inatividade do ponteiro do mouse quando este deixar de se movimentar ou não tiver os botões pressionados. O valor é especificado em segundos e deve estar no intervalo entre **0,1** a **10** segundos.

Cada elemento `target` pode ter um elemento `hide_inactive` para definir se os ponteiros considerados inativos poderão ser ocultados. Se o valor for `0` (zero), os ponteiros inativos não serão ocultados e se for `1`, os ponteiros inativos poderão ser ocultados, mas as visualizações de layout ainda poderão especificar que os ponteiros inativos não devem ser ocultados.

Veja a seguir um exemplo que demonstra a utilização deste recurso:

```
<system name="default">
  <pointer_input>
    <target index="0" activity_timeout="1.5" />
  </pointer_input>
</system>
<system name="intellec4.cpp">
  <pointer_input>
    <target index="0" hide_inactive="0" />
  </pointer_input>
</system>
```

Em todos os sistemas, os ponteiros sobre a primeira tela ou janela final serão considerados inativos após não se moverem por 1.5 segundos ou se nenhum botão for pressionado. Para os sistemas definidos em `intellec4.cpp`, os ponteiros inativos sobre a primeira janela não serão ocultados.

7.9 ID fixo para os controles

É predefinido que o MAME não atribua números fixos (IDs) para os dispositivos da entrada. Por exemplo, o driver de um controle "joystick 1" pode ser atribuído inicialmente para Joy 1, porém, depois de uma reinicialização o mesmo controle pode aparecer como Joy 3.

O MAME enumera os dispositivos conectados e os atribui IDs com base na ordem da enumeração do controlador. Os fatores que podem causar a alteração destas IDs são, conectar ou desconectar os controles ou dispositivos USB, alterar as portas ou "hubs", assim como até mesmo a reinicialização do computador. Tais números são impossíveis de se prever.

É aqui onde a configuração do mapdevice entra em ação. Ao adicionar esta configuração ao [arquivo de configuração do controle](#), é possível atribuir uma identificação fixa para o controlador do dispositivo, garantindo ao MAME que o dispositivo definido sempre seja mapeado com a mesma identificação.

7.9.1 Usando o mapdevice

O elemento XML mapdevice é adicionado na entrada do elemento XML do arquivo de configuração do controle. Dois atributos são necessários, device e controller. Observe que os elementos mapdevice apenas surtem efeito no arquivo de configuração do controle (definido através da opção *-ctrlr*). Eles são ignorados nos arquivos de configuração do sistema e no arquivo da configuração padrão.

O atributo device determina a ID do dispositivo correspondente da entrada. Também pode ser uma *substring* da ID do dispositivo. Para obter a ID do dispositivo da entrada, selecione-o no menu [Dispositivos de entrada](#):

- Inicie a emulação de um sistema qualquer (mame sf2 por exemplo).
- Pressione Tab e selecione *Configurações da entrada*.
- Faça um clique duplo em *Dispositivos de entrada*.
- Clique em qualquer um dos itens apresentados.
- Faça um clique duplo em *Copia a ID do dispositivo* para que a ID seja copiada para a área de transferência.

Também será possível ver as IDs dos dispositivos da entrada através do prompt de comando ou do terminal ao ativar o modo loquaz através da opção *-verbose* (mais detalhes [logo abaixo](#)). O formato das IDs dos dispositivos depende do tipo do dispositivo, do controlador do dispositivo (*driver*), do módulo do provedor da entrada selecionada e do sistema operacional. A ID dos seus dispositivo de entrada podem ser muito diferentes dos exemplos mostrados aqui.

O atributo controller atribui o token para o tipo do dispositivo da entrada (JOYCODE, GUNCODE, MOUSECODE por exemplo) e o número para atribuir ao dispositivo separados por um sublinhado. A numeração começa a partir do 1, por exemplo, o *token* para o primeiro controle será JOYCODE_1, JOYCODE_2 e assim sucessivamente.

7.9.2 Exemplo de configuração

```
<!-- Veja uma versão deste arquivo aqui https://pastebin.com/fTbEsANJ -->
<mameconfig version="10">
<system name="default">
  <input>
    <mapdevice device="VID_D209&PID_1601" controller="GUNCODE_1" />
    <mapdevice device="VID_D209&PID_1602" controller="GUNCODE_2" />
    <mapdevice device="XInput Player 1" controller="JOYCODE_1" />
    <mapdevice device="XInput Player 2" controller="JOYCODE_2" />
    <port type="P1_JOYSTICK_UP">
      <newseq type="standard">
        JOYCODE_1_YAXIS_UP_SWITCH OR KEYCODE_8PAD
      </newseq>
    </port>
  </input>
</system>
</mameconfig>
```

Acima definimos quatro mapeamentos, **GUNCODE 1/2** e **JOYCODE 1/2**:

- Os dois primeiros mapdevice definem o controle da pistola de luz do jogador 1 e 2 (player 1 e 2) para **Gun 1** e **Gun 2** respectivamente.
- Nós usamos uma cadeia de caracteres com os seus nomes brutos para que cada dispositivo combinem forma individual. Observe que, como este é um arquivo em formato XML, precisamos usar o caractere de escape & para representar o caractere &.
- Os duas últimos elementos mapdevices, definem o o controle do jogador 1 e do jogador 2 para **Joy 1** e **Joy 2** respectivamente. Neste caso, estes são controles do tipo XInput.

7.9.3 Listando os dispositivos disponíveis

Você deve estar se perguntando, como foi que nós obtivemos as IDs dos dispositivos usados no exemplo acima?

Rode o MAME com o parâmetro `-v` para ativar o modo *loquaz*. Assim será exibido no uma lista de dispositivos disponíveis no terminal com a etiqueta device id.

Aqui um exemplo:

```
Input: Adding lightgun #1:
Input: Adding lightgun #2:
Input: Adding lightgun #3: HID-compliant mouse (device id: \\?\HID#VID_045E&PID_0053#7&
↳18297dcb&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Adding lightgun #4: HID-compliant mouse (device id: \\?\HID#IrDeviceV2&Col08#2&
↳2818a073&0&0007#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Adding lightgun #5: HID-compliant mouse (device id: \\?\HID#VID_D209&PID_1602&MI_02#8&
↳389ab7f3&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Adding lightgun #6: HID-compliant mouse (device id: \\?\HID#VID_D209&PID_1601&MI_02#9&
↳375eebb1&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Adding lightgun #7: HID-compliant mouse (device id: \\?\HID#VID_1241&PID_1111#8&
↳198f3adc&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Skipping DirectInput for XInput compatible joystick Controller (XBOX 360 For Windows).
Input: Adding joystick #1: ATRAK Device #1 (device id: ATRAK Device #1)
```

(continua na próxima página)

(continuação da página anterior)

```

Skipping DirectInput for XInput compatible joystick Controller (XBOX 360 For Windows).
Input: Adding joystick #2: ATRAK Device #2 (device id: ATRAK Device #2)
Input: Adding joystick #3: XInput Player 1 (device id: XInput Player 1)
Input: Adding joystick #4: XInput Player 2 (device id: XInput Player 2)

```

Além disso, quando os dispositivos são reatribuídos usando elementos os mapdevice no arquivo de configuração do controle, para ver se as configurações foram aplicadas ou não use o *modo loquaz* e veja as linhas com Remapped indicam que as configurações foram aplicadas com sucesso, exemplo:

```

Input: Remapped lightgun #1: HID-compliant mouse (device id: \\?\HID#VID_D209&PID_1601&MI_02
↳#9&375eebb1&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Remapped lightgun #2: HID-compliant mouse (device id: \\?\HID#VID_D209&PID_1602&MI_02
↳#8&389ab7f3&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
Input: Remapped joystick #1: XInput Player 1 (device id: XInput Player 1)
Input: Remapped joystick #2: XInput Player 2 (device id: XInput Player 2)

```

7.9.4 Limitações

Apenas será possível atribuir números fixos aos dispositivos da entrada caso o MAME receba as IDs fixas e únicas dos dispositivos do provedor do dispositivo e do sistema operacional. Isso nem sempre é o caso. Por exemplo, o provedor de um controle SDL pode não ser capaz de fornecer IDs exclusivas para muitos controles USB.

No caso de nenhum dos dispositivos que já foram configurados estejam conectados quando o MAME for iniciado, os dispositivos que forem conectados podem não estar com a numeração esperada.

7.10 A arma de luz

- *Funcionamento*
- *Configuração da arma no Windows*
- *Configuração da arma no Linux*
 - *Configuração das regras para udev*
- *E caso eu tenha uma arma de outro modelo ou fabricante?*
- *Configurando as entradas no Xorg*
- *Configuração genérica*
- *Configurando uma Pistola de luz*
- *Ativando a arma apenas nos jogos que precisam*
- *Separando apenas as ROMs com jogos de tiro*
- *Compilando uma versão do MAME só com jogos de tiros*

7.10.1 Funcionamento

Em geral a arma possui um fotorreceptor, repare que elas têm uma lente no final do cano, quando o jogador aperta o gatilho o fotorreceptor é acionado, a tela escurece e o alvo acende, tudo isso em milésimos de segundo. Caso o alvo consiga excitar o fotorreceptor no tempo adequado o programa do jogo entende se o jogador acertou ou não o alvo.

Este é o conceito básico, no entanto, cada empresa desenvolve a sua própria tecnologia. Neste [vídeo](#) é possível ver como funciona em detalhes a bazuca do SNES chamada de **Super Scope**, apesar de estar em Inglês, acredito que a ilustração do autor seja de fácil compreensão.

Hoje vivemos num mundo onde as telas de CRT deixaram de ser fabricadas já faz algum tempo, atualmente no entanto, existem armas desenvolvidas para funcionar com telas mais modernas como LCD, Plasma, OLED, etc. Temos por exemplo a arma **Aimtrak** da empresa **Ultimarc** que utiliza um sensor no topo da tela para detectar o movimento da arma.

Outro projeto muito interessante lançado recentemente no [kickstarter](#) é da arma **Sinden Lightgun** da empresa **Sinden Technology** que desenvolveu uma nova tecnologia onde não é mais necessário o uso de um sensor em cima da TV para que o jogo possa identificar o movimento da arma. Esse projeto promete fazer com que a arma de luz deles funcionem em qualquer TV de tela plana e usando desde os consoles de videogames originais, sistemas de fliperama e até mesmo emuladores como o MAME.

Nota: Consulte o capítulo *Opções da mira* para saber como trocar a mira padrão do MAME por uma outra qualquer.

7.10.2 Configuração da arma no Windows

A configuração no Windows é um processo mais simples, a princípio basta conectar o acessório na porta USB e instalar o driver. Depois use o aplicativo de configuração que acompanha o produto e faça os devidos ajustes, testes e calibragem de mira.

Para identificar a sua arma no Windows, conecte-a no computador e baixe o utilitário gratuito chamado [USBDeview](#), rode ele e vá em Exibir > escolher colunas > Deselect all e selecione as opções Nome do dispositivo, descrição, ID do Vendedor, ID do produto, ID da Instância, caso ele esteja em inglês, vá então em View > Choose colums > Deselect all e selecione Device name, description, Vendor ID, Product ID, Instance ID.

O nome da sua arma deve aparecer no campo Nome do dispositivo/Device name, tome nota do **ID do Vendedor (Vendor ID)** e do **ID do Produto (Product ID)**, para o nosso exemplo o dispositivo que usaremos é um mouse gamer com a identificação **VID_1D57&PID_AD04**. Ainda com o dispositivo conectado rode o MAME com o comando `mame -v` e observe a listagem de **Input** (Entradas) que ele gera e tome nota!

Para não deixar a saída muito grande separei apenas o que nos interessa:

```
Input: Adding mouse #2: HID-compliant mouse (device id: \\?\HID#VID_1D57&PID_AD04&MI_01&Col01
↪#7&ecdb012&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
```

Para que fique fácil a compreensão, vamos quebrar essa linha em pedaços listando apenas as partes que nos interessa para a configuração:

- **Input** - O MAME exibe a descrição do que ele identificou como entrada.
- **Adding mouse #2** - Informa o que é este dispositivo e em que ordem ele foi adicionado.
- **HID-compliant mouse** - Informa o nome do dispositivo, muita atenção quanto ao nome que aparece aqui pois o MAME costuma gerar um crash com ACCESS VIOLATION ou ignorar o dispositivo caso ele tenha o nome errado, ainda que o PID e VID estejam certos. As armas Aimtrak podem exibir algo como ATRAK Device #1 product_1601d209-... neste caso, o que importa para nós é o ATRAK Device #1.

Para remapear este dispositivo como lightgun #0, nós usamos o exemplo da configuração abaixo:

```
<mameconfig version="10">
<system name="default">
  <input>
    <!--Atenção ao nome correto do dispositivo, esta parte configura a parte de_
↪controle da arma-->
    <mapdevice device="HID-compliant mouse" controller="JOYCODE_1" />
    <!--Aqui definimos o VID e PID da arma-->
    <mapdevice device="VID_1D57&PID_AD04" controller="GUNCODE_1" />
    <!--Para 2 armas ou mais, basta usar JOYCODE_2/GUNCODE2 para o jogador 2 e_
↪assim por diante-->
  </input>
</system>
</mameconfig>
```

Salve a configuração como `arma.cfg` dentro do diretório `ctrl`, caso o MAME esteja aberto, feche. Inicie-o novamente com o comando `mame -v -ctrlr arma`, você deverá ter na saída algo deste tipo:

```
Attempting to parse: arma.cfg
Input: Remapped lightgun #0: HID-compliant mouse (device id: \\?\HID#VID_1D57&PID_AD04&MI_01&
↪Col01#7&ecdb012&0&0000#{378de44c-56ef-11d1-bc8c-00a0c91405dd})
```

7.10.3 Configuração da arma no Linux

No Linux o processo é mais complicado e exige um pouco mais de trabalho na parte de configuração porém não desanime, é mais fácil fazer do que descrever todo o processo. Existem diferentes meios de se alcançar este objetivo, dentre os mais conhecidos fazem com que o MAME veja essa arma de luz como um mouse, o que faz com que a experiência final do usuário não seja das melhores. Não há qualquer alinhamento prévio entre a interface do mouse com a mira externa, isso exige que uma configuração individual seja feita para cada jogo e ainda assim não é a mais precisa.

Existe no entanto um outro método fazendo a configuração através do udev e Xorg.conf que permite um acesso direto ao acessório e com isso obter uma melhora significativa na questão da precisão da mira.

A base de referência usada aqui é o Debian e Ubuntu, talvez alguns ajustes na configuração sejam necessárias para outros sistemas Linux, no entanto este apanhado geral serve como um guia do que precisa ser feito.

Configuração das regras para udev

A arma AimTrak assim como as de outras marcas, quando conectadas na porta USB, exibem geralmente 2 mouses e 1 joystick. O que faremos será fazer uma configuração estática usando o udev em conjunto com o **libinput**, fazendo com que todo o resto seja ignorado, menos os dispositivos que precisamos. Isso evita conflitos do sistema que identifica mais de um mouse para cada arma USB que for conectada.

Crie um novo arquivo chamado **99-aimtrak.rules** em /etc/udev/rules.d com o comando `sudo touch /etc/udev/rules.d/99-aimtrak.rules`, usamos um valor numérico alto pois este arquivo não tem prioridade alguma, assim deixamos que ele seja carregado por último, evitando possíveis conflitos. Usando o seu editor preferido, cole a configuração abaixo:

```
# Veja uma cópia deste arquivo no link abaixo:
# https://pastebin.com/HQvML0Dg
#
# Define o modo "0666" e desabilita a assistência do libinput evitando
# que o X11 use as interfaces ou dispositivos errados.
SUBSYSTEMS=="usb", ATTRS{idVendor}=="d209", ATTRS{idProduct}=="160*", MODE="0666", ENV{ID_
↳ INPUT}="", ENV{LIBINPUT_IGNORE_DEVICE}="1"

# Caso ID_USB_INTERFACE_NUM==2, habilite a assistência do libinput.
SUBSYSTEMS=="usb", ATTRS{idVendor}=="d209", ATTRS{idProduct}=="160*", ENV{ID_USB_INTERFACE_
↳ NUM}=="02", ENV{ID_INPUT}="1", ENV{LIBINPUT_IGNORE_DEVICE}="0"
```

NÃO PULE ESTA ETAPA!

Salve, saia e recarregue a configuração com o comando `udevadm control --reload-rules`. Antes de prosseguir faça o comando `sudo journalctl -b -p err` e verifique se não há o retorno de qualquer erro em vermelho relacionado com essa regra que você acabou de adicionar, caso contrário você vai perder o acesso ao computador e só será possível recuperá-lo reiniciando e entrando no modo de recuperação do sistema para apagar ou arrumar a regra.

A configuração acima é voltada especificamente para as armas **AimTrak**, porém cada modelo de arma precisará de uma configuração específica. Atenção a formatação deste arquivo, há distribuições que ignoram a quebra de linha, porém distribuições como o Debian interpretam a quebra de linha como um erro fazendo com que você perca o controle do teclado e do mouse quando o computador é reiniciado, use o link acima para obter uma cópia deste arquivo.

7.10.4 E caso eu tenha uma arma de outro modelo ou fabricante?

Toda a vez que um dispositivo USB é conectado no Linux ele faz um registro dessa conexão, para acessar esse registro faça o comando `sudo dmesg` no terminal logo depois de conectar a sua arma USB. Aqui um exemplo do que aparece no terminal logo depois que um mouse gamer USB é conectado:

```
[12119.580375] usb 2-1.3: new full-speed USB device number 3 using xhci_hcd
[12119.688300] usb 2-1.3: New USB device found, idVendor=1d57, idProduct=ad04
[12119.688303] usb 2-1.3: New USB device strings: Mfr=2, Product=1, SerialNumber=0
[12119.688305] usb 2-1.3: Product: Gaming Mouse
[12119.688306] usb 2-1.3: Manufacturer: LXD
[12119.694168] input: LXD Gaming Mouse as /devices/pci0000:00/0000:00:09.0/0000:02:00.0/usb2/
↳ 2-1/2-1.3/2-1.3:1.0/0003:1D57:AD04.0006/input/input17
[12119.753002] hid-generic 0003:1D57:AD04.0006: input,hidraw5: USB HID v1.10 Keyboard [LXD_
↳ Gaming Mouse] on usb-0000:02:00.0-1.3/input0
[12119.759341] input: LXD Gaming Mouse as /devices/pci0000:00/0000:00:09.0/0000:02:00.0/usb2/
↳ 2-1/2-1.3/2-1.3:1.1/0003:1D57:AD04.0007/input/input18
[12119.816761] hid-generic 0003:1D57:AD04.0007: input,hidraw6: USB HID v1.10 Mouse [LXD_
↳ Gaming Mouse] on usb-0000:02:00.0-1.3/input1
```

O exemplo mostra duas interfaces **input** assim como é com a arma **AimTrak**, basta agora substituir os valores de **idVendor** e **idProduct** para bater com o nosso dispositivo, assim a configuração ficaria assim:

```
# Veja uma cópia deste arquivo no link abaixo:
# https://pastebin.com/gw0VszkK
#
# Define o modo "0666" e desabilita a assistência do libinput evitando
# que o X11 use as interfaces ou dispositivos errados.
SUBSYSTEMS=="usb", ATTRS{idVendor}=="1d57", ATTRS{idProduct}=="ad04", MODE="0666", ENV{ID_
↳ INPUT}="", ENV{LIBINPUT_IGNORE_DEVICE}="1"

# Caso ID_USB_INTERFACE_NUM==2, habilite a assistência do libinput.
SUBSYSTEMS=="usb", ATTRS{idVendor}=="1d57", ATTRS{idProduct}=="ad04", ENV{ID_USB_INTERFACE_
↳ NUM}=="02", ENV{ID_INPUT}="1", ENV{LIBINPUT_IGNORE_DEVICE}="0"
```

NÃO PULE ESTA ETAPA!

Salve, saia e recarregue a configuração com o comando `udevadm control --reload-rules`. Antes de prosseguir faça o comando `sudo journalctl -b -p err` e verifique se não há o retorno de qualquer erro em vermelho relacionado com essa regra que você acabou de adicionar, caso contrário você vai perder o acesso ao computador e só será possível recuperá-lo reiniciando e entrando no modo de recuperação do sistema para apagar ou arrumar a regra.

7.10.5 Configurando as entradas no Xorg

Vale lembrar que algumas distribuições Linux migraram para o Wayland, apesar da migração o Wayland ainda compartilha configurações muito semelhantes ao Xorg/X11, no entanto são poucas as distribuições que ainda usam o arquivo de configuração **xorg.conf** assim como, o diretório de configuração pode estar localizado num outro lugar qualquer, assim a sua sorte pode variar bastante.

Para que mais de uma arma funcione de forma correta, é necessário configurar o Xorg para tratá-la(s) como dispositivos "*flutuantes*", fazendo com que a mira de cada arma não seja confundida com o ponteiro do mouse usado pelo sistema.

No **Ubuntu** e **Fedora** crie o arquivo **99-arma.conf** no diretório `/etc/X11/xorg.conf.d`, no **Debian** e no **Arch Linux** o diretório fica em `/usr/share/X11/xorg.conf.d`. Devido a grande variedade de distribuições Linux é inviável tentar descrever o caminho completo do diretório **xorg.conf.d** para cada uma delas, isso sem contar macOS e as várias variantes de BSD's, no entanto, é possível usar o comando abaixo para tentar localizá-lo caso a sua distribuição utilize um diretório de mesmo nome para armazenar essas configurações porém em um local diferente do predefinido:

```
sudo find /usr -name xorg.conf.d
```

Caso o comando acima não retorne nada, verifique o diretório correto para a distribuição que você estiver usando.

Dependendo da quantidade de dispositivos USB que você tenha conectado no seu computador eles ocuparão diferentes input/event, ainda usando o nosso *mouse gamer* como exemplo, você pode fazer o comando `libinput list-devices` no **Ubuntu** e **Fedora** ou `libinput list-devices` no **Debian**. Caso o comando não funcione, tenha certeza de ter instalado o pacote **libinput-tools**.

Para mais informações acesse este [link](#).

O comando deve listar todos os dispositivos, aqui limitado apenas para o nosso caso:

```
Device:      LXD Gaming Mouse
Kernel:      /dev/input/event14
Group:       3
...
Device:      LXD Gaming Mouse
Kernel:      /dev/input/event15
Group:       3
```

A saída completa foi eliminada para exibir apenas o que nos interessa, caso a sua distribuição não tenha o **libinput-tools** por algum motivo, podemos usar o bom e velho comando `cat /proc/bus/input/devices`:

```
I: Bus=0003 Vendor=1d57 Product=ad04 Version=0110
N: Name="LXD Gaming Mouse"
P: Phys=usb-0000:02:00.0-1.3/input0
U: Uniq=
H: Handlers=sysrq kbd leds event14

I: Bus=0003 Vendor=1d57 Product=ad04 Version=0110
N: Name="LXD Gaming Mouse"
P: Phys=usb-0000:02:00.0-1.3/input1
U: Uniq=
H: Handlers=kbd mouse2 event15
```


Veja que o comando também mostra o Vendor e Product ID's, com essa informação em mãos criamos o seguinte conteúdo para o nosso arquivo [99-arma.conf](#), novamente, usamos **99** para que este seja o último arquivo a ser lido pelo sistema:

```
Section "InputClass"
    Identifier "LXD Gaming Mouse"
    MatchDevicePath "/dev/input/event*"
    MatchUSBID "1d57:ad04"
    Driver "libinput"
    Option "Floating" "yes"
    Option "AccelerationProfile" "-1"
    Option "AutoServerLayout" "no"
EndSection
```

Um cuidado especial com a opção **Floating**, pode ser que dependendo do seu dispositivo, deixar em **yes** pode fazer com que a sua arma ou mouse fique limitado a um pequeno espaço na tela, caso seja o seu caso, mude essa opção para **no**, salve o arquivo e encerre a sessão (retorne para a tela de login). Isso precisa ser feito pois o arquivo só é lido novamente quando a sessão é encerrada ou o computador é reiniciado.

O **AccelerationProfile** serve para lidar com a aceleração ou não do dispositivo, pode ser que no seu ambiente a mira esteja lenta demais, arrastada ou rápida demais, etc. Ajuste conforme a sua necessidade, a ideia é fazer com que a mira responda de forma rápida e precisa conforme os seus movimento. Os valores válidos segundo a [documentação oficial](#) são:

- **-1** Nenhuma aceleração.
- **1** Com aceleração caso o dispositivo suporte.
- **2** Escala Polinomial, a velocidade serve como um coeficiente e a aceleração um expoente. Em resumo, tente este primeiro.
- **3** Linear suave, escala linear na maioria do tempo com um início suave.
- **4** Simples, transição suave entre aceleração e sem, este é o valor predefinido caso nada seja definido.
- **5** Power, aceleração acentuada, difícil de controlar.
- **6** Linear, velocidade e aceleração linear.
- **7** Limitado, ascende a aceleração de forma suave até um limite.

7.10.6 Configuração genérica

Existem diferentes maneiras de fazer este tipo de configuração no MAME, a primeira seria editando o seu `~/.mame/mame.ini` com as configurações abaixo para **Windows**:

```
lightgun          1
lightgun_device   lightgun
offscreen_reload  1
```

Adicione as opções acima no seu `mame.ini` e pronto.

Aqui a configuração para **Linux** e variantes **SDL**:

```
lightgun          1
lightgun_device   mouse
lightgunprovider  x11
lightgun_index1   "Continue lendo para saber o que usar aqui"
offscreen_reload  1
```

Lembrando que estamos usando um mouse como teste, assim estamos usando `lightgun_device` como **mouse**, caso você esteja usando uma arma de luz mude para **lightgun**.

Na versão SDL precisamos definir `lightgun_index[1-8]`, geralmente o valor que precisamos usar é o **nome do dispositivo** ou o seu **ID**. É usando o `lightgun_index` entre 1 e 8 que você vai adicionando todas as armas que você tiver no sistema, cada uma com o seu ID único. Com a arma ou o mouse conectado, inicie o MAME com o comando `mame -v`, o MAME deve exibir uma mensagem como essa (ela vai variar muito de caso para caso):

```
Evaluating device with name: Virtual core pointer
Evaluating device with name: Virtual core keyboard
Evaluating device with name: Virtual core XTEST pointer
Evaluating device with name: Virtual core XTEST keyboard
Evaluating device with name: Power Button
Evaluating device with name: Power Button
Evaluating device with name: Logitech USB Optical Mouse
Evaluating device with name: Microsoft Microsoft® 2.4GHz Transceiver v8.0
Evaluating device with name: Microsoft Microsoft® 2.4GHz Transceiver v8.0
Evaluating device with name: Microsoft Microsoft® 2.4GHz Transceiver v8.0
Evaluating device with name: Microsoft Microsoft® 2.4GHz Transceiver v8.0
Evaluating device with name: Microsoft Microsoft® 2.4GHz Transceiver v8.0
Evaluating device with name: LXD Gaming Mouse
Evaluating device with name: LXD Gaming Mouse
Evaluating device with name: LXD Gaming Mouse
```

No nosso exemplo o **LXD Gaming Mouse** repete 3x e ao usá-lo com o `lightgun_index1`:

```
lightgun_index1   LXD Gaming Mouse
```

O MAME reclama dizendo:

```
Warning: There are multiple devices named "LXDGamingMouse".
To ensure the correct one is selected, please use the device ID
instead.
```

Traduzindo a mensagem fica assim:

Atenção: Existe mais de um dispositivo com o nome "LXDGamingMouse".
Favor usar o ID do dispositivo para ter certeza que apenas um seja escolhido.

Para encontrar o ID do dispositivo precisamos do programa **xinput**, verifique no gerenciador de pacotes da sua distribuição como fazer para instalá-lo, no **Debian** e **Ubuntu** seria `sudo apt-get install xinput`.

Execute o comando `xinput list`:

Virtual core pointer	id=2	[master pointer (3)]
Virtual core XTEST pointer	id=4	[slave pointer (2)]
Logitech USB Optical Mouse	id=8	[slave pointer (2)]
Microsoft Microsoft® 2.4GHz Transceiver v8.0	id=10	[slave pointer (2)]
Microsoft Microsoft® 2.4GHz Transceiver v8.0	id=11	[slave pointer (2)]
Virtual core keyboard	id=3	[master keyboard (2)]
Virtual core XTEST keyboard	id=5	[slave keyboard (3)]
Power Button	id=6	[slave keyboard (3)]
Power Button	id=7	[slave keyboard (3)]
Microsoft Microsoft® 2.4GHz Transceiver v8.0	id=9	[slave keyboard (3)]
Microsoft Microsoft® 2.4GHz Transceiver v8.0	id=12	[slave keyboard (3)]
Microsoft Microsoft® 2.4GHz Transceiver v8.0	id=13	[slave keyboard (3)]
LXD Gaming Mouse	id=14	[floating slave]
LXD Gaming Mouse	id=15	[floating slave]
LXD Gaming Mouse	id=16	[floating slave]

O comando exibe a **id=14**, **id=15** e **id=16** para o **LXD Gaming Mouse**, nos testes o id que funciona com o nosso dispositivo é o **id=15**, logo a configuração final fica assim:

```
lightgun          1
lightgun_device   mouse
lightgunprovider  x11
lightgun_index1   15
offscreen_reload  1
```

Salve o seu `mame.ini` com as opções acima e inicie o MAME com o comando `mame -v`, na saída agora temos:

```
Lightgun: Begin initialization
Lightgun mapping: Logical id 1: 15
Input: Adding lightgun #0: 15 (device id: 15)
0: 15
...
...
Motion = 71
Device 15: Registered 3 events.
Events types to register: motion:71, press:69, release:70
Lightgun: End initialization
```

Escolha um jogo de tiro qualquer e verá que a sua arma ou mouse deve funcionar sem qualquer problema.

7.10.7 Configurando uma Pistola de luz

Agora que tudo está funcionando a parte mais chata seria fazer configuração da sua arma para cada uma das trezentas e poucos sistemas, porém isso é mais simples do que parece. O MAME oferece a opção `-ctrlr` para que você possa carregar a configuração que você já fez para um sistema mas que podem ser utilizados em outros.

Inicie uma sistema qualquer como **bang** por exemplo, `mame bang`, quando ela iniciar pressione **TAB** para acessar a interface e vá em *Atribuições da entrada (este sistema)*. Para o **Jogador 1** selecione *Lightgun X Analog* e pressione **Enter**, mova a arma da esquerda para direita, deve aparecer *Gun 1 X*, faça o mesmo com *Lightgun X Analog* mas mova a arma de cima para baixo, agora a opção deve aparecer como *Gun 1 X*. Caso tenha mais uma arma para o jogador 2 faça o mesmo em *Lightgun X 2 Analog* e *Lightgun Y 2 Analog*.

Pressione Esc para sair do MAME, vá até o diretório **cfg** e localize o arquivo **bang.cfg**, nele está toda a configuração que você fez, exemplo:

```
<?xml version="1.0"?>
<!-- This file is autogenerated; comments and unknown tags will be stripped -->
<mameconfig version="10">
  <system name="bang">
    <counters>
      <coins index="0" number="10" />
    </counters>
    <input>
      <port tag=":LIGHT0_X" type="P1_LIGHTGUN_X" mask="255" defvalue="128">
        <newseq type="standard">
          GUNCODE_1_XAXIS
        </newseq>
      </port>
      <port tag=":LIGHT0_Y" type="P1_LIGHTGUN_Y" mask="255" defvalue="128">
        <newseq type="standard">
          GUNCODE_1_YAXIS
        </newseq>
      </port>
    </input>
  </system>
</mameconfig>
```

O exemplo acima foi gerado no Linux, no Windows e outros sistemas será gerado o mesmo arquivo mas com uma [configuração diferente](#), aqui o exemplo para o Aimtrak no Windows:

```
<?xml version="1.0"?>
<!-- This file is autogenerated; comments and unknown tags will be stripped -->
<mameconfig version="10">
  <system name="bang">
    <counters>
      <coins index="0" number="10" />
    </counters>
    <input>
      <mapdevice device="ATRAK Device #1 product_XXXXXXX-0000-0000-0000-XXXXXXXXXXXX"
→instance_XXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX" controller="GUNCODE_1" />
      <mapdevice device="ATRAK Device #2 product_YYYYYYY-0000-0000-0000-YYYYYYYYYYYY"
→instance_YYYYYYY-YYYY-YYYY-YYYY-YYYYYYYYYYYY" controller="GUNCODE_2" />
      <port type="P1_LIGHTGUN_X">
        <newseq type="standard">
          GUNCODE_1_XAXIS
        </newseq>
      </port>
      <port type="P1_LIGHTGUN_Y">
        <newseq type="standard">
          GUNCODE_1_YAXIS
        </newseq>
      </port>
      <port type="P2_LIGHTGUN_X">
        <newseq type="standard">
          GUNCODE_2_XAXIS
        </newseq>
      </port>
      <port type="P2_LIGHTGUN_Y">
        <newseq type="standard">
          GUNCODE_2_YAXIS
        </newseq>
      </port>
    </input>
  </system>
</mameconfig>
```

Independente do arquivo que você tenha gerado edite a linha `<system name="bang">` para `<system name="default">` e salve o arquivo como **arma.cfg** dentro do diretório **ctrl**. Agora sempre que você for iniciar o MAME com essa configuração, basta fazer o comando `mame -ctrlr arma bang`. Assim o MAME inicia o sistema com as suas configurações predefinidas.

Caso não queira fazer isso para cada jogo, adicione a configuração no seu **mame.ini**:

ctrlr	arma
-------	------

Lembrando que é possível também fazer como foi ensinado em [Ativando a arma apenas em jogos que precisam](#) adicionando esta opção em **cfg.txt**.

7.10.8 Ativando a arma apenas nos jogos que precisam

O problema de usar o `mame.ini` é que o MAME **sempre** vai carregar e ativar a arma em máquinas que não precisam, num PC com bastante recursos pode não ser problema, no entanto, caso o MAME esteja rodando num ambiente com recursos limitados isso pode ser um problema. Ou simplesmente, é como o autor deste texto que gosta das coisas bem organizadas.

O que faremos é replicar a configuração que temos e sabemos que funciona apenas para os sistemas que usam arma, deixando o `mame.ini` livre de modificações. Para realizar essa façanha *é bem simples*, basta criarmos um arquivo ***.ini para cada uma dos 362 sistemas conhecidos** e salvar a configuração acima **EM CADA UM DESTES ARQUIVOS**. Ainda bem que temos as ferramentas certas no **Linux** para nos ajudar, certo?

Todos os procedimentos abaixo são feitos num ambiente **Linux** mas podem funcionar num ambiente **MINGW** ou similares.

- Abra o seu arquivo `~/mame/mame.ini`, em **inipath** substitua o `$HOME/.mame;. ;ini` por `$HOME/.mame;. ;ini;arma`
- Salve e saia.
- O site do projeto **Project-Snaps** mantém um arquivo chamado **category.ini** com uma lista de jogos separados por diversas categorias diferentes, dentre elas há a categoria de jogos de tiro que usam armas definido na lista como **[Shooting / Guns]**, usaremos os nomes desta lista para preparar a nossa.
- Acesse [este link](#) do site Project-Snaps e baixe o arquivo **category.ini** mais recente.
- Abra o arquivo compactado e extraia o diretório **folders** no diretório raiz do MAME.
- **No terminal**, vá até o diretório raiz do MAME e faça o comando `mkdir arma` para criar o diretório seguido de `cd arma` para entrar nele.
- Execute o comando abaixo para filtrar apenas os nomes dos sistemas que queremos e em seguida salvamos eles num arquivo chamado **maquinas** em formato de fim de linha para Unix:

```
awk '/Gun/{flag=1; next} / {flag=0} flag' ../folders/category.ini | head -n -6 >_
↪maquinas && sed -i 's/\r//g' maquinas
```

- Caso o seu `cfg.txt` esteja em formato Unix, ele precisa ser convertido antes de ser usado no Windows com o comando `sed -i 's/$\r/' cfg.txt`.
- Copie e cole a configuração abaixo num arquivo texto e salve **dentro do diretório arma** localizado no diretório raiz do MAME como `cfg.txt`, no exemplo estou usando **mouse** como **lightgun_device**, caso esteja usando uma arma substitua por **lightgun**:

```
lightgun          1
lightgun_device   mouse
lightgunprovider  x11
lightgun_index1   15
offscreen_reload  1
```

- No terminal, ainda dentro do diretório `arma`, execute o comando abaixo para criar uma configuração com o nome de cada sistema:

```
while read lista; do cp cfg.txt "$lista".ini; done < maquinas
```

Agora dentro do diretório arma estará cheia de arquivos *.ini como o nome de cada sistema que usa uma arma e com a configuração correta dentro de cada um deles.

Estou disponibilizando esses arquivos *.ini já prontos visando facilitar a vida de todos, a versão para Windows é bem genérica e deve funcionar de imediato sem muitos ajustes, porém o mesmo não ocorre com a versão Linux, a configuração precisa ser customizada individualmente para cada caso, principalmente o **lightgun_index**, caso o nome ou o ID esteja errado a sua arma não vai funcionar, de qualquer maneira aqui estão os arquivos.

- Arquivos ini com **lightgun_device** como mouse. <https://www.mediafire.com/file/2vh06q6lbvcur8a/ini-mouse.7z>
- Arquivos ini com **lightgun_device** como lightgun. <https://www.mediafire.com/file/ytmnp3ik9avyfjm/ini-lightgun.7z>
- Arquivos ini com **lightgun_device** como mouse para Windows. <http://www.mediafire.com/file/1zz6vfk7jh7tj8/ini-mouse-windows.7z>
- Arquivos ini com **lightgun_device** como lightgun para Windows. <http://www.mediafire.com/file/hi7864yk8s09o78/ini-lightgun-windows.7z>

Use o 7-zip para descompactar os arquivos dentro do diretório arma.

7.10.9 Separando apenas as ROMs com jogos de tiro

Da mesma maneira que podemos criar uma lista de configuração individual para cada sistema, podemos também usar a mesma lista para copiar apenas as suas ROMs atendendo a necessidade das pessoas que configuram os seus sistemas dessa forma.

Ainda usando o arquivo *maquinas* executaremos as seguintes ações:

- Crie um diretório **roms** em qualquer outro lugar fora do diretório onde se encontra o MAME.
- Copie o arquivo **maquinas** (ou gere um novo caso tenha apagado) para dentro deste diretório.
- Você precisa encontrar o caminho completo onde todas as suas ROMs se encontram, vamos supor que seja `/home/mame/mame/roms`, abra um terminal neste diretório e execute o comando abaixo:

```
while read maquinas; do echo /home/mame/mame/roms/"$maquinas".zip ; done < maquinas >_
↪ lista-roms
```

- O comando acima vai ser alimentado pelo arquivo **maquinas** e substituir "**\$maquinas**" pelos nomes que forem aparecendo linha a linha, depois `> lista-roms` faz o redirecionamento completo para o arquivo **lista-roms**. Ao final o arquivo ficará com o seguinte conteúdo:

```
/home/mame/mame/roms/2spicy.zip
/home/mame/mame/roms/alien3.zip
/home/mame/mame/roms/alien3j.zip
/home/mame/mame/roms/alien3u.zip
/home/mame/mame/roms/aplatoon.zip
/home/mame/mame/roms/area51.zip
```

- Agora com a lista das ROMs e seu caminho completo basta copiá-los com o comando abaixo:

```
while read copy ; do cp "$copy" . ; done < lista-roms
```

O ponto depois de "**\$copy**" faz com que o comando `cp` copie todos os arquivos para o diretório onde você está, caso queira copiá-los para outro lugar basta usar o caminho, assim:

```
while read copy; do cp "$copy" /caminho/completo ; done < lista-roms
```

Apesar do comando **cp** funcionar bem para a maioria dos casos, é impossível saber se o arquivo foi copiado de forma correta ou não para o destino, nestes casos a melhor opção é usar o programa **rsync** que durante o processo de cópia verifica a integridade do arquivo no destino, além de ser a melhor opção para a cópia de arquivos nós podemos também registrar num arquivo toda a operação que ele fez, seja bem sucedida ou não, assim basta usar o comando anterior com algumas alterações:

```
while read copy; do rsync --info=name,progress2 --log-file=registro "$copy" . ; done < lista-
↪ roms
```

Neste novo comando a opção `--info=name,progress2` vai exibir estatísticas da operação que ele estiver fazendo de um determinado arquivo, o `log-file=registro` armazena todo o processo, seja ele bem sucedido ou não assim como erros informando as ROMs que não foram encontradas. É possível filtrar essas ROMs que não foram encontradas com o comando:

```
cat registro | grep "No such file or directory" | awk '{print $6}' > roms-ausentes
```

O exemplo que foi demonstrado aqui serve para qualquer outro tipo de lista, você pode por exemplo gerar uma lista para os sistemas dentro dos sistemas CPS1/CPS2/ZN e depois copiar essas ROMs em diretórios separados, o céu é o limite.

7.10.10 Compilando uma versão do MAME só com jogos de tiros

O MAME disponibiliza a opção de filtrar a lista dos sistemas por categoria, para mais informações veja [Categoria](#), os jogos de tiro estão listados como [**Shooting / Guns**]. No entanto está se tornando muito comum o uso de miniPC's como Raspberry Pi, é muito comum também ver pessoas perguntando como compilar uma versão customizada do MAME só com jogos de tiro em fóruns e comunidades espalhadas pela internet. Os motivos são diversos, o mais comum sendo a limitação de espaço que impossibilita ter um binário completo do MAME.

Aqui iremos demonstrar como isso pode ser feito, recomendamos que antes de prosseguir leia o capítulo [Compilando o MAME](#) para obter maiores informações e detalhes que não serão abordados aqui. O autor assume que o você já tenha lido e compreendido o capítulo sobre a compilação do MAME e que você já esteja familiarizado com o processo.

No [capítulo anterior](#) nós demonstramos como criar o arquivo **maquinas** usando o arquivo **category.ini** que fica dentro do diretório **folders**, naquele arquivo ficam todos os sistemas dentro da categoria de tiro, porém para compilar o MAME com elas nós necessitamos encontrar **TODOS** os sistemas responsáveis por eles e repassar essa informação aos scripts de compilação usando a opção **SOURCES**.

- Precisamos do arquivo **maquinas** com a listagem de todas elas, lembramos que o MAME está sempre em evolução, logo a lista disponível [aqui](#) pode mudar com o tempo, assim recomendamos manter o seu arquivo **categories.ini** atualizado e se for o caso, gere um novo arquivo seguindo as instruções do capítulo anterior.
- Para encontrar os drivers responsáveis pelos sistemas da lista nós usamos a função **-listsource / -ls** do MAME, por exemplo:

```
./mame -ls area51 | awk '{print $2}'
atari/jaguar.cpp
```

- Copie o arquivo **maquinas** (gerado ou [baixado](#)) dentro do diretório do MAME e execute o comando abaixo:

```
while read lista; do ~/mame/mame -ls "$lista"; done < maquinas | awk '{print $2}' | awk
↵ '!seen[$0]++' | sort -d > drivers
```

O comando vai alimentar o MAME com o nome dos sistemas, ~/mame/mame mostra o caminho completo onde se encontra o binário do MAME, o comando `awk '{print $2}'` vai selecionar apenas a segunda coluna onde estão os **drivers.cpp**, o comando `awk '!seen[$0]++'` elimina todos os nomes duplicados, já o último comando dessa cadeia, `sort -d > drivers` organiza a lista em ordem alfabética e redireciona a sua saída para um arquivo chamado **drivers**.

- Copie o arquivo **drivers** para dentro do diretório raiz onde se encontra o código-fonte do MAME.
- Apesar da lista ter sido gerada, ela ainda não é útil para nós pois precisamos que ela esteja disposta numa só linha e separada por vírgula, para isso executamos o comando abaixo:

```
cat list-drivers | sed ':a;N;$!ba;s/\n/,/g' > compile-drivers
```

Aqui o comando `cat list-drivers` lista todo o conteúdo de **list-drivers**, já `sed ':a;N;$!ba;s/\n/,/g'` > `compile-drivers` vai quebrar o final da linha depois do último caractere, o substituirá por vírgula e redirecionará a sua saída para o arquivo **compile-drivers**, exemplo:

```
amiga/alg.cpp,atari/atarittl.cpp,atari/cops.cpp,etc
```

- Com a nossa [lista completa](#), basta agora executar o comando de compilação do MAME:

```

make SYMBOLS=1 SYMLEVEL=1 PTR64=1 SSE2=1 OPTIMIZE=3 SOURCES=amiga/alg.cpp,atari/
↪atarittl.cpp,atari/cops.cpp,atari/jaguar.cpp,atari/mediagx.cpp,atari/triplhnt.cpp,
↪dataeast/deco32.cpp,edevices/fantland.cpp,exidy/exidy440.cpp,gaelco/gaelco2.cpp,
↪gaelco/targeth.cpp,igs/lordgun.cpp,itech/iteagle.cpp,itech/itech32.cpp,jaleco/
↪cischeat.cpp,konami/gticlub.cpp,konami/hornet.cpp,konami/konamigq.cpp,konami/konamigv.
↪cpp,konami/konamigx.cpp,konami/konamim2.cpp,konami/ksys573.cpp,konami/lethal.cpp,
↪konami/viper.cpp,midw8080/8080bw.cpp,midw8080/mw8080bw.cpp,midway/midxunit.cpp,midway/
↪midyunit.cpp,midway/midzeus.cpp,midway/seattle.cpp,midway/williams.cpp,misc/3do.cpp,
↪misc/lethalj.cpp,misc/oneshot.cpp,misc/policetr.cpp,misc/psattack.cpp,misc/pse.cpp,
↪misc/skeetsht.cpp,misc/tickee.cpp,misc/vcombat.cpp,misc/voyager.cpp,misc/vp101.cpp,mr/
↪sshot.cpp,namco/cswat.cpp,namco/namconb1.cpp,namco/namcops2.cpp,namco/namcos10.cpp,
↪namco/namcos11.cpp,namco/namcos12.cpp,namco/namcos22.cpp,namco/namcos23.cpp,namco/
↪namcos2.cpp,namco/shootaway2.cpp,nintendo/dkong.cpp,nintendo/playch10.cpp,nintendo/
↪vsnes.cpp,pc/calchase.cpp,sega/chihiro.cpp,sega/dc_atomiswave.cpp,sega/hikaru.cpp,
↪sega/lindbergh.cpp,sega/model2.cpp,sega/model3.cpp,sega/naomi.cpp,sega/segag80r.cpp,
↪sega/segas18.cpp,sega/segas32.cpp,sega/segaxbd.cpp,sega/segaybd.cpp,sega/system1.cpp,
↪seta/seta2.cpp,sega/seta.cpp,sega/ssv.cpp,snk/bbusters.cpp,snk/hng64.cpp,snk/mechatt.
↪cpp,sony/zn.cpp,sterm/mazerbla.cpp,taito/gunbustr.cpp,taito/nycaptor.cpp,taito/opwolf.
↪cpp,taito/othunder.cpp,taito/slapshot.cpp,taito/taitopjc.cpp,taito/taito_z.cpp,taito/
↪undrfire.cpp,unico/unianapc.cpp,unico/unico.cpp -j5

```

No final da compilação você terá um executável personalizado do MAME com um tamanho reduzido e que vai incluir os jogos de tiro, assim como, todos os outros jogos compatíveis com estes drivers. Ainda que agora a gente tenha um binário do MAME apenas com os drivers que tenham jogos de tiros, os drivers também possuem outros tipos de jogos, para exibir apenas os jogos de tiro, use o filtro *Categoria*.

7.11 MAME em câmera lenta

O **Slow Motion Mame** ou **slowmomame** é uma condição em que o tempo de emulação do MAME seja atrasado de forma que este fique em câmera lenta, fazendo com que a emulação passe a trabalhar quase que quadro a quadro. A utilidade do **slowmomame** é voltada para exibição de jogadas mais arriscadas, tomadas de decisão rápidas uma vez que a emulação está em câmera lenta o usuário tem tempo de sobra para reagir a tiros, golpes e sistemas no estilo BEMANI da Konami.

Neste modo e dependendo do sistema, o jogador leva em média cerca de 1 hora para concluir uma única fase, no entanto, o resultado final de todo este esforço é [uma jogabilidade perfeita](#).

Abordaremos agora como atrasar o tempo de emulação do MAME para que essa condição seja atingida, uma delas é editando o código-fonte, a segunda maneira é utilizando uma interface lenta e a última é através de configuração usando a opção `-speed`.

7.11.1 Alterando o código-fonte

Antes de prosseguir é importante que saiba como [compilar o MAME](#), com um editor de texto abra o arquivo `src/emu/sound.cpp` e procure pela linha:

```
#define VERBOSE          (0)
```

Altere o valor de **0** para **1** e adicione a opção `DEBUG=1` às suas opções de compilação ou ao seu arquivo `useroptions.mak` e inicie a compilação. Geralmente não há a necessidade de se compilar todo o código-fonte do MAME com essas alterações pois como já sabemos o MAME abrange muito mais do que sistemas arcade e consoles. Prefira usar a opção `SUBTARGET=arcade` para compilar apenas os sistemas arcades ou então defina apenas o driver do sistema que deseja usar com o comando `SOURCES=caminho do driver`.

Por exemplo, caso queira saber qual a ROM responsável pelo sistema **DDR Max - Dance Dance Revolution 6th Mix** use o comando:

```
mame "DDR Max"
ddrmax          DDR Max - Dance Dance Revolution 6th Mix (G*B19 VER. JAA)    (Konami, 2001)
```

Com o nome da ROM em mãos, **ddrmax**, podemos descobrir qual o driver responsável por ele usando o comando `-listsource / -ls`, exemplo:

```
mame -ls ddrmax
ddrmax          konami/ksys573.cpp
```

Com o nome do driver em mãos, use-o com a opção `SOURCES`, como por exemplo, `SOURCES=konami/ksys573.cpp` ou `SOURCES=konami\ksys573.cpp` caso esteja usando o Windows.

Abaixo uma sugestão de linha de comando para a compilação do MAME:

```
make SUBTARGET=arcade DEBUG=1 -j5
```

Ou usando **SOURCES**:

```
make SOURCES=konami/ksys573.cpp DEBUG=1 -j5
```

Ao final da compilação renomeie o executável e se for o caso, o respectivo arquivo `*.syn`, para um outro nome qualquer como **slowmo** por exemplo. Para rodar um sistema enquanto grava, use o comando:

```
slowmo ddrmax -aviwrite ddrmax.avi
```

Repare que o terminal será inundado com mensagens de depuração sempre que houver som, isso deixa o MAME tão ocupado que o sistema passa a rodar quadro a quadro na tela ao mesmo tempo que toda a ação, áudio e vídeo é gravado no arquivo **ddrmax.avi** dentro do diretório **snap**.

7.11.2 Usando uma interface lenta

Diferente do capítulo anterior, neste caso não há a necessidade de recompilar o MAME, abra o seu `mame.ini` e altere o caminho do **snapshot_directory** para um driver lento, como um HDD externo via **USB1**, um leitor de cartão lento e assim por diante. Devido a lentidão do destino, o MAME ficará ocupado esperando a gravação do arquivo ***.avi** resultando num gameplay em câmera lenta.

7.11.3 Usando a opção speed

Aqui também não há a necessidade de recompilar o MAME, porém se consegue no máximo uma redução de velocidade em torno de 10 fps e não é possível gravar um arquivo ***.avi** durante o gameplay sendo necessário que se jogue uma vez e só depois seja possível a gravação do ***.avi**.

Execute o sistema com o comando:

```
mame -speed 0.2 -rec dance.inp -sound none ddrmax
```

Neste estágio o som não é necessário pois estamos gravando apenas os comandos, ao terminar pressione **ESQ** para encerrar a emulação e gravar o arquivo `dance.inp`, agora execute o comando abaixo:

```
mame -pb dance.inp -aviwrite ddrmax.avi ddrmax
```

O MAME executará todos os comandos gravados em `dance.inp` ao mesmo tempo em que grava o vídeo. A jogabilidade não chega a ser tão precisa, mas nada que uma pouco de prática não resolva. Nos links abaixo é possível ver uma pequena demonstração de como fica a jogabilidade usando a opção `-speed`.

<https://www.youtube.com/watch?v=JaX2EYJKiJA>

<https://www.youtube.com/watch?v=6HSrrd1Vf2M>

<https://www.youtube.com/watch?v=MxktlN9moGs>

Para os mais interessados, recomendo a pesquisa do termo **TAS** no YouTube.

raw:: latex

clearpage

7.12 Truques para tornar a vida mais fácil

- *Configurando a ordem dos botões*
- *Configurando mais de um botão*
- *Combinando os botões*
- *Usando o controle 1 para controlar qualquer outro jogador*
- *Criando uma configuração de controle por sistema*
- *Gerando arquivos DAT para diferentes gerenciadores de ROMs*
- *Criando arquivos DAT separados por sistema*
- *Criando um ROMSET apenas com sistemas tipo Arcade*
 - *Usando a interface*
 - *Usando o Clrmamepro*
 - *Usando Scripts*
 - *Usando a linha de comando no Windows*
 - *Usando o terminal no Linux, macOS e *NIX em geral*
- *Resolvendo o redirecionamento do MAME com PowerShell da Microsoft #5694*
- *Resolvendo o problema "please call attendant" dos sistemas DrumMania*
- *Usando uma imagem de fundo no MAME*
- *Melhorando o desempenho em sistemas Linux*
 - *Ajustando a frequência da CPU*
 - *Identificando o modelo da sua placa de vídeo*
 - *Ativando o Vulkan no Fedora*
 - *Ativando o Vulkan no Debian 10/11*
 - *Ops, alguma coisa deu errado!*
 - *Removendo a âncora*
 - *Configurando o MAME com o vulkan*
- *Excluindo arquivos NVRAM (script)*
- *Removendo a suavização dos efeitos crt-geom/crt-geom-deluxe*
- *Listando as diferentes BIOS do Neo Geo*
- *Criando uma lista com todas as BIOS compatíveis usadas pelo MAME*
- *Listando e iniciando diferentes sistemas MSX*
- *Filtrando o nome das ROMs a partir do arquivo XML*
- *Criando uma lista de jogos*
- *Correção da proporção da tela para o SNES*
- *Usando o HLSL no Windows 8/10/11 sem precisar instalar o DirectX 9*

- *Convertendo alguns arquivos .CHD antigos para o novo formato createdvd*
- *Convertendo um CD-ROM para CHD com o chdman*
- *Gerando um arquivo de registro de um arquivo CHD*
- *Extraindo uma lista de sistemas usando a interface*
- *Extraindo uma lista de sistemas de um determinado driver*
- *Como criar apenas um mame.ini com as configurações de fábrica*

7.12.1 Configurando a ordem dos botões

O MAME por predefinição já assume uma ordem para os botões de um controle, porém nem sempre essa ordem estará na ordem dos botões originais de um determinado sistema, sendo necessário a configuração manual, os exemplos a seguir mostram como alterar essa ordem.

Veja *Alterando os valores* antes de prosseguir.

O processo é muito simples, inicie um sistema qualquer como `mame galaxian`, depois que o sistema iniciar pressione Tab e selecione *Atribuições da entrada (este sistema)*, no campo *P1 BUTTON 1* (primeiro botão de disparo/tiro do jogador 1) e defina o botão de tiro, pressione Tab novamente para fechar a interface.

Depois de confirmar o funcionamento do botão, pressione Esc para encerrar a emulação e criar um arquivo `galaxian.cfg` no diretório **cfg**.

7.12.2 Configurando mais de um botão

Usando um controle de **Playstation 2** (ou qualquer outro tipo de controle) ligado no PC com um adaptador USB como exemplo, faremos uma configuração dos botões para os sistemas **Neo-Geo** executando o comando `mame kof2000`, pressione Tab, depois selecione *Atribuições da entrada (este sistema)* e configure os botões com a ordem da sua preferência.

Dentro do diretório **cfg** será criado um arquivo chamado `kof2000.cfg`, abra ele num editor de texto qualquer, no topo do arquivo modifique o `<system name=kof2000>` para `<system name=default>` e salve este arquivo no diretório **ctrlr** como `neogeo.cfg`. No nosso exemplo a ordem dos botões ficou assim, o **quadrado** é **soco fraco**, o **triângulo** é **soco forte**, o **xis** é **chute fraco** e o **círculo** é **chute forte**:

Baixe uma cópia deste arquivo no link ao lado <https://pastebin.com/9Xp97xcd>

```
<?xml version="1.0"?>
<!-- This file is autogenerated; comments and unknown tags will be stripped -->
<nameconfig version="10">
<system name="default">
  <input>
    <port tag=":edge:joy:JOY1" type="P1_BUTTON1" mask="16" defvalue="16">
      <newseq type="standard">
        JOYCODE_1_BUTTON4
      </newseq>
    </port>
    <port tag=":edge:joy:JOY1" type="P1_BUTTON2" mask="32" defvalue="32">
      <newseq type="standard">
        JOYCODE_1_BUTTON3
      </newseq>
    </port>
    <port tag=":edge:joy:JOY1" type="P1_BUTTON3" mask="64" defvalue="64">
      <newseq type="standard">
        JOYCODE_1_BUTTON1
      </newseq>
    </port>
    <port tag=":edge:joy:JOY1" type="P1_BUTTON4" mask="128" defvalue="128">
      <newseq type="standard">
        JOYCODE_1_BUTTON2
      </newseq>
    </port>
  </input>
</system>
</nameconfig>
```

Agora sempre que quiser usar essa configuração para os botões, basta usar a opção `-ctrlr`, exemplo `mame kof2003 -ctrlr neogeo`, você pode também adicionar esta opção ao seu `mame.ini` porém note que esta configuração será aplicada em todos os sistemas!

7.12.3 Combinando os botões

O personagem Zangief do **Street Fighter II** possui um golpe chamado **Double Lariat** que é ativado ao se pressionar os três botões de soco ao **mesmo tempo**, é possível criar um arquivo de configuração para que essa ação aconteça ao toque de um botão apenas.

Inicie o MAME com qualquer jogo da série mame sf2, pressione Tab, depois selecione *Atribuições da entrada (este sistema)*, ao configurar os botões para os três socos, pressione Del para apagar o valor, logo depois escolha o botão que deseja ser soco fraco, pressione Enter e escolha quase será o seu botão de **três socos**. Caso tenha feito tudo certo, deverá aparecer algo como **Joy 1 button 0 or Joy 1 button 1**, é muito importante aparecer o **OR** entre os botões.

Usando o mesmo controle de **Playstation 2** a ordem dos botões ficou dessa forma, o **quadrado** é **soco fraco**, o **triângulo** é **soco forte**, o **Xis** é **chute médio**, o **círculo** é **chute forte**, o **L1** é **soco médio**, o **R1** é **chute fraco** e o botão **L2** identificado como **JOYCODE_1_BUTTON5** faz o papel dos **três botões de soco**:

Baixe uma cópia deste arquivo no link ao lado <https://pastebin.com/p6dB9DMY>

```
<?xml version="1.0"?>
<mameconfig version="10">
<system name="default">
  <input>
    <port tag=":IN1" type="P1_BUTTON1" mask="16" defvalue="16">
      <newseq type="standard">
        JOYCODE_1_BUTTON4 OR JOYCODE_1_BUTTON5
      </newseq>
    </port>
    <port tag=":IN1" type="P1_BUTTON2" mask="32" defvalue="32">
      <newseq type="standard">
        JOYCODE_1_BUTTON7 OR JOYCODE_1_BUTTON5
      </newseq>
    </port>
    <port tag=":IN1" type="P1_BUTTON3" mask="64" defvalue="64">
      <newseq type="standard">
        JOYCODE_1_BUTTON1 OR JOYCODE_1_BUTTON5
      </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON4" mask="1" defvalue="1">
      <newseq type="standard">
        JOYCODE_1_BUTTON8
      </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON5" mask="2" defvalue="2">
      <newseq type="standard">
        JOYCODE_1_BUTTON3
      </newseq>
    </port>
    <port tag=":IN2" type="P1_BUTTON6" mask="4" defvalue="4">
      <newseq type="standard">
        JOYCODE_1_BUTTON2
      </newseq>
    </port>
  </input>
</system>
</mameconfig>
```

Uma nota quanto a configuração acima, ela foi feita no Linux (SDL) e pode ser que no Windows a definição para o botão **L1** seja alternada para o botão **L2**, porém basta redefini-lo no Windows ou alterná-lo para um outro botão qualquer depois.

O mesmo tipo de configuração também se aplica para qualquer sistema, cito por exemplo os sistema rítmicos da série **Guitar Freaks**, **Dance Dance Revolution**, **Beatmania** e tantas outros onde em alguns momentos, necessitam que mais de um botão seja acionado ao mesmo tempo.

Baixe um exemplo de configuração de controle para [Guitar Freaks](#) e [Dance Dance Revolution](#).

7.12.4 Usando o controle 1 para controlar qualquer outro jogador

Nos sistemas onde a posição do jogador é fixo como em **Sunset Riders**, a série **Muscle Bomber**, **Teenage Mutant Ninja Turtles**, **Teenage Mutant Ninja Turtles - Turtles in Time** e tantas outros sistemas, é possível utilizar o controle do jogador 1 em qualquer uma das posições com sistemas que aceitem 2 ou mais jogadores.

Inicie o sistema que deseja configurar e siga as instruções descritas no capítulo [Configurando mais de um botão](#) para gerar o seu arquivo com a configuração personalizada para o seu controle. Abra o arquivo num editor de textos e logo no começo haverá algo do tipo (pode variar dependendo do seu controle):

```
<port tag=":P1" type="P1_JOYSTICK_LEFT" mask="1" defvalue="1">
```

O que nos interessa são todas as partes que definem os comando para **P1**, no seu editor de texto pressione CTRL + H ou o comando usado para substituir texto e substitua **P1** por **P2** ficando assim:

```
<port tag=":P2" type="P2_JOYSTICK_LEFT" mask="1" defvalue="1">
```

Salve este arquivo como 2P.cfg para ser usado como jogador 2, faça o mesmo para o jogador 3 e 4. Não foi disponibilizado aqui todas as linhas que foram substituídas, porém, veja como fica a configuração completa para [2P.cfg](#), [3P.cfg](#) e [4P.cfg](#). Não custa lembrar que todos estes arquivos devem ficar armazenados dentro do diretório **ctrlr**.

Em sistemas Linux por exemplo a diferença entre maiúsculas e minúsculas são levadas em consideração, caso salve estes arquivos com **P** maiúsculo, faça o mesmo ao informar o nome da configuração, caso contrário o MAME acusará um erro dizendo que os arquivos não foram encontrados.

Para jogar com o **Donatello** (jogador 3) no sistema **Teenage Mutant Ninja Turtles - Turtles in Time** faça o comando:

```
mame tmnt2 -ctrlr 3P
```

Para inserir o crédito para o jogador 3 (Coin 3) pressione 7, a partida deve iniciar com o **Donatello** ao clicar em qualquer um dos botões do controle, abaixo tem uma colinha para facilitar, para ver a listagem completa consulte o capítulo [Cardápio de opções](#).

Tabela 5: Colinha básica.

Descrição	Jogador 1	Jogador 2	Jogador 3	Jogador 4
Crédito	5	6	7	8
Início da Partida	1	2	3	4

Repare que há sistemas como as da série **Muscle Bomber** por exemplo, é preciso pressionar os botões relacionados ao início da partida de cada jogador e não apenas pressionar os botões do controle para iniciar a partida.

7.12.5 Criando uma configuração de controle por sistema

No exemplo de [configuração de botões](#) aprendemos como mapear os botões de um controle para um determinado sistema, no entanto, um sistema pode ter sistemas com diferentes configurações de botões, se pegarmos o driver CPS1 por exemplo, o jogo **Street Fighter II** utiliza 6 botões já o jogo **Final Fight** utiliza apenas 2 e ambos compartilham o mesmo driver **CPS1**. Apesar da dica ter facilitado bastante a configuração do controle para os sistemas que usam 6 botões, ela não irá funcionar com todos os outros.

Para criar um mapa customizado para **Final Fight** por exemplo, siga as instruções descritas em [Configurando mais de um botão](#), porém usando o sistema **Final Fight** (`mame ffight`), uma vez que os botões forem definidos e você sair do MAME, encontre o arquivo `ffight.cfg` no diretório **cfg** e faça as alterações necessárias. Copie-o para o diretório **ctrlr** como `ffight.cfg`. Entre no diretório **ini** e crie um arquivo chamado `ffight.ini`, abra-o num editor de texto e adicione:

```
ctrlr ffight
```

Salve e saia do editor, agora sempre que o sistema **Final Fight** for iniciado, ele usará o novo mapa de configuração dos botões que foi criado só para ele.

Repare que não é necessário que o arquivo de configuração tenha o mesmo nome da sistema, é possível renomear estes arquivos como por exemplo, `2-botoes.cfg`, `3-botoes.cfg`, `4-botoes.cfg` e compartilhar estas configurações conforme necessário.

7.12.6 Gerando arquivos DAT para diferentes gerenciadores de ROMs

Arquivos DAT são usados por gerenciadores de ROMs como [RomCenter \(Windows\)](#), [RomVault \(Linux e Windows\)](#), [Romulus \(Windows\)](#), [Clrnamepro \(Windows\)](#), [Clrnamepro \(Mac\)](#) dentre outros que aferem a validade de cada arquivo existente dentro de um arquivo ROM identificando o CRC e SHA1 de cada um, dentre outras funções.

Execute o MAME com o comando:

```
mame -listxml >mame.xml
```

Baixe o [DatUtil](#), extraia-o no mesmo diretório do MAME e execute o comando:

```
datutil mame.xml
```

Será criado o arquivo `datutil.dat`.

7.12.7 Criando arquivos DAT separados por sistema

Para criar um DAT para o sistema CPS1 (**cps1.dat**) compatível com o **Clrnamepro** faça o comando:

```
datutil.exe -G cps1.cpp -o cps1.dat -f cmp datutil.dat
```

Para o sistema CPS2:

```
datutil.exe -G cps2.cpp -o cps2.dat -f cmp datutil.dat
```

Para o sistema Neo-Geo:

```
datutil.exe -G neogeo.cpp -o neogeo.dat -f cmp datutil.dat
```

Para uma lista de Neo-Geo sem clones:

```
datutil.exe -G neogeo.cpp -o neogeo.dat -r -f cmp datutil.dat
```

Nota: A estrutura interna do MAME para a criação de projetos foi modificada depois [desta alteração](#). Assim, em vez de usar `-G neogeo.cpp` (por exemplo), é preciso usar `neogeo/neogeo.cpp` para que o `datutil` consiga identificar e criar o respectivo arquivo `dat`, caso contrário, ele aponta um erro na criação do arquivo. Para identificar a nomenclatura correta do driver, utilize a opção `-ls` seguido do nome do sistema, para o **Street Fighter II** (`mame -ls sf2`), por exemplo, a opção retorna `capcom/cps1.cpp`. Esta deve ser a opção usada com o `datutil` nas versões mais novas do MAME a partir da versão **0.246**.

E assim por diante, para criar um DAT em formato **RomCenter** troque o `cmp` por `rc`, para **RomCenter 2** use `rc2` e para criar um arquivo XML genérico aceito pelos outros gerenciadores use `gx` ou `generic`. Para mais informações sobre outros formatos leia o arquivo **Readme.txt** que acompanha o `DatUtil`, para ver alguns outros exemplos práticos do programa, acesse [este link](#).

Estes mesmos arquivos `neogeo.dat`, `cps2.dat` e qualquer outro que for criado poderá ser utilizado pelos gerenciadores de ROMs para construir um ROM SET para cada um destes sistemas. É uma maneira muito mais fácil de se separar as ROMs do que ter que fazer e [usar scripts](#). No entanto, apesar de ser mais fácil utilizar um gerenciador, repare que é bem genérico. O uso de scripts permitem que a separação seja bem mais específica caso seja necessário.

7.12.8 Criando um ROMSET apenas com sistemas tipo Arcade

Umas das maneiras de se criar tal ROMSET é baixando o código-fonte e compilando o MAME com a opção SUBTARGET=arcade, isso fará com que o MAME funcione e exiba apenas uma lista com sistemas classificados internamente como "arcade", simples assim. Para mais informações leia o capítulo [Compilando o MAME](#).

Usando a interface

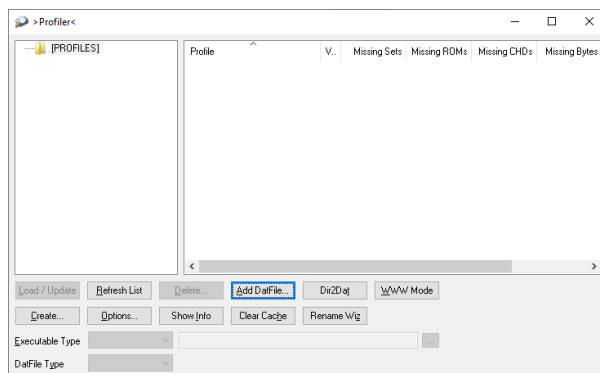
Para aqueles que não estão familiarizados(as) com o processo de compilação, só utilizam a versão oficial do MAME e que também não tenham interesse em montar todo um ambiente de desenvolvimento só para isso, é possível criar essa lista através da interface do MAME, o que facilita muito a nossa vida.

- Faça o download da última versão do arquivo `category.ini` no site [Progetto-Snaps](#) e extraia o diretório **folders** dentro do diretório do MAME.
- Inicie o MAME, no lado esquerdo da interface selecione *Categoria*, em *Arquivo* escolha *Working Arcade Clean.ini*, em *Incluir Clones* escolha *Não* e clique em *Retorna ao menu anterior*.
- No topo da interface, clique com o mouse no ícone do disquete para exportar a lista e escolha *Exporta a lista em formato XML (igual -listxml)*, depois de alguns segundos será gerado um arquivo **exported.xml** dentro do diretório **ui**.
- Assim como foi explicado no [capítulo anterior](#), é possível usar o *DatUtil* para transformar o arquivo XML num arquivo DAT compatível com um dos gerenciadores listados no capítulo anterior ou utilizar diretamente o arquivo XML nos gerenciadores que não dependam de um arquivo DAT (como o *Clrmamepro*).
- Convertendo ou não o arquivo XML em DAT, use o seu gerenciador preferido para **reconstruir (Re-build)** as ROMs, usando o diretório onde as suas ROMs se encontram e com o destino o diretório onde deseja ter somente as ROMs de arcade.

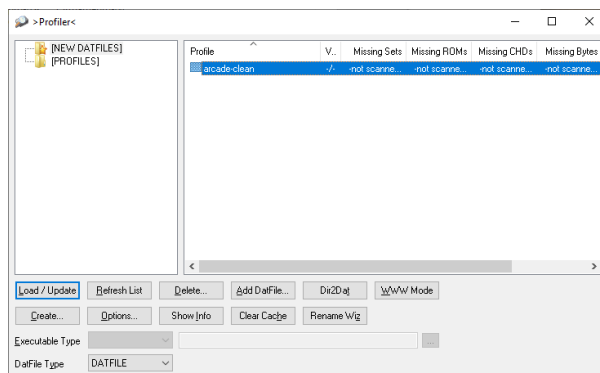
Usando o Clrmamepro

O **Clrmamepro** é apenas um dos diversos programas disponíveis para gerenciar as suas ROMs, o exemplo abaixo mostra como usar o arquivo XML gerado no passo anterior para **reconstruir (Rebuild)** o seu ROMSET apenas com ROMs arcade.

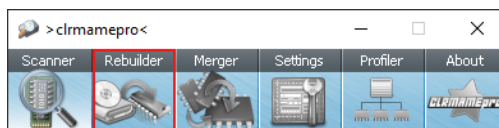
- Baixe e instale o **Clrmamepro** é importante saber em qual diretório ele foi instalado!
- Renomeie o arquivo **exported.xml** que está dentro do diretório **ui** para **arcade-clean.xml** assim é possível ter um controle do arquivo que está sendo carregado.
- Localize o diretório onde o **Clrmamepro** foi instalado e copie o arquivo **arcade-clean.xml** para dentro do diretório **datfiles**.
- Rode o **Clrmamepro** no campo direito onde diz **Profile** deve estar vazio.



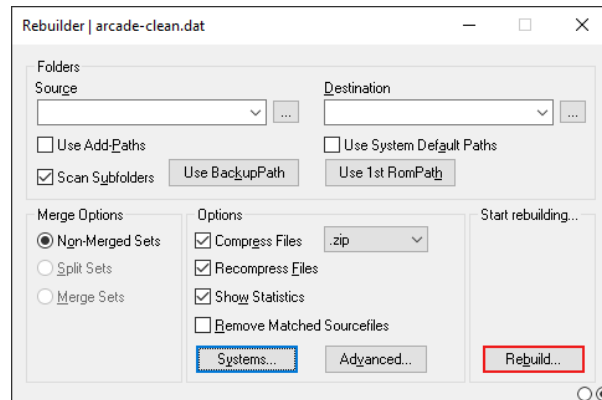
- Clique em **Add DatFile...** e selecione o arquivo **arcade-clean.xml**, na próxima tela apenas clique em **OK**.
- Clique em **Load / Update**, na próxima tela selecione **Default**.



- Durante a leitura deve aparecer alguns erros do tipo **DatFile Problem**, apenas clique em **OK TO ALL**.
- Ao concluir clique no ícone **Rebuilder**.



- Sem alterar nenhuma das opções, vá em *Source* e selecione o diretório onde se encontram todas as suas ROMs. Em *Destination* selecione o diretório de destino onde será criada a sua nova *ROMSET*.



- ATENÇÃO! Não escolha o mesmo diretório de origem**, escolha um diretório completamente diferente e se possível que seja em outro HDD, cartão de memória, pen-drive, etc!
- Clique em *Rebuild...* para iniciar e aguarde pois dependendo da quantidade de arquivos todo o processo será demorado.

Ao final do processo você terá um *ROMSET* apenas com as ROMs dos sistemas arcades, o mesmo pode ser feito para qualquer outro sistema, Mega Drive/Genesis, SNES, o céu é o limite!

Usando Scripts

Uma outra maneira para obter o mesmo resultado é através da utilização de pequenos *scripts* usando a linha de comandos, apesar de ser um processo mais manual e um pouco trabalhoso, o processo acaba sendo mais poderoso pois permite que a filtragem e a seleção dos arquivos possa ser mais refinada e podendo ser utilizada em qualquer sistema operacional e não algo exclusivo do Windows apenas.

- Faça o download da última versão do arquivo `category.ini` no site [Progetto-Snaps](#).
- Abra e extraia apenas o arquivo `Working Arcade.ini`.
- Apague tudo e deixe apenas o que estiver depois de `[ROOT_FOLDER]`.
- Salve este arquivo modificado como `arcade.txt`.

Usando a linha de comando no Windows

Abra o prompt de comando no mesmo diretório onde se encontra o arquivo `arcade.txt` defina o caminho completo para onde deseja copiar os arquivos:

```
set DST=H:\arcade-roms
```

Seguido do comando abaixo:

```
for /F %f in ('type arcade.txt') do @echo G:\roms\%f.zip >> caminho-roms.txt
```

O comando acima vai ler todos os nomes dos sistemas em `arcade.txt`, incluir o caminho completo onde estão armazenadas as suas ROMs, adicionar o nome do sistema + a extensão `.zip` e por fim redirecionar a saída para o arquivo `caminho-roms.txt`.

Execute o comando abaixo para realizar a cópia dos arquivos com base na lista que acabamos de criar:

```
for /F %f in ('type caminho-roms.txt') do copy %f %DST%
```

Assim como no comando acima, o arquivo caminho-roms.txt será lido e posteriormente irá alimentar o comando copy com o devido caminho e o destino H:\arcade-roms.

Usando o terminal no Linux, macOS e *NIX em geral

Como descrito acima, abra o terminal no mesmo diretório onde se encontra o arquivo arcade.txt e defina o diretório de destino:

```
export DST=/mnt/usb/arcade-roms
```

É necessário converter o formato do arquivo de Windows (quebra de linha **CRLF**) para um formato compatível com *nix (quebra de linha **LF**), caso contrário a lista ficará toda bagunçada:

```
sed -i 's/\r//g' arcade.txt
```

Execute o comando abaixo para gerar o arquivo caminho-roms.txt onde **/home/mame/roms** é o caminho completo onde as ROMs estão armazenadas:

```
for f in $(< arcade.txt); do echo /home/mame/roms/"$f".zip; done > caminho-roms.txt
```

Execute o comando abaixo para fazer a cópia dos arquivos:

```
for f in $(< caminho-roms.txt); do cp "$f" "$DST"; done
```

Para separar um *ROMSET* com todas as ROMs para **Neo Geo** usando apenas o terminal, crie o mame.xml com o comando:

```
mame -lx > mame.xml
```

Faça o comando abaixo para criar uma lista destas ROMs nas versões anteriores do **MAME 0.246**:

```
cat mame.xml | grep 'sourcefile="neogeo.cpp"' | sed -rn 's/.* name="([a-z0-9]+)" .*/\1/p' | \
↪awk '!seen[$0]++' | sort -d > maquinas
```

Para novas versões após a versão **0.246**:

```
cat mame.xml | grep 'sourcefile="neogeo/neogeo.cpp"' | sed -rn 's/.* name="([a-z0-9]+)" .*/\1/p' | \
↪awk '!seen[$0]++' | sort -d > maquinas
```

O primeiro comando `cat mame.xml` lista o arquivo mame.xml, o segundo comando filtra as linhas que contém `sourcefile="neogeo/neogeo.cpp"`, o terceiro `sed -rn 's/.* name="([a-z0-9]+)" .*/\1/p'` seleciona os nomes, o quarto comando `awk '!seen[$0]++'` remove os itens repetidos, o último `sort -d` organiza a lista em ordem alfabética e por último `> maquinas` redireciona todo o processamento para o arquivo maquinas.

Dentro do arquivo maquinas nós teremos uma lista que inclui os clones e a BIOS:

```
2020bb
2020bba
2020bbh
3countb
...
```


Usando o mesmo exemplo, porém, criando uma lista **sem clones** e **sem BIOS**, use o comando abaixo:

```
cat mame.xml | grep 'romof="neogeo"' | sed -rn 's/.* name="([a-z0-9]+)" .*/\1/p' | awk '!
↪seen[$0]++' | sort -d > maquinas
```

Assim teremos a seguinte lista:

```
2020bb
3countb
alpham2
androdun
...
```

Agora com ou sem clones, geramos o arquivo com o caminho completo para as ROMs que nós queremos:

```
while read maquinas; do echo /media/mame/roms/"$maquinas".zip ; done < maquinas > lista-roms
```

O arquivo maquinas alimenta maquinas do while read que vai substituindo os valores da lista em "\$maquinas" assim que eles vão sendo concluídos e no final redireciona a nossa lista pronta para lista-roms. Isso gera a seguinte lista:

```
/media/mame/roms/2020bb.zip
/media/mame/roms/3countb.zip
/media/mame/roms/alpham2.zip
/media/mame/roms/androdun.zip
...
```

Com a lista em mãos, supondo que eu queira criar uma pasta exclusiva para ROMs de *Neo Geo* como por exemplo `/home/mame/roms/neogeo`, primeiro eu crio o diretório com `mkdir /home/mame/roms/neogeo` e em seguida, posso usar o comando abaixo para copiar todas as ROMs para dentro desta pasta:

```
while read copy ; do cp "$copy" /home/mame/roms/neogeo ; done < lista-roms
```

Assim como no exemplo anterior, copy de while read é alimentado por lista-roms que vai substituindo os valores da lista em "\$copy" assim que eles vão sendo concluídos e copiando os arquivos da lista para `/home/mame/roms/neogeo`.

Ao final, nós teremos todas as ROMs de *Neo Geo* dentro da pasta escolhida. Note porém que o processo não é 100% perfeito para todos os sistemas e talvez seja necessário verificar as ROMs com o seu *gerenciador de ROM* preferido. Com o Windows, é preferível gerar um arquivo DAT (neogeo.dat) como explicado em *Criando arquivos DAT separados por sistema* e depois usar o Clrmamepro para verificar se todas as ROMs foram mesmo corretamente copiadas.

7.12.9 Resolvendo o redirecionamento do MAME com PowerShell da Microsoft #5694

Ao redirecionar a saída do MAME com o comando `-listxml / -lx` usando o PowerShell da Microsoft, a saída tem o dobro de tamanho se comparado com a saída do mesmo comando ao se utilizar o terminal do Linux, macOS ou o comando prompt do Windows.¹

Segundo mostra [este artigo](#) hospedado num blog de desenvolvimento da Microsoft, a codificação predefinida do PowerShell não é UTF-8, originalmente ele vem como `us-ascii`:

¹ #5694 <https://github.com/mamedev/mame/issues/5694>

```
$OutputEncoding
```

```
IsSingleByte      : True
BodyName          : us-ascii
EncodingName      : US-ASCII
HeaderName        : us-ascii
WebName           : us-ascii
WindowsCodePage   : 1252
IsBrowserDisplay  : False
IsBrowserSave     : False
IsMailNewsDisplay : True
IsMailNewsSave    : True
EncoderFallback   : System.Text.EncodingReplacementFallback
DecoderFallback   : System.Text.DecoderReplacementFallback
IsReadOnly        : True
CodePage          : 20127
```

Ao fazer o redirecionamento, a saída é codificada para [iso-10646-ucs-2 BOM](#), isso faz com que cada caractere comum seja armazenado com 2 bytes. Geralmente o UTF-8 por exemplo utiliza de 1 a 4 bytes para caracteres [diacríticos](#), assim como caracteres Cirílico, Grego, etc.

Para arrumar apenas o redirecionamento > ou >> faça o comando no terminal do PowerShell:

```
$PSDefaultParameterValues['Out-File:Encoding'] = 'utf8'
```

Para mudar a codificação de todo o terminal, faça o comando:

```
$OutputEncoding = [Console]::OutputEncoding = [Text.UTF8Encoding]::UTF8
```

Um exemplo de como agora fica a codificação do terminal:

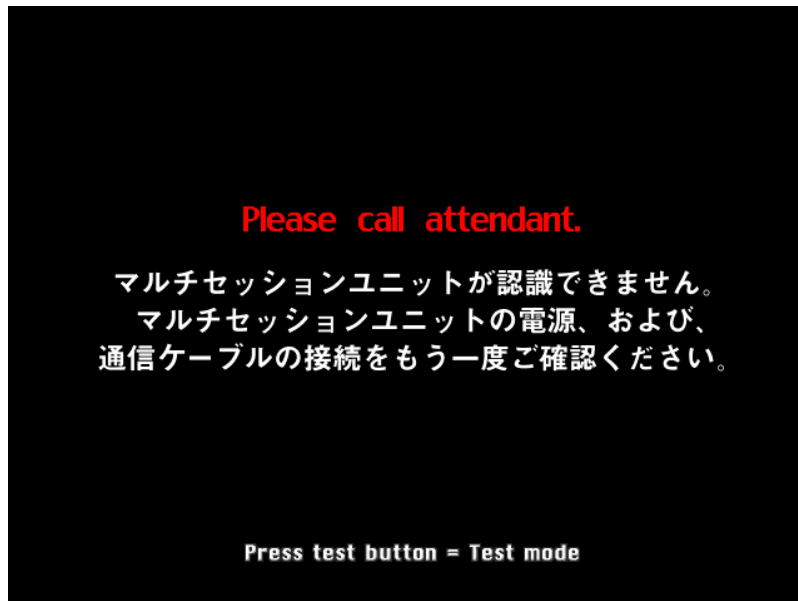
```
$OutputEncoding
```

```
BodyName          : utf-8
EncodingName      : Unicode (UTF-8)
HeaderName        : utf-8
WebName           : utf-8
WindowsCodePage   : 1200
IsBrowserDisplay  : True
IsBrowserSave     : True
IsMailNewsDisplay : True
IsMailNewsSave    : True
IsSingleByte      : False
EncoderFallback   : System.Text.EncodingReplacementFallback
DecoderFallback   : System.Text.DecoderReplacementFallback
IsReadOnly        : True
CodePage          : 65001
```

Qualquer uma das opções funcionam, não é necessário usar as duas. Para mais informações [veja este post](#) para saber localizar os perfis do PowerShell no Windows e alternar estes valores para que fiquem permanentes ou que sejam executados sempre que uma sessão do PowerShell seja iniciada.

7.12.10 Resolvendo o problema "please call attendant" dos sistemas DrumMania

Os sistemas **DruMania 4th Mix** em diante não iniciam e param numa tela de erro como mostra a imagem abaixo:



Existem duas maneiras de resolver o problema, aplicando um patch na imagem o que altera a sua integridade ou usando um **cheat**. Crie os arquivos abaixo dentro do diretório **cheat**, estes arquivos não são de minha autoria e desconheço o autor original, caso alguém conheça, entre em contato que o devido crédito será dado.

drmn4m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@80047F24=00000000</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/JyaTSr4c>

drmn5m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@8003BAF8=00000000</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/dTQMeJB1>

drmn6m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@8004F6E0=00000000</action>
      <action>maincpu.pd@8004F740=00000000</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/tRAfhfaS>

drmn7m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@80073F84=00000000</action>
      <action>maincpu.pd@80073FE4=00000000</action>
      <action>maincpu.pd@800E1B64=10000017</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/LtMyNZ7i>

drmn7ma.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@8006170C=00000000</action>
      <action>maincpu.pd@8006176C=00000000</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/MyX6scPk>

drmn8m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@800A4544=00000000</action>
      <action>maincpu.pd@800A45A4=00000000</action>
      <action>maincpu.pd@800E35F0=10000017</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/LGTnUd4Y>

drmn9m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@800B92C0=00000000</action>
      <action>maincpu.pd@800B9320=00000000</action>
      <action>maincpu.pd@80106634=10000017</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/at99MLqz>

drmn10m.xml

```
<mamecheat version="1">
  <cheat desc="Please Call Attendant Fix">
    <script state="run">
      <action>maincpu.pd@800BC854=00000000</action>
      <action>maincpu.pd@800BC8B4=00000000</action>
      <action>maincpu.pd@8010C4F4=10000017</action>
    </script>
  </cheat>
</mamecheat>
```

<https://pastebin.com/EaJes6Eh>

7.12.11 Usando uma imagem de fundo no MAME

É possível utilizar uma imagem de fundo no MAME que é exibida na interface como um papel de parede, somado com a opção de customização, é possível alterar a aparência da sua interface. Escolha a imagem que deseja usar em formato .jpg ou .png e renomeie o arquivo para background.jpg ou background.png, cuidado para não usar imagens muito pesadas, prefira o formato .jpg.

Ao iniciar o MAME vá em *Definições gerais*, *Opções diversas* e ative a opção *Usa uma imagem como plano de fundo*.

Para ter uma interface com cores diferentes ao do padrão do MAME, experimente a configuração abaixo, ela usa uma paleta de cores do filme Tron:

```
# UI OPTIONS
#
infos_text_size      0.700000
font_rows            37
hide_main_panel      0
ui_border_color      ffb7e7eb
ui_bg_color           c8022f35
ui_clone_color       ff808080
ui_dipsw_color       ff03d2d8
ui_gfxviewer_color   ef101030
ui_mousedown_bg_color 9470b3d0
ui_mousedown_color   fffa26ec
ui_mouseover_bg_color 70276e84
ui_mouseover_color   ff25b9a9
ui_selected_bg_color  ef047289
ui_selected_color    ff03d2d8
ui_slider_color      ffffffff
ui_subitem_color     ffffffff
ui_text_bg_color     ef000000
ui_text_color        ffffffff
ui_unavail_color     ff404040
```

Salve estas opções no arquivo ui.ini no Windows e no Linux fica em ~/.mame/ui.ini.

Caso queira brincar com as cores, eu gosto do site [Hex Colors Tools](#). No site é possível você entrar com o valor de cores em hex (#000000) e ela oferece a possibilidade de variar a cor para mais claro ou mais escuro em intervalos de 10% ou um outro valor qualquer.

Já para a combinação das cores é necessário saber sobre cores primárias, cores frias, quentes, monocromáticas, análogas, saber como utilizar o círculo cromático (dentre outras ferramentas) etc. Um tópico desta natureza daria um livro (ou mais de um livro) só sobre o assunto, logo este tópico não é coberto por este documento, no entanto, deixo algumas sugestões de leitura como [combinando cores 1](#), [combinando cores 2](#), [combinando cores 3](#), isso sem falar nas centenas de milhares de vídeos no YouTube sobre o assunto.

Já em termos de ferramentas eu gosto bastante da [Paletton](#) e a [Adobe](#).

Infelizmente a customização das cores da interface do MAME é muito limitada pois algumas cores são fixas como o verde que fica no título das janelas ou o texto verde das opções que estão ligadas, aquele azul da seleção das ROMs, etc.

7.12.12 Melhorando o desempenho em sistemas Linux

Os sistemas como **Dance Dance Revolution**, **Guitar Freaks**, **DrumMania** dentre outros no Windows, eles funcionam sem qualquer problema, porém sofrem com diversos problemas no Linux, um dos motivos é a configuração "padrão" que "vem de fábrica". Geralmente o seu sistema Linux vem configurado em modo de economia de energia, drivers genéricos e configurações básicas para o seu hardware, claro que não podemos nos esquecer que o MAME é um ávido consumidor de recursos de hardware, logo, quanto melhor, bem configurado e mais recente for o seu hardware melhor será a sua experiência com o MAME. A melhoria no desempenho contudo não será absoluta, extrairemos o melhor possível porém este desempenho se limita ao desenvolvimento do MAME, se os drivers responsáveis pelo sistema em questão já foram concluídos ou não, se o desenvolvimento da emulação como um todo já foi concluído ou não, etc.

Os testes foram realizados com o **Debian 11.5** (Buster) e o **Fedora 33** usando uma **AMD Radeon HD 7750** porém as configurações descritas aqui devem ser compatíveis com outras distribuições Linux ou talvez sirva como um guia para outros modelos de placas de vídeo. Não entraremos nas questões de instalação de pacotes dada a complexidade de cobrir todas as suas dependências e sim apenas na configuração.

Nota: Tenha certeza de utilizar uma versão mais recente do Linux e do MAME!

Nota: Considere o site pkg.org para pesquisar os pacotes para a sua distribuição.

Aviso: Antes de prosseguir saiba que dependendo da versão do driver amdgpu que você estiver usando ele pode **não ter áudio HDMI**, será preciso usar a sua placa de som ou fones de ouvidos.

Ajustando a frequência da CPU

O modo de economia de energia do processador pode atrapalhar bastante o desempenho do MAME, incluindo a lentidão de alguns jogos quando rodados no Linux que rodam sem lentidão no Windows, assim como, jogos que ficam com o áudio falhando ou picotando.

Para ver em que modo o seu processador está rodando, execute o comando abaixo no seu terminal:

```
cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
```

O padrão para a maioria dos casos é conservative, isso faz com que a frequência do seu processador *seja mantida no mínimo*, conservando energia, porém, atrapalhando o desempenho geral da emulação.

Para alterar isso, no Debian instale o pacote linux-cpupower:

```
sudo apt install linux-cpupower
```

No Fedora, instale o kernel-tools:

```
sudo dnf install kernel-tools
```

Acesse o site pkgs.org para identificar em qual pacote vem o cpupower para a sua distro.

Um vez instalado, rode o comando abaixo para identificar quais os modos o seu processador suporta:

```
sudo cpupower frequency-info
analizando o CPU 0:
  driver: acpi-cpufreq
  CPUs que rodam na mesma frequência de hardware: 0
  CPUs que precisam ter suas frequências coordenadas por software: 0
  maior latência de transição: 4.0 us
  limites do hardware: 1.40 GHz - 4.00 GHz
  available frequency steps: 4.00 GHz, 3.40 GHz, 2.80 GHz, 2.10 GHz, 1.40 GHz
  reguladores do cpufreq disponíveis: performance schedutil
  política de frequência atual deve estar entre 1.40 GHz e 4.00 GHz.
    O regulador "performance" deve decidir qual velocidade usar
    dentro desse limite.
  current CPU frequency: 4.00 GHz (asserted by call to hardware)
  boost state support:
    Supported: yes
    Active: yes
    Boost States: 2
    Total States: 7
    Pstate-Pb0: 4200MHz (boost state)
    Pstate-Pb1: 4100MHz (boost state)
    Pstate-P0: 4000MHz
    Pstate-P1: 3400MHz
    Pstate-P2: 2800MHz
    Pstate-P3: 2100MHz
    Pstate-P4: 1400MHz
```

No nosso caso podemos utilizar performance e schedutil, o modo performance faz com que o processador rode com a sua frequência máxima, no nosso caso, **4000MHz**. Já o modo schedutil faz com que a frequência do processador varie conforme a demanda.

Para alterar o modo, execute o comando abaixo:

```
sudo cpupower -c all frequency-set -g schedutil
```

É possível deixar como performance porém lembre-se, neste modo o processador sempre vai trabalhar com a frequência máxima, ainda que nada esteja sendo feito no seu computador, causando um aquecimento e um consumo extra de energia sem qualquer necessidade. O modo schedutil é o melhor dos dois mundos pois acelera a frequência do processador quando for preciso e reduz ao mínimo quando nada estiver sendo feito.

Em sistemas com KDE e Gnome, é possível ir nas configurações de **energia** do sistema e escolher a opção *balanceado*, que também define o modo de operação do processador como schedutil.

Acesse a [documentação do kernel](#) para obter mais informações estas e outras opções do gerenciamento de energia do kernel.

Identificando o modelo da sua placa de vídeo

No terminal execute o comando:

```
lspci |grep VGA  
01:00.0 VGA compatible controller: Advanced Micro Devices, Inc.  
[AMD/ATI] Cape Verde PRO [Radeon HD 7750/8740 / R7 250E]
```

O que nos interessa é o nome do *chipset* da placa **Cape Verde**, ela é da família **Southern Islands** ou **SI**. Consulte [a lista completa](#).

Confira qual o driver que está sendo utilizado no momento:

```
lspci -vs 01:00.0|grep driver  
Kernel driver in use: radeon
```

Ativando o Vulkan no Fedora

Nem todos os pacotes e as suas respectivas dependências estão listadas, rode o comando abaixo para instalar os pacotes necessários:

```
sudo dnf install linux-firmware xorg-x11-drv-amdgpu vulkan-tools vulkaninfo radeontop mesa-  
↪vulkan-drivers mesa-dri-drivers tuned glx-utils
```

É preciso passar alguns parâmetros para o kernel no arquivo `/etc/default/grub`, na opção `GRUB_CMDLINE_LINUX` deve haver algo do tipo:

```
GRUB_CMDLINE_LINUX="rhgb quiet"
```

Adicione as opções para a sua placa de vídeo, para o nosso exemplo eles seriam:

```
GRUB_CMDLINE_LINUX="rhgb quiet pcie_aspm.policy=performance radeon.si_support=0 amdgpu.si_  
↪support=1 modprobe.blacklist=radeon amdgpu.gpu_recovery=1 amdgpu.pcie_gen2=1 amdgpu.dpm=1"
```

Aviso: Observe que independente de como a linha acima apareça aqui neste documento, ela é direta e contínua!

- **pcie_aspm.policy**

Desliga o gerenciamento de energia dos slots PCIe e permite que os dispositivos conectados a ele trabalhem com o máximo desempenho. Os valores válidos são `default`, `powersave` e `performance`, por predefinição o sistema usa economia de energia.

- **radeon.si_support**

Ativa (1) ou desativa (0) o suporte ao driver **radeon**.

- **amdgpu.si_support**

Ativa (1) ou desativa (0) o suporte ao driver **amdgpu**.

- **modprobe.blacklist**

Evita que o kernel carregue o driver **radeon**.

- **amdgpu.gpu_recovery**

Caso a sua placa de vídeo trave por algum motivo qualquer deixando a sua tela parada, o mecanismo de recuperação entra em ação.

- **amdgpu.pcie_gen2**

Impõem o uso da geração da PCIe mais recente, não use se a sua placa-mãe e a placa de vídeo não forem compatíveis.

- **amdgpu.dpm**

Gerenciamento dinâmico de energia, faz com que a sua GPU economize energia e trabalhe fria quando não estiver em uso e ofereça o máximo desempenho apenas quando for preciso.

Execute o comando abaixo caso o seu PC use EFI:

```
sudo grub2-mkconfig -o /boot/efi/EFI/fedora/grub.cfg
```

Ou sem EFI:

```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

Caso não queira lidar com o grub ou se a sua distribuição não usar o grub, crie um arquivo `amdgpu.conf` dentro do diretório `/etc/modprobe.d` com as mesmas opções:

```
options radeon si_support=0
options amdgpu si_support=1
options amdgpu pcie_gen2=1
options amdgpu gpu_recovery=1
options amdgpu dpm=1
options pcie_aspm policy=performance
blacklist radeon
```

Regenere o **initramfs** no Fedora com o comando `sudo dracut -fv` e **reinicie o seu computador**. Para aqueles que tem a opção de usar ambos, escolha um ou o outro, **não utilizem os dois juntos!** Particularmente prefiro usar o **modprobe** em vez do **grub** pois qualquer erro que seja feito na configuração do arquivo do grub o seu sistema não inicia mais e dá um baita trabalho arrumar depois, já pelo modprobe a única coisa que acontece são alguns erros no seu registro de logs. Em termos de desempenho ambos são iguais.

É possível listar todos os parâmetros disponíveis do módulo **amdgpu** (ou qualquer outro módulo) com o comando `modinfo amdgpu|grep parm`, quase todos eles estão disponíveis no diretório `/sys/class/drm/card0/device/driver/module/parameters/`, apesar de estarem disponíveis não significa que todos eles sejam compatíveis com a sua placa de vídeo.

Isso nada tem a ver com o MAME e o MAME tão pouco tira proveito desta configuração específica, no entanto como já estamos turbinando as configurações, caso o seu monitor e a sua placa de vídeo sejam compatíveis com "Deep Color" ela pode ser ativada com a opção:

```
options amdgpu deep_color=1
```

Regenere o **initramfs** e reinicie.

Nota: Antes das telas "Full HD" os monitores trabalhavam com VGA e usavam **8 bit** para cada canal de cor RGB (Vermelho, Verde e Azul) ou 256 (2^8) variações de cores para cada componente RGB dando um total de **16.777.216** (256^3) ou 16.7 milhões de cores, nos PC's do final dos anos 90 o Windows exibia esta configuração como **True Color**. Com o **Deep Color** nós temos **12 bit** por canal, ou seja, 4096 (2^{12}) variações de cores para cada componente, estamos falando de um total de **68.719.476.736** (4096^3) ou cerca de **68.7 bilhões de cores**.

Tabela 6: Um exemplo **exagerado** das diferenças.



Verifique se o driver **amdgpu** está em uso:

```
lspci -vs 01:00.0 | grep driver
Kernel driver in use: amdgpu
```

Verifique se tudo está em ordem:

```
glxinfo -B | grep "OpenGL renderer" && glxinfo -B | grep "OpenGL version"

OpenGL renderer string: AMD Radeon HD 7700 Series
(VERDE, DRM 3.39.0, 5.9.13-200.fc33.x86_64, LLVM 11.0.0)
OpenGL version string: 4.6 (Compatibility Profile) Mesa 20.2.4
```

Execute o comando **vulkaninfo** e verifique se ele não acusa qualquer erro, se tudo estiver certo aparecerá uma lista detalhada com as informações da sua placa de vídeo e das extensões que estão ativas para ela, a lista abaixo é um **resumo** com informações da placa apenas:

```
Layers: count = 1
=====
VK_LAYER_MESA_device_select (Linux device selection layer) Vulkan version 1.1.73, layer_
↳ version 1:
Layer Extensions: count = 0
Devices: count = 2
    GPU id = 0 (AMD RADV VERDE (ACO))
    Layer-Device Extensions: count = 0

    GPU id = 1 (AMD Radeon HD 7700 Series)
    Layer-Device Extensions: count = 0

GPU0:
VkPhysicalDeviceProperties:
-----
```

(continua na próxima página)

(continuação da página anterior)

```

apiVersion      = 4202627 (1.2.131)
driverVersion   = 83894276 (0x5002004)
vendorID        = 0x1002
deviceID        = 0x683f
deviceType      = PHYSICAL_DEVICE_TYPE_DISCRETE_GPU
deviceName      = AMD RADV VERDE (ACO)

```

GPU1:

VkPhysicalDeviceProperties:

```

-----
apiVersion      = 4202655 (1.2.159)
driverVersion   = 8388775 (0x8000a7)
vendorID        = 0x1002
deviceID        = 0x683f
deviceType      = PHYSICAL_DEVICE_TYPE_DISCRETE_GPU
deviceName      = AMD Radeon HD 7700 Series

```

VkPhysicalDeviceDriverProperties:

```

-----
driverID         = DRIVER_ID_MESA_RADV
driverName       = radv
driverInfo       = Mesa 20.2.4 (ACO)
conformanceVersion = 1.2.3.0

```

Nota: Ignore o aviso **WARNING: radv is not a conformant vulkan implementation, testing use only.**

Ativando o Vulkan no Debian 10/11

O Debian exige um tratamento todo especial, por ser uma distro bem conservadora e que visa a extrema estabilidade a versão dos seus pacotes são antigos se comparados com a versão da atualidade, portanto é necessário fazer alterações significativas para que seja possível usar o driver amdgpu compatível com o vulkan.

Os procedimentos a seguir foram feitos a partir de uma instalação nova do Debian 10 (Buster), não recomendamos o procedimento no seu computador de uso diário pois você pode perder totalmente o acesso a interface gráfica, inclusive do terminal local.

Depois de terminada a instalação adicione um usuário comum e adicione-o ao grupo sudo com o comando `usermod -aG sudo nome_do_usuario` para que ele possa usar o comando sudo, encerre a sessão caso esteja logado na interface gráfica.

Pressione **CTRL+ALT+F1** e se logue como **root**, faça um backup do arquivo `/etc/apt/source.list`:

```
cp /etc/apt/source.list /etc/apt/source.list~
```

Faça `echo "" > /etc/apt/source.list` para limpar o arquivo e adicione o seguinte conteúdo:

```

deb http://ftp.br.debian.org/debian/ testing main contrib non-free
deb http://ftp.br.debian.org/debian/ testing-updates main contrib non-free
deb http://security.debian.org/ testing-security main

```

Faça o comando `apt-get update` && `apt-get upgrade` e aguarde a atualização de todos os pacotes do sistema, isso pode levar um pouco mais de meia hora. Quando todo o processo terminar faça o comando `apt dist-upgrade`, este comando vai atualizar o restante dos pacotes que não foram atualizados no processo anterior e também vai atualizar o kernel.

Agora instale os seguintes pacotes, independente de como apareça, a linha abaixo é uma linha inteira e sem quebras:

```
sudo apt-get install firmware-amd-graphics xserver-xorg-video-amdgpu
libgl1-mesa-dri libdrm-amdgpu1 firmware-linux-nonfree libgl1-mesa-dri
vulkan-tools radeontop mesa-vulkan-drivers mesa-utils libglvnd0
tuned vulkan-validationlayers mesa-ocl-icd lm-sensors
```

Crie o arquivo `/etc/modprobe.d/amdgpu.conf` com o seguinte conteúdo:

```
options radeon si_support=0
options amdgpu si_support=1
options amdgpu dpm=0
options amdgpu deep_color=1
options amdgpu dc=1
```

Nota: Dependendo da versão da sua *VGA/GPU* você precisa usar `amdgpu dc=1`, caso contrário a tela fica preta no próximo reboot, [consulte este link](#) para mais informações.

Crie o arquivo `/etc/modprobe.d/pcie-perf.conf` com o seguinte conteúdo:

```
options pcie_aspm policy=performance
```

Crie o arquivo `/etc/modprobe.d/blacklist.conf` com o seguinte conteúdo:

```
blacklist radeon
```

Quando terminar faça o comando `sudo update-grub` && `sudo update-initramfs -u` para atualizar o grub e criar um novo initramfs seguido de `systemctl reboot` para reiniciar. Rode o comando abaixo e verifique se o driver **amdgpu** está em uso:

```
lspci -vs 01:00.0 | grep driver
Kernel driver in use: amdgpu

glxinfo -B | grep "OpenGL renderer" && glxinfo -B | grep "OpenGL version"
OpenGL renderer string: AMD Radeon HD 7700 Series (VERDE, DRM 3.40.0, 5.10.0-1-amd64, LLVM_
→11.0.1)
OpenGL version string: 4.6 (Compatibility Profile) Mesa 20.3.2
```

Execute o comando `vulkaninfo` e verifique se ele não acusa qualquer erro, se tudo estiver certo aparecerá uma lista detalhada com as informações da sua placa de vídeo e das extensões que estão ativas para ela, a lista abaixo é um **resumo** com informações da placa apenas:

```
vulkaninfo | grep GPU
WARNING: radv is not a conformant vulkan implementation, testing use only.
WARNING: lavapipe is not a conformant vulkan implementation, testing use only.
GPU id = 0 (AMD RADV VERDE (ACO))
GPU id = 1 (llvmpipe (LLVM 11.0.1, 256 bits))
```

(continua na próxima página)

(continuação da página anterior)

```
GPU id = 0 (AMD RADV VERDE (ACO))
GPU id = 1 (llvmpipe (LLVM 11.0.1, 256 bits))
GPU id = 0 (AMD RADV VERDE (ACO))
GPU id = 1 (llvmpipe (LLVM 11.0.1, 256 bits))
GPU id : 0 (AMD RADV VERDE (ACO)):
GPU id : 1 (llvmpipe (LLVM 11.0.1, 256 bits)):
```

Se chegou até aqui não é preciso definir a variável **VK_ICD_FILENAMES**.

Ops, alguma coisa deu errado!

Caso a sua distribuição não configure a variável **VK_ICD_FILENAMES**, o vulkaninfo e toda a configuração feita até aqui não vai funcionar fazendo com que o teste falhe. Se for o caso, ao rodar o comando vulkaninfo deve aparecer o erro logo no início:

```
ERROR: Failed to find Vulkan Driver JSON
```

Ou pior:

```
Cannot create Vulkan instance.  
This problem is often caused by a faulty installation of the Vulkan  
driver or attempting to use a GPU that does not support Vulkan.  
ERROR at ../vulkaninfo/vulkaninfo.h:641:vkCreateInstance failed with  
ERROR_INCOMPATIBLE_DRIVER
```

Tanto no Fedora quanto no Debian os arquivos *.json devem estar instalados no diretório /usr/share/vulkan/icd.d, caso não estejam tenha certeza de ter instalado o pacote mesa-vulkan-drivers, o nome do pacote é o mesmo tanto para Fedora quanto para o Debian. Verifique a existência dos arquivos com o comando:

```
sudo find /usr/share -name *_icd.*  
/usr/share/vulkan/icd.d/intel_icd.x86_64.json  
/usr/share/vulkan/icd.d/amd_icd.x86_64.json  
/usr/share/vulkan/icd.d/radeon_icd.x86_64.json
```

Edite o arquivo /etc/profile e no final do arquivo coloque:

```
export XDG_RUNTIME_DIR=/run/user/$UID  
export VK_ICD_FILENAMES=/usr/share/vulkan/icd.d/amd_icd.x86_64.json:/usr/share/vulkan/icd.d/  
↪radeon_icd.x86_64.json:/usr/share/vulkan/icd.d/intel_icd.x86_64.json
```

A linha acima deve ser contínua, encerre a sua sessão e faça login novamente. No terminal rode o comando journalctl -b -p err e tenha **CERTEZA** que não há qualquer erro relacionado com o vulkan.

Nota: Se mesmo depois de adicionar as entradas no /etc/profile, reiniciar o computador e ainda aparecer exatamente o mesmo erro, adicione as duas linhas acima no final do seu perfil em ~/.profile, encerre a sua sessão e faça login novamente.

Tente rodar novamente o vulkaninfo e dessa vez ele deve rodar sem problemas exibindo todas as informações da sua placa de vídeo.

No caso do Linux acusar a falta de algum firmware para o **amdgpu**:

```
update-initramfs: Generating /boot/initrd.img-5.10.0-7-amd64
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_gpu_info.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_ta.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_sos.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_ta.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_asd.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_sos.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_rlc.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_mec2.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_mec.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_rlc.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_mec2.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_mec.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_me.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_pfp.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_ce.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_sdma.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_sdma.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/sienna_cichlid_mes.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navi10_mes.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_vcn.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_vcn.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_smc.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/arcturus_smc.bin for module amdgpu
W: Possible missing firmware /lib/firmware/amdgpu/navy_flounder_dmcub.bin for module amdgpu
```

É possível procurar por eles no site [PKGS](#), geralmente será preciso baixar arquivos de outra distro, descompactar e copiar para o local apropriado ou usar o site do [Umio-Yasuno](#) que mantém uma lista atualizada dos novos *firmwares* sempre que eles forem aparecendo.

Para evitar ficar copiando manualmente estes arquivos um a um, crie uma lista deles, salve a lista acima num arquivo qualquer (bin.txt) e execute o comando:

```
cat bin.txt | awk '{print $5}' | awk -F "/lib/firmware/amdgpu/" '{print $2}' > missing.txt
```

Para gerar a lista abaixo:

```
arcturus_gpu_info.bin
navy_flounder_ta.bin
navy_flounder_sos.bin
arcturus_ta.bin
arcturus_asd.bin
arcturus_sos.bin
arcturus_rlc.bin
arcturus_mec2.bin
arcturus_mec.bin
navy_flounder_rlc.bin
navy_flounder_mec2.bin
navy_flounder_mec.bin
navy_flounder_me.bin
navy_flounder_pfp.bin
navy_flounder_ce.bin
```

(continua na próxima página)

(continuação da página anterior)

```
arcturus_sdma.bin
navy_flounder_sdma.bin
sienna_cichlid_mes.bin
navi10_mes.bin
navy_flounder_vcn.bin
arcturus_vcn.bin
navy_flounder_smc.bin
arcturus_smc.bin
navy_flounder_dmcub.bin
```

Clone o repositório do site do **Umio-Yasuno** em algum lugar do seu computador com o comando:

```
git clone https://github.com/Umio-Yasuno/unofficial-amdgpu-firmware-repo.git
```

Salve a lista como `missing.txt`, copie este arquivo para dentro da pasta **amdgpu**, abra o terminal dentro desta pasta e faça o comando:

```
for firmware in $(cat missing.txt); do sudo cp "$firmware" /lib/firmware/amdgpu; done
```

Ou para os mais puritanos:

```
while read -r firmware; do sudo cp $firmware /lib/firmware/amdgpu; done < missing.txt
```

Agora atualize o seu `initramfs` com `sudo update-initramfs -u` no **Debian** ou `sudo dracut -fv` no **Fedora**.

Para casos onde o `amdgpu` trava.

Adicione estas linhas extras ao seu `/etc/modprobe.d/amdgpu.conf`:

```
options amdgpu gpu_recovery=1
options amdgpu lockup_timeout=6000
options amdgpu noretry=0
```

A primeira opção ativa a recuperação do `amdgpu`, isso resolve a questão das mensagens de erros "*amdgpu: GPU recovery disabled*" no registro de eventos. A segunda opção determina o tempo limite para que a recuperação aconteça, o padrão é 10s, o valor foi alterado para 6s. A terceira opção é necessária para o processo de recuperação.

Para mais informações consulte [amdgpu](#).

Removendo a âncora

Em geral as distros linux vem com o modo mais agressivo de economia de energia ativo, seria colocar uma âncora num carro de corrida. Isso sacrifica o desempenho do seu computador visando a economia exagerada de energia,

Instale o `tuned` com `sudo dnf install tuned` no **Fedora** ou `sudo apt-get install tuned` no **Debian**. Inicie o `tuned` com o comando:

```
systemctl start tuned
```

Faça com que ele seja sempre inicializado no boot:

```
systemctl enable tuned
```

Definimos o perfil desktop com o comando:

```
tuned-adm profile desktop
```

O perfil **desktop** fica no meio termo, salva energia quando estiver tudo calmo e acelera quando precisar. Para ver a lista dos outros perfis execute o comando `tuned-adm profile`, há o perfil `latency-performance` que elimina o gerenciamento de energia e deixa tudo no máximo ao custo de um alto consumo de energia.

Para conferir qual o perfil ativo faça:

```
tuned-adm active
Current active profile: desktop
```

Os perfis com cada configuração ficam no diretório `/usr/lib/tuned`.

Para deixar o gerenciamento de energia em modo **performance** crie o arquivo `10-amdgpu.rules` em `/etc/udev/rules.d` com o comando `sudo touch /etc/udev/rules.d/10-amdgpu.rules` e adicione estas configurações:

```
KERNEL=="card0", SUBSYSTEM=="drm", DRIVERS=="amdgpu", ATTR{device/power_dpm_force_
↳performance_level}=="high"
```

Note porém que `high` pode ser incompatível com o modelo da sua placa de vídeo, nestes casos tente `auto` ou experimente com as outras opções [disponíveis](#). Na dúvida ou incerteza, não faça esta configuração.

Salve o arquivo e execute o comando `sudo udevadm control --reload-rules` para atualizar o `udev`, em seguida rode o comando `journalctl -b -p err` e tenha certeza que não há **qualquer** erro em vermelho relacionado ao `amdgpu`, se houver verifique o arquivo `10-amdgpu.rules` e o seu conteúdo, repita o comando `udevadm control --reload-rules`. Caso o erro persista, apague o arquivo `10-amdgpu.rules` e repita o comando `udevadm control --reload-rules` novamente para eliminar as configurações, talvez haja algum problema com a versão do driver ou da compatibilidade com a sua placa de vídeo.

Há situações onde pode ocorrer o corrompimento dos gráficos na sua tela como um todo ou em partes dela, se for o seu caso troque a opção `performance` por `high` seguido do comando `udevadm control --reload-rules`, novamente, verifique com o comando `journalctl -b -p err` se não há erros do **amdgpu** em vermelho.

Execute o comando para verificar a temperatura da sua placa de vídeo:

```
sensors

amdgpu-pci-0100
Adapter:      PCI adapter
fan1:         N/A
edge:         +43.0°C (crit = +120.0°C, hyst = +90.0°C)
```

Para encerrar a configuração com chave de ouro, ative a renderização direta da placa de vídeo, edite o arquivo `/usr/share/X11/xorg.conf.d/10-amdgpu.conf` e adicione a opção `Option "DRI" "3"` como mostra o exemplo abaixo:

```
Section "OutputClass"
    Identifier "AMDgpu"
    MatchDriver "amdgpu"
    Driver "amdgpu"
    Option "DRI" "3"
EndSection
```

Rode um vídeo qualquer, pode ser do Youtube, em seguida execute o comando `radeontop` e veja se está havendo atividade enquanto o vídeo está sendo executado, tecla **c** para ativar o modo colorido. Se não houver qualquer atividade é porque há algum erro na sua configuração.

Configurando o MAME com o vulkan

Antes de prosseguir leia com atenção:

- **AS CONFIGURAÇÕES SÓ FUNCIONAM COM A VERSÃO MAIS RECENTE DO MAME!**

Elas foram testadas com a versão **0.226**, portanto as configurações valem desta versão ou versões mais recentes.

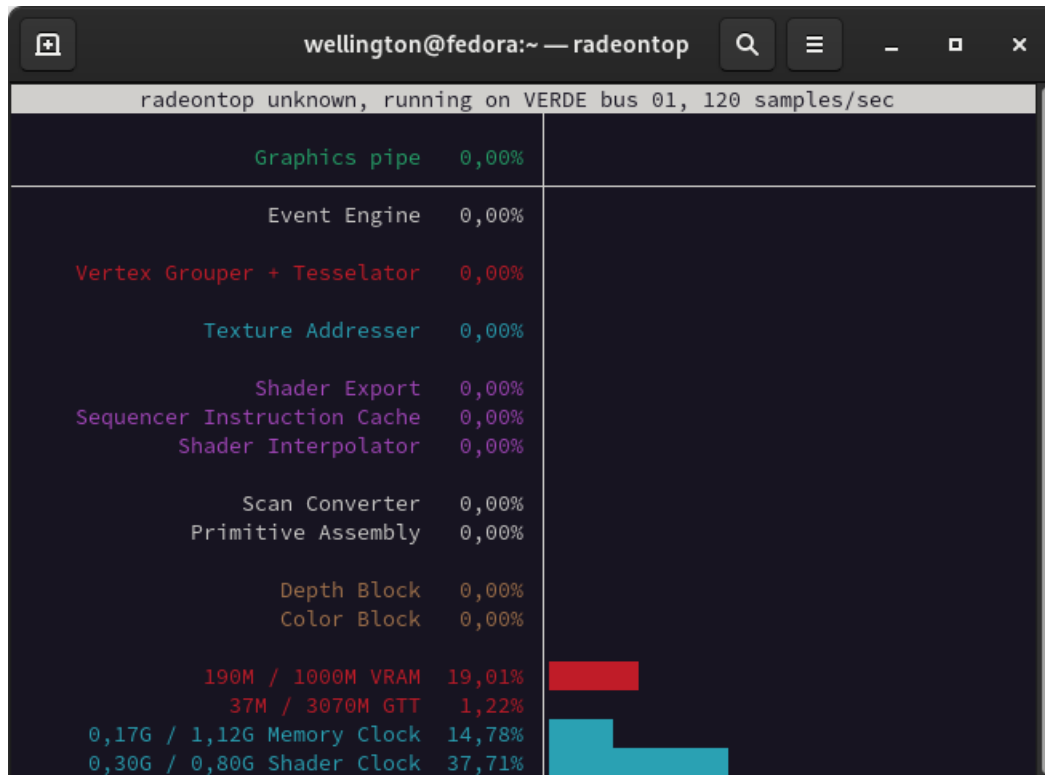
É importante que não haja conflitos de configuração, portanto, faça o backup dos seus arquivos `mame.ini`, `ui.ini` e `plugins.ini`. Crie novos arquivos com o comando `mame -cc`.

Vá até onde o seu MAME está instalado, dentro do diretório **ini** crie um arquivo `raster.ini` e edite-o com as seguintes configurações:

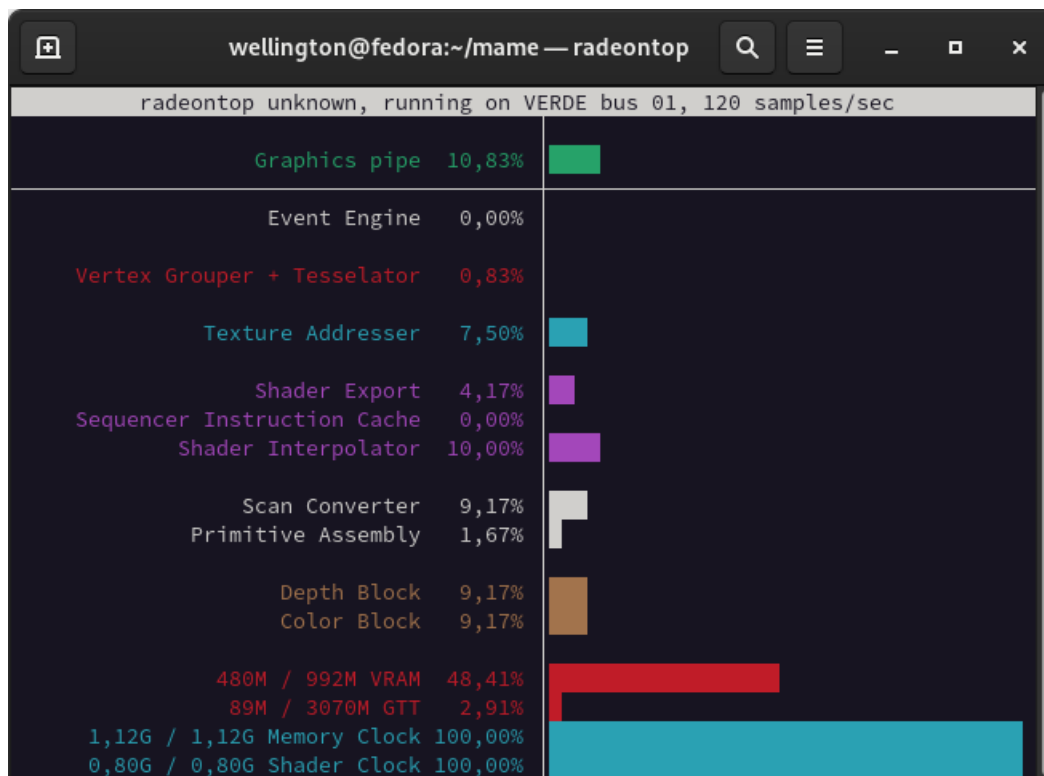
<code>rompath</code>	<code>roms;outro_caminho_completo_das_suas_roms</code>
<code># Video</code>	
<code>video</code>	<code>bgfx</code>
<code>bgfx_backend</code>	<code>vulkan</code>
<code>bgfx_screen_chains</code>	<code>crt-geom</code>
<code>window</code>	<code>1</code>

Nota: A mesma configuração serve para o **Windows**, tenha certeza de estar usando a última versão dos drivers da sua placa de vídeo.

No terminal rode o comando `radeontop`, ele deve exibir algumas estatísticas.



Em outro terminal rode o sistema `ddrmax2` por exemplo (ou qualquer outro listado no driver **ksys573**) com o comando `mame ddrmax2 -v`, além da mensagem **"WARNING: radv is not a conformant..."** não deve haver nada fora do normal, repare porém no terminal rodando o **radeontop** que as estatísticas passam a se alterar e em especial a frequência do **Memory Clock** e do **Shader Clock** que sobem para 100%, repare que também haverá um aumento do consumo da memória de vídeo **VRAM**.



Na janela do MAME pressione Esc para encerrar a emulação, se tudo estiver corretamente configurado como demonstramos aqui, a frequência do **Memory Clock** e do **Shader Clock** devem recuar logo após o encerramento do MAME, caso não recue aguarde aproximadamente uns 10 minutos e se ainda assim continuarem em 100% significa que você está utilizando algum perfil de alto desempenho que desativou o gerenciamento de energia, não há problema deixar a sua placa de vídeo rodando no máximo desde que você saiba **EXATAMENTE** o que está fazendo.

7.12.13 Excluindo arquivos NVRAM (script)

Algumas vezes é preciso excluir o diretório **NVRAM** de um determinado sistema durante a depuração ou até mesmo antes de um *-record* e antes de iniciar um *-playback* por motivos já explicados nestes capítulos. Contudo, caso a exclusão destes diretórios seja constante, eu crie dois scripts que me ajudam na tarefa de eliminar estes diretórios. No **Windows**, crie o arquivo `limpa.bat` dentro da pasta principal do MAME (ou onde a pasta **nvr** se encontra) com o conteúdo abaixo:

```
@echo off

rem Verifica se o nome da ROM foi informada junto com o comando
if "%1" == "" (
    echo use limpa nome_da_rom
    exit /b 1
)

set rom_name=%1

rem Exclui os diretórios que batem com o nome da ROM.
if exist "nvr\%rom_name%" (
    rmdir /s /q "nvr\%rom_name%"
)

for /L %i in (0,1,16) do (
    rem Verifica a existência de outros diretórios seguido de número antes de tentar excluí-
    los.
    if exist "nvr\%rom_name%_%i" (
        rmdir /s /q "nvr\%rom_name%_%i"
    )
)
```

Para o **Linux** e **macOS** crie o arquivo `limpa` dentro da pasta principal do MAME (ou onde a pasta **nvr**am se encontra) com o conteúdo abaixo:

```
#!/bin/bash

# Verifica se o nome da ROM foi informada junto com o comando
if [ "$1" == "" ]; then
    echo "use ./limpa nome_da_rom"
    exit 1
fi

rom_name="$1"

# Exclui os diretórios que batem com o nome da ROM
if [ -d "nvr$rom_name" ]; then
    rm -rf "nvr$rom_name"
fi

for i in {0..16}; do
    # Verifica a existência de outros diretórios seguido de número antes de tentar excluí-los.
    if [ -d "nvr/${rom_name}_$i" ]; then
        rm -rf "nvr/${rom_name}_$i"
    fi
done
```

Ambos precisam rodar através do prompt de comando ou do terminal, no caso do *shell script* para Linux/macOS, antes que ele possa ser executado; também é preciso fazer `chmod +x limpa`. O uso é simples, no Windows, basta executar o script seguido do nome da ROM, por exemplo:

```
limpa sf2
```

No Linux/macOS faça:

```
./limpa sf2
```

Isso excluirá a pasta **sf2** dentro de **nvr**am, limpando todas as definições, pontuações, configurações e tudo mais que o respectivo sistema registrar em sua memória nvr. Note que em alguns casos como os sistemas **Neo Geo** por exemplo, segundo a lista de BIOS compatíveis listadas no [código-fonte do MAME](#), dependendo da BIOS selecionada, o nome da pasta será seguida por um número, então se usarmos a BIOS "*Unibios*", o nome da pasta termina com **_16**, ou seja, `kof94_16` e assim por diante, neste caso, o script os excluirá também.

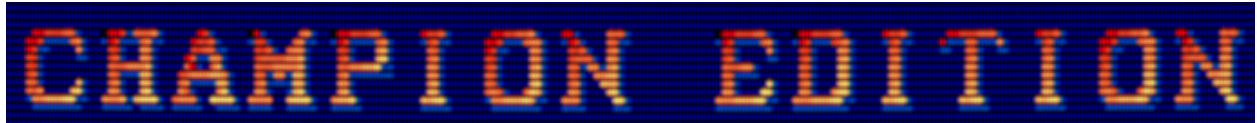
7.12.14 Removendo a suavização dos efeitos crt-geom/crt-geom-deluxe

A ideia destes efeitos é simular uma tela CRT com todas as suas qualidades e defeitos, no que tange a questão de defeito, a tela é suavizada de tal maneira que parece um embaçamento na tela inteira deixando uma imagem mais "soft", porém, é uma questão de gosto individual. Algumas pessoas que sofrem com miopia por exemplo, utilizam óculos para justamente corrigir a visão embaçada, estes efeitos borram a tela toda causando desconforto nas pessoas que já sofrem deste problema.

Para aqueles que não sabem do que estamos falando, inicie o MAME com o comando abaixo:

```
mame sf2ce -video bgfx -bgfx_backend opengl -bgfx_screen_chains crt-geom
```

Repare que a imagem aparece num formato de tela CRT (com curvatura e linhas de escaneamento) levemente embaçada, usaremos como referência o recorte abaixo:



O primeiro efeito a ser desligado é a máscara², feche/encerre o MAME, dentro da pasta do MAME encontre a pasta chamada **bgfx** e edite o arquivo `chains\crt-geom.json`, no final dele, altere a linha:

```
"sampler": "mask_texture", "texture": "bgfx/chains/crt-geom/aperture_1_2_bgr.png"
```

Para:

```
"sampler": "mask_texture", "texture": "bgfx/chains/crt-geom/none.png"
```

Salve o arquivo como `crt-geom-edit.json` ou qualquer outro nome, assim ao atualizar o MAME você não perde as suas alterações, ao iniciar o MAME novamente com os parâmetros abaixo:

```
mame sf2ce -video bgfx -bgfx_backend opengl -bgfx_screen_chains crt-geom-edit
```

Repare que agora a imagem mantém a curvatura e as linhas de escaneamento, um pouco mais clara e sem o efeito da máscara da tela.



O último passo agora é remover esse efeito de embaçamento da tela, ainda com o arquivo `chains\crt-geom.json` aberto, altere a linha:

```
"text": "Horizontal interpolation",  
"default": 2,
```

Para:

```
"text": "Horizontal interpolation",  
"default": 0,
```

² Para mais detalhes, acesse <http://www.fazendovideo.com.br/infotec/crt.html>

Note que mantivemos o efeito da tela CRT com as linhas de escaneamento, sem os efeitos de embaçamento.



Com a tela com uma aparência mais limpa e sem os "defeitos" da tela CRT, você pode alterar os outros valores do `chains\crt-geom-edit.json` para fazer um ajuste fino do efeito ou experimentar os outros disponíveis dentro da pasta `bgfx\chains`. Para quem achar a tela um pouco escura demais, altere o valor abaixo:

```
"text": "Gamma of simulated CRT",  
  "default": 2.4,
```

Para:

```
"text": "Gamma of simulated CRT",  
  "default": 2.0,
```

Nota: Em alguns [documentos técnicos](#), o valor indicado do gamma para monitores é **2.2** no Windows e **1.8** no macOS.

Os arquivos da máscara como o `aperture_1_2_bgr.png` estão dentro da pasta `artwork\bgfx\chains\crt-geom`. Apesar de particularmente preferir o efeito `none`, há efeitos mais sutis como os efeitos `delta_`, eles dão uma aparência bacana (para quem gosta) sem borrar a tela toda.

7.12.15 Listando as diferentes BIOS do Neo Geo

Pode ser que por diferentes motivos, você queira usar uma BIOS diferente do padrão. Estas BIOS estão listadas no [código-fonte do MAME](#), caso esteja usando um Linux e tenha o código-fonte do MAME no computador, entre na pasta onde o código-fonte está e faça o comando:

```
find . -name neogeo.cpp
./src/mame/neogeo/neogeo.cpp
```

Agora faça o comando abaixo para extrair apenas a lista das BIOS e vamos redirecionar esta saída para o arquivo bios.txt:

```
cat src/mame/neogeo/neogeo.cpp|grep -i --color=auto rom_system_bios > bios.txt
```

No arquivo nós teremos a lista abaixo:

```
ROM_SYSTEM_BIOS( x+ 0, "unibios40", "Universe BIOS (Hack, Ver. 4.0)" ) \
ROM_SYSTEM_BIOS( x+ 1, "unibios33", "Universe BIOS (Hack, Ver. 3.3)" ) \
ROM_SYSTEM_BIOS( x+ 2, "unibios32", "Universe BIOS (Hack, Ver. 3.2)" ) \
ROM_SYSTEM_BIOS( x+ 3, "unibios31", "Universe BIOS (Hack, Ver. 3.1)" ) \
ROM_SYSTEM_BIOS( x+ 4, "unibios30", "Universe BIOS (Hack, Ver. 3.0)" ) \
ROM_SYSTEM_BIOS( x+ 5, "unibios23", "Universe BIOS (Hack, Ver. 2.3)" ) \
ROM_SYSTEM_BIOS( x+ 6, "unibios23o", "Universe BIOS (Hack, Ver. 2.3, older?)" ) \
ROM_SYSTEM_BIOS( x+ 7, "unibios22", "Universe BIOS (Hack, Ver. 2.2)" ) \
ROM_SYSTEM_BIOS( x+ 8, "unibios21", "Universe BIOS (Hack, Ver. 2.1)" ) \
ROM_SYSTEM_BIOS( x+ 9, "unibios20", "Universe BIOS (Hack, Ver. 2.0)" ) \
ROM_SYSTEM_BIOS( x+10, "unibios13", "Universe BIOS (Hack, Ver. 1.3)" ) \
ROM_SYSTEM_BIOS( x+11, "unibios12", "Universe BIOS (Hack, Ver. 1.2)" ) \
ROM_SYSTEM_BIOS( x+12, "unibios12o", "Universe BIOS (Hack, Ver. 1.2, older)" ) \
ROM_SYSTEM_BIOS( x+13, "unibios11", "Universe BIOS (Hack, Ver. 1.1)" ) \
ROM_SYSTEM_BIOS( x+14, "unibios10", "Universe BIOS (Hack, Ver. 1.0)" ) \
ROM_SYSTEM_BIOS( 0, "euro", "Europe MVS (Ver. 2)" ) \
ROM_SYSTEM_BIOS( 1, "euro-s1", "Europe MVS (Ver. 1)" ) \
ROM_SYSTEM_BIOS( 2, "asia-mv1c", "Asia NEO-MVH MV1C" ) \
ROM_SYSTEM_BIOS( 3, "asia-mv1b", "Asia MV1B" ) \
ROM_SYSTEM_BIOS( 4, "us", "US MVS (Ver. 2?)" ) \
ROM_SYSTEM_BIOS( 5, "us-e", "US MVS (Ver. 1)" ) \
ROM_SYSTEM_BIOS( 6, "us-v2", "US MVS (4 slot, Ver 2)" ) \
ROM_SYSTEM_BIOS( 7, "us-u4", "US MVS (U4)" ) \
ROM_SYSTEM_BIOS( 8, "us-u3", "US MVS (U3)" ) \
ROM_SYSTEM_BIOS( 9, "japan", "Japan MVS (Ver. 3)" ) \
ROM_SYSTEM_BIOS( 10, "japan-s2", "Japan MVS (Ver. 2)" ) \
ROM_SYSTEM_BIOS( 11, "japan-s1", "Japan MVS (Ver. 1)" ) \
ROM_SYSTEM_BIOS( 12, "japan-mv1b", "Japan MV1B" ) \
ROM_SYSTEM_BIOS( 13, "japan-j3a", "Japan MVS (J3, alt)" ) \
ROM_SYSTEM_BIOS( 14, "japan-mv1c", "Japan NEO-MVH MV1C" ) \
ROM_SYSTEM_BIOS( 15, "japan-hotel", "Custom Japanese Hotel" ) \
ROM_SYSTEM_BIOS( 0, "asia", "Asia AES" )
ROM_SYSTEM_BIOS( 1, "japan", "Japan AES" )
ROM_SYSTEM_BIOS( 2, "devel", "Development System ROM" )
ROM_SYSTEM_BIOS( 0, "asia", "NEO-MVH MV1C" )
ROM_SYSTEM_BIOS( 1, "japan", "Japan MVS (J3)" )
ROM_SYSTEM_BIOS( 0, "asia-sp1", "Asia MV1B 263" )
```

Agora podemos por exemplo, listar apenas os nomes das BIOS com o comando:

```
cat bios.txt | awk -F ', ' '{print $2}' | awk -F ' "' '{print $1}'
unibios40
unibios33
unibios32
unibios31
...
```

Podemos listar apenas as descrições:

```
cat bios.txt | awk -F ', ' '{print $3}' | awk -F ' "' '{print $1}'
Universe BIOS (Hack, Ver. 4.0)
Universe BIOS (Hack, Ver. 3.3)
Universe BIOS (Hack, Ver. 3.2)
Universe BIOS (Hack, Ver. 3.1)
...
```

Usando o script abaixo:

```
#!/bin/bash

# Define o nosso arquivo de entrada e o que nos interessa
input_file='bios.txt'
regex='^ROM_SYSTEM_BIOS\[^\]*"([^\]*)"(.*)"([^\]*)"'
```

```
# Usamos o grep para extrair o que desejamos e as organizamos lado a lado
while read -r line; do
    if [[ $line =~ $regex ]]; then
        value1="${BASH_REMATCH[1]}"
        value2="${BASH_REMATCH[2]}"
        echo "$value1 - $value2"
    fi
done < "$input_file"
```

Nós podemos gerar uma lista com o nome da BIOS e a sua respectiva descrição lado a lado, salve o script como filtra_bios, torne-o executável com `chmod +x filtra_bios` e rode-o com `./filtra_bios` para obter o resultado:

```
unibios40 - Universe BIOS (Hack, Ver. 4.0)
unibios33 - Universe BIOS (Hack, Ver. 3.3)
unibios32 - Universe BIOS (Hack, Ver. 3.2)
unibios31 - Universe BIOS (Hack, Ver. 3.1)
...
```

Caso queira a lista em ordem alfabética rode como `./filtra_bios | sort -d`:

```
asia - Asia AES
asia-mv1b - Asia MV1B
asia-mv1c - Asia NEO-MVH MV1C
asia - NEO-MVH MV1C
asia-sp1 - Asia MV1B 263
devel - Development System ROM
euro - Europe MVS (Ver. 2)
```

(continua na próxima página)

(continuação da página anterior)

```

euro-s1 - Europe MVS (Ver. 1)
japan-hotel - Custom Japanese Hotel
japan-j3a - Japan MVS (J3, alt)
japan - Japan AES
japan - Japan MVS (J3)
japan - Japan MVS (Ver. 3)
japan-mv1b - Japan MV1B
japan-mv1c - Japan NEO-MVH MV1C
japan-s1 - Japan MVS (Ver. 1)
japan-s2 - Japan MVS (Ver. 2)
unibios10 - Universe BIOS (Hack, Ver. 1.0)
unibios11 - Universe BIOS (Hack, Ver. 1.1)
unibios12o - Universe BIOS (Hack, Ver. 1.2, older)
unibios12 - Universe BIOS (Hack, Ver. 1.2)
unibios13 - Universe BIOS (Hack, Ver. 1.3)
unibios20 - Universe BIOS (Hack, Ver. 2.0)
unibios21 - Universe BIOS (Hack, Ver. 2.1)
unibios22 - Universe BIOS (Hack, Ver. 2.2)
unibios23o - Universe BIOS (Hack, Ver. 2.3, older?)
unibios23 - Universe BIOS (Hack, Ver. 2.3)
unibios30 - Universe BIOS (Hack, Ver. 3.0)
unibios31 - Universe BIOS (Hack, Ver. 3.1)
unibios32 - Universe BIOS (Hack, Ver. 3.2)
unibios33 - Universe BIOS (Hack, Ver. 3.3)
unibios40 - Universe BIOS (Hack, Ver. 4.0)
us-e - US MVS (Ver. 1)
us-u3 - US MVS (U3)
us-u4 - US MVS (U4)
us - US MVS (Ver. 2?)
us-v2 - US MVS (4 slot, Ver 2)

```

Uma outra maneira de listar apenas os nomes dessas BIOS com o MAME, é rodar o comando abaixo no terminal ou no prompt de comando:

```

mame kof94 -lx | grep -i "bios=" | awk '{print $4}' | awk -F '"' '{print $2}' | sort -d
asia-mv1b
asia-mv1c
euro
euro-s1
...

```

No Windows também é possível fazer o mesmo com o comando abaixo:

```

mame -lx kof94|findstr bios=
euro
euro-s1
asia-mv1c
asia-mv1b
...

```

A partir da versão 0.252, o MAME agora tem a opção `-listbios` para listar a BIOS de qualquer sistema compatível.

7.12.16 Criando uma lista com todas as BIOS compatíveis usadas pelo MAME

Agora com a opção `-listbios` nós podemos criar um arquivo texto com uma lista de todas as BIOS compatíveis com o MAME. Crie a lista com o comando:

```
mame -listbios > lista_bios.txt
```

Usando um terminal Linux, macOS ou até mesmo o ambiente MinGW/MSYS2 no Windows, faça o comando:

```
awk '!/No BIOSes available/' lista_bios.txt | sed '/^$/d' | sed -e '/ BIOSes available for /  
↵i\\' | sed '1{/^$/d}' > bios.txt
```

O primeiro comando exclui as linhas listadas como No BIOSes available, o segundo comando `sed '/^$/d'`, elimina todos os espaços em branco. Já o terceiro, `sed -e '/ BIOSes available for /i\\'` cria uma linha em branco antes das linhas que tiverem BIOSes available for e por último, o comando `sed '1{/^$/d}'` elimina a primeira linha em branco da nossa lista.

O resultado final fica assim:

```
41 BIOSes available for driver "100lions".  
Name:      Description:  
au-nsw1    "Aristocrat MK6 Base (24013001, NSW/ACT)"  
au-nsw2    "Aristocrat MK6 Base (21012901, NSW/ACT)"  
...  
  
25 BIOSes available for driver "at".  
Name:      Description:  
at         "PC 286"  
ami206     "AMI C 206.1"  
amiht21    "AMI HT 21.1"
```

Caso você tenha o `enscript` e o `ghostscript` instalado no seu sistema, execute os comandos abaixo para transformar o arquivo texto em PDF:

```
enscript -p bios.ps bios.txt && ps2pdf bios.ps
```

Baixe [aqui](#) os arquivos bios.txt, bios.ps e bios.pdf.

7.12.17 Listando e iniciando diferentes sistemas MSX

Dentro da pasta com o código-fonte do MAME os principais sistemas MSX podem ser encontrados dentro do diretório `src/mame/msx`, mais especificamente o meu foco está dentro dos arquivos `src/mame/msx/msx1.cpp` e `src/mame/msx/msx2.cpp`. Todo o diretório `src/mame/msx` também se encontra no [repositório oficial do MAME](#)

Para listar todos os sistemas **MSX 1** do arquivo-fonte `src/mame/msx/msx1.cpp` nós fazemos assim:

```
cat src/mame/msx/msx1.cpp | grep -i rom_start | sed 's/ROM_START(\(.*\))/\1/' | sort -d
ax150
canonv10
canonv20
canonv20e
...
```

Para o **MSX 2** é a mesma coisa, apenas alteramos a versão:

```
cat src/mame/msx/msx2.cpp | grep -i rom_start | sed 's/ROM_START(\(.*\))/\1/' | sort -d
ax350
ax350ii
ax350iif
ax370
...
```

Outra maneira de fazer o mesmo usando o próprio MAME é fazer o comando:

```
mame ax150 -lb
Source file:      Name:      Parent:
msx/msx1.cpp      ax150
msx/msx1.cpp      ax170
msx/msx1.cpp      ax230
msx/msx1.cpp      canonv10      canonv20
...
```

Para MSX 2:

```
mame ax350 -lb
Source file:      Name:      Parent:
msx/msx2.cpp      ax350      ax350ii
msx/msx2.cpp      ax350ii
msx/msx2.cpp      ax350iif      ax350ii
msx/msx2.cpp      ax370
...
```

Essas dicas também podem ser utilizadas com outros sistemas.

7.12.18 Filtrando o nome das ROMs a partir do arquivo XML

No Linux/macOS ou no Windows com ferramentas [MinGW/MSYS2](#) já disponibilizada pelo MAMEDEV, faça o comando abaixo para gerar o arquivo `mame.xml`:

```
mame -lx > mame.xml
```

Para listar as ROMs do Neo Geo por exemplo:

```
grep 'romof="neogeo"' mame.xml | sed -rn 's/. * name="([a-z0-9]+)" .*/\1/p'
2020bb
3countb
alpham2
androdun
aodk
...
```

Para as ROMs do CPS2:

```
grep 'sourcefile="capcom/cps2.cpp"' mame.xml | sed -rn 's/. * name="([a-z0-9]+)" .*/\1/p'
1944
1944d
1944j
1944u
...
```

É possível fazer o mesmo usando o código-fonte, exemplo:

```
cat src/mame/capcom/cps2.cpp | grep "GAME( " | awk '{print $3}' | sed 's/,,$// ' | sort -d
1944
1944d
1944j
1944u
...
```

7.12.19 Criando uma lista de jogos

Para criar uma lista de jogos com o nome da ROM e a sua respectiva descrição, faça o comando:

```
mame -ll > gamelist
```

Com a lista em mãos, nós pegamos essas informações, separamos elas com vírgulas e depois salvamos como `gamelist.csv`:

```
cat gamelist | awk '{$1=$1}1' OFS="," > gamelist.csv
```

Abra o arquivo `gamelist.csv` no [LibreOffice](#), no [Office da Microsoft](#) ou mesmo no [Google Docs](#) e aceite o padrão, você deverá ter as ROMs do lado esquerdo e a sua descrição do lado direito.

Fonte: [u/cd4053b](#).

7.12.20 Correção da proporção da tela para o SNES

Na época que este sistema foi lançado o padrão das TVs do mundo inteiro tinha uma proporção de tela (*Display Aspect Ratio* ou **DAR**) de **4:3** com uma resolução comum de 640x480 para **PAL-M** (Brasil) e **NTSC** (EUA e outros países).

O SNES gerava diferentes resoluções, entre 256x224 e 512x448, sendo que a maioria ficava em 256x224 PAL-M/NTSC e 256x240 PAL (Europa e outros países 50 Hz). Assim a resolução gerada tinha um **SAR** (Source/Storage/Sample Display Ratio) de **8:7**, um **PAR** (Pixel Aspect Ratio) de **1:1** (quadrado ou "*square*") que era literalmente esticada na TV para um **DAR 4:3** (não quadrado ou "*non-square*").

Os desenvolvedores criavam os seus jogos com tudo isso em mente para que quando o jogo aparecesse na TV, a proporção dos gráficos seria corrigida pela própria TV, por isso que a regra geral era um **DAR 4:3** para todos os jogos (inclusive os de arcade). No entanto, muitos jogos parecem ter saído fora deste padrão na época, e hoje com a emulação, algumas pessoas acabaram notando diferenças. Objetos que deveriam ter uma forma redonda acabavam ficando com um formato ovalado, o mesmo para quadrados que ficavam retangulares na proporção 4:3, veja este [exemplo](#) (em Inglês).

Então, qual é a proporção correta?

Sem dúvida é **4:3**! Porém ainda assim, a imagem aparece distorcida nas nossas telas.

Este arquivo serve para te dar uma opção, você vai poder escolher entre o padrão 4:3, 8:7 ou o formato "*pixel perfect*" que nada mais é do que usar o SAR como DAR. Baixe o arquivo [snes.zip](#) e coloque na pasta **artwork** do MAME (não é preciso descompactar). Inicie um jogo qualquer do snes, pressione Tab, vá em *Opções do vídeo > Tela #0* e escolha uma das novas opções disponíveis:

- *Aspect Ratio Fix NTSC/PAL 240 (4:3)*
- *Aspect Ratio Fix NTSC (Nintendo 8:7)*
- *Aspect Ratio Fix PAL (Nintendo 8:7)*
- *Pixel Perfect NTSC (256x224)*
- *Pixel Perfect PAL (256x240)*

Ao escolher uma, a sua preferencia ficará salva no arquivo `cfg\snes.cfg`. Caso queira redefinir a configuração, basta apagar este arquivo.

7.12.21 Usando o HLSL no Windows 8/10/11 sem precisar instalar o DirectX 9

Para que os *efeitos HLSL funcionem no Windows* é preciso definir a opção de *video* como `d3d` e para que esta opção funcione, é preciso instalar o [DirectX End-User Runtime](#) que é compatível até o Windows 7, segundo a página da própria Microsoft. O problema é que Windows 10 e mais recente não são compatíveis com o DirectX 9 e se instalar, isso pode causar problemas com o DirectShow e outros sistemas críticos do Windows, inclusive causando travamentos e outras coisas desagradáveis.

Para resolver este empecilho, basta descompactar alguns arquivos dentro da pasta do MAME (o mesmo lugar onde o executável do MAME se encontra), baixe o [DirectX End-User Runtime](#), abra o arquivo com o 7-zip, extraia apenas os arquivos que começam com `Jun2010_*` numa pasta vazia qualquer no seu desktop, abra um prompt de comando onde estes arquivos foram extraídos e rode o comando abaixo:

```
mkdir dlls
expand *_x64.cab -F:*.dll -R dlls
```

O comando extrairá os seguintes arquivos dentro da pasta **dlls**:

- D3DCompiler_43.dll
- d3dcsx_43.dll
- d3dx10_43.dll
- d3dx11_43.dll
- d3dx9_43.dll
- xactengine3_7.dll
- XAPOFX1_5.dll
- XAudio2_7.dll

Copie todos estes arquivos para a pasta do MAME, agora a opção *-video d3d* deverá funcionar sem maiores problemas.

Nota: Caso queira eliminar o efeito de "grelha" ("*shadow mask*") da tela, use a opção *-noshadow_mask_alpha* na linha de comando ou salve a opção *shadow_mask_alpha 0* em algum *.ini* específico.

7.12.22 Convertendo alguns arquivos .CHD antigos para o novo formato createdvd

A partir da versão 0.255 do MAME através do [commit 8f05076](#), o MAME passa a poder usar o formato DVD CHD, assim sendo, será preciso converter o formato de alguns sistemas como popn5, popn6, popn7, popn8, popnam e alguns outros para o novo formato DVD CHD.

Para fazer a conversão é simples, baixe a versão 0.255 do MAME do [site oficial](#) e use o programa *chdman* que já vem com ele e rode o comando abaixo para converter o CHD do popn5 por exemplo:

```
chdman extractraw -i a04jaa02.chd -o a04jaa02.iso
```

Exclua ou renomeie o arquivo *a04jaa02.chd* para algo como *a04jaa02.chd.old*, geralmente quando o sistema acompanha mais de um CHD, basta converter **o maior arquivo da pasta**, assim sendo no caso do popn5, basta converter o arquivo *a04jaa02.chd*. Faça o comando abaixo para criar um novo CHD no novo formato:

```
chdman createdvd -i a04jaa02.iso -o a04jaa02.chd
```

Aguarde alguns minutos até a conclusão da conversão. O antigo arquivo *a04jaa02.chd* tinha as seguintes características:

```
chdman info -i a04jaa02.chd
chdman - MAME Compressed Hunks of Data (CHD) manager 0.255 (mame0255)
Input file:  a04jaa02.chd
File Version: 5
Logical size: 2,686,978,048 bytes
Hunk Size:   4,096 bytes
Total Hunks: 656,001
Unit Size:   512 bytes
Total Units: 5,248,004
Compression: lzma (LZMA), zlib (Deflate), huff (Huffman), flac (FLAC)
CHD size:    1,152,280,500 bytes
```

(continua na próxima página)

(continuação da página anterior)

```
Ratio:          42.9%
SHA1:          49a017dde76f84829f6e99a678524c40665c3bfd
Data SHA1:     568c4dcb15f3c1bb295cd30fb79b8ae1330667d5
Metadata:      Tag='GDDD'  Index=0  Length=36 bytes
               CYLS:1312001,HEADS:2,SECS:2,BPS:512.
```

Ao final da conversão o novo arquivo passa a ter as seguintes características:

```
chdman info -i a04jaa02.chd
chdman - MAME Compressed Hunks of Data (CHD) manager 0.255 (mame0255)
Input file:   a04jaa02.chd
File Version: 5
Logical size: 2,686,978,048 bytes
Hunk Size:    2,048 bytes
Total Hunks:  1,312,001
Unit Size:    2,048 bytes
Total Units:  1,312,001
Compression:  lzma (LZMA), zlib (Deflate), huff (Huffman), flac (FLAC)
CHD size:     1,147,285,394 bytes
Ratio:        42.7%
SHA1:         058167a6ac910183a701920021cfbc0933428e97
Data SHA1:    568c4dcb15f3c1bb295cd30fb79b8ae1330667d5
Metadata:     Tag='DVD '  Index=0  Length=1 bytes
```

Repare que em **Tag** em vez de GDDD agora temos DVD.

Aviso: Antes de prosseguir, **FAÇA UM BACKUP DO SEU ARQUIVO CHD ANTES DE USAR ESTES SCRIPTS CASO VOCÊ COLECIONE VERSÕES ANTIGAS DELES!**

Visando facilitar esse trabalho, eu criei um script chamado `convert_chd_dvd.bat` para Windows (há também uma versão para Linux/macOS), no Windows basta copiar o script para dentro da pasta do CHD, abrir num editor de texto e definir o `CHDMAN_HOME` (este é o caminho onde se encontra o arquivo **chdman.exe** mais recente) e salvar, depois basta arrastar o arquivo precisa ser convertido em cima do script, se for o caso do `a04jaa02.chd` basta arrastar e soltar o arquivo em cima do script `convert_chd_dvd.bat` que ele fará todo o trabalho sujo deixando apenas o arquivo convertido e **excluindo** os arquivos antigos (incluindo o arquivo ISO usado na conversão).

Script para Windows

```
REM Criado por Wellington Terumi Uemura 08/05/2023
REM CC by 4.0

@echo off
setlocal
mode con cp select=850
cls

set CHDMAN_HOME=

if "%CHDMAN_HOME%" == "" (
    echo.
    echo O CHDMAN_HOME não foi configurado!
    echo.
    echo Encerrando...
    pause
    goto :end
)

if exist %CHDMAN_HOME%\chdman.exe (
    echo.
    set chdman_cmd=%CHDMAN_HOME%\chdman.exe
    goto :command
) else (
    echo Precisamos do chdman.exe para funcionar!
    echo.
    pause
    echo Encerrando...
    goto :end
)

:command
    for %%A in (*) do %chdman_cmd% extractraw -i %%A -o %%~nA.iso && del %%A
    echo.
    for %%A in (*) do %chdman_cmd% createdvd -i %%~nA.iso -o %%~nA.chd
    goto :clean

:clean
    cd %~dp0
    for %%i in (*.iso) do del %%i

:end
endlocal
exit /B
```

Nesta versão para Linux/macOS crie o arquivo `convert_chd_dvd`, faça o comando `chmod +x` para torná-lo executável e cole o script abaixo nele:

```
#!/bin/bash
# Criado por Wellington Terumi Uemura 08/05/2023
# CC by 4.0

export CHDMAN_HOME=~/.mame/tools

if [ -z "$CHDMAN_HOME" ]; then
    echo "CHDMAN_HOME is not set!"
    echo "Closing..."
    read -n 1 -s -r -p "Press any key to continue"
    exit 1
fi

if [ ! -f "$CHDMAN_HOME/chdman" ]; then
    echo "We need chdman to work!"
    echo "Closing..."
    read -n 1 -s -r -p "Press any key to continue"
    exit 1
fi

for file in "$@"; do
    "$CHDMAN_HOME/chdman" extractraw -i "$file" -o "${(basename "$file" .*)}.iso" && rm "$file"
done

for iso_file in *.iso; do
    "$CHDMAN_HOME/chdman" createdvd -i "$iso_file" -o "${iso_file%.*}"
done

rm *.iso
```

Na versão atual do Gnome que eu utilizo, ele não permite "arrastar e soltar" em cima do arquivo como no Windows. Então como no exemplo anterior, defina o seu `CHDMAN_HOME` e decida se vai copiar o script para o seu PATH padrão (`/usr/bin` por exemplo) ou se vai copiá-lo nas pastas que ele será utilizado e rode-o como um comando no seu terminal, exemplo:

```
convert_chd_dvd a04jaa02.chd
```

Assim como na versão do Windows, ele fará todo o trabalho para você deixando apenas o arquivo convertido no lugar e excluindo o antigo.

7.12.23 Convertendo um CD-ROM para CHD com o chdman

Uma das grandes vantagens do chdman é o seu poder de compressão de dados, com ele é possível reduzir o tamanho de uma imagem de uma mídia de CD-ROM em quase **50%** ou mais dependendo da mídia. O chdman aceita uma grande variedade de formatos como .cue + .bin, .toc + bin, .iso, .nrg, .gdi, .cdr, .img e outros.

Para realizar tal tarefa o comando básico é:

```
chdman createdcd -i imagem_ou_cue -o arquivo.chd
```

Visando facilitar a vida de quem precisar converter uma certa quantidade de imagens de uma vez, foi criado o script batch (Windows) abaixo, basta arrastar os arquivos .cue e .toc em cima do script para iniciar todo o processo, neste caso em específico, não é necessário arrastar junto o arquivo .bin. Nos outros formatos de imagem sem .cue e .toc como .iso por exemplo, basta arrastar estes arquivos direto em cima do script. Salve como **cd-to-chd.bat** e coloque-o na mesma pasta que o programa chdman.exe.

```
REM Criado por Wellington Terumi Uemura 31/03/2024
REM CC by 4.0
REM Arraste o arquivo .cue em cima deste batch script.

@echo off

setlocal
mode con cp select=850

cls

Title Converte imagem de CD para CHD

if exist %~dp0\chdman.exe (

set CHDMAN=%~dp0\chdman.exe
set CHDMAN_CONFIG=createcd -np 4

) else (
    echo Não encontrei o chdman!
    echo Baixe de https://github.com/mamedev/mame/releases
    echo.
    pause
    echo Encerrando...
    goto :end
)

if not exist chd (
goto :dir

) else (
goto :command
)

:dir
mkdir chd
goto :command
```

(continua na próxima página)

(continuação da página anterior)

```

:command
for %%A in (*) do %CHDMAN% %CHDMAN_CONFIG% -i %%A -o "chd\%%~nA.chd"

:end
endlocal
exit /B

```

Na variável CHDMAN_CONFIG definimos -np 4 (quantidade de processadores), isso serve para limitar a quantidade de processadores já que a compressão de dados consome uma grande quantidade de recursos do processador. Caso tenha um processador moderno ou não tenha problema com limitação de recursos, remova a opção -np 4.

7.12.24 Gerando um arquivo de registro de um arquivo CHD

O batch script abaixo gera um arquivo nome_do_arquivo_chd.log no mesmo diretório do arquivo .chd. Salve como **chd-log.bat** e coloque-o na mesma pasta que o programa chdman.exe e arraste um arquivo .chd em cima dele.

```

REM Criado por Wellington Terumi Uemura 31/03/2024
REM CC by 4.0
REM Arraste o arquivo .chd em cima deste batch script.

@echo off

setlocal
mode con cp select=850

cls

Title Gera arquivo de informação do CHD

if exist %~dp0\chdman.exe (

set CHDMAN=%~dp0\chdman.exe
set CHDMAN_CONFIG=info -v

) else (
    echo Não encontrei o chdman!
    echo Baixe de https://github.com/mamedev/mame/releases
    echo.
    pause
    echo Encerrando...
    goto :end
)

for %%A in (*) do %CHDMAN% %CHDMAN_CONFIG% -i %%A > "%~nA.log" 2>&1

:end
endlocal
exit /B

```

7.12.25 Extraíndo uma lista de sistemas usando a interface

Gerar uma lista de sistema com base no arquivo de código fonte usando a interface é bem simples, inicie o MAME e siga os passos a seguir:

- No lado esquerdo da interface do MAME (onde há uma lista de itens disponíveis como *Sem filtro*, *Disponível*, *Indisponível*, etc.) selecione com um duplo clique no item *Source File*.
- Escolha o código fonte desejado, neste exemplo, selecionamos o `capcom/cps2.cpp`.
- A interface deverá exibir uma lista com todos os sistemas existentes em `capcom/cps2.cpp`.
- Clique no ícone do disquete no topo da tela e selecione *Exporta e lista em formato TXT*.

Um arquivo texto chamado **exported.txt** será criado dentro da pasta **ui** com a sua lista, exemplo:

Nome:	Descrição:
1944	"1944: The Loop Master (Euro 000620)"
...	
armwar	"Armored Warriors (Euro 941024)"
...	
avsp	"Alien vs. Predator (Euro 940520)"

7.12.26 Extraíndo uma lista de sistemas de um determinado driver

O comando `-listfull / -ll` oferece uma listagem de sistemas com o nome da ROM seguido da sua descrição (como também foi demonstrado no capítulo *Criando uma lista de jogos*), no entanto, o comando não cria uma lista com base num arquivo de código fonte (driver) específico, como um determinado sistema que esteja dentro de `src/mame/capcom/cps1.cpp` por exemplo, se usarmos o comando `-ll` com sistema `sf2` como exemplo, ele retorna:

```
mame -ll sf2
sf2 "Street Fighter II: The World Warrior (World 910522)"
```

Pessoas que têm uma leve noção de como o MAME é estruturado sabe que este sistema pertence ao CPS-1 da CAPCOM e está dentro do arquivo `src/mame/capcom/cps1.cpp`. O comando `-listsource / -ls` ajuda a identificar qual o arquivo fonte este sistema pertence, ainda usando o `sf2` como exemplo:

```
mame -ls sf2
sf2 capcom/cps1.cpp
```

Até o presente momento o MAME não possui nenhuma opção via linha de comando para listar todos os sistemas de um determinado driver como o `capcom/cps1.cpp` por exemplo, por isso que neste caso precisamos recorrer a ferramentas externas para obter tal lista.

O MAME suporta uma grande quantidade de ROMs, na casa de centena de milhares, são tantas que é impossível tentar relatar o número exato, pois mais e mais ROMs são adicionadas todos os dias. No entanto, é mais fácil para a maioria das pessoas lembrar o nome específico do título em vez do nome da ROM.

Para extrair todos os sistemas de `src/mame/capcom/cps1.cpp`, podemos utilizar as ferramentas `grep`, `awk` e `sed`, estas ferramentas são nativas do Linux, macOS e podem ser usadas no Windows através do ambiente de desenvolvimento `MSYS2`. Baixe o código fonte do mame no [site oficial do github](#) do MAME, geralmente o arquivo com o código fonte segue o padrão `mame[versão]` com um `s` no final. Usar o `git` é mais fácil, no Linux (Debian, Ubuntu, etc.) você instala com:

```
sudo apt install git-all
```


Para sistemas como pacotes RPM (Fedora, RHEL, CentOS, etc.):

```
sudo dnf install git-all
```

Veja [aqui](#) como instalar no macOS.

No terminal, execute este comando e aguarde a sua conclusão:

```
git clone https://github.com/mamedev/mame.git
```

Será criada uma pasta chamada **mame** entre nela com `cd mame` e rode o comando abaixo:

```
cat src/mame/capcom/cps1.cpp|grep "GAME( " | awk -F' "' '{print $4}' | sort -d
```

O comando vai extrair a lista de todos os sistemas em `src/mame/capcom/cps1.cpp`, exemplo:

```
1941: Counter Attack (Japan)
1941: Counter Attack (USA 900227)
...
Adventure Quiz Capcom World 2 (Japan 920611)
...
Area 88 (Japan)
...
```

Agora nós precisamos adicionar o nome da ROM ao lado dos títulos, para isso fazendo o comando abaixo:

```
cat src/mame/capcom/cps1.cpp|grep "GAME( " | awk -F' ' '{print $2 " - " $10}' | sed 's/"//g' | \
↵sort -d

1941 - 1941: Counter Attack (World 900227)
1941j - 1941: Counter Attack (Japan)
...
cworld2j - Adventure Quiz Capcom World 2 (Japan 920611)
...
area88 - Area 88 (Japan)
...
```

Fica mais fácil redirecionar a lista num arquivo texto e usar o Ctrl + F para buscar pelas palavras chaves:

```
cat src/mame/capcom/cps1.cpp|grep "GAME( " | awk -F' ' '{print $2 " - " $10}' | sed 's/"//g' | \
↵sort -d > lista_cps1.txt
```

Ou ainda continuar usando o próprio terminal para filtrar o que deseja, como Air por exemplo:

```
cat src/mame/capcom/cps1.cpp|grep "GAME( " | awk -F' ' '{print $2 " - " $10}' | sed 's/"//g' | \
↵sort -d | grep Air
cawing - Carrier Air Wing (World 901012)
cawingr1 - Carrier Air Wing (World 901009)
cawingu - Carrier Air Wing (USA 901130)
cawingur1 - Carrier Air Wing (USA 901012)
```

Ou agora usando o arquivo `src/mame/capcom/cps2.cpp` e filtrando por Gem:

```
cat src/mame/capcom/cps2.cpp|grep "GAME( " | awk -F' ' '{print $2 " - " $10}' | sed 's/"//g' | \
↵sort -d | grep Gem
```

(continua na próxima página)

(continuação da página anterior)

```
sgemfa - Super Gem Fighter: Mini Mix (Asia 970904)
sgemfd - Super Gem Fighter Mini Mix (USA 970904 Phoenix Edition) (bootleg)
sgemfh - Super Gem Fighter: Mini Mix (Hispanic 970904)
sgemf - Super Gem Fighter Mini Mix (USA 970904)
```

O mesmo pode ser feito em qualquer arquivo fonte, aqui para quem precisa buscar por Money no arquivo fonte para sistemas Neo Geo:

```
cat src/mame/neogeo/neogeo.cpp | grep "GAME( " | awk -F', ' '{print $2 " - " $10}' | sed 's/"//g'
↪ | sort -d | grep Money
miexchg - Money Puzzle Exchanger / Money Idol Exchanger
```

Letas maiúsculas e minúsculas alteram o resultado, maiúscula:

```
cat src/mame/neogeo/neogeo.cpp | grep "GAME( " | awk -F', ' '{print $2 " - " $10}' | sed 's/"//g'
↪ | sort -d | grep Bang
b2b - Bang Bang Busters (2010 NCI release)
bangbead - Bang Bead
```

Minúscula:

```
cat src/mame/neogeo/neogeo.cpp | grep "GAME( " | awk -F', ' '{print $2 " - " $10}' | sed 's/"//g'
↪ | sort -d | grep bang
bangbead - Bang Bea
```

Indiferente (opção -i):

```
cat src/mame/neogeo/neogeo.cpp | grep "GAME( " | awk -F', ' '{print $2 " - " $10}' | sed 's/"//g'
↪ | sort -d | grep -i bang
b2b - Bang Bang Busters (2010 NCI release)
bangbead - Bang Bead
```

7.12.27 Como criar apenas um mame.ini com as configurações de fábrica

A opção `-cc` cria um novo `mame.ini` junto com `ui.ini` e `plugin.ini`, infelizmente a opção não permite criar apenas o `mame.ini`. Neste caso, rode o comando abaixo:

```
mame -norc -sc > mame_padrao.ini
```

A opção `-norc` ignora qualquer configuração que você possa ter no seu computador, já a opção `-sc` mostra as predefinições internas do MAME, a parte final do comando redireciona essa lista para o arquivo `mame_padrao.ini`.

Se quiser substituir o seu `mame.ini` antigo, **faça o backup dele primeiro** e em seguida:

```
mame -norc -sc > mame.ini
```

Como este arquivo tem apenas os valores predefinidos do MAME, não se esqueça de reconfigurar o seu **rompath** para que o MAME possa localizar as suas ROMs, assim como outras configurações de vídeo e áudio para que o seu MAME possa funcionar corretamente.

- *Introdução*
- *Os comandos do depurador*
- *Determinando os dispositivos e as faixas de endereço*
- *A sintaxe das expressões do depurador*
 - *Números*
 - *Valores booleanos*
 - *Acesso à memória*
 - *Funções*

8.1 Introdução

O MAME inclui um depurador interativo de baixo nível que tem como alvo o sistema de emulação. Pode ser uma ferramenta útil para diagnosticar problemas relacionados à emulação, no desenvolvimento do software para rodar em sistemas antigos, na criação das trapaças, para o "*hacking*" de uma ROM ou simplesmente para investigar como o software funciona.

Use a opção `-debug` na linha de comando para iniciar o MAME com o depurador ativado. Como padrão, pressione a tecla `'` (`"`) durante a emulação para entrar no depurador (é possível alterar a tecla alterando a configuração *Entra no depurador*).

A aparência exata do depurador depende do seu sistema operacional e das opções com as quais o MAME foi compilado. Todas as variantes do depurador fornecem uma interface com várias janelas para a visualização do conteúdo da memória e do código desmontado.

A janela do console do depurador é uma janela especial que mostra o conteúdo dos registros da *CPU*, o código desmontado em torno do endereço do contador atual do programa e fornece também uma interface

com linha de comando que auxilia em toda a funcionalidade do depurador.

8.2 Os comandos do depurador

Os comandos do depurador são descritos nas seções abaixo. Também é possível digitar `help` comando no console do depurador para que a ajuda seja exibida diretamente na tela do depurador.

8.2.1 Comandos gerais do depurador

help

Exibe a ajuda integrada no console.

do

Avalia a expressão fornecida.

symlist

Lista os símbolos registrados.

softreset

Executa uma redefinição simples¹.

hardreset

Executa uma redefinição completa².

print

Imprime um ou mais *<item>*s no console.

printf

Imprime um ou mais *<item>*s no console usando *<formato>*.

logerror

Grava um ou mais *<item>*s no `error.log`.

tracelog

Grava um ou mais *<item>*s no arquivo de rastreamento usando *<formato>*.

tracesym

Grava um ou mais *<item>*s no arquivo de rastreamento.

history

Exibe os endereços PC e os *opcodes* visitados recentemente.

trackpc

Rastreia visualmente os *opcodes* já visitados.

trackmem

Grava qual PC escreveu em cada endereço da memória.

pcatmem

Consulta qual PC escreveu num determinado endereço da memória.

rewind

Volta no tempo carregando o estado mais recente da rebobinagem.

statesave

Grava o estado no arquivo para o sistema que está sendo emulado.

stateload

Lê o arquivo do estado do sistema que está sendo emulado.

snap

¹ *Soft reset* no Inglês.

² *Hard reset* no Inglês.

Grava uma captura da tela.

source

Faz a leitura dos comando a partir do arquivo e os executa individualmente.

time

Imprime o tempo atual do sistema no console.

quit

Encerra o depurador e a emulação.

help

help [<tópico>]

Exibe a ajuda integrada do depurador no console. Caso nenhum <tópico> seja definido, os principais tópicos serão listados. A maioria dos comandos do depurador possuem tópicos correspondentes de ajuda.

Exemplos:

help

Lista os principais tópicos de ajuda.

help expressions

Exibe a ajuda integrada para a sintaxe de expressão do depurador.

help wpiiset

Exibe a ajuda integrada para o comando *wpiiset*.

Retorna para *Comandos gerais do depurador*.

do

do <expressão>

O comando do simplesmente avalia a expressão fornecida. Geralmente é utilizado para definir ou alterar a variável do estado do dispositivo (os registros da *CPU* por exemplo) ou para escrever na memória. Consulte *A sintaxe das expressões do depurador* para obter mais informações sobre a sintaxe da expressão.

Exemplo:

do pc = 0

Define o registro pc como 0.

Retorna para *Comandos gerais do depurador*.

symlist

symlist [<CPU>]

Lista os símbolos registrados e seus respectivos valores. Quando a <CPU> não for definida, os símbolos na tabela de símbolos globais são exibidos, caso contrário, são exibidos os símbolos específicos do dispositivo <CPU>. Os símbolos são listados em ordem alfabética. São anotados apenas os símbolos que sejam somente de leitura. Consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações de como definir uma CPU.

Exemplos:

```
symlist
```

Exibe a tabela global dos símbolos.

```
symlist 2
```

Exibe os símbolos para a 3ª CPU do sistema (num índice com base zero).

```
symlist audiocpu
```

Exibe os símbolos para a CPU com o caminho absoluto da etiqueta :audiocpu.

Retorna para [Comandos gerais do depurador](#).

softreset

softreset

Executa uma redefinição simples. Invoca as funções do membro *reset* de todos os dispositivos do sistema (por padrão, o mesmo ocorre ao pressionar F3 durante a emulação).

Exemplo:

```
softreset
```

Executa uma redefinição simples.

Retorna para [Comandos gerais do depurador](#).

hardreset

hardreset

Executa uma redefinição completa. Isso faz o reinício completo da sessão e inicia uma nova com o mesmo sistema e com as mesmas opções. (por padrão, o mesmo ocorre ao pressionar Shift + F3 durante a emulação). Observe que isso ocasiona a perda do histórico no console do emulador e do registro de erro.

Exemplo:

```
hardreset
    Executa uma redefinição completa.
```

Retorna para *Comandos gerais do depurador*.

print

print <item> [,...]

O comando print imprime o resultado de uma ou mais expressões no console do depurador como números hexadecimais.

Exemplos:

```
print pc
    Imprime o valor do registro pc no console como um número hexadecimal.
print a,b,a+b
    Imprime o valor de a, b e de a + b no console como números hexadecimais.
```

Retorna para *Comandos gerais do depurador*.

printf

printf <formato> [,<argumento> [,...]]

Imprime uma mensagem formatada em estilo C no console de depuração. Apenas um subconjunto muito limitado dos especificadores de formato e das sequências de escape estão disponíveis:

```
%c
    Imprime o argumento correspondente como um caractere de 8-bit.
%[-][0][<n>]d
    Imprime o argumento correspondente como um número decimal com justificação opcional à esquerda, preenchimento zero e largura mínima do campo.
%[-][0][<n>]o
    Imprime o argumento correspondente como um número octal com largura opcional mínima do campo, com preenchimento zero usando letras minúsculas.
```

`%[-][0][<n>]x`

Imprime o argumento correspondente como um número hexadecimal em minúsculas com justificação opcional à esquerda, preenchimento zero e largura mínima do campo.

`%[-][0][<n>]X`

Imprime o argumento correspondente como um número hexadecimal maiúsculo com justificação opcional à esquerda, preenchimento zero e largura mínima do campo.

`%[-][<n>][. [<n>]]s`

Imprime uma cadeia de caracteres de 8-bits com terminação nula a partir do endereço e do espaço de endereço através do argumento correspondente, com justificação opcional à esquerda e larguras de campo mínima e máxima.

`\%%`

Imprime literalmente um símbolo de porcentagem.

`\n`

Imprime uma quebra de linha.

`\\`

Imprime literalmente uma barra inclinada à esquerda.

Qualquer outro formato que for utilizado será ignorado.

Exemplos:

```
printf "PC=%04X", pc
```

Imprime PC=<pcval> onde <pcval> é o valor hexadecimal do registro pc com um mínimo de quatro dígitos e preenchimento zero.

```
printf "A=%d, B=%d\\nC=%d", a, b, a+b
```

Imprime A=<aval>, B=<bval> numa linha e C=<a+bval> na segunda.

Retorna para *Comandos gerais do depurador*.

logerror

logerror <formato> [, <argumento> [, ...]]

Imprime uma mensagem formatada em estilo C no registro de erro. Consulte *printf* para mais detalhes sobre o limitado subconjunto dos especificadores de formato e das sequências de escape.

Exemplos:

```
logerror "PC=%04X", pc
```

Registra PC=<pcval> onde <pcval> é o valor hexadecimal do registro pc com um mínimo de quatro dígitos e preenchimento zero.

```
logerror "A=%d, B=%d\\nC=%d", a, b, a+b
```

Registra A=<aval>, B=<bval> numa linha, e C=<a+bval> na segunda.

Retorna para *Comandos gerais do depurador*.

tracelog

tracelog <format> [, <argumento> [...]]

Imprime uma mensagem formatada em estilo C para o arquivo de rastreamento aberto no momento (consulte [trace](#) para obter mais informações). O comando não tem qualquer efeito caso nenhum arquivo de rastreamento esteja aberto. Consulte [printf](#) para obter mais informações sobre o limitado subconjunto dos especificadores de formato e das sequências de escape.

Exemplos:

```
tracelog "PC=%04X", pc
```

Produz PC=<pcval> onde <pcval> é o valor hexadecimal do registro pc com um mínimo de quatro dígitos e preenchimento zero caso o arquivo de registro de rastreamento esteja aberto.

```
tracelog "A=%d, B=%d\\nC=%d", a, b, a+b
```

Produz A=<aval>, B=<bval> numa linha e C=<a+bval> na segunda, caso o arquivo de registro de rastreamento esteja aberto.

Retorna para [Comandos gerais do depurador](#).

tracesym

tracesym <item> [...]

Imprime os símbolos indicados no arquivo de rastreamento aberto no momento (consulte [trace](#) para obter mais informações). O comando não tem qualquer efeito caso nenhum arquivo esteja aberto.

Exemplo:

```
tracesym pc
```

Produz PC=<pcval> onde <pcval> é o valor do registro pc em seu formato padrão caso o arquivo de registro de rastreamento esteja aberto.

Retorna para [Comandos gerais do depurador](#).

history

history [<CPU>[,<comprimento>]]

Exibe os endereços *PC* visitados recentemente e a desmontagem das instruções nestes endereços. Se presente, o primeiro argumento seleciona a *CPU* (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações), caso contrário, assume a *CPU* que estiver visível. Caso o segundo argumento esteja presente, faz a limitação da quantidade de endereços que será exibido. Os endereços são apresentados na ordem do menor para os mais recentemente visitados.

Exemplos:

history ,5

Exibe até cinco endereços e instruções *PC* mais recentemente visitados para a *CPU* que estiver visível.

history 3

Exibe os endereços e as instruções *PC* mais recentemente visitados para a 4ª *CPU* do sistema (num índice com base zero).

history audiocpu,1

Exibe os endereços e as instruções *PC* mais recentemente visitados com o caminho absoluto da etiqueta :audiocpu.

Retorna para [Comandos gerais do depurador](#).

trackpc

trackpc [<ativa>[,<CPU>[,<apaga>]]]

Liga ou desliga o rastreamento do endereço *PC* visitado para a visualização da desmontagem. As instruções nos endereços visitados são destacadas nas visualizações de desmontagem do depurador enquanto o rastreamento estiver ligado. O 1º argumento é um booleano determinando se o rastreamento deve ser ativado ou não (o padrão é enabled). Já o 2º argumento determina em qual *CPU* o rastreamento deve ser ativado ou não (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações). Quando nenhuma *CPU* for definida, assume a 1ª *CPU* que estiver visível. O 3º argumento é um booleano que determina se os dados existentes devem ser apagados ou não (o padrão é false).

Exemplos:

trackpc 1

Inicia ou faz o rastreamento *PC* da *CPU* atual.

trackpc 1,0,1

Inicia ou continua o rastreamento *PC* na 1ª *CPU* do sistema (num índice com base zero), porém apaga o histórico que foi rastreado até o momento.

Retorna para [Comandos gerais do depurador](#).

trackmem

trackmem [*<ativa>*],[*<CPU>*],[*<apaga>*]]

Liga ou desliga o registro do endereço *PC* cada vez que um endereço da memória for escrito. O 1º argumento é um booleano determinando se o rastreamento deve ser ativado ou não (o padrão é *enabled*). Já o 2º argumento determina em qual *CPU* o rastreamento deve ser ativado ou não (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações). Quando nenhuma *CPU* for definida, assume a primeira *CPU* que estiver visível. O 3º argumento é um booleano que determina se os dados existentes devem ser apagados ou não (o padrão é *false*).

Utilize [pcatmem](#) para reaver estes dados. Ao clicar com o botão direito do mouse no visualizado de memória do depurador, também será exibido o valor registrado no *PC* para dado endereço em algumas configurações.

Exemplos:

trackmem

Inicia ou continua o rastreamento das escritas na memória na *CPU* que estiver visível no momento.

trackmem 1,0,1

Inicia ou continua o rastreamento das escritas na memória na 1ª *CPU* do sistema (num índice com base zero), porém apaga os dados de rastreamentos já existentes.

Retorna para [Comandos gerais do depurador](#).

pcatmem

pcatmem[{*d|i|o*}] *<endereço>*[:*<espaço>*]

Retorna o valor *PC* no momento onde o endereço especificado tenha sido recentemente escrito. O argumento é o endereço solicitado, opcionalmente seguido por dois pontos e uma *CPU* e/ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações).

Os sufixos *d*, *i* ou *o* controlam a faixa de endereços padrão para o comando.

O rastreamento deve estar ativado para que os dados utilizados por este comando possam ser gravados (consulte [trackmem](#)). Ao clicar com o botão direito do mouse no visualizado de memória do depurador, também será exibido o valor registrado no *PC* para dado endereço em algumas configurações.

Exemplos:

pcatmem 400000

Imprime o valor recentemente escrito no *PC* quando a localização 400000 estiver na região visível do programa na *CPU*.

pcatmem 3bc:io

Imprime o valor recentemente escrito no *PC* quando a localização 3bc estiver na faixa *io* da *CPU* que estiver visível.

pcatmem 1400:audiocpu

Imprime o valor recentemente escrito no *PC* quando a localização 1400 estiver na faixa do programa *:audiocpu* da *CPU*.

Retorna para *Comandos gerais do depurador*.

rewind

rewind

Carrega o estado salvo mais recente da *RAM*. Quando ativado os estados *rewind* são gravados quando os comando *step*, *over* e *out* são utilizados, armazenando o estado do sistema num momento antes do avanço. Este comando pode ser abreviado para *rw*.

A carga consecutiva dos estados *rewind* pode funcionar como uma execução inversa. Dependendo das etapas que foram tomadas anteriormente, o comportamento pode ser semelhante ao dos comandos *reverse-stepi* e *reverse-next* do *GDB*. Toda a saída deste comando é ecoada em tempo real na janela do sistema que estiver em execução.

As estatísticas anteriores da memória e do rastreamento do PC são apagadas. A atual execução reversa não ocorre.

Exemplos:

rewind

Carrega o estado anterior da *RAM*.

rw

Uma forma breviada do comando.

Retorna para *Comandos gerais do depurador*.

statesave

statesave <nome_do_arquivo>

Grava um estado do sistema do momento atual no tempo do emulador. O arquivo do estado é gravado no diretório de gravação do estado (consulte a opção *state_directory*), a extensão *.sta* é adicionada automaticamente ao nome do arquivo. Pode ser abreviado para *ss*.

Toda a saída deste comando é ecoada em tempo real na janela do sistema que estiver em execução.

Exemplos:

statesave foo

Grava o estado do sistema que está sendo emulado no arquivo *foo.sta* no diretório de gravação do estado.

ss bar

A forma abreviada do comando, grava o estado do sistema que está sendo emulado no arquivo *bar.sta*.

Retorna para *Comandos gerais do depurador*.

stateload

stateload <nome_do_arquivo>

Carrega o arquivo de estado a partir do disco. O arquivo do estado é carregado a partir do diretório de gravação do estado (consulte a opção [state_directory](#)), a extensão `.sta` é adicionada automaticamente ao nome do arquivo. Pode ser abreviado para `sl`.

Toda a saída deste comando é ecoada em tempo real na janela do sistema que estiver em execução. A memória anterior e as estatísticas de rastreamento do PC serão apagadas.

Exemplos:

```
stateload foo
```

Carrega o arquivo do estado `foo.sta` a partir do diretório de gravação do estado.

```
sl bar
```

A forma abreviada do comando, carrega o arquivo `bar.sta`.

Retorna para [Comandos gerais do depurador](#).

snap

snap [<nome_do_arquivo>[,<scrnum>]]

Faz uma captura da tela emulada e grava no diretório *snapshot* (consulte a opção [snapshot_directory](#)).

Quando um nome é definido, uma única captura é salva usando este nome (o a primeira tela do sistema caso nenhuma tenha sido definida). Quando um nome não é definido, as configurações predefinidas são utilizadas (consulte as opções [snapview](#) e [snapname](#)).

A extensão `.png` é adicionada automaticamente ao nome do arquivo. O número da tela é atribuída num índice com base zero, conforme é visto nos nomes das visualizações com uma única tela que são geradas automaticamente nos menus das opções de vídeo do MAME.

Exemplos:

```
snap
```

Faz uma captura da tela usando as configurações já predefinidas.

```
snap shinobi
```

Faz uma captura da primeira tela e salva como `shinobi.png` no diretório configurado para *snapshot*.

Retorna para [Comandos gerais do depurador](#).

source

source <nome_do_arquivo>

Carrega o arquivo em modo texto e executa cada linha como se fossem comandos de depuração. É equivalente a rodar um *script shell* ou um arquivo *batch*.

Exemplo:

```
source break_and_trace.cmd
```

Carrega e executa os comandos do depurador a partir do arquivo `break_and_trace.cmd`.

Retorna para *Comandos gerais do depurador*.

time

Imprime o tempo total decorrido da emulação no console do depurador.

Exemplos:

```
time
```

Imprime o tempo decorrido da emulação.

Retorna para *Comandos gerais do depurador*.

quit

quit

Fecha o depurador e encerra a emulação imediatamente. Ou encerra o MAME ou retorna ao menu de seleção do sistema dependendo de como o sistema foi usado na linha de comando, se o MAME foi iniciado sozinho e sem comandos ele retorna para a interface gráfica, caso tenha sido iniciado com algum sistema, o MAME encerra imediatamente.

Exemplo:

```
quit
```

Encerra a emulação imediatamente.

Retorna para *Comandos gerais do depurador*.

8.2.2 Comandos para a depuração da memória

Nota: **Pesquisa** é quando se deseja levantar mais informações sobre algo desconhecido. **Busca** é quando se deseja levantar informações sobre algo que já se sabe.

dasm

Desmonta o código num determinado arquivo.

find

Faz uma pesquisa por dados na memória da emulação.

fill

Preenche a memória da emulação num determinado padrão.

dump

Despeja a memória da emulação num arquivo texto.

strdump

Despeja strings¹ delimitadas a partir da memória da emulação num arquivo.

save

Grava os dados binários da memória da emulação num arquivo.

saver

Grava os dados binários de uma determinada faixa da memória num arquivo.

load

Carrega os dados binários a partir de um arquivo na memória da emulação.

loadr

Carrega os dados da memória a partir de um arquivo na memória da emulação.

map

Faz o mapeamento do endereço lógico ao endereço físico e ao manipulador correspondente.

memdump

Despeja o mapa da memória atual num arquivo.

dasm

dasm <nome_do_arquivo>,<endereço>,<comprimento>[,<opcodes>[,<CPU>]]

Desmonta a memória do programa num arquivo definido pelo parâmetro <nome_do_arquivo>. O parâmetro <endereço> determina o endereço para o início da desmontagem, o parâmetro <comprimento> determina o quanto da memória será desmontada. A faixa do <endereço> através de <endereço> + <comprimento> inclusive, será desmontada num arquivo. É predefinido que os dados brutos do *opcode* sejam gerados em cada linha. O parâmetro opcional <opcodes> é um booleano que ativa ou desativa essa funcionalidade. Também é predefinido que a memória do programa seja desmontada na *CPU* que estiver visível. Defina o 5º parâmetro para desmontar a memória do programa numa *CPU* diferente (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes).

¹ Pode ser traduzido como cadeia de caracteres, cadência de caracteres, sequência de caracteres, caracteres, texto, dentre outras variantes dependendo do contexto, [mais informações](#).

Exemplos:

```
dasm venture.asm,0,10000
```

Desmonta o endereço 0-ffff na CPU que estiver visível, incluindo os dados *opcode* bruto no arquivo *venture.asm*.

```
dasm harddriv.asm,3000,1000,0,2
```

Desmonta o endereço 3000-3fff na 3ª CPU do sistema (num índice com base zero), sem os dados *opcode* bruto no arquivo *harddriv.asm*.

Retorna para [Comandos para a depuração da memória](#).

find

f[ind][{d|i|o}] <endereço>[:<faixa>],<comprimento>[,<dados>[,...]]

Faz uma busca na memória por uma determinada sequência de dados. O <endereço> determina o início da busca, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). A <faixa> determina o quanto da memória será buscada. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando *find* exiba a primeira faixa de endereços que for exposta pelo dispositivo, *findd* retorna para a faixa do índice 1 (dados), *findi* retorna para a faixa do índice 2 (E/S), já *findo* retorna para a faixa do índice 3 (*opcodes*).

No parâmetro <dados> pode ser usado uma *string*, um valor numérico, uma expressão ou um caractere curinga ?. É predefinido que os dados não *string* sejam escritos usando o tamanho nativo da palavra na faixa do endereço. Para substituir o tamanho dos dados para dados não *string*, você pode prefixar os valores com b. para impor a busca com um tamanho de byte, com w. para fazer a busca com o tamanho da palavra (*word*), com d. para palavra dupla (*double word*) e q. para palavra quádrupla (*quadruple word*). As sobreposições se propagam para os valores subsequentes, assim sendo, caso queira fazer uma busca por uma sequência de palavras, é preciso prefixar o primeiro valor com w.. Observe também que é possível misturar os tamanhos para realizar buscas mais complexas.

Toda a faixa do <endereço> através do <endereço>+<comprimento>-1 inclusive, será buscado pela sequência e todas as ocorrências serão exibidas.

Exemplos:

```
find 0,10000,"HIGH SCORE",0
```

Faz uma busca pelos caracteres "HIGH SCORE" na faixa do endereço 0-ffff do programa da CPU que estiver visível seguido pelo byte 0.

```
find 300:tms9918a,100,w.abcd,4567
```

Faz uma busca com o tamanho palavra *abcd* e seguido pelo o valor 4567 na faixa do endereço 300-3fff no primeiro espaço do endereço que estiver exposto pelo dispositivo com o caminho absoluto da etiqueta : *tms9918a*.

```
find 0,8000,"AAR",d.0,"BEN",w.0
```

Faz uma busca pelos caracteres "AAR" na faixa do endereço 0000-7fff seguido por uma palavra dupla (*double word*) 0 e pelos caracteres "BEM" com tamanho 0.

Retorna para [Comandos para a depuração da memória](#).

fill

fill{**d|i|o**} <endereço>[:<faixa>],<comprimento>[,<dados>[,...]]

Sobrescreve um bloco da memória com uma cópia da sequência de dados que foi fornecida.

O <endereço> determina o início de onde a escrita deve começar, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). A <faixa> determina o quanto da memória deve ser preenchida. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando fill exiba a primeira faixa de endereços que for exposta pelo dispositivo, filld retorna para a faixa do índice 1 (dados), filli retorna para a faixa do índice 2 (E/S), já fillo retorna para a faixa do índice 3 (opcodes).

No parâmetro <dados> pode ser usado uma *string*, um valor numérico, uma expressão ou um caractere curinga ?. É predefinido que os dados não *string* sejam escritos usando o tamanho nativo da palavra na faixa do endereço. Para substituir o tamanho dos dados para dados não *string*, você pode prefixar os valores com b. para impor o preenchimento com um tamanho de byte, com w. para fazer o preenchimento com o tamanho da palavra (*word*), com d. para palavra dupla (*double word*) e q. para palavra quádrupla (*quadruple word*). As sobreposições se propagam para os valores subsequentes, assim sendo, caso queira fazer um preenchimento por uma sequência de palavras, é preciso prefixar o primeiro valor com w.. Observe também que é possível misturar os tamanhos para realizar preenchimentos mais complexos. A operação de preenchimento poderá ser truncada caso ocorra um erro falha da página (*page fault*) ou caso uma parte da sequência ou da *string* falhe além do <endereço>+<comprimento>-1.

Retorna para [Comandos para a depuração da memória](#).

dump

dump{**d|i|o**} <nome_do_arquivo>,<endereço>[:<faixa>],<comprimento>[,<grupo>[,<ascii>[,<tamanho_da_linha>]]]

Faz o despejo do conteúdo da memória num arquivo texto definido pelo parâmetro <nome_do_arquivo>.

O <endereço> determina o início de onde o despejo deve começar, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). A <faixa> determina o quanto da memória deve ser despejada. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando dump exiba a primeira faixa de endereços que for exposta pelo dispositivo, dumpd retorna para a faixa do índice 1 (dados), dumpi retorna para a faixa do índice 2 (E/S), já dumpt retorna para a faixa do índice 3 (opcodes).

Toda a faixa do <endereço> através do <endereço>+<comprimento>-1 inclusive, será despejado no arquivo. É predefinido que os dados sejam produzidos usando o tamanho nativo da palavra na faixa do endereço. É possível alterar esta funcionalidade ao definir o parâmetro <group> que pode ser utilizado para agrupar os dados em pedaços de 1-, 2-, 4-, 8 bytes. O parâmetro opcional <ASCII> é um booleano que ativa ou desativa a geração dos caracteres *ASCII* do lado direito de cada linha (a predefinição é enabled). Já o parâmetro opcional <tamanho_da_linha> determina a quantidade de dados nas unidades de endereço de cada linha (é predefinido que seja 16 bytes).

Exemplos:

```
dump venture.dmp,0,10000
```

Faz o despejo do endereço 0-ffff em pedaços de 1-byte na região visível do programa na CPU incluindo dados ASCII no arquivo venture.dmp.

```
dumpd harddriv.dmp,3000:3,1000,4,0
```

Faz o despejo dos dados do endereço da memória 3000-3fff em pedaços de 4-bytes a partir da 4ª CPU do sistema (num índice com base zero) no arquivo harddriv.dmp.

```
dump vram.dmp,0:sms_vdp:videoram,4000,1,false,8
```

Faz o despejo da faixa do endereço 0000-3fff da videoram em pedaços de 1-byte a partir do dispositivo com o caminho absoluto da etiqueta :sms_vdp, sem dados ASCII, com 8 bytes por linha no arquivo vram.dmp.

Retorna para [Comandos para a depuração da memória](#).

strdump

strdump[[{d|i|o}] <nome_do_arquivo>,<endereço>[:<faixa>],<comprimento>[,<term>]

Faz o despejo do conteúdo da memória num arquivo texto definido pelo parâmetro <nome_do_arquivo>.

O <endereço> determina o início de onde o despejo deve começar, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). A <faixa> determina o quanto da memória deve ser despejada. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando strdump retorne para a primeira faixa de endereços que for exposta pelo dispositivo, strdumpd retorna para a faixa do índice 1 (dados), strdumpi retorna para a faixa do índice 2 (E/S), já strdumpo retorna para a faixa do índice 3 (opcodes).

É predefinido que os dados sejam interpretados como uma série de caracteres com uma terminação NULA (ASCIIZ), o despejo terá um caractere por linha, com sequências de escape em estilo C que serão utilizadas nos bytes que não representem caracteres ASCII imprimíveis. O parâmetro opcional <term> pode ser usado para determinar um caractere de terminação diferente da cadeia de caracteres.

Retorna para [Comandos para a depuração da memória](#).

save

save[[{d|i|o}] <nome_do_arquivo>,<endereço>[:<faixa>],<comprimento>

Grava os dados brutos da memória num arquivo binário determinado pelo parâmetro <nome_do_arquivo>.

O <endereço> determina o início de onde a gravação deve começar, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). A <faixa> determina o quanto da memória deve ser gravada. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando save retorne para a primeira faixa de endereços que for exposta pelo dispositivo, saved retorna para a faixa do índice 1 (dados), savei retorna para a faixa do índice 2 (E/S), já saveo retorna para a faixa do índice 3 (opcodes).

Toda a faixa do <endereço> através do <endereço>+<comprimento>-1 inclusive, será gravado num arquivo.

Exemplos:

```
save venture.bin,0,10000
```

Grava o endereço 0-ffff na região visível do programa na *CPU* atual em formato binário no arquivo *venture.bin*.

```
saved haddriv.bin,3000:3,1000
```

Grava os dados do endereço da memória 3000-3fff a partir da 4ª *CPU* do sistema (num índice com base zero) no arquivo *haddriv.bin*.

```
save vram.bin,0:sms_vdp:videoram,4000
```

Grava a faixa do endereço 0000-3fff da *videoram* a partir do dispositivo com o caminho absoluto da etiqueta *:sms_vdp* em formato binário no arquivo *vram.bin*.

Retorna para *Comandos para a depuração da memória*.

saver

saver <nome_do_arquivo>,<endereço>,<comprimento>,<região>

Grava o conteúdo bruto de uma região da memória determinado pelo parâmetro <região> num arquivo binário determinado pelo parâmetro <nome_do_arquivo>. As etiquetas da região seguem as mesmas regras que as etiquetas dos dispositivos (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais informações). O <endereço> determina o início de onde a gravação deve começar. O <comprimento> determina o quanto da memória deve ser gravada.

Toda a faixa do <endereço> através do <endereço>+<comprimento>-1 inclusive, será gravado num arquivo.

Exemplos:

```
saver data.bin,200,100,:monitor
```

Grava a região do endereço 200-2ff do *:monitor* em formato binário no arquivo *data.bin*.

```
saver cpuom.bin,1000,400,.
```

Grava o endereço 1000-13ff na região visível do programa na *CPU* em formato binário no arquivo *cpuom.bin*.

Retorna para *Comandos para a depuração da memória*.

load

load[{d|i|o}] <nome_do_arquivo>,<endereço>[:<faixa>][,<comprimento>]

Carrega uma memória em formato bruto a partir de um arquivo binário determinado pelo parâmetro <nome_do_arquivo>. O <endereço> determina a partir de onde o arquivo deve ser lido, opcionalmente por ser seguido por um dispositivo ou uma faixa de endereços (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes). O <comprimento> determina o quanto da memória deve ser lida. Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando *load* retorne para a primeira faixa de endereços que for exposta pelo dispositivo,

loadb retorna para a faixa do índice 1 (dados), loadl retorna para a faixa do índice 2 (E/S), já loado retorna para a faixa do índice 3 (*opcodes*).

Toda a faixa do *<endereço>* através do *<endereço>+<comprimento>-1* inclusive, será gravado num arquivo. Quando o parâmetro *<comprimento>* for omitido, caso seja zero ou caso seja maior que o comprimento total do arquivo, todo o conteúdo do arquivo será lido e nada mais.

Observe que isso tem o mesmo efeito que fazer a escrita na faixa do endereço a partir do visualizador da memória do depurador ou utilizando os acessórios da memória *b@*, *w@*, *d@* ou *q@* nas expressões do depurador. O conteúdo da memória que seja de leitura apenas não será substituído e a escrita nos endereços de E/S terão efeitos que vão além das configurações dos valores dos registros.

Exemplos:

```
load venture.bin,0,10000
```

Carrega o endereço 0-ffff na região visível do programa na CPU a partir do arquivo binário *venture.bin*.

```
loadb harddriv.bin,3000,1000,3
```

Carrega os dados da região da memória 3000-3fff a partir da 4ª CPU do sistema (num índice com base zero) a partir do arquivo binário *harddriv.bin*.

```
load vram.bin,0:sms_vdp:videoram
```

Carrega a faixa da videoram para o dispositivo com o caminho absoluto da etiqueta *:sms_vdp* iniciando no endereço 0000 com todo o conteúdo binário do arquivo *vram.bin*.

Retorna para [Comandos para a depuração da memória](#).

loadr

loadr *<nome_do_arquivo>*,*<endereço>*,*<comprimento>*,*<região>*

Carrega uma memória na região da memória determinado pelo parâmetro *<região>* a partir de arquivo binário determinado pelo parâmetro *<nome_do_arquivo>*. As etiquetas da região seguem as mesmas regras que as etiquetas dos dispositivos (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais informações). O *<endereço>* determina o início de onde a leitura deve começar. O *<comprimento>* determina o quanto da memória deve ser carregada.

Toda a faixa do *<endereço>* através do *<endereço>+<comprimento>-1* inclusive, será lido a partir do arquivo. Quando o *<comprimento>* for zero ou caso seja maior que o comprimento total do arquivo, todo o conteúdo do arquivo será lido e nada mais.

Exemplos:

```
loadr data.bin,200,100,:monitor
```

Carrega a região do endereço 200-2ff do *:monitor* a partir do arquivo binário *data.bin*.

```
loadr cpurom.bin,1000,400,.
```

Carrega o endereço 1000-13ff na região visível do programa na CPU a partir do arquivo binário *cpurom.bin*.

Retorna para [Comandos para a depuração da memória](#).

map

map[{**d**|**i**|**o**}] <endereço>[:<faixa>]

Faz o mapeamento lógico do endereço da memória no endereço físico correspondente, assim como informa o nome do manipulador. O endereço opcionalmente pode ser seguido por dois pontos e um dispositivo ou uma faixa de endereço (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. É predefinido que o comando map retorne para a primeira faixa de endereços que for exposta pelo dispositivo, mapd retorna para a faixa do índice 1 (dados), mapi retorna para a faixa do índice 2 (E/S), já mapo retorna para a faixa do índice 3 (*opcodes*).

Exemplos:

map 152d0

Fornece o endereço físico e o nome do manipulador para o endereço lógico 152d0 na região visível do programa na *CPU*.

map 107:sms_vdp

Fornece o endereço físico e o nome do manipulador para o endereço lógico 107 na primeira faixa do endereço para o dispositivo com o caminho absoluto da etiqueta :sms_vdp.

Retorna para [Comandos para a depuração da memória](#).

memdump

memdump [<nome_do_arquivo>,[<dispositivo>]]

Faz o despejo dos mapas da memória atual num arquivo determinado pelo parâmetro <nome_do_arquivo> ou memdump.log caso nenhum parâmetro seja usado. Quando um <dispositivo> foi definido (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes) apenas os mapas da memória para a parte da árvore enraizada neste dispositivo serão despejados.

Exemplos:

memdump mylog.log

Faz o despejo do mapa da memória para todos os dispositivos do sistema no arquivo mylog.log.

memdump

Faz o despejo do mapa da memória para todos os dispositivos do sistema no arquivo memdump.log.

memdump audiomaps.log,audiopcb

Faz o despejo do mapa da memória para todos os dispositivos com o caminho absoluto da etiqueta :audiopcb e todos os dispositivos relacionados no arquivo audiomaps.log.

memdump mylog.log,1

Faz o despejo do mapa da memória para o 2º dispositivo da *CPU* (num índice com base zero) e todos os dispositivos relacionados no arquivo mylog.log.

Retorna para [Comandos para a depuração da memória](#).

8.2.3 Comandos de execução do depurador

step

Passo único para instruções *<quantidade>* (F11).

over

Passo único durante instruções *<quantidade>* (F10).

out

Passo único até que o manipulador atual da sub-rotina/execução retorne (Shift-F11).

go

Retoma a execução (F5).

gbt

Retoma a execução até que a próxima verdadeira ramificação seja executada.

gbf

Retoma a execução até que a próxima falsa ramificação seja executada.

gex

Retoma a execução até que a exceção seja criada.

gint

Retoma a execução até que a interrupção seja obtida (F7).

gni

Retoma a execução até a próxima instrução mais adiante.

gtime

Retoma a execução até que o atraso determinado tenha decorrido.

gvblank

Retoma a execução até o próximo intervalo de apagamento vertical² (F8).

next

Retoma a execução até a próxima alternância da CPU (F6).

focus

Foca o depurador apenas na *<CPU>*.

ignore

Para a depuração na *<CPU>*.

observe

Retoma a depuração na *<CPU>*.

trace

Rastreia a determinada CPU para um arquivo.

traceover

Rastreia a determinada CPU para um arquivo, mas ignore as sub-rotinas.

traceflush

Elimina todos os arquivos de rastreamento que estiverem abertos.

² Vertical Blanking Interval, também conhecido como intervalo vertical ou VBLANK.

step**s[tep]** [<quantidade>]

Avança uma ou mais instruções sobre a *CPU* que estiver atualmente em execução. Executa uma instrução caso <quantidade> seja omitido ou a <quantidade> de passos para as instruções caso seja informada.

Exemplos:

s

Avança uma instrução na *CPU* atual.

step 4

Avança 4 instruções na *CPU* atual.

Retorna para *Comandos de execução do depurador*.

over**o[ver]** [<quantidade>]

O comando over avança um único passo sobre uma ou mais instruções que estiverem sendo executadas na *CPU*, passando por cima das chamadas da sub-rotina e das capturas do manipulador e contando-os como uma única instrução. Observe que, ao passar por cima de uma chamada da sub-rotina o código pode ser executado nas outras *CPUs* antes do retorno da chamada.

Passa por cima de uma instrução caso a <quantidade> seja omitida, ou passe por cima das instruções caso a <quantidade> seja informada.

Observe que esta funcionalidade pode não estar implementada em todos os tipos de *CPU*. Caso não esteja, então o comando over se comportará exatamente como o comando *step*.

Exemplos:

o

Avança uma instrução na *CPU* atual.

over 4

Avança 4 instruções na *CPU* atual.

Retorna para *Comandos de execução do depurador*.

out

out

O comando out avança um único passo até encontrar um retorno da sub-rotina ou caso o retorno de uma instrução em exceção seja encontrada. Observe que como ele detecta o retorno das condições da exceção, caso tente sair de uma sub-rotina e uma interrupção/exceção ocorra antes de atingir o final, será possível interromper o final das exceções do manipulador prematuramente.

Observe que a funcionalidade para sair pode não estar implementada em todos os tipos de *CPU*. Caso não esteja, então o comando out se comportará exatamente como o comando *step*.

Exemplo:

out

Continua os passos até que uma sub-rotina ou um manipulador das exceções retorne.

Retorna para *Comandos de execução do depurador*.

go

g[o] [<endereço>]

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou que um ponto de controle¹ seja atingido ou até que você faça uma interrupção manual. Caso o <endereço> opcional seja fornecido, um ponto temporário de interrupção incondicional será definido na *CPU* que estiver visível no endereço determinado. Este ponto será eliminado automaticamente quando for atingido.

Exemplos:

g

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que uma interrupção (*break*) seja manualmente requisitada.

g 1234

Retoma a execução parando no endereço 1234 até que outra condição faça com que a execução pare antes dela.

Retorna para *Comandos de execução do depurador*.

¹ Watchpoint no Inglês

gbf**gbf [<condição>]**

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja acionado ou até que uma ramificação condicional ou uma instrução ignorada não seja tomada, após qualquer slot atrasado.

O parâmetro opcional **<condição>** permite que você especifique uma expressão que será avaliada cada vez que uma ramificação condicional seja encontrada. Caso o resultado da expressão seja verdadeiro (não zero), a execução será interrompida se a ramificação não tiver sido tomada; caso contrário, a execução continuará sem qualquer notificação.

Exemplos:

gbf

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até a próxima ramificação falsa.

gbf {pc != 1234}

Retoma a execução até que a próxima ramificação seja falsa, independente da instrução no endereço 1234.

Retorna para *Comandos de execução do depurador*

gbt**gbt [<condição>]**

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja acionado ou até que uma ramificação condicional ou uma instrução ignorada seja tomada, seguido de qualquer slot atrasado.

O parâmetro opcional **<condição>** permite que você especifique uma expressão que será avaliada cada vez que uma ramificação condicional seja encontrada. Caso o resultado da expressão seja verdadeiro (não zero), a execução será interrompida após a ramificação ter sido tomada; caso contrário, a execução continuará sem qualquer notificação.

Exemplos:

gbt

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até a próxima ramificação verdadeira.

gbt {pc != 1234}

Retoma a execução até que a próxima ramificação seja verdadeira, independente da instrução no endereço 1234.

Retorna para *Comandos de execução do depurador*

gex**ge[x]** [*<exceção>*],[*<condição>*]

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja atingido, ou até que seja levantada uma condição de exceção na *CPU* atual. Use o parâmetro opcional *<exceção>* para parar a execução apenas numa condição de exceção específica. Caso a *<exceção>* não seja usada, a execução irá parar em qualquer condição de exceção.

O parâmetro opcional *<condição>* permite determinar uma expressão que será avaliada cada vez que uma condição específica de exceção for levantada. Caso o resultado da expressão seja verdadeiro (não zero), a exceção interromperá a execução; caso contrário, a execução continuará sem qualquer notificação.

Exemplos:

gex

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que uma condição de exceção seja levantada na *CPU* atual.

ge 2

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que uma condição de exceção 2 seja levantada na *CPU* atual.

Retorna para *Comandos de execução do depurador*.

gint**gi[nt]** [*<irqline>*]

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja atingido, ou até que uma interrupção seja confirmada e reconhecida na *CPU* atual. Use o parâmetro opcional *<irqline>* para parar a execução apenas na interrupção determinada da linha que está sendo confirmada e reconhecida. Caso *<irqline>* não seja usado, a execução será parada quando qualquer interrupção for reconhecida.

Exemplos:

gi

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que uma interrupção seja confirmada e reconhecida na *CPU* atual.

gint 4

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que uma interrupção requeira que a linha 4 seja confirmada e reconhecida na *CPU* atual.

Retorna para *Comandos de execução do depurador*.

gni**gni [<quantidade>]**

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja acionado. Um ponto de interrupção temporário e incondicional é definido no endereço do programa <quantidade> atual, sendo passado de forma sequencial. Ele é automaticamente removido quando este ponto de interrupção é acionado.

O parâmetro <quantidade> é opcional e retorna para o padrão 1 quando nada for definido. O comando não faz nada caso <quantidade> seja definido como 0. O valor limite para <quantidade> é o decimal 512.

Exemplos:

gni

Retoma a execução até que o ponto de interrupção/controle seja atingido, incluindo o ponto de interrupção temporário definido no endereço da instrução seguinte.

gni 2

Retoma a execução até que o ponto de interrupção/controle seja atingido. Um ponto de interrupção é definido após duas instruções passarem da atual.

Retorna para *Comandos de execução do depurador*

gtime**gt[ime] <milissegundos>**

Retoma a execução. O controle não será devolvido ao depurador até que o tempo interno da emulação tenha decorrido. O intervalo é determinado em milissegundos.

Exemplo:

gtime #10000

Retoma a execução por 10 segundos do tempo de emulação.

Retorna para *Comandos de execução do depurador*.

gvblank**gv[blank]**

Retoma a execução. O controle não será devolvido ao depurador até que um ponto de interrupção ou de controle seja atingido, ou até que se inicie o intervalo de apagamento vertical para uma tela emulada.

Exemplos:

gv

Retoma a execução até que o ponto de interrupção/controle seja atingido ou até que um intervalo de apagamento vertical se inicie.

Retorna para *Comandos de execução do depurador*.

next

n[ext]

Retoma a execução até que uma *CPU* diferente seja agendada. Caso seja ignorada pelo uso dos comandos *ignore* ou *focus* a execução da *CPU* não vai parar.

Exemplo:

n

Retoma a execução, parando quando uma *CPU* diferente que não foi ignorada estiver agendada.

Retorna para *Comandos de execução do depurador*.

focus

focus <CPU>

Foca de forma exclusiva na <CPU> definida ignorando todas as outras. O argumento <CPU> pode ser a etiqueta de um dispositivo ou um número de depuração da *CPU* (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes). É o mesmo que usar o comando *ignore* para ignorar todas as *CPUs* que não seja a *CPU* que foi definida.

Exemplos:

focus 1

Concentre-se exclusivamente na segunda *CPU* do sistema (num índice com base zero), ignorando todas as outras *CPUs*.

focus audiopcb:melodycpu

Concentre-se exclusivamente na *CPU* com o caminho absoluto da etiqueta :audiopcb:melodycpu.

Retorna para *Comandos de execução do depurador*.

ignore

ignore [<CPU>[,<CPU>[,...]]]

Ignora determinadas *CPUs* no depurador. As *CPUs* podem ser definidas através de uma etiqueta ou pelo número da *CPU* no depurador (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes). O depurador nunca mostra a execução para as *CPUs* que forem ignoradas e os pontos de interrupção ou de observação nas *CPUs* ignoradas não têm qualquer efeito. Caso nenhuma *CPUs* seja indicada, as *CPUs* atualmente ignoradas serão listadas. Utilize o comando *observe* para parar de ignorar uma *CPU*.

Observe que você não pode ignorar todas as *CPUs*; pelo menos uma *CPU* deve ser observada o tempo todo.

Exemplos:

```
ignore audiocpu
```

Ignora a CPU com o caminho absoluto da etiqueta :audiocpu ao utilizar o depurador.

```
ignore 2,3,4
```

Ignora a terceira, quarta e quinta *CPU* no sistema (num índice com base zero) ao utilizar o depurador.

```
ignore
```

Lista as CPUs que estiverem sendo ignoradas pelo depurador.

Retorna para *Comandos de execução do depurador*.

observe

observe [*<CPU>* [, *<CPU>* [, ...]]]

Permite a interação com determinada *CPU* no depurador. As *CPUs* podem ser definidas através de etiquetas ou pelo número da *CPU* (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes). Este comando reverte o comando *ignore*. Quando nenhuma *CPUs* for definida, apenas as *CPUs* observadas no momento serão listadas.

Exemplos:

```
observe audiocpu
```

Para de ignorar a CPU com o caminho absoluto da etiqueta :audiocpu ao utilizar o depurador.

```
observe 2,3,4
```

Para de ignorar a 3ª, 4ª, 5ª *CPU* no sistema (num índice com base zero) ao utilizar o depurador.

```
observe
```

Lista as CPUs que estão atualmente sendo observadas pelo depurador.

Retorna para *Comandos de execução do depurador*.

trace

trace {*<nome_do_arquivo>* | off} [, *<CPU>* [, [noloop | logerror] [, *<ação>*]]]

Inicia ou interrompe o rastreamento da execução de determinada *<CPU>* ou da *<CPU>* atualmente visível caso nenhuma tenha sido definida. Para ativar o rastreamento defina o nome do arquivo para o registro do rastreamento no parâmetro *<nome_do_arquivo>*. Para desativar o rastreamento use o termo off no parâmetro *<nome_do_arquivo>*. Quando o argumento *<nome_do_arquivo>* começar com dois chevrons (>>), ele é tratado como uma diretiva para abrir o arquivo para anexar em vez de gravar por cima.

O terceiro parâmetro opcional é um campo sinalizador. As sinalizações compatíveis são noloop e logerror. As diversas sinalizações devem ser separadas por caracteres | (barra vertical). Por padrão, os laços são detectados e condensados numa única linha. Quando a sinalização noloop for definida, os loops não serão

detectados e todas as instruções serão registradas como já executadas. Quando a sinalização `logerror` for definida, a saída do registro de erro será incluída no registro de rastreamento.

O parâmetro opcional `<ação>` é um comando de depuração para ser executado antes que cada mensagem de rastreamento seja registrada. Geralmente, isto incluirá um comando `tracelog` ou `tracesym` incluindo informações adicionais no registro de rastreamento. Observe que talvez seja necessário cercar a ação dentro de chaves `{ }` garantindo que as vírgulas e os ponto-e-vírgulas dentro do comando não sejam interpretadas no contexto do próprio comando `trace`.

Exemplos:

```
trace joust.tr
    Inicia o rastreio da execução na CPU que estiver visível no momento, registrando a saída no arquivo
    joust.tr.
trace dribling.tr,maincpu
    Inicia o rastreio da execução com o caminho absoluto da etiqueta :maincpu:, registrando a saída no
    arquivo dribling.tr.
trace starswep.tr,,nolop
    Inicia o rastreio da execução na CPU que estiver visível no momento, registrando a saída no arquivo
    starswep.tr, com a detecção de loop desativada.
trace starswep.tr,1,logerror
    Inicia o rastreio da execução da segunda CPU do sistema (num índice com base zero), registrando a
    saída junto com a saída do registro de erro no arquivo starswep.tr.
trace starswep.tr,0,logerror|nolop
    Inicia o rastreio da execução da primeira CPU do sistema (num índice com base zero), registrando a
    saída junto com a saída do registro de erro no arquivo starswep.tr, com a detecção de loop
    desativada.
trace >>pigskin.tr
    Inicia o rastreio da execução na CPU que estiver visível no momento, agregando a saída do registro ao
    arquivo pigskin.tr.
trace off,0
    Desativa o rastreamento para a primeira CPU no sistema (num índice com base zero).
trace asteroid.tr,,,{tracelog "A=%02X ",a}
    Inicia o rastreio da execução na CPU que estiver visível no momento, registrando a saída no arquivo
    asteroid.tr. Antes de cada linha registra A=<aval> ao registro de rastreamento.
```

Retorna para *Comandos de execução do depurador*.

traceover

traceover {<nome_do_arquivo>|off}[,<CPU>[,<nolop|logerror>][,<ação>]]

Inicia ou interrompe o rastreamento da execução de determinada <CPU> ou da <CPU> atualmente visível caso nenhuma tenha sido definida. No momento que o retorno a sub-rotina é encontrada, o rastreamento será ignorado

No momento que uma chamada de sub-rotina é encontrada, a sub-rotina será ignorada pelo rastreamento. É usado o mesmo algoritmo que é usado no comando *step over*. Ele não funcionará corretamente com funções recursivas ou caso o endereço retornado não siga imediatamente a instrução da chamada.

Este comando aceita os mesmos parâmetros que o comando *trace*. Favor consultar a seção correspondente para uma descrição mais detalhada das opções e para obter mais exemplos.

Exemplos:

```
traceover joust.tr
```

Inicia o rastreo da execução na CPU que estiver visível no momento, registrando a saída no arquivo joust.tr.

```
traceover dribling.tr,maincpu
```

Inicia o rastreo da execução com o caminho absoluto da etiqueta :maincpu:, registrando a saída no arquivo dribling.tr.

```
traceover starswep.tr,,nolop
```

Inicia o rastreo da execução na CPU que estiver visível no momento, registrando a saída no arquivo starswep.tr, com a detecção de loop desativada.

```
traceover off,0
```

Desativa o rastreamento para a primeira CPU no sistema (num índice com base zero).

```
traceover asteroid.tr,,,{tracelog "A=%02X ",a}
```

Inicia o rastreo da execução na CPU que estiver visível no momento, registrando a saída no arquivo asteroid.tr. Antes de cada linha registra A=<aval> ao registro de rastreamento.

Retorna para *Comandos de execução do depurador*

traceflush

traceflush

Grava no disco todos os arquivos dos registros de rastreamento que estiverem abertos.

Exemplo:

```
traceflush
```

Grava todos os arquivos dos registros de rastreamento.

Retorna para *Comandos de execução do depurador*

8.2.4 Comandos do ponto de interrupção do depurador

bpset

Define o ponto de interrupção¹ no *<endereço>*.

bpclear

Limpa todos ou um determinado ponto de interrupção.

bpdisable

Desativa todos ou um determinado ponto de interrupção.

bpenable

Ativa todos ou um determinado ponto de interrupção.

bplist

Lista todos os ponto de interrupção.

bpset

bp[set] <endereço>[:<CPU>][,<condição>[,<ação>]]

Define uma nova execução do *breakpoint* num determinado *<endereço>*. O *<endereço>* pode ser seguido por dois pontos, uma etiqueta ou o número de depuração da *CPU* para definir um ponto de interrupção para uma determinada *CPU*. Caso nenhuma *CPU* seja informada, o ponto de interrupção será definido na *CPU* que estiver atualmente visível no depurador.

O parâmetro de *<condição>* opcional permite que você possa definir uma expressão que será avaliada toda a vez que o endereço de um ponto de interrupção for alcançado.

Caso o resultado da expressão seja verdadeiro (não zero), o ponto de interrupção suspenderá a execução; caso contrário, a execução continuará sem qualquer notificação. O parâmetro opcional *<ação>* fornece um comando para ser executado sempre que o ponto de interrupção for alcançado e a *<condição>* seja verdadeira. Observe que talvez seja necessário cercar a ação com chaves { } para garantir que as vírgulas e o ponto e vírgula dentro do comando não sejam interpretados no contexto do próprio comando *bpset*.

A cada ponto de interrupção definido é atribuído a um índice numérico que pode ser usado para se referir a ele em outros comandos de ponto de interrupção. Ao longo de uma sessão, os índices do ponto de interrupção são únicos.

Exemplos:

bp 1234

Define um ponto de interrupção na *CPU* que estiver visível e suspenda² uma execução sempre que PC seja igual à 1234.

bp 23456,a0 == 0 && a1 == 0

Define um ponto de interrupção na *CPU* que estiver visível e suspenda uma execução sempre que PC seja igual à 23456 e a expressão (a0 == 0 && a1 == 0) seja verdadeira.

bp 3456:audiocpu,1,{ printf "A0=%08X\n",a0 ; g }

Define um ponto de interrupção na *CPU* com o caminho absoluto da etiqueta :audiocpu que suspenda uma execução sempre que PC seja igual à 3456. Quando isso acontecer, imprime A0=<a0val> no console do depurador e continua a execução.

bp 45678:2,a0==100,{ a0 = ff ; g }

¹ Breakpoint no Inglês.

² Halt no Inglês.

Define um ponto de interrupção na terceira *CPU* do sistema (num índice com base zero) e que suspenderá uma execução sempre que PC seja igual à 45678 e a expressão `a0 == 100` seja verdadeira. Quando isso acontecer, define `a0` para `ff` e continua a execução.

```
temp0 = 0 ; bp 567890,++temp0 >= 10
```

Define um ponto de interrupção na *CPU* que estiver visível e suspenda uma execução sempre que PC seja igual à 567890 e a expressão `++temp0 >= 10` seja verdadeira. Isso apenas será interrompido de forma efetiva depois que o ponto de interrupção for alcançado 16 vezes.

Retorna para *Comandos do ponto de interrupção do depurador*.

bpclear

bpclear [*<bnum>* [,...]]

Limpa todos os pontos de interrupção. Caso um *<bnum>* seja definido, apenas os pontos de interrupção relacionados a ele serão limpos, caso contrário, todos serão.

Exemplos:

```
bpclear 3
```

Limpa o ponto de interrupção com o índice 3.

```
bpclear
```

Limpa todos os pontos de interrupção.

Retorna para *Comandos do ponto de interrupção do depurador*.

bpdisable

bpdisable [*<bnum>* [,...]]

Desativa os pontos de interrupção. Caso um *<bnum>* seja definido, apenas os pontos de interrupção citados serão desativados, caso contrário, todos serão.

Observe que ao desativar um ponto de interrupção ele não é eliminado, apenas marca temporariamente o ponto de interrupção como inativo. Os pontos de interrupção desativados não causarão a suspensão da execução, as expressões relacionadas com as suas respectivas condições não serão avaliadas e seus comandos relacionados não serão executados.

Exemplos:

```
bpdisable 3
```

Desativa o ponto de interrupção com o índice 3.

```
bpdisable
```

Desativa todos os pontos de interrupção.

Retorna para *Comandos do ponto de interrupção do depurador*.

bpenable

bpenable [*<bpnum>* [, ...]]

Ativa os pontos de interrupção. Caso um *<bpnum>* seja definido, apenas os pontos de interrupção citados serão ativados, caso contrário, todos serão.

Exemplos:

```
bpenable 3
```

Ativa o ponto de interrupção com o índice 3.

```
bpenable
```

Ativa todos os pontos de interrupção.

Retorna para *Comandos do ponto de interrupção do depurador*.

bplist

bplist [*<CPU>*]

Lista os pontos de interrupção existentes juntamente com seus índices e quaisquer outras condições ou ações associadas. Caso nenhuma *<CPU>* esteja definida, serão listados os pontos de interrupção para todas as *CPUs* do sistema, caso contrário, serão listados apenas os pontos de interrupção numa determinada *CPU*. A *<CPU>* pode ser definida pela etiqueta ou pelo número da *CPU* no depurador (consulte o capítulo *Determinando os dispositivos e as faixas de endereço* para obter mais informações).

Exemplos:

```
bplist
```

Lista todos os pontos de interrupção.

```
bplist .
```

Lista todos os pontos de interrupção em todas as *CPUs* disponíveis.

```
bplist maincpu
```

Lista todos os pontos de interrupção na *CPU* com o caminho absoluto da etiqueta `:maincpu`.

Retorna para *Comandos do ponto de interrupção do depurador*.

8.2.5 Comandos do ponto de controle do depurador

wpset

Define o ponto de controle do acesso da memória.

wpclear

Apaga os pontos de controle.

wpdisable

Desativa os pontos de controle.

wpenable

Ativa os pontos de controle.

wplist

Lista os pontos de controle.

Os pontos de controle¹ interrompe a execução do depurador quando uma *CPU* acessa um ponto numa determinada faixa da memória.

wpset

wp[*{d|i|o}*][set] <endereço>[:<faixa>],<comprimento>,<tipo>[,<condição>[,<ação>]]

Define um novo ponto de controle iniciando em determinado <endereço> que se estende através do parâmetro <comprimento>. Toda a faixa do ponto de controle é o <endereço> através do <endereço> + <comprimento> - 1, inclusive. O <endereço> pode ser opcionalmente seguido pela faixa de endereços da *CPU* (consulte [Determinando os dispositivos e as faixas de endereço](#) para obter mais detalhes). Quando nenhuma faixa de endereços for definida, o sufixo do comando define a faixa do endereço. O comando *wpset* retorna a primeira faixa de endereços que for exposta pela *CPU*, *wpdset* retorna para a faixa do índice 1 (dados), *wpiset* retorna para a faixa do índice 2 (E/S), já *wposet* retorna para a faixa do índice 3 (*opcodes*). O parâmetro <tipo> determina o tipo de acesso que serão retidos, ele pode ser um destes 3 valores *r* para os acessos de leitura, *w* para os acessos para a escrita ou *rw* para ambos os acessos de leitura e escrita.

Já o parâmetro opcional <condição> permite que uma expressão seja definida para que ela seja avaliada cada vez que um ponto de controle seja alcançado. Quando o resultado da expressão for verdadeiro (não zero) o ponto de controle interromperá a execução, caso contrário, a execução continuará sem qualquer notificação. O parâmetro opcional <ação> oferece um comando para ser executado sempre que um ponto de controle seja alcançado e a <condição> seja verdadeira. Observe que talvez seja necessário cercar a ação dentro de chaves { } garantindo que as vírgulas e os ponto-e-vírgulas dentro do comando não sejam interpretadas no contexto do próprio comando *wpset*.

Cada ponto de controle que for definido é atribuído a um índice numérico que pode ser utilizado como referência em outros comandos do ponto de controle. Durante a sessão os índices do ponto de controle são únicos.

Duas variáveis estão disponíveis para tornar as expressões <condição> mais uteis: para todos os pontos de controle, a variável *wpaddr* é definido no endereço de acesso que foi alcançado; para a gravação no ponto de controle, a variável *wpdata* é definida nos dados que estão sendo gravados.

Exemplos:

wp 1234,6,rw

Define um ponto de controle que interromperá uma execução na *CPU* que estiver visível sempre que uma leitura ou escrita acontecer na primeira faixa de endereço entre 1234-1239.

¹ Watchpoint no Inglês.

`wp 23456:data,a,w,wpdata == 1`

Define um ponto de controle que interromperá uma execução na *CPU* que estiver visível sempre que uma gravação ocorrer no espaço data na faixa de endereço entre 23456-2345f e a gravação dos dados seja igual à 1.

`wp 3456:maincpu,20,r,1,{ printf "Read @ %08X\n",wpaddr ; g }`

Define um ponto de controle na *CPU* com o caminho absoluto da etiqueta :maincpu a execução será interrompida sempre que ocorrer uma leitura na primeira faixa de endereço entre 3456-3475. Quando ocorrer, imprima Read @ <wpaddr> no console do depurador e a execução é resumida.

`temp0 = 0 ; wp 45678,1,w,wpdata==f0,{ temp0++ ; g }`

Define um ponto de controle na *CPU* que estiver visível que interromperá a execução sempre que ocorrer uma gravação no endereço 45678 e onde o valor que será escrito seja igual à f0. Quando ocorrer, incremente o valor temp0 e a execução é resumida.

Retorna para *Comandos do ponto de controle do depurador*.

wpclear

wpclear [<wpnum>[,...]]

Apaga os pontos de controle. Quando o <wpnum> for definido, as referências dos pontos de controle serão apagadas. Ao não ser definido, todos os pontos de controle serão apagados.

Exemplos:

`wpclear 3`

Apaga o ponto de controle com o índice 3.

`wpclear`

Apaga todos os pontos de controle.

Retorna para *Comandos do ponto de controle do depurador*.

wpdisable

wpdisable [<wpnum>[,...]]

Desativa os pontos de controle. Quando o <wpnum> for definido, as referências dos pontos de controle serão desativadas. Ao não ser definido, todos os pontos de controle serão desativados.

Observe que ao desativar um ponto de controle ele não é excluído, marca temporariamente o ponto de controle como inativo. Os pontos de controle que forem desativados não causam a interrupção da execução, as condições associadas às expressões não serão avaliadas e seus respectivos comandos não serão executados.

Exemplos:

`wpdisable 3`

Desativa o ponto de controle com o índice 3.

`wpdisable`

Desativa todos os pontos de controle.

Retorna para *Comandos do ponto de controle do depurador*.

wpenable

wpenable [<wpnum>[,...]]

Ativa os pontos de controle. Quando o <wpnum> for definido, as referências dos pontos de controle serão ativadas. Ao não ser definido, todos os pontos de controle serão ativados.

Exemplos:

```
wpenable 3
```

Ativa o ponto de controle com o índice 3.

```
wpenable
```

Ativa todos os pontos de controle.

Retorna para *Comandos do ponto de controle do depurador*.

wplist

wplist [<CPU>]

Lista os pontos de controle atuais junto com seus índices e quaisquer ações ou condições associadas. Quando nenhuma <CPU> for definida, os pontos de controle em todas as CPUs do sistema serão listadas, ao ser definida, apenas os pontos de controle para esta CPU será listada. A <CPU> pode ser determinada por uma etiqueta ou através de um número do depurador. (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes).

Exemplos:

```
wplist
```

Lista todos os pontos de controle.

```
wplist .
```

Lista todos os pontos de controle para a CPU que estiver visível.

```
wplist maincpu
```

Lista todos os pontos de controle para a CPU com o caminho absoluto da etiqueta :maincpu.

Retorna para *Comandos do ponto de controle do depurador*.

8.2.6 Comandos do ponto de identificação

rpset

Define um ponto de identificação¹ que será disparado quando atingir uma *<condição>*.

rpclear

Limpa os pontos de identificação.

rpdisable

Desativa um ponto de identificação.

rpenable

Ativa um ponto de identificação.

rplist

Lista os pontos de identificação.

Os pontos de identificação avaliam uma expressão cada vez que a *CPU* executa uma instrução interrompendo-a e ativando o depurador caso o seu resultado seja verdadeiro (igual à 1).

rpset

rp[set] <condição> [, <ação>]

Define um novo ponto de identificação que será iniciado quando a *<ação>* informada for verdadeira (igual à 1). Para prevenir que a *<condição>* seja interpretada como uma atribuição, ela deve ser cercada por { }. O parâmetro opcional *<ação>* oferece um comando que será executado sempre que um ponto de identificação for disparado. Observe que talvez seja necessário cercar a ação dentro de chaves { } garantindo que as vírgulas e os ponto-e-vírgulas dentro do comando não sejam interpretadas no contexto do próprio comando *rpset*.

Cada ponto de identificação que for definido será atribuído a um índice numérico que pode ser utilizado como referência em outros comandos do ponto de identificação e seus índices são únicos durante a seção.

Exemplos:

rp {PC==150}

Define um ponto de identificação que irá interromper a execução, sempre que o registro **PC** for igual à 150.

temp0=0; rp {PC==150},{temp0++; g}

Define um ponto de identificação que aumentará o valor da variável *temp0* sempre que o registro **PC** for igual à 150.

rp {temp0==5}

Define um ponto de identificação que irá interromper a execução sempre que a variável *temp0* for igual à 150.

Retorna para *Comandos do ponto de identificação*.

¹ Registerpoint no Inglês.

rpclear

rpclear [*<rpnum>*, [...]]

Limpa os pontos de identificação. Quando um *<rpnum>* é definido, apenas essa referência será limpa, caso contrário, todos serão.

Exemplos:

```
rpclear 3
```

Limpa o ponto de identificação com o índice 3.

```
rpclear
```

Limpa todos os pontos de identificação.

Retorna para *Comandos do ponto de identificação*.

rpdisable

rpdisable [*<rpnum>*, [...]]

Desativa os pontos de identificação. Quando um *<rpnum>* é definido, apenas essa referência será desativada, caso contrário, todas serão.

Observe que ao desativar um ponto de identificação ele não é excluído, apenas o marca temporariamente como inativo. Os pontos de identificação que forem desativados não causam a interrupção da execução, as condições associadas às expressões não serão avaliadas e seus respectivos comandos não serão executados.

Exemplos:

```
rpdisable 3
```

Desativa o ponto de identificação com o índice 3.

```
rpdisable
```

Desativa todos os pontos de identificação.

Retorna para *Comandos do ponto de identificação*.

rpenable

rpenable [*<rpnum>*, [...]]

Ativa os pontos de identificação. Quando um *<rpnum>* é definido, apenas essa referência será ativada, caso contrário, todas serão.

Exemplos:

```
rpenable 3
```

Ativa o ponto de identificação com o índice 3.

```
rpenable
```

Ativa todos os pontos de identificação.

Retorna para *Comandos do ponto de identificação*.

rplist

rplist [<CPU>]

Lista todos os pontos de identificação atuais junto com seus respectivos índices, condições e ações associadas. Quando nenhuma <CPU> é definida, os pontos de identificação de todas as CPUs do sistema serão listadas. A <CPU> pode ser determinada por uma etiqueta ou pelo número da CPU no depurador (consulte *Determinando os dispositivos e as faixas de endereço* para obter mais detalhes).

Exemplos:

`rplist`

Lista todos os pontos de identificação.

`rplist .`

Lista todos os pontos de identificação na CPU que estiver visível.

`rplist maincpu`

Lista todos os pontos de identificação na CPU com o caminho absoluto da etiqueta :maincpu.

Retorna para *Comandos do ponto de identificação*.

8.2.7 Comandos de depuração do ponto de exceção

epset

define um novo ponto de exceção.

epclear

limpa um ponto de exceção específico ou todos.

epdisable

desativa um ponto de exceção específico ou todos.

epenable

ativa um ponto de exceção específico ou todos.

eplist

lista os pontos de exceção.

Os pontos de exceção interrompem a execução e ativam a depuração quando uma CPU gera uma exceção com número específico.

epset

ep[set] <tipo> [, <condição> [, <ação>]]

Define um novo ponto de exceção do tipo <tipo>. O parâmetro da opção <condição> permite que você defina uma expressão que pode ser avaliada cada vez que um ponto de exceção for atingido. Quando o resultado da expressão for verdadeiro (não zero), o ponto de exceção na verdade irá interromper a execução no início da execução do manipulador; caso contrário, a execução continuará sem qualquer outra notificação. O parâmetro da opção <ação> permite que um comando seja executado sempre que o ponto de exceção for atingido e a <condição> seja verdadeira. Observe que talvez seja necessário incorporar a ação entre chaves {} visando que as vírgulas e o ponto e vírgula sejam interpretados como se fossem aplicados ao próprio comando epset.

A quantidade de exceções dependem do tipo da CPU. As causas das exceções podem incluir interrupções vetorizadas sejam elas internas ou externas, assim como, erros que ocorrem nas instruções e nas chamadas do sistema.

Cada ponto de exceção que for definido, recebe um índice que pode ser usado em outros comandos de ponto de exceção para fazer uma referência a este ponto de exceção.

Exemplos:

ep 2

Define um ponto de exceção que interromperá a execução sempre que a CPU visível gerar uma exceção número 2.

Retorna para *Comandos de depuração do ponto de exceção*

epclear

epclear [<epnum>[,...]]

O comando *epclear* limpa os pontos de exceção. Quando um <epnum> é definido, são limpos apenas os pontos de exceção requisitados, caso contrário, todos os pontos de exceção serão limpos.

Exemplos:

```
epclear 3
```

Limpa o índice 3 do ponto de exceção.

```
epclear
```

Limpa todos os pontos de exceção.

Retorna para *Comandos de depuração do ponto de exceção*

epdisable

epdisable [<epnum>[,...]]

O comando *epdisable* desativa os pontos de exceção. Quando um <epnum> é definido, apenas são desativados os pontos de exceção requisitados, caso contrário, todos os pontos de exceção serão desativados. Observe que ao desativar um ponto de exceção, ele não é excluído, apenas o marca temporariamente como inativo.

Exemplos:

```
epdisable 3
```

Desativa o índice 3 do ponto de exceção.

```
epdisable
```

Desativa todos os pontos de exceção.

Retorna para *Comandos de depuração do ponto de exceção*

epenable

epenable [<epnum>[,...]]

O comando *epenable* ativa os pontos de exceção. Quando um <epnum> é definido, apenas são ativados os pontos de exceção requisitados, caso contrário, todos os pontos de exceção serão ativados.

Exemplos:

```
epenable 3
```

Ativa o índice 3 do ponto de exceção.

```
epenable
```

Ativa todos os pontos de exceção.

Retorna para *Comandos de depuração do ponto de exceção*

eplist

eplist

O comando *eplist* lista todos os pontos de exceção atuais, junto com seus respectivos índices, condições ou ações que forem anexadas a eles.

Retorna para *Comandos de depuração do ponto de exceção*

8.2.8 Comandos da anotação do código do depurador

comadd

Inclui um comentário num determinado endereço do código desmontado.

comdelete

Remove o comentário de um determinado endereço.

comsave

Salva o comentário atual num arquivo.

comlist

Imprime os comentários disponíveis a partir de um arquivo.

commit

Aplica um *comadd* seguido de *comsave*

comadd

comadd <endereço>, <comentário>

Define o comentário indicado num determinado endereço do código desmontado na *CPU* que estiver visível. Este comando pode ser abreviado para *//*.

Exemplos:

```
comadd 0,hello world.
```

Inclui o comentário "hello world." ao código no endereço 0.

```
// 10,opcode não documentado!
```

Inclui o comentário "**opcode não documentado!**" ao código no endereço 10.

Retorna para *Comandos da anotação do código do depurador*.

comdelete

comdelete

Exclui determinado endereço na *CPU* que estiver visível.

Exemplo:

```
comdelete 10
```

Exclui o comentário do código no endereço 10 na *CPU* que estiver visível.

Retorna para *Comandos da anotação do código do depurador*.

comsave

Grava os comentários atuais no arquivo de comentários XML para o sistema emulado. Este arquivo será carregado pelo depurador na próxima vez em que o sistema for executado com a depuração ativada. A opção que define onde estes arquivos são salvos é definido pela opção *comment_directory*.

Exemplo:

comsave

Grava os comentários atuais no arquivo de comentários do sistema.

Retorna para *Comandos da anotação do código do depurador*.

comlist

Faz a leitura dos comentários armazenados no arquivo de comentário XML para o sistema emulado e os imprime no console de depuração. Este comando não afeta os comentários da sessão atual pois faz a leitura direta do arquivo. A opção que define onde estes arquivos são salvos é definido pela opção *comment_directory*.

Exemplo:

comlist

Mostra os comentários armazenados num arquivo de comentário do sistema.

Retorna para *Comandos da anotação do código do depurador*.

commit

commit <endereço>,<comentário>

Define o comentário indicado em determinado endereço no código desmontado na CPU que estiver visível e grava os comentários no arquivo para o sistema que estiver sendo emulado no momento. (é equivalente ao comando *comadd* seguido de *comsave*). Este comando pode ser abreviado para */**.

Exemplos:

commit 0,hello world.

Inclui o comentário "hello world." ao código no endereço 0 na CPU que estiver visível e salva os comentários.

/* 10,opcode não documentado!

Inclui o comentário "opcode não documentado!" ao código no endereço 10 na CPU que estiver visível e salva os comentários.

Retorna para *Comandos da anotação do código do depurador*.

8.2.9 Comandos para depuração de trapaça

Nota: **Pesquisa** é quando se deseja levantar mais informações sobre algo desconhecido. **Busca** é quando se deseja levantar informações sobre algo que já se sabe.

cheatinit

Inicializa uma pesquisa da trapaça na região selecionada da memória.

cheatrange

Adiciona uma região selecionada da memória para a pesquisa da trapaça.

cheatnext

Filtra os candidatos da trapaça ao comparar com os valores anteriores.

cheatnextf

Filtra os candidatos da trapaça ao comparar com os valores iniciais.

cheatlist

Mostra a lista das trapaças que tiveram correspondências ou salve-os num arquivo.

cheatundo

Desfaz a última trapaça que foi buscada (estado apenas).

O depurador inclui uma funcionalidade simples de busca por trapaça que funciona salvando o conteúdo da memória e depois filtrando os locais de acordo com a maneira que os valores se alteram.

Demonstraremos o uso da funcionalidade da busca por uma trapaça de vida infinita para o sistema **Raiden** (raidен):

- Inicie o jogo com o depurador ativo (`mame raiden -window -debug`). Na **janela do depurador** pressione F5 para que o jogo rode e clique na **janela do jogo**. Insira um crédito com 5 e inicie a partida pressionando a tecla 1, entre no depurador clicando na tecla " ou ' (fica do lado esquerdo da tecla 1 ou consulte *a mapa do teclado*).
- Clique na **janela do depurador**, tenha certeza que a CPU principal :maincpu esteja visível no topo da janela do depurador, inicie uma pesquisa por valores 8-bit sem assinatura usando o comando *cheatinit command*, digite o comando na parte debaixo da janela do depurador seguido da tecla Enter:

```
>cheatinit ub
36928 cheat locations initialized for NEC V30 ':maincpu' program space
```

- Pressione F5 para que o jogo continue, clique na **janela do jogo** e perca 1 vida, pressione novamente a tecla " para interromper o jogo, clique na **janela do depurador**.
- Use o comando *cheatnext command* para filtrar as localizações onde o valor tenha sido reduzido por 1:

```
>cheatnext -,1
19 cheats found
```

- Pressione F5 para que o jogo continue novamente, clique na **janela do jogo** e perca mais 1 vida, novamente pressione a tecla ", clique na **tela do depurador** e repita o comando anterior:

```
>cheatnext -,1
Address=00B85 Start=03 Current=01
1 cheats found
```

- Use o comando *cheatlist* para salvar o endereço encontrado num arquivo na pasta principal do MAME:

```
>cheatlist raiden-j1-vidas.xml
```

- O arquivo terá um fragmento em formato XML com o endereço e todos os valores encontrados para a trapaça:

```
<cheat desc="Possibility 1: 00B85 (01)">
  <script state="run">
    <action>:main*CPU*.pb@0x00B85=0x03</action>
  </script>
</cheat>
```

cheatinit

cheatinit [[<senalização> [<largura> [<ordem>]]], [<endereço>, <comprimento> [, <espaço>]]]

Inicia a busca pela trapaça nas áreas da RAM que possam ser escritas na região do endereços indicado. Pode ser abreviado para *ci*.

O primeiro argumento determina o formato do dado que será buscado. A <senalização> (*sign*) pode ser *u* para sem assinatura (*unsigned*) ou *s* para assinado, <largura> (*width*) pode ser *b* para 8-bit (byte), *w* para **16-bit** (*word*), *d* para **32-bit** (*double word*) ou *q* para **64-bit** (*quadruple word*); <ordem> (*swap*) pode ser *s* para um byte com a ordem invertida. Quando o primeiro argumento for omitido ou estiver vazio, será usado o formato do dado da busca anterior ou um formato com **8-bit** sem assinatura caso este seja o formato usado na primeira busca.

O <endereço> (*address*) determina o endereço para o início da pesquisa e o <comprimento> (*length*) determina a quantidade da memória que será utilizada para esta pesquisa.

Caso seja definido, será pesquisada a RAM gravável no intervalo do <endereço> até o <endereço> + <comprimento> - 1; caso contrário, toda a RAM gravável será pesquisada na faixa do endereço.

Consulte *Determinando os dispositivos e as faixas de endereço* para mais detalhes em como determinar a faixa de endereços. Caso a região do endereço não seja definido, ele retorna para a primeira faixa de endereços que for exposto pela primeira CPU que estiver visível.

Exemplos:

```
cheatinit ub,0x1000,0x10
```

Inicializa a busca pela trapaça com valores **8-bit** sem assinatura nos endereços 0x1000-0x100f na região do programa visível da CPU.

```
cheatinit sw,0x2000,0x1000,1
```

Inicializa a busca pela trapaça com valores **16-bit** com assinatura nos endereços 0x2000-0x2fff na região do programa visível pela 2ª CPU do sistema (num índice com base zero).

```
cheatinit uds,0x0000,0x1000
```

Inicializa a busca pela trapaça com valores **64-bit** sem assinatura com a ordem invertida dos bytes nos endereços 0x0000-0xffff na região visível do programa na CPU.

Retorna para *Comandos para depuração de trapaça*.

cheatrange

cheatrange <endereço>, <comprimento>

Acrescente as áreas da RAM que podem ser escritas na busca por trapaças. Pode ser abreviado para *cr*. Antes de utilizar este comando, o comando *cheatinit* deve ser utilizado para inicializar a busca pela trapaça, para definir o espaço do endereçamento e o formato dos dados.

O <endereço> determina o endereço a partir de onde a busca deve começar, o <comprimento> determina a quantidade de memória que se deseja pesquisar. A RAM gravável na faixa <endereço> até <endereço> + <comprimento> - 1, será adicionada às áreas que serão pesquisadas.

Exemplos:

```
cheatrange 0x1000,0x10
```

Adiciona os endereços 0x1000-0x100f nas regiões para a busca das trapaças.

Retorna para *Comandos para depuração de trapaça*.

cheatnext

cheatnext <condição> [, <valor_para_comparação>]

Faz o filtro dos candidatos comparando com os valores das buscas anteriores. Caso restem cinco ou menos candidatos, eles serão mostrados no console de depuração. Este comando pode ser abreviado para *cn*.

Argumentos possíveis para <condição>:

all

Use para atualizar o último valor sem alterar as correspondências atuais (o <valor_para_comparação> não é usado).

equal (eq)

Sem o <valor_para_comparação>, procure pelos valores que sejam iguais a pesquisa anterior; com o <valor_para_comparação>, busque pelos valores que sejam iguais ao <valor_para_comparação>.

notequal (ne)

Sem o <valor_para_comparação>, pesquise pelos valores que não sejam iguais a pesquisa anterior; com o <valor_para_comparação>, busque pelos valores que não sejam iguais ao <valor_para_comparação>.

decrease (de, -)

Sem o <valor_para_comparação>, pesquise pelos valores que foram reduzidos desde a pesquisa anterior; com o <valor_para_comparação>, busque por valores que diminuam com base no <valor_para_comparação> desde a última pesquisa.

increase (in, +)

Sem o <valor_para_comparação>, pesquise pelos valores que aumentaram desde a pesquisa anterior; com o <valor_para_comparação>, busque por valores que aumentaram com base no <valor_para_comparação> desde a última pesquisa.

decreaseorequal (deeq)

Busca pelos valores que foram reduzidos ou que não tenham se alterado desde a pesquisa anterior (o <valor_para_comparação> não é usado).

increaseorequal (ineq)

Busca pelos valores que tenham aumentado ou que não tenham se alterado desde a pesquisa anterior (o <valor_para_comparação> não é usado).

smallerof (lt, <)

Busca pelos valores que sejam menores que o <valor_para_comparação> (o <valor_para_comparação> é obrigatório).

greaterof (gt, >)

Busca pelos valores que sejam maiores que o <valor_para_comparação> (o <valor_para_comparação> é obrigatório).

changedby (ch, ~)

Busca pelos valores que tenham se alterado com base no <valor_para_comparação> desde a pesquisa anterior (o <valor_para_comparação> é obrigatório).

Exemplos:

cheatnext increase

Busca todos os valores que tenham aumentado desde a pesquisa anterior.

cheatnext decrease,1

Busca todos os valores que foram reduzidos por 1 desde a pesquisa anterior.

Retorna para [Comandos para depuração de trapaça](#).

cheatnextf

cheatnextf <condição>[,<valor_para_comparação>]

Faz o filtro dos candidatos comparando com os valores iniciais das buscas. Caso restem cinco ou menos candidatos, eles serão mostrados no console de depuração. Pode ser abreviado para cnf.

Argumentos possíveis para <condição>:

all

Use para atualizar o último valor sem alterar as correspondências atuais (o <valor_para_comparação> não é usado).

equal (eq)

Sem o <valor_para_comparação>, procure pelos valores que sejam iguais a pesquisa inicial; com o <valor_para_comparação>, busque pelos valores que sejam iguais ao <valor_para_comparação>.

notequal (ne)

Sem o <valor_para_comparação>, pesquise pelos valores que não sejam iguais a pesquisa inicial; com o <valor_para_comparação>, busque pelos valores que não sejam iguais ao <valor_para_comparação>.

decrease (de, -)

Sem o <valor_para_comparação>, pesquise pelos valores que foram reduzidos desde a pesquisa inicial; com o <valor_para_comparação>, busque por valores que diminuam com base no <valor_para_comparação> desde a última pesquisa.

increase (in, +)

Sem o `<valor_para_comparação>`, pesquise pelos valores que aumentaram desde a pesquisa inicial; com o `<valor_para_comparação>`, busque por valores que aumentaram com base no `<valor_para_comparação>` desde a última pesquisa.

`decreaseorequal (deeq)`

Busca pelos valores que foram reduzidos ou que não tenham se alterado desde a pesquisa inicial (o `<valor_para_comparação>` não é usado).

`increaseorequal (ineq)`

Busca pelos valores que tenham aumentado ou que não tenham se alterado desde a pesquisa inicial (o `<valor_para_comparação>` não é usado).

`smallerof (lt, <)`

Busca pelos valores que sejam menores que o `<valor_para_comparação>` (o `<valor_para_comparação>` é obrigatório).

`greaterof (gt, >)`

Busca pelos valores que sejam maiores que o `<valor_para_comparação>` (o `<valor_para_comparação>` é obrigatório).

`changedby (ch, ~)`

Busca pelos valores que tenham se alterado com base no `<valor_para_comparação>` desde a pesquisa anterior (o `<valor_para_comparação>` é obrigatório).

Exemplos:

`cheatnextf increase`

Busca todos os valores que tenham aumentado desde a pesquisa inicial.

`cheatnextf decrease,1`

Busca todos os valores que foram reduzidos por 1 desde a pesquisa inicial.

Retorna para [Comandos para depuração de trapaça](#).

cheatlist

cheatlist [`<nome_do_arquivo>`]

Sem o `<nome_do_arquivo>`, mostre as trapaças encontradas no momento no console do depurador; com o `<nome_do_arquivo>`, salve as trapaças encontradas num formato XML para o arquivo determinado. Pode ser abreviado para `cl`.

Exemplos:

`cheatlist`

Mostra o que foi encontrado no console.

`cheatlist cheat.xml`

Grava o que foi encontrado no arquivo `cheat.xml` em formato XML.

Retorna para [Comandos para depuração de trapaça](#).

cheatundo**cheatundo**

Desfaz o filtro da trapaça pelo mais recente comando *cheatnext* ou *cheatnextf*. Observe que os valores anteriores **não** retrocedem. Pode ser abreviado com cu.

Exemplos:

cheatundo

Restaura os candidatos filtrados pelo comando *cheatnext* ou *cheatnextf* mais recente.

Retorna para *Comandos para depuração de trapaça*.

8.2.10 Comandos para a depuração de imagens

images

Lista todos os dispositivos de imagem e todas as que estiverem montadas.

mount

Monta o arquivo da imagem num determinado dispositivo.

unmount

Desmonta a imagem.

images

images

Faz a listagem dos nomes das instâncias dos dispositivos de imagens de mídia no sistema e as imagens montadas no momento, caso haja alguma. Os nomes curtos das instâncias também são listados, assim como é permitido através da linha de comando. Os itens da lista de programa que estiverem montados são exibidos com um nome na lista, com um nome curto do programa e da sua parte separados por dois-pontos; outras imagens que estiverem montadas são mostradas como nome de arquivo.

Exemplo:

images

Lista o nome da imagem e as imagens que estiverem montadas.

Retorna para *Comandos para a depuração de imagens*.

mount

mount <instância>,<nome_do_arquivo>

Monta um arquivo num dispositivo. O dispositivo pode ser especificado pela sua instância ou pelo seu nome curto, assim como é permitido através da linha de comando.

Alguns dispositivos de mídia permitem que os itens da lista de programas sejam montados usando este comando, ao oferecer um nome curto do item da lista no lugar parâmetro <nome_do_arquivo>.

Exemplos:

```
mount flop1,os1xutls.td0
```

Monta o arquivo os1xutls.td0 no dispositivo com o nome flop1.

```
mount cart,10yard
```

Monta o item da lista de programas 10yard no dispositivo cart.

Retorna para *Comandos para a depuração de imagens*.

unmount**unmount <instance>**

Desmonta a imagem que estiver montada no dispositivo (caso haja alguma). O dispositivo pode ser informado pelo seu nome completo ou nome curto, assim como é permitido através da linha de comando.

Exemplo:

```
unmount cart
```

Desmonta qualquer imagem que estiver montada com o nome cart.

Retorna para *Comandos para a depuração de imagens*.

8.3 Determinando os dispositivos e as faixas de endereço

Os diversos comandos de depuração aceitam parâmetros que determinam em qual dispositivo se vai operar. Caso um dispositivo não seja definido de forma explícita, a *CPU* que será utilizada é a que estiver visível no momento. Os dispositivos podem ser definidos através de uma etiqueta ou através do número da *CPU* no depurador:

- As etiquetas são os caminhos separados por dois pontos (:) que o MAME utiliza para identificar os dispositivos dentro de um sistema (:maincpu por exemplo). Você os vê nas opções de configuração dos dispositivos de slot, nas listas de origem da desmontagem do depurador, no visualizador da memória e em vários outros lugares dentro da interface do MAME.
- A numeração da *CPU* é um incremento monotônico dos números que o depurador atribui aos dispositivos do tipo *CPU* dentro de um sistema, começando por zero. O símbolo cpunum contém o número da *CPU* que estiver visível no momento (você pode vê-la usando o comando `print cpunum` no console).

Quando uma etiqueta é iniciada com um circunflexo ^ ou ponto ., ela é interpretada com relação a *CPU* que estiver visível no momento, caso contrário, ela é interpretada com relação ao dispositivo raiz do sistema. Quando um argumento do dispositivo for ambíguo como uma etiqueta ou um número da *CPU*, ele será interpretado como uma etiqueta.

Exemplos:

maincpu

O dispositivo com a etiqueta absoluta :maincpu.

^melodypsg

O dispositivo irmão da *CPU* visível com a etiqueta melodypsg.

.:adc

O sub-dispositivo da *CPU* visível com a etiqueta adc.

2

O terceiro dispositivo do tipo *CPU* no sistema (num índice com base zero).

Os comandos que operam na memória estendem isso ao permitir que a etiqueta do dispositivo ou o número da *CPU* seja opcionalmente seguido por um identificador na região do endereçamento, ele são cadeias de caracteres do tipo etiqueta. Você pode vê-los nas listas de origem do visualizador da memória do depurador. Quando o identificador da região do endereçamento for omitido, uma região predefinida do endereço será usada. Normalmente, esta é a região que aparece primeiro para no dispositivo. Muitos comandos têm variantes com sufixos d, i e o (dados, E/S e *opcodes*) que, por padrão, são as regiões do endereçamento nos índices 1, 2 e 3, respectivamente, pois estes têm um significado especial nos dispositivos do tipo *CPU*.

Em casos ambíguos, a região predefinida do endereçamento de um sub-dispositivo será usado no lugar de uma determinada região específica do endereço.

Exemplos:

ram

A região do endereçamento padrão com o caminho absoluto da etiqueta na região :ram ou a ram da região visível da *CPU*.

.:io

A região do endereçamento padrão do sub-dispositivo da *CPU* visível com a etiqueta :io ou o io da região visível da *CPU*.

:program

A região do endereçamento padrão com o caminho absoluto da etiqueta na região :program ou o program no dispositivo raiz do sistema.

^vdp

A região do endereçamento padrão do dispositivo irmão da região visível da *CPU* com a etiqueta vdp.

^:data

A região do endereçamento padrão do dispositivo irmão da região visível da *CPU* com a etiqueta data ou data na região do dispositivo principal da região visível da *CPU*.

1:rom

A região do endereçamento padrão do sub-dispositivo da *CPU* visível da 2ª *CPU* do sistema (num índice com base zero) com a etiqueta :rom ou da rom da 2ª *CPU* do sistema.

2

A região do endereçamento padrão do 3º dispositivo tipo *CPU* no sistema (num índice com base zero).

Quando um comando toma como parâmetro um endereço emulado da memória, o endereço pode opcionalmente ser seguido por uma determinação na faixa de endereços, conforme descrito acima.

Exemplos:

0220

O endereço 0220 na faixa de endereço padrão da região visível da *CPU*.

0378:io

O endereço 0378 na faixa de endereço padrão do dispositivo com a etiqueta absoluta :io ou io do endereçamento da região visível da *CPU*.

1234:.:rom

O endereço 1234 na faixa de endereço padrão no dispositivo relacionado da região visível da *CPU* com a etiqueta :rom ou da rom do endereçamento da região visível da *CPU*.

1260:^vdp

O endereço 1260 na faixa de endereço padrão do dispositivo irmão da região visível da *CPU* com a etiqueta vdp.

8008:^:data

O endereço 8008 na faixa de endereço padrão do dispositivo irmão da região visível da *CPU* com a etiqueta :data ou data na região do dispositivo principal da região visível da *CPU*.

9660:.:ram

O endereço 9660 na faixa de endereço padrão do dispositivo com a etiqueta absoluta :ram ou ram no dispositivo raiz do sistema.

Os exemplos aqui incluem muitos atalhos, mas geralmente, o depurador deve tomar o significado mais provável para um dispositivo ou uma especificação na região do endereçamento.

8.4 A sintaxe das expressões do depurador

As expressões podem ser usadas em qualquer lugar onde um parâmetro numérico ou um parâmetro booleano seja esperado. A sintaxe das expressões é semelhante a um subconjunto da sintaxe de expressão no estilo C, com precedência total do operador e dos parênteses. Faltam alguns operadores (notadamente o operador condicional ternário) e alguns novos (assessores da memória).

A tabela abaixo lista todos os operadores, ordenados da mais alta para a mais baixa prioridade:

()	Parenteses padrão.
++ --	Postfix aumento/redução.
++ -- ~ ! - + b@ w@ d@ q@ b! w! d! q!	Prefix aumento/redução, complemento binário, complemento lógico, identidade unária/negação, acesso da memória.
* / %	Multiplicação, divisão, modulo.
+ -	Adição, subtração.
<< >>	Lógica binária ¹ deslocamento ² esquerdo/direito
< <= > >=	Menor, menor ou igual, maior, maior ou igual.
== !=	Igual, não igual.
&	Lógica binária de conjunção (AND).
^	Lógica binária exclusiva (OR).
	Lógica binária de união (OR).
&&	Conjunção lógica (AND).
	Disjunção lógica (OR).
= *= /= %= += -= <<= >>= &= = ^=	Atribuição e alteração da atribuição.
,	Separa os termos e os parâmetros das funções.

¹ Bitwise no Inglês.

² Shift no Inglês.

As principais diferenças em relação à expressão semântica com C:

- Todos os números são valores não assinados com **64-bits**. Isso significa que números negativos não são permitidos.
- A conjunção lógica e os operadores de disjunção `&&` e `||` não apresentam propriedades de curto-circuito - ambos os lados da expressão são sempre avaliados.

8.4.1 Números

Os números literais são prefixados de acordo com suas respectivas bases:

- Hexadecimal (**base-16**) com `$` or `0x`.
- Decimal (**base-10**) com `#`.
- Octal (**base-8**) com `0o`.
- Binário (**base-2**) com `0b`.
- Números não prefixados são hexadecimais (**base-16**).

Exemplos:

- 123 é **123** em hexadecimal (**291** decimal).
- `$123` é **123** em hexadecimal (**291** decimal).
- `0x123` é **123** em hexadecimal (**291** decimal).
- `#123` é **123** em decimal.
- `0o123` é **123** em octal (**83** decimal).
- `0b1001` é **1001** binário (**9** decimal).
- `0b123` é **inválido**.

8.4.2 Valores booleanos

Qualquer expressão que avalia até um número pode ser utilizado onde um valor booleano seja necessário. Zero é tratado como falso, todos os valores *não zero* são tratados como verdadeiros. Além disso, uma *string*³ `true` é tratada como verdadeira e uma *string* `false` é tratada como falsa.

Uma expressão vazia pode ser apresentada como um argumento para que os parâmetros booleanos sejam usados como comandos padrão que possam ser usados para depuração, mesmo quando os parâmetros subsequentes sejam definidos.

³ Pode ser traduzido como cadeia de caracteres, cadência de caracteres, sequência de caracteres, caracteres, texto, linha, dentre outras variantes dependendo do contexto, [mais informações](#).

8.4.3 Acesso à memória

Os operadores do prefixo de acesso à memória permitem a leitura e a escrita nas regiões dos endereços emulados. Os operadores do prefixo da memória determinam o tamanho do acesso e se os efeitos colaterais estão desativados, opcionalmente, podem ser precedidos por uma determinação na região do endereçamento. Os tamanhos do acesso suportado e os modos dos efeitos colaterais são:

- **b** determina um acesso **8-bit** (byte).
- **w** determina um acesso **16-bit** (word).
- **d** determina um acesso **32-bit** (double word ou dword).
- **q** determina um acesso **64-bit** (quadruple word ou qword).
- **@** suprime os efeitos colaterais.
- **!** não suprime os efeitos colaterais.

Ao suprimir os efeitos colaterais de um acesso de leitura, a leitura do valor de um endereço seria obtida sem mais efeitos. Por exemplo, a leitura de uma "caixa postal" (*mailbox*) com efeitos colaterais desativada não limpará a sinalização que estiver pendente e a leitura de uma *FIFO* com os efeitos colaterais desativado, não fará a remoção de um item.

Nos acessos de escrita, ao suprimir os efeitos colaterais, o comportamento na maioria dos casos não é alterado, é preciso ver os efeitos da escrita no local. Entretanto, há algumas exceções onde é útil separar os diversos efeitos de um acesso de escrita, por exemplo:

- Alguns registros precisam ser escritos em sequência para evitar as condições de corrida. O depurador pode emitir várias escritas no mesmo ponto no tempo da emulação, de modo que estas condições de corrida não possam ser evitadas de forma trivial. Por exemplo, a escrita para a saída do MC68HC05 compara o registro com byte alto (*OCRH*) e impede a comparação até que a saída compare o registro com byte baixo (*OCRL*) seja escrita para evitar as condições de corrida. Como o depurador pode escrever em ambos os locais ao mesmo tempo do ponto de vista do sistema que está sendo emulado, assim no geral, a condição de corrida não é relevante. Ela é mais propensa a erros quando se pode acidentalmente definir seu estado oculto quando tudo o que se realmente deseja fazer é alterar o valor; assim ao escrever no *OCRH* com efeitos colaterais suprimidos, ele não inibe a comparação, apenas altera o valor no registro do comparador da saída.
- Ao escrever em alguns registros, os efeitos são diversos, o que pode ser útil para fins de depuração. Usando novamente o MC68HC05 como exemplo, ao escrever no *OCRL* o valor no registro de comparação da saída se altera e também limpa a sinalização de comparação da saída (*OCF*) permitindo comparar se ela foi anteriormente inibida durante a escrita no *OCRH*. A escrita no *OCRL* com efeitos colaterais desativado altera apenas o valor no registro sem apagar o *OCF* ou permite a comparação uma vez que é útil para a depuração. A escrita no *OCRL* com efeitos colaterais ativados tem efeitos adicionais.

Opcionalmente, o tamanho pode ser precedido por uma indicação do tipo de acesso:

- **p** ou **lp** define um endereço lógico predefinido para a região 0 (programa).
- **d** ou **ld** define um endereço lógico predefinido para a região 1 (dados).
- **i** ou **li** define um endereço lógico predefinido para a região 2 (E/S).
- **3** ou **l3** define um endereço lógico predefinido para a região 3 (*opcodes*).
- **pp** define um endereço físico predefinido para a região 0 (programa).
- **pd** define um endereço físico predefinido para a região 1 (dados).
- **pi** define um endereço físico predefinido para a região 2 (E/S).

- p3 define um endereço físico predefinido para a região 3 (opcodes).
- r define um ponteiro de acesso predefinido de leitura/escrita para a região 0 (programa).
- o define um ponteiro de acesso predefinido de leitura/escrita para a região 3 (opcodes).
- m define uma região da memória.

Finalmente, isso pode ser precedido por uma etiqueta e/ou um nome na região do endereçamento seguido por um ponto (.).

Isso pode parecer muita coisa para ser assimilado, então vamos olhar nos exemplos mais simples:

b@<addr>

Com referência ao byte em <addr> na região do programa da CPU atual enquanto estiver suprimindo os efeitos colaterais.

b!<addr>

Com referência ao byte em <addr> na região do programa da CPU atual enquanto **não** estiver suprimindo os efeitos colaterais como ao ler a "caixa de mensagens" (*mailbox*) limpando a sinalização pendente ou ao ler o *FIFO* removendo um item.

w@<addr> e w!<addr>

Com referência ao byte em <addr> na região do programa da CPU atual suprimindo ou não respectivamente os efeitos colaterais.

d@<addr> e d!<addr>

Consulte a palavra dupla (*double word*) no <addr> na região do programa da CPU atual, respectivamente suprimindo ou não os efeitos colaterais.

q@<addr> e q!<addr>

Consulte a palavra quadrupla (*quadruple word*) no <addr> na região do programa da CPU atual, respectivamente suprimindo ou não os efeitos colaterais.

A adição dos tipos de acesso oferece possibilidades adicionais:

dw@300

Com referência a palavra em 300 na região de dados da CPU atual enquanto estiver suprimindo os efeitos colaterais.

id@400

Com referência a palavra dupla em 400 na região de E/S da CPU atual enquanto estiver suprimindo os efeitos colaterais.

ppd!<addr>

Com referência a palavra dupla no endereço físico <addr> na região do programa da CPU atual enquanto **não** estiver suprimindo os efeitos colaterais.

rw@<addr>

Com referência a palavra no endereço <addr> na região do programa da CPU atual usando um ponteiro de acesso direto para leitura/escrita.

Se quisermos acessar uma faixa de endereços de um dispositivo que não seja a CPU atual, uma faixa de endereços além dos quatro primeiros índices ou uma região da memória, precisamos incluir uma etiqueta ou um nome:

`ramport.b@<addr>`

Com referência ao byte no endereço `<addr>` na região `ramport` na região do programa da *CPU* atual.

`audiocpu.dw@<addr>`

Com referência ao byte no endereço `<addr>` na região de dados da *CPU* com a etiqueta absoluta `:audiocpu`.

`maincpu.status.b@<addr>`

Com referência ao byte no endereço `<addr>` na região `status` da *CPU* com a etiqueta absoluta `:maincpu`.

`monitor.mb@78`

Com referência ao byte na região 78 da memória com a etiqueta absoluta `:monitor`.

`..md@202`

Com referência a palavra dupla no endereço 22 da região da memória com o mesmo caminho da etiqueta como a da *CPU*.

Algumas combinações não são úteis. Os endereços físicos e lógicos por exemplo, eles são equivalentes em algumas *CPUs* e o ponteiro de acesso direto para a leitura/escrita nunca tem um efeito colateral. Ao acessar a região da memória (acesso do tipo `m`) é preciso que uma etiqueta seja definida.

O acesso à memória pode ser usado com ambos os `lvalues` e `rvalues`, assim é possível escrever `b@100 = ff` para armazenar um byte na memória.

8.4.4 Funções

O depurador suporta uma quantidade de funções úteis nas expressões.

`min(<a>,)`

Retorna o menor valor de dois argumentos.

`max(<a>,)`

Retorna o maior valor de dois argumentos.

`if(<condição>, <valor_verdadeiro>, <valor_falso>)`

Retorna o `<valor_verdadeiro>` caso a `<condição>` seja verdadeira (não zero), ou caso contrário, `<valor_falso>`. Observe que ambas as expressões para `<valor_verdadeiro>` e `<valor_falso>` são ambos avaliados independentemente ainda que a `<condição>` seja verdadeira ou falsa.

`abs(<x>)`

Reinterpreta o argumento como um inteiro assinado com **64-bit** e retorna um valor absoluto.

`bit(<x>, <n>[, <w>])`

Extraí e alinha à direita, um campo do bit `<w>` a partir dos bits com largura `<x>` na posição do bit de menor importância `<n>`, contando a partir do bit menos importante. Caso `<w>` seja omitido, um único bit é extraído.

`s8(<x>)`

Prolonga a assinatura do argumento de **8** para **64-bits** (substitui os bits **8** até **63**, inclusive, com o valor do bit **7** contando a partir do bit com menor importância).

`s16(<x>)`

Prolonga a assinatura do argumento de **16** para **64-bits** (substitui os bits **16** até **63**, inclusive, com o valor do bit **15** contando a partir do bit com menor importância).

`s32(<x>)`

Prolonga a assinatura do argumento de **32** para **64-bits** (substitui os bits **32** até **63**, inclusive, com o valor do bit **31** contando a partir do bit com menor importância).

INTERFACE PARA SCRIPTS LUA

- *Introdução*
- *Características*
- *Referência da API*
- *Tutorial do console interativo Lua*

9.1 Introdução

O MAME oferece um conjunto útil de funcionalidades básicas para scripts Lua¹. Estas funcionalidades apareceram inicialmente na versão **0.148** quando uma reduzida interface Lua foi implementada. Hoje em dia, a interface Lua é rica o suficiente para manipular os estados dos dispositivos, acesso aos registros do CPU, ler e escrever na memória, desenhar um painel customizado na tela, etc.

Há três maneiras de utilizar os recursos de script Lua do MAME:

- Usando o *console interativo Lua*, ativado através da opção *-console*.
- Ao rodar um script via arquivo usando a opção *-autoboot_script*. A opção *-autoboot_delay* controla quanto tempo o MAME deve aguardar para roda o script depois que a emulação for iniciada.
- Ao escrever *plug-ins Lua*. Alguns já estão inclusos com o MAME.

Internamente, o MAME faz o uso intensivo do *Sol3* para implementar as funcionalidades Lua.

A API Lua ainda não é considerado estável, havendo a possibilidade de ser alterada repentinamente sem nenhum aviso prévio. No entanto, podemos demonstrar metodologias onde é possível saber qual a versão do API está sendo rodada, a maioria dos objetos suporta algum nível de introspeção durante o tempo de execução.

¹ Acesse o [site do projeto Lua](#) para obter mais informações.

9.2 Características

Pelo fato da API estar incompleta, abaixo uma lista parcial de recursos disponíveis atualmente com os scripts Lua:

- Informações da sessão (versão do aplicativo, sistema atualmente emulado).
- Controle da sessão (iniciar, pausar, redefinir, parar).
- Ganchos de eventos (na pintura dos quadros e nos eventos do usuário).
- Introspecção dos dispositivos (listagem da árvore de dispositivos, enumeração da memória e registros).
- Introspecção da tela (listagem das telas, detalhes da tela, contagem de quadros).
- Desenho da sobreposição de tela (texto, linhas, caixas em várias telas).
- Leitura/escrita da memória (8, 16, 32 e 64 bits, signed e unsigned).
- Registro e controle de estado (enumeração de estado, get e set).

9.3 Referência da API

9.3.1 Tipos Lua comuns e globais

- *Contêineres*
- *Interface do emulador*

Contêineres

Várias propriedades rendem envoltórios dos recipientes. Os envoltórios dos contêineres são fáceis de serem criados e fornecem uma interface semelhante a uma tabela só de leitura. A complexidade das operações podem variar. Os envoltórios dos contêineres geralmente disponibilizam a maioria destas operações:

#c

Obtém a quantidade dos itens dentro do contêiner.

c[k]

Retorna o item que corresponda a tecla k ou nil caso a chave não esteja presente.

pairs(c)

Repete o contêiner por chave e por valor. A chave é o que você passaria para operador do índice ou o método get para obter o valor.

ipairs(c)

Repete o contêiner através de um índice e de um valor. O índice é o que você passaria para ao método at para obter o valor (pode ser o mesmo como a chave para alguns contêineres).

c:empty()

Retorna um booleano indicando se não há itens no contêiner.

c:get(k)

Retorna o item que corresponda a tecla k ou nil caso a tecla não esteja presente. Normalmente é o equivalente ao operador do índice.

c:at(i)

Retorna o valor no índice com base 1 (1-based) i ou nil caso não esteja fora do alcance.

c:find(v)

Retorna a chave para o item v ou nil caso não esteja no contêiner. A chave é o que você passaria ao índice do operador para obter o valor.

c:index_of(v)

Retorna o índice com base 1 (1-based) para o item v ou nil caso não esteja no contêiner. O índice é o que você passaria ao método at para obter o valor.

Interface do emulador

A interface `emu` fornece o acesso à principal funcionalidade do emulador. Diversas classes também estão disponíveis como propriedades na interface do emulador.

Métodos

`emu.wait(duração, ...)`

Aguarda a duração determinada pelo tempo da emulação. A duração pode ser definida como *attotime* ou um valor numérico em segundos. Quaisquer argumentos adicionais será retornado a quem os invocou. Retorna um booleano indicando se a duração expirou normalmente.

Todas as invocações pendentes para `emu.wait`, imediatamente retornarão `false` caso um estado salvo seja carregado ou se a sessão da emulação for encerrada. Retornará um erro ao invocar esta função através dos retornos de chamada (*callbacks*) que não forem executados como rotinas conjuntas.

`emu.wait_next_update(...)`

Aguarda até a próxima atualização de vídeo/UI. Quaisquer argumentos adicionais será retornado a quem os invocou. Retornará um erro ao invocar esta função através dos retornos de chamada (*callbacks*) que não forem executados como rotinas conjuntas.

`emu.wait_next_frame(...)`

Aguarda até que o próximo quadro da emulação seja concluído. Quaisquer argumentos serão retornados a quem os invocou. Retornará um erro ao invocar esta função através dos retornos de chamada (*callbacks*) que não forem executados como rotinas conjuntas.

`emu.add_machine_reset_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações quando o sistema emulado for reinicializado. Retorna um *notificador da assinatura*.

`emu.add_machine_stop_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações quando o sistema emulado for parado. Retorna um *notificador da assinatura*.

`emu.add_machine_pause_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações quando o sistema emulado for pausado. Retorna um *notificador da assinatura*.

`emu.add_machine_resume_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações quando o sistema emulado resumir as operações. Retorna um *notificador da assinatura*.

`emu.add_machine_frame_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações quando o sistema emulado concluir um quadro. Retorna um *notificador da assinatura*.

`emu.add_machine_pre_save_notifier(callback)`

Adiciona um retorno de chamada para receber as notificações antes que o estado da emulação seja salvo. Retorna um *notificador da assinatura*.

emu.add_machine_post_load_notifier(callback)

Adiciona um retorno de chamada para receber as notificações depois que o estado da emulação seja salvo. Retorna um *notificador da assinatura*.

emu.print_error(mensagem)

Exibe uma mensagem de erro.

emu.print_warning(mensagem)

Exibe uma mensagem de alerta.

emu.print_info(mensagem)

Exibe uma mensagem informacional.

emu.print_verbose(mensagem)

Exibe uma mensagem loquaz de diagnóstico (desativado por padrão).

emu.print_debug(mensagem)

Exibe uma mensagem loquaz de depuração (ativada por padrão apenas em versões de depuração).

emu.lang_translate([contexto], mensagem)

Procure uma mensagem com contexto opcional no catálogo atual das mensagens traduzidas. Retorna a mensagem original caso nenhuma mensagem traduzida correspondente seja encontrada.

emu.subst_env(string)

Variáveis de ambiente substituíveis em texto (*string*). A sintaxe depende do sistema operacional do host.

9.3.2 As principais classes Lua

Muitas das principais classes do MAME usadas para implementar a emulação de uma sessão, estão disponíveis para os *scripts* Lua.

- *Notificador da assinatura*
- *Attotime*
- *O gerenciador de sistema do MAME*
- *O sistema em execução*
- *O gerenciador de vídeo*
- *O gerenciador de áudio*
- *O gerenciador da saída*
- *O gerenciador de parâmetros*
- *O gerenciador da IU*
- *Os metadados do driver do sistema*
- *Plug-in Lua*

Notificador da assinatura

Envelopa a classe `util::notifier_subscription` do MAME, faz o gerenciamento de uma assinatura numa notificação de difusão.

Métodos

subscription:unsubscribe()

Faz a remoção da notificação das assinaturas . A assinatura se torna inativa e nenhuma outra notificação será recebida.

Propriedades

subscription.is_active (somente leitura)

Um booleano que indica se a assinatura está ativa. A assinatura se torna inativa depois que a assinatura for explicitamente cancelada ou caso o notificador subjacente seja destruído.

Attotime

Encapsula a classe *attotime* do MAME, o que representa um intervalo de tempo de alta precisão. Os valores *attotime* suportam a adição e a subtração com outros valores *attotime*, assim como a multiplicação e a divisão por números inteiros.

Instanciação

emu.attotime()

Cria um valor *attotime* representando zero (ou seja, sem tempo decorrido).

emu.attotime(segundos, attosegundos)

Cria um *attotime* com as partes inteiras e fracionárias específicas.

emu.attotime(attotime)

Cria uma cópia de um valor *attotime* existente no momento.

emu.attotime.from_double(segundos)

Cria um valor *attotime* representando um número específico em segundos.

emu.attotime.from_ticks(períodos, frequência)

Cria um *attotime* representando um número específico dos períodos da frequência informada em Hertz.

emu.attotime.from_seconds(segundos)

Cria um *attotime* representando um número inteiro específico em segundos.

emu.attotime.from_msec(milissegundos)

Cria um *attotime* representando um número inteiro específico em milissegundos.

emu.attotime.from_usec(microssegundos)

Cria um *attotime* representando um número inteiro específico em microssegundos.

emu.attotime.from_nsec(nanossegundos)

Cria um *attotime* representando um número inteiro específico em nanossegundos.

Métodos

t:as_double()

Retorna o intervalo de tempo em segundos como um valor de ponto flutuante.

t:as_hz()

Interpreta o intervalo como um período e retorna o valor correspondente da frequência em Hertz como um ponto flutuante. Retorna zero caso *t.is_never* seja verdadeiro. O intervalo não deve ser zero.

t:as_khz()

Interpreta o intervalo como um período e retorna o valor correspondente da frequência em quilo hertz como um ponto flutuante. Retorna zero caso *t.is_never* seja verdadeiro. O intervalo não deve ser zero.

t:as_mhz()

Interpreta o intervalo como um período e retorna o valor correspondente da frequência em mega hertz como um ponto flutuante. Retorna zero caso `t.is_never` seja verdadeiro. O intervalo não deve ser zero.

t:as_ticks(frequência)

Retorna o intervalo como um número inteiro em períodos da frequência definida. O valor da frequência é definida em Hertz.

Propriedades

t.is_zero (somente leitura)

Um booleano que indica se o valor não representa um tempo transcorrido.

t.is_never (somente leitura)

Um booleano que indica se o valor for maior que a quantidade máxima de segundos inteiros que possam ser representados (tratados como um tempo inalcançável no futuro ou num estouro).

t.attoseconds (somente leitura)

A fração do intervalo dos segundos em atossegundos.

t.seconds (somente leitura)

A quantidade de segundos inteiros no intervalo.

t.msec (somente leitura)

A quantidade de milissegundos inteiros na porção de segundos fracionários do intervalo.

t.usec (somente leitura)

A quantidade de microssegundos inteiros na porção de segundos fracionários do intervalo.

t.nsec (somente leitura)

A quantidade de nanossegundos inteiros na porção de segundos fracionários do intervalo.

O gerenciador de sistema do MAME

Encapsula a classe `mame_machine_manager` do MAME que contém o sistema em execução, o gerenciador da IU e os outros componentes globais.

Instanciação

manager

O gerenciador do sistema do MAME está disponível como uma variável global no ambiente Lua.

Propriedades

manager.machine (somente leitura)

O sistema em execução para a sessão da emulação atual.

manager.ui (somente leitura)

O gerenciador da IU para a sessão da emulação atual.

manager.options (somente leitura)

As `luascript-ref-emuopts` para a sessão da emulação atual.

manager.plugins[] (somente leitura)

Obtém informações sobre o *plug-in Lua* que estão presentes, indexados por nome. Os métodos do índice obtém `at` e `index_of` com complexidade $O(n)$.

O sistema em execução

Encapsula a classe `running_machine` do MAME que representa uma sessão da emulação. Ele fornece acesso aos outros principais objetos que implementam uma sessão da emulação, bem como a árvore dos dispositivos emulados.

Instanciação

manager.machine

Obtém a instância do sistema em execução para a sessão de emulação atual.

Métodos

machine:exit()

Agenda o encerramento da sessão da emulação atual. Isso irá retornar ao menu da seleção do sistema ou encerrar o aplicativo, dependendo de como ele foi iniciado. Este método retorna imediatamente antes que o encerramento do programada ocorra.

machine:hard_reset()

Agenda uma reinicialização a frio. Isso é implementado destruindo a sessão da emulação e iniciando outra sessão para o mesmo sistema. Este método retorna imediatamente antes que a reinicialização programada aconteça.

machine:soft_reset()

Agenda uma reinicialização suave. Isso é implementado chamando o método da redefinição do dispositivo principal, que é propagado pela árvore dos dispositivos. Este método retorna imediatamente antes que a reinicialização programada aconteça.

machine:save(nome_do_arquivo)

Agenda o salvamento do estado do sistema no arquivo informado. Caso o nome do arquivo seja um caminho relativo, ele será considerado relativo ao primeiro diretório do estado de salvamento configurado. Este método retorna imediatamente antes que o estado do sistema seja salvo. Caso este método seja chamado quando uma operação de salvar ou de carregar já esteja pendente, a operação pendente anterior será cancelada.

machine:load(nome_do_arquivo)

Agenda o carregamento do estado do sistema a partir do arquivo informado. Caso o nome do arquivo seja um caminho relativo, os diretórios configurados para o estado de salvamento serão pesquisados. Este método retorna imediatamente antes que o estado do sistema seja salvo. Caso este método seja chamado quando uma operação de salvar ou de carregar já esteja pendente, a operação pendente anterior será cancelada.

machine:popmessage([msg])

Exibe uma mensagem pop-up para o usuário. Caso a mensagem não seja informada, a mensagem de pop-up exibida no momento (caso haja) ficará oculta.

machine:logerror(msg)

Grava a mensagem no log de erros do sistema. Isso pode ser exibido numa janela do depurador, gravado num arquivo ou gravado na saída de erro predefinida.

Propriedades**machine.time** (somente leitura)

O tempo decorrida da emulação para a sessão atual assim como em *attotime*.

machine.system (somente leitura)

os metadados do driver do sistema para o sistema atual.

machine.parameters (somente leitura)

O *gerenciador dos parâmetros* para a sessão da emulação atual.

machine.video (somente leitura)

O *gerenciador de vídeo* para a sessão da emulação atual.

machine.sound (somente leitura)

O *gerenciador do áudio* para a sessão da emulação atual.

machine.output (somente leitura)

O *gerenciador da saída* para a sessão da emulação atual.

machine.memory (somente leitura)

O *gerenciador da memória* para a sessão da emulação atual.

machine.ioport (somente leitura)

O *gerenciador da porta de E/S* para a sessão da emulação atual.

machine.input (somente leitura)

O *gerenciador da entrada* para a sessão da emulação atual.

machine.natkeyboard (somente leitura)

Obtém o *gerenciador do teclado natural*, usado para controlar a entrada do teclado e do teclado numérico no sistema emulado.

machine.uiinput (somente leitura)

O *gerenciador da entrada da IU* para a sessão da emulação atual.

machine.render (somente leitura)

O *gerenciador do renderizador* para a sessão da emulação atual.

machine.debugger (somente leitura)

O *gerenciador do depurador* para a sessão da emulação atual ou nil se o depurador não estiver ativado.

machine.options (somente leitura)

As opções definidas pelo usuário para a sessão da emulação atual.

machine.samplerate (somente leitura)

A taxa de amostragem da saída do áudio em Hertz.

machine.paused (somente leitura)

Um booleano que indica se a emulação não está em execução no momento, geralmente porque a sessão foi pausada ou o sistema emulado não concluiu a inicialização.

machine.exit_pending (somente leitura)

Um booleano que indica se a sessão da emulação está programada para encerrar.

machine.hard_reset_pending (somente leitura)

Um booleano que indica se uma reinicialização forçada do sistema emulado está pendente.

machine.devices (somente leitura)

Um *dispositivo enumerador* que produz todos os *dispositivos* no sistema que está sendo emulado.

machine.palettes (somente leitura)

Um *dispositivo enumerador* que produz todos os *dispositivos paleta* no sistema que está sendo emulado.

machine.screens (somente leitura)

Um *dispositivo enumerador* que produz todos os *dispositivos tela* no sistema que está sendo emulado.

machine.cassettes (somente leitura)

Um *dispositivo enumerador* que produz todos os *dispositivos da imagem em fita cassete* no sistema que está sendo emulado.

machine.images (somente leitura)

dispositivo enumerador que produz toda a *interface para os dispositivos de imagem* no sistema que está sendo emulado.

machine.slots (somente leitura)

dispositivo enumerador que produz toda a *dispositivos slot* no sistema que está sendo emulado.

O gerenciador de vídeo

Encapsula a classe `video_manager` do MAME que é responsável por coordenar a exibição do vídeo que está sendo emulado, a aceleração da velocidade e da leitura das entradas do host.

Instanciação

manager.machine.video

Obtém o gerenciador do vídeo para a sessão da emulação atual.

Métodos

video:frame_update()

Atualiza as telas emuladas, lê as entradas do host e atualiza a saída de vídeo.

video:snapshot()

Salva os arquivos da captura da tela de acordo com a configuração atual. Caso o MAME esteja configurado para obter as capturas da tela emulada de forma nativa, a captura da tela que será salvo será de todas as telas que estiverem visíveis numa janela ou da tela do host com a configuração da exibição atual. Caso o MAME esteja configurado para obter as capturas da tela emulada de forma nativa, ou seja, o sistema não tiver uma tela emulada, uma captura da tela será salva usando a visualização selecionada no momento.

video:begin_recording([nome_do_arquivo], [formato])

Interrompe todas as gravações de vídeo em andamento e começa a gravar as telas emuladas que estão visíveis ou a exibição do captura da tela atual, dependendo se o MAME está configurado para obter as capturas nativas da tela emulada. Caso o nome do arquivo não seja informado, a configuração do nome do arquivo da captura da tela será usada. Caso o nome do arquivo seja um caminho relativo, ele será interpretado em relação ao primeiro diretório da configuração da captura da tela. Caso o formato seja informado ele deve ser `avi` ou `mng`. Se não for informado, a predefinição é `AVI`.

video:end_recording()

Interrompe qualquer gravação de vídeo em andamento.

video:snapshot_size()

Retorna a largura e a altura em pixels das capturas da tela criados com a configuração atual do destino e o estado da tela emulada. Isso pode ser configurado de forma explícita pelo usuário, calculado com base na visualização da captura selecionada e na resolução de quaisquer telas visíveis e que estejam sendo emuladas.

video:snapshot_pixels()

Retorna os pixels de uma captura criado usando a configuração do destino da captura atual em inteiros com 32 bits e compactados numa *string* binária com ordem Endian do host. Os pixels são organizados em ordem maior da linha, da esquerda para a direita e de cima para baixo. Os valores do pixel são cores no formato RGB compactadas em inteiros com 32 bits.

Propriedades**video.speed_factor** (somente leitura)

Ajuste de velocidade da emulação configurada em escala de mil (ou seja, a proporção para a velocidade normal multiplicada por 1.000).

video.throttled (leitura e escrita)

Um booleano que indica se o MAME deve esperar antes das atualizações do vídeo para evitar a execução mais rápida do que a velocidade desejada.

video.throttle_rate (leitura e escrita)

A velocidade de emulação desejada como uma proporção da velocidade total ajustada através do fator de velocidade (ou seja, 1 é a velocidade normal ajustada pelo fator de velocidade, números maiores são mais rápidos e números menores são mais lentos).

video.frameskip (leitura e escrita)

A quantidade dos quadros emulados do vídeo para serem ignorados a cada doze ou -1 para ajustar automaticamente a quantidade de quadros para ignorar visando para manter a velocidade da emulação desejada.

video.speed_percent (somente leitura)

A velocidade emulada atualmente em porcentagem da velocidade total ajustada pelo fator da velocidade.

video.effective_frameskip (somente leitura)

A quantidade dos doze quadros emulados que são ignorados.

video.skip_this_frame (somente leitura)

Um booleano que indica se o gerenciador do vídeo vai ignorar as telas emuladas para o quadro atual.

video.snap_native (somente leitura)

Um booleano que indica se o gerenciador do vídeo fará capturas nativa da tela emulada. Além da definição da configuração relevante, o sistema emulado deve ter pelo menos uma tela que esteja sendo emulada.

video.is_recording (somente leitura)

Um booleano que indica se alguma gravação de vídeo está em andamento.

video.snapshot_target (somente leitura)

Um *alvo do renderizador* usado para produzir as capturas da tela e para as gravações de vídeo.

O gerenciador de áudio

Encapsula a classe `sound_manager` do MAME que gerencia o gráfico do fluxo do áudio emulado e coordena a sua saída.

Instanciação

manager.machine.sound

Obtém o gerenciador do áudio para a sessão da emulação atual.

Métodos

sound:start_recording([nome_do_arquivo])

Inicia a gravação num arquivo WAV. Não tem efeito se estiver gravando. Caso o nome do arquivo não seja informado usa o nome do arquivo WAV configurado (da linha de comando ou do arquivo INI) ou não tem efeito se nenhum nome do arquivo WAV estiver configurado. Retorna `true` se a gravação foi iniciada ou `false` se a gravação já estiver em andamento, a abertura do arquivo gerado falhou ou nenhum nome para o arquivo foi informado ou foi configurado.

sound:stop_recording()

Interrompe a gravação e fecha o arquivo se estiver um arquivo WAV estiver sendo gravado.

sound:get_samples()

Retorna o conteúdo atual do buffer da amostra gerada como uma *string* binária. As amostras são inteiros com 16 bits na ordem dos bytes do host. As amostras dos canais estéreo esquerdo e direito são intercaladas.

Propriedades

sound.muted (somente leitura)

Um booleano que indica se a saída do áudio está silenciada por algum motivo.

sound.ui_mute (leitura e escrita)

Um booleano que indica se a saída do áudio está silenciada a pedido do usuário.

sound.debugger_mute (leitura e escrita)

Um booleano que indica se a saída do áudio está silenciada a pedido do depurador.

sound.system_mute (leitura e escrita)

Um booleano que indica se a saída do áudio foi silenciada a pedido do sistema que está sendo emulado.

sound.attenuation (leitura e escrita)

A atenuação do volume da saída em decibéis. Geralmente deve ser um número inteiro negativo ou zero.

sound.recording (somente leitura)

Um booleano que indica se a saída do áudio está sendo gravada num arquivo WAV.

O gerenciador da saída

Encapsula a classe `output_manager` do MAME que fornece acesso às saídas do sistema que podem ser usadas para arte interativa ou consumidas por programas externos.

Instanciação**manager.machine.output**

Obtém o gerenciador da saída para a sessão da emulação atual.

Métodos**output:set_value(nome, valor)**

Define o valor de saída informada. O valor deve ser um número inteiro. A saída será criada caso ainda não exista.

output:set_indexed_value(prefixo, índice, valor)

Acrescenta o índice (formatado como um inteiro decimal) ao prefixo e define o valor da saída correspondente. O valor deve ser um número inteiro. A saída será criada caso ainda não exista.

output:get_value(nome)

Retorna o valor da saída informada ou zero caso não exista.

output:get_indexed_value(prefixo, índice)

Anexa o índice (formatado como um inteiro decimal) ao prefixo e retorna o valor da saída correspondente ou zero caso não exista.

output:name_to_id(nome)

Obtém o ID com número inteiro exclusivo por sessão para a saída informada ou zero caso não exista.

output:id_to_name(id)

Obtém o nome da saída com o ID exclusivo por sessão informada ou `nil` caso não exista. Este método tem complexidade $O(n)$, portanto, evite chamá-lo quando o desempenho for importante.

O gerenciador de parâmetros

Wraps MAME's `parameters_manager` class, which provides a simple key-value store for metadata from system ROM definitions.

Instanciação

Encapsula a classe `parameters_manager` do MAME que fornece um armazenamento simples do valor da chave para os metadados das definições da ROM do sistema.

Métodos

`parameters:lookup(tag)`

Obtém o valor do parâmetro informado caso esteja definido ou uma *string* vazia se não estiver.

`parameters:add(tag, valor)`

Define o parâmetro informado caso não esteja. Não tem efeito se o parâmetro informado já estiver definido.

O gerenciador da IU

Encapsula a classe `mame_ui_manager` do MAME que lida com menus e as outras funcionalidades da interface do usuário.

Instanciação

`manager.ui`

Obtém o gerenciador da IU para a sessão atual.

Métodos

`ui:get_char_width(ch)`

Obtém a largura de um caractere Unicode como uma proporção da largura do contêiner da IU na fonte atualmente utilizada na altura configurada da linha da IU.

`ui:get_string_width(str)`

Obtém a largura de uma *string* como uma proporção da largura do contêiner da IU na fonte atualmente utilizada na altura configurada da linha da IU.

`ui:set_aggressive_input_focus(ativa)`

Em algumas plataformas isso controla se o MAME deve aceitar o foco da entrada em mais situações do que quando as suas janelas têm o foco da IU.

`ui:get_general_input_setting(type, [jogador])`

Obtém uma descrição da *sequência da entrada* configurada para o tipo da entrada indicada e o jogador adequado para usar nos prompts. O tipo da entrada é um valor enumerado. O número do jogador é um índice com base no número zero. Caso o número do jogador não seja informado, é assumido o valor zero.

Propriedades

ui.options (somente leitura)

As opções da interface para a sessão atual.

ui.line_height (somente leitura)

A altura configurada da linha de texto da interface como uma proporção da altura do contêiner da interface.

ui.menu_active (somente leitura)

Um booleano que indica se um elemento da interface interativa está atualmente ativa. Os exemplos incluem os menus e os controles deslizantes.

ui.ui_active (leitura e escrita)

Um booleano que indica se as entradas de controle da IU estão ativadas.

ui.single_step (leitura e escrita)

Um booleano que controla se o sistema emulado deve ser pausado automaticamente quando o próximo quadro for desenhado. Esta propriedade é redefinida automaticamente quando acontecer a pausa automática.

ui.show_fps (leitura e escrita)

Um booleano que controla se a velocidade atual da emulação e as configurações do salto de quadro devem ser exibidas.

ui.show_profiler (leitura e escrita)

Um booleano que controla se as estatísticas da criação do perfil devem ser exibidas.

Os metadados do driver do sistema

Fornece alguns metadados para um sistema que estiver sendo emulado.

Instanciação

emu.driver_find(nome)

Obtém os metadados do driver informado para o sistema com o nome abreviado ou nil caso o sistema não exista.

manager.machine.system

Obtém os metadados do driver para o sistema atual.

Propriedades

driver.name (somente leitura)

O nome abreviado do sistema, conforme usado na linha de comando, nos arquivos de configuração e ao pesquisar os recursos.

driver.description (somente leitura)

O nome completo da exibição do sistema.

driver.year (somente leitura)

O ano do lançamento do sistema. Pode conter pontos de interrogação caso não seja totalmente conhecido.

driver.manufacturer (somente leitura)

O fabricante, o desenvolvedor ou o distribuidor do sistema.

driver.parent (somente leitura)

O nome abreviado do sistema principal para fins de organização ou "" se o sistema não venha de uma matriz.

driver.compatible_with (somente leitura)

O nome abreviado de um sistema onde este sistema seja compatível com o software ou nil caso o sistema não esteja listado como compatível com um outro sistema.

driver.source_file (somente leitura)

O arquivo de origem onde este driver do sistema estiver definido. O formato do caminho depende do conjunto das ferramentas onde o emulador foi compilado.

driver.rotation (somente leitura)

Uma *string* que indica a rotação aplicada a todas as telas no sistema depois que a orientação da tela informada na configuração do sistema seja aplicado. Será um dos "rot0", "rot90", "rot180" ou "rot270".

driver.not_working (somente leitura)

Um booleano que indica se o sistema foi marcado como não funcionando.

driver.supports_save (somente leitura)

Um booleano que indica se o sistema é compatível com salvamento de estado.

driver.no_cocktail (somente leitura)

Um booleano que indica se se não existe compatibilidade para a inversão da tela em modo coquetel.

driver.is_bios_root (somente leitura)

Um booleano que indica se se este sistema representa um sistema que roda programas a partir de uma mídia removível sem que a mídia esteja presente.

driver.requires_artwork (somente leitura)

Um booleano que indica se se o sistema requer a utilização de uma ilustração externa.

driver.unofficial (somente leitura)

Um booleano que indica se se esta é uma alteração oficial, porém uma alteração comum do usuário para o sistema.

driver.no_sound_hw (somente leitura)

Um booleano que indica se o sistema não possui nenhum hardware de saída de áudio.

driver.mechanical (somente leitura)

Um booleano que indica se o sistema depende de recursos mecânicos que não podem ser devidamente simulados.

driver.is_incomplete (somente leitura)

Um booleano que indica se o sistema é um protótipo com funcionalidades incompletas.

Plug-in Lua

Fornece uma descrição de um plug-in Lua que esteja disponível.

Instanciação

manager.plugins[nome]

Obtém a descrição do plug-in Lua com o nome informado ou nil caso o plug-in não esteja disponível.

Propriedades

plugin.name (somente leitura)

O nome abreviado do plug-in usado na configuração e durante o acesso.

plugin.description (somente leitura)

Exibe o nome do plug-in.

plugin.type (somente leitura)

O tipo do plug-in. Pode ser "plugin" para os plug-ins que podem ser carregados pelo usuário ou "library" para as bibliotecas que fornecem funcionalidades comum aos diferentes plug-ins.

plugin.directory (somente leitura)

O caminho para o diretório que contém os arquivos de plug-in.

plugin.start (somente leitura)

Um booleano que indica se o plug-in está ativado.

9.3.3 Classes dos dispositivos Lua

Diversas classes de dispositivos e classes combinadas de dispositivos são expostas ao Lua. Os dispositivos podem ser pesquisados através das tags ou podem ser enumerados.

- *Os enumeradores de dispositivos*
- *Dispositivo*
- *Dispositivo paleta*
- *Dispositivo tela*
- *Dispositivo de imagem em fita cassete*
- *Dispositivos de interface de imagem*
- *Dispositivo de interface slot*
- *Estado da entrada do dispositivo*
- *Formato da imagem da mídia*
- *Opções do slot*

Os enumeradores de dispositivos

Os enumeradores de dispositivos são contêineres especiais que permitem iterar (repetir) e pesquisar os dispositivos através de uma tag. Um enumerador pode ser criado para encontrar qualquer tipo de dispositivo, para encontrar dispositivos de um tipo em particular ou para encontrar dispositivos que implementem uma interface específica. Ao iterar os dispositivos utilizando `pairs` ou `ipairs`, tais dispositivos retornam primeiro através da árvore dos dispositivos em ordem de criação.

O índice faz com que o operador procure um dispositivo através da tag. Ele retorna `nil` caso nenhum dispositivo com a tag especificada seja encontrado ou se o dispositivo com tal tag não atenda aos requisitos do tipo ou da interface do enumerador dos dispositivos. A complexidade é $O(1)$ caso o resultado seja colocado em cache, porém, a busca de um dispositivo sem cache é custosa. O método `at` tem complexidade $O(n)$.

Caso crie um enumerador dos dispositivos com um ponto de partida diferente do dispositivo do sistema principal, a entrega de uma tag completa ou uma tag contendo as referências principais para o operador do índice pode fazer com que retorne um dispositivo que não seria descoberto pela iteração. Se você criar um enumerador dos dispositivos com uma extensão restrita, os dispositivos que não seriam encontrados por serem muito extensos dentro hierarquia ainda podem ser pesquisados através da tag.

A criação de um enumerador para os dispositivos com extensão restrita a zero pode ser usada para reduzir um dispositivo ou testar se um dispositivo consegue implementar uma determinada interface. Por exemplo, isto testará se um dispositivo consegue implementar a interface de imagem da mídia:

```
image_intf = emu.image_enumerator(device, 0):at(1)
if image_intf then
    print(string.format("Device %s mounts images", device.tag))
end
```


Instanciação

manager.machine.devices

Retorna um dispositivo enumerador que irá iterar sobre o *dispositivo* no sistema.

manager.machine.palettes

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos paleta* no sistema.

manager.machine.screens

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos da tela* no sistema.

manager.machine.cassettes

Retorna um dispositivo enumerador que irá iterar sobre o *dispositivo de imagem em fita cassete* no sistema.

manager.machine.images

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos de interface de imagem* no sistema.

manager.machine.slots

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos slot* no sistema.

emu.device_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre o *dispositivo* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído. Caso a profundidade seja informada, este deve ser um valor inteiro que irá definir a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

emu.palette_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos paleta* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído caso seja um dispositivo paleta. Caso a profundidade seja informada, este deve ser um valor inteiro que irá definir a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

emu.screen_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos tela* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído se for um dispositivo tela. Caso a profundidade seja informada este deve ser um valor inteiro que irá definir a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

emu.cassette_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre o *dispositivo de imagem em fita cassete* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído se for um dispositivo cassete. Caso a profundidade seja informada, este deve ser um valor inteiro que irá definir a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

emu.image_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos de imagem em mídia* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído caso seja uma mídia de um dispositivo de imagem. Caso a profundidade seja informada este deve ser um valor inteiro que definirá a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

emu.slot_enumerator(dispositivo, [profundidade])

Retorna um dispositivo enumerador que irá iterar sobre os *dispositivos slot* na sub-árvore começando num dispositivo específico. O dispositivo informado será incluído se for um dispositivo slot. Caso a profundidade seja informada, este deve ser um valor inteiro que definirá a quantidade máxima dos níveis que serão iterados abaixo do dispositivo informado (Por exemplo, 1 irá limitar a iteração do dispositivo e dos dispositivos relacionados).

Dispositivo

Encapsula a classe `device_t` do MAME que serve de base para todas as classes dos dispositivos.

Instanciação

manager.machine.devices[tag]

Obtém um dispositivo através de uma tag com relação ao dispositivo do sistema principal ou nil caso o dispositivo não exista.

manager.machine.devices[tag]:subdevice(tag)

Obtém um dispositivo através de uma tag com relação a outro dispositivo arbitrário ou nil caso o dispositivo não exista.

Métodos

device:subtag(tag)

Converte uma tag com relação ao dispositivo numa tag absoluta.

device:siblingtag(tag)

Converte uma tag com relação ao dispositivo principal do dispositivo numa tag absoluta.

device:memshare(tag)

Obtém um *compartilhamento da memória* através de uma tag com relação ao dispositivo ou nil caso o compartilhamento da memória não exista.

device:membank(tag)

Obtém um *banco da memória* através de uma tag com relação ao dispositivo ou nil caso o banco da memória não exista.

device:memregion(tag)

Obtém uma *região da memória* através de uma tag com relação ao dispositivo ou nil caso a região da memória não exista.

device:ioport(tag)

Obtém uma *porta de E/S* através da tag com relação ao dispositivo ou nil caso a porta de E/S não exista.

device:subdevice(tag)

Obtém um dispositivo através de uma tag com relação ao dispositivo.

device:siblingdevice(tag)

Obtém um dispositivo através de uma tag com relação ao dispositivo principal.

device:parameter(tag)

Obtém o valor do parâmetro através da tag relativa ao dispositivo ou uma *string* vazia caso não esteja definida.

Propriedades**device.tag** (somente leitura)

A tag absoluta do dispositivo em forma canônica.

device.basetag (somente leitura)

O último componente da tag do dispositivo (Por exemplo, quando a sua tag for relativa ao dispositivo principal) ou "root" para o dispositivo raiz do sistema.

device.name (somente leitura)

Exibe o nome completo para o tipo do dispositivo.

device.shortname (somente leitura)

O nome curto do tipo do dispositivo (usado, por exemplo, na linha de comando, ao procurar por recursos como ROMs ou a ilustração e em vários arquivos de dados).

device.owner (somente leitura)

A relação direta do dispositivo na árvore do dispositivo ou nil para o dispositivo raiz do dispositivo do sistema.

device.configured (somente leitura)

Um booleano que indica se o dispositivo concluiu a configuração.

device.started (somente leitura)

Um booleano que indica se o dispositivo concluiu a inicialização.

device.debug (somente leitura)

A *interface de depuração do dispositivo* para o dispositivo caso seja um dispositivo CPU ou nil caso não seja ou se o depurador não estiver ativado.

device.state[] (somente leitura)

O *estado das entradas* para os dispositivos que expõem a interface de registro do estado, indexadas por símbolos ou nil para outros dispositivos. O operador do índice e os métodos `index_of` têm complexidade $O(n)$; todas as outras operações compatíveis têm complexidade $O(1)$.

device.spaces[] (somente leitura)

A tabela dos *espaços de endereçamento da memória* do dispositivo, indexado por nome. Válido apenas para os dispositivos que implementam a interface da memória. Observe que os nomes são específicos para o tipo do dispositivo e não têm um significado especial.

Dispositivo paleta

Encapsula a classe `device_palette_interface` do MAME que representa um dispositivo que traduz uma cadeia de valores em cores.

As cores são representadas no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores das cores do canal não são multiplicados previamente pelo valor alpha. Os valores do canal devem ser empacotados em bytes com 32 bits inteiros não assinados pelo valor do canal alfa, na ordem alpha, vermelho, verde, azul a partir do byte mais importante até o byte com menor importância.

Instanciação

manager.machine.palettes[tag]

Obtém um dispositivo paleta através da tag em relação ao dispositivo raiz do sistema ou nil caso o dispositivo não exista ou caso não seja um dispositivo paleta.

Métodos

palette:pen(índice)

Obtém o número da cadeia remapeada para o índice especificado da paleta.

palette:pen_color(pen)

Obtém a cor para o número da cadeia especificada.

palette:pen_contrast(pen)

Obtém o valor do contraste para o o número da cadeia especificada. O contraste é um valor de ponto flutuante.

palette:pen_indirect(índice)

Obtém o índice indireto da cadeia para um índice específico da cadeia.

palette:indirect_color(índice)

Obtém o índice indireto da cadeia de cores para um índice específico da cadeia.

palette:set_pen_color(pen, cor)

Define a cor para um número específico da cadeia. A cor pode ser definida como um único valor empacotado de 32 bits; ou valores individuais para os canais vermelho, verde e azul, nesta ordem.

palette:set_pen_red_level(pen, nível)

Define o valor do canal da cor vermelho para o número da cadeia especificada. Os valores dos outros canais não são afetados.

palette:set_pen_green_level(pen, nível)

Define o valor do canal da cor verde para o número da cadeia especificada. Os valores dos outros canais não são afetados.

palette:set_pen_blue_level(pen, nível)

Define o valor do canal da cor azul para o número da cadeia especificada. Os valores dos outros canais não são afetados.

palette:set_pen_contrast(pen, fator)

Define o valor do contraste para o número da cadeia especificada. O contraste é um valor de ponto flutuante.

palette:set_pen_indirect(pen, índice)

Define o índice indireto para um número específico da cadeia.

palette:set_indirect_color(índice, color)

Define um índice indireto da cor da cadeia para um índice específico da paleta. A cor pode ser definida como um único valor empacotado de 32 bits; ou valores individuais para os canais vermelho, verde e azul, nesta ordem.

palette:set_shadow_factor(fator)

Define o valor do contraste para o grupo "shadow" atual. O contraste é um valor de ponto flutuante.

palette:set_highlight_factor(fator)

Define o valor do contraste para o grupo atual em destaque. O contraste é um valor de ponto flutuante.

palette:set_shadow_mode(modos)

Define o modo "shadow". O valor é o índice da tabela "shadow" desejada.

Propriedades**palette.palette** (somente leitura)

A *paleta* adjacente gerenciada pelo dispositivo.

palette.entries (somente leitura)

A quantidade dos registros de cores na paleta.

palette.indirect_entries (somente leitura)

A quantidade de registros indiretos da cadeia na paleta.

palette.black_pen (somente leitura)

O índice fixo do registro da cor preta na cadeia.

palette.white_pen (somente leitura)

O índice fixo do registro da cor branca na cadeia.

palette.shadows_enabled (somente leitura)

Um booleano que indica se as cores "shadow" estão ativadas.

palette.highlights_enabled (somente leitura)

Um booleano que indica se as cores em destaque estão ativadas.

palette.device (somente leitura)

O dispositivo *subjacente*.

Dispositivo tela

Encapsula a classe screen_device do MAME que representa uma saída emulada de vídeo.

Instanciação

manager.machine.screens[tag]

Obtém um dispositivo tela através da tag em relação ao dispositivo raiz do sistema, ou nil caso o dispositivo não exista ou caso não seja um dispositivo tela.

Classes base

- *Dispositivo*

Métodos

screen:orientation()

Retorna o ângulo de rotação em graus (será um de 0, 90, 180 ou 270), ou se a tela está virada da esquerda para a direita e se está invertida de cima para baixo. Essa é a orientação final da tela depois que a orientação tenha sido definida na configuração do sistema e a rotação tenha sido aplicada.

screen:time_until_pos(v, [h])

Obtém o tempo restante até que o raster atinja a posição especificada. Caso o componente horizontal da posição não é seja informado, a predefinição é zero (0, ou seja, o início da linha). O resultado é um número de ponto flutuante em unidades de segundos.

screen:time_until_vblank_start()

Obtém o tempo restante até o início do intervalo de apagamento vertical. O resultado é um número de ponto flutuante em unidades de segundos.

screen:time_until_vblank_end()

Obtém o tempo restante até o final do intervalo de apagamento vertical. O resultado é um número de ponto flutuante em unidades de segundos.

screen:snapshot([nome_do_arquivo])

Salva uma captura da tela em formato PNG. Caso nenhum nome do arquivo seja informado, será usado o caminho e o formato padrão configurado para a captura da tela. Caso o nome do arquivo informado não seja um caminho absoluto, ele será interpretado em relação ao primeiro caminho que foi configurado. O nome do arquivo pode conter variáveis que serão substituídas pelo nome do sistema ou por um número incremental.

Caso contrário, retorna um erro caso a leitura do arquivo da captura da tela falhe ou `nil`.

screen:pixel(x, y)

Obtém o pixel no local informado. As coordenadas estão em pixels, com a origem no canto superior esquerdo da área visível, aumentando para a direita e para baixo. Retorna um índice da paleta ou de uma cor no formato RGB compactado num inteiro com 32 bits. Retorna zero (0) se o ponto informado estiver fora da área visível.

screen:pixels()

Retorna todos os pixels visíveis, assim como, a região visível da largura e da altura.

Os pixels retornam como inteiros com 32 bits encapsulados numa *string* binária ordenado em *Endian*. Os pixels são organizados em ordem maior da linha, da esquerda para direita e depois de cima para baixo. Os valores dos pixels são índices da paleta ou cores no formato RGB encapsuladas em inteiros com 32 bits.

screen:draw_box(left, up, right, down, [linha], [preenchimento])

Desenha um retângulo delineado com bordas nas posições informadas.

As coordenadas são números de ponto flutuante em unidades de pixels da tela emulada, com a origem em (0, 0). Observe que os pixels da tela emulada geralmente não são quadrados. O sistema de coordenadas é rotacionada caso a tela seja girada, o que geralmente é o caso para as telas no formato vertical. Antes da rotação, a origem está na parte superior esquerda e as coordenadas aumentam para a direita e para baixo. As coordenadas são limitadas à área da tela.

A abrangência das cores de preenchimento e da linha estão no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa. Os valores dos canais devem ser empacotados em bytes de um inteiro com 32 bits sem assinatura na ordem alfa, vermelho, verde, azul do byte mais importante para o de menor importância. Caso a cor da linha não seja informada, é usada a cor do texto da interface; caso a cor de preenchimento não seja informada, é usada a cor de fundo da interface.

screen:draw_line(x0, y0, x1, y1, [cor])

Desenha uma linha a partir de (x0, y0) a (x1, y1).

As coordenadas são números de ponto flutuante em unidades de pixels da tela emulada, com a origem em (0, 0). Observe que os pixels da tela emulada geralmente não são quadrados. O sistema de coordenadas é rotacionado caso a tela seja girada, o que geralmente é o caso para as telas no formato vertical. Antes da rotação, a origem está na parte superior esquerda e as coordenadas aumentam para a direita e para baixo. As coordenadas são limitadas à área da tela.

A abrangência da cor da linha está no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa. Os valores dos canais devem ser empacotados em bytes de um inteiro com 32 bits sem assinatura na ordem alfa, vermelho, verde, azul do byte mais importante para o de menor importância. Caso a cor da linha não seja informada, é usada a cor do texto da interface.

screen:draw_text(x|justify, y, text, [primeiro plano], [plano de fundo])

Desenha o texto na posição informada. Se a tela for rotacionada, o texto será girado.

Caso o primeiro argumento seja um número, o texto será alinhado à esquerda nesta coordenada X. Caso o primeiro argumento seja uma *string*, ela deve ser "left", "center" ou "right" para desenhar o texto alinhado à esquerda na borda esquerda da tela, centralizado horizontalmente na tela ou alinhado à direita na borda direita da tela respectivamente. O segundo argumento determina a coordenada Y da altura máxima do texto.

As coordenadas são números de ponto flutuante em unidades de pixels da tela emulada, com a origem em (0, 0). Observe que os pixels da tela emulada geralmente não são quadrados. O sistema de coordenadas é rotacionado caso a tela seja girada, o que geralmente é o caso para as telas no formato vertical. Antes da rotação, a origem está na parte superior esquerda e as coordenadas aumentam para a direita e para baixo. As coordenadas são limitadas à área da tela.

As cores do primeiro plano e do plano de fundo estão no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais da cor não são previamente multiplicados pelo valor alpha. Os valores do canal devem ser empacotados em bytes com 32 bits inteiros não assinados pelo valor do canal alfa, na ordem alpha, vermelho, verde, azul a partir do byte mais importante até o byte com menor importância. Caso a cor do primeiro plano não seja informado, a cor do texto da interface será usada; caso a cor de fundo não seja informada, a cor do fundo da interface será usada.

Propriedades

screen.width (somente leitura)

A largura do bitmap produzido pela tela emulada em pixels.

screen.height (somente leitura)

A altura do bitmap produzido pela tela emulada em pixels.

screen.refresh (somente leitura)

A taxa de atualização configurada da tela em Hertz (isso pode não refletir o valor atual).

screen.refresh_attoseconds (somente leitura)

O intervalo de atualização configurado da tela em "attosegundos" (isso pode não refletir o valor atual).

screen.xoffset (somente leitura)

O *offset* predefinido da posição X da tela. Este é um número de ponto flutuante onde um (1) corresponde ao tamanho X do contêiner da tela. Isso pode ser útil para restaurar o valor original após ajustar o *offset* X através do contêiner da tela.

screen.yoffset (somente leitura)

O *offset* predefinido da posição Y da tela. Este é um número de ponto flutuante onde um (1) corresponde ao tamanho Y do contêiner da tela. Isso pode ser útil para restaurar o valor original após ajustar o *offset* Y através do contêiner da tela.

screen.xscale (somente leitura)

O fator de escala original da tela X, como um número de ponto flutuante. Isso pode ser útil para restaurar o valor original após ajustar a escala X através do contêiner da tela.

screen.yscale (somente leitura)

O fator de escala original da tela Y, como um número de ponto flutuante. Isso pode ser útil para restaurar o valor original após ajustar a escala Y através do contêiner da tela.

screen.pixel_period (somente leitura)

O intervalo necessário para desenhar um pixel horizontal, como um número de ponto flutuante em unidades de segundos.

screen.scan_period (somente leitura)

O intervalo necessário para desenhar uma linha de varredura (incluindo o intervalo horizontal de apagamento), como um número de ponto flutuante em unidades de segundos.

screen.frame_period (somente leitura)

O intervalo necessário para desenhar um quadro completo (incluindo os intervalos de apagamento), como um número de ponto flutuante em unidades de segundos.

screen.frame_number (somente leitura)

A quantidade dos quadros da tela atual. Isso aumenta monotonicamente cada intervalo dos quadros.

screen.container (somente leitura)

O *contêiner do renderizador* usado para desenhar a tela.

screen.palette (somente leitura)

O *dispositivo paleta* é utilizado para traduzir os valores dos pixels para cores ou nil caso a tela utilize um formato de pixel de cor direta.

Dispositivo de imagem em fita cassette

Encapsula a classe `cassette_image_device` do MAME que representa um mecanismo cassette compacto normalmente usado por um computador doméstico para o armazenamento dos programas.

Instanciação

manager.machine.cassettes[tag]

Obtém a imagem de um dispositivo cassete por tag em relação ao dispositivo raiz do sistema ou nil caso o dispositivo não exista ou caso não seja a imagem de um dispositivo cassete.

Classes base

- *Dispositivo*
- *Dispositivos de interface de imagem*

Métodos

cassette:stop()

Desativa a reprodução.

cassette:play()

Ativa a reprodução. O cassete tocará se o motor estiver ativado.

cassette:forward()

Avança a reprodução.

cassette:reverse()

Retrocede a reprodução.

cassette:seek(tempo, de_onde)

Salte para a posição informada na fita. O tempo é um número de ponto flutuante em unidades de segundos, em relação ao ponto informado no argumento de_onde. O argumento de_onde deve ser "set", "cur" ou "end" para realizar a busca com relação ao início da fita, a posição atual ou o fim da fita, respectivamente.

Propriedades

cassette.is_stopped (somente leitura)

Um booleano que indica se a fita está parada (ou seja, não está gravando e nem reproduzindo).

cassette.is_playing (somente leitura)

Um booleano que indica se a reprodução está ativada (ou seja, o cassete vai reproduzir se o motor estiver ativado).

cassette.is_recording (somente leitura)

Um booleano que indica se a gravação está ativada (ou seja, o gravador da fita vai gravar se o motor estiver ativado).

cassette.motor_state (leitura e escrita)

Um booleano que indica se o motor do cassete está ativado.

cassette.speaker_state (leitura e escrita)

Um booleano que indica se o alto-falante do cassete está ativado.

cassette.position (somente leitura)

A posição atual como um número de ponto flutuante em unidades de segundos com relação ao início da fita.

cassette.length (somente leitura)

A duração da fita como um número de ponto flutuante em unidades de segundos, ou zero (0) caso nenhuma imagem da fita seja montada.

Dispositivos de interface de imagem

Encapsula a classe `device_image_interface` do MAME que é uma mistura implementada através dos dispositivos que podem carregar arquivos de imagem a partir de uma mídia.

Instanciação**manager.machine.images[tag]**

Obtém um dispositivo de imagem por tag em relação ao dispositivo do sistema raiz, ou nil caso o dispositivo não exista ou caso não seja um dispositivo de imagem da mídia.

Métodos**image:load(nome_do_arquivo)**

Carrega o arquivo informado como uma imagem de mídia. Retorna nil caso não haja erro ou um texto descrevendo o que houve de errado.

image:load_software(nome)

Carrega uma imagem da mídia descrita numa lista de software. Retorna nil caso não haja erro ou um texto descrevendo o que houve de errado.

image:unload()

Descarrega a imagem que foi montada.

image:create(nome_do_arquivo)

Cria e monta um arquivo de imagem da mídia com o nome informado. Retorna nil caso não haja erro ou um texto descrevendo o que houve de errado.

image:display()

Retorna uma *string* do “front panel display” para o dispositivo, caso seja compatível. Isso pode ser usado para exibir as informações de status, como a posição atual da cabeça ou do estado do motor.

Propriedades

image.is_readable (somente leitura)

Um booleano que indica se o dispositivo oferece suporte à leitura.

image.is_writeable (somente leitura)

Um booleano que indica se o dispositivo oferece suporte para gravação.

image.must_be_loaded (somente leitura)

Um booleano que indica se o dispositivo requer que uma imagem da mídia seja carregada para começar.

image.is_reset_on_load (somente leitura)

Um booleano que indica se o dispositivo requer uma reinicialização forçada para alterar as imagens da mídia (geralmente para slots de cartucho que contêm um hardware adicional para os chips de memória).

image.image_type_name (somente leitura)

Uma *string* para categorizar o dispositivo da mídia.

image.instance_name (somente leitura)

O nome da instância do dispositivo na configuração atual. Isso é usado para configurar a carga da imagem da mídia na linha de comando ou nos arquivos INI. Isso não é estável, pode ter um número anexado que pode mudar dependendo da configuração do slot.

image.brief_instance_name (somente leitura)

O nome curto da instância do dispositivo na configuração atual. Isto é, usado para definir a imagem da mídia que será carregada na linha de comando ou nos arquivos INI. Isso não é estável, pode ter um número anexado que pode mudar dependendo da configuração do slot.

image.formatlist[] (somente leitura)

O *formato da imagem da mídia* são suportados pelo dispositivo, indexado por nome. O operador do índice e dos métodos `index_of` têm complexidade $O(n)$; todas as outras operações compatíveis têm complexidade $O(1)$.

image.exists (somente leitura)

Um booleano que indica se um arquivo de imagem da mídia está montado.

image.readonly (somente leitura)

Um booleano que indica se um arquivo de imagem da mídia está montado em modo de somente leitura.

image.filename (somente leitura)

O caminho completo para o arquivo montado da imagem da mídia ou `nil` se nenhuma imagem da mídia estiver montada.

image.crc (somente leitura)

A verificação de redundância cíclica com 32 bits do conteúdo do arquivo da imagem montada caso a imagem não tenha sido carregada a partir de uma lista de software, é montado como somente leitura e não for um CD-ROM, caso contrário é zero (0).

image.loaded_through_softlist (somente leitura)

Um booleano que indica se a imagem da mídia montada foi carregada a partir de uma lista de software ou false caso nenhuma imagem da mídia tenha sido montada.

image.software_list_name (somente leitura)

O nome curto da lista de software caso a imagem da mídia montada tenha sido carregada a partir de uma lista de software.

image.software_longname (somente leitura)

O nome completo do item do software caso a imagem da mídia montada tenha sido carregada a partir de uma lista de software ou caso contrário, nil.

image.software_publisher (somente leitura)

O editor do item do software caso a imagem da mídia montada tenha sido carregada a partir de uma lista de software ou caso contrário, nil.

image.software_year (somente leitura)

O ano de lançamento do item do software caso a imagem da mídia montada tenha sido carregada a partir de uma lista de software ou caso contrário, nil.

image.software_parent (somente leitura)

O nome abreviado do item do software principal caso a imagem da mídia montada tenha sido carregada a partir de uma lista de software ou caso contrário, nil.

image.device (somente leitura)

O *dispositivo* subjacente.

Dispositivo de interface slot

Encapsula a classe `device_slot_interface` do MAME que é uma mistura implementada através dos dispositivos que instanciam um dispositivo herdado que foi definido pelo usuário.

Instanciação**manager.machine.slots[tag]**

Obtém um dispositivo slot através da tag com relação ao dispositivo raiz do sistema ou nil caso o dispositivo não exista ou caso não seja um dispositivo slot.

Propriedades

slot.fixed (somente leitura)

Um booleano que indica se este é um slot com um cartão informado na configuração do sistema que não possa ser alterado pelo usuário.

slot.has_selectable_options (somente leitura)

Um booleano que indica se o slot tem alguma opção selecionável pelo usuário (ao contrário das opções que só podem ser selecionadas programaticamente, normalmente para os slots fixos ou para carregar as imagens da mídia).

slot.options[] (somente leitura)

As *opções do slot* que descrevem os dispositivos herdados que podem ser instanciados pelo slot, indexados pelo valor da opção. Os métodos `at` e `index_of` possuem complexidade $O(n)$; todas as outras operações compatíveis têm complexidade $O(1)$.

slot.device (somente leitura)

O *dispositivo* subjacente.

Estado da entrada do dispositivo

Envelopa a classe `device_state_entry` do MAME, permite acesso aos nomes dos registos expostos por um *dispositivo*. É compatível com a conversão de "string" para exibição.

Instanciação

manager.machine.devices[tag].state[símbolo]

Obtém o estado da entrada para um determinado dispositivo através de um símbolo.

Propriedades

entry.value (leitura e escrita)

O valor numérico do estado da entrada, seja como um número inteiro ou de ponto flutuante. É gerado um erro caso haja a tentativa de definir um valor do estado numa entrada que seja de apenas leitura.

entry.symbol (somente leitura)

O nome simbólico do estado da entrada.

entry.visible (somente leitura)

Retorna um booleano indicando se o estado da entrada deve ser mostrada na visualização de registo da depuração.

entry.writeable (somente leitura)

Retorna um booleano indicando se é possível alterar o valor do estado da entrada.

entry.is_float (somente leitura)

Retorna um booleano indicando se o valor do estado da entrada é um número de ponto flutuante.

entry.datamask (somente leitura)

Uma máscara de bits com valores válidos de bits para o estado com valor inteiro das entradas.

entry.datasize (somente leitura)

O tamanho do valor subjacente em bytes para o estado com valor inteiro das entradas.

entry.max_length (somente leitura)

O comprimento máximo da string de exibição para o estado da entrada.

Formato da imagem da mídia

Encapsula a classe `image_device_format` do MAME que descreve o formato do arquivo da mídia compatível através da *dispositivo de interface de imagem*.

Instanciação

manager.machine.images[tag].formatlist[nome]

Obtém um formato da imagem da mídia compatível com um determinado dispositivo através de um nome.

Propriedades

format.name (somente leitura)

Um nome abreviado usado para identificar o formato. Isso geralmente corresponde a extensão do nome do arquivo principal usado para o formato.

format.description (somente leitura)

O nome completo do formato.

format.extensions[] (somente leitura)

Produz uma tabela das extensões do nome do arquivo usados no formato.

format.option_spec (somente leitura)

Uma *string* que descreve as opções disponíveis durante a criação do formato da imagem da mídia. A *string* não se destina a ser legível para humanos.

Opções do slot

Encapsula a classe `device_slot_interface::slot_option` do MAME que representa um dispositivo herdado da *dispositivos de interface slot* que podem ser instanciados para configuração.

Instanciação

manager.machine.slots[tag].options[nome]

Obtém uma opção do slot para uma determinada *dispositivos de interface slot* através do nome (ou seja, o valor usado para selecionar a opção).

Propriedades

option.name (somente leitura)

O nome da opção do slot. Este é o valor usado para selecionar esta opção na linha de comando ou num arquivo INI.

option.device_fullname (somente leitura)

O nome completo da exibição do tipo do dispositivo instanciado por esta opção.

option.device_shortcode (somente leitura)

O nome abreviado do tipo de dispositivo instanciado por esta opção.

option.selectable (somente leitura)

Um Booleano que indica se a opção pode ser selecionada pelo usuário (as opções que não são selecionáveis pelo usuário geralmente são usados para os slots fixos ou para carregar as imagens da mídia).

option.default_bios (somente leitura)

A configuração padrão da BIOS para o dispositivo instanciado usando esta opção, ou nil caso a BIOS informada nas definições da ROM do dispositivo seja usada.

option.clock (somente leitura)

A frequência do *clock* configurada para o dispositivo instanciado usando esta opção. Este é um número inteiro com 32 bits não assinado. Se os oito primeiros bits mais importantes forem configurados, é uma proporção da frequência do *clock* do dispositivo principal, com o numerador nos bits 12-23 e o denominador nos bits 0-11. Se os 8 bits mais importantes não estiverem todos configurados, a frequência será em Hertz.

9.3.4 Classes dos sistemas de memória Lua

A interface Lua do MAME expõe vários objetos da memória do sistema, incluindo os espaços de endereçamento, compartilhamentos, seus bancos e as regiões da memória. Os *scripts* podem ler e escrever a partir do sistema emulado da memória.

- *O gerenciador de memória*
- *O espaço de endereçamento da memória*
- *O manipulador pass-through*
- *O mapa de endereçamento da memória*
- *As entradas do endereçamento da memória*
- *Dados do manipulador do mapa de endereçamento da memória*
- *Compartilhamento da memória*
- *Banco de memória*
- *Região da memória*

O gerenciador de memória

Encapsula a classe `memory_manager` do MAME que permite os compartilhamentos da memória, os bancos e as regiões num sistema que será enumerado.

Instanciação

manager.machine:memory()

Obtém a instância do gerenciador global da memória para o sistema emulado.

Propriedades

memory.shares[]

O *compartilhamento da memória* no sistema, indexada pela tag absoluta. Os métodos `at` e `index_of` têm $O(n)$ complexidade; todas outras operações compatíveis têm complexidade $O(1)$.

memory.banks[]

Os *banco da memória* no sistema, indexada pela tag absoluta. Os métodos `at` e `index_of` têm $O(n)$ complexidade; todas outras operações compatíveis têm complexidade $O(1)$.

memory.regions[]

As *regiões da memória* no sistema, indexada pela tag absoluta. Os métodos `at` e `index_of` têm $O(n)$ complexidade; todas outras operações compatíveis têm complexidade $O(1)$.

O espaço de endereçamento da memória

Encapsula a classe `address_space` do MAME que representa um espaço do endereço pertencente a um dispositivo.

Instanciação

`manager.machine.devices[tag].spaces[nome]`

Obtém o espaço do endereço com um nome específico para um determinado dispositivo. Observe que esses nomes são específicos para o tipo do dispositivo.

Métodos

`space:read_i{8,16,32,64}(endereço)`

Lê um valor inteiro assinado com o tamanho em bits do endereço informado.

`space:read_u{8,16,32,64}(endereço)`

Lê um valor inteiro não assinado com o tamanho em bits a partir do endereço informado.

`space:write_i{8,16,32,64}(endereço, valor)`

Grava um valor inteiro assinado com o tamanho em bits no endereço informado.

`space:write_u{8,16,32,64}(endereço, valor)`

Grava um valor inteiro não assinado com o tamanho em bits para o endereço informado.

`space:readv_i{8,16,32,64}(endereço)`

Lê um valor inteiro assinado com o tamanho em bits a partir do endereço virtual informado. O endereço é traduzido com a intenção da leitura da depuração. Retorna zero se a tradução do endereço falhar.

`space:readv_u{8,16,32,64}(endereço)`

Lê um valor inteiro não assinado com o tamanho em bits a partir do endereço informado. O endereço é traduzido com a intenção da leitura da depuração. Retorna zero se a tradução do endereço falhar.

`space:writenv_i{8,16,32,64}(endereço, valor)`

Grava um valor inteiro assinado com o tamanho em bits para o endereço virtual informado. O endereço é traduzido com a intenção de gravação da depuração. Não escreva se a tradução do endereço falhar.

`space:writenv_u{8,16,32,64}(endereço, valor)`

Grava um valor inteiro não assinado com o tamanho em bits para o endereço informado. O endereço é traduzido com a intenção de gravação da depuração. Não grava se a tradução do endereço falhar.

`space:read_direct_i{8,16,32,64}(endereço)`

Lê um valor inteiro assinado com o tamanho em bits do endereço informado, um byte de cada vez, obtendo um ponteiro de leitura para cada byte do endereço. Caso um ponteiro de leitura não pode ser obtido para o byte de um endereço, o byte do resultado correspondente será zero.

space:read_direct_u{8,16,32,64}(endereço)

Lê um valor inteiro não assinado com o tamanho em bits a partir do endereço informado, um byte de cada vez, obtendo um ponteiro de leitura para cada byte informado. Caso a leitura de um ponteiro não possa ser obtido para o endereço do byte, o resultado do byte correspondente será zero.

space:write_direct_i{8,16,32,64}(endereço, valor)

Grava um valor inteiro assinado com o tamanho em bits no endereço informado, um byte de cada vez, obtendo um ponteiro de gravação para cada endereço do byte. Caso um ponteiro de escrita não possa ser obtido para o endereço de um byte, o byte correspondente não será escrito.

space:write_direct_u{8,16,32,64}(endereço, valor)

Grava um valor inteiro não assinado com o tamanho em bits para o endereço informado, um byte de cada vez, obtendo um ponteiro de gravação para cada byte informado. Caso um ponteiro de gravação não possa ser obtido para o endereço de um byte, o byte correspondente não será escrito.

space:read_range(início, fim, largura, [passo])

Lê um intervalo de endereços como uma *string* binária. O endereço final deve ser maior ou igual ao endereço inicial. A largura deve ser 8, 16, 30 ou 64. Caso o passo seja informado, ele deve ser um número positivo dos elementos.

space:add_change_notifier(callback)

Adiciona um *callback* para receber as notificações das alterações do manipulador no espaço de endereçamento. A função de *callback* é repassada numa *string* simples como um argumento, seja *r* caso os manipuladores de leitura tenham se alterado de forma potencial, *w* no caso dos manipuladores de escrita e *rw* em ambos os casos.

Retorna um *notificador da assinatura*.

space:install_read_tap(início, fim, nome, callback)

Faz a instalação de um *manipulador pass-through* que fará a recepção das notificações de leitura a partir de uma determinada faixa de endereços no espaço de endereçamento da memória. O início e o fim do endereço são abrangentes. O nome deve ser uma *string* e o *callback* uma função.

O *callback* repassa 3 argumentos para o *offset* do acesso, para a leitura dos dados e a máscara de acesso à memória. A compensação é a compensação absoluta no espaço de endereçamento. Para alterar os dados que estão sendo lidos, retorne o valor alterado da função do *callback* como um número inteiro. Caso o *callback* não retorne um valor inteiro, os dados não serão alterados.

space:install_write_tap(início, fim, nome, callback)

Faz a instalação de um *manipulador pass-through* que fará a recepção das notificações de escrita a partir de uma determinada faixa de endereços no espaço de endereçamento da memória. O nome deve ser uma *string* e o *callback* uma função.

O *callback* repassa 3 argumentos para o *offset* do acesso, para a escrita dos dados e a máscara de acesso à memória. A compensação é a compensação absoluta no espaço de endereçamento. Para alterar os dados que estão sendo escritos, retorne o valor alterado da função do *callback* como um número inteiro. Caso o *callback* não retorne um valor inteiro, os dados não serão alterados.

Propriedades

space.name (somente leitura)

O nome da exibição do espaço do endereço.

space.shift (somente leitura)

A granularidade do endereço para o espaço do endereçamento informado como a transferência necessária para traduzir o endereço de um byte num endereço nativo. Os valores positivos se transferem para o bit mais importante (à esquerda) e os valores negativos se transferem em direção ao byte menos importante (à direita).

space.index (somente leitura)

O índice do espaço com base zero. Alguns índices do espaço têm significados especiais para o depurador.

space.address_mask (somente leitura)

A máscara do espaço do endereço.

space.data_width (somente leitura)

A largura dos dados para o espaço em bits.

space.endianness (somente leitura)

O Endianness do espaço ("big" ou "little").

space.map (somente leitura)

O *mapa do endereçamento da memória* configurado para o espaço ou nil.

O manipulador pass-through

Faz o rastreo do manipulador "pass-through" instalado num *espaço de endereçamento da memória*. Ele recebe as notificações dos acessos numa determinada faixa de endereçamento, pode alterar os dados que são lidos ou escritos se assim for preciso. Observe que as chamadas de retorno do manipulador "pass-through" não são executadas como corrotinas.

Instanciação

manager.machine.devices[tag].spaces[nome]:install_read_tap(início, fim, nome, callback)

Faz a instalação de um manipulador "pass-through" que receberá as notificações das leituras a partir de uma determinada faixa de endereçamento num *espaço de endereçamento da memória*.

manager.machine.devices[tag].spaces[nome]:install_write_tap(início, fim, nome, callback)

Faz a instalação de um manipulador "pass-through" que receberá as notificações das escritas a partir de uma determinada faixa de endereçamento num *espaço de endereçamento da memória*.

Métodos

passthrough.reinstall()

Reinstala o manipulador *pass-through* no espaço de endereçamento da memória. Pode ser necessário caso o manipulador seja removido devido as alterações dos outros manipuladores dentro do espaço de endereçamento da memória.

passthrough.remove()

Faz a remoção do manipulador *pass-through* do espaço de endereçamento da memória. O *callback* associado não será invocado em resposta aos futuros acessos da memória.

Propriedades

passthrough.addrstart (somente leitura)

Abrange o início do endereço da faixa do endereçamento que foi alterado pelo manipulador *pass-through* (quando o manipulador for notificado no endereçamento mais baixo por exemplo).

passthrough.addrend (somente leitura)

Abrange o fim do endereço da faixa do endereçamento que foi alterado pelo manipulador *pass-through* (quando o manipulador for notificado no endereçamento mais alto por exemplo).

passthrough.name (somente leitura)

O nome de exibição para o manipulador *pass-through*.

O mapa de endereçamento da memória

Encapsula a classe `address_map` do MAME que é usada para configurar os manipuladores para um espaço do endereço.

Instanciação

manager.machine.devices[tag].spaces[nome].map

Obtém o mapa do endereço configurado para o espaço de um endereço ou `nil` caso nenhum mapa seja configurado.

Propriedades

map.spacenum (somente leitura)

A quantidade do espaço de endereço do espaço de endereço onde o mapa está associado.

map.device (somente leitura)

O dispositivo que possui o endereçamento onde o mapa está associado.

map.unmap_value (somente leitura)

O valor constante para retornar a partir das leituras não mapeadas.

map.global_mask (somente leitura)

Máscara global que será aplicada a todos os endereços ao acessar o espaço.

map.entries[] (somente leitura)

As *entradas do endereçamento da memória* não configuradas no mapa do endereço. Usa índices inteiros com base 1. O operador do índice e o método `at` tem complexidade $O(n)$.

As entradas do endereçamento da memória

Encapsula a classe `address_map_entry` do MAME que representa uma entrada na configuração de um mapa de endereços.

Instanciação

manager.machine.devices[tag].spaces[nome].map.entries[índice]

Obtém uma entrada a partir do mapa configurado para um espaço de endereço.

Propriedades

entry.address_start (somente leitura)

Endereço inicial do intervalo da entrada.

entry.address_end (somente leitura)

Endereço final do intervalo da entrada (inclusive).

entry.address_mirror (somente leitura)

Bits do espelho do endereço.

entry.address_mask (somente leitura)

Bits da máscara do endereço. É válido apenas para os manipuladores.

entry.mask (somente leitura)

Máscara da pista, indicando quais as linhas dos dados do barramento estão conectadas ao manipulador.

entry.cswidth (somente leitura)

A largura do gatilho para um manipulador que não está conectado a todas as linhas de dados.

entry.read (somente leitura)

Os *dados do manipulador do mapa de endereçamento da memória* para a leitura do manipulador.

entry.write (somente leitura)

Os *dados do manipulador do mapa de endereçamento da memória* para a escrita no manipulador.

entry.share (somente leitura)

A tag do compartilhamento da memória para tornar as entradas da RAM acessíveis ou `nil`.

entry.region (somente leitura)

A tag explícita da região da memória para entradas da ROM, ou `nil`. Para entradas da ROM, o `nil` deduz a região da tag do dispositivo.

entry.region_offset (somente leitura)

O *offset* inicial na região da memória para as entradas da ROM.

Dados do manipulador do mapa de endereçamento da memória

Encapsula a classe `map_handler_data` do MAME que oferece os dados de configuração para os manipuladores nos mapas dos endereços.

Instanciação

`manager.machine.devices[tag].spaces[nome].map.entries[índice].read`

Obtém os dados do manipulador de leitura para uma entrada do mapa dos endereços.

`manager.machine.devices[tag].spaces[nome].map.entries[índice].write`

Obtém os dados do manipulador de gravação para uma entrada do mapa dos endereços.

Propriedades

`data.handlertype` (somente leitura)

O tipo do manipulador. Será um dos "none", "ram", "rom", "nop", "unmap", "delegate", "port", "bank", "submap" ou "unknown". Observe que os vários valores dos tipos do manipulador podem produzir "delegate" ou "unknown".

`data.bits` (somente leitura)

A largura dos dados para o manipulador em bits.

`data.name` (somente leitura)

Nome de exibição para o manipulador ou nil.

`data.tag` (somente leitura)

A tag para portas de E/S, os bancos da memória ou nil.

Compartilhamento da memória

Encapsula a classe `memory_share` do MAME que representa um nome alocado na região da memória.

Instanciação

`manager.machine.memory.shares[tag]`

Obtém um compartilhamento da memória através da tag absoluta ou nil caso o compartilhamento da memória não exista.

`manager.machine.devices[tag]:memshare(tag)`

Obtém um compartilhamento da memória através da tag em relação a um dispositivo ou nil caso o compartilhamento da memória não exista.

Métodos

share:read_i{8,16,32,64}(offs)

Lê um valor inteiro assinado do tamanho em bits do *offset* informado no compartilhamento da memória.

share:read_u{8,16,32,64}(offs)

Lê um valor inteiro não assinado com o tamanho em bits a partir do *offset* do compartilhamento da memória.

share:write_i{8,16,32,64}(offs, valor)

Grava um valor inteiro assinado com o tamanho em bits para o *offset* informado no compartilhamento da memória.

share:write_u{8,16,32,64}(offs, valor)

Grava um valor inteiro não assinado com o tamanho em bits para o *offset* informado no compartilhamento da memória.

Propriedades

share.tag (somente leitura)

A marca absoluta do compartilhamento da memória.

share.size (somente leitura)

O tamanho do compartilhamento da memória em bytes.

share.length (somente leitura)

O comprimento do compartilhamento da memória em elementos da largura nativa.

share.endianness (somente leitura)

O endianness do compartilhamento da memória ("big" ou "little").

share.bitwidth (somente leitura)

A largura do elemento nativo do compartilhamento da memória em bits.

share.bytewidth (somente leitura)

A largura do elemento nativo do compartilhamento da memória em bytes.

Banco de memória

Encapsula a classe `memory_bank` do MAME que representa uma região determinada da memória.

Instanciação

manager.machine.memory.banks[tag]

Obtém uma região da memória por tag absoluta, ou nil caso o banco da memória não exista.

manager.machine.devices[tag]:membank(tag)

Obtém uma região da memória por tag relativa a um dispositivo ou nil caso o banco da memória não exista.

Propriedades

bank.tag (somente leitura)

A tag absoluta do banco da memória.

bank.entry (leitura e escrita)

O número da entrada com base zero atualmente selecionado.

Região da memória

Encapsula a classe `memory_region` do MAME que representa a região da memória usada para armazenar dados somente leitura como as ROMs ou o resultado fixo do que for descriptografado.

Instanciação

manager.machine.memory.regions[tag]

Obtém uma região de memória por tag absoluta ou nil caso nenhuma região da memória exista.

manager.machine.devices[tag]:memregion(tag)

Obtém uma região da memória por tag relativa a um dispositivo ou nil caso o banco da memória não exista.

Métodos

region:read(offsets, len)

Lê até o comprimento especificado em bytes a partir do *offset* definido na região da memória. Os bytes lidos serão retornados como uma *string*. A tentativa de leitura além da região da memória faz com que a *string* retorne mais curta do que foi solicitado. Observe que os dados estarão na sequência de bytes do host.

region:read_i{8,16,32,64}(offsets)

Lê um valor inteiro assinado do tamanho em bits do *offset* informado na região da memória. O *offset* é definido em bytes. A tentativa de leitura além da região da memória retorna zero.

region:read_u{8,16,32,64}(offsets)

Lê um valor inteiro não assinado com o tamanho em bits a partir do *offset* da região da memória. O *offset* é definido em bytes. A tentativa de leitura além da região da memória retorna zero.

region:write_i{8,16,32,64}(offs, valor)

Grava um valor inteiro assinado com o tamanho em bits para o *offset* informado da região da memória. O *offset* é definido em bytes. A tentativa de escrever além da região da memória não surte nenhum efeito.

region:write_u{8,16,32,64}(offs, valor)

Grava um valor inteiro não assinado com o tamanho em bits para o *offset* informado na região da memória. O *offset* é definido em bytes. A tentativa de escrever além da região da memória não surte nenhum efeito.

Propriedades

region.tag (somente leitura)

A tag absoluta da região da memória.

region.size (somente leitura)

O tamanho da região da memória em bytes.

region.length (somente leitura)

O comprimento da região da memória com elementos nativos de largura.

region.endianness (somente leitura)

O endianness da região de memória ("big" ou "little").

region.bitwidth (somente leitura)

A largura do elemento nativo da região da memória em bits.

region.bytestwidth (somente leitura)

A largura do elemento nativo da região da memória em bytes.

9.3.5 Classes do sistema de entrada Lua

Permite que os scripts obtenham informações recebidas do usuário e acessem as portas de E/S da entrada do sistema emulado.

- *Gerenciador da porta E/S*
- *Gerenciador de teclado natural*
- *Dispositivo de entrada do teclado*
- *Porta de E/S*
- *Campo da porta de E/S*
- *Estado do campo da porta de E/S em tempo real*
- *Tipo da entrada*
- *Gerenciador da entrada*
- *Código da condição da entrada*
- *Sequência da condição da entrada*
- *Sequência da entrada*
- *Host da classe do dispositivo da entrada*
- *Host do dispositivo da entrada*
- *Host do item do dispositivo da entrada*
- *Gerenciador da entrada da IU*

Gerenciador da porta E/S

Encapsula a classe `ioport_port` do MAME que representa uma porta emulada de E/S, assim como, lida com as configurações da entrada.

Instanciação

manager.machine:ioport()

Obtém a instância do gerenciador global da porta de E/S para o sistema emulado.

Métodos

ioport:count_players()

Retorna a quantidade dos controladores do jogador no sistema.

ioport:type_pressed(tipo, [jogador])

Retorna um booleano indicando se a entrada informada foi atualmente pressionada. O tipo da entrada pode ser um valor enumerado ou um *tipo de acesso*. Caso o tipo de acesso seja um valor enumerado, número do jogador é um índice com base zero. Se o número do jogador não for informado, será presumido que seja zero, sendo o tipo da entrada um tipo de acesso, o número do jogador não poderá ser informado de forma separada.

ioport:type_name(tipo, [jogador])

Retorna o nome da exibição para o tipo da entrada informada e o número do jogador. O tipo da entrada é um valor enumerado. O número do jogador é um índice com base zero. Se o número do jogador não for informado, será presumido que seja zero.

ioport:type_group(tipo, [jogador])

Retorna a entrada do grupo para o tipo da entrada informada e o número do jogador. O tipo da entrada é um valor enumerado. O número do jogador é um índice com base zero. Retorna um valor inteiro informando o agrupamento para a entrada. Se o número do jogador não for informado, será presumido que seja zero.

Deve ser invocado com os valores obtidos a partir dos campos da porta de E/S para informar o agrupamento canônico da configuração da entrada numa IU.

ioport:type_seq(tipo, [jogador], [tipo_da_sequência])

Obtenha a configuração do tipo da *sequência de entrada* para um determinado tipo de acesso, o número do jogador e o tipo da sequência. O tipo da entrada pode ser um valor enumerado ou um *tipo de acesso*. Caso o tipo de acesso seja um valor enumerado, o número do jogador pode ser informado como um índice com base zero, caso contrário, será presumido que seja zero, sendo o tipo da entrada um tipo de acesso, o número do jogador não poderá ser informado de forma separada. Havendo a informação do tipo da sequência, ela deve ser "standard", "increment" ou "decrement", na sua ausência, será presumido que seja "standard".

Isso fornece acesso à configuração geral da entrada.

ioport:set_type_seq (tipo, [jogador], tipo_da_sequência, sequência)

Define a configuração da *sequência de entrada* para um determinado tipo de acesso, o número do jogador e o tipo da sequência. O tipo da entrada pode ser um valor enumerado ou um *tipo de acesso*. Caso o tipo de acesso seja um valor enumerado, o número do jogador pode ser informado como um índice com base zero, sendo o tipo da entrada um tipo de acesso, o número do jogador não poderá ser informado de forma separada. A sequência deve ser "standard", "increment" ou "decrement".

Isso permite que a configuração geral da entrada possa ser definida.

ioport:token_to_input_type(string)

Retorna uma *string* com o tipo da entrada e o número do jogador para o tipo da entrada do token informado.

ioport:input_type_to_token(tipo, [jogador])

Retorna uma *string* do token para o tipo da entrada informada e o número do jogador. Se o número do jogador não for informado, será presumido que seja zero.

Propriedades**ioport.types[]** (somente leitura)

Obtém o *tipo de acesso* compatível. As teclas são índices arbitrários. Todas as operações compatíveis possuem complexidade O(1).

ioport.ports[]

Obtém a emulação da *porta de E/S* no sistema. Chaves são tags absolutas. Os métodos `at` e `index_of` têm complexidade O(n); todas as outras operações compatíveis têm complexidade O(1).

Gerenciador de teclado natural

Encapsula a classe `natural_keyboard` do MAME que gerencia o teclado emulado e as entradas do teclado.

Instanciação

`manager.machine.natkeyboard`

Obtém a instância do gerenciador do teclado natural global para o sistema que está sendo emulado.

Métodos

`natkeyboard:post(texto)`

Publique um texto literal no sistema emulado. O sistema deve ter uma entrada de teclado com os caracteres vinculados e o dispositivo correto da entrada do teclado deve estar ativado.

`natkeyboard:post_coded(texto)`

Publique o texto no sistema que está sendo emulado. Os códigos entre chaves são interpretados no texto. O sistema deve ter as entradas do teclado com os caracteres vinculados e o dispositivo correto da entrada do teclado deve estar ativado.

Os códigos reconhecidos são {BACKSPACE}, {BS}, {BKSP}, {DEL}, {DELETE}, {END}, {ENTER}, {ESC}, {HOME}, {INS}, {INSERT}, {PGDN}, {PGUP}, {SPACE}, {TAB}, {F1}, {F2}, {F3}, {F4}, {F5}, {F6}, {F7}, {F8}, {F9}, {F10}, {F11}, {F12} e {QUOTE}.

`natkeyboard:paste()`

Publique o conteúdo da área de transferência do host no sistema emulado. O sistema deve ter as entradas do teclado com caracteres vinculados e o dispositivo correto da entrada do teclado deve estar ativado.

`natkeyboard:dump()`

Retorna uma *string* com uma descrição legível do teclado e dos dispositivos de entrada do teclado numérico no sistema, se eles estão ativados e os seus caracteres vinculados.

Propriedades

`natkeyboard.empty` (somente leitura)

Um booleano que indica se o buffer da entrada do gerenciador do teclado natural está vazio.

`natkeyboard.full` (somente leitura)

Um booleano que indica se o buffer da entrada do gerenciador do teclado natural está cheio.

`natkeyboard.can_post` (somente leitura)

Um booleano que indica se o sistema emulado suporta a postagem dos dados dos caracteres através do gerenciador do teclado natural.

`natkeyboard.is_posting` (somente leitura)

Um booleano que indica se os dados postados dos caracteres estão sendo entregues ao sistema que está sendo emulado.

natkeyboard.in_use (leitura e escrita)

Um booleano que indica se o modo “teclado natural” está ativado. Quando O modo “teclado natural” está ativado o gerenciador do teclado natural traduz a entrada de caractere do host para pressionamentos da tecla do sistema emulado.

natkeyboard.keyboards[]

Obtém o *dispositivo de entrada do teclado* no sistema que está sendo emulado, indexado através da tag absoluta do dispositivo. O índice get tem O(n) complexidade; todas as outras operações compatíveis têm complexidade O(1).

Dispositivo de entrada do teclado

Representa um teclado ou dispositivo de entrada do teclado que é gerenciado pelo *gerenciador do teclado natural*. Observe que isso não é uma classe de *dispositivo*.

Instanciação

manager.machine.natkeyboard.keyboards[tag]

Obtém o dispositivo da entrada do teclado com a tag informada ou nil se a tag não corresponder a um dispositivo da entrada do teclado.

Propriedades

keyboard.device (somente leitura)

O *dispositivo* subjacente.

keyboard.tag (somente leitura)

A tag absoluta do dispositivo subjacente.

keyboard.basetag (somente leitura)

O último componente da tag do dispositivo subjacente ou "root" para o dispositivo raiz do sistema.

keyboard.name (somente leitura)

A descrição legível para as pessoas do tipo do dispositivo subjacente.

keyboard.shortname (somente leitura)

O identificador do tipo do dispositivo subjacente.

keyboard.is_keypad (somente leitura)

Um booleano que indica se o dispositivo subjacente possui as entradas do teclado numérico, mas não para a entradas do teclado. Isso é usado para determinar quais dispositivos da entrada do teclado deve ser ativado por padrão.

keyboard.enabled (leitura e escrita)

Um booleano que indica se as entradas do teclado e/ou do teclado numérico do dispositivo estão ativados.

Porta de E/S

Encapsula a classe `ioport_port` do MAME que representa uma porta emulada de E/S.

Instanciação

manager.machine.ioport.ports[tag]

Obtém uma porta de E/S emulada através da tag absoluta ou `nil` caso a tag não corresponda a uma porta de E/S.

manager.machine.devices[devtag]:ioport(porttag)

Obtém uma porta de E/S emulada através da tag relativa a um dispositivo ou `nil` se não houver nenhuma porta de E/S.

Métodos

port:read()

Leia o valor de entrada atual. Retorna um número inteiro com 32 bits.

port:write(valor, máscara)

Grave nos campos da saída da porta de E/S que são configuradas na máscara informada. A máscara e o valor devem ser inteiros e com 32 bits. Observe que isso não define os valores para os campos da entrada.

port:field(máscara)

Obtenha o primeiro *campo da porta de E/S* correspondente aos bits que são definidos na máscara informada ou `nil` se não houver nenhum campo correspondente.

Propriedades

port.device (somente leitura)

O dispositivo que possui a porta de E/S.

port.tag (somente leitura)

A etiqueta absoluta da porta E/S

port.active (somente leitura)

Uma máscara indicando quais os bits da porta E/S correspondem aos campos ativos (isto é, os bits que não utilizados ou não foram atribuídos).

port.live (somente leitura)

O estado ativo da porta de E/S.

port.fields[] (somente leitura)

Obtém uma tabela do *campo da porta de E/S* indexados por nome.

Campo da porta de E/S

Encapsula a classe `ioport_field` do MAME que representa um campo dentro da porta de E/S.

Instanciação

`manager.machine.ioport.ports[tag]:field(máscara)`

Obtém um campo para a porta informada através dos bits da máscara.

`manager.machine.ioport.ports[tag].fields[nome]`

Obtém um campo para a porta informada através do nome de exibição.

Métodos

`field:set_value(valor)`

Define o valor do campo da porta de E/S. Para os campos digitais, o valor é comparado com zero para determinar se o campo deve estar ativo; para os campos analógicos, o valor deve estar alinhado à direita e no intervalo correto.

`field:clear_value()`

Limpa o valor programado excedente e restaura o comportamento regular do campo.

`field:set_input_seq(tipo_da_sequência, sequência)`

Define a *sequência de entrada* para o tipo da sequência informada. Isso é usado para definir as configurações da entrada por sistema. O tipo da sequência deve ser "standard", "increment" ou "decrement".

`field:input_seq(tipo_da_sequência)`

Obtenha a *sequência de entrada* configurada para o tipo da sequência informada. Isso obtém as configurações da entrada por sistema. O tipo da sequência deve ser "standard", "increment" ou "decrement".

`field:set_default_input_seq(tipo_da_sequência, sequência)`

Define a *sequência de entrada* predefinida para o tipo da sequência informada. É usado para definir as configurações gerais da entrada. O tipo da sequência deve ser "standard", "increment" ou "decrement".

`field:default_input_seq(tipo_da_sequência)`

Obtém a *sequência de entrada* predefinida para o tipo da sequência informada. Obtém as configurações gerais da entrada. O tipo da sequência deve ser "standard", "increment" ou "decrement".

`field:keyboard_codes(shift)`

Obtém uma tabela dos caracteres correspondentes ao campo para o estado do shift informado. O estado do shift é uma máscara de bits das teclas ativas do shift.

Propriedades

field.device (somente leitura)

O dispositivo que possui a porta que o campo pertence.

field.port (somente leitura)

A *porta de E/S* que o campo pertence.

field.live (somente leitura)

O *estado do campo da porta de E/S em tempo real* do campo.

field.type (somente leitura)

O tipo da entrada do campo. Este é um valor enumerado.

field.name (somente leitura)

O nome da exibição do campo.

field.default_name (somente leitura)

O nome da configuração para o campo do sistema emulado (não pode ser substituído por *scripts* ou *plug-ins*).

field.player (somente leitura)

O número do jogador para o campo com base zero.

field.mask (somente leitura)

Os Bits na porta de E/S correspondente a este campo.

field.defvalue (somente leitura)

O valor predefinido do campo.

field.minvalue (somente leitura)

O valor mínimo permitido nos campos analógicos ou nil nos campos digitais.

field.maxvalue (somente leitura)

O valor máximo permitido nos campos analógicos ou nil nos campos digitais.

field.sensitivity (somente leitura)

A sensibilidade ou ganho para os campos analógicos ou nil nos campos digitais.

field.way (somente leitura)

A quantidade das direções permitidas através do restritor da placa/portão para um joystick digital ou zero (0) para as outras entradas.

field.type_class (somente leitura)

O tipo da classe para o campo da entrada para um dos "keyboard", "controller", "config", "dipswitch" ou "misc".

field.is_analog (somente leitura)

Um booleano que indica se o campo é um eixo analógico ou controle posicional.

field.is_digital_joystick (somente leitura)

Um booleano que indica se o campo corresponde ao comutador de um joystick digital.

field.enabled (somente leitura)

Um booleano que indica se o campo está ativado.

field.optional (somente leitura)

Um booleano que indica se o campo é opcional e não é obrigatório para uso no sistema que está sendo emulado.

field.cocktail (somente leitura)

Um booleano que indica se o campo é usado apenas quando o sistema é configurado para um gabinete de mesa tipo coquetel.

field.toggle (somente leitura)

Um booleano que indica se o campo corresponde a uma botão do hardware tipo liga/desliga ou um botão de pressão.

field.rotated (somente leitura)

Um booleano que indica se o campo corresponde a um controle que é rotacionado em relação à orientação padrão.

field.analog_reverse (somente leitura)

Um booleano que indica se o campo corresponde a um controle analógico que aumenta na direção oposta à convenção (por exemplo, valores maiores quando um pedal é solto ou um joystick é movido para a esquerda).

field.analog_reset (somente leitura)

Um booleano que indica se o campo corresponde a um incremental da posição da entrada (por exemplo, um dial ou eixo do trackball) que deve ser redefinida para zero para cada quadro do vídeo.

field.analog_wraps (somente leitura)

Um booleano que indica se o campo corresponde a uma entrada analógica que encapsula a partir de uma extremidade da sua faixa para a outra (por exemplo, uma posição incremental como a entrada de um dial ou o eixo do trackball).

field.analog_invert (somente leitura)

Um booleano que indica se o campo corresponde a uma entrada analógica que tem o seu valor complementado.

field.impulse (somente leitura)

Um booleano que indica se o campo corresponde a uma entrada digital que é ativado por um determinado período de tempo fixo.

field.crosshair_scale (somente leitura)

O fator de escala para traduzir o intervalo do campo para a posição da mira. Um valor de um (1) que traduz o intervalo total do campo para a largura total ou altura da tela.

field.crosshair_offset (somente leitura)

O *offset* para traduzir o intervalo do campo para a posição da mira.

field.user_value (leitura e escrita)

O valor da chave DIP ou das definições da configuração.

field.settings[] (somente leitura)

Obtém uma tabela das configurações ativadas atualmente para um interruptor DIP ou o campo de configuração, indexado por valor.

Estado do campo da porta de E/S em tempo real

Encapsula a classe `ioport_field_live` do MAME que representa o estado em tempo real de uma porta de E/S.

Instanciação

manager.machine.ioport.ports[tag]:field(máscara).live

Obtém o estado em tempo real para um campo da porta de E/S.

Propriedades

live.name

O nome da exibição do campo.

Tipo da entrada

Envelopa a classe `input_type_entry` do MAME, faz a representação do tipo de acesso ou do tipo de acesso da interface do usuário no emulador. Os tipo de acesso são identificados de forma única através da combinação do índice do jogador e do tipo do valor da sua enumeração.

Instanciação

manager.machine.ioport.types[índice]

Obtém um tipo de acesso compatível.

Propriedades

type.type (somente leitura)

Um valor enumerado que representa o tipo de acesso.

type.group (somente leitura)

Um valor inteiro que fornece o agrupamento para o tipo de acesso. Deve ser utilizado para fornecer um agrupamento canônico numa configuração de entrada da interface do usuário (IU).

type.player (somente leitura)

Número do jogador com base zero ou zero para controles não relacionados com o jogador.

type.token (somente leitura)

A *string* de um token para o tipo de acesso, usado nos arquivos de configuração.

type.name (somente leitura)

O nome do tela para o tipo de acesso.

type.is_analog (somente leitura)

Retorna um booleano indicando se o tipo de acesso é analógico ou digital. As entradas que possuam apenas as condições ligado ou desligado são consideradas digitais, enquanto todas as outras são consideradas analógicas ainda que elas representem apenas valores discretos ou posições.

Gerenciador da entrada

Encapsula a classe `input_manager` do MAME que lê os dispositivos da entrada do host e verifica se as entradas configuradas estão ativas.

Instanciação

manager.machine:input()

Obtém a instância global do gerenciador da entrada para o sistema que está sendo emulado.

Métodos

input:code_value(código)

Obtém o valor atual para a entrada do host correspondente ao código informado. Retorna um valor inteiro assinado onde zero é a posição neutra.

input:code_pressed(código)

Retorna um booleano indicando se o código informado da entrada do host correspondente tem um valor diferente de zero (ou seja, não é uma posição neutra).

input:code_pressed_once(código)

Retorna um booleano indicando se o código informado da entrada do host correspondente saiu da posição neutra desde a última vez que foi verificado através desta função. O gerenciador da entrada pode rastrear uma quantidade de entradas desta forma.

input:code_name(código)

Obtenha o nome de exibição para um código da entrada.

input:code_to_token(código)

Obtenha a *string* do token para um código da entrada. Isso deve ser usado ao salvar uma configuração.

input:code_from_token(token)

Converta uma *string* do token num código de entrada. Retorna o código de entrada inválido se o token não for válido ou caso pertença a um dispositivo de entrada que não está presente.

input:seq_pressed(sequência)

Retorna um booleano indicando se a *sequência da entrada* informada foi realmente pressionada.

input:seq_clean(sequência)

Remova os elementos inválidos da *sequência da entrada* informada. Retorna uma nova, sequência limpa da entrada.

input:seq_name(sequência)

Obtenha o texto de exibição para uma *sequência da entrada*.

input:seq_to_tokens(sequência)

Converta uma *sequência da entrada* numa *string* token. Isso deve ser usado quando for salvar na configuração.

input:seq_from_tokens(tokens)

Converta uma *string* token numa *sequência da entrada*. Isso deve ser usado quando for carregar uma configuração.

input:axis_code_poller()

Retorna um *código da condição da entrada* para obter um código da entrada do host analógico.

input:switch_code_poller()

Retorna um *código da condição da entrada* para obter um código da entrada do interruptor do host.

input:keyboard_code_poller()

Retorna um *código da condição da entrada* para a obtenção de um código da entrada do interruptor do host que considera apenas a entrada dos dispositivos do teclado.

input:axis_sequence_poller()

Retorna uma *sequência da condição da entrada* para obter uma *sequência da entrada* para configurar uma entrada analógica.

input:axis_sequence_poller()

Retorna uma *sequência da condição da entrada* para obter uma *sequência da entrada* para configurar uma entrada digital.

Propriedades**input.device_classes[]** (somente leitura)

Pega uma tabela host *host da classe do dispositivo da entrada* indexada por nome.

Código da condição da entrada

Encapsula a classe `input_code_poller` do MAME que é usada para pesquisar as entradas do host que estão sendo ativadas.

Instanciação

`manager.machine.input:axis_code_poller()`

Retorna uma condição do código da entrada que pesquisa as entradas analógicas que estão sendo ativadas.

`manager.machine.input:switch_code_poller()`

Retorna uma condição do código da entrada que pesquisa as entradas do interruptor do host que estão sendo ativadas.

`manager.machine.input:keyboard_code_poller()`

Retorna uma condição do código da entrada que pesquisa as entradas do interruptor do host que estão sendo ativadas, considerando apenas os dispositivos da entrada do teclado.

Métodos

`poller:reset()`

Redefine a lógica da pesquisa. As entradas do interruptor ativo são apagadas e as entradas das posições analógica são definidas.

`poller:poll()`

Retorna um código da entrada correspondente à primeira entrada relevante do host que foi ativado desde a última vez que o método foi invocado. Retorna um código de entrada inválido caso nenhuma entrada relevante tenha sido ativada.

Sequência da condição da entrada

Encapsula a classe da condição `input_sequence_poller` do MAME que permite que os usuários atribuam combinações na entrada do host para as entradas emuladas e outras ações.

Instanciação

`manager.machine.input:axis_sequence_poller()`

Retorna uma condição da sequência da entrada para atribuir as entradas do host a uma entrada analógica.

`manager.machine.input:switch_sequence_poller()`

Retorna uma condição da sequência da entrada para atribuir as entradas do host a entrada de um interruptor.

Métodos

poller.start([seq])

Comece a obter. Caso uma sequência seja fornecida, ela será usada como uma sequência inicial para entradas analógicas, o usuário pode alternar entre a faixa completa e as porções positivas e negativas de um eixo; para as entradas do interruptor, um código “or” é anexado e o usuário pode adicionar uma combinação alternativa da entrada do host.

poller.poll()

Obtém a entrada do usuário e atualiza a sequência, caso seja apropriado. Retorna um booleano que indica se a entrada da sequência está completa. Se este método retornar falso, você deve continuar com o processo de obtenção.

Propriedades

poller.sequence (somente leitura)

A *sequência da entrada* atual. É atualizado durante o processo de obtenção. É possível para que a sequência se torne inválida.

poller.valid (somente leitura)

Um booleano que indica se a sequência da entrada atual é válida.

poller.modified (somente leitura)

Um booleano que indica se a sequência foi alterada através de alguma entrada do usuário desde o início do processo.

Sequência da entrada

Encapsula a classe `input_seq` do MAME que representa a combinação das entradas do host que possam ser lidos ou designados para uma determinada entrada da emulação. As sequências da entrada podem ser manipuladas usando os métodos do *gerenciador da entrada*. Use um *obtentor da sequência de entrada* para obter uma sequência da entrada a partir do usuário.

Instanciação

emu.input_seq()

Cria uma sequência vazia da entrada.

emu.input_seq(seq)

Cria uma cópia de uma sequência já existente da entrada.

Métodos

seq:reset()

Limpa a sequência da entrada, removendo todos os itens.

seq:set_default()

Define a sequência da entrada num único item contendo o valor meta que definindo qual a configuração padrão deve ser usada.

Propriedades

seq.empty (somente leitura)

Um booleano indicando se a sequência da entrada está vazia (não possui quaisquer itens, indicando uma entrada sem atribuição).

seq.length (somente leitura)

A quantidade dos itens na sequência da entrada.

seq.is_valid (somente leitura)

Um booleano indicando se a sequência da entrada é válida. Para ser válido, deve conter pelo menos um item, todos os itens devem possuir códigos válidos, todos os grupos dos produtos devem conter pelo menos um item que não seja negado e os itens referentes aos eixos absolutos e relativos não devem ser misturados dentro de um grupo de produtos.

seq.is_default (somente leitura)

Um booleano indicando se a sequência da entrada define se a configuração deve ser usada.

Host da classe do dispositivo da entrada

Encapsula a classe `input_class` do MAME que representa uma categoria da entrada do host dos dispositivos (por exemplo, teclados ou joysticks).

Instanciação

manager.machine.input.device_classes[nome]

Obtém uma entrada da classe do dispositivo por nome.

Propriedades

devclass.name (somente leitura)

O nome da classe do dispositivo.

devclass.enabled (somente leitura)

Um booleano que indica se a classe do dispositivo está ativo.

devclass.multi (somente leitura)

Um booleano que indica se a classe do dispositivo oferece suporte a vários dispositivos ou as entradas de todos os dispositivos da classe são combinadas e tratadas como um único dispositivo.

devclass.devices[] (somente leitura)

Obtém uma tabela *host do dispositivo da entrada* na classe. As chaves são os índices com base 1.

Host do dispositivo da entrada

Encapsula a classe `input_device` do MAME que representa um dispositivo da entrada do host.

Instanciação

manager.machine.input.device_classes[nome].devices[índice]

Obtém um dispositivo de entrada específica de um host.

Propriedades

inputdev.name (somente leitura)

Nome da exibição do dispositivo. Não há garantia de que isso seja exclusivo.

inputdev.id (somente leitura)

A *string* do identificador exclusivo para o dispositivo. Isso pode não ser legível para as pessoas.

inputdev.devindex (somente leitura)

O índice do dispositivo dentro da classe de dispositivo. Isso não é necessariamente o mesmo que o índice na propriedade `devices` da classe do dispositivo o índice do `devindex` podem não ser contíguos.

inputdev.items (somente leitura)

Obtém as tabelas *host do item do dispositivo da entrada*, indexado através da ID do item. A ID do item é um valor enumerado.

Host do item do dispositivo da entrada

Encapsula a classe `input_device_item` do MAME que representa uma única entrada do host (por exemplo uma chave, botão ou eixo).

Instanciação

manager.machine.input.device_classes[nome].devices[índice].items[id]

Obtém um item individual da entrada do host. A ID do item é um valor enumerado.

Propriedades

item.name (somente leitura)

O nome da exibição da entrada do item. Observe que este é apenas o nome do próprio item que não inclui o nome do dispositivo. O nome completo da exibição para o item pode ser obtido ao invocar o método `code_name` no *gerenciador da entrada* com o código do item.

item.code (somente leitura)

O código de identificação da entrada do item. Isso é usado por vários métodos do *gerenciador da entrada*.

item.token (somente leitura)

A *string* token do item da entrada. Observe que este é um fragmento do token para o próprio item que não inclui a parte do dispositivo. O token completo para o item pode ser obtido ao invocar o método `code_to_token` no *gerenciador da entrada* com o código do item.

item.current (somente leitura)

O valor atual do item. Este é um número inteiro assinado onde zero é a posição neutra.

Gerenciador da entrada da IU

Encapsula a classe `ui_input_manager` do MAME que é usada para a entrada de alto nível.

Instanciação

manager.machine.uiinput

Obtém a instância do gerenciador da entrada global da IU para a sistema.

Métodos

uiinput:reset()

Limpa os eventos pendentes e os estados da interface de entrada do usuário. Deve ser chamado ao encerrar o modo de um estado onde a entrada é tratada diretamente (ao configurar uma combinação de entrada por exemplo).

uiinput:pressed(tipo)

Retorna um booleano indicando se a entrada da IU informada foi pressionada. O tipo da entrada é um valor enumerado.

uiinput:pressed_repeat(tipo, velocidade)

Retorna um booleano indicando se a entrada da IU informada foi pressionada ou a repetição automática foi disparada na velocidade informada. O tipo da entrada é um valor enumerado; a velocidade é um intervalo em sessenta avos de um segundo.

Propriedades

uiinput.presses_enabled (leitura e escrita)

Se o gerenciador da entrada da IU verificará se há atualizações do quadro das entradas da IU.

9.3.6 Classes do sistema de renderização Lua

O sistema de renderização é responsável por desenhar o que você vê nas janelas do MAME, incluindo as telas emuladas, a arte e os elementos da interface do usuário.

- *Limites do renderizador*
- *Renderização da cor*
- *Paleta*
- *Bitmap*
- *Renderização da textura*
- *Gerenciador do renderizador*
- *Alvo do renderizador*
- *Contêiner do renderizador*
- *Configurações do contêiner do usuário*
- *Arquivo layout*
- *Visualização do layout*
- *Visualização do item do layout*

Limites do renderizador

Encapsula a classe `render_bounds` do MAME que representa um retângulo usando as coordenadas de ponto flutuante.

Instanciação

`emu.render_bounds()`

Cria os limites da renderização de um objeto que representa uma unidade quadrada com o canto superior esquerdo em (0, 0) e canto inferior direito em (1, 1). Observe que ao renderizar as coordenadas do alvo elas não possuem necessariamente as mesmas escalas X e Y, então isso pode não representar a geração de um quadrado.

`emu.render_bounds (esquerda, cima, direita, baixo)`

Cria os limites da renderização de um objeto representando um retângulo com o canto superior esquerdo em (x0, y0) e o canto inferior direito em (x1, y1).

Todos os argumentos devem ser em números de ponto flutuante.

Métodos

bounds:includes(x, y)

Retorna um booleano indicando se o ponto informado está dentro do retângulo. O retângulo deve ser normalizado para que funcione (direito maior que o esquerdo e baixo maior do que cima). Os argumentos devem ser números de ponto flutuante.

bounds:set_xy(esquerda, cima, direita, baixo)

Define a posição e o tamanho do retângulo nos termos das posições das bordas. Todos os argumentos devem ser em números de ponto flutuante.

bounds:set_wh(esquerda, cima, largura, altura)

Define a posição e o tamanho do retângulo nos termos da posição do canto superior esquerdo, da largura e da altura. Todos os argumentos devem ser em números de ponto flutuante.

Propriedades

bounds.x0 (leitura e escrita)

A coordenada mais à esquerda no retângulo (ou seja, a coordenada X do lado da borda esquerda ou no canto superior esquerdo).

bounds.x1 (leitura e escrita)

A coordenada mais à direita no retângulo (ou seja, a coordenada X da borda direita ou do canto inferior direito).

bounds.y0 (leitura e escrita)

A coordenada superior no retângulo (ou seja, a coordenada Y da parte da borda superior ou no canto superior esquerdo).

bounds.y1 (leitura e escrita)

A coordenada mais inferior do retângulo (ou seja, a coordenada Y da borda inferior ou do canto inferior direito).

bounds.width (leitura e escrita)

A largura do retângulo. Ao definir esta propriedade a posição da extremidade direita muda.

bounds.height (leitura e escrita)

A altura do retângulo. Ao definir esta propriedade a posição da borda inferior muda.

bounds.aspect (somente leitura)

A proporção entre a largura e altura do retângulo. Observe que o alvo geralmente é usado em coordenadas de renderização que não necessariamente têm escalas X e Y iguais. Um retângulo representando um quadrado na saída final não necessariamente tem uma proporção 1.

Renderização da cor

Encapsula a classe `render_color` do MAME que representa um formato de cor ARGB (alfa, vermelho, verde, azul). Os canais são valores de ponto flutuante que variam entre zero (0, alfa transparente ou sem cor) a um (1, opaco ou totalmente colorido). Os valores do canal da cor não são pré-multiplicados pelo valor do canal alfa.

Instanciação

`emu.render_color()`

Cria um objeto colorido representando o branco opaco (todos os canais definidos como 1). Este é o valor da identidade, a multiplicação do ARGB por este valor não irá alterar uma cor.

`emu.render_color(a, r, g, b)`

Cria a renderização de um objeto colorido com o alfa, vermelho, verde e os valores do canal azul. Os argumentos devem ser todos números de ponto flutuante no intervalo de zero (0) até um (1).

Métodos

`color:set(a, r, g, b)`

Define os valores dos canais alfa, vermelho, verde e azul da cor do objeto. Todos os argumentos devem ser números de ponto flutuante no intervalo de zero (0) até um (1).

Propriedades

`color.a` (leitura e escrita)

O valor alfa, no intervalo entre zero (0, transparente) até um (1, opaco).

`color.r` (leitura e escrita)

O valor do canal vermelho, no intervalo entre zero (0 desligado) até um (1, intensidade total).

`color.g` (leitura e escrita)

O valor do canal verde, no intervalo entre zero (0 desligado) até um (1, intensidade total).

`color.b` (leitura e escrita)

O valor do canal azul, no intervalo entre zero (0 desligado) até um (1, intensidade total).

Paleta

Encapsula a classe `palette_t` do MAME que representa uma tabela de cores que pode ser buscada pelo índice com base zero. As paletas sempre contêm as entradas adicionais especiais para o preto e branco.

Cada cor possui um valor associado do ajuste do contraste. Cada grupo de ajuste possui valores associados do ajuste do brilho e do contraste. A paleta também possui valores gerais para o ajuste do brilho, do contraste e do gama.

As cores são representadas no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais da cor não são previamente multiplicados pelo valor alpha. Os valores do canal devem ser empacotados em bytes com 32 bits inteiros não assinados pelo valor do canal alfa, na ordem alpha, vermelho, verde, azul a partir do byte mais importante até o byte menos importante.

Instanciação

emu.palette(cores, [grupos])

Cria uma paleta com uma quantidade determinada de cores e de grupos para o ajuste de brilho e contraste. A quantidade dos grupos das cores retorna para um caso não seja definido. As cores são inicializadas em preto, o ajuste do brilho é inicializado com o valor 0,0, o contraste com 1,0 e o gama com 1,0.

Métodos

palette:entry_color(índice)

Obtém a cor especificada no índice com base zero.

Os valores do índice variam entre zero e a quantidade das cores na paleta menos um. Retorna a cor preta caso o índice seja maior ou igual à quantidade de cores da paleta.

palette:entry_contrast(índice)

Obtém o ajuste de contraste para a cor no índice com base zero. Este é um número de ponto flutuante.

Indexa a faixa de valores entre zero e a quantidade das cores na paleta menos um. Retorna 1.0 caso o índice seja maior ou igual à quantidade de cores da paleta.

palette:entry_adjusted_color(índice, [grupo])

Obtém a cor com os ajustes aplicados de brilho, contraste e gama.

Caso o grupo seja definido, os valores do índice das cores variam de zero a quantidade das cores na paleta menos um e os valores do grupo variam entre zero e a quantidade dos grupos de ajuste na paleta menos um.

Quando um grupo não é definido, os valores do índice variam entre zero a quantidade de cores multiplicado pelo número dos grupos de ajuste mais um. Os valores do índice podem ser calculados multiplicando o índice do grupo com base zero pela quantidade de cores na paleta e adicionando o índice das cores com base zero. Os dois últimos valores do índice correspondem às entradas especiais para o preto e o branco respectivamente.

Retorna a cor preta caso a combinação definida do índice e do grupo de ajuste seja inválida.

palette:entry_set_color(índice, cor)

Define a cor especificada no índice com base zero. A cor pode ser definida através de um único valor de 32 bits compactado ou como valores individuais para os canais vermelho, verde e azul (nesta ordem).

Os valores do índice variam entre zero a quantidade das cores na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:entry_set_red_level(índice, nível)

Define o valor da cor vermelha do canal especificada no índice com base zero. Os outros valores não são afetados.

Os valores do índice variam entre 0 até a quantidade das cores na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:entry_set_green_level(índice, nível)

Define o valor da cor verde do canal especificada no índice com base zero. Os outros valores não são afetados.

Os valores do índice variam entre 0 até a quantidade das cores na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:entry_set_blue_level(índice, nível)

Define o valor da cor azul do canal especificada no índice com base zero. Os outros valores não são afetados.

Os valores do índice variam entre 0 até a quantidade das cores na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:entry_set_contrast(índice, nível)

Define o valor de ajuste do contraste da cor especificada no índice com base zero. Este é um número de ponto flutuante.

Os valores do índice variam entre 0 até a quantidade das cores na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:group_set_brightness(grupo, brilho)

Define o valor de ajuste do brilho para o grupo de ajuste especificada no índice com base zero. Este é um número de ponto flutuante.

Os valores do grupo variam entre 0 até a quantidade dos grupos de ajuste na paleta menos um. Gera um erro caso o valor do índice seja inválido.

palette:group_set_contrast(grupo, contraste)

Define o valor de ajuste do contraste para o grupo de ajuste especificada no índice com base zero. Este é um número de ponto flutuante.

Os valores do grupo variam entre 0 até a quantidade dos grupos de ajuste na paleta menos um. Gera um erro caso o valor do índice seja inválido.

Propriedades

palette.colors (somente leitura)

A quantidade de entradas das cores de cada grupo de cores na paleta.

palette.groups (somente leitura)

A quantidade dos grupos de cores na paleta.

palette.max_index (somente leitura)

A quantidade valida dos índices de cores na paleta.

palette.black_entry (somente leitura)

O índice da entrada especial para a cor preta.

palette.white_entry (somente leitura)

O índice da entrada especial para a cor branca.

palette.brightness (somente escrita)

Ajuste geral do brilho para a paleta. Este é um número de ponto flutuante.

palette.contrast (somente escrita)

Ajuste geral do contraste para a paleta. Este é um número de ponto flutuante.

palette.gamma (somente escrita)

Ajuste geral do gama para a paleta. Este é um número de ponto flutuante.

Bitmap

Envelopa a implementação das classes `bitmap_t` e `bitmap_specific`, que representam bitmaps bidimensionais armazenados em ordem de maior prioridade na fila. As coordenadas do pixel têm base zero, aumentando para a direita e para baixo. Vários formatos de pixels são suportados.

Instanciação

emu.bitmap_ind8(paleta, [largura, altura], [xslop, yslop])

Cria um bitmap indexado com 8 bit. Numa *paleta*, cada pixel é um índice de 8 bits não assinado, base zero. Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y, assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_ind16(paleta, [largura, altura], [xslop, yslop])

Cria um bitmap indexado com 16 bit. Numa *paleta*, cada pixel é um índice de 16 bits não assinado, base zero. Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y, assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_ind32(paleta, [largura, altura], [xslop, yslop])

Cria um bitmap indexado com 32 bit. Numa *paleta*, cada pixel é um índice de 32 bits não assinado, base zero. Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y, assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_ind64(paleta, [largura, altura], [xslop, yslop])

Cria um bitmap indexado com 64 bit. Numa *paleta*, cada pixel é um índice de 64 bits não assinado, base zero. Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y, assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_yuy16([largura, altura], [xslop], yslop])

Cria um bitmap em formato Y'CbCr com subamostragem de croma¹ 4:2:2 (os pares dos pixels horizontais têm valores individuais de luma², porém, compartilham os valores de croma). Cada pixel é um valor inteiro de 16 bits. O byte mais importante do valor de 8 bits não assinado da cor do pixel, é o componente Y' (luma). Para cada par horizontal dos pixels, o byte menos importante do primeiro valor do pixel (base zero par X coordenada) é o valor Cb de 8 bits assinado para o par de pixels, e o byte menos importante do segundo pixel (base zero ímpar X coordenada) é o valor Cr de 8 bits assinado para o par de pixels.

¹ Cor

² Luminância

Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X , também é obrigatório definir um valor de inclinação Y . Na ausência dos valores de inclinação X e Y , assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y , assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_rgb32([largura, altura], [xslop, yslop])

Cria um bitmap em formato RGB sem o canal alfa (transparência). Cada pixel é representado por um valor inteiro de 16 bit. O byte mais importante do valor do pixel é ignorado. Os três bytes restantes, do mais importante ao menos importante, são valores de 8 bit sem assinatura dos canais vermelho, verde e azul (valores maiores correspondem a intensidades mais altas).

Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X , também é obrigatório definir um valor de inclinação Y . Na ausência dos valores de inclinação X e Y , assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y , assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_argb32([largura, altura], [xslop, yslop])

Cria um bitmap em formato ARGB. Cada pixel é representado por um valor inteiro de 32 bit. O byte mais importante do valor do canal do pixel é o valor alpha de 8 bit sem assinatura (transparência) onde valores menores são mais transparentes. Os três bytes restantes, do mais importante ao menos importante, são valores de 8 bit sem assinatura dos canais vermelho, verde e azul (valores maiores correspondem a intensidades mais altas). Os valores dos canais de cores não são previamente multiplicados pelos valores dos canais alpha.

Na ausência dos valores de largura e de altura, estes valores são considerados como sendo zero. Caso a largura seja definida, é obrigatório que a altura também seja especificada. Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X , também é obrigatório definir um valor de inclinação Y . Na ausência dos valores de inclinação X e Y , assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y , assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O recorte inicial do retângulo é definido em todo o bitmap.

emu.bitmap_yuy16(source, [x0, y0, x1, y1])

Cria um bitmap em formato Y'CbCr com subamostragem 4:2:2 de croma representando uma exibição de uma parte do bitmap existente. O retângulo de recorte inicial é definido para os limites da exibição. O bitmap original será bloqueado, evitando o redimensionamento e a realocação.

Na ausência das coordenadas, o novo bitmap representará uma visualização do retângulo do recorte atual do bitmap original. Caso as coordenadas sejam fornecidas, o novo bitmap representará uma exibição do retângulo com o canto superior esquerdo em (x0, y0) e com o canto inferior direito em (x1, y1) no bitmap original. As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas.

O bitmap original deve ser de propriedade do *script* Lua e deve usar o formato Y'CbCr. Gera um erro caso as coordenadas sejam especificadas representando um retângulo que não esteja totalmente contido dentro do retângulo de recorte do bitmap original.

emu.bitmap_rgb32(source, [x0, y0, x1, y1])

Cria um bitmap em formato RGB com subamostragem 4:2:2 de croma representando uma exibição de uma parte do bitmap existente. O retângulo de recorte inicial é definido para os limites da exibição. O bitmap original será bloqueado, evitando o redimensionamento e a realocação.

Na ausência das coordenadas, o novo bitmap representará uma visualização do retângulo do recorte atual do bitmap original. Caso as coordenadas sejam fornecidas, o novo bitmap representará uma exibição do retângulo com o canto superior esquerdo em (x0, y0) e com o canto inferior direito em (x1, y1) no bitmap original. As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas.

O bitmap original deve ser de propriedade do *script* Lua e deve usar o formato RGB. Gera um erro caso as coordenadas sejam especificadas representando um retângulo que não esteja totalmente contido dentro do recorte do retângulo do bitmap original.

emu.bitmap_argb32(source, [x0, y0, x1, y1])

Cria um bitmap em formato ARGB com subamostragem 4:2:2 de croma representando uma exibição de uma parte do bitmap existente. O retângulo de recorte inicial é definido para os limites da exibição. O bitmap original será bloqueado, evitando o redimensionamento e a realocação.

Na ausência das coordenadas, o novo bitmap representará uma visualização do retângulo do recorte atual do bitmap original. Caso as coordenadas sejam fornecidas, o novo bitmap representará uma exibição do retângulo com o canto superior esquerdo em (x0, y0) e com o canto inferior direito em (x1, y1) no bitmap original. As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas.

O bitmap original deve ser de propriedade do *script* Lua e deve usar o formato ARGB. Gera um erro caso as coordenadas sejam especificadas representando um retângulo que não esteja totalmente contido dentro do recorte do retângulo do bitmap original.

emu.bitmap_argb32.load(data)

Cria um bitmap em formato **ARGB** a partir de dados em formato PNG, JPEG (JFIF/EXIF) ou Microsoft DIB (BMP). Gera um erro caso os dados sejam inválidos ou caso o formato não seja compatível.

Métodos

bitmap:cliprect()

Retorna as coordenadas esquerda, cima, direita e baixo do retângulo de recorte do bitmap. As coordenadas estão em unidades de pixels. As coordenadas baixo e direita são inclusivas.

bitmap:reset()

Define a largura e a altura como zero e libera o armazenamento dos pixels caso o bitmap possua o seu próprio armazenamento ou libera o bitmap original se ele representar uma exibição de um outro bitmap.

O bitmap deve ser de propriedade do *script* Lua. Gera um erro caso o armazenamento do bitmap seja referenciado por um outro bitmap ou uma *textura*.

bitmap:allocate(largura, altura, [xslop, yslop])

Reatribui o armazenamento para o bitmap, define a sua largura, a sua altura e ajusta o retângulo de recorte para a totalidade do bitmap. Caso o bitmap já possua um armazenamento alocado, ele sempre será liberado e realocado; se o bitmap representar uma visão de um outro bitmap, o bitmap original será liberado bitmap, o bitmap original será liberado. O armazenamento do pixel será preenchido com o valor zero.

Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (o armazenamento será dimensionado para se ajustar ao conteúdo do bitmap). Quando a largura e/ou a altura for menor ou igual à zero, nenhum armazenamento será alocado, independentemente dos valores de inclinação X e Y, assim como, ambos os valores para a largura e para a altura do bitmap serão definidos como zero.

O bitmap deve ser de propriedade do *script* Lua. Gera um erro caso o armazenamento do bitmap seja referenciado por um outro bitmap ou uma *textura*.

bitmap:resize(largura, altura, [xslop, yslop])

Altera a largura, a altura e define o recorte do retângulo em todo o bitmap.

Os valores de inclinação X e Y definem a quantidade do armazenamento extra em pixels para reservar cada linha à esquerda/direita e respectivamente a parte superior/inferior de cada coluna. Ao definir o valor de inclinação X, também é obrigatório definir um valor de inclinação Y. Na ausência dos valores de inclinação X e Y, assume-se que seus valores também sejam zero (as linhas serão armazenadas de forma contígua, a linha superior será colocada no início do armazenamento do bitmap).

Caso o bitmap já possua um armazenamento alocado e for grande o suficiente para o tamanho atualizado, ele será usado sem ser liberado; caso seja muito pequeno, este sempre será liberado e realocado. Se o bitmap representar uma exibição de um outro bitmap, o bitmap original será liberado. Quando o armazenamento do pixel for alocado, ele será preenchido com o valor zero (na utilização de um armazenamento já existente, o seu conteúdo não será alterado).

O bitmap deve ser de propriedade do *script* Lua. Gera um erro caso o armazenamento do bitmap seja referenciado por um outro bitmap ou uma *textura*.

bitmap:wrap(source, [x0, y0, x1, y1])

Faz com que o bitmap represente uma visualização de uma parte de um outro bitmap e define o recorte do retângulo para os limites da visualização.

Na ausência das coordenadas, o novo bitmap representará uma visualização do retângulo do recorte atual do bitmap original. Caso as coordenadas sejam fornecidas, o novo bitmap representará uma visualização do retângulo com o canto superior esquerdo em (x0, y0) e com o canto inferior direito em (x1, y1) no bitmap original. As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas.

O bitmap original será bloqueado, evitando o redimensionamento e a realocação. Caso o bitmap já possua um armazenamento alocado, ele sempre será liberado e realocado; se o bitmap representar uma visão de um outro bitmap, o bitmap original será liberado e o bitmap original será liberado.

O bitmap deve ser de propriedade do *script* Lua e devem usar o mesmo formato de pixel. Gera um erro caso as coordenadas sejam definidas representando um retângulo que não esteja totalmente contido dentro do retângulo de recorte do bitmap original; caso o bitmap seja referenciado por um outro bitmap ou *textura*; ou caso a origem e o destino sejam o mesmo bitmap.

bitmap:pix(x, y)

Retorna o valor da cor do pixel no local determinado. As coordenadas nas unidades dos pixels têm base zero.

bitmap:pixels([x0, y0, x1, y1])

Retorna os pixels, a largura e a altura da parte do bitmap com o canto superior esquerdo em (x0, y0) e com o canto inferior direito em (x1, y1). As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas. Na ausência das coordenadas, o retângulo do recorte do bitmap será usado.

Os pixels retornam empacotados numa string binária na ordem "*Endian*" do host. Os pixels são organizados na ordem principal da linha, da esquerda para a direita e de cima para baixo. O tamanho e o formato dos valores do pixel dependem do formato do bitmap. Gera um erro caso forem especificadas coordenadas que representem um retângulo não totalmente contido no retângulo de recorte do bitmap.

bitmap:fill(cor, [x0, y0, x1, y1])

Preenche uma parte do bitmap com o valor da cor especificada. Na ausência das coordenadas, o recorte do retângulo será preenchido; caso as coordenadas sejam fornecidas, a interseção do recorte do retângulo e do retângulo com canto superior esquerdo em (x0, y0) e canto inferior direito em (x1, y1) serão preenchidos. As coordenadas estão em unidades de pixels. As coordenadas do canto inferior direito são inclusivas.

bitmap:plot(x, y, cor)

Define o valor da cor do pixel no local especificado caso esteja dentro do recorte do retângulo. As coordenadas nas unidades dos pixels têm base zero.

bitmap:plot_box(x, y, largura, altura, cor)

Preenche a interseção do recorte do retângulo e o retângulo com o canto superior esquerdo (x, y) e a altura e largura especificadas com o valor especificada da cor. As coordenadas e as dimensões estão em unidades de pixels.

Propriedades

bitmap.palette (leitura e escrita)

A *paleta* usada para traduzir os valores do pixel para cores. Aplicável apenas para bitmaps que utilizem formatos de pixel indexados.

bitmap.width (somente leitura)

Largura do bitmap em pixels.

bitmap.height (somente leitura)

Altura do bitmap em pixels.

bitmap.rowpixels (somente leitura)

Passo da linha do armazenamento do bitmap em pixels. Ou seja, a diferença na compensação dos pixels no mesmo local horizontal em linhas consecutivas. Pode ser maior que a largura.

bitmap.rowbytes (somente leitura)

Passo da linha do armazenamento do bitmap em bytes. Ou seja, a diferença em endereços de byte dos pixels na mesma localização horizontal das linhas consecutivas.

bitmap.bpp (somente leitura)

O tamanho do tipo usado para representar os pixels no bitmap em bits (pode ser maior que a quantidade de bits importantes).

bitmap.valid (somente leitura)

Um booleano que indica se o bitmap tem armazenamento disponível (pode ser *false* para bitmaps vazios).

bitmap.locked (somente leitura)

Um booleano que indica se o armazenamento do bitmap é referenciado por outro bitmap ou *textura*.

Renderização da textura

Encapsula a classe `render_texture` do MAME, representa a textura que pode ser desenhada no *contêiner do renderizador*. As texturas renderizadas devem ser liberadas antes que a emulação em andamento seja encerrada.

Instanciação

manager.machine.render:texture_alloc(bitmap)

Cria uma *textura* com base num *bitmap*. O bitmap deve ser de propriedade do *script* Lua e deve usar o formato YCbCr, RGB ou ARGB. O bitmap original será bloqueado, evitando o redimensionamento e a realocação.

Métodos

texture:free()

Libera a textura. O armazenamento do bitmap subjacente será liberado.

Propriedades

texture.valid (somente leitura)

Um booleano que indica se a textura é válida (*false* no caso da textura ter sido liberada).

Gerenciador do renderizador

Encapsula a classe `render_manager` do MAME que é responsável pelo gerenciamento do destino da renderização e das texturas.

Instanciação

manager.machine.render

Obtém a instância do gerenciador da renderização global para a sessão emulada.

Métodos

render:texture_alloc(bitmap)

Cria uma *textura* com base num *bitmap*. O bitmap deve ser de propriedade do *script* Lua e deve usar o formato Y'CbCr, RGB ou ARGB. O bitmap original será bloqueado, evitando o redimensionamento e a realocação. As texturas renderizadas devem ser liberadas antes que a seção da emulação seja encerrada.

Propriedades

render.max_update_rate (somente leitura)

A taxa de atualização máxima em Hertz. Este é um número de ponto flutuante.

render.ui_target (somente leitura)

O *alvo do renderizador* usado para desenhar a interface do usuário (incluindo os menus, os controles deslizantes e as mensagens de pop-up). Geralmente é a primeira janela ou a primeira tela do host.

render.ui_container (somente leitura)

O *contêiner do renderizador* usado para desenhar a interface do usuário.

render.targets[] (somente leitura)

A lista da renderização dos alvos, incluindo as janelas e as telas geradas, bem como os alvos ocultos renderizados para coisas como a renderização das capturas da tela. Usa índices inteiros com base 1. O operador de índice e o método `at` têm O(n) complexidade.

Alvo do renderizador

Encapsula a classe `render_target` do MAME que representa a saída de um canal de vídeo. Pode ser uma janela, a tela do host ou um alvo oculto usado para a renderização da captura da tela.

Instanciação

manager.machine.render.targets[índice]

Obtenha a renderização de um alvo por índice.

manager.machine.render.ui_target

Obtenha a renderização de um alvo usado para exibir a interface do usuário (incluindo os menus, os controles deslizantes e as mensagens de pop-up). Geralmente é a primeira janela do host ou tela.

manager.machine.video.snapshot_target

Obtém o destino de renderização usado para produzir instantâneos e gravações de vídeo.

Propriedades

target.ui_container (somente leitura)

O *contêiner do renderizador* para desenhar elementos da interface do usuário sobre este alvo de renderização ou `nil` para alvos ocultos de renderização (alvos que não são mostrados diretamente ao usuário).

target.index (somente leitura)

O índice do destino de renderização com base 1. Isso tem complexidade $O(n)$.

target.width (somente leitura)

A geração da largura da renderização do alvo em pixels. Este é um número inteiro.

target.height (somente leitura)

A geração da altura da renderização do alvo em pixels. Este é um número inteiro.

target.pixel_aspect (somente leitura)

A geração da renderização da proporção entre a largura e a altura dos pixels. Isto é um número de ponto flutuante.

target.hidden (somente leitura)

Um booleano que indica se este alvo é uma renderização interna que não é exibido diretamente para o usuário (por exemplo, o alvo da renderização usado para criar as capturas da tela).

target.is_ui_target (somente leitura)

Um booleano que indica se este é o destino de renderização usado para exibir a interface do usuário.

target.max_update_rate (leitura e escrita)

A taxa de atualização máxima para a renderização do alvo em Hertz.

target.orientation (leitura e escrita)

Os sinalizadores de orientação do alvo. Esta é uma máscara de bits inteira, onde o bit 0 (0x01) é definido para espelhar horizontalmente, o bit 1 (0x02) é definido para espelhar verticalmente e o bit 2 (0x04) é definido para espelhar ao longo do canto superior esquerdo inferior e a diagonal direita.

target.view_names[]

Os nomes das visualizações disponíveis para a renderização deste alvo. Usa base 1 e índices inteiros. Os métodos `find` e o `index_of` têm $O(n)$ complexidade; todas as outras operações compatíveis têm complexidade $O(1)$.

target.current_view (somente leitura)

A visualização selecionada atualmente para o alvo renderizado. Isto é um objeto da *visualização do layout*.

target.view_index (leitura e escrita)

O índice base 1 da visualização selecionada para a renderização deste alvo.

target.visibility_mask (somente leitura)

Uma máscara de bits inteira indicando quais as coleções dos itens estão visíveis no momento da visualização atual.

target.screen_overlay (leitura e escrita)

Um booleano que indica se as sobreposições da tela estão ativadas.

target.zoom_to_screen (leitura e escrita)

Um booleano que indica se renderização do alvo está configurado para escalar fazendo com que as telas emuladas preencham toda a janela/tela o quanto for possível.

Contêiner do renderizador

Encapsula a classe `render_container`.

Instanciação**manager.machine.render.ui_container**

Obtém o contêiner de renderização usado para desenhar a interface do usuário, incluindo menus, controles deslizantes e as mensagens de pop-up.

manager.machine.render.targets[índice].ui_container

Obtém o contêiner de renderização usado para desenhar elementos da interface do usuário num determinado alvo de renderização.

manager.machine.screens[tag].container

Obtém o contêiner de renderização usado para desenhar uma determinada tela.

Métodos**container:draw_box(esquerda, cima, direita, baixo, [linha], [preenchimento])**

Desenha um retângulo delineado com bordas nas posições indicadas.

As coordenadas são números de ponto flutuante no intervalo entre 0 (zero) até 1 (um), com (0, 0) na parte superior esquerda e (1, 1) na parte inferior direita da janela ou da tela que mostra a interface do usuário. Observe que a relação de aspecto geralmente não é quadrada. As coordenadas são limitadas à área da janela ou da tela.

As cores de preenchimento e da linha estão no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa. Os valores dos canais devem ser empacotados em bytes de um inteiro com 32 bits sem assinatura na ordem alfa, vermelho, verde, azul do byte mais importante para o de menor importância. Caso a cor da linha não seja informada, é usada a cor do texto da interface; caso a cor de preenchimento não seja informada, é usada a cor de fundo da interface do usuário.

container:draw_line(x0, y0, x1, y1, [cor])

Desenha uma linha a partir de (x0, y0) até (x1, y1).

As coordenadas são números de ponto flutuante no intervalo entre 0 (zero) até 1 (um), com (0, 0) na parte superior esquerda e (1, 1) na parte inferior direita da janela ou da tela que mostra a interface do usuário. Observe que a relação de aspecto geralmente não é quadrada. As coordenadas são limitadas à área da janela ou da tela.

As coordenadas são números de ponto flutuante em unidades de pixels da tela emulada, com a origem em (0, 0). Observe que os pixels da tela emulada geralmente não são quadrados. O sistema de coordenadas é rotacionado caso a tela seja girada, o que geralmente é o caso para as telas no formato vertical. Antes da rotação, a origem está na parte superior esquerda e as coordenadas aumentam para a direita e para baixo. As coordenadas são limitadas à área da tela.

A cor da linha está no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) e 255 (opaco ou com intensidade total). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa. Os valores dos canais devem ser empacotados em bytes de um inteiro com 32 bits sem assinatura na ordem alfa, vermelho, verde, azul do byte mais importante para o de menor importância.. Caso a cor da linha não seja informada, é usada a cor do texto da interface do usuário.

container:draw_quad(textura, x0, y0, x1, y1, [cor])

Desenha um retângulo texturizado com o canto superior esquerdo em (x0, y0) e o canto inferior direito em (x1, y1). Caso uma cor seja especificada, os valores dos pixels da textura do canal ARGB são multiplicados pelos valores correspondentes da cor que foi especificada.

As coordenadas são números de ponto flutuante na faixa entre 0 (zero) até 1 (um) com (0, 0) na parte superior esquerda e (1, 1) na parte inferior direita da janela ou da tela que mostra a interface do usuário. Observe que a relação de aspecto geralmente não é quadrada. Caso o retângulo se estenda além dos limites do contêiner, este será recortado.

A cor está no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) e 255 (opaco ou com intensidade total). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa. Os valores dos canais devem ser empacotados em bytes de um inteiro com 32 bits sem assinatura na ordem alfa, vermelho, verde, azul do byte mais importante para o de menor importância..

container:draw_text(x|justificativa, y, texto, [primeiro plano], [plano de fundo])

Desenha uma linha na posição definida. Se a tela for racionada o texto também será.

Quando o primeiro argumento for um número, o texto será alinhado à esquerda nesta coordenada X. Quando o primeiro argumento for um texto, este deve ser "left", "center" ou "right" para que o texto seja desenhado e alinhado à esquerda/ao centro/à direita da janela ou da tela respectivamente. O segundo argumento define a coordenada Y da ascensão máxima do texto.

As coordenadas são números de ponto flutuante no intervalo entre 0 (zero) até 1 (um), com (0, 0) na parte superior esquerda e (1, 1) na parte inferior direita da janela ou da tela que mostra a interface do usuário. Observe que a relação de aspecto geralmente não é quadrada. As coordenadas são limitadas à área da janela ou da tela.

As cores do primeiro plano e do plano de fundo estão no formato alfa/vermelho/verde/azul (ARGB). Os valores dos canais estão no intervalo entre 0 (transparente ou desligado) até 255 (opaco ou com intensidade total). Os valores dos canais da cor não são previamente multiplicados pelo valor alpha. Os valores do canal devem ser empacotados em bytes com 32 bits inteiros não assinados pelo valor do canal alfa, na ordem alpha, vermelho, verde, azul a partir do byte mais importante até o byte com menor importância. Caso a cor do primeiro plano não seja informado, a cor do texto da interface será usada; caso a cor de fundo não seja informada, a cor do fundo da interface será usada.

Propriedades

container.user_settings (leitura e escrita)

A *configuração do usuário do contêiner*. Pode ser usado para controlar uma série de ajustes de imagem.

container.orientation (leitura e escrita)

Os sinalizadores de orientação do contêiner. Esta é uma máscara de bits inteira, onde o bit 0 (0x01) é definido para espelhar horizontalmente, o bit 1 (0x02) é definido para espelhar verticalmente e o bit 2 (0x04) é definido para espelhar ao longo do canto superior esquerdo inferior e a diagonal direita.

container.xscale (leitura e escrita)

O fator de escala X do contêiner. Este é um número de ponto flutuante.

container.yscale (leitura e escrita)

O fator de escala Y do contêiner. Este é um número de ponto flutuante.

container.xoffset (leitura e escrita)

O *offset* X do contêiner. Este é um número de ponto flutuante onde um (1) corresponde ao tamanho X do contêiner.

container.yoffset (leitura e escrita)

O *offset* Y do contêiner. Este é um número de ponto flutuante onde um (1) corresponde ao tamanho Y do contêiner.

container.is_empty (somente leitura)

Um booleano que indica se o contêiner não possui itens.

Configurações do contêiner do usuário

Encapsula a classe `render_container::user_settings` do MAME que representa os ajustes da imagem aplicados a um *contêiner do renderizador*.

Instanciação

manager.machine.screens[tag].container

Obtém a renderização atual do contêiner usado para desenhar uma determinada tela.

Propriedades

settings.orientation (leitura e escrita)

Os sinalizadores de orientação do contêiner. Esta é uma máscara de bits inteira, onde o bit 0 (0x01) é definido para espelhar horizontalmente, o bit 1 (0x02) é definido para espelhar verticalmente e o bit 2 (0x04) é definido para espelhar ao longo do canto superior esquerdo inferior e a diagonal direita.

settings.brightness (leitura e escrita)

O ajuste do brilho aplicado ao contêiner. Este é um número de ponto flutuante.

settings.contrast (leitura e escrita)

O ajuste do contraste aplicado ao contêiner. Este é um número de ponto flutuante.

settings.gamma (leitura e escrita)

O ajuste gama aplicado ao contêiner. Este é um número de ponto flutuante.

settings.xscale (leitura e escrita)

O fator de escala X do contêiner. Este é um número de ponto flutuante.

settings.yscale (leitura e escrita)

O fator de escala Y do contêiner. Este é um número de ponto flutuante.

settings.xoffset (leitura e escrita)

O *offset* X do contêiner. Este é um número de ponto flutuante onde um (1) representa o tamanho X do contêiner.

settings.yoffset (leitura e escrita)

O *offset* Y do contêiner. Este é um número de ponto flutuante onde um (1) representa o tamanho Y do contêiner.

Arquivo layout

Encapsula a classe `layout_file` do MAME, faz a representação das visualizações carregadas a partir de um *arquivo layout* que pode ser utilizado para uma renderização final.

Instanciação

Um objeto do arquivo layout é fornecido ao seu *script* layout na variável `file`. Os objetos do arquivo layout não são instanciados diretamente a partir dos *scripts* Lua.

Métodos

`layout:set_resolve_tags_callback(cb)`

Define uma função para realizar tarefas adicionais depois que o sistema emulado tenha finalizado a sua inicialização, quando as tags nas visualizações do layout tenham sido resolvidas e os manipuladores dos itens da visualização principal tenham sido configurados. A função não deve aceitar nenhum argumento.

Use com `nil` para remover o retorno de chamada.

Propriedades

`layout.device` (somente leitura)

O dispositivo que fez com que o arquivo layout fosse carregado. Normalmente o dispositivo raiz do sistema no caso dos layouts externos.

`layout.views[]` (somente leitura)

As *visualizações do layout* criados a partir do arquivo layout. As visualizações são indexadas por nomes não qualificados (ou seja, o valor do atributo `name`). As visualizações são ordenadas como aparecem no arquivo de layout ao iterar ou usar o método `at`. Os métodos do índice obtêm `at` e `index_of` com complexidade $O(n)$.

Observe que nem todas as visualizações no arquivo XML podem ser criadas. Por exemplo, as visualizações não são criadas se a referência das telas forem fornecidas pelos dispositivos do cartão do slot caso o os referidos dispositivos do cartão do slot não estiverem presentes no sistema.

Visualização do layout

Encapsula a classe `layout_view` do MAME que representa uma visualização que pode ser renderizada num determinado alvo. As visualizações são criadas a partir dos arquivos layout, podem ser carregados a partir da arte externa, interna do MAME ou gerada automaticamente com base nas telas do sistema que está sendo emulado.

Instanciação

`manager.machine.render.targets[índice].current_view`

Obtém a visualização selecionada atualmente para renderizar um determinado alvo.

`file.views[nome]`

Obtém a visualização de um determinado nome definido a partir de um *arquivo de layout*. De maneira geral, é assim que um *script* de layout obtém as visualizações.

Métodos

view:has_screen(tela)

Retorna um booleano indicando se a tela está presente na visualização. Isso é verdadeiro para telas que estão presentes, mas não visíveis porque o usuário ocultou a coleção dos itens que pertencem à ela.

view:set_prepare_items_callback(cb)

Define uma função para realizar tarefas adicionais antes que os itens da visualização sejam adicionados na renderização do alvo em preparação para o desenho de um quadro de vídeo. A função não deve aceitar quaisquer argumentos. Use com `nil` para remover o retorno de chamada.

view:set_preload_callback(cb)

Define uma função para realizar tarefas adicionais após pré-carregar a visualização dos itens visíveis. A função não deve aceitar quaisquer argumentos. Use com `nil` para remover o retorno de chamada. Esta função pode ser invocada quando o usuário seleciona uma visualização ou torna a visualização do item de uma coleção visível. Ele pode ser invocado várias vezes para obter uma exibição, portanto, evite repetir tarefas dispendiosas.

view:set_recomputed_callback(cb)

Define uma função para realizar tarefas adicionais depois que as dimensões da visualizações tenham sido recomputadas. A função não deve aceitar quaisquer argumentos. Use com `nil` para remover o retorno de chamada.

As coordenadas da visualização são recalculadas em vários eventos, incluindo a janela que estiver sendo redimensionada, entrando ou saindo do modo de tela inteira e alterando a configuração de zoom para região da tela.

view:set_pointer_updated_callback(cb)

Define uma função para receber notificações quando um ponteiro ingressar, se mover ou alterar os estados do botão na exibição. A função deve aceitar nove argumentos:

- O tipo do ponteiro (`mouse`, `pen`, `touch` ou `unknown`).
- A ID do ponteiro (um inteiro não negativo que não se alterará durante a vida do ponteiro).
- A ID do dispositivo para grupos de ponteiros para reconhecer multitoque. (inteiro não negativo).
- A posição horizontal nas coordenadas do layout.
- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões pressionados no momento.
- Uma máscara de bits que representa os botões que foram pressionados nesta atualização.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização.
- A contagem de cliques (positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)).

Invoque com `nil` para remover o retorno de chamada.

view:set_pointer_left_callback(cb)

Define uma função para receber notificações quando um ponteiro deixar de ser visível normalmente. A função deve aceitar nove argumentos:

- O tipo do ponteiro (mouse, pen, touch ou unknown).
- A ID do ponteiro (um inteiro não negativo que não se alterará durante a vida do ponteiro). A ID pode ser reutilizada para um novo ponteiro após receber esta notificação.
- A ID do dispositivo para grupos de ponteiros para reconhecer multitoque. (inteiro não negativo).
- A posição horizontal nas coordenadas do layout.
- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização.
- A contagem de cliques (positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)).

Invoque com nil para remover o retorno de chamada.

view:set_pointer_aborted_callback(cb)

Define uma função para receber notificações quando o ponteiro deixar de estar visível de maneira anormal. A função deve aceitar nove argumentos:

- O tipo do ponteiro (mouse, pen, touch ou unknown).
- A ID do ponteiro (um inteiro não negativo que não se alterará durante a vida do ponteiro). A ID pode ser reutilizada para um novo ponteiro após receber esta notificação.
- A ID do dispositivo para grupos de ponteiros para reconhecer multitoque. (inteiro não negativo).
- A posição horizontal nas coordenadas do layout.
- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização.
- A contagem de cliques (positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)).

Invoque com nil para remover o retorno de chamada.

view:set_forget_pointers_callback(cb)

Define uma função para receber notificações quando a visualização deve parar de processar a entrada do ponteiro. Invoque com nil para remover o retorno de chamada.

Isso pode ocorrer em várias situações, por exemplo, quando ocorre a alteração da configuração da visualização ou a questão de um menu assumir o controle da entrada.

Propriedades

view.items[] (somente leitura)

O elemento do layout e da tela de *visualização do item do layout* numa visualização. Este contêiner não suporta iteração por chave usando pairs; só é compatível a iteração através do índice usando ipairs. A chave é o valor do atributo id, caso esteja presente. Apenas itens com atributos id podem ser pesquisados através das chaves. O método index get tem complexidade $O(1)$ e os métodos at e o index_of têm complexidade $O(n)$.

view.name (somente leitura)

Exibe o nome da visualização. Isso pode ser qualificado para indicar o dispositivo que causou o carregamento do *arquivo layout*, quando não for o dispositivo raiz do sistema.

view.unqualified_name (somente leitura)

O nome não qualificado da visualização, exatamente como aparece no atributo name no arquivo layout.

view.visible_screen_count (somente leitura)

A quantidade dos itens nas telas que estão atualmente ativados na visualização.

view.effective_aspect (somente leitura)

A proporção efetiva entre a largura e a altura da visualização com a sua configuração atual.

view.bounds (somente leitura)

O *limites do renderizador* do objeto que representa os limites efetivos da visualização na sua configuração atual. As coordenadas estão em unidades de visualização, que são arbitrárias, porém assumidas como tendo uma proporção quadrada.

view.has_art (somente leitura)

Um booleano que indica se a visualização possui itens que não são da tela, incluindo itens que não são visíveis porque o usuário ocultou a coleção dos itens aos quais elas pertencem.

view.show_pointers (leitura e escrita)

Um booleano que define se os ponteiros do mouse e da caneta devem ser exibidos na visualização.

view.hide_inactive_pointers (leitura e escrita)

Um booleano que define se os ponteiros do mouse devem ser ocultados na exibição após um período de inatividade.

Visualização do item do layout

Encapsula a classe `layout_view_item` do MAME que representa um item numa *visualização do layout*. Um item é desenhado como uma superfície retangular texturizada. A textura é fornecida por uma tela emulada ou um elemento do layout. Observe que as chamadas de retorno dos itens de visualização do layout não são executadas como corrotinas.

Instanciação**layout.views[name].items[id]**

Obtém um item da visualização através do ID. O item deve ter um atributo `id` no arquivo layout para que possa ser pesquisado através do ID.

Métodos**item:set_state(state)**

Define o valor usado como o estado do elemento e o estado da animação na ausência dos vínculos. O argumento deve ser um número inteiro.

item:set_element_state_callback(cb)

Define uma função a ser invocada para obter o estado do elemento para o item. A função não deve aceitar quaisquer argumentos e deve retornar um número inteiro. Use com `nil` para restaurar o estado original do retorno de chamada do elemento (com base nos vínculos do arquivo layout).

Observe que a função não deve acessar a propriedade `element_state` do item, pois isso resultará numa repetição infinita. Este retorno de chamada não será usado para obter o estado de animação para o item, mesmo se o item não tiver vínculos explícitos do estado de animação no arquivo layout.

item:set_animation_state_callback(cb)

Define uma função que será invocada para obter o estado de animação do item. A função não deve aceitar quaisquer argumentos e deve retornar um número inteiro. Use com `nil` para restaurar o estado de animação original do retorno de chamada (com base nos vínculos do arquivo layout).

Observe que a função não deve acessar a propriedade `animation_state` do item, pois isso resultará numa repetição infinita.

item:set_bounds_callback(cb)

Define uma função que será invocada para obter os limites do item. A função não deve aceitar qualquer argumento e deve retornar um *limites do renderizador* do objeto nas coordenadas do alvo renderizado. Use com `nil` para restaurar o estado do limite original do retorno de chamada (com base no estado da animação do item e nos elementos `bounds` herdados a partir do arquivo layout).

Observe que a função não deve acessar a propriedade `bounds` do item, pois isso resultará numa repetição infinita.

item:set_color_callback(cb)

Defina uma função que será invocada para obter a cor do multiplicador para o item. A função não deve aceitar qualquer argumento e deve retornar um objeto *renderização da cor*. Use com `nil` para restaurar a cor original do retorno de chamada (com base no estado da animação do item e dos elementos `color` herdados a partir do arquivo layout).

Observe que a função não deve acessar a propriedade `color` do item, pois isso resultará numa repetição infinita.

item:set_scroll_size_x_callback(cb)

Define uma função que será invocada para obter o tamanho da janela de rolagem horizontal como uma proporção da largura do elemento associado. A função não deve aceitar nenhum argumento e retornar um valor de ponto flutuante. Use com `nil` para restaurar o tamanho padrão da rolagem horizontal da janela (com base no elemento `xscroll` relacionado no arquivo de layout XML).

Observe que a função não deve acessar a propriedade `scroll_size_x` do item, pois isto resultará numa repetição infinita.

item:set_scroll_size_y_callback(cb)

Define uma função que será invocada para obter o tamanho da janela de rolagem vertical como uma proporção da altura do elemento associado. A função não deve aceitar nenhum argumento e retornar um valor de ponto flutuante. Use com `nil` para restaurar o tamanho padrão da rolagem horizontal da janela (com base no elemento `yscroll` relacionado no arquivo de layout XML).

Observe que a função não deve acessar a propriedade `scroll_size_y` do item, pois isto resultará numa repetição infinita.

item:set_scroll_pos_x_callback(cb)

Define uma função que será invocada para obter a posição da rolagem horizontal. Um valor zero coloca a janela de rolagem horizontal na borda esquerda do elemento associado. Se o item não se enrola horizontalmente, um valor `1.0` posiciona a janela de rolagem horizontal na borda direita do elemento associado; se o item se enrola horizontalmente, um valor `1.0` corresponde ao enrolamento de retorno à borda esquerda do elemento associado. A função não deve aceitar nenhum argumento e deve retornar um valor de ponto flutuante. Use com `nil` para restaurar a posição padrão da rolagem horizontal (com base nas ligações no elemento relativo `xscroll` no arquivo de layout XML).

Observe que a função não deve acessar a propriedade `scroll_pos_x` do item, pois isto resultará numa repetição infinita.

item.set_scroll_pos_y_callback(cb)

Define uma função que será invocada para obter a posição da rolagem vertical. Um valor zero coloca a janela de rolagem vertical na borda superior do elemento associado. Se o item não se enrola verticalmente, um valor `1.0` posiciona a janela de rolagem vertical na borda de baixo do elemento associado; se o item se enrola verticalmente, um valor `1.0` corresponde ao enrolamento de retorno à borda esquerda do elemento associado. A função não deve aceitar nenhum argumento e deve retornar um valor de ponto flutuante. Use com `nil` para restaurar a posição padrão da rolagem horizontal (com base nas ligações no elemento relativo `yscroll` no arquivo de layout XML).

Observe que a função não deve acessar a propriedade `scroll_pos_y` do item, pois isto resultará numa repetição infinita.

Propriedades

item.id (somente leitura)

Obtenha o identificador opcional do item. Este é o valor do atributo `id` no arquivo layout, caso esteja presente ou `nil`.

item.bounds_animated (somente leitura)

Um booleano que indica se os limites do item dependem de seu estado de animação.

item.color_animated (somente leitura)

Um booleano que indica se a cor do item depende de seu estado de animação.

item.bounds (somente leitura)

Os limites do item para o estado atual. Este é um *limitador do renderizador* do objeto nas coordenadas do alvo renderizado.

item.color (somente leitura)

A cor do item para o estado atual. A cor da tela ou da textura do elemento é multiplicada por esta cor. Este faz a *renderização da cor* do objeto.

item.scroll_wrap_x (somente leitura)

Um booleano indicando se o item se enrola horizontalmente.

item.scroll_wrap_y (somente leitura)

Um booleano indicando se o item se enrola verticalmente.

item.scroll_size_x (leitura e escrita)

Obtém o tamanho da janela de rolagem horizontal do item para a condição atual, ou configure o tamanho da janela de rolagem horizontal para usar na ausência de ligações. Este é um valor de ponto flutuante que representa uma proporção da largura do elemento associado.

item.scroll_size_y (leitura e escrita)

Obtém o tamanho da janela de rolagem vertical do item para a condição atual, ou configure o tamanho da janela de rolagem vertical para usar na ausência de ligações. Este é um valor de ponto flutuante que representa uma proporção da largura do elemento associado.

item.scroll_pos_x (leitura e escrita)

Obtém a posição de rolagem horizontal do item para a condição atual, ou defina o tamanho da posição de rolagem horizontal que deve ser usada na ausência das ligações. Este é um valor de ponto flutuante.

item.scroll_pos_y (leitura e escrita)

Obtém a posição de rolagem vertical do item para a condição atual, ou defina o tamanho da posição de rolagem vertical que deve ser usada na ausência das ligações. Este é um valor de ponto flutuante.

item.blend_mode (somente leitura)

Obtém modo de mesclagem do item. Este é um valor inteiro, onde 0 significa sem mesclagem, 1 significa mesclagem alfa, 2 significa multiplicação por RGB, 3 significa mesclagem aditiva e -1 permite que os itens dentro de um contêiner determinem os seus próprios modos de mesclagem.

item.orientation (somente leitura)

Obtém os sinalizadores da orientação do item. Esta é uma máscara de bits inteira onde o bit 0 (0x01) é definido para espelhar horizontalmente, o bit 1 (0x02) é definido para espelhar verticalmente e o bit 2 (0x04) é definido para espelhar ao longo da diagonal superior esquerda e inferior direita.

item.element_state (somente leitura)

Obtenha o estado atual do elemento. Isso invocará a função retorno de chamada do estado do elemento para lidar com os vínculos.

item.animation_state (somente leitura)

Obtém o estado atual da animação. Isso invocará a função retorno de chamada do estado de animação do elemento para lidar com os vínculos.

9.3.7 Classes de depuração Lua

Alguns dos principais recursos de depuração do MAME podem ser controlados a partir de um *script* Lua. O depurador deve estar ativado para que seja possível usar os recursos de depuração (normalmente usando `-debug` na linha de comando).

- *Tabela de símbolos*
- *A interpretação das expressões*
- *Acesso do símbolo*
- *Gerenciador do depurador*
- *Interface de depuração do dispositivo*
- *Breakpoint*
- *Watchpoint*
- *Erro da expressão*

Tabela de símbolos

Encapsula a classe `symbol_table` do MAME, fornece o nome dos símbolos que pode ser usado nas expressões. Note que a tabela de símbolos pode ser criada e até mesmo utilizada quando o depurador não estiver ativado.

Instanciação

`emu.symbol_table(sistema)`

Cria uma nova tabela de símbolos dentro do contexto de uma determinado sistema,

`emu.symbol_table(parent, [dispositivo])`

Cria uma nova tabela de símbolos dentro de uma determinada tabela relacionada dos símbolos. No caso de um dispositivo ser definido e implemente um `device_memory_interface`, este será utilizado como a base para a busca dos endereços e das regiões da memória. Note que caso um dispositivo que não implemente um `device_memory_interface` seja fornecido, este não será utilizado (os endereços e a região da memória serão buscadas com relação à raiz do dispositivo).

`emu.symbol_table(dispositivo)`

Cria uma nova tabela de símbolos dentro do contexto de um determinado dispositivo. Caso um dispositivo implemente o `device_memory_interface`, este será utilizado como a base para a busca dos endereços e das regiões da memória. Note que caso um dispositivo que não implemente um `device_memory_interface` seja fornecido, este não será utilizado (os endereços e a região da memória serão buscadas com relação à raiz do dispositivo).

Métodos

symbols:set_memory_modified_func(cb)

Define uma função que será invocada quando a memória for alterada através da tabela de símbolos. Nenhum argumento é passado para a função e quaisquer valores retornados são ignorados. Invoque com `nil` para eliminar o *callback*.

symbols:add(nome, [valor])

Denomina um símbolo inteiro seu nome deve ser uma *string*. Ao informar um valor, este deve ser um número inteiro. Durante o fornecimento do valor, um símbolo só de leitura é adicionado em conjunto com o valor informado. Na ausência deste valor, é criado um símbolo de leitura/escrita com valor zero. O símbolo será substituído na existência de um símbolo com um determinado nome já existente na tabela de símbolos..

Retorna o [acesso do símbolo](#).

symbols:add(nome, getter, [setter], [formato])

Denomina um símbolo inteiro usando a chamada *getter* e a chamada opcional *setter*. Onde seu nome deve ser uma *string*. A chamada *getter* deve uma função que retorne um valor inteiro para o valor do símbolo. Caso seja informado, o *setter* deve ser uma função que aceite um único valor inteiro para o novo valor do símbolo. Um formato *string* para exibir o valor do símbolo pode ser usado de forma opcional. O símbolo será substituído na existência de um símbolo com um determinado nome já existente na tabela de símbolos..

Retorna o [acesso do símbolo](#).

symbols:add(nome, minparams, maxparams, execute)

Denomina a função de um símbolo. Onde seu nome deve ser uma *string*. O valor mínimo e máximo da quantidade dos valores devem ser inteiros. O símbolo será substituído na existência de um símbolo com um determinado nome já existente na tabela de símbolos..

Retorna o [acesso do símbolo](#).

symbols:find(nome)

Retorna o [acesso do símbolo](#) com um determinado nome ou `nil` caso não haja um símbolo com um determinado nome na tabela de símbolos.

symbols:find_deep(nome)

Retorna o [acesso do símbolo](#) com um determinado nome ou `nil` caso não haja um símbolo com um determinado nome na tabela de símbolos ou em quaisquer outra tabela de símbolos relacionada.

symbols:value(nome)

Retorna o valor inteiro do símbolo com um determinado nome ou zero caso não haja um símbolo com um determinado nome na tabela de símbolos ou em quaisquer outra tabela de símbolos relacionada. Gera um erro caso o símbolo com um determinado nome seja uma função de um determinado símbolo.

symbols:set_value(nome, valor)

Define o valor de um símbolo com um determinado nome. Gera um erro caso o símbolo com um determinado nome seja um símbolo inteiro, somente leitura, ou caso seja uma função de um determinado símbolo. Não há qualquer efeito caso não exista um símbolo com determinado nome na tabela ou em quaisquer tabelas de símbolos relacionados.

symbols:memory_value(nome, espaço, offset, tamanho, disable_se)

Lê o valor a partir da memória. Oferece um nome ou uma etiqueta no espaço de endereçamento, na região da memória onde a leitura será feita ou nil para usar o espaço de endereçamento ou a região da memória imposta por space. Consulte [acesso à memória](#) para conhecer os tipos de acesso e as definições que podem ser utilizadas com space. O tamanho do acesso pode ter 1, 2, 4 ou 8 bytes. O argumento disable_se determina se os efeitos colaterais do acesso à memória deve ser desativado ou não.

symbols:set_memory_value(nome, espaço, offset, valor, tamanho, disable_se)

Escreve um valor na memória. Oferece um nome ou uma etiqueta no espaço de endereçamento, na região da memória onde a leitura será feita ou nil para usar o espaço de endereçamento ou a região da memória imposta por space. Consulte [acesso à memória](#) para conhecer os tipos de acesso e as definições que podem ser utilizadas com space. O tamanho do acesso pode ter 1, 2, 4 ou 8 bytes. O argumento disable_se determina se os efeitos colaterais do acesso à memória deve ser desativado ou não.

symbols:read_memory(espaço, endereço, tamanho, apply_translation)

Lê um valor a partir de um espaço de endereçamento. O tamanho do acesso pode ter 1, 2, 4 ou 8 bytes. Caso o argumento apply_translation seja verdadeiro, o endereço será traduzido com a intensão de leitura do depurador ou retornará o valor do tamanho requisitado com todos os bits definidos caso a tradução do endereço falhe.

symbols:write_memory(espaço, endereço, dado, tamanho, apply_translation)

Escreve um valor num determinado espaço de endereçamento. O tamanho do acesso pode ter 1, 2, 4 ou 8 bytes. Caso o argumento apply_translation seja verdadeiro, o endereço será traduzido com a intensão de escrita do depurador. A função de alteração da tabela dos símbolos da memória será invocada depois que o valor for escrito. caso a tradução do endereço falhe, o valor não será escrito e a função de alteração da tabela dos símbolos da memória não será invocada.

Propriedades

symbols.entries[]

O [acesso do símbolo](#) na tabela de símbolos indexada por nomes. Os métodos at e index_of têm complexidade O(n). Todas as outras operações possuem complexidade O(1).

symbols.parent (somente leitura)

A tabela de símbolos relacionados ou nil caso não haja nada relacionado na tabela de símbolos.

A interpretação das expressões

Encapsula a classe parsed_expression, representa uma expressão simbólica do depurador. Repare que as expressões que forem interpretadas podem ser criadas e até mesmo utilizadas quando o depurador não estiver ativado.

Instanciação

emu.parsed_expression(símbolo)

Cria uma expressão vazia que usará a *tabela de símbolos* para fazer uma procura pelos símbolos.

emu.parsed_expression(símbolos, string, [default_base])

Cria uma expressão ao analisar uma *string*, também é usada um *tabela de símbolos* para fazer uma procura pelos símbolos. Na ausência de uma base padrão para fazer a busca literal dos inteiros, o hexadecimal 16 é usado. Gera um erro caso a *string* tenha *erros de sintaxe* ou utilize símbolos não definidos.

Métodos

expression:set_default_base(base)

Define a base padrão para a interpretação dos literais numéricos. A base deve ser um valor inteiro e positivo.

expression:parse(string)

Analisar uma cadeia de expressões de depuração. Faz a substituição do conteúdo atual da expressão caso não esteja vazia. Gera um erro caso a *string* tenha *erros de sintaxe* ou utilize símbolos não definidos. O conteúdo da expressão anterior não é preservada ao tentar analisar uma cadeia de expressões inválidas.

expression:execute()

Faz a avaliação da expressão, como resultado, retorna um inteiro não assinado. Gera um erro caso haja um *erros de sintaxe* e a expressão não puder ser analisada (ao invocar a função com uma quantidade de argumentos inválidos por exemplo).

Propriedades

expression.is_empty (somente leitura)

Um booleano que indica se a expressão possui *tokens*.

expression.original_string (somente leitura)

A *string* original que foi analisada para criar uma expressão.

expression.symbols (leitura e escrita)

A *tabela de símbolos* usada para procurar pelos símbolos numa expressão.

Acesso do símbolo

Envelopa a classe `symbol_entry` do MAME, faz a representação do acesso numa *tabela de símbolos*. Observe que as entradas dos símbolos não devem ser usados depois que a tabela dos símbolos a que pertencem seja destruída.

Instanciação

symbols:add(nome, [valor])

Adiciona um símbolo inteiro à *tabela de símbolos*, retornando um novo símbolo para o acesso.

symbols:add(nome, getter, [setter], [formato])

Adiciona um símbolo inteiro à *tabela de símbolos*, retornando um novo símbolo para o acesso.

symbols:add(nome, minparams, maxparams, execute)

Adiciona uma nova função à *tabela de símbolos*, retornando um novo símbolo para o acesso.

Propriedades

entry.name (somente leitura)

O nome do acesso do símbolo.

entry.format (somente leitura)

O formato da *string* usada para converter o acesso do símbolo para ser exibido como texto na tela.

entry.is_function (somente leitura)

Um booleano indicando se o acesso do símbolo é uma função que pode ser invocada.

entry.is_lval (somente leitura)

Um booleano indicando se o acesso do símbolo é um símbolo inteiro que pode ser definido (se ele pode ser usado no lado esquerdo das expressões da atribuição por exemplo).

entry.value (leitura e escrita)

Um valor inteiro do acesso do símbolo. Gera um erro caso tente definir um valor caso o acesso do símbolo seja de apenas leitura, assim como, ao tentar obter ou definir o valor da função de um símbolo.

Gerenciador do depurador

Encapsula a classe `debugger_manager` do MAME que fornece a interface principal para controlar o depurador.

Instanciação

manager.machine:debugger()

Retorna a instância do gerenciador de depuração global ou `nil` se o depurador não estiver ativado.

Métodos

debugger:command(str)

Execute um comando no console do depurador. O argumento é a *string* do comando. A saída é enviada ao console do depurador e ao console Lua.

Propriedades

debugger.consolelog[] (somente leitura)

As linhas no log do console (saída dos comandos do depurador). Este contêiner suporta apenas o comprimento das operações do índice.

debugger.errorlog[] (somente leitura)

As linhas no registro log de erros (saída logerror). Este contêiner suporta apenas o comprimento das operações do índice.

debugger.visible_cpu (leitura e escrita)

O dispositivo CPU com foco no depurador. As alterações se tornam visíveis no console do depurador depois da próxima etapa. Não há efeito quando configurando para um dispositivo que não seja uma CPU.

debugger.execution_state (leitura e escrita)

Tanto "run" se o sistema que estiver sendo emulado estiver rodando, ou "stop" caso esteja parado no depurador.

Interface de depuração do dispositivo

Encapsula a classe `device_debug` do MAME que fornece a interface do depurador para um dispositivo emulado da CPU.

Instanciação

manager.machine.devices[tag].debug

Retorna a interface do depurador para um dispositivo emulado da CPU ou `nil` caso o dispositivo não seja uma CPU.

Métodos

debug:step([cnt])

Avance através da quantidade especificada das instruções. Caso a contagem das instruções não seja informada, o padrão é uma única instrução.

debug:go()

Execute a CPU que estiver sendo emulada.

debug:bpset(addr, [cond], [act])

Define um breakpoint no endereço informado com uma condição e ações opcionais. Caso a ação não seja informada, o padrão é apenas parar no depurador. Retorna o número do breakpoint para o novo breakpoint.

Caso seja especificada, a condição deve ser uma expressão do depurador que será avaliada sempre que o breakpoint for atingido. A execução só será interrompida se a expressão for avaliada como um valor diferente de zero. Caso a condição não seja informada, o padrão é sempre estar ativo.

debug:bpnable([bp])

Ative o breakpoint informado ou todos os breakpoints do dispositivo caso nenhum número do breakpoint seja informado. Retorna se o número informado correspondeu a um breakpoint caso o número do breakpoint seja informado ou nil caso nenhum seja informado.

debug:bpdisable([bp])

Desative o breakpoint informado ou todos os breakpoints do dispositivo caso nenhum número seja informado. Retorna se o número informado correspondeu a um breakpoint caso o número do breakpoint seja informado ou nil caso nenhum seja informado.

debug:bpclear([bp])

Limpe o breakpoint informado ou todos caso se nenhum número seja informado. Retorna se o número informado correspondeu a um breakpoint caso o número do breakpoint seja informado ou nil caso nenhum seja informado.

debug:bpelist()

Retorna uma tabela dos breakpoints para o dispositivo. As chaves são os números dos breakpoints e os valores são os *breakpoints*.

debug:wpset(espaco, tipo, endereco, comprimento, [condição], [act])

Define um watchpoint sobre o intervalo dos endereços informados com uma ação e condição opcionais. O tipo deve ser "r", "w" ou "rw" para a leitura, a escrita ou a leitura e a escrita do breakpoint. Caso a ação não seja informada, o padrão é apenas parar no depurador. Retorna o número do watchpoint para o novo watchpoint.

Caso seja especificada, a condição deve ser uma expressão do depurador que será avaliada sempre que o breakpoint for atingido. A execução só será interrompida caso a expressão seja avaliada como um valor diferente de zero. A variável wpaddr é definida para o atual endereço que acionou o watchpoint, a variável wpdata é definida para o dado que está sendo lido ou gravado, a variável wpsize é definida para o tamanho do dado em bytes.

Caso a condição não seja definida, ela sempre estará ativa.

debug:wpenable([wp])

Ative o watchpoint informado ou todos os watchpoints do dispositivo caso nenhum número seja informado. Retorna se o número informado correspondeu a watchpoint caso um número seja informado ou nil caso nenhum seja.

debug:wpdisable([wp])

Desative o watchpoint informado ou todos os watchpoints do dispositivo caso nenhum número seja informado. Retorna se o número informado correspondeu a watchpoint caso um número seja informado ou nil caso nenhum seja.

debug:wpclear([wp])

Limpe o watchpoint informado ou todos os watchpoints do dispositivo caso nenhum número seja informado. Retorna se o número informado correspondeu a watchpoint caso um número seja informado ou nil caso nenhum seja.

debug:wplist(espaco)

Retorna uma tabela com os watchpoints para o espaço de endereço informado para o dispositivo. As chaves são os números watchpoints e os valores são os *watchpoints*.

Breakpoint

Encapsula a classe debug_breakpoint do MAME que representa um ponto de interrupção (breakpoint) para um dispositivo emulado da CPU.

Instanciação**manager.machine.devices[tag].debug:bplist()[bp]**

Obtém o breakpoint informado para um dispositivo emulada da CPU ou nil caso nenhum breakpoint corresponda ao índice informado.

Propriedades**breakpoint.index** (somente leitura)

O índice do breakpoint. Pode ser usado para ativar, desativar ou limpar o breakpoint através da *interface de depuração do dispositivo*.

breakpoint.enabled (leitura e escrita)

Um booleano que indica se o breakpoint no momento está ativo.

breakpoint.address (somente leitura)

O endereço do breakpoint.

breakpoint.condition (somente leitura)

Uma expressão do depurador avaliada cada vez que o breakpoint for atingido. A ação só será disparada caso esta expressão seja avaliada como um valor diferente de zero. Uma *string* vazia caso nenhuma condição seja informada.

breakpoint.action (somente leitura)

Uma ação que o depurador executará quando o breakpoint for atingido e a condição for avaliada como um valor diferente de zero. Uma *string* vazia caso nenhuma ação seja informada.

Watchpoint

Encapsula a classe `debug_watchpoint` do MAME que representa um ponto de controle (watchpoint) para um dispositivo emulado da CPU.

Instanciação

`manager.machine.devices[tag].debug:wplist(espaco)[wp]`

Obtém o watchpoint informado para um watchpoint de um dispositivo emulado da CPU ou `nil` caso nenhum watchpoint no espaço de endereço corresponda ao índice informado.

Propriedades

`watchpoint.index` (somente leitura)

O índice do watchpoint. Pode ser usado para ativar, desativar ou limpar o watchpoint através da *interface de depuração do dispositivo*.

`watchpoint.enabled` (leitura e escrita)

Um booleano que indica se o watchpoint no momento está ativo.

`watchpoint.type` (somente leitura)

O tipo deve ser "r", "w" ou "rw" para a leitura, a escrita ou a leitura e a escrita do watchpoint.

`watchpoint.address` (somente leitura)

O endereço inicial do intervalo dos endereços do watchpoint.

`watchpoint.length` (somente leitura)

O comprimento do intervalo dos endereços do watchpoint.

`watchpoint.condition` (somente leitura)

Uma expressão do depurador avaliada cada vez que o watchpoint for atingido. A ação só será disparada caso esta expressão seja avaliada como um valor diferente de zero. Uma *string* vazia caso nenhuma condição seja informada.

`watchpoint.action` (somente leitura)

Uma ação que o depurador executará quando o watchpoint for atingido e a condição for avaliada como um valor diferente de zero. Uma *string* vazia caso nenhuma ação seja informada.

Erro da expressão

Encapsula a classe `expression_error` do MAME que descreve um erro que esteja acontecendo enquanto interpretando ou executando expressões do depurador. Gera um erro caso durante a *interpretação das expressões*. Pode ser convertida para texto para fornecer uma descrição do erro.

Propriedades

err.code (somente leitura)

Um número dependente da implementação que representa a categoria de erro. Não deve ser exibido para o usuário.

err.offset (somente leitura)

O deslocamento da string de expressão onde o erro foi encontrado.

9.4 Tutorial do console interativo Lua

No prompt de comando ou terminal rode um jogo arcade qualquer com as opções `-console` e `-window`

First run an arcade game in MAME at the command prompt with the `-console` and `-window` para ativar o console Lua:

[illegible]

Neste ponto, o seu jogo provavelmente pode estar sendo executado, use o comando abaixo para pausá-lo:

```
[MAME]> emu.pause()
[MAME]>
```

Mesmo sem qualquer tipo de retorno no console, é possível notar que o jogo pausou. Em geral, os comandos não retornam informação de confirmação e o terminal retorna apenas mensagens de erro.

Durante a execução, é possível verificar qual a sua versão do MAME com o comando abaixo:

```
[MAME]> print(emu.app_name() .. " " .. emu.app_version())
name 0.255
```

Nós agora começaremos a explorar os métodos relacionadas à tela. Primeiro, vamos enumerar as *telas disponíveis* no sistema:

```
[MAME]> for tag, screen in pairs(manager.machine.screens) do print(tag) end
:screen
```

O `manager.machine` é o objeto da *maquina em execução* da sessão atual que está sendo emulada. Será utilizada com bastante frequência. Os `screens` são *enumeradores de dispositivos* que geram todas as telas do sistema; a maioria dos sistemas arcade só tem uma tela principal. No nosso caso a única tela principal é marcada como `:screen` e podemos inspecioná-la mais a fundo:

```
[MAME]> -- mantendo a referência para a tela principal numa variável
[MAME]> s = manager.machine.screens[':screen']
[MAME]> print(s.width .. 'x' .. s.height)
320x224
```

Temos diferentes métodos para desenhar um painel (HUD) na tela composta de camadas com linhas, caixas e textos:

```
[MAME]> -- definimos a função para desenhar a interface e a chamamos
[MAME]> function draw_overlay()
[MAME]>> s:draw_text(40, 40, 'foo') -- (x0, y0, msg)
[MAME]>> s:draw_box(20, 20, 80, 80, 0xff00ffff, 0) -- (x0, y0, x1, y1, line-color, fill-
--color)
[MAME]>> s:draw_line(20, 20, 80, 80, 0xff00ffff) -- (x0, y0, x1, y1, line-color)
[MAME]>> end
[MAME]> draw_overlay()
```

Isso desenha alguns desenhos inúteis na tela. No entanto, seu painel desaparecerá caso não seja atualizado ao sair da pausa. Para evitar isso, é preciso registrar uma função que será chamado a cada atualização do vídeo:

```
[MAME]> emu.register_frame_done(draw_overlay, 'frame')
```

Todas as cores são no formato ARGB (8 bit por canal), enquanto a origem da tela geralmente corresponde ao canto superior esquerdo da tela (0,0).

Da mesma forma para telas, é possível inspecionar todos os dispositivos conectados num sistema:

```
[MAME]> for tag, device in pairs(manager.machine.devices) do print(tag) end
:audiocpu
:maincpu
:saveram
:screen
:palette
[...]
```

Em alguns casos, também é possível inspecionar e manipular a memória e o estado:

```
[MAME]> cpu = manager.machine.devices[':maincpu']
[MAME]> -- enumera, lê e escreve registros de estado
[MAME]> for k, v in pairs(cpu.state) do print(k) end
CURPC
rPC
IR
CURFLAGS
SSR
D0
[...]
[MAME]> print(cpu.state['D0'].value)
303
[MAME]> cpu.state['D0'].value = 255
[MAME]> print(cpu.state['D0'].value)
255
```

```
[MAME]> -- inspeciona a memória
[MAME]> for name, space in pairs(cpu.spaces) do print(name) end
program
cpu_space
[MAME]> mem = cpu.spaces['program']
```

(continua na próxima página)

(continuação da página anterior)

```
[MAME]> print(mem:read_i8(0xc000))  
41
```

Observe que muitos objetos suportam o preenchimento automático ao pressionar a tecla Tab durante a escrita parcial de símbolos ou da propriedade de um nome:

```
[MAME]> print(mem:read_<TAB>  
read_direct_i8  
read_u16  
read_range  
read_direct_u16  
read_direct_i64  
read_i64  
read_i32  
read_direct_u64  
read_i8  
read_u32  
read_u8  
read_u64  
read_direct_u32  
read_direct_i16  
read_direct_i32  
read_direct_u8  
read_i16  
[MAME]> print(mem:read_direct_i8
```

FERRAMENTAS ADICIONAIS

10.1 Chdman

O chdman é um gerenciador de arquivos CHD que pode ser usado para criar, converter, verificar a integridade dos arquivos e extrair dados das imagens de mídia no formato CHD. CHD é um acrônimo para pedaços ou blocos de dados.

A utilização típica é `chdman <comando> <opção> . . .`

Nota: Parent CHD = CHD principal, para obter mais informações sobre termos como "parent", "clone" e outros, consulte *Sobre as ROMs e os seus conjuntos*.

- *Opções comuns*
- *Comandos*
 - *info*
 - *verify*
 - *createraw*
 - *createhd*
 - *createcd*
 - *createdvd*
 - *createld*
 - *extractraw*
 - *extracthd*
 - *extractcd*
 - *extractdvd*
 - *extractld*
 - *addmeta*
 - *delmeta*
 - *dumpmeta*
 - *listtemplates*
- *Algoritmos de compressão*

10.1.1 Opções comuns

A disponibilidade das opções abaixo variam de acordo com o comando:

--input <arquivo> / -i <arquivo>

Define o arquivo de entrada. Esta opção é obrigatória para a maioria dos comandos. Dependendo do comando, os formatos compatíveis do arquivo de entrada poderão variar.

--inputparent <arquivo_chd> / -ip <arquivo_chd>

Define o arquivo CHD principal como o arquivo de entrada. Esta opção é compatível com a maioria dos comandos que operam com arquivos de entrada no formato CHD. Esta opção deve ser usada caso o arquivo de entrada seja um *delta CHD*. O *delta CHD* é um arquivo que armazena apenas a diferença dos blocos do CHD principal.

--inputstartbyte <offset> / -isb <offset>

Define o offset em bytes para os dados no arquivo de entrada. Isso é útil para criar arquivos no formato CHD a partir de arquivos de entrada que possuam um cabeçalho antes do início dos dados, ou, para extrair conteúdo parcial de um arquivo no formato CHD. Não pode ser usado em conjunto com a opção --inputstarthunk/-ish.

--inputstarthunk <offset> / -ish <offset>

Define o offset para os dados no arquivo de entrada em blocos. Não pode ser usado em conjunto com a opção --inputstartbyte/-isb.

--inputbytes <comprimento> / -ib <comprimento>

Define a quantidade de dados de entrada em bytes, começando pelo offset dos dados na entrada do arquivo. Isso é útil para criar arquivos no formato CHD a partir de arquivos de entrada que contêm conteúdo adicional após os dados ou para extrair conteúdo parcial de um arquivo no formato CHD. Não pode ser usado em conjunto com a opção --inuthunks/-ih.

--inuthunks <comprimento> / -ih <comprimento>

Define a quantidade de dados de entrada nos blocos, começando pelo offset dos dados na entrada do arquivo. Não pode ser usado em conjunto com a opção --inputbytes/-ib.

--output <arquivo> / -o <arquivo>

Define o nome do arquivo de saída. Esta opção é necessária para comandos que produzem arquivos de saída. Dependendo do comando, os formatos compatíveis do arquivo de saída poderão variar.

--outputparent <arquivo_chd> / -op <arquivo_chd>

Define o nome do arquivo de saída do CHD principal. Esta opção é compatível com os comandos que geram arquivos no formato CHD. O uso desta opção produz um *delta CHD*. O *delta CHD* é um arquivo que armazena apenas a diferença dos blocos do CHD principal. A mídia do CHD principal deve ser do mesmo tipo, do mesmo tamanho e ter blocos com o mesmo tamanho.

--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...

Define quais algoritmos de compressão usar. Esta opção é compatível com os comandos que geram arquivos no formato CHD. Use none para desativar a compressão ou especifique até quatro algoritmos de compressão separados por vírgula. Consulte [algoritmos de compressão](#) para conhecer os algoritmos de compressão compatíveis. A compressão deve ser desativada para criar arquivos de imagem de mídia graváveis.

--hunksize <bytes> / -hs <bytes>

Define o tamanho do bloco em bytes. Esta opção é compatível com os comandos que geram arquivos no formato CHD. O tamanho do bloco não pode ser menor que 16 bytes nem maior que 1048576 bytes (1 MiB). O tamanho do bloco deve ser um múltiplo do tamanho do setor ou do tamanho da unidade da mídia. Blocos com tamanhos maiores podem proporcionar melhores taxas de compressão, mas reduzem o desempenho nas pequenas leituras aleatórias, pois um bloco inteiro precisa ser lido e descomprimido de cada vez.

--force / -f

Substitui os arquivos gerados caso já existam. Esta opção é compatível com os comandos que geram arquivos.

--verbose / -v

Ativa o modo loquaz. Em alguns comandos esta opção gera informações mais detalhadas.

--numprocessors <quantidade> / -np <quantidade>

Limita a quantidade máxima de threads utilizada para a compressão de dados. Esta opção é compatível com os comandos que geram arquivos no formato CHD.

10.1.2 Comandos

info

Exibe informações de um arquivo no formato CHD. As informações incluem:

- Versão do formato CHD e os algoritmos de compressão utilizados.
- O valor de tamanhos compactados e não compactados e a taxa geral de compressão.
- O tamanho do bloco, o tamanho da unidade e a quantidade de blocos no arquivo.
- Informações do SHA1 e dos metadados.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--verbose / -v`

Consulte também o capítulo *Gerando um arquivo de registro de um arquivo CHD*.

verify

Verifica a integridade de um arquivo no formato CHD. O arquivo de entrada deve ser um arquivo CHD somente leitura (a integridade dos arquivos graváveis do CHD não poderá ser verificada). Observe que este comando altera o arquivo de entrada caso a opção `--fix/-f` seja utilizada.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--inputparent <arquivo_chd> / -ip <arquivo_chd>`

Opções adicionais:

- `--fix / -f`

createraw

Cria um arquivo no formato CHD a partir de uma imagem de mídia em formato RAW.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--inputstartbyte <offset> / -isb <offset>`
- `--inputstarthunk <offset> / -ish <offset>`
- `--inputbytes <comprimento> / -ib <comprimento>`
- `--inputhunks <comprimento> / -ih <comprimento>`
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--outputparent <arquivo_chd> / -op <arquivo_chd>`
- `--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...`
- `--hunksize <bytes> / -hs <bytes>`
- `--force / -f`
- `--numprocessors <quantidade> / -np <quantidade>`

Opções adicionais:

--unitsize <bytes> / -us <bytes> (obrigatório)

A unidade de tamanho em bytes para gerar o arquivo CHD. Esta é a menor unidade de dados que pode ser endereçada dentro do arquivo CHD. Ela deve corresponder ao tamanho do setor ou da página original da mídia. O tamanho do bloco deve ser um múltiplo inteiro do tamanho da unidade. A unidade deve ser definida caso nenhum CHD principal tenha sido informado para gerar o arquivo final. Caso um CHD principal seja informado para gerar um arquivo CHD, o tamanho da unidade deverá corresponder ao tamanho da unidade do arquivo do CHD principal.

Na ausência das opções `--hunksize` ou `-hs`, as predefinições serão:

- O tamanho do bloco do arquivo do CHD principal, caso um arquivo CHD principal seja informado para gerar o arquivo final.
- O menor múltiplo inteiro do tamanho da unidade, não maior que 4 KiB, caso o tamanho da unidade não seja maior que 4 KiB (4096 bytes).
- O tamanho da unidade, caso seja maior que 4 KiB (4096 bytes).

Na ausência das opções `--compression` ou `-c`, a predefinição será `lzma,zlib,huff,flac`.

createhd

Cria um arquivo de imagem de um disco rígido no formato CHD.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>`
- `--inputstartbyte <offset> / -isb <offset>`
- `--inputstarthunk <offset> / -ish <offset>`
- `--inputbytes <comprimento> / -ib <comprimento>`
- `--inputhunks <comprimento> / -ih <comprimento>`
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--outputparent <arquivo_chd> / -op <arquivo_chd>`
- `--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...`
- `--hunksize <bytes> / -hs <bytes>` (obrigatório)
- `--force / -f`
- `--numprocessors <quantidade> / -np <quantidade>`

Opções adicionais:

- `--sectorsize <bytes> / -ss <bytes>`
- `--size <bytes> / -s <bytes>`
- `--chs <cylinders>,<heads>,<sectors> / -chs <cylinders>,<heads>,<sectors>`
- `--template <template> / -tp <template>`

Cria uma imagem de disco rígido vazia (preenchida com zeros). Caso nenhuma entrada seja fornecida, as seguintes opções de entrada início/comprimento não poderão ser utilizadas (`--inputstartbyte/-isb`, `--inputstarthunk/-ish`, `--inputbytes/-ib` e `--inputhunks/-ih`).

Na ausência das opções `--hunksize` ou `-hs`, as predefinições serão:

- O tamanho do bloco do arquivo do CHD principal, caso um arquivo CHD principal seja informado para gerar o arquivo final.
- O menor múltiplo inteiro do tamanho do setor, não maior que 4 KiB, caso o tamanho do setor não seja maior que 4 KiB (4096 bytes).
- O tamanho do setor, caso seja maior que 4 KiB (4096 bytes).

Na ausência das opções `--compression` ou `-c` e se um arquivo de entrada for informado, a predefinição será `lzma`, `zlib`, `huff`, `flac` ou `none` na ausência de um arquivo de entrada.

createcd

Cria um arquivo de imagem de um CD-ROM no formato CHD.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--outputparent <arquivo_chd> / -op <arquivo_chd>`
- `--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...`
- `--hunksize <bytes> / -hs <bytes>` (obrigatório)
- `--force / -f`
- `--numprocessors <quantidade> / -np <quantidade>`

Caso as opções `--hunksize` ou `-hs` não sejam usadas, a predefinição será o tamanho do bloco do CHD principal se o CHD principal for informado para gerar o arquivo final, caso contrário, serão oito setores por bloco (18.816 bytes).

Na ausência das opções `--compression` ou `-c` a predefinição será `cdlz`, `cdzl`, `cdf1`.

Consulte também o capítulo [Convertendo um CD-ROM para CHD com o chdman](#).

createdvd

Cria um arquivo de imagem de um DVD-ROM no formato CHD.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--inputstartbyte <offset> / -isb <offset>`
- `--inputstarthunk <offset> / -ish <offset>`
- `--inputbytes <comprimento> / -ib <comprimento>`
- `--inputhunks <comprimento> / -ih <comprimento>`
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--outputparent <arquivo_chd> / -op <arquivo_chd>`
- `--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...`
- `--hunksize <bytes> / -hs <bytes>` (obrigatório)
- `--force / -f`
- `--numprocessors <quantidade> / -np <quantidade>`

Caso as opções `--hunksize` ou `-hs` não sejam usadas, a predefinição será o tamanho do bloco do CHD principal se o CHD principal for informado para gerar o arquivo final, caso contrário, serão dois setores por bloco (4096 bytes).

Na ausência das opções `--compression` ou `-c` a predefinição será `lzma,zlib,huff,flac`.

createld

Cria um arquivo de imagem de um LaserDisc no formato CHD.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--outputparent <arquivo_chd> / -op <arquivo_chd>`
- `--compression none|<tipo>[,<tipo>]... / -c none|<tipo>[,<tipo>]...`
- `--hunksize <bytes> / -hs <bytes>` (obrigatório)
- `--force / -f`
- `--numprocessors <quantidade> / -np <quantidade>`

Opções adicionais:

- `--inputstartframe <offset> / -isf <offset>`
- `--inputframes <comprimento> / -if <comprimento>`

Na ausência das opções `--compression` ou `-c` a predefinição será `avhu`.

extractraw

Extrai dados de um CHD para uma imagem de mídia bruta (RAW).

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--inputparent <arquivo_chd> / -ip <arquivo_chd>`
- `--inputstartbyte <offset> / -isb <offset>`
- `--inputstarthunk <offset> / -ish <offset>`
- `--inputbytes <comprimento> / -ib <comprimento>`
- `--inputhunks <comprimento> / -ih <comprimento>`
- `--output <arquivo> / -o <arquivo>` (obrigatório)
- `--force / -f`

extracthd

Extrai dados de um CHD para uma imagem de disco rígido.

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)
- --inputparent <arquivo_chd> / -ip <arquivo_chd>
- --inputstartbyte <offset> / -isb <offset>
- --inputstarthunk <offset> / -ish <offset>
- --inputbytes <comprimento> / -ib <comprimento>
- --inputhunks <comprimento> / -ih <comprimento>
- --output <arquivo> / -o <arquivo> (obrigatório)
- --force / -f

extractcd

Extrai dados de um CHD para uma imagem de CD-ROM.

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)
- --inputparent <arquivo_chd> / -ip <arquivo_chd>
- --output <arquivo> / -o <arquivo> (obrigatório)
- --force / -f

Opções adicionais:

- --outputbin <arquivo> / -ob <arquivo>
- --splitbin / -sb

extractdvd

Extrai dados de um CHD para uma imagem de DVD-ROM.

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)
- --inputparent <arquivo_chd> / -ip <arquivo_chd>
- --inputstartbyte <offset> / -isb <offset>
- --inputstarthunk <offset> / -ish <offset>
- --inputbytes <comprimento> / -ib <comprimento>
- --inputhunks <comprimento> / -ih <comprimento>
- --output <arquivo> / -o <arquivo> (obrigatório)
- --force / -f

extractld

Extrai dados de um CHD para uma imagem de LaserDisc.

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)
- --inputparent <arquivo_chd> / -ip <arquivo_chd>
- --output <arquivo> / -o <arquivo> (obrigatório)
- --force / -f

Opções adicionais:

- --inputstartframe <offset> / -isf <offset>
- --inputframes <comprimento> / -if <comprimento>

addmeta

Adiciona metadados ao arquivo CHD. Este comando altera o seu arquivo de entrada..

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)

Opções adicionais:

- --tag <tag> / -t <tag> (obrigatório)
- --index <index> / -ix <index>
- --valuetext <text> / -vt <text>
- --valuefile <arquivo> / -vf <arquivo>
- --nochecksum / -nocs

delmeta

Excluí metadados do arquivo CHD. Este comando altera o seu arquivo de entrada..

Opções compatíveis:

- --input <arquivo> / -i <arquivo> (obrigatório)

Opções adicionais:

- --tag <tag> / -t <tag> (obrigatório)
- --index <index> / -ix <index>

dumpmeta

Extraí os metadados de um arquivo CHD para uma saída padrão ou para um arquivo.

Opções compatíveis:

- `--input <arquivo> / -i <arquivo>` (obrigatório)
- `--output <arquivo> / -o <arquivo>`
- `--force / -f`

Opções adicionais:

- `--tag <tag> / -t <tag>` (obrigatório)
- `--index <index> / -ix <index>`

listtemplates

Lista os modelos de disco rígido disponíveis. Esse comando não aceita nenhuma opção.

10.1.3 Algoritmos de compressão

Há suporte para os seguintes algoritmos de compressão:

zlib – zlib deflate

Comprime os dados usando o algoritmo *zlib deflate*.

zstd – Zstandard

Comprime os dados usando o algoritmo *Zstandard*. Isso proporciona um desempenho muito bom de compressão e descompressão com taxas de compressão melhores do que o *zlib deflate*, porém, programas mais antigos podem não ser compatíveis com arquivos CHD que usam a compressão *Zstandard*.

lzma – Lempel-Ziv-Markov chain algorithm

Comprime os dados usando o algoritmo *Lempel-Ziv-Markov-chain (LZMA)*. Isso proporciona altas taxas de compressão ao custo de um desempenho ruim de compressão e descompressão.

huff – Huffman coding

Comprime os dados usando a codificação de entropia Huffman de 8 bits.

flac – Free Lossless Audio Codec

Comprime os dados como áudio PCM com dois canais (estéreo) de 16 bits e 44,1 kHz usando o *Free Lossless Audio Codec (FLAC)*. Isso proporciona boas taxas de compressão caso a mídia contenha dados de áudio PCM com 16 bits.

cdzl – zlib deflate for CD-ROM data

Comprime os dados de áudio e comprime separadamente os dados dos subcanais dos setores do CD-ROM usando o algoritmo *zlib deflate*.

cdzs – Zstandard for CD-ROM data

Comprime os dados de áudio e comprime separadamente os dados dos subcanais dos setores do CD-ROM usando o algoritmo *Zstandard*. Isso proporciona um desempenho muito bom de compressão e descompressão tendo melhores taxas de compressão se comparado com o *zlib deflate*, porém, programas mais antigos podem não ser compatíveis com arquivos CHD que usam a compressão *Zstandard*.

cdlz - Lempel-Ziv-Markov chain algorithm/zlib deflate for CD-ROM data

Comprime os dados de áudio e comprime separadamente os dados dos subcanais dos setores do CD-ROM usando o algoritmo de cadência *Lempel-Ziv-Markov (LZMA)* para dados de áudio e o algoritmo *zlib deflate* para dados dos subcanais. Isso proporciona altas taxas de compressão ao custo de um desempenho ruim de compressão e descompressão.

cdfl – Free Lossless Audio Codec/zlib deflate for CD-ROM data

Comprime os dados de áudio e comprime separadamente os dados dos subcanais dos setores do CD-ROM usando o algoritmo *Free Lossless Audio Codec (FLAC)* para dados de áudio e o algoritmo *zlib deflate* para os dados dos subcanais. Isso proporciona boas taxas de compressão para as trilhas de áudio CD.

avhu – Huffman coding for audio-visual data

Esse é um algoritmo de compressão especializado para dados audiovisuais (A/V). Ele só deve ser usado para criar arquivos CHD vindos de um LaserDisc.

10.2 Imgtool

Imgtool é uma ferramenta usada para a manutenção e manipulação de imagens de discos de diferentes tipos que os usuários precisam aprender a usar. As funções incluem recuperar e armazenar os arquivos com verificação/validação por CRC.

A ferramenta faz parte do projeto MAME compartilhando grande parte do seu código e a mesma não existiria se não fosse pelo MAME. Logo, os termos da sua distribuição seguem os mesmos termos existentes para o MAME. Favor ler a toda [LICENÇA](#) com atenção.

Algumas porções do Imgtool contém direitos autorais dos Regentes da Universidade da Califórnia (c) 1989, 1993. Todos os direitos reservados.

10.2.1 Utilização

Imgtool é um programa de linha comando que contém alguns "subcomandos" que fazem todo o trabalho. A maioria dos comandos são invocados usando uma cadência de instruções, exemplo:

```
imgtool <subcommand> <format> <image> ...
```

- **<subcommand>** é o nome do subcomando
- **<format>** é o formato da imagem
- **<image>** é o nome da imagem

Exemplo de uso:

```
imgtool dir coco_jvc_rsdos myimageinazip.zip
imgtool get coco_jvc_rsdos myimage.dsk myfile.bin mynewfile.txt
imgtool getall coco_jvc_rsdos myimage.dsk
```

As variações dos subcomandos serão dadas mais adiante. Observe que nem todos os subcomandos são compatíveis ou aplicáveis em todos os variados tipos diferentes de imagem.

10.2.2 Subcomandos

create

```
imgtool create <format> <imagename> [--(createoption)=value]
```

- **<format>** é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- **<imagename>** é o nome de destino da imagem, é possível especificar um arquivo ZIP como nome da imagem

Cria uma imagem

dir

```
imgtool dir <format> <imagename> [path]
```

- **<format>** é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- **<imagename>** é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Lista o conteúdo de uma imagem

get

imgtool get *<format>* *<imagenname>* *<filename>* [*newname*] [--filter=filter] [--fork=fork]

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Extrai um arquivo da imagem

put

imgtool put *<format>* *<imagenname>* *<filename>*... *<destname>* [--(fileoption)=value] [--filter=filter] [--fork=fork]

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Adiciona um arquivo na imagem (é compatível com coringas)

getall

imgtool getall *<format>* *<imagenname>* [*path*] [--filter=filter]

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Extrai todos os arquivos de uma imagem

del

imgtool del *<format>* *<imagenname>* *<filename>*...

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Apaga todos os arquivos de uma imagem

mkdir

imgtool mkdir *<format>* *<imagenname>* *<dirname>*

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Cria um subdiretório numa imagem

rmdir

imgtool rmdir *<format>* *<imagenname>* *<dirname>*...

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagenname>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Apaga um subdiretório numa imagem

readsector

imgtool readsector *<format>* *<imagename>* *<track>* *<head>* *<sector>* *<filename>*

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagename>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Lê o setor de uma imagem e grava num nome de arquivo *<filename>* específico.

writesector

imgtool writesector *<format>* *<imagename>* *<track>* *<head>* *<sector>* *<filename>*

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagename>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

Escreve no setor de uma imagem vinda de um arquivo *<filename>* especificado

identify

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo
- *<imagename>* é o nome de destino da imagem; é possível especificar um arquivo ZIP como nome da imagem

imgtool identify *<imagename>*

listformats

Exibe uma lista com todos os formatos de imagem compatíveis com o imgtool

listfilters

Exibe uma lista de todos os filtros compatíveis com o imgtool

listdriveroptions

imgtool listdriveroptions *<format>*

- *<format>* é o nome do formato da imagem, coco_jvc_rsdos por exemplo

Exibe uma lista completa de todas as opções relacionadas a um formato em específico para os comandos 'put' e 'create'.

10.2.3 Filtros

Os filtros são uma maneira de processar a maneira que os dados estão sendo escritos ou lidos numa imagem. Os filtros podem ser usados nos comandos **get**, **put** e **getall** ao usar a opção **--filter=xxxx** na linha de comando. Atualmente, os seguintes filtros são compatíveis:

ascii

Converte o final de linha dos arquivos para o formato apropriado

cocobas

Processa programas BASIC tokenizados para Computadores TRS-80 Color (CoCo)

dragonbas

Processa programas BASIC tokenizados para o Tano/Dragon Data Dragon 32/64

macbinary

Processa arquivos de imagem (merged forks) Apple em formato MacBinary

vzsnapshot

[a fazer: VZ Snapshot? Descobrir o que é isso...]

vzbas

Processa programas BASIC tokenizados para o Laser/VZ

thombas5

Processa programas BASIC tokenizados para o Thomson MO5 com BASIC 1.0 (apenas leitura, descriptografia automática)

thombas7

Processa programas BASIC tokenizados para o Thomson TO7 com BASIC 1.0 (apenas leitura, descriptografia automática)

thombas128

Processa programas BASIC tokenizados para o Thomson com BASIC 128/512 (apenas leitura, descriptografia automática)

thomcrypt

Processa programas BASIC tokenizados para o Thomson BASIC, protegidos por criptografia (sem tokenização)

bm13bas

Processa arquivos BASIC, Basic Master Level 3 tokenizados

10.2.4 Lista dos formatos compatíveis

10.2.5 Imagem de disquete do Amiga (formato OFS/FFS) - (*amiga_floppy*)

Opções específicas do driver para o módulo '*amiga_floppy*':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 1: *amiga_floppy*

Opções	Valores permitidos	Descrição
--density	dd/hd	Densidade
--filesystem	ofs/ffs	Sistema de Arquivos
--mode	none/intl/dirc	Opções do sistema de arquivos

10.2.6 Apple][imagem de disco DOS order (formato ProDOS) - (*apple2_do_prodos_525*)

Opções específicas do driver para o módulo 'apple2_do_prodos_525':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 2: apple2_do_prodos_525

Opções	Valores permitidos	Descrição
--heads	1	Cabeças
--tracks	35	Pistas
--sectors	16	Setores
--sectorlength	256	Bytes por setor
--firstsectorid	0	Primeiro setor

10.2.7 Apple][imagem de disco Nibble order (formato ProDOS) - (*apple2_nib_prodos_525*)

Opções específicas do driver para o módulo 'apple2_nib_prodos_525':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 3: apple2_nib_prodos_525

Opções	Valores permitidos	Descrição
--heads	1	Cabeças
--tracks	35	Pistas
--sectors	16	Setores
--sectorlength	256	Bytes por setor
--firstsectorid	0	Primeiro setor

10.2.8 Apple][imagem de disco ProDOS order (formato ProDOS) - (*apple2_po_prodos_525*)

Opções específicas do driver para o módulo 'apple2_po_prodos_525':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 4: apple2_po_prodos_525

Opções	Valores permitidos	Descrição
--heads	1	Cabeças
--tracks	35	Pistas
--sectors	16	Setores
--sectorlength	256	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.9 Apple][imagem de disco gs 2IMG (formato ProDOS) - (*apple35_2img_prodos_35*)

Opções específicas do driver para o módulo 'apple35_2img_prodos_35':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 5: apple35_2img_prodos_35

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.10 Imagem de disco para o Apple DiskCopy (Disquete Mac HFS) - (*apple35_dc_mac_hfs*)

Opções específicas do driver para o módulo 'apple35_dc_mac_hfs':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 6: apple35_dc_mac_hfs

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.11 Imagem de disco para o Apple DiskCopy (Disquete Mac MFS) - (*apple35_dc_mac_mfs*)

Opções específicas do driver para o módulo 'apple35_dc_mac_mfs':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 7: apple35_dc_mac_mfs

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.12 Imagem de disco para o Apple DiskCopy (formato ProDOS) - (*apple35_dc_prodos_35*)

Opções específicas do driver para o módulo 'apple35_dc_prodos_35':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 8: apple35_dc_prodos_35

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.13 Imagem de disco para o Apple raw 3.5' (Disquete Mac HFS) - (*apple35_raw_mac_hfs*)

Opções específicas do driver para o módulo 'apple35_raw_mac_hfs':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 9: apple35_raw_mac_hfs

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.14 Imagem de disco para o Apple raw 3.5' (Disquete Mac MFS) - (*apple35_raw_mac_mfs*)

Opções específicas do driver para o módulo 'apple35_raw_mac_mfs':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 10: apple35_raw_mac_mfs

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.15 Imagem de disco para o Apple raw 3.5' (formato ProDOS) - (*apple35_raw_prodos_35*)

Opções específicas do driver para o módulo 'apple35_raw_prodos_35':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 11: apple35_raw_prodos_35

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	80	Pistas
--sectorlength	512	Bytes por Setor
--firstsectorid	0	Primeiro setor

10.2.16 Imagem de disco para o CoCo DMK (formato OS-9) - (*coco_dmk_os9*)

Opções específicas do driver para o módulo 'coco_dmk_os9':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 12: coco_dmk_os9

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	1-18	Setores
--sectorlength	128/256/512/1024/2048/4096/8192	Bytes por Setor
--interleave	0-17	Intercalação
--firstsectorid	0-1	Primeiro setor

10.2.17 Imagem de disco para o CoCo DMK (formato RS-DOS) - (*coco_dmk_rsdos*)

Opções específicas do driver para o módulo 'coco_dmk_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 13: coco_dmk_rsdos (put)

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário ¹

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 14: coco_dmk_rsdos (create)

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	1-18	Setores
--sectorlength	128/256/512/1024/2048/4096/8192	Bytes por Setor
--interleave	0-17	Intercalação
--firstsectorid	0-1	Primeiro setor

10.2.18 Imagem de disco para o CoCo JVC (formato OS-9) - (*coco_jvc_os9*)

Opções específicas do driver para o módulo 'coco_jvc_os9':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 15: coco_jvc_os9

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	1-255	Setores
--sectorlength	128/256/512/1024	Tamanho do setor
--firstsectorid	0-1	Primeiro setor

10.2.19 Imagem de disco para o CoCo JVC (formato RS-DOS) - (*coco_jvc_rsdos*)

Opções específicas do driver para o módulo 'coco_jvc_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 16: coco_jvc_rsdos (put)

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

¹ ASCII flag. (Nota do tradutor)

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 17: coco_jvc_rsdos (create)

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	1-255	Setores
--sectorlength	128/256/512/1024	Tamanho do setor
--firstsectorid	0	Primeiro setor

10.2.20 Imagem de disco para o CoCo OS-9 (formato OS-9) - (coco_os9_os9)

Opções específicas do driver para o módulo 'coco_os9_os9':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 18: coco_os9_os9

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	1-255	Setores
--sectorlength	128/256/512/1024	Tamanho do setor
--firstsectorid	1	Primeiro setor

10.2.21 Imagem de disco para o CoCo VDK (formato OS-9) - (coco_vdk_os9)

Opções específicas do driver para o módulo 'coco_vdk_os9':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 19: coco_vdk_os9

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	18	Setores
--sectorlength	256	Tamanho do setor
--firstsectorid	1	Primeiro setor

10.2.22 Imagem de disco para o CoCo VDK (formato RS-DOS) - (*coco_vdk_rsdos*)

Opções específicas do driver para o módulo 'coco_vdk_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 20: coco_vdk_rsdos (put)

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 21: coco_vdk_rsdos (create)

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	35-255	Pistas
--sectors	18	Setores
--sectorlength	256	Tamanho do setor
--firstsectorid	1	Primeiro setor

10.2.23 Imagem de disquete para o Concept - (*concept*)

Opções específicas do driver para o módulo 'concept':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.24 Imagem de disquete para o CopyQM (formato Basic Master Level 3) - (*cqm_bml3*)

Opções específicas do driver para o módulo 'cqm_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 22: cqm_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.25 Imagem de disquete para o CopyQM (formato FAT) - (*cqm_fat*)

Opções específicas do driver para o módulo 'cqm_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.26 Imagem de disquete para o CopyQM (Mac HFS Floppy) - (*cqm_mac_hfs*)

Opções específicas do driver para o módulo 'cqm_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.27 Imagem de disquete para o CopyQM (Disquete Mac MFS) - (*cqm_mac_mfs*)

Opções específicas do driver para o módulo 'cqm_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.28 Imagem de disquete para o CopyQM (formato OS-9) - (*cqm_os9*)

Opções específicas do driver para o módulo 'cqm_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.29 Imagem de disquete para o CopyQM (formato ProDOS) - (*cqm_prodos_35*)

Opções específicas do driver para o módulo 'cqm_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.30 Imagem de disquete para o CopyQM (formato ProDOS) - (*cqm_prodos_525*)

Opções específicas do driver para o módulo 'cqm_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.31 Imagem de disquete para o CopyQM (formato RS-DOS) - (*cqm_rsdos*)

Opções específicas do driver para o módulo 'cqm_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 23: cqm_rsdos

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.32 Imagem de disquete para o CopyQM (formato VZ-DOS) - (*cqm_vzdos*)

Opções específicas do driver para o módulo 'cqm_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 24: cqm_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.33 Sistema de arquivos para o Cybiko Classic - (*cybiko*)

Opções específicas do driver para o módulo 'cybiko':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 25: cybiko

Opções	Valores permitidos	Descrição
--flash	AT45DB041/AT45DB081/AT45DB161	Modelo da memória flash

10.2.34 Sistema de arquivos para o Cybiko Xtreme - (*cybikox*)

Opções específicas do driver para o módulo 'cybikox':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.35 Imagem de disquete para o D88 (formato Basic Master Level 3) - (*d88_bml3*)

Opções específicas do driver para o módulo 'd88_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 26: d88_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.36 Imagem de disquete para o D88 (formato FAT) - (*d88_fat*)

Opções específicas do driver para o módulo 'd88_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.37 Imagem de disquete para o D88 (Disquete Mac HFS) - (*d88_mac_hfs*)

Opções específicas do driver para o módulo 'd88_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.38 Imagem de disquete para o D88 (Disquete Mac MFS) - (*d88_mac_mfs*)

Opções específicas do driver para o módulo 'd88_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.39 Imagem de disquete para o D88 (formato OS-9) - (*d88_os9*)

Opções específicas do driver para o módulo 'd88_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.40 Imagem de disquete para o D88 (formato OS-9) - (*d88_os9*)

Opções específicas do driver para o módulo 'd88_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.41 Imagem de disquete para o D88 (formato ProDOS) - (*d88_prodos_525*)

Opções específicas do driver para o módulo 'd88_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.42 Imagem de disquete para o D88 (formato RS-DOS) - (*d88_rsdos*)

Opções específicas do driver para o módulo 'd88_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 27: d88_rsdos

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.43 Imagem de disquete para o D88 (formato VZ-DOS) - (*d88_vzdos*)

Opções específicas do driver para o módulo 'd88_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 28: d88_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.44 Imagem de disquete para o DSK (formato Basic Master Level 3) - (*dsk_bml3*)

Opções específicas do driver para o módulo 'dsk_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 29: dsk_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.45 Imagem de disquete para o DSK (formato FAT) - (*dsk_fat*)

Opções específicas do driver para o módulo 'dsk_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.46 Imagem de disquete para o DSK (disquete Mac HFS) - (*dsk_mac_hfs*)

Opções específicas do driver para o módulo 'dsk_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.47 Imagem de disquete DSK (Disquete Mac MFS) - (*dsk_mac_mfs*)

Opções específicas do driver para o módulo 'dsk_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.48 Imagem de disquete para o DSK (formato OS-9) - (*dsk_os9*)

Opções específicas do driver para o módulo 'dsk_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.49 Imagem de disquete para o DSK (formato ProDOS) - (*dsk_prodos_35*)

Opções específicas do driver para o módulo 'dsk_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.50 Imagem de disquete para o DSK (formato ProDOS) - (*dsk_prodos_525*)

Opções específicas do driver para o módulo 'dsk_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.51 Imagem de disquete para o DSK (formato RS-DOS) - (*dsk_rsdos*)

Opções específicas do driver para o módulo 'dsk_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 30: dsk_rsdos

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.52 Imagem de disquete para o DSK (formato VZ-DOS) - (*dsk_vzdos*)

Opções específicas do driver para o módulo 'dsk_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 31: cqm_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.53 Imagem de Disco Formatado (formato Basic Master Level 3) - (*fdi_bml3*)

Opções específicas do driver para o módulo 'fdi_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 32: fdi_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.54 Imagem de Disco Formatado (formato FAT) - (*fdi_fat*)

Opções específicas do driver para o módulo 'fdi_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.55 Imagem de Disco Formatado (Disquete Mac HFS) - (*fdi_mac_hfs*)

Opções específicas do driver para o módulo 'fdi_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.56 Imagem de Disco Formatado (Disquete Mac MFS) - (*fdi_mac_mfs*)

Opções específicas do driver para o módulo 'fdi_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.57 Imagem de Disco Formatado (formato OS-9) - (*fdi_os9*)

Opções específicas do driver para o módulo 'fdi_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.58 Imagem de Disco Formatado (formato ProDOS) - (*fdi_prodos_35*)

Opções específicas do driver para o módulo 'fdi_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.59 Imagem de Disco Formatado (formato ProDOS) - (*fdi_prodos_525*)

Opções específicas do driver para o módulo 'fdi_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.60 Imagem de Disco Formatado (formato RS-DOS) - (*fdi_rsdos*)

Opções específicas do driver para o módulo 'fdi_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 33: *fdi_rsdos*

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.61 Imagem de Disco Formatado (formato VZ-DOS) - (*fdi_vzdos*)

Opções específicas do driver para o módulo 'fdi_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 34: fdi_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.62 Cartão de memória para o HP48 SX/GX - (*hp48*)

Opções específicas do driver para o módulo 'hp48':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 35: hp48

Opções	Valores permitidos	Descrição
--flash	AT45DB041/AT45DB081/AT45DB161	Modelo da memória flash

10.2.63 Imagem de disquete IMD (formato Basic Master Level 3) - (*imd_bml3*)

Opções específicas do driver para o módulo 'imd_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 36: imd_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.64 Imagem de disquete IMD (formato FAT) - (*imd_fat*)

Opções específicas do driver para o módulo 'imd_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.65 Imagem de disquete IMD (disquete Mac HFS) - (*imd_mac_hfs*)

Opções específicas do driver para o módulo 'imd_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.66 Imagem de disquete IMD (Disquete Mac MFS) - (*imd_mac_mfs*)

Opções específicas do driver para o módulo 'imd_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.67 Imagem de disquete IMD (formato OS-9) - (*imd_os9*)

Opções específicas do driver para o módulo 'imd_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.68 Imagem de disquete IMD (formato ProDOS) - (*imd_prodos_35*)

Opções específicas do driver para o módulo 'imd_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.69 Imagem de disquete IMD (formato ProDOS) - (*imd_prodos_525*)

Opções específicas do driver para o módulo 'imd_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.70 Imagem de disquete IMD (formato RS-DOS) - (*imd_rsdos*)

Opções específicas do driver para o módulo 'imd_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 37: imd_rsdos

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.71 Imagem de disquete IMD (formato VZ-DOS) - (*imd_vzdos*)

Opções específicas do driver para o módulo 'imd_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 38: imd_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.72 Imagem de disco rígido para o MESS - (*mess_hd*)

Opções específicas do driver para o módulo 'mess_hd':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 39: mess_hd

Opções	Valores permitidos	Descrição
--blocksize	1-2048	Tamanho do Bloco
--cylinders	1-65536	Cilindros
--heads	1-64	Cabeças
--sectors	1-4096	Setores
--seclen	128/256/512/1024/2048/4096 8192/16384/32768/65536	Tamanho do setor

10.2.73 Disquete para o TI99 (formato PC99) - (*pc99fm*)

Opções específicas do driver para o módulo 'pc99fm':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.74 Disquete para o TI99 (formato PC99 MFM) - (*pc99mfm*)

Opções específicas do driver para o módulo 'pc99mfm':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.75 Imagem de disco para o PC CHD - (*pc_chd*)

Opções específicas do driver para o módulo 'pc_chd':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 40: pc_chd

Opções	Valores permitidos	Descrição
--cylinders	10/20/30/40/50/60/70/80/90/100/110/120 130/140/150/160/170/180/190/200	Cilindros
--heads	1-16	Cabeças
--sectors	1-63	Setores

10.2.76 Imagem de disquete para o PC (formato FAT) - (*pc_dsk_fat*)

Opções específicas do driver para o módulo 'pc_dsk_fat':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 41: pc_dsk_fat

Opções	Valores permitidos	Descrição
--heads	1-2	Quantidade de cabeças
--tracks	40/80	Quantidade de pistas
--sectors	8/9/10/15/18/36	Quantidade de setores

10.2.77 Psion Organiser II Datapack - (*psionpack*)

Opções específicas do driver para o módulo 'psionpack':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 42: psionpack (put)

Opções	Valores permitidos	Descrição
--type	OB3/OPL/ODB	Tipo do arquivo
--id	0/145-255	ID do arquivo

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 43: psionpack (create)

Opções	Valores permitidos	Descrição
--size	8k/16k/32k/64k/128k	Tamanho do datapack
--ram	0/1	EPROM/RAM datapack
--paged	0/1	linear/paged datapack
--protect	0/1	datapack com escrita protegida
--boot	0/1	datapack inicializável
--copy	0/1	datapack com permissão de cópia

10.2.78 Imagem de disquete para o Teledisk (formato Basic Master Level 3) - (td0_bml3)

Opções específicas do driver para o módulo 'td0_bml3':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 44: td0_bml3

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.79 Imagem de disquete para o Teledisk (formato FAT) - (td0_fat)

Opções específicas do driver para o módulo 'td0_fat':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.80 Imagem de disquete para o Teledisk (Disquete Mac HFS) - (td0_mac_hfs)

Opções específicas do driver para o módulo 'td0_mac_hfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.81 Imagem de disquete para o Teledisk (Disquete Mac MFS) - (td0_mac_mfs)

Opções específicas do driver para o módulo 'td0_mac_mfs':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.82 Imagem de disquete para o Teledisk (OS-9 format) - (td0_os9)

Opções específicas do driver para o módulo 'td0_os9':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.83 Imagem de disquete para o Teledisk (formato ProDOS) - (td0_prodos_35)

Opções específicas do driver para o módulo 'td0_prodos_35':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.84 Imagem de disquete para o Teledisk (formato ProDOS) - (td0_prodos_525)

Opções específicas do driver para o módulo 'td0_prodos_525':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.85 Imagem de disquete para o Teledisk (RS-DOS format) - (td0_rsdos)

Opções específicas do driver para o módulo 'td0_rsdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 45: td0_rsdos

Opções	Valores permitidos	Descrição
--ftype	basic/data/binary/assembler	Tipo do arquivo
--ascii	ascii/binary	Sinaliza como ASCII ou binário

Nenhuma opção específica para a criação da imagem

10.2.86 Imagem de disquete para o Teledisk (VZ-DOS format) - (td0_vzdos)

Opções específicas do driver para o módulo 'td0_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 46: td0_vzdos

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Nenhuma opção específica para a criação da imagem

10.2.87 Imagem de disquete Thomson .fd, formato BASIC - (*thom_fd*)

Opções específicas do driver para o módulo 'thom_fd':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 47: thom_fd (put)

Opções	Valores permitidos	Descrição
--ftype	auto/B/D/M/A	Tipo do arquivo
--format	auto/B/A	Indicador do formato
--comment	(string)	Comentário

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 48: thom_fd (create)

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	40/80	Pistas
--density	SD/DD	Densidade Simples ou Dupla
--name	(string)	Nome do disquete

10.2.88 Imagem de disquete Thomson .fd, formato BASIC - (*thom_qd*)

Opções específicas do driver para o módulo 'thom_qd':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 49: thom_qd (put)

Opções	Valores permitidos	Descrição
--ftype	auto/B/D/M/A	Tipo do arquivo
--format	auto/B/A	Indicador do formato
--comment	(string)	Comentário

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 50: thom_qd (create)

Opções	Valores permitidos	Descrição
--heads	1-2	Cabeças
--tracks	25	Pistas
--density	SD/DD	Densidade Simples ou Dupla
--name	(string)	Nome do disquete

10.2.89 Imagem de disquete Thomson .fd, formato BASIC - (*thom_sap*)

Opções específicas do driver para o módulo 'thom_sap':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 51: thom_sap (put)

Opções	Valores permitidos	Descrição
--ftype	auto/B/D/M/A	Tipo do arquivo
--format	auto/B/A	Indicador do formato
--comment	(string)	Comentário

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 52: thom_sap (create)

Opções	Valores permitidos	Descrição
--heads	1	Cabeças
--tracks	40/80	Pistas
--density	SD/DD	Densidade Simples ou Dupla
--name	(string)	Nome do disquete

10.2.90 Imagem de Disco Rígido para o TI990 - (*ti990hd*)

Opções específicas do driver para o módulo 'ti990hd':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 53: ti990hd

Opções	Valores permitidos	Descrição
--cylinders	1-2047	Quantidade de cabeças
--heads	1-31	Quantidade de pistas
--sectors	1-256	Quantidade de Setores
--bytes per sector	Geralmente 25256-512 256-512	Bytes Por Setor [A fazer: O imgtool está com falhas nesta seção]

10.2.91 Disquete para o TI99 (formato antigo do MESS) - (*ti99_old*)

Opções específicas do driver para o módulo 'ti99_old':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 54: ti99_old

Opções	Valores permitidos	Descrição
--sides	1-2	Lados
--tracks	1-80	Pistas
--sectors	1-36	Setores 1 → 9 para DS, 1 → 18 para DD, 1 → 36 para AD
--protection	0-1	Proteção (0 para normal, 1 para protegido)
--density	Auto/DS/DD/AD	Tipos de densidade

10.2.92 Disco Rígido para o TI99 - (*ti99hd*)

Opções específicas do driver para o módulo '*ti99hd*':

Nenhuma opção específica da imagem

Nenhuma opção específica para a criação da imagem

10.2.93 Disquete para o TI99 (formato V9T9) - (v9t9)

Opções específicas do driver para o módulo 'v9t9':

Nenhuma opção específica da imagem

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 55: v9t9

Opções	Valores permitidos	Descrição
--sides	1-2	Lados
--tracks	1-80	Pistas
--sectors	1-36	Setores 1 → 9 para DS, 1 → 18 para DD, 1 → 36 para AD
--protection	0-1	Proteção (0 para normal, 1 para protegido)
--density	Auto/DS/DD/AD	Tipos de densidade

10.2.94 Imagem de disco para o Laser/VZ (formato VZ-DOS) - (vtech1_vzdos)

Opções específicas do driver para o módulo 'vtech1_vzdos':

Opções específicas para o arquivo (utilizável com o comando 'put')

Tabela 56: vtech1_vzdos (put)

Opções	Valores permitidos	Descrição
--ftype	basic/binary/data	Tipo do arquivo
--ascii	intern/extern	Nome do arquivo

Opções específicas para a criação da imagem (utilizável com o comando 'create'):

Tabela 57: vtech1_vzdos (create)

Opções	Valores permitidos	Descrição
--heads	1	Cabeças
--tracks	40	Pistas
--sectors	16	Setores
--sectorlength	154	Tamanho do setor
--firstsectorid	0	Primeiro setor

[A fazer: preencher as estruturas e descrever melhor os comandos. Essas descrições vieram do arquivo imgtool.txt e estão muito simplificadas]

10.3 Castool

Castool é uma ferramenta para manutenção e manipulação de imagens de fita k7 que os usuários precisarão aprender a usar. O MAME já é compatível com formatos de áudio em .WAV, porém muitas das imagens existentes, podem conter outros formatos como .TAP vinda de fitas Commodore 64, .CAS para Tandy Color Computer e assim por diante. A ferramenta Castool irá converter esses outros formatos para .WAV caso seja usada no MAME.

A ferramenta faz parte do projeto MAME. Ele compartilha grande parte do seu código com o MAME e a mesma não existiria se não fosse pelo MAME. Logo os termos da sua distribuição seguem os mesmos termos existentes para o MAME. Favor ler a toda [LICENÇA](#) com atenção.

10.3.1 Utilização

Castool é um programa de linha comando que contém um conjunto simples de instruções. Os comandos são invocados usando uma cadência de instruções, exemplo:

```
castool convert <format> <inputfile> <outputfile>
```

- **<format>** é o formato da imagem
- **<inputfile>** é o nome do arquivo que você estiver convertendo
- **<outputfile>** é o nome de saída do arquivo WAV

Exemplo de uso:

```
castool convert coco zaxxon.cas zaxxon.wav
castool convert cbm arkanoid.tap arkanoid.wav
castool convert ddp mybasicprogram.ddp mybasicprogram.wav
```

10.3.2 Lista dos formatos compatíveis

Estes são os formatos compatíveis com o Castool para a conversão dos arquivos para o formato WAV.

Tabela 58: Listagem dos formatos

Formato	Descrição	Nome da Extensão
A26	Imagem do Atari 2600 SuperCharger	a26
APF	APF Imagination Machine	cas, cpf, apt
ATOM	Acorn Atom	tap, csw, uef
BBC	Acorn BBC & Electron	csw, uef
CBM	Série Commodore 8-bits	tap
CDT	Amstrad CPC	cdt
CGENIE	EACA Colour Genie	cas
COCO	Tandy Radio Shack Color Computer	cas
CSW	Compressed Square Wave	csw
DDP	Coleco ADAM	ddp
FM7	Fujitsu FM-7	t77
FMSX	MSX	tap, cas
GTP	Elektronika inzenjering Galaksija	gtp

continua na próxima página

Tabela 58 – continuação da página anterior

Formato	Descrição	Nome da Extensão
HECTOR	Micronique Hector & Interact Family Computer	k7, cin, for
JUPITER	Jupiter Cantab Jupiter Ace	tap
KC85	VEB Mikroelektronik KC 85	kcc, kcb, tap, 853, 854, 855, tp2, kcm, sss
KIM1	MOS KIM-1	kim, kim1
LVIV	PK-01 Lviv	lvt, lvr, lv0, lv1, lv2, lv3
MO5	Thomson MO-series	k5, k7
MZ	Sharp MZ-700	m12, mzf, mzt
ORAO	PEL Varazdin Orazo	tap
ORIC	Tangerine Oric	tap
PC6001	NEC PC-6001	cas
PHC25	Sanyo PHC-25	phc
PMD85	Tesla PMD-85	pmd, tap, ptp
PRIMO	Microkey Primo	ptp
RKU	UT-88	rku
RK8	Mikro-80	rk8
RKS	Specialist	rks
RKO	Orion	rko
RKR	Radio-86RK	rk, rkr, gam, g16, pki
RKA	Zavod BRA Apogee BK-01	rka
RKM	Mikrosha	rkm
RKP	SAM SKB VM Partner-01.01	rkp
SC3000	Sega SC-3000	bit
SOL20	PTC SOL-20	svt
SORCERER	Exidy Sorcerer	tape
SORDM5	Sord M5	cas
SPC1000	Samsung SPC-1000	tap, cas
SVI	Spectravideo SVI-318 & SVI-328	cas
TO7	Thomson TO-series	k7
TRS8012	TRS-80 Level 2	cas
TVC64	Videoton TVC 64	cas
TZX	Sinclair ZX Spectrum	tzx, tap, blk
VG5K	Philips VG 5000	k7
VTECH1	Video Technology Laser 110-310	cas
VTECH2	Video Technology Laser 350-700	cas
X07	Canon X-07	k7, lst, cas
X1	Sharp X1	tap
ZX80_O	Sinclair ZX80	o, 80
ZX81_P	Sinclair ZX81	p, 81

10.4 Floptool

Floptool é uma ferramenta para manutenção e manipulação de imagens de disquete que os usuários precisam aprender a usar. O MAME já é compatível com formatos de áudio em .WAV, porém muitas das imagens existentes, podem conter outros formatos como .TAP vinda de fitas Comodore 64, .CAS para Tandy Color Computer e assim por diante. A ferramenta Castool irá converter esses outros formatos para .WAV caso seja usada no MAME.

A ferramenta faz parte do projeto MAME compartilhando grande parte do seu código e a mesma não existiria se não fosse pelo MAME. Logo, os termos da sua distribuição seguem os mesmos termos existentes para o MAME. Favor ler a toda [LICENÇA](#) com atenção.

10.4.1 Utilização

Floptool é um programa de linha comando que contém um conjunto simples de instruções. Os comandos são invocados usando uma cadência de instruções, exemplo:

floptool identify <inputfile> [<inputfile> ...]

floptool convert [input_format | auto] output_format <inputfile> <outputfile>

- <format> é o formato da imagem
- <input_format> é o formato do arquivo de entrada, se for desconhecido, use auto
- <output_format> é o formato de destino do arquivo
- <inputfile> é o nome do arquivo que você deseja converter ou identificar
- <outputfile> é o nome do arquivo final que foi convertido

Exemplo de uso:

```
floptool convert coco zaxxon.cas zaxxon.wav
floptool convert cbm arkanoid.tap arkanoid.wav
floptool convert ddp mybasicprogram.ddp mybasicprogram.wav
```

10.4.2 Lista dos formatos compatíveis

Estes são os formatos compatíveis com o Floptool usados para a conversão em outros formatos.

Tabela 59: Listagem dos formatos

Formato	Descrição	Nome da Extensão
MFI	Imagem de disquete do mame	mfi
DFI	Formato de extração DiscFerret flux	dfi
IPF	Imagem de disquete do SPS	ipf
MFM	Imagem de disquete do HxC Floppy Emulator	mfm
ADF	Imagem de disquete do Amiga ADF	adf
ST	Imagem de disquete do Atari ST	st
MSA	Imagem de disquete do Atari MSA	msa
PASTI	Imagem de disquete do Atari PASTI	stx
DSK	Formato CPC DSK	dsk
D88	Imagem de disco do D88	d77, d88, 1dd
IMD	Imagem de disco do IMD	imd
TD0	Imagem de disco do Teledisk	td0
CQM	Imagem de disco do CopyQM	cqm, cqi, dsk
PC	Imagem de disquete do PC	dsk, ima, img, ufi, 360
NASLITE	Imagem de disco do NASLite	img
DC42	Imagem DiskCopy 4.2	dc42
A2_16SECT	Imagem de disco do Apple II com 16 setores	dsk, do, po
A2_RWTS18	Imagem tipo RWTS18 do Apple II	rti
A2_EDD	Imagem EDD do Apple II	edd
ATOM	Imagem de disco do Acorn Atom	40t, dsk
SSD	Imagem de disco do Acorn SSD	ssd, bbc, img
DSD	Imagem de disco do Acorn DSD	dsd
DOS	Imagem de disco do Acorn DOS	img
ADFS_O	Imagem de disco do Acorn ADFS (OldMap)	adf, ads, adm, adl
ADFS_N	Imagem de disco do Acorn ADFS (NewMap)	adf
ORIC_DSK	Imagem de disco do Oric	dsk
APPLIX	Imagem de disco do Applix	raw
HPI	Imagem de disquete do HP9845A	hpi

10.5 Outras ferramentas incluídas com o MAME

10.5.1 ledutil.exe/ledutil.sh

No Microsoft Windows o **ledutil.exe** pode ser usado para controlar as luzes led do seu teclado para espelhar aquelas luzes presentes nos primeiros jogos de arcade como o Asteroids por exemplo.

Para ativar essa funcionalidade inicie o **ledutil.exe** da linha de comando. Rode o comando **ledutil.exe -kill** para interrompê-lo.

Nas plataformas SDLMAME como o macOS e Linux, o **ledutil.sh** poderá ser usado. Use o comando **ledutil.sh -a** para que ele seja fechado automaticamente ao sair do SDLMAME.

10.6 Ferramentas voltadas ao desenvolvimento

10.6.1 pngcmp

Essa ferramenta é usada em teste de regressão ao comparar instantâneos PNG vindos de um script teste **runtest.cmd** encontrado nos arquivos de código-fonte. Esse script só funciona no Windows.

10.6.2 nltool

Componente de conversão discreto.

10.6.3 nlwav

Componente discreto de conversão e ferramenta de teste.

10.6.4 jedutil

Ferramenta útil para extração de **PAL/PLA/PLD/GAL**. Ele pode converter entre o formato JED padrão da indústria e o formato binário compactado proprietário do MAME, pode mostrar também equações lógicas para os tipos de dispositivos que conhecem tal lógica interna.

10.6.5 ldresample

Essa ferramenta comprime novamente os dados de vídeo para laserdisc e VHS.

10.6.6 ldverify

Essa ferramenta é usada para comparar imagens de laserdisc ou VHS CHD vinda de uma fonte AVI.

10.6.7 romcmp

Esta ferramenta é utilizada para realizar comparações de dados básicos e verificações de integridade em "*dumps*" binários. Com a opção `-h`, ela também pode ser utilizada para calcular as funções de "*hash*".

10.6.8 unidasm

Disassembler universal para muitas das arquiteturas compatíveis com o MAME.

CONTRIBUINDO COM O MAME

Deseja contribuir com o MAME mas não sabe por onde começar? A boa notícia é que sempre há muito o que ser feito, especialmente para pessoas com uma grande variedade de habilidades.

11.1 Testando e relatando bugs

Algo que o MAME sempre pode fazer é mais testes e mais relatórios de bugs. Caso esteja familiarizado com um sistema que o MAME emula e nota algo de errado ou caso encontre um bug na interface do MAME, crie uma conta em [MAME Testers](#) e assumindo que já não tenha sido relatado, crie uma conta e relate o problema encontrado. Tenha certeza que você leia primeiro as regras para ter certeza de que vai começar da maneira correta. Observe que o "MAME Testers" só aceita bugs voltados ao usuário em versões específicas.

Para outros tipos de bugs, temos o [github issues](#) onde há um pouco mais de liberdade. Por exemplo, aqui aceitamos problemas voltados ao desenvolvimento (problemas com a API interna ou problemas com o sistema de compilação), solicitações e a correção de grandes regressões que evita antes cheguem a versão de um lançamento. Respeite o fato que o "issue tracker" **não é** um fórum de discussão ou de suporte, é apenas para relatar problemas que possam ser reproduzidos. Não abra um chamado para fazer perguntas ou para pedir ajuda. Assim como, tenha ciência que o versão que ali está é instável. Caso a revisão atual não compile ou esteja totalmente quebrada, é bem provável que a gente já saiba e não é necessário abrir um novo chamado para relatar o problema. Dê um tempo e veja se aparece uma atualização. Talvez queira comentar algo no respectivo commit sobre a mensagem de erro do compilador, especialmente se estiver fazendo uma compilação não ortodoxa, mas, suportada.

Ao abrir um "ticket", lembre-se de fornecer o máximo de informações possíveis para auxiliar outras pessoas a compreender, reproduzir e diagnosticar o problema.

Informações que são importantes descrever:

- O comportamento incorreto e o comportamento esperado ou correto. Seja específico: apenas dizer que "não funciona" geralmente não informa muita coisa.
- Os detalhes do ambiente, incluindo o seu sistema operacional, arquitetura da CPU, localidade do sistema e o idioma de exibição, se aplicável.
- Para problemas relacionados com a saída de vídeo, informe o seu hardware de vídeo (GPU), a versão do driver e a opção de vídeo usada no MAME que você usou (ou não).
- Para problemas relacionados com a entrada, inclua os periféricos de entrada e as opções de entrada MAME que você está usando.
- A versão exata do MAME que você está usando, incluindo um *"git commit digest"* caso não seja uma versão final de lançamento, incluindo qualquer opção de compilação fora do padrão.
- A descrição exata do sistema e o software que está sendo emulado (pode não ser aplicável para problemas com partes da interface do usuário (IU) do MAME, como o menu de seleção do sistema). Inclua itens como a versão do BIOS selecionada e a configuração dos periféricos emulados (dispositivo de slot).
- Os passos para reproduzir o problema. Assuma que a pessoa que está lendo é familiar com o próprio MAME, mas não necessariamente familiarizado com o sistema emulado e o programa em questão. Para problemas de emulação, a gravação dos comandos e/ou arquivos de estado salvos para reproduzir o problema podem ser muito importante.
- Se possível, uma referência do original que indique o comportamento correto. Caso tenha acesso ao hardware original do sistema emulado, isso ajuda a fazer uma gravação para comparar o comportamento correto.

11.1.1 Contribuindo para o código-fonte do MAME

O MAME é desenvolvido em C++, porém, ele não é todo desenvolvido com essa linguagem. O código-fonte também inclui:

- A documentação hospedada neste site (e também disponibilizada nas versões finais do MAME como um PDF), escrito em formato [reStructuredText](#).
- Os *plug-ins*, oferecidos desenvolvidos em [Lua 5.3](#).
- Layouts internos para máquinas emuladas que precisam exibir mais que uma simples tela emulada como interfaces, botões, etc. Estes são aplicações em XML [descritos aqui](#).
- As listagens dos programas que descrevem os programas em mídia emulados pelo MAME. Essas listagens também estão em formato XML.
- A tradução da interface do usuário está em formato [GNU gettext PO](#). Eles podem ser editados com um bom editor de texto ou uma ferramenta dedicada como o [Poedit](#).

O nosso repositório principal está [hospedado no GitHub](#). Nós preferimos receber contribuições para o código-fonte em forma de [pull requests](#). Você precisará aprender o básico do git e se familiarizar com as ferramentas do git.

O processo básico para criar um "*pull request*" é o seguinte:

- Crie uma conta no GitHub.
- Faça um *fork* do repositório mamedev/mame.
- Crie uma nova ramificação do nosso master no repositório na sua conta.
- Faça a clonagem do *fork* do seu repositório bifurcado e verifique a sua nova ramificação.
- Faça as suas alterações, compile e faça os seus testes localmente.
- Aplique as suas alterações e faça um "*push*" na sua ramificação do GitHub.
- Opcionalmente, ative o "*GitHub Actions*" no seu repositório, permitindo que as suas alterações possam ser compiladas no Windows, no macOS e no Linux.
- Abra uma solicitação para mesclar as suas alterações no repositório master do mamedev/mame.

Lembre-se do seguinte (observe que nem todos os pontos são relevantes a todos os tipos de alterações):

- Detalhe bem o assunto e a mensagem do seu *commit*. Inclua o que a sua alteração faz e qual a sua finalidade. Uma pessoa que leia o registro log do *commit* não precisa ter que examinar todo o *diff* para ter uma ideia básica do que um *commit* deve fazer. As mensagens padrão do *commit* fornecidas pelo GitHub são totalmente inúteis, pois não oferecem quaisquer indicação do que uma alteração deveria fazer.
- Teste as suas alterações. Garanta que toda a compilação do MAME possa ser concluída sem erros e que o código que você alterou funciona. É uma boa ideia compilar com `DEBUG=1` para verificar se todas as asserções são compiladas e não disparadas.
- Utilize um título e uma descrição elucidativa. O título deve fornecer um resumo de uma linha do que a alteração geral afeta e o que ela faz. Já na descrição, ela deve ser bem mais detalhada. Não deixe a descrição em branco e descreva a alteração nos comentários, pois isso torna a busca e a filtragem muito mais difíceis.
- Esteja ciente que o "*GitHub Actions*" possui recursos limitados. Não fica claro quando você está perto ou atinge tais limites, nós tivemos colaboradores banidos do "*GitHub Actions*" por violar tais limites. Mesmo que você apele, eles ainda não vão te dizer quais são os limites reais, justificando que caso você saiba os limites reais, será possível criar meios para evitá-los. Caso ative o "*GitHub Actions*", considere não enviar *commits* individuais se não precisar que eles sejam automaticamente compilados ou executar o cancelamento do fluxo de trabalho quando não mais precisar dos resultados.

Temos diretrizes para partes específicas do código-fonte:

Diretriz de programação em C++

- *Introdução*
- *Definições*
- *Formato do arquivo de código-fonte*
- *Convenção de nomenclatura*
- *Variáveis e literais*
- *Contraventamento e indentação*
- *Espaçamento*
- *Escopo*
- *Const Correctness*
- *Comentários*
- *Auxiliares específicos do MAME*
- *Registrando*
- *Organização estrutural*

Introdução

Em termos de convenções de programação, o estilo presente num arquivo de código-fonte existente deve ser favorecido em relação aos padrões encontrados abaixo.

Quando um novo código-fonte está sendo criado, as seguintes convenções de programação devem ser observadas ao criar um novo arquivo dentro do núcleo do MAME (`src/emu` e `src/lib`). Caso o código-fonte esteja fora do núcleo, pode-se dar deferência ao estilo preferido do contribuidor, embora seja fortemente encorajado a programar entendendo que o arquivo pode precisar ser compreensível por outras pessoas com o passar do tempo.

Definições

Snake case

Tudo é escrito em minúsculas e os espaços são substituídos por sublinhados: `como_neste_exemplo`

Screaming snake case

Tudo é escrito em minúsculas e os espaços são substituídos por sublinhados: `COMO_NESTE_EXEMPLO`

Camel case

As palavras ou AS frases são escritas sem espaço onde o início de cada palavra começa com a letra em maiúsculas, menos a primeira: `comoNesteExemplo`

Llama case

As palavras as ou frases são escritas sem espaço entre as palavras e onde o início de cada palavra começa com a primeira letra em maiúsculas: `ComoNesteExemplo`

Formato do arquivo de código-fonte

Os arquivos C++ de código-fonte do MAME estão no formato texto UTF-8, assumindo os caracteres com largura fixa, com paradas de tabulação em intervalos com quatro espaços. Os arquivos de código-fonte devem terminar com um fim de linha. Qualquer texto "unicode" válido e imprimível é permitido nos comentários. Comentários e textos externos, são permitidos apenas o subconjunto "Unicode ASCII" imprimível.

A ferramenta `srcclean` é usada para impor regras de formato no arquivo do código-fonte antes de cada lançamento. É possível compilar essa ferramenta e aplicá-la aos arquivos que você alterar antes de abrir uma solicitação "pull" evitando posteriores conflitos ou alterações inesperadas.

Convenção de nomenclatura

Macros do pré-processador

Os nomes das macros devem usar o **screaming snake case**. As macros são sempre globais e os nomes conflitantes podem causar erros, pense com cuidado sobre o que as macros precisam ser nos cabeçalhos e as nomeie de acordo.

Include guards

A inclusão das *guard macros* devem começar com `MAME_` e devem terminar com um uma versão em maiúsculas do nome do arquivo, com espaços sendo substituídos por sublinhados.

Constantes

As constantes devem usar o **screaming snake case**, sejam elas constantes globais, membros de dados constantes, enumeradores ou pré-processadores constantes.

Funções

Os nomes de funções livres devem usar o **snake case**. Existem alguns utilitários funções que foram implementadas anteriormente como macros dos pré-processadores que ainda usam o **screaming snake case**.

Classes

Os nomes das classes devem usar um **snake case**. Os nomes de classes abstratas devem terminar com `_base`. Os membros de funções públicas (incluindo funções de membro estático) devem usar o **snake case**.

As Classes dos dispositivos

Os nomes específicos da implementação do `driver_device` convencionalmente termina com `_state`, enquanto a outra classe do nome do dispositivo específico termina com `_device`. Os nomes específicos do `device_interface` convencionalmente começam com `device_` e terminam com `_interface`.

Os tipos dos dispositivos

Os tipos dos dispositivos devem usar **screaming snake case**. Lembre-se que os tipos dos dispositivos são nomes dentro do namespace global, então escolha de forma explícita, nomes unívocos e diretos.

As enumerações

O nome da enumeração deve usar maiúsculas e minúsculas. Os enumeradores devem usar **screaming snake case**.

Os parâmetros usados como modelo

Os parâmetros usados como modelo devem usar maiúsculas e minúsculas (ambos os parâmetros de tipo e de valor).

Os identificadores que tenham dois sublinhados consecutivos ou que comece com um sublinhado seguido de uma letra maiúscula, estão sempre reservados e não podem ser usados.

Os nomes do tipo e dos outros identificadores com um sublinhado à esquerda, devem ser evitados no espaço de nomes globais (*namespace*), pois são reservados de forma explícita de acordo com o padrão C++. Além disso, os identificadores sufixados com `_t` devem ser evitados dentro do espaço de nomes globais, pois eles também são reservados de acordo com os padrões POSIX. Embora o MAME viole esta política ocasionalmente, principalmente com `device_t`, é considerado uma infeliz decisão herdada que deve ser evitada em todo e qualquer novo código.

Variáveis e literais

O uso de literais octais é desencorajado fora de casos bem específicos. Eles não possuem os prefixos óbvios com base em letras encontrados nas literais hexadecimais e nos binários, portanto, podem ser difíceis de distinguir rapidamente de um literal decimal para codificadores que não estão familiarizados com a notação octal.

É preferido que seja utilizado os literais hexadecimais em minúsculas, por exemplo, `0xbadc0de` em vez de `0xBADC0DE`. Para maior clareza, tente não exceder a largura de bits da variável que será utilizada para armazená-la.

Os literais binários raramente foram usados no código-fonte do MAME devido ao prefixo `0b` não ser padronizado até o C++14, mas não há nenhuma política para evitar a sua utilização.

A notação de sufixo inteiro deve ser usada ao especificar literais de 64 bits, mas não é estritamente necessária em outros casos. É possível, no entanto, rapidamente deixar claro o uso pretendido de um determinado literal. Os longos sufixos literais inteiros em maiúsculas devem ser utilizados para evitar confusão com o dígito 1, por exemplo `7LL` em vez de `711`.

O agrupamento dos dígitos deve ser usado para literais numéricos mais longos, pois ajuda a reconhecer a ordem de magnitude ou as posições do campo de bits mais rapidamente. Os literais decimais devem

usar grupos com três dígitos e os literais hexadecimais devem usar grupos com quatro dígitos, excluindo situações específicas onde diferentes agrupamentos seriam mais fáceis de entender, por exemplo 4'433'619 ou 0xffff8'1fff.

Os tipos que não possuam um tamanho especificamente definido, devem ser evitados caso sejam registrados no sistema "save-state" do MAME, pois isso prejudica a portabilidade. Em geral, isso significa evitar o uso de int para estes membros.

É recomendável, porém não obrigatório, que os membros dos dados da classe sejam prefixados com m_ nos membros com instância não estáticos e s_ para membros estáticos. Isso não se aplica as classes ou às estruturas aninhadas.

Contraventamento e indentação

As tabulações são usadas para o recuo inicial das linhas, com uma tabulação usada por nível do escopo agrupado. As declarações que forem divididas em várias linhas devem ser recuadas por duas tabulações. Os espaços são usados para alinhamento em outros lugares dentro de uma linha.

É preferível que a órtese seja no estilo **K&R** ou no estilo **Allman**. Não há uma preferência específica para os colchetes nas instruções com linha única, embora o colchete deva ser consistente num determinado bloco if/else, conforme é mostrado abaixo:

```
if (x == 0)
{
    return;
}
else
{
    call_some_function();
    x--;
}
```

Ao utilizar uma série de blocos if/else ou if/else if/else com comentários no recuo superior, evite novas linhas adicionais. O uso de novas linhas adicionais pode levar à perda dos blocos else if ou else devido às novas linhas empurrando os blocos para fora da altura visível do editor:

```
// O início do seu contador hipotético acabou.
if (x == 0)
{
    return;
}
// Devemos fazer algo se o contador estiver em execução.
else
{
    call_some_function();
    x--;
}
```

A indentação para as instruções case dentro de um corpo switch pode estar no mesmo nível que a instrução switch ou para dentro um nível. Não há um estilo específico que seja usado em todos os principais arquivos, embora o recuo num nível pareça ser usado com mais frequência.

Espaçamento

O espaçamento simples e consistente entre os operadores binários, as variáveis e os literais é veementemente recomendado. Os exemplos a seguir exibem um espaçamento razoavelmente consistente:

```
uint8_t foo = (((bar + baz) + 3) & 7) << 1;
uint8_t foo = ((bar << 1) + baz) & 0x0e;
uint8_t foo = bar ? baz : 5;
```

Os exemplos a seguir exibem extremos em qualquer direção, embora ter espaços adicionais seja menos difícil de ler do que ter poucos:

```
uint8_t foo = ( ( ( bar + baz ) + 3 ) & 7 ) << 1;
uint8_t foo = ((bar<<1)+baz)&0x0e;
uint8_t foo = (bar?baz:5);
```

Um espaço deve ser usado entre uma instrução C++ fundamental e o seu parêntese de abertura, por exemplo:

```
switch (value) ...
if (a != b) ...
for (int i = 0; i < foo; i++) ...
```

Escopo

O escopo das variáveis devem ser o mais restrito possível. Existem muitas declarações das instâncias da variável local no estilo C89 na base do código do MAME, mas isso é em grande parte um resquício dos primeiros dias do MAME, que antecede a especificação C99.

Os dois trechos a seguir mostram o estilo legado da declaração da variável local, seguido pelo estilo mais moderno e recomendado:

```
void dispositivo_exemplo::alguma_funcao()
{
    int i;
    uint8_t data;

    for (i = 0; i < std::size(m_buffer); i++)
    {
        data = m_buffer[i];
        if (data)
        {
            alguma_outra_funcao(data);
        }
    }
}
```

```
void dispositivo_exemplo::alguma_funcao()
{
    for (int i = 0; i < std::size(m_buffer); i++)
    {
        const uint8_t data = m_buffer[i];
```

(continua na próxima página)

(continuação da página anterior)

```
    if (data)
    {
        alguma_outra_funcao(data);
    }
}
```

Os valores enumerados, structs e as classes usadas apenas por um dispositivo específico, devem ser declarados dentro da própria classe do dispositivo. Isso evita a poluição do "namespace" global e torna o uso específico do dispositivo mais óbvio à primeira vista.

Const Correctness

A correção *const* não tem sido historicamente um requisito estrito do código que entra no MAME, mas há um valor crescente nisso à medida que a quantidade de refatoração do código aumenta e a dívida técnica diminui.

Ao escrever um novo código, vale a pena dedicar um tempo para determinar se uma variável local pode ser declarada como *const*. Da mesma forma, é recomendável considerar quais as funções do membro de uma nova classe podem ser qualificadas como *const*.

Assim como, as matrizes das constantes devem ser declaradas como *constexpr* e devem usar o *Screaming Snake Case*, conforme é descrito no início deste documento. Por fim, ambas as *arrays* das strings no estilo C devem ser declarados como *array const* das *strings const*, assim:

```
static const char *const NOMES_EXEMPLO[4] =
{
    "1-bit",
    "2-bit",
    "4-bit",
    "Invalid"
};
```

Comentários

Embora */* os comentários em ANSI C */* sejam frequentemente encontrados na base do código, houve uma alteração gradual para *// comentários no estilo C++* nos casos de comentários com única linha. Isso é basicamente uma diretriz e os programadores são encorajados a usar o estilo que for mais confortável.

A menos que citem especificamente o conteúdo de uma máquina ou materiais auxiliares, os comentários devem ser em inglês para corresponder ao idioma predominante que a equipe do MAME compartilha com todos ao redor do mundo.

O código comentado normalmente deve ser removido antes de criar um *pull request*, pois há uma tendência de ficar obsoleta devido à natureza de rápida movimentação da API principal do MAME. Se houver um desejo conhecido de antemão de que o código eventualmente seja incluído, ele deve ser marcado em *if (0)* ou *if (false)*, pois o código removido por meio de uma macro do pré-processador ficará obsoleta na mesma velocidade.

Auxiliares específicos do MAME

Sempre que possível, use funções auxiliares e macros para operações de manipulação dos bits.

O auxiliar `BIT(valor, bit)` pode ser usado para extrair o estado de um bit numa determinada posição de um valor inteiro. O valor resultante será alinhado à posição do bit de menor importância, ou seja, será 0 ou 1.

Uma sobrecarga da mesma função, `BIT(valor, bit, largura)` pode ser usada para extrair um bit do campo de uma determinada largura de um valor inteiro, começando na posição determinada do bit. O resultado também será justificado à direita e será do mesmo tipo que o valor da entrada.

Há, adicionalmente, uma série de auxiliares para funcionalidades como a contagem de zeros/uns à esquerda, para a contagem populada e para a multiplicação e a divisão dos números inteiros assinados/não assinados nos resultados de 32 bits e de 64 bits. Nem todos esses auxiliares têm amplo uso no código base do MAME, mas usá-los num novo código é altamente recomendável quando este código for crítico para questões de desempenho, pois eles utilizam montagem *"inline"* ou intrínsecos do compilador por plataforma, quando estiverem disponíveis.

`count_leading_zeros_32/64(T value)`

Aceita um valor não assinado com 32/64 bits e retorna um valor não assinado de 8 bits contendo a quantidade de zeros consecutivos a partir do bit mais importante.

`count_leading_ones_32/64(T value)`

Funcionalidade idêntica a da anterior, porém, examinando um bit consecutivo.

`population_count_32/64(T value)`

Aceita um valor com 32/64 bits não assinado e retorna a quantidade encontrada dos bits, ou seja, o peso *Hamming* do valor.

`rotl_32/64(T value, int shift)`

Executa um deslocamento circular/barril à esquerda de um valor não assinado com 32/64 bits usando um valor determinado de deslocamento. O valor do deslocamento será mascarado para o intervalo válido de bits para um valor com 32 ou com 64 bits.

`rotr_32/64(T value, int shift)`

Funcionalidade idêntica a da anterior, mas com o deslocamento à direita.

Para documentação sobre os auxiliares relacionados à multiplicação e divisão, consulte `src/osd/eminline.h`.

Registrando

O MAME possui diversas funções de registro para diferentes propósitos. Duas das funções de registro log utilizadas com mais frequência são o `logerror` e o `osd_printf_verbose`:

- Os dispositivos herdam uma função de membro `logerror`. Isso inclui automaticamente a *tag* totalmente qualificada do dispositivo que invoca as mensagens de registro. A saída é enviada para o registro log rotativo do *buffer* do depurador do MAME caso o depurador esteja ativado. Se a *opção* `-log` estiver ativada, ela também será registrada no arquivo `error.log` dentro do diretório de trabalho. Se a *opção* `-oslog` estiver ativada, ela também será enviada para a saída de diagnóstico do sistema operacional (o registro de diagnóstico do host do depurador do Windows, caso um host de depuração esteja conectado ou, caso contrário, usa o modo de erro padrão).
- A saída da função `osd_printf_verbose` é enviada para o modo de erro padrão caso a *opção* `-verbose` esteja ativada.

A função `osd_printf_verbose` deve ser usada para fazer o registro que é muito útil no diagnóstico de problemas do usuário, enquanto o `logerror` deve ser usado para mensagens mais relevantes aos desenvolvedores (durante o desenvolvendo do próprio MAME ou desenvolvendo programas para sistemas emulados usando o depurador do próprio MAME).

Para o registro da depuração, existe um sistema de registro com base em um canal através do cabeçalho `logmacro.h`. Ele pode ser usado como um sistema de registro genérico, sem a necessidade de usar a sua capacidade de mascarar canais específicos da seguinte maneira:

```
// Todos os outros cabeçalhos no arquivo .cpp devem estar acima desta linha.
#define VERBOSE (1)
#include "logmacro.h"
...
void some_device::some_reg_write(u8 data)
{
    LOG("%s: some_reg_write: %02x\n", machine().describe_context(), data);
}
```

O exemplo acima também faz uso de uma função auxiliar que está disponível em todas que sejam derivadas de `device_t`: `machine().describe_context()`. Esta função retornará uma *string* que descreve o contexto da emulação onde a função está sendo executada. Isso inclui a *tag* totalmente qualificada do dispositivo que está atualmente em execução (se houver). Caso o dispositivo relevante implemente um `device_state_interface`, ele também incluirá o valor do contador do programa atual relatado pelo dispositivo.

Para um controle mais refinado, as máscaras dos bits específicos podem ser definidos e usados através da macro `LOGMASKED`:

```
// Todos os outros cabeçalhos no arquivo .cpp devem estar acima desta linha.
#define LOG_F00 (1 << 1U)
#define LOG_BAR (1 << 2U)

#define VERBOSE (LOG_F00 | LOG_BAR)
#include "logmacro.h"
...
void some_device::some_reg_write(u8 data)
{
    LOGMASKED(LOG_F00, "some_reg_write: %02x\n", data);
}

void some_device::another_reg_write(u8 data)
```

(continua na próxima página)

(continuação da página anterior)

```
{  
    LOGMASKED(LOG_BAR, "another_reg_write: %02x\n", data);  
}
```

Observe que a posição do bit menos importante para as máscaras informadas pelo usuário é 1, pois a posição do bit 0 é reservada para o LOG_GENERAL.

É predefinido que LOG e o LOGMASKED usarão a função logerror fornecida pelo dispositivo. No entanto, isso pode ser redirecionado conforme seja preciso. O caso de uso mais comum seria direcionar a saída para a saída padrão, o que pode ser feito definindo explicitamente o LOG_OUTPUT_FUNC da seguinte maneira:

```
#define LOG_OUTPUT_FUNC osd_printf_info
```

Um desenvolvedor deve sempre garantir que a opção VERBOSE esteja definido como 0 e que qualquer definição de LOG_OUTPUT_FUNC seja comentada antes de abrir um "pull request".

Organização estrutural

Todos os arquivos de código-fonte C++ devem começar com dois comentários listando a licença de distribuição e os detentores dos direitos autorais num formato padronizado. As licenças são especificadas por seu identificador SPDX curto, caso esteja disponível. Abaixo um exemplo do formato padrão:

```
// license:BSD-3-Clause  
// copyright-holders:David Haywood, Tomasz Slanina
```

Os cabeçalhos incluídos geralmente devem ser agrupados do mais dependente ao menos dependente e classificados alfabeticamente dentro dos seus referidos grupos:

- O cabeçalho do prefixo do projeto, emu.h, deve ser a primeira coisa numa unidade de tradução.
- Cabeçalhos locais do projeto (cabeçalhos que estão junto com os arquivos de código-fonte por exemplo).
- Para os cabeçalhos em src/devices.
- Para os cabeçalhos em src/emu.
- Para os cabeçalhos em src/lib/util.
- Para os cabeçalhos da camada OSD.
- Para os cabeçalhos predefinidos da biblioteca C++.
- Para os cabeçalhos específicos do sistema operacional.
- Para os cabeçalhos layout.

Por fim, os cabeçalhos específicos da tarefa, como o logmacro.h descritos na seção anterior, eles devem ser incluídos por último. Abaixo segue um exemplo prático:

```
#include "emu.h"  
  
#include "cpu/m68000/m68000.h"  
#include "machine/mc68328.h"  
#include "machine/ram.h"  
#include "sound/dac.h"  
#include "video/mc68328lcd.h"
```

(continua na próxima página)

(continuação da página anterior)

```
#include "video/sed1375.h"

#include "emupal.h"
#include "screen.h"
#include "speaker.h"

#include "pilot1k.lh"

#define VERBOSE (0)
#include "logmacro.h"
```

Na maioria dos casos, a declaração da classe para um controlador do sistema, deve estar junto no arquivo do código-fonte correspondente da implementação. Nesses casos, a declaração da classe e todo o conteúdo do arquivo de código-fonte, menos a macro GAME, COMP ou CONS, devem ser colocados num *namespace* anônimo (isso produz melhores diagnósticos do compilador, permite uma otimização mais agressiva, reduz a chance de símbolos duplicados e também reduz o tempo de linkagem).

Dentro de uma declaração da classe, deve haver uma seção para cada nível de acesso do membro (public, protected e private) quando for possível. Isso pode não ser possível em casos onde as constantes e/ou os tipos privados precisam ser declarados antes dos membros públicos. Os membros devem usar o menor nível de acesso público necessário. As funções do membro virtual substituídas, geralmente devem usar o mesmo nível de acesso que a função do membro correspondente da classe base.

As declarações da classe dos membros devem ser agrupados para auxiliar na sua compreensão:

- Dentro de uma seção de nível de acesso dos membros, constantes, tipos, membros de dados, funções do membro da instância e funções estáticas do membro devem ser agrupados.
- Nas classes dos dispositivos, as funções de configuração do membro devem ser agrupadas separadamente das funções de sinal ativo do membro.
- As funções virtuais do membro que forem substituídas, devem ser agrupadas de acordo com as classes base das quais elas forem herdadas.

Para as classes sobrecarregadas com diversos construtores, sempre que possível, a delegação do construtor deve ser usada visando evitar listas repetidas dos inicializadores dos membros.

As constantes que são usadas por um controlador (*driver*) de dispositivo ou de uma máquina, devem estar na forma de valores enumerados com tamanho explícito dentro da declaração da classe ou ser relegados a macros #define dentro do arquivo de origem. Isso ajuda a evitar a poluição do pré-processador.

Diretrizes para os catálogos de programas

- [Introdução](#)
- [Itens e partes](#)
- [Metadados](#)

Introdução

Os catálogos de programas do MAME descrevem o programa em mídia conhecido para os sistemas emulados de uma maneira que pode ser usada para identificar arquivos conhecidos da imagem de um programa em mídia, para verificar a integridade da imagem do arquivo e para carregar os arquivos da imagem para serem emulados. Os catálogos de programas são implementadas como arquivos XML na pasta hash. A estrutura XML é descrita no arquivo hash/softwarelist.dtd.

Filosoficamente, os itens do catálogo de programa devem representar a mídia original, em vez de uma cópia específica da mídia. Idealmente, deve ser possível para qualquer pessoa com a mídia fazer um *dump* (cópia) dela e produzir uma imagem idêntica encontrada no catálogo. Claro, isso nem sempre é possível na prática, principalmente, é problemático em mídias inerentemente analógicas, como um programa de computador doméstico armazenado em fitas cassete.

O MAME se esforça para documentar as melhores imagens disponíveis das mídias. Não é nossa intenção propagar imagens corrompidas, truncadas, desfiguradas, com marcas d'água ou de outra forma, ruins. Sempre que possível, são preferidas as estruturas correspondentes do arquivo à estrutura da mídia original. Por exemplo, preferimos arquivos individuais para os chips ROM separados na mídia do cartucho e usamos as imagens do disco original em vez de arquivos extraídos dos mesmo discos.

Itens e partes

Um catálogo de programa é uma coleção de itens e cada item pode ter várias partes. Um item representa um pedaço do programa, distribuído como um pacote completo. Outra parte representa uma única peça de mídia dentro do pacote. As peças podem ser montadas individualmente nos dispositivos emulados da mídia. Por exemplo, um programa distribuído em três disquetes por exemplo, será listado como um único item, enquanto cada disquete será uma parte deste item.

Às vezes, partes logicamente separadas de uma única mídia física são representadas como partes separadas num item do programa. Por exemplo, cada lado de uma fita cassete é representado como uma parte separada. No entanto, os chips ROM individuais dentro de um cartucho, podem ser arquivos separados, mas não são partes separadas, pois o cartucho é montado como um todo.

Cada item é um elemento software. O elemento software pode ter os seguintes atributos:

name (obrigatório)

Um nome curto que identifica o item. É usado para os nomes dos arquivos, argumentos da linha de comando, chaves do banco de dados, fragmentos de URL dentre outros. Deve ser conciso, porém reconhecível. Deve ser o único dentro da lista de programa. Os caracteres válidos são letras minúsculas do inglês, dígitos decimais e sublinhados. O comprimento máximo permitido é de dezesseis caracteres.

cloneof (opcional)

O nome abreviado do item principal caso o item seja um clone. O principal deve estar no mesmo catálogo de programa, programas associados ao clone/principal espalhados em vários catálogos de programas não são suportados.

supported (opcional)

Um dos valores yes (totalmente utilizável na emulação), no (não utilizável na emulação) ou partial (utilizável na emulação com certas limitações). Caso o atributo não esteja presente, é o equivalente ao yes. Exemplos de programas com suporte parcial incluem jogos que podem ser reproduzidos com falhas gráficas e programas de escritório onde algumas, mas nem todas as suas funcionalidades, funcionam.

Cada part é um elemento interna do elemento software. O elemento part deve ter os seguintes atributos:

name (obrigatório)

Um nome abreviado que identifica a parte. Isso é usado para argumentos da linha de comando, chaves do banco de dados, fragmentos de URL dentre outros. Deve ser único dentro do item. Ele também é usado como o nome de exibição caso um nome de exibição separado não seja definido. Os caracteres válidos são letras minúsculas do inglês, dígitos decimais e sublinhados. O comprimento máximo permitido é de dezesseis caracteres.

interface (obrigatório)

Este atributo é usado para identificar a mídia dos dispositivos emulados que são adequados para montar uma parte do programa. Os valores aplicáveis dependem do sistema que está sendo emulado.

Metadados

Os catálogos de programas suportam vários tipos de metadados. Todos os itens do catálogo de programa requerem a presença dos seguintes elementos de metadados:

description

Este é o nome principal do item para exibição do programa. Deve ser o nome original do programa, transliterado para o alfabeto latino inglês, caso seja necessário. Deve ser único dentro do catálogo de programa. Caso o texto extra, além do próprio título, seja necessário para desambiguação, use letras minúsculas fora dos nomes próprios, inicialismos e citações literais.

year

O ano do lançamento ou o ano dos direitos autorais do programa. Caso seja desconhecido, use uma estimativa com um ponto de interrogação. Os itens podem ser filtrados por ano no menu da seleção do programa.

publisher

A produtora do programa. Isso pode ser o mesmo que o desenvolvedor caso o programa seja auto-publicado. Os itens podem ser filtrados pela publicação no menu da seleção do programa.

A maioria dos metadados dos itens visíveis do programa ao usuário é fornecida usando elementos `info`. Cada elemento `info` deve ter um atributo `name` e um atributo `value`. O atributo `name` identifica o tipo dos metadados e o atributo `value` é o próprio valor dos metadados. Observe que os atributos `name` não precisam ser exclusivos num item. Vários elementos `info` com o mesmo nome podem estar presentes, caso seja apropriado. Isso é visto com frequência em programas vendidos com títulos diferentes e em regiões diferentes.

O MAME exibe os metadados dos elementos `info` no menu de seleção do programa. Os seguintes atributos `name` são especificamente identificados e podem mostrar nomes localizados:

alt_title

Usado para títulos alternativos. Exemplos são diferentes blocos usados em diferentes idiomas, scripts, regiões ou diferentes títulos usados na tela de título e na embalagem. O MAME pesquisa os títulos alternativos, bem como a descrição.

author

O autor do programa. Os itens podem ser filtrados pelo autor no menu da seleção do programa.

barcode

O número do código de barras que identifica o pacote do programas (normalmente um EAN).

developer

o desenvolvedor responsável pela implementação do programa. Os itens podem ser filtrados pelo desenvolvedor no menu da seleção do programa.

distributor

Parte responsável pela distribuição do programa aos varejistas (ou os clientes no caso de vendas diretas). Os itens podem ser filtrados pelo distribuidor no menu da seleção do programa.

install

Instruções de instalação.

isbn

O código ISBN incluso num livro disponível comercialmente.

oem

O fabricante original do equipamento, normalmente usado com versões personalizadas do programa e distribuídas por um fornecedor de hardware.

original_publisher

A produtora original, para os itens que representam o programa que foi relançado por uma produtora diferente.

partno

O número da peça do distribuidor do programa.

pcb

O identificador da placa de circuito impresso, tipicamente usado para as mídias em cartucho.

programmer

O nome do programador que fez a programação do programa.

release

A data de lançamento detalhada do programa, caso seja conhecida. Use o formato AAAAMDD sem pontuação. Se apenas o mês for conhecido, use xx para os dígitos do dia, como 199103xx ou 19940729.

serial

O número que identifica o programa dentro de uma série de versões.

usage

Instruções de uso.

version

O número da versão do programa.

CAPÍTULO 12

DEFINIÇÕES TÉCNICAS

Esta seção aborda as definições técnicas úteis para os programadores que estão trabalhando no código-fonte do MAME ou trabalhando em scripts LUA que são executados na estrutura do MAME.

12.1 As convenções das nomenclaturas usadas pelo MAME

- *Introdução*
- *Transliteração*
- *Os títulos e as descrições*

12.1.1 Introdução

Visando promover a consistência e a legibilidade no código-fonte do MAME, utilizamos algumas convenções de nomenclatura em diversos elementos.

12.1.2 Transliteração

O script amplamente mais conhecido no mundo é o Alfabeto Latino que também é chamado de Alfabeto Romano. Convenientemente, também está incluído na codificação de quase todos os caracteres. Para tornar o MAME mais acessível globalmente, nós requisitamos que as transliterações dos títulos e dos outros metadados vindos a partir dos outros scripts sejam feitas com o Alfabeto Latino. Não use traduções nos metadados, as traduções são inerentemente sujeitas a erros e interpretações erradas. As traduções podem ser incluídas nos comentários caso sejam úteis.

Em geral, caso um nome oficial da escrita latina for conhecido, este deve ser utilizado em favor de uma simples transliteração. Para os títulos contendo estrangeirismos e palavras emprestadas de outros idiomas, a grafia latina convencional deve ser usada (um exemplo prático disso é usar "Mahjong" nos títulos japoneses em vez de "Maajan").

Chinês

Onde o público alvo fale Mandarim, o **Hanyu Pinyin** deve ser usado. Dentro dos contextos onde diacríticos não são permitidos (quando for limitado ao ASCII por exemplo), os números dos tons devem ser omitidos. Quando os tons estão sendo indicados por diacríticos, as regras do **Sandhi** de tom devem ser aplicados. Onde o público alvo fale Cantonês (principalmente Hong Kong e Guandong), o **Jyutping** deve ser usado com a omissão dos números de tom. Na dúvida, utilize o **Hanyu Pinyin**.

Grego

Utilize as regras **ISO: type (TR)**. Não utilize o inglês tradicional nas grafias para os nomes gregos (pessoas ou lugares).

Japonês

Em geral as regras alteradas **Hepburn** devem ser utilizadas. Use um apóstrofo entre o **N** silábico e uma vogal seguinte (incluindo as vogais iotizadas). Não use hífens para transliterar vogais prolongadas.

Coreano

Para os sobrenomes Coreanos utilize as regras revisadas da romanização Coreana (RR) com as tradicionais da ortografia Inglesa. Não use regras ALA LC para a divisão das palavras e uso de hífens.

Vietnamita

Quando os diacríticos não podem ser utilizados, omita os tons e substitua as vogais com simples vogais em Inglês, não utilize as convenções **VIQR** ou **TELEX** ("um chuot nuong" em vez de "a(n chuo^.t nu*o*'ng" ou "awn chuootj nuowngs" por exemplo).

12.1.3 Os títulos e as descrições

Tente reproduzir o título original de forma mais fiel possível sempre que houver a possibilidade. Tente preservar a convenção das maiúsculas e das minúsculas utilizada pelo fabricante ou distribuidor. Caso nenhum título oficial em Alfabeto Latino seja conhecido, use uma transliteração padrão. Para as entradas na lista de software onde uma transliteração é usada para a descrição do elemento (atributo `description`), coloque o título num elemento de informação (atributo `info`) com um atributo `name="alt_title"`.

Para os itens de software que possuam diferentes títulos (diferentes títulos separados por região com a mesma mídia de instalação por exemplo), utilize um título em Alfabeto Latino para o elemento da descrição (atributo `description`) e coloque os outros títulos nos elementos de informação (atributo `info`) com atributos `name = "alt_title"`.

Caso uma desambiguação seja necessária, tente ser o mais descritivo possível. Use o número da versão do fabricante, o nome do licenciado regional ou faça uma descrição concisa entre as diferenças do hardware com preferência aos números dos conjuntos arbitrários por exemplo. Cerque o texto de desambiguação entre parênteses, preserve a caixa original para os nomes e o texto da versão, porém use minúsculas para qualquer outra coisa exceto os nomes próprios e os acrônimos.

Para as convenções das nomenclaturas C++, consulte a seção da diretriz de programação em C++ [Convenção de nomenclatura](#).

12.2 Os arquivos layout

- *Introdução*
- *Conceitos básicos*
 - *Números*
 - *Coordenadas*
 - *Cores*
 - *Parâmetros*
 - *Parâmetros já predefinidos*
- *As partes de um layout*
 - *Os elementos*
 - *As visualizações*
 - *Coleções*
 - *Grupos reutilizáveis*
 - *Repetindo os blocos*
- *Interatividade*
 - *Itens que podem ser clicados*
 - *O Estado do elemento*
 - *A visualização de um item animado*
- *Lidando com erros*
- *As visualizações que são geradas automaticamente*
- *Usando o complay.py*
- *Criando diferentes arquivos layout na prática*
 - *Ferramentas necessárias*
 - *Identificando as partes*
 - *A Posição e a proporção de tela*
 - *Posicionando os botões na tela*
 - *Conectando os botões e ativando as suas funções lógicas*
 - *Alertas e retornos externos*
 - *Colocando para funcionar e adicionando efeitos na tela*
 - *Adicionando diagonais nos controles com 8 direções usando o inputraw*
 - *Sistemas com dois ou mais controles*
 - *Um único arquivo para diferentes visualizações*
 - *Desativando objetos na tela*
 - *Criando as posições de um controle arcade*

- * *Exportando arquivos SVG*
- * *Exportando arquivos PNG*
- *Trabalhando com grupos*
 - * *Usando os botões gerados pelo MAME e os arquivos SVG*
 - * *Simplificando o layout*
 - * *Usando imagens PNG para o mesmo tipo de layout*
 - * *Simplificando o layout novamente*
- *Dividindo a tela com um controle ou outro tipo de informação*
- *Exemplos de outros arquivos layout*

12.2.1 Introdução

Os arquivos layout¹ são usados para informar ao MAME o que exibir enquanto a emulação de um sistema estiver rodando e também como organizá-los na tela. O MAME pode renderizar a emulação das telas originais dos sistemas, as imagens, os textos, as formas e objetos especiais na saída comum dos dispositivos. Os elementos podem ser estáticos ou se atualizar de forma dinâmica para refletir a condição das entradas e das saídas. Os layouts podem ser gerados automaticamente com base no número ou no tipo da tela que será emulada, ser construído e conectado internamente ao binário do MAME ou sendo disponibilizado externamente. Para o MAME os arquivos layout são interpretados como arquivos XML porém utilizam a extensão .lay.

¹ Arquivos que organizam a disposição dos elementos gráficos na tela. (Nota do tradutor)

12.2.2 Conceitos básicos

Números

Os layouts do MAME possuem dois tipos de números, números inteiros e os de ponto flutuante.

Os números inteiros podem ser usados com notação decimal ou hexadecimal. Um número decimal inteiro consiste num prefixo opcional # (hash²), um caractere opcional +/- (mais ou menos) ou uma sequência de dígitos entre **0-9**.

Um número hexadecimal consiste em que um dos prefixos seja o \$ (cifrão) ou **0x** (zero xis) seguido por uma sequência de números decimais entre **0-9** em conjunto com as letras entre **A-F**. Não há diferenciação entre as letras maiúsculas e as letras minúsculas no índice e nos dígitos dos números hexadecimais.

Os números de ponto flutuante podem ser usados com decimal de ponto fixo ou com notação científica. Observe que os prefixos do número inteiro e os valores hexadecimais *não* são aceitos caso um número de ponto flutuante seja esperado.

São permitidos como atributos ambos os números inteiros e os números de ponto flutuante. Nesses casos a presença de um prefixo # (hash), \$ (cifrão) ou **0x** (zero xis) faz com que o valor seja interpretado como um número inteiro. Caso nenhum prefixo de número inteiro, o ponto decimal ou a letra E (maiúsculo ou minúsculo) seja encontrado na introdução de um expoente, este será interpretado como um número de ponto flutuante. Caso nenhum prefixo de número inteiro, ponto decimal ou a letra E seja encontrado, o número será interpretado como um número inteiro.

Os números são analisados usando uma acentuação de caracteres em C³ por questões de portabilidade.

² Em nosso idioma conhecido também como cerquilha, jogo da velha, sustenido e atualmente como **hashtag**. (Nota do tradutor)

³ *C locale* em Inglês. (Nota do tradutor)

Coordenadas

As coordenadas do layout são representadas internamente através da norma IEEE754 como um número binário de 32-bit de ponto flutuante (também conhecido como "*precisão simples*"). O incremento das coordenadas se dão nas direções da direita e para baixo. A origem **(0,0)** não possui um significado em particular e valores negativos podem ser usados.

O MAME pressupõe que as coordenadas da visualização possuem a mesma proporção de aspecto com relação aos pixels gerados pelo dispositivo (janela ou nativa). Considerando que sejam pixels quadrados e sem rotação, isso significa que a distância seja igual nos eixos **X** e **Y** o que corresponde a distâncias iguais na vertical e na horizontal que for gerado pela renderização.

Todos os elementos, os grupos e as visualizações possuem os seus sistemas internos de coordenadas. Quando um elemento ou um grupo é referenciado a partir de uma visualização ou de um outro grupo, as suas coordenadas são dimensionadas de acordo com a necessidade para que os limites sejam definidos.

Os objetos são posicionados e dimensionados através do elemento `bounds` que define os seus limites e também as suas fronteiras. A posição horizontal e o seu tamanho podem ser definidos de três maneiras:

- A borda esquerda e a largura usando atributos `x` e `width`.
- O eixo horizontal centralizado onde **c** significa que a referência usada será o **centro** do objeto/imagem e a largura usando atributos `xc` e `width`.
- As bordas esquerda e direita usando atributos `left` e `right`.
- De maneira semelhante a posição vertical e o seu tamanho podem ser definidos através da borda superior e a altura usando atributos `y` e `height`.
- O eixo vertical centralizado e a altura usando atributos `yc` e `height`.
- As bordas superiores e inferiores usando atributos `top` e `bottom`.

No exemplo abaixo estes três elementos `bounds` são equivalentes:

```
<bounds x="455" y="120" width="12" height="8" />
<bounds xc="461" yc="124" width="12" height="8" />
<bounds left="455" top="120" right="467" bottom="128" />
```

É possível utilizar diferentes esquemas nas direções horizontal e vertical. Por exemplo, estes elementos `bounds` equivalentes também são válidos:

```
<bounds x="455" top="120" width="12" bottom="128" />
<bounds left="455" yc="124" right="467" height="8" />
```

Caso nenhum valor seja informado, é predefinido que seja **1.0** para o atributo `width/height` ou `right/bottom`. O MAME irá considerar como um erro caso os atributos `width` ou `height` tenham valores negativos, `right` tenha um valor menor que `left` ou caso `bottom` tenha um valor menor que `top`.

Cores

As cores são definidas no espaço RGBA. O MAME não trabalha com todo o leque da gama de cores, portanto, as cores serão interpretadas como sRGB em conjunto da definição do gamma do seu sistema que geralmente é 2.2. Os valores dos canais são definidos como números de ponto flutuante. Os valores dos canais vermelho, verde e azul variam entre 0.0 (desligado) até 1.0 (intensidade plena). Os valores alfa variam entre 0.0 (transparência absoluta) até 1.0 (opaco). Os valores dos canais das cores não são previamente multiplicados pelo valor alfa.

O componente e a cor do item da visualização são definidas através dos elementos color. Os atributos relevantes são vermelho red, verde green, azul blue e alpha. Este exemplo do elemento color determina todos os valores dos canais:

```
<color red="0.85" green="0.4" blue="0.3" alpha="1.0" />
```

Qualquer atributo omitido do canal terá o seu valor predefinido para 1.0 (intensidade absoluta ou opaca). Será considerado como um erro caso os valores do canal estejam fora do intervalo entre de 0.0 até 1.0.

Nem toda a ferramenta de edição de imagens trabalhe com o mesmo sistema que o MAME, assim sendo, utilize [esta calculadora](#) para converter um valor RGB hexadecimal usado em HTML ou RGB por exemplo, para o formato que o MAME aceita.

Parâmetros

Os parâmetros funcionam como variáveis que podem ser utilizadas para substituir o valor dos atributos, basta cercar o seu nome com caracteres til (~). Nenhuma substituição será feita caso nenhum parâmetro seja definido. No exemplo abaixo é possível ver como os valores dos parâmetros digitno e do x substituirão o ~digitno~ e o ~x~:

```
<repeat count="8">
  <param name="digitno" start="1" increment="1" />
  <param name="x" start="0" increment="114" />
  <element name="digit~digitno~" ref="digit">
    <bounds x=~x~ y="80" width="25" height="40" />
  </element>
```

Um nome para o parâmetro é uma sequência de letras maiúsculas das letras **A-Z**, das letras minúsculas **a-z**, dígitos decimais **0-9**, ou caracteres subtraço (**_**). As letras maiúsculas e as letras minúsculas são levadas em consideração nos nomes dos parâmetros. Durante a procura por um parâmetro o motor do layout começando a trabalhar da parte mais interna do escopo atual até a sua parte mais externa. O nível mais periférico do escopo corresponde ao elemento do primeiro nível mamelayout. Cada elemento repeat, group ou view cria um novo nível de encadeamento do escopo.

Internamente um parâmetro pode conter uma string, números inteiros ou números de ponto flutuante, porém esta é bem mais óbvia. Os números inteiros são armazenados como *64-bit signed* com dois valores complementares, já os números de ponto flutuante são armazenados como binários *IEEE754* com *64-bit*, estes números também são conhecido como "precisão dupla". Os números inteiros são substituídos em notação decimal, já os números de ponto flutuante são substituídos pelo seu formato padrão que pode ser um decimal de ponto fixo ou dependendo do valor pode ser uma notação científica. Não há nenhuma maneira de substituir a formatação predefinida dos parâmetros de um número inteiro ou de ponto flutuante.

Existem dois tipos de parâmetros: os *valores* e os *geradores*. O parâmetro "value" mantém o seu valor atribuído até que eles sejam alterados, já o parâmetro "gerador" possui um valor inicial, um incremento e/ou um deslocamento⁴ aplicado em cada interação.

⁴ O termo *shift* é muito amplo, também pode ser interpretado como desvio, mudança, turno, inversão, câmbio, etc. (Nota do

Os valores dos parâmetros são atribuídos através do elemento `param` junto com os elementos `name` e `value`, os seus valores podem aparecer de dentro de um elemento de primeiro nível `mamelayou` e dentro dos elementos `repeat`, `view` assim como dentro da definição dos elementos `group` (isso é, elementos `group` dentro do nível superior do elemento `mamelayou`, ao contrário dos elementos `group` dentro de elementos `view` definidos por outros elementos `group`). O valor do parâmetro pode ser reatribuído a qualquer momento.

Aqui está um exemplo de como atribuir o valor "4" para o parâmetro "firstdigit":

```
<param name="firstdigit" value="4" />
```

Os parâmetros dos geradores são atribuídos através do elemento `param` em conjunto com os atributos `name`, `start`, `increment`, `lshift` e `rshift`. Os parâmetros dos geradores só podem aparecer de dentro dos elementos `repeat` (consulte [Repetindo os blocos](#) para obter mais informações) e também não devem ser reatribuídos dentro do mesmo escopo (um parâmetro com um nome idêntico pode ser atribuído num escopo através da sua ramificação). Abaixo alguns parâmetros de exemplos dos geradores:

```
<param name="nybble" start="3" increment="-1" />
<param name="switchpos" start="74" increment="156" />
<param name="mask" start="0x0800" rshift="4" />
```

- O parâmetro `nybble` geram os valores 3, 2, 1...
- O parâmetro `switchpos` geram os valores 74 (74), 230 (74 + 156), 386 (230 + 156)...
- O parâmetro `mask` geram os valores 2048 (0x0800), 128 (0x0800 >> 4), 8 (0x80 >> 4)...

O atributo `increment` deve ser um número inteiro ou de ponto flutuante que será adicionado ao valor do parâmetro. Os atributos `lshift` e `rshift` devem ser números positivos e inteiros pois definem a quantidade dos bits que serão aplicados aos parâmetros. O deslocamento (`shift`) e o incremento são aplicados no final do bloco que está sendo repetido antes do início da próxima iteração. O valor do parâmetro poderá ser interpretado como um número de ponto flutuante ou um número inteiro antes que o incremento ou o deslocamento seja aplicado. Caso informe ambos os valores para incremento e para o deslocamento, então o valor do incremento será aplicado primeiro e depois o valor deslocado.

Caso o atributo `increment` esteja presente e seja um número de ponto flutuante, o seu valor será convertido para um número de ponto flutuante caso seja necessário antes que o incremento seja adicionado. Caso o atributo `increment` esteja presente e seja um valor inteiro enquanto o valor do parâmetro seja um número de ponto flutuante, o valor do incremento será convertido para um número de ponto flutuante antes que o valor seja adicionado.

Caso os atributos `lshift` ou `rshift` estejam presentes porém não sejam iguais, o valor do parâmetro será convertido para um número inteiro e deslocado conforme a necessidade. O deslocamento para a esquerda é definido como um deslocamento feito em direção ao bit de maior importância. Caso ambos os parâmetros `lshift` e `rshift` sejam passados, estes serão compensados antes dos valores serem aplicados. Significa que não é possível usar atributos iguais tanto para o `lshift` como para o `rshift` por exemplo para limpar os bits num valor do final do parâmetro após a primeira iteração.

Será considerado um erro caso o elemento `param` não esteja em qualquer um dos atributos `value` ou `start`, será também considerado um erro caso ambos os elementos `param` tiverem os mesmos atributos `value` ou qualquer um dos mesmos atributos `start`, `increment`, `lshift`, ou `rshift`.

Um elemento `param` define ou reatribui o seu valor num parâmetro no escopo atual mais interno. Não é possível definir ou reatribuir os parâmetros num escopo de contenção.

Parâmetros já predefinidos

Uma certa quantidade de valores predefinidos nos parâmetros já estão disponíveis e fornecem informações sobre o sistema que está em execução:

devicetag

Um exemplo do caminho completo da etiqueta⁹ dispositivo que será responsável pela leitura do layout, seria : para o driver do controlador do dispositivo raiz ou :tty:ie15 para o terminal conectado numa porta. Este parâmetro é uma sequência de caracteres definida no escopo global de visualização do layout.

devicebasetag

A base da etiqueta do dispositivo que será responsável pela leitura do layout, como por exemplo root para o driver do dispositivo raiz ou ie15 para o terminal que estiver conectado numa porta. Este parâmetro é uma sequência de caracteres definida no escopo global do layout.

devicename

O nome completo (descrição) do dispositivo que será responsável pela leitura do layout, como por exemplo os terminais AIM-65/40 ou IE15. Este parâmetro é uma sequência de caracteres definida no escopo global do layout.

deviceshortname

Um nome curto do dispositivo que será responsável pela leitura do layout, como por exemplo os terminais aim65_40 ou ie15. Este parâmetro é uma sequência de caracteres definida no escopo global do layout.

scr0physicalxaspect

A parte horizontal da relação de aspecto físico da primeira tela (caso esteja presente). A relação de aspecto físico é fornecida como uma fração impropriamente reduzida. Observe que este é o componente horizontal aplicado *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

scr0physicalyaspect

A parte vertical da relação de aspecto físico da primeira tela (caso esteja presente). A relação de aspecto físico é fornecida como uma fração impropriamente reduzida. Observe que este é o componente vertical aplicado *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

scr0nativexaspect

A parte horizontal da relação de aspecto do pixel visível na região da primeira tela (caso esteja presente). A relação de aspecto do pixel é fornecida como uma fração impropriamente reduzida. Observe que este é o componente horizontal aplicado *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

scr0nativeyaspect

A parte vertical da relação de aspecto do pixel visível na região da primeira tela (caso esteja presente). A relação de aspecto do pixel é fornecida como uma fração impropriamente reduzida. Observe que este é o componente vertical aplicado *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

⁹ Tag no Inglês.

scr0width

A largura da região visível da primeira tela (se houver) nos pixels emulados. Observe que a largura é aplicada *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

scr0height

A altura da região visível da primeira tela (se houver) nos pixels emulados. Observe que a altura é aplicada *antes* da rotação. Este parâmetro é um número inteiro definido no escopo global do layout.

scr1physicalxaspect

A parte horizontal da relação de aspecto físico da primeira tela (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr1physicalyaspect

A parte vertical da relação de aspecto físico da segunda tela (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr1nativexaspect

A parte horizontal da relação de aspecto do pixel visível na região da segunda tela (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global de visualização do layout.

scr1nativeyaspect

A parte vertical da relação de aspecto do pixel visível na região da segunda tela (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global de visualização do layout.

scr1width

A largura da região visível da segunda tela (se houver) nos pixels emulados. Este parâmetro é um número inteiro definido no escopo global do layout.

scr1height

A altura da região visível da segunda tela (se houver) nos pixels emulados. Este parâmetro é um número inteiro definido no escopo global do layout.

scr*N*physicalxaspect

A parte horizontal da relação de aspecto físico da tela (base-zero) *N*th (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr*N*physicalyaspect

A parte vertical da relação de aspecto físico da tela (base-zero) *N*th (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr*N*nativexaspect

A parte horizontal da relação de aspecto da parte visível da tela (base-zero) *N*th (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr*N*nativeyaspect

A parte vertical da relação de aspecto da parte visível da tela (base-zero) *N*th (caso esteja presente). Este parâmetro é um número inteiro definido no escopo global do layout.

scr*N*width

A largura da região visível da tela (base-zero) Nth (se presente) nos pixels emulados. Este parâmetro é um número inteiro definido no escopo da visualização do layout.

scr*N*height

A largura da região visível da tela (base-zero) Nth (se presente) nos pixels emulados. Este parâmetro é um número inteiro definido no escopo de visualização do layout.

viewname

O nome da visualização atual. Este parâmetro é uma sequências de caracteres definido no escopo de visualização. Não é definido fora do campo de visão.

Para parâmetros relacionados à tela, elas são numeradas do zero na ordem em que aparecem na configuração do sistema. Todas as telas estão inclusas (não apenas nos sub-dispositivos do dispositivo que fizeram com que o layout fosse carregado). **X/width** e **Y/height** referem-se as dimensões horizontal e vertical da tela *antes* da rotação ser aplicada. Os valores baseados na região visível são calculados no final da configuração. Caso o sistema não reconfigure a tela durante a execução os valores dos parâmetros não serão atualizados assim como os layouts não serão recalculados.

12.2.3 As partes de um layout

Uma visualização define a disposição de um objeto gráfico a ser exibido. O arquivo layout do MAME pode conter diversas visualizações. As visualizações são construídas a partir de elementos *elements* e telas *screens*. Para simplificar a organização dos layouts complexos são compatíveis entre si a repetição dos blocos e dos grupos que podem ser reutilizados.

O primeiro elemento do cabeçalho de um arquivo layout do MAME deve ser um elemento chamado `mamelayou`t junto com um atributo `version`. O atributo `version` deve ser um valor inteiro. Atualmente, o MAME suporta apenas a versão 2 e não carregará qualquer outra versão diferente. Este é um exemplo de uma tag inicial para um elemento `mamelayou`t:

```
<mamelayou version="2">
```

Para fins de compatibilidade na identificação do arquivo com diversos editores de texto é possível declarar que o mesmo é um arquivo XML, logo, o MAME também aceita o arquivo com uma declaração XML:

```
<?xml version="1.0"?>
<mamelayou version="2">
```

Da mesma maneira que é possível usar o identificador XML, também é possível identificar a codificação do arquivo caso seja necessário:

```
<?xml version="1.0" encoding="UTF-8"?>
<mamelayou version="2">
```

Os comentários podem ser adicionados em qualquer parte do arquivo desde que estejam entre `<!--` e `-->`:

```
<?xml version="1.0"?>
<!-- Este é um comentário -->

<!--
Este tipo de comentário também é válido.
-->

<!--
    Também é possível incluir longas instruções ou informações
    relevantes no seu arquivo layout para que as pessoas saibam
    o que fazer ou como prosseguir caso seja necessário.
    Identifique os seus arquivos, utilize estes espaços para deixar
    o seu nome ou apelido, a versão, a data que o layout foi criado
    ou que o arquivo foi alterado, a descrição das alterações, os
    direitos autorais, etc.

    The Alpha Betas and the Lambda Lambda Lambda Fraternity
    Versão: 1.0
    Criado em: 10/11/2020
    Licença: CC by 4.0
-->

<mamelayou version="2">
```

Algumas regras devem ser observadas ao adicionar os comentários:

- Os comentários não devem aparecer antes da declaração XML.
- Os comentários não devem aparecer dentro da etiqueta de um elemento.
- Os comentários não devem aparecer dentro do valor de um atributo.
- Os comentários não pode ter a sequência de caracteres --.

Em geral, as ramificações do primeiro elemento `mamelayou` são processados na ordem em que eles chegam, de cima para baixo, exceto as visualizações que são processadas por último. Isso significa que as visualizações veem os valores finais de todos os parâmetros no final do elemento `mamelayou` e pode se referir a elementos e grupos que possam aparecer depois deles.

Os seguintes elementos são permitidos dentro do primeiro elemento `mamelayou`:

param

Define ou reatribui um valor ao parâmetro. Consulte [Parâmetros](#) para mais informações.

element

Define um elemento, um dos objetos primários a serem organizados numa Visualização. Consulte [Os elementos](#) para obter mais informações.

group

Define um grupo dos elementos ou das telas que possam ser reutilizáveis e que também possam ser usados como referência numa visualização ou nos outros grupos.

Consulte [Grupos reutilizáveis](#) para obter mais informações.

repeat

Um grupo de elementos repetidos que podem conter os elementos `param`, `element`, `group` e `repeat`. Consulte [Repetindo os blocos](#) para obter mais informações.

view

Um arranjo dos elementos ou das telas que podem ser exibidos na saída de um dispositivo (uma janela ou uma tela do host). Consulte [As visualizações](#) para obter mais informações.

script

Permite que scripts lua sejam usados num layout aprimorado ainda mais a interação.

Os elementos

Os elementos são um dos objetos visuais mais básicos que podem ser organizados em conjunto com as telas na composição de uma visualização. Os elementos podem ser construídos com um ou mais componentes porém um elemento é tratado como uma única superfície na composição do gráfico da cena e da sua renderização. Um elemento pode ser usado em diversas visualizações e pode também serem utilizadas várias vezes dentro da visualização.

A aparência de um elemento depende do seu *estado*. O estado é um valor inteiro que geralmente vem de uma região da porta E/S ou da emulação gerada (consulte [O Estado do elemento](#) para obter mais informações de como conectar um elemento numa porta ou na saída E/S de uma emulação). Qualquer componente de um elemento pode estar restrito apenas ao desenho quando o estado do elemento tiver um valor em particular. Alguns componentes (como os mostradores com múltiplos segmentos por exemplo) que usam diretamente o seu estado para determinar a sua aparência final.

Cada elemento possui o seu próprio sistema interno de coordenadas. Os limites dos elementos dos sistema de coordenadas são computados através da união dos limites individuais dos componentes que ele é composto.

Todo elemento deve ter o seu nome definido através do atributo `name`. Os elementos são mencionados através do nome quando forem solicitados nos grupos ou nas visualizações. Haverá um erro caso o arquivo de layout tenha vários elementos `name` com valores iguais. Os elementos podem de forma opcional, ser utilizado para informar um valor padrão do seu estado através do atributo `defstate` caso esteja conectado numa saída emulada ou numa porta E/S. O valor do atributo `defstate` deve possuir um valor inteiro e positivo, os valores negativos geram erros e fazem com que o layout não seja mais carregado.

As ramificações do elemento `element` instanciam componentes que são desenhados na textura do elemento na ordem de leitura a partir do primeiro ao último elemento utilizando alpha blending (os componente são desenhados por cima e podem se sobrepor aos componentes que venham antes dele). Todos os componentes são compatíveis com algumas características em comum:

- Os componentes podem ser desenhados de forma condicional dependendo da condição do elemento ao informar os atributos `state` ou `statemask`. Caso estejam presentes, estes atributos devem ser inteiros com valores positivos. Caso apenas o atributo `state` esteja presente, então o componente só será desenhado na tela quando o elemento `state` coincidir com o seu valor. Caso apenas o atributo `statemask` esteja presente, então o componente só será desenhado na tela caso todos os bits estejam definidos e os seus valores estejam definidos através do atributo `state`.

Na existência de ambos os atributos `state` e `statemask`, então o componente só será desenhado na tela quando os bits no elemento `state` corresponderem ao bit que estiver definido no atributo `statemask` e também corresponder com os bits do valor do atributo `state`.

O componente sempre será desenhado na ausência de ambos os atributos `state` ou `statemask` ou caso o valor do atributo `statemask` for zero.

- Cada componente pode ter um sub-elemento `bounds` definindo a sua posição e o seu tamanho (consulte [Coordenadas](#)). Na ausência de tal elemento os limites serão predefinidos a uma unidade quadrada com o valor igual à **1.0** tanto para a largura quanto para a altura e com o canto superior esquerdo com valor **0.0**.

A posição ou o tamanho de um componente pode ser animado de acordo com o estado do elemento ao prover diversos elementos `bounds` em conjunto com atributos `state`. O atributo `state` de cada ramificação do elemento `bounds` deve ser um número inteiro e positivo. Os atributos `state` não devem ser iguais para quaisquer um dos dois elementos `bounds` que estiverem dentro de um componente.

Caso o estado do elemento seja inferior que o valor do atributo `state` de qualquer uma das ramificações do elemento `bounds`, será utilizada a posição/tamanho definido através do elemento `bounds` com o menor valor do atributo `state`. Já quando o estado do elemento for maior que o valor do atributo `state` de qualquer elemento `bounds`, será utilizada a posição/tamanho especificado através do elemento `bounds` com o maior valor do atributo `state`. Se o estado do elemento estiver entre os valores do atributo `state` dos dois elementos `bounds`, a posição/tamanho será interpolada de forma linear.

- Cada componente de cor pode ter um elemento `color` definindo uma cor RGBA (Consulte [Cores](#) para obter mais informações). Isto pode ser usado para controlar a geometria da cor dos componentes desenhados de forma algorítmica ou textual. Para os componentes `image`, a cor dos pixels da imagem são multiplicadas através da cor que foi definida. Caso tal elemento não esteja presente, será usada uma cor branca opaca já predefinida.

A cor do componente pode ser animada de acordo com o estado do elemento ao prover diversos elementos `color` em conjunto com os atributos `state`. Os atributos `state` não devem ser iguais em qualquer um dos dois elementos `color` internos de um componente.

Caso o estado do elemento seja inferior ao valor do atributo `state` de qualquer elemento `color`, será utilizada a cor especificada através do elemento `color` com o menor valor do atributo `state`.

Caso o estado do elemento seja superior ao valor do atributo `state` de qualquer elemento `color`, será utilizada a cor especificada através do elemento `color` com o maior valor do atributo `state`. Caso o estado do elemento estiver entre os valores do atributo `state` de dois elementos `color`, os componentes de cor RGBA serão interpolados de forma linear.

Há suporte para os seguintes componentes:

rect

Desenha um retângulo colorido uniforme com as suas bordas preenchidas.

disk

Desenha uma elipse (círculo) colorido e uniforme.

image

Exibe uma imagem na tela a partir de um arquivo PNG, JPEG, Window DIB (BMP) ou um arquivo SVG. O nome do arquivo que será carregado (incluindo a sua extensão) é definido usando o atributo `file`. Adicionalmente, é possível utilizar um atributo opcional chamado `alphafile` para determinar o nome de um arquivo PNG (incluindo a sua extensão) para ser carregada dentro do canal alfa.

Alternativamente, os dados da imagem podem ser informados no próprio arquivo layout utilizando um sub-elemento `data`. Isto pode ser útil para oferecer gráficos SVG simples e legíveis. Será considerado como um erro caso nenhum atributo `file` ou `data` seja informado.

O arquivo usado como `alphafile` deve ter as mesmas dimensões (em pixels) que o arquivo do atributo `file` e a sua profundidade de bits por pixel da imagem não deve ser maior que 8 bits por canal. A intensidade de `brightness` desta imagem é copiada para o canal alfa com total intensidade (branco em escala de cinza) o que corresponde a um opaco completo e o preto uma total transparência.

O atributo `alphafile` será ignorado caso o atributo `file` aponte para um arquivo SVG ou um sub-elemento `data` contendo dados SVG, o atributo é apenas utilizado com imagens do tipo bitmap.

O(s) arquivo(s) da(s) imagem(s) devem ser colocados no mesmo diretório que o arquivo layout. Os formatos da imagem são detectados durante a análise do conteúdo dos arquivos, os nomes das extensões dos arquivos não são levados em consideração. Note porém que nos sistemas `*nix` o nome dos arquivos com maiúsculas e com minúsculas são levadas em consideração quando não estiverem dentro de um arquivo `.zip` ou `.7z`.

É possível identificar quando o MAME não conseguir carregar as imagens pois aparecem uma sequência de pequenas bolinhas cinzas na tela, isso mostra que ou o MAME não encontrou os arquivos ou houve algum outro erro com o formato do arquivo.

text

Desenha o texto usando a fonte da interface e na cor definida pelo usuário. O texto que será desenhado deve ser informado através do atributo `string`. Um atributo `align` pode ser usado para definir o alinhamento do texto. Se presente, o atributo `align` deve ser um valor inteiro onde (zero) significa centralizado, 1 (um) alinhado à esquerda e 2 (dois) alinhado à direita. Caso o atributo `align` esteja ausente o texto será centralizado automaticamente.

led7seg

Desenha um mostrador LED ou fluorescente alfanumérico comum com dezesseis segmentos e o mostrador numa determinada cor. Os oito bits baixos do estado do elemento controlam quais os segmentos estarão acesos. Começando pelo bit de menor importância a sequência de atualização dos bits correspondentes começam no segmento superior, superior direito, depois continuando no sentido horário para o segmento superior esquerdo, a barra central e o ponto decimal. Os pixels que estiverem apagados são desenhados com uma intensidade menor (`0x20/0xff`).

led14seg

Desenha um mostrador LED ou fluorescente alfanumérico padrão com catorze segmentos numa determinada cor. Os 14 bits mais baixos do controle de estado do elemento determinam quais os segmentos estarão acesos. Começando pelo bit com menor importância, os bits correspondentes ao segmento superior, o segmento superior direito, continuando no sentido horário para o segmento superior esquerdo, as metades esquerda e direita da barra central horizontal, as metades superior e inferior do meio vertical da barra, e as barras diagonais no sentido horário da parte inferior esquerda para a direita inferior. Os pixels que estiverem apagados são desenhados com uma intensidade menor (**0x20/0xff**).

led14segsc

Desenha um mostrador LED ou fluorescente alfanumérico padrão com catorze segmentos com ponto decimal/vírgula numa determinada cor. Os 16 bits baixos do elemento controlam quais segmentos estarão acesos. Os 14 bits baixos correspondem aos mesmos segmentos que no componente led14seg. Os dois bits adicionais correspondem ao ponto decimal e a vírgula. Os pixels que estiverem apagados são desenhados com uma intensidade menor (**0x20/0xff**).

led16seg

Desenha um mostrador LED ou fluorescente alfanumérico padrão com dezesseis segmentos numa determinada cor. Os 16 bits baixos do elemento controlam quais os elementos que estarão acesos. Começando pelo bit de menor importância a sequência de atualização dos bits correspondentes começam da metade esquerda da barra superior, a metade direita da barra superior, continuando no sentido horário para o segmento superior esquerdo, as metades esquerda e direita da barra central e horizontal, as metades superior e inferior da barra do meio vertical, e as barras diagonais no sentido horário a partir do canto inferior esquerdo até a parte inferior direito. Os pixels que estiverem apagados são desenhados com uma intensidade menor (**0x20/0xff**).

led16segsc

Desenha um mostrador LED ou fluorescente alfanumérico padrão com dezesseis segmentos e o ponto decimal numa determinada cor. Os 16 bits baixos do elemento controlam quais os segmentos estarão acesos. Os 18 bits inferiores correspondem aos mesmos controles do estado dos segmentos que em led16seg. Os dois bits adicionais correspondem ao ponto decimal e a vírgula. Os pixels que estiverem apagados são desenhados com uma intensidade menor (**0x20/0xff**).

simplecounter

Exibe o valor numérico do estado do elemento usando a fonte do sistema numa determinada cor. O valor é formatado em notação decimal. Um atributo `digits` pode ser informado para definir a quantidade mínima de dígitos que serão exibidos. Se presente, o atributo `digits` deve ser um número inteiro, na sua ausência será exibido um dígito com no mínimo dois dígitos.

O atributo `maxstate` pode ser informado para definir o valor máximo do estado que será exibido. Se presente, o atributo `maxstate` deve ser um número positivo; na sua ausência o valor predefinido é **999**. Um atributo `align` pode ser usado para determinar o alinhamento do texto através do atributo `align` que deve ser um número inteiro onde 0 significa alinhar ao centro, 1 alinhar à esquerda e 2 alinhar à direita. Na sua ausência, o texto será centralizado automaticamente.

Um exemplo de um elemento que desenha um texto estático do lado esquerdo da tela:

```
<element name="label_reset_cpu">
  <text string="CPU" align="1"><color red="1.0" green="1.0" blue="1.0" /></text>
</element>
```

Um exemplo de um elemento que mostra um LED redondo onde a intensidade do seu brilho depende do nível do seu estado na saída:

```
<element name="led" defstate="0">
  <disk state="0"><color red="0.43" green="0.35" blue="0.39" /></disk>
  <disk state="1"><color red="1.0" green="0.18" blue="0.20" /></disk>
</element>
```

Um exemplo de um elemento para um botão que retorna um efeito visual quando ele for clicado:

```
<element name="btn_rst">
  <rect state="0"><bounds x="0.0" y="0.0" width="1.0" height="1.0" /><color red="0.2"
  ↪green="0.2" blue="0.2" /></rect>
  <rect state="1"><bounds x="0.0" y="0.0" width="1.0" height="1.0" /><color red="0.1"
  ↪green="0.1" blue="0.1" /></rect>
  <rect state="0"><bounds x="0.1" y="0.1" width="0.9" height="0.9" /><color red="0.1"
  ↪green="0.1" blue="0.1" /></rect>
  <rect state="1"><bounds x="0.1" y="0.1" width="0.9" height="0.9" /><color red="0.2"
  ↪green="0.2" blue="0.2" /></rect>
  <rect><bounds x="0.1" y="0.1" width="0.8" height="0.8" /><color red="0.15" green="0.
  ↪15" blue="0.15" /></rect>
  <text string="RESET"><bounds x="0.1" y="0.4" width="0.8" height="0.2" /><color red=
  ↪"1.0" green="1.0" blue="1.0" /></text>
</element>
```

Um exemplo de um elemento que desenha um LED com sete segmentos usando imagens externas:

```
<element name="digit_a" defstate="0">
  <image file="a_off.png" />
  <image file="a_a.png" statemask="0x01" />
  <image file="a_b.png" statemask="0x02" />
  <image file="a_c.png" statemask="0x04" />
  <image file="a_d.png" statemask="0x08" />
  <image file="a_e.png" statemask="0x10" />
  <image file="a_f.png" statemask="0x20" />
  <image file="a_g.png" statemask="0x40" />
```

(continua na próxima página)

(continuação da página anterior)

```
<image file="a_dp.png" statemask="0x80" />
</element>
```

Um exemplo de um gráfico com barras que crescem verticalmente e mudam da cor verde, passando pelo amarelo e para o vermelho à medida que o nível for aumentando:

```
<element name="pedal">
  <rect>
    <bounds state="0x000" left="0.0" top="0.9" right="1.0" bottom="1.0" />
    <bounds state="0x610" left="0.0" top="0.0" right="1.0" bottom="1.0" />
    <color state="0x000" red="0.0" green="1.0" blue="0.0" />
    <color state="0x184" red="1.0" green="1.0" blue="0.0" />
    <color state="0x610" red="1.0" green="0.0" blue="0.0" />
  </rect>
</element>
```

Um exemplo de um gráfico com barras que crescem horizontalmente para a esquerda ou para a direita e muda de cor do verde, passando pelo amarelo e para o vermelho à medida que o nível muda da posição neutra:

```
<element name="wheel">
  <rect>
    <bounds state="0x800" left="0.475" top="0.0" right="0.525" bottom="1.0" />
    <bounds state="0x280" left="0.0" top="0.0" right="0.525" bottom="1.0" />
    <bounds state="0xd80" left="0.475" top="0.0" right="1.0" bottom="1.0" />
    <color state="0x800" red="0.0" green="1.0" blue="0.0" />
    <color state="0x3e0" red="1.0" green="1.0" blue="0.0" />
    <color state="0x280" red="1.0" green="0.0" blue="0.0" />
    <color state="0xc20" red="1.0" green="1.0" blue="0.0" />
    <color state="0xd80" red="1.0" green="0.0" blue="0.0" />
  </rect>
</element>
```

As visualizações

Uma visualização (elemento `view`) define um arranjo dos elementos ou das imagens exibidas da tela emulada numa janela ou numa tela. As exibições também conectam os elementos, as entradas E/S e as saídas emuladas. Um arquivo layout pode conter vários elementos `view`, caso uma delas corresponda a uma tela inexistente, esta se torna *inválida*.

O MAME exibirá uma mensagem de aviso ignorando toda a visualização que for considerada inválida e continuará a carregar aquelas que estiverem corretas. Isso é muito útil nos sistemas onde uma tela seja opcional, como computadores que tenham apenas controles no painel frontal e onde um terminal serial seja opcional.

As visualizações são identificadas através do nome na interface de usuário do MAME ou na linha de comando. Para os arquivos dos layouts que sejam associados aos dispositivos ou a outros onde o dispositivo do controlador principal, os nomes das visualizações dos dispositivos sejam precedidos por uma tag (com os dois pontos iniciais omitidos) por exemplo, para exibir um dispositivo chamado "Keyboard LEDs" vindo do dispositivo `:tty:ie15`, ele deve ser associado como **`tty:ie15 Keyboard LEDs`**.

As visualizações são exibidas na ordem em que forem sendo carregadas.

As visualizações são criadas com elementos `view` dentro de um atributo do primeiro nível do elemento `mamelayou`t. É obrigatório que cada elemento `view` tenha um atributo `name` informando um nome único que será disponibilizado na interface do usuário e nas opções da linha de comando. Este é um exemplo de um atributo válido para um elemento `view`:

```
<view name="Painel de controle">
```

O elemento "view" cria uma seção visível do `mamelayou`t. Os elementos `view` apenas são processados **depois** que todas as outras ramificações dos outros elementos do `mamelayou`t forem corretamente carregadas. Isso significa que uma visualização pode fazer referência a elementos e aos grupos que apareçam posteriormente naquele arquivo assim como os valores finais dos parâmetros que estejam anexados ao escopo do `mamelayou`t.

Um elemento `view` pode ter um atributo `showpointers` para definir se os ponteiros do mouse e da caneta devem ser exibidos na visualização. Quando o atributo estiver presente, o valor deverá ser `yes` ou `no`, quando não estiver, os ponteiros do mouse e da caneta serão mostrados para as exibições que contêm itens vinculados as portas de E/S, exemplo:

```
<view name="Painel de controle" showpointers="no">
```

Nota: A partir da versão **0.265** do MAME é preciso definir o `showpointers` como **no** em seu arquivo de layout, caso contrário, o ponteiro laranja do mouse irá aparecer nas suas ilustrações gráficas (artwork) ainda que ele não seja necessário para aquela ilustração. A predefinição é exibir o ponteiro, caso esta seja a sua intenção, não é preciso alterar nada.

As seguintes ramificações dos elementos são permitidos dentro de um elemento `view`:

bounds

Define a origem e o tamanho da visualização através das coordenadas interna do sistema caso um esteja presente. Consulte [Coordenadas](#) para obter mais informações. Em sua ausência os limites da visualização serão computadas unindo os limites de todas as telas e dos elementos dentro da região exibida. Só faz sentido ter um elemento `bounds` caso seja uma ramificação direta de um elemento `view`. Qualquer conteúdo fora dos limites da visualização ficarão recortados e a visualização será redimensionada de forma proporcional para que se ajuste aos limites da tela ou da janela.

param

Define ou reatribui um valor no parâmetro do escopo da visualização. Consulte [Parâmetros](#) para obter mais informações.

element

Adiciona um elemento à visualização (consulte [Os elementos](#)) através do atributo do elemento obrigatório `ref`. Haverá um erro caso nenhum elemento `ref` seja definido no arquivo layout.

Opcionalmente pode estar conectada numa porta E/S emulada através dos atributos `inputtag` e `inputmask` ou através da emulação de uma saída usando um atributo `name`. Consulte [Itens que podem ser clicados](#) e também [O Estado do elemento](#) para obter mais detalhes sobre como informar o valor de uma condição/estado para o elemento que for solicitado.

screen

Adiciona uma imagem emulada da tela na visualização. A tela deve ser identificada através do atributo `index` ou do atributo `tag` (um elemento `screen` não pode ter ambos os atributos `index` e `tag`). Caso esteja presente, o atributo `index` deve ter um valor inteiro e positivo. As telas são numeradas através da ordem em que aparecem na configuração do sistema, começando com zero (0). Caso o atributo `tag` esteja presente, este deve ser o caminho da etiqueta para a tela com relação ao dispositivo para que provoque a leitura do layout. As telas são desenhadas na ordem em que aparecem no arquivo layout.

Pode opcionalmente estar conectada numa porta E/S emulada através dos atributos `inputtag` e `inputmask` ou através de uma saída emulada através do atributo `name`. Consulte [Itens que podem ser clicados](#) para obter mais informações.

collection

Adiciona as telas ou os itens numa coleção de itens que poderão ser exibidos ou escondidos pelo usuário (consulte [Coleções](#)). O nome da coleção é definida através do atributo `name`. Há um limite de até 32 `collection` por visualização.

group

Adiciona o conteúdo do grupo na visualização (consulte [Grupos reutilizáveis](#)). O nome do grupo que será adicionado pode ser definido através do atributo `ref`. Haverá um erro caso nenhum grupo com este atributo seja definido no arquivo layout. Veja abaixo para mais informações sobre a questão de posicionamento.

repeat

Repete seu conteúdo pela quantidade de vezes que estiver definida no atributo `count`. O atributo `count` deve ser um número inteiro e positivo. O elemento `repeat` aceita os elementos `element`, `screen`, `group` mais os elementos `repeat` que funcionam da mesma maneira que quando colocados numa visualização direta. Consulte [Repetindo os blocos](#) para saber como usar os elementos `repeat`.

As telas com os elementos `screen` e os elementos `element` do layout podem ter um atributo `id`. Quando presente o atributo `id` deve ser único entre os elementos `view` e seu valor não pode ser vazio, isso inclui telas (`screens`) e os elementos instanciados através dos grupos reutilizáveis e da repetição dos blocos. Os elementos de tela e layout com o atributo `id` são identificados através de scripts Lua (consulte [Usando scripts no layout do MAME](#)).

As telas com os elementos `screen`, elementos do layout `element` e os elementos de grupo `group`, podem ter a sua orientação alterada usando o elemento `orientation`. Para as telas, os modificadores de orientação são aplicados em conjunto com os modificadores de orientação definido na tela do dispositivo e no sistema. O elemento `orientation` suporta os seguintes atributos opcionais:

rotate

Se presente, aplica rotação no sentido horário em incrementos de 90 graus. Deve ser um número inteiro igual a 0 90, 180 (90 + 90) ou 270 (180 + 90).

swapxy

Permite que a tela, elemento ou grupo seja espelhado ao longo de uma linha em 45 graus na vertical, da esquerda para a direita. Se presente o seu valor deve ser `yes` ou `no`. O espelhamento se aplica logicamente após a rotação.

flipx

Permite que a tela, elemento ou grupo sejam espelhados à partir de uma linha com 45 graus em torno de seu eixo vertical, vindo da quina superior esquerda até a quina inferior direita. Se presente o seu valor deve ser `yes` ou `no`. O espelhamento ocorre após a rotação.

flipy

Permite que a tela, elemento ou grupo sejam espelhados ao redor do seu eixo horizontal de cima para baixo. Se presente o seu valor deve ser `yes` ou `no`. O espelhamento ocorre após a rotação.

As telas (elementos `screen`) e os elementos do layout (elementos `element`) podem conter um atributo `blend` para determinar o modo de mesclagem dos elementos gráficos. Os valores válidos são `none` (sem mesclagem), `alpha` (transparência)⁵, `multiply` (soma dos valores RGB)⁶ e `add` (soma das camadas)⁷. A predefinição para a tela é `alpha` permitindo que o driver defina a mesclagem dos elementos do layout através de camadas.

As telas (elementos `screen`), elementos do layout (elementos `element`) e elementos de grupo (`group`) podem ser posicionados e redimensionados usando um elemento `bounds` (consulte [Coordenadas](#) para mais informações). Na ausência do sub-elemento `bounds` os elementos `"screen"` e `"layout"` retornam aos valores predefinidos em unidades quadradas (origem em **0,0** e ambos os valores de altura e largura serão igual a 1

Na ausência do elemento `bounds`, os grupos são expandidos sem qualquer tradução ou redimensionamento (note que os grupos podem posicionar as telas ou elementos fora dos seus limites. Este exemplo mostra uma visualização com referência a posição da tela com um elemento individual do layout e dois grupos de elementos:

```
<view name="LED Displays, Terminal and Keypad" showpointers="no">
  <screen index="0"><bounds x="0" y="132" width="320" height="240" /></screen>
  <element ref="beige"><bounds x="320" y="0" width="172" height="372" /></element>
  <group ref="displays"><bounds x="0" y="0" width="320" height="132" /></group>
  <group ref="keypad"><bounds x="336" y="16" width="140" height="260" /></group>
</view>
```

⁵ Alpha Blending

⁶ RGB multiplication

⁷ Additive blending

As telas (elementos screen), os elementos do layout (element) e os elementos do grupos (group) podem ter um sub-elemento color (consulte [Cores](#)) ao definir uma cor modificadora. O valor dessa cor será usada como multiplicador para alterar as cores componentes da tela ou dos elementos do layout.

As telas (elementos screen) e os elementos do layout (element) podem ter a sua cor, posição e tamanho animados ao informar diversos elementos color e/ou sub-elementos bounds em conjunto com o atributo state. Consulte [A visualização de um item animado](#) para obter mais informações.

Coleções

As coleções das telas ou dos elementos do layout que são agrupados de maneira que possam ser exibidos ou não pelo usuário conforme a sua necessidade caso este esteja definido pelo autor do layout. Numa visualização única, é possível ambas as visualizações e um teclado numérico (keypad) selecionável por exemplo, permitir que o usuário esconda o teclado numérico deixando visível apenas a visualização. As coleções são criadas através do elemento collection dentro dos elementos view, group e dos outros elementos collection.

Um elemento collection deve ter um atributo name informando o nome da visualização. Os nomes destinados para collection devem ser únicos. A visualização inicial da coleção deve ser definida através do atributo visible. Defina o atributo visible para yes caso a coleção deva estar visível desde o início ou no caso queira escondê-la. É predefinido que as coleções estejam visíveis.

Aqui um exemplo demonstrando a utilização de um collection permitindo que partes de uma visualização possam ser escondidas pelo usuário:

```
<view name="Telas LED, CRT e Teclado Numérico" showpointers="no">
  <collection name="Telas LED">
    <group ref="displays"><bounds x="240" y="0" width="320" height="47" /></group>
  </collection>
  <collection name="Teclado numérico">
    <group ref="keypad"><bounds x="650" y="57" width="148" height="140" /></group>
  </collection>
  <screen tag="screen"><bounds x="0" y="57" width="640" height="480" /></screen>
</view>
```

Uma coleção cria um escopo de parâmetros agrupados. Qualquer elemento param que estiver dentro do elemento de coleção define os parâmetros no escopo local para a coleção. Para mais detalhes sobre os parâmetros consulte [Parâmetros](#). Observe que o nome da coleção e a visualização predefinida não fazem parte do seu conteúdo, quaisquer referências dos parâmetros nos atributos name e visible serão substituídos usando os valores dos parâmetros a partir da origem do escopo relacionado com a coleção.

Para mais informações consulte [Desativando objetos na tela](#).

Grupos reutilizáveis

Os grupos permitem que um arranjo das telas ou dos elementos do layout sejam usados várias vezes numa visualização ou outros grupos. Os grupos podem ser de grande ajuda mesmo que seja usado o arranjo apenas uma vez, pois eles podem ser usados para agregar parte de um layout complexo. Os grupos são definidos usando elementos `group` dentro de elementos `mamelayout` de primeiro nível e representados ao usar elementos `group` dentro de elementos `view` e outros elementos `group`.

Cada definição de grupo deve ter um atributo `name` informando um identificador único. Será considerado um erro caso o arquivo layout tenham várias definições de grupos usando um atributo `name` idêntico. O valor do atributo `name` é usado quando for justificar a visualização de um grupo ou outro. Este é um exemplo da abertura da etiqueta para definir o grupo de um elemento dentro do primeiro elemento `mamelayout`:

```
<group name="panel">
```

Este grupo pode então ser justificado numa visualização ou em outro elemento `group` usando um elemento de grupo como referência. Opcionalmente os limites de destino, a orientação e as modificações das cores poderão ser informados também. O atributo `ref` identifica o grupo a qual faz referência, neste exemplo são informados os limites dos valores:

```
<group ref="panel"><bounds x="87" y="58" width="23" height="23.5" /></group>
```

Os elementos de definição dos grupos permitem que todos os elementos filhos que forem iguais, sejam exibidos. O posicionamento e as orientações das telas, os elementos do layout e o arranjo destes grupos para que funcionem da mesma maneira que as visualizações. Veja [As visualizações](#) para mais informações. Um grupo pode justificar outros grupos, porém loops recursivos não são permitidos. Será considerado um erro caso um grupo represente a si mesmo de forma direta ou indireta.

Os grupos possuem seus próprios sistemas de coordenadas internas. Caso um elemento de definição de grupo não tenha um elemento limitador `bounds` como filho direto, os seus limites serão computados junto com a união dos limites de todas as telas, os elementos do layout ou dos grupos relacionados. Um elemento filho `bounds` pode ser usado para definir explicitamente grupos limitadores (consulte [Coordenadas](#) para mais informações). Observe que os limites dos grupos são usados com a única justificativa para calcular as coordenadas de transformação quando forem relacionados a um grupo. Um grupo pode posicionar as telas ou os elementos fora dos seus limites sem que sejam cortados.

Para demonstrar como o cálculo dos limites funcionam, considere este exemplo:

```
<group name="autobounds">
  <!-- limites automaticamente calculados com sua origem em (5,10), largura 30, e altura_
↳15 -->
  <element ref="topleft"><bounds x="5" y="10" width="10" height="10" /></element>
  <element ref="bottomright"><bounds x="25" y="15" width="10" height="10" /></element>
</group>

<view name="Teste" showpointers="no">
  <!--
    Os grupos limitadores são traduzidos e escalonados para preencher 2/3 da escala
    horizontal e o dobro verticalmente.
    O elemento superior esquerdo posicionado em (0,0) com 6.67 de largura e 20 de altura
    O elemento inferior direito posicionado em (13.33,10) com 6.67 de largura e 20 de_
↳altura
    Os elementos de visualização calculado com origem em (0,0) 20 de largura e 30 de_
↳altura
  -->
  <group ref="autobounds"><bounds x="0" y="0" width="20" height="30" /></group>
</view>
```

Como todos os elementos inerentemente caem dentro dos limites calculados ao grupo de forma automática. Agora, considere o que acontece caso a posição dos elementos de um grupo estejam fora dos seus limites:

```
<group name="periphery">
  <!-- os limites dos elementos estão acima da quina superior e à direita da quina direita_
↳-->
  <bounds x="10" y="10" width="20" height="25" />
  <element ref="topleft"><bounds x="10" y="0" width="10" height="10" /></element>
  <element ref="bottomright"><bounds x="30" y="20" width="10" height="10" /></element>
</group>

<view name="Test" showpointers="no">
  <!--
    Os grupos limitadores são traduzidos e escalonados para preencher 2/3 da escala
    horizontal unido verticalmente.
    O elemento superior esquerdo posicionado em (5,-5) com 15 de largura e 10 de altura
    O elemento inferior direito posicionado em (35,15) com 15 de largura e 10 de altura
    Os elementos de visualização calculado com origem em (5,-5) 45 de largura e 30 de_
↳altura
  -->
  <group ref="periphery"><bounds x="5" y="5" width="30" height="25" /></group>
</view>
```

Os elementos de grupo são traduzidos e escalonados conforme sejam necessários para distorcer os limites internos dos grupos para o limite de visualização final. O conteúdo dos grupos não ficam restritos aos seus limites. A visualização considera os limites dos elementos atuais ao calcular os seus próprios limites e não aos limites do destino definido para o grupo.

Quando um grupo é instanciado¹⁰, ele cria um escopo agrupado do parâmetro. A lógica do escopo principal é o escopo do parâmetro de visualização, do grupo ou do bloco de repetição onde o grupo for instanciado (*não* é um parente léxico ao elemento de primeiro nível `mamelayout`). Qualquer elemento `param` dentro da definição do conjunto, estabelece os parâmetros dos elementos no escopo local para o grupo instanciado. Os parâmetros locais não se preservam através das várias instancias.

Consulte [Parâmetros](#) para obter mais informações sobre os parâmetros. (Observe que o nome dos grupos não fazem parte do seu conteúdo e qualquer referência de parâmetro no próprio atributo `name` será substituído no ponto onde a definição do grupo aparecer no primeiro nível do elemento de escopo `mamelayout`.)

¹⁰ Em programação orientada a objetos, chama-se instância de uma classe, um objeto cujo comportamento e estado sejam definidos pela classe. ([Wikipedia](#))

Repetindo os blocos

A repetição dos blocos fornecem uma maneira concisa de gerar ou para organizar uma grande quantidade de elementos iguais. A repetição dos blocos são geralmente usados em conjunto com o gerador de parâmetros (consulte [Parâmetros](#)). A repetição dos blocos podem ser agrupados para criar arranjos mais complexos.

Os blocos repetidos são criados através do elemento repeat. Cada elemento repeat requer um atributo count definindo uma quantidade de iterações que serão geradas. O atributo count deve ser um número inteiro e positivo. A repetição dos blocos é permitida dentro do elemento de primeiro nível mamelayout, dentro dos elementos group e view assim como dentro dos outros elementos repeat. O exato sub-elemento permitido dentro do elemento repeat depende de onde ele for aparecer:

- Um bloco repetido dentro do elemento de primeiro nível mamelayout podem conter os seguintes elementos param, element, group (definição) e repeat.
- Um bloco repetido dentro de um elemento group ou view podem conter os seguintes elementos, param, element (referência), screen, group (referência) e repeat.

Um bloco de repetição repete o seu conteúdo diversas vezes dependendo do valor definido no atributo count. Consulte as seções relevantes para obter mais informações de como os sub-elementos são usados ([As partes de um layout](#), [Grupos reutilizáveis](#) e [As visualizações](#)). Um bloco que se repete cria um escopo de parâmetros agrupados dentro do escopo do parâmetro do seu elemento léxico principal (DOM).

O exemplo abaixo geram rótulos numéricos em branco a partir de 0 até 11 com o nome label_0, label_1 e assim por diante (dentro do elemento de primeiro nível mamelayout):

```
<repeat count="12">
  <param name="labelnum" start="0" increment="1" />
  <element name="label_~labelnum~">
    <text string="~labelnum~"><color red="1.0" green="1.0" blue="1.0" /></text>
  </element>
</repeat>
```

Uma fileira horizontal com 40 mostradores digitais, separadas por cinco unidades de espaço entre elas, controladas pelas saídas digit0 até digit39 (dentro de um elemento group ou view):

```
<repeat count="40">
  <param name="i" start="0" increment="1" />
  <param name="x" start="5" increment="30" />
  <element name="digit~i~" ref="digit">
    <bounds x="~x~" y="5" width="25" height="50" />
  </element>
</repeat>
```

Oito mostradores com matrix de ponto medindo cinco por sete numa linha, com pixels controlados por Dot_000 até Dot_764 (dentro de um elemento group ou view):

```
<!-- 8 dígitos -->
<repeat count="8">
  <param name="digitno" start="1" increment="1" />
  <!-- a distância entre os dígitos ((111 * 5) + 380) -->
  <param name="digitx" start="0" increment="935" />
  <!-- 7 linhas para cada dígito -->
  <repeat count="7">
    <param name="rowno" start="1" increment="1" />
    <!-- a distância vertical entre os LEDs -->
    <param name="rowy" start="0" increment="114" />
    <!-- 5 colunas em cada dígito -->
    <repeat count="5">
      <param name="colno" start="1" increment="1" />
      <!-- a distância horizontal entre os LEDs -->
      <param name="colx" start="~digitx~" increment="111" />
      <element name="Dot_~digitno~~rowno~~colno~" ref="Pixel" state="0">
        <!-- o tamanho de cada LED -->
        <bounds x="~colx~" y="~rowy~" width="100" height="100" />
      </element>
    </repeat>
  </repeat>
</repeat>
```

Dois teclados que podem ser clicados, separados horizontalmente por um teclado numérico quatro por quatro (dentro de um elemento group ou view):

```
<repeat count="2">
  <param name="group" start="0" increment="4" />
  <param name="padx" start="10" increment="530" />
  <param name="mask" start="0x01" lshift="4" />
  <repeat count="4">
    <param name="row" start="0" increment="1" />
    <param name="y" start="100" increment="110" />
    <repeat count="4">
      <param name="col" start="~group~" increment="1" />
      <param name="btnx" start="~padx~" increment="110" />
      <param name="mask" start="~mask~" lshift="1" />
      <element ref="btn~row~~col~" inputtag="row~row~" inputmask="~mask~">
        <bounds x="~btnx~" y="~y~" width="80" height="80" />
      </element>
    </repeat>
  </repeat>
</repeat>
```

Os botões são desenhados usando os elementos `btn00` na parte superior esquerda, `btn07` na parte superior direita, `btn30` na parte inferior esquerda e `btn37` na parte inferior direita contando entre eles. As quatro colunas são conectadas às portas E/S `row0`, `row1`, `row2`, and `row3` de cima para baixo. As colunas consecutivas são conectadas aos bits das portas E/S começando com o bit de menor importância do lado esquerdo.

Observe que o parâmetro `mask` no elemento mais interno `repeat` recebe o seu valor inicial a partir do parâmetro correspondentemente nomeado no delimitador do escopo, porém sem alterá-lo.

Gerando um tabuleiro de xadrez com valores alfa alternados entre 0.4 e 0.2 (dentro de um elemento `group` ou `view`):

```
<repeat count="4">
  <param name="pairy" start="3" increment="20" />
  <param name="pairno" start="7" increment="-2" />
  <repeat count="2">
    <param name="rowy" start="~pairy~" increment="10" />
    <param name="rowno" start="~pairno~" increment="-1" />
    <param name="lalpha" start="0.4" increment="-0.2" />
    <param name="ralpha" start="0.2" increment="0.2" />
    <repeat count="4">
      <param name="lx" start="3" increment="20" />
      <param name="rx" start="13" increment="20" />
      <param name="lmask" start="0x01" lshift="2" />
      <param name="rmask" start="0x02" lshift="2" />
      <element ref="hl" inputtag="board:IN.~rowno~" inputmask="~lmask~">
        <bounds x="~lx~" y="~rowy~" width="10" height="10" />
        <color alpha="~lalpha~" />
      </element>
      <element ref="hl" inputtag="board:IN.~rowno~" inputmask="~rmask~">
        <bounds x="~rx~" y="~rowy~" width="10" height="10" />
        <color alpha="~ralpha~" />
      </element>
    </repeat>
  </repeat>
</repeat>
```

O elemento `repeat` mais externo gera um grupo com duas colunas em cada interação; o próximo elemento `repeat` gera uma coluna individual em cada interação; o elemento `repeat` interno produz dois recortes horizontais adjacentes em cada interação. As colunas são conectadas às portas E/S através do `board:IN.7` no topo do `board.IN.0` na parte inferior.

12.2.4 Interatividade

As visualizações com interatividade são suportadas através da permissão dos itens que serão vinculados nas saídas e nas portas E/S. Há suporte para cinco tipos de interatividades:

Itens que podem ser clicados

Caso um item numa visualização esteja vinculado com uma região dos interruptores da porta E/S, será possível clicar no item para ativar o interruptor ou um botão emulado.

Componentes que dependam de uma condição

Dependendo do estado do elemento que o contiver, alguns componentes serão desenhados de forma diferente. Isso inclui a matriz de pontos, o display de LEDs com vários segmentos, os contadores simples e os elementos com mostradores rotativos. Consulte [Os elementos](#) para obter mais detalhes.

Componentes desenhados de forma condicional

Os componentes podem ser desenhados de forma condicional ou escondidos dependendo da condição do conteúdo do elemento a partir da informação dos valores para os elementos state e/ou statemask. Consulte [Os elementos](#) para obter mais detalhes.

Parâmetros para a animação dos componentes

A posição, tamanho e a cor dos componentes contido em seus elementos talvez possam ser animados de acordo com a condição do elemento a partir da informação dos diversos elementos color e/ou bounds em conjunto com os atributos de condição state. Consulte [Os elementos](#) para obter mais detalhes.

Parâmetros para a animação dos itens

A cor, a posição e o tamanho dos itens restritos ao seu espaço de visualização podem ser animados de acordo com a sua condição.

Itens que podem ser clicados

Caso um item de visualização (elemento `element` ou `screen`) tenham atributos `inputtag` e `inputmask` com valores que correspondam a uma região com interruptores digitais no sistema emulado, será possível clicar no elemento para que determinado interruptor seja ativado. O interruptor permanecerá ativo enquanto o botão o botão primário estiver pressionado e o ponteiro estiver dentro dos limites do item. (Observe que os limites podem mudar dependendo da condição do estado de animação do item, consulte [A visualização de um item animado](#)).

O atributo `inputtag` determina o caminho do identificador de uma porta E/S relativa ao dispositivo responsável pelo carregamento do arquivo layout. O atributo `inputmask` deve ser um valor inteiro definindo os bits da região da porta de E/S que o item deve ativar.

Este exemplo demonstra a instanciação dos botões que podem ser clicados:

```
<element ref="btn_3" inputtag="X2" inputmask="0x10">
  <bounds x="2.30" y="4.325" width="1.0" height="1.0" />
</element>
<element ref="btn_0" inputtag="X0" inputmask="0x20">
  <bounds x="0.725" y="5.375" width="1.0" height="1.0" />
</element>
<element ref="btn_rst" inputtag="RESET" inputmask="0x01">
  <bounds x="1.775" y="5.375" width="1.0" height="1.0" />
</element>
```

Ao lidar com o retorno das informações vindas da entrada, o MAME trata todos os elementos do layout como sendo um retângulo.

Para **bloquear** o elemento de ser clicado na tela crie uma camada vazia com as mesmas dimensões do item bloqueado antes do item que você deseja bloquear. Primeiro crie-o no início do layout:

```
<element name="cobertura" defstate="0">
  <text string=" " />
</element>
```

Em seguida use-o **antes** do elemento que será protegido. A organização precisa ser feita desta maneira pois é nesta ordem que o MAME renderiza os elementos do layout na tela, primeiro vem o elemento "cobertura" seguido por outros elemento **abaixo dele**. O atributo `bounds` definirá a sua posição na tela (x e y), `width` a largura da "cobertura" (em pixels) e `height` a altura:

```
<element ref="cobertura" blend="add" inputtag="IN0" inputmask="0x0" inputraw="yes">
  <bounds x="1783" y="3919" width="270" height="270" />
</element>
<element ref="controle" inputtag="IN0" inputmask="0xf" inputraw="yes">
  <bounds x="1783" y="3919" width="270" height="270" />
</element>
```

Na versão **0.265** do MAME em diante, o atributo `clickthrough` controla se os cliques podem passar através do item de visualização para outros itens desenhados acima dele. Caso esteja presente, o valor do atributo `clickthrough` deve ser `yes` ou `no`. A predefinição é `no` (os cliques não atravessam) nos itens de visualização com atributos `inputtag` e `inputmask`. Já a predefinição se torna `yes` (os cliques atravessam) para os outros itens de visualização.

Da mesma maneira que o exemplo anterior, agora podemos utilizar o atributo `clickthrough` para bloquear os cliques. Primeiro definimos um retângulo transparente:

```
<element name="cobertura" defstate="0">
  <rect><color alpha="0" /></rect>
</element>
```

Agora nós invocamos a nossa "cobertura" de proteção na região onde queremos que ela seja aplicada:

```
<element ref="cobertura" clickthrough="no">
  <bounds x="1783" y="3919" width="270" height="270" />
</element>
<element ref="controle" inputtag="IN0" inputmask="0xf" inputraw="yes">
  <bounds x="1783" y="3919" width="270" height="270" />
</element>
```

No exemplo acima estamos criando uma "cobertura" de proteção para o "controle", repare que a nossa cobertura utiliza exatamente os mesmos parâmetros de posição e tamanho do elemento "controle" justamente para cobrir toda a área que ele estiver ocupando. Não é obrigatório que a nossa "cobertura" tenha exatamente o mesmo tamanho da região que desejamos proteger. Dependendo da ilustração e do sistema onde esta ilustração será utilizada, talvez seja interessante ir um pouco além dos limites do "controle" e cobrir uma área um pouco maior evitando falsos cliques, caso seja necessário.

Consulte também *[Adicionando diagonais nos controles com 8 direções usando o inputraw](#)*.

O Estado do elemento

Um item de visualização que instancie um elemento (elemento `element`) pode fornecer um valor da sua condição para o elemento a partir de uma porta emulada de E/S ou para a saída. Consulte *Os elementos* para obter mais detalhes sobre como o estado de um elemento afeta sua aparência.

O valor do estado do elemento será obtido através do valor da saída emulada que corresponda a tal nome caso o elemento `element` tenha um atributo `name`. Observe que os nomes das saídas são globais e podem se tornar um problema quando um sistema utilizar várias instâncias do mesmo tipo do dispositivo. Este exemplo mostra como os monitores digitais podem ser conectados na saída emulada:

```
<element name="digit6" ref="digit"><bounds x="16" y="16" width="48" height="80" /></element>
<element name="digit5" ref="digit"><bounds x="64" y="16" width="48" height="80" /></element>
<element name="digit4" ref="digit"><bounds x="112" y="16" width="48" height="80" /></element>
<element name="digit3" ref="digit"><bounds x="160" y="16" width="48" height="80" /></element>
<element name="digit2" ref="digit"><bounds x="208" y="16" width="48" height="80" /></element>
<element name="digit1" ref="digit"><bounds x="256" y="16" width="48" height="80" /></element>
```

O valor do estado do elemento será obtido a partir do valor da porta correspondente ao E/S mascarado com o valor do `inputmask` caso o elemento `element` tenha os atributos `inputtag` e `inputmask` porém não tenha um atributo `name`. O atributo `inputtag` determina o caminho do identificador de uma porta E/S relativa ao dispositivo responsável pelo carregamento do arquivo layout. O atributo `inputmask` deve ser um valor inteiro para definir os bits da região da porta E/S que o item deve ativar.

O valor da porta E/S é mascarado com o valor do `inputmask` e feito uma operação XOR¹¹ com o valor predefinido da região da porta E/S caso o elemento `element` não tenha qualquer atributo `inputraw` ou caso o valor do atributo `inputraw` seja `no`. Em geral é utilizado para fornecer um retorno visual para os botões que sejam clicáveis como valores normais para os interruptores alto-ativo e baixo-ativo.

O estado do elemento será obtido a partir dos valores da porta E/S mascarado com o valor do `inputmask` e deslocada para a direita para remover os zeros restantes caso o elemento `element` tenha um atributo `inputraw` com o valor `yes` (por exemplo, uma máscara com o valor `0x5` não terá deslocamento algum enquanto uma máscara com o valor `0xb0` resultará num deslocamento com quatro bits à direita). É útil para obter os valores analógicos das entradas ou das posições.

¹¹ Exclusive OR ou operador exclusivo, é um operando que sempre retorna 1 quando os bits da sua entrada são diferentes e 0 (zero) quando forem iguais.

A visualização de um item animado

A cor, a posição e o tamanho dos itens que estejam dentro dos limites da visualização poderão ser animados. Isso é feito através da definição dos diversos sub-elementos color ou bounds com atributos state. O atributo state deve ser um número inteiro positivo para cada elemento color ou sub-elemento bounds. Dentro do item de visualização os dois elementos color e os dois elementos bounds não podem ter os mesmos atributos state com os mesmos valores.

Para definir a posição ou o tamanho do item através do sub-elemento bounds será usado o menor valor do atributo state caso o estado de animação do item seja menor que o valor do atributo state de qualquer um dos sub-elementos bounds. Já a posição ou o tamanho definido pelo sub-elemento bounds será utilizado com o maior valor do atributo state caso o estado da animação do item seja maior que o valor do atributo state de qualquer um dos sub-elementos bounds. No entanto a posição ou o tamanho será interpolada de forma linear caso o estado da animação do item esteja entre os valores do atributo state dos dois sub-elementos bounds.

A cor será atribuída através do sub-elemento color com o menor valor do atributo state caso o estado de animação do item seja menor do que o valor do atributo state de qualquer sub-elemento color. O mesmo princípio é usado com o maior valor do atributo state. Os componentes da cor RGBA serão interpolados de forma linear caso o estado da animação do item esteja entre os valores do atributo state dos dois sub-elementos color.

O estado da animação de um item pode estar limitada a uma saída emulada ou a entrada de uma porta durante o fornecimento de um sub-elemento animate. Quando estiver presente o elemento animate deve possuir ou um atributo inputtag ou um atributo name (porém não ambos). Na ausência do sub-elemento animate o estado de animação do item será idêntico ao estado do seu elemento (consulte [O Estado do elemento](#)).

Quando um sub-elemento animate estiver presente e tiver um atributo inputtag, o estado da animação do item será obtido a partir do valor correspondente à porta E/S. O atributo inputtag determina o caminho da etiqueta de uma porta E/S relativa ao dispositivo que provoque a leitura do arquivo layout. São utilizados os valores brutos da porta da entrada, os valores baixo-ativo do interruptor não são normalizados.

Na presença de um sub-elemento animate com o atributo name o estado da animação do item será obtido através do valor do nome correspondente a saída emulada. Observe que os nomes das saídas são globais e podem se tornar um problema quando um sistema utilizar várias instâncias do mesmo tipo do dispositivo.

O estado da animação será mascarado com o valor mask e deslocada para a direita para remover os zeros restantes caso um sub-elemento animate tenha um atributo mask (por exemplo, uma máscara com o valor 0x5 não terá deslocamento algum enquanto uma máscara com o valor 0xb0 resultará num deslocamento com quatro bits à direita). Observe que o atributo mask aplica o valor da saída (determinado através do atributo inputtag). Na presença do atributo mask o seu valor deve ser inteiro, na ausência, é equivalente a todas as definições com 32 bits.

Este exemplo exibe elementos com estado independente para o elemento e para a animação obtendo o estado da animação a partir das saídas emuladas para controlar a sua posição:

```
<repeat count="5">
  <param name="x" start="10" increment="9" />
  <param name="i" start="0" increment="1" />
  <param name="mask" start="0x01" lshift="1" />

  <element name="cg_sol~i~" ref="cosmo">
    <animate name="cg_count~i~" />
    <bounds state="0" x=~x~" y="10" width="6" height="7" />
    <bounds state="255" x=~x~" y="48.5" width="6" height="7" />
  </element>

  <element ref="nothing" inputtag="FAKE1" inputmask=~mask~">
    <animate name="cg_count~i~" />
    <bounds state="0" x=~x~" y="10" width="6" height="7" />
    <bounds state="255" x=~x~" y="48.5" width="6" height="7" />
  </element>
</repeat>
```

Assim como no exemplo anterior porém agora usa o estado da emulação a partir da posição emulada da entrada para controlar as suas posições:

```
<repeat count="4">
  <param name="y" start="1" increment="3" />
  <param name="n" start="0" increment="1" />
  <element ref="ledr" name=~n~.7">
    <animate inputtag="IN.1" mask="0x0f" />
    <bounds state="0" x="0" y=~y~" width="1" height="1" />
    <bounds state="11" x="16.5" y=~y~" width="1" height="1" />
  </element>
</repeat>
```

12.2.5 Lidando com erros

- Para os arquivos internos do layout (fornecidos pelo desenvolvedor), os erros são detectados através script `complay.py` durante uma falha de compilação.
- O MAME irá parar de carregar um arquivo layout caso encontre um erro de sintaxe, fazendo assim com que nenhuma visualização do layout fique disponível. Alguns exemplos de erros de sintaxe incluem referências para elementos ou grupos indefinidos, limites inválidos, cores inválidas, grupos recursivamente emaranhados e a redefinição do gerador dos parâmetros.
- O MAME exibirá uma mensagem de aviso e continuará caso uma visualização faça referência à uma tela inexistente durante o carregamento de um layout. Visualizações apontando para telas não existentes não são exibidas, elas são consideradas inviáveis e tão pouco estarão disponíveis para o usuário.

12.2.6 As visualizações que são geradas automaticamente

Após o carregamento interno dos layouts (fornecido pelo desenvolvedor) e do layout externo (fornecido pelo usuário). As seguintes visualizações são geradas automaticamente:

- Será exibido a mensagem **"No screens Attached to the system"** ou **"Sem telas anexadas ao sistema"** caso o sistema não possua telas e tão pouco sejam encontradas visualizações viáveis no sistema interno ou externo do layout.
- A tela será exibida com a sua proporção física e com a rotação aplicada em cada tela que for emulada.
- A tela será exibida numa proporção onde os pixels sejam quadrados e com a rotação aplicada para cada tela emulada onde a proporção configurada para o pixel não corresponda a proporção física.
- Serão exibidos duas cópias da imagem da tela uma sobreposta a outra com um pequeno espaço entre elas caso o sistema emule apenas uma tela. A cópia da parte de cima será rotacionada em 180 graus. Esta visão pode ser usada num cabine tipo cocktail, que disponibiliza uma mesa onde os jogadores se sentam frente a frente e cada um com a sua tela, ou alternando os jogos que não girem automaticamente a tela para o segundo jogador.
- As telas serão organizadas horizontalmente da esquerda para a direita e verticalmente de cima para baixo, ambos com e sem as pequenas lacunas entre elas caso o sistema tenha exatamente duas telas emuladas e nenhuma visualização no layout interno ou no layout externo exibindo todas as telas, ou caso o sistema tenha mais de duas telas emuladas.
- As telas serão exibidas em formato de grade em ambas as fileiras principais (da esquerda para a direita e de cima para baixo) e o pilar principal (de cima para baixo e depois da esquerda para a direita). As visualizações são geradas com e sem intervalos entre as telas.

12.2.7 Usando o `complay.py`

No código-fonte do MAME existe um script Python chamado **`complay.py`**, encontrado no subdiretório **`scripts/build`**. Como parte do processo de compilação do MAME esse script é usado para reduzir o tamanho dos dados dos layouts internos e para convertê-los de maneira que possam ser anexados dentro do executável.

O script pode também detectar muitos erros comuns de formatação exibindo mensagens de erro com mais informações das que o MAME exhibe.

Observe que o script não executa todo o mecanismo do layout e portanto não tem a capacidade de detectar erros nos parâmetros usados como referências para elementos indefinidos ou para agrupamentos dos grupos organizados de forma recursiva. O script **`complay.py`** é compatível com os interpretadores Python a partir das versões 2.7, 3 ou mais recente, ele usa três parâmetros, um nome do arquivo na entrada, um nome do arquivo na saída e um nome base para as variáveis na saída:

```
python scripts/build/complay.py <input> [<output> [<varname>]]
```

É obrigatório o uso de um arquivo na entrada. Caso nenhum nome de arquivo seja usado na saída, o **`complay.py`** irá analisar e verificar apenas o arquivo da entrada, informando quaisquer erros que forem encontrados e não gerando qualquer tipo de arquivo na saída. Caso nenhum varname seja informado, o **`complay.py`** irá gerar um com base no nome do arquivo da entrada. Isso não garante a geração de identificadores válidos.

O script retorna as seguintes condições:

- 0 (zero) quando for concluído com êxito.
- 1 quando houver um erro durante a invocação através da linha de comando.
- 2 caso haja erro no arquivo de entrada.
- 3 caso seja um erro de E/S.

Ao definir um arquivo na saída, este será criado ou substituído caso seja concluído com sucesso ou será removido caso haja um erro.

Para aferir e testar um arquivo layout, execute o script apontando o caminho completo do arquivo como mostra o exemplo abaixo:

```
python scripts/build/complay.py artwork/dino/default.lay
```

12.2.8 Criando diferentes arquivos layout na prática

Neste capítulo criaremos um layout do zero para o sistema **Galaxian** demonstrando como definir todos os parâmetros para que todos os objetos apareçam na tela em seus devidos lugares e com o tamanho correto, no final será possível ver a capacidade do MAME de apresentar o design completo na tela, com a devida animação dos controles e dos botões e com todos os botões clicáveis.

Ferramentas necessárias

Para esta tarefa precisamos dos seguintes itens:

- Um editor de texto da sua preferência, recomendo o [Notepad++](#) no Windows ou o [Geany](#) para *nix e macOS.
- [Gimp](#).
- A [versão básica](#) do layout do sistema Galaxian usada neste documento.
- A versão básica do layout do sistema Galaxian usando o [método inputraw](#).
- A [versão avançada](#) do layout com diferentes versões do sistema Galaxian.
- O [layout modelo](#) criado para identificar as posições do controle para 2 e 4 jogadores.
- A arte utilizada aqui foi criada por [Etienne MacGyver](#).
- Uma [planilha](#) feita com [LibreOffice](#) para facilitar o cálculo da relação de aspecto da tela.
- Os botões foram criados pela minha amiga [u/cd4053b](#).
- A rom do sistema **Galaxian**.
- O MAME configurado e instalado no seu computador.

A versão básica do **Galaxian** já deve ter um arquivo `default.lay` montado e funcionando, porém vamos descrever como encontramos cada um dos valores utilizados nele. Os arquivos vêm com os respectivos nomes `basic_galaxian.zip`, `inputraw_galaxian.zip` e `advanced_galaxian.zip`, para melhor acompanhar o andamento dos capítulos faça a descompressão dos arquivos dentro do diretório **artwork** onde cada um esteja dentro do seu próprio diretório ou seja `basic_galaxian`, `inputraw_galaxian` e `advanced_galaxian`. Quando quiser avaliar qualquer um deles basta renomear **um deles** para **galaxian**

Identificando as partes

No diretório onde o seu MAME está instalado vá até **artwork**, dentro dele crie outro diretório chamado **galaxian** extraia o conteúdo do arquivo de imagens dentro deste diretório. Abra o seu editor de texto e adicione as duas primeiras linhas:

```
<?xml version="1.0"?>
<mamelayout version="2">
```

Salve o arquivo como `default.lay`.

O próximo passo é definir um nome para a nossa imagem de fundo, estamos usando o nome "Italiano" pois é a versão italiana do sistema Galaxian e também precisamos anexar junto ao nome a imagem que servirá como o fundo do nosso sistema:

```
<element name="Italiano">
    <image file="arte.png" />
</element>
```

Todas as imagens em grupos, é importante utilizar nomes bem específicos para cada uma delas. O defstate define a sua condição inicial e o state define o seu o estado em cada condição onde 0 (zero) significa quando o botão não estiver pressionado e 1 quando estiver, observe que a imagem usada para os direcionais e para o disparo pode ser a mesma:

```
<element name="J1" defstate="0">
  <image file="btn0.png" state="0" />
  <image file="btn1.png" state="1" />
</element>

<element name="J2" defstate="0">
  <image file="btn0.png" state="0" />
  <image file="btn1.png" state="1" />
</element>

<element name="esquerda" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>

<element name="direita" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>

<element name="disparo" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>

<element name="pisca" defstate="0">
  <image file="pisca1.png" state="1" />
</element>
```

Usamos o exemplo abaixo para definir o nome da visualização que vai aparecer na interface do MAME para ser selecionada na opção **Vídeo** (Tab --> *Opções do vídeo*):

```
<view name="Galaxian Italiano" showpointers="no">
```

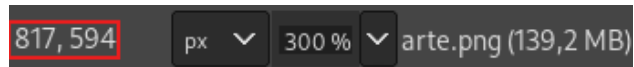
Precisamos informar ao MAME o tamanho exato da imagem, para ver estas informações clique com o mouse direito do mouse em cima dela e selecione **Propriedades** ou abra o arquivo arte.png no *Gimp* e selecione *Imagem --> Propriedades da imagem* para identificar que a imagem tem 3296 x 4093:

```
<element ref="Italiano">
  <bounds x="0" y="0" width="3296" height="4093" />
</element>
```

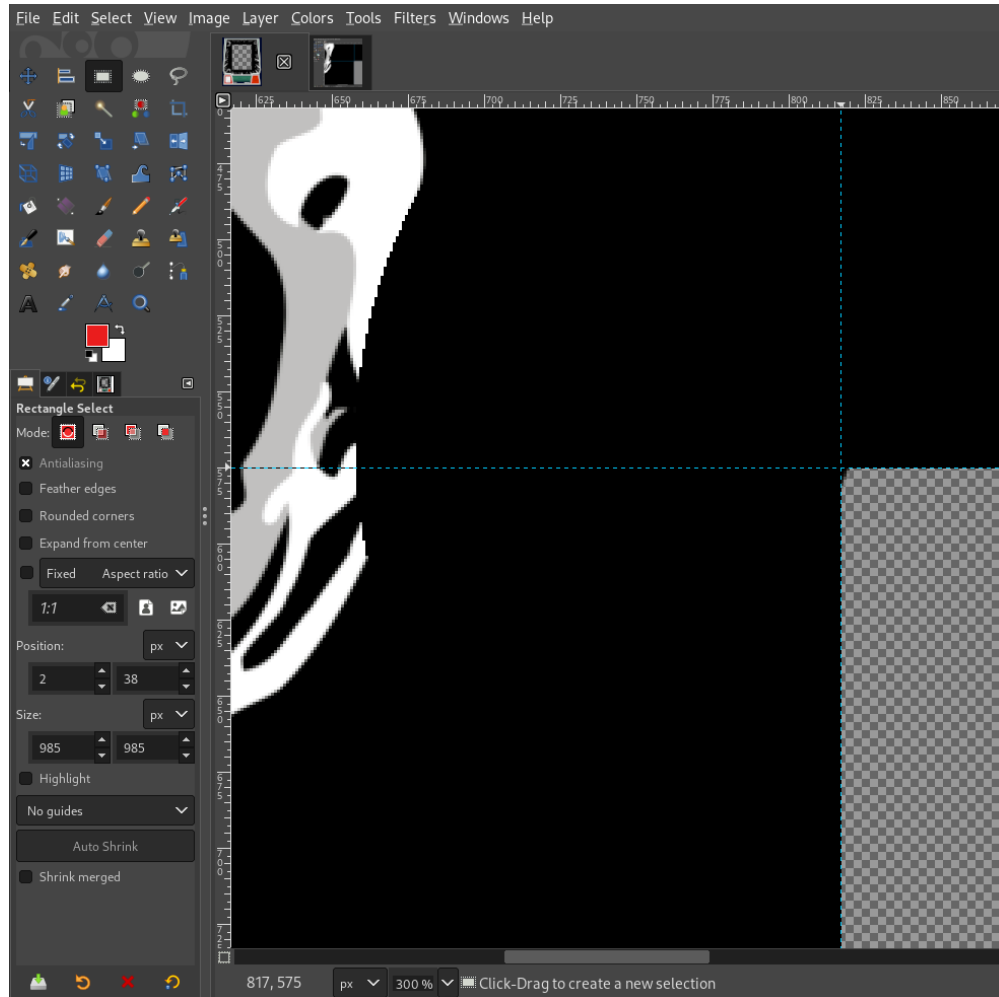
A Posição e a proporção de tela

Para aqueles que nunca trabalharam com gráficos de linhas o **eixo x** vai lidar com as coordenadas da posição **horizontal** e o **eixo y** da **vertical**.

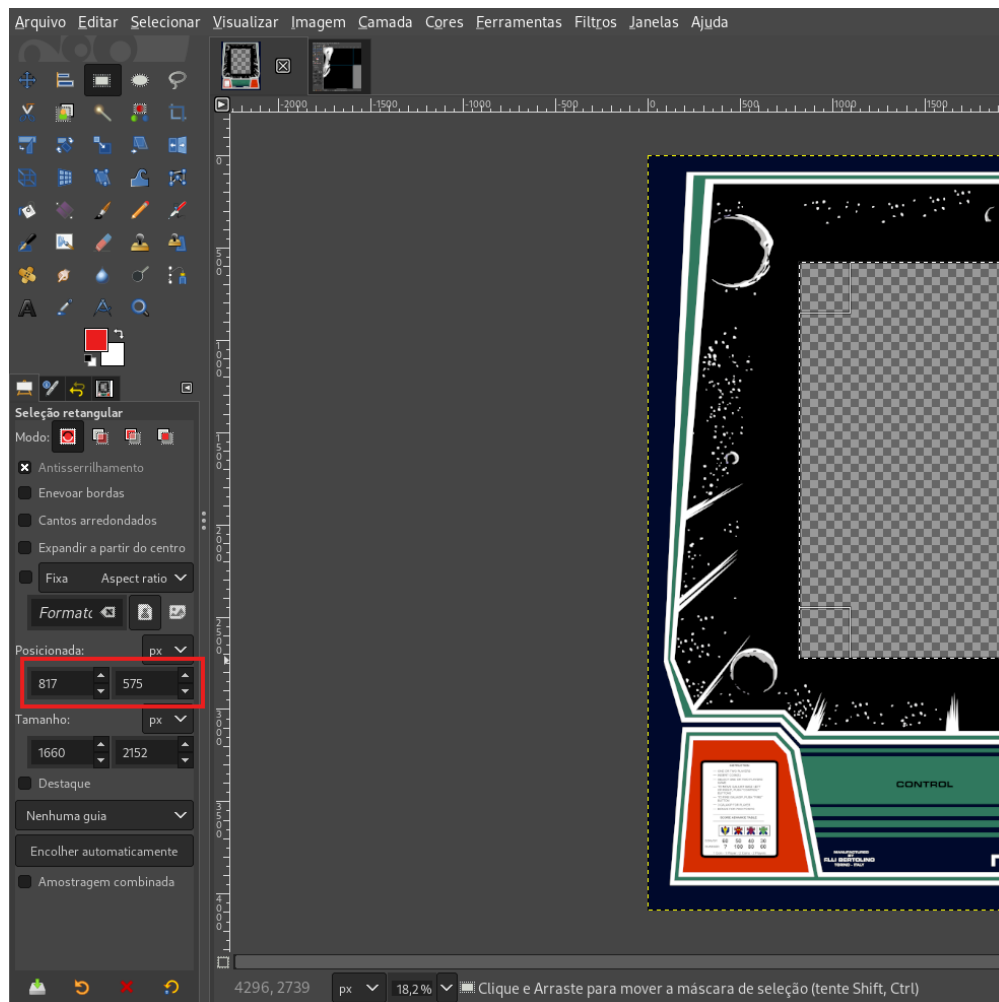
Para descobrir os valores **x** e **y** abra o arquivo **arte.png** no *Gimp*, na parte de baixo da tela próximo ao zoom ficam as coordenadas **x,y** como mostra a imagem abaixo. É dali que obtemos os valores e eles aparecem conforme movimentamos o mouse:



O espaço quadriculado ao centro é a área vazia da imagem, dê um zoom na imagem na casa dos 300% ou mais posicione o mouse na borda entre o quadriculado e a parte preta da imagem do lado **ESQUERDO** para encontrar o valor de **x** e em **CIMA** para encontrar o valor de **y**, para facilitar foram posicionados duas linhas azuis na imagem abaixo indicando a posição que o mouse deve estar para obter as coordenadas que são 817 e 575:



Uma maneira ainda mais fácil de se obter estes valores é pressionando a tecla U do seu teclado para selecionar a "varinha" ou a *Ferramenta de seleção contígua*, clique dentro da área quadriculada da imagem para selecioná-la. Em seguida pressione a tecla R ou a *Ferramenta de seleção retangular* e clique em qualquer região da área quadriculada, no painel à esquerda deve aparecer os mesmos valores para a coordenada x (817) e y (575):



Observe que nem sempre haverá um vazio selecionável na imagem para ser mensurado, assim sendo, utilize a técnica que funcionar melhor com o desenho ou a arte que estiver utilizando.

Com a posição da tela definida agora é necessário dimensioná-la mantendo a sua proporção **4:3** que é o padrão para a maioria dos arcades e telas CRT da época. A tela está invertida com proporção **3:4** então pegue o valor da **largura** como mostra a imagem acima e faça as contas:

```
x = 1660 * (4 / 3)
x = 1660 * 1,333333333
x = 2213
```

Ou utilize a ferramenta disponibilizada em [Ferramentas necessárias](#) para facilitar o cálculo destas dimensões exibindo o mesmo resultado no campo **3:4**.

Insira os valores no campo **verde**, o primeiro campo verde no topo serve como uma fácil identificação da relação de aspecto da tela de um valor qualquer, caso um valor seja inserido no segundo campo será feito o cálculo da largura com base nos dados da primeira linha, se nenhum valor for inserido, nada será calculado. O segundo campo em verde serve para situações como demonstrada acima onde você identifica uma área qualquer e quer saber qual seria a proporção **4:3 ideal** para ela, você insere o valor da altura quando a tela for horizontal ou largura quando a tela estiver na vertical para que a planilha calcule os valores.

	A	B	C	D	E	F	G
1	Width	Height	GCD	Width / GCD	Height / GCD	SAR	PIXELS
2	256	224	32	8	7	8:7	57344
3							
4		Type in the height		1660			
5		Digite a altura					
6		Calculated Width		1897			
7		Largura Calculada					
8				1897 x 1660			
9							
10			DAR 4:3	2213,333333333	2213 x 1660		
11			DAR 3:4	2213,333333333	1660 x 2213		
12			DAR 5:4	2075	2075 x 1660		
13			DAR 4:5	2075	1660 x 2075		
14			DAR 16:9	2951,111111111	2951 x 1660		
15			DAR 9:16	2951,111111111	1660 x 2951		
16							

A planilha também faz o cálculo da largura com o valor do **SAR** (Storage Aspect Ratio ou Relação de Aspecto da Origem), este valor é vulgarmente conhecido como **pixel perfect**. Supondo que você vá fazer um layout para um jogo de um determinado console e queira o tal "pixel perfect", insira uma das resoluções do console no primeiro campo para obter o **SAR** e descobrir o valor da largura com base neste **SAR** em vez de utilizar o **DAR** (Display Aspect Ratio ou a Proporção da Imagem na Tela).

Esta planilha foi criada com a intenção de facilitar os cálculos e para ser usada no desenvolvimento dos layouts, ela não serve para nada muito técnico ou avançado, porém a planilha está aberta, podendo ser alterada para atender qualquer outra necessidade que você venha a ter.

Nota: Sempre que possível, procure usar valores inteiros na definição da resolução da sua tela. Em certos sistemas como a do exemplo acima não haverá qualquer diferença visível, contudo, certos sistemas não são tão tolerantes assim e podem apresentar pixels distorcidos na tela ou até mesmo artefatos estranhos durante o uso de shaders como o CRT-geom por exemplo. Assim, aumente o valor da altura na planilha até quem um valor inteiro seja encontrado.

Se fosse o caso do exemplo acima, o valor ideal seria 1662 x 1899.

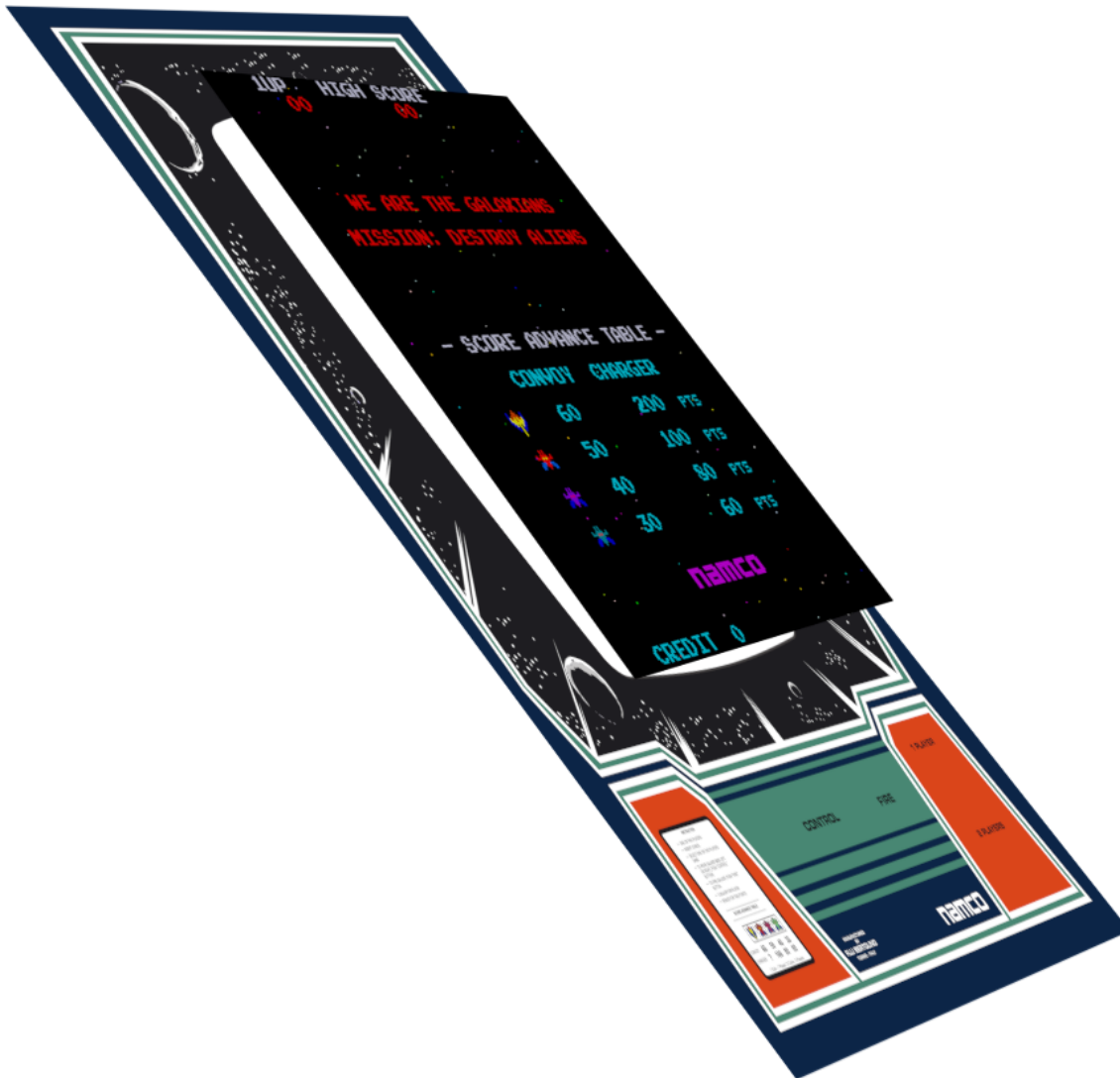
Aviso: Em alguns sistemas que usam diferentes resoluções (como o Neo Geo por exemplo) o design da sua ilustração pode ficar fora das proporções ideais, assim, em vez de utilizar os valores **DAR**, experimente usar os valores **SAR**. Nestes casos muito específicos, pode ser que o mecanismo de ajuste de proporção de tela do MAME faça com que haja bordas pretas em quaisquer um dos lados da tela ou partes da tela podem ficar ocultas pela sua ilustração.

Com o valor calculado em mãos, a resolução final será 1660 x 2213. Assim temos todos os valores para posicionarmos a nossa tela emulada na tela (screen) e a sua correta proporção:

```
<screen index="0">  
  <bounds x="817" y="575" width="1660" height="2213" />  
</screen>
```

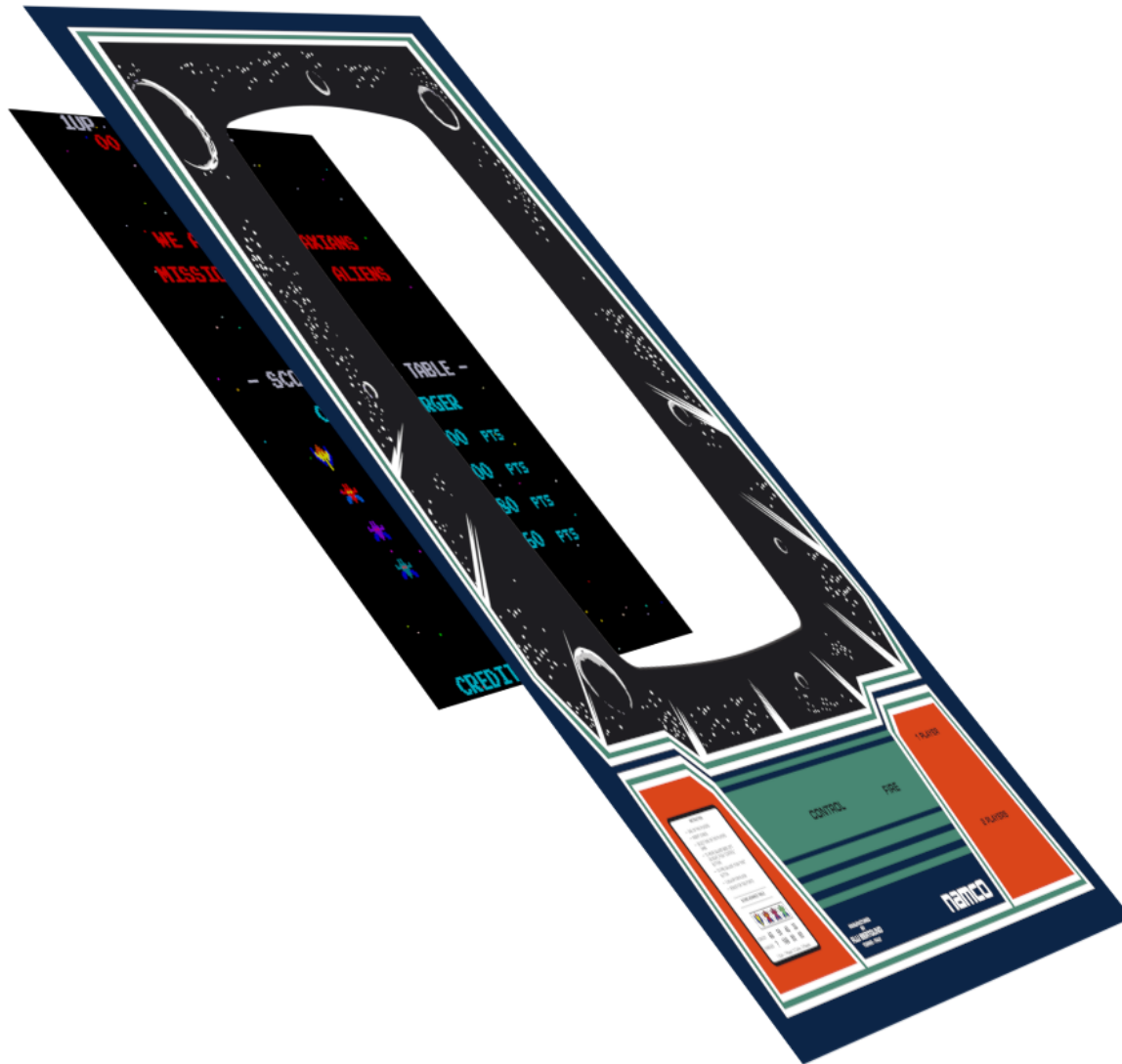
É possível organizar este layout de duas maneiras diferentes dependendo do efeito que você queira dar ao seu design. O MAME organiza o layout em camadas obedecendo a ordem em que elas forem definidas no arquivo de layout, então a composição da sua tela emulada e o gráfico pode começar com a ilustração no fundo e a tela emulada em cima desta arte:

```
<element ref="Italiano">
  <bounds x="0" y="0" width="3296" height="4093" />
</element>
<screen index="0">
  <bounds x="817" y="575" width="1660" height="2213" />
</screen>
```



Ou ao contrário, com a arte gráfica por cima da tela:

```
<screen index="0">  
  <bounds x="817" y="575" width="1660" height="2213" />  
</screen>  
<element ref="Italiano">  
  <bounds x="0" y="0" width="3296" height="4093" />  
</element>
```



Os motivos de se escolher um ou outro depende do efeito final que você queira dar na tela. A arte que estamos utilizando tem uma área preta na região da tela fazendo com que ambos se misturem quase que criando uma composição de um fundo infinito, porém caso eu queira dar um efeito de tela recortada como uma tela CRT eu usaria a segunda opção para aproveitar o recorte do design.

Na Galaxian não dá para perceber este recorte da tela pois tanto a arte quanto a tela são pretas, porém com outros sistemas é possível por exemplo, colocar uma moldura em volta da tela com efeitos de sombra, como mostra [este exemplo](#):



Neste caso nós definimos a tela primeiro, a arte e por último a moldura (screen_bezel) da tela já com alguns efeitos de sombra e transparência:

```
<screen index="0">
  <bounds x="292" y="43" width="1340" height="997" />
</screen>
<element ref="Artwork_1">
  <bounds x="0" y="0" width="1920" height="1080" />
</element>
<element ref="screen_bezel">
  <bounds x="272" y="0" width="1376" height="1080" />
</element>
```

Repare que os efeitos da sombra e da transparência da moldura aparecem perfeitamente sobre a tela emulada. Neste caso específico a tela foi um pouco esticada horizontalmente para cobrir as barras pretas que aparecem nos cantos da tela.

Posicionando os botões na tela

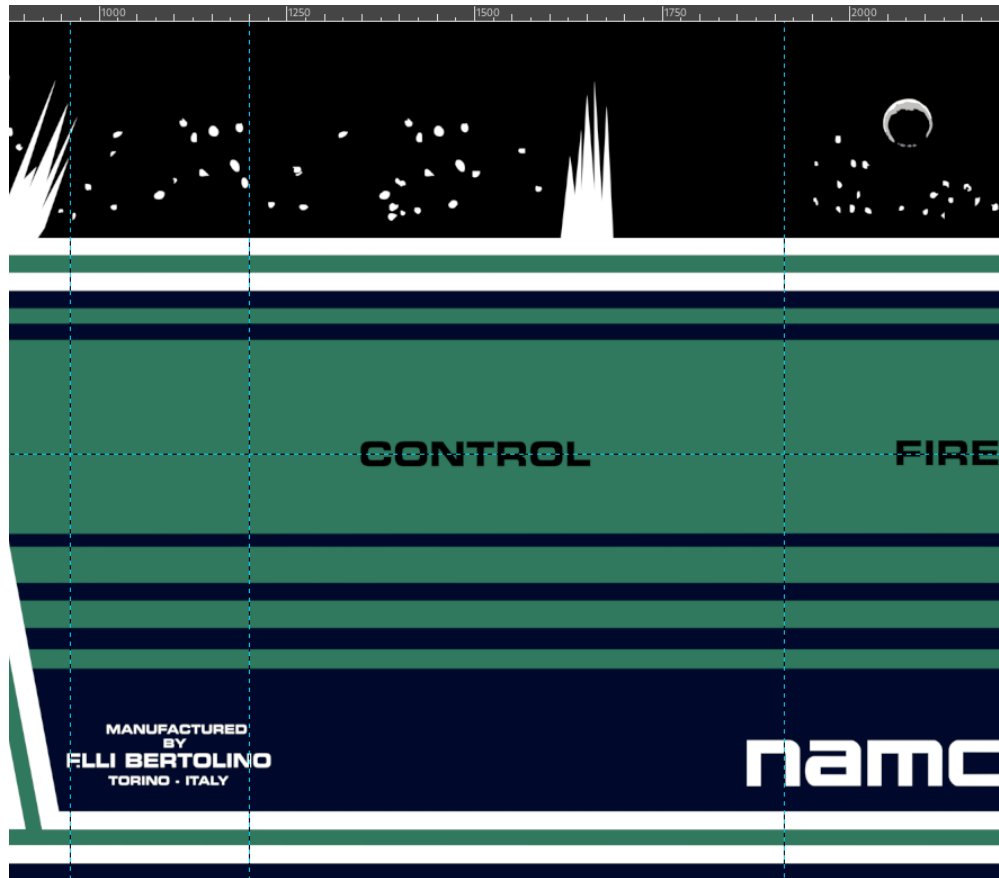
Para posicionar qualquer outra imagem na tela como botões, controles ou o que quer que seja, será preciso estar com a arte que será usada no MAME já aberta no *Gimp*, pegue a imagem de um botão por exemplo e arraste para a tela do *Gimp*, ele deverá aparecer como uma camada, posicione-o na região desejada e se for o caso redimensione-o usando a ferramenta de redimensionamento Shift + S, para o botão escolhi o tamanho de 190 **px**. Quando concluir selecione a camada do botão com o botão direito do mouse e escolha *Alfa para a seleção*, em seguida pressione R ou escolha *Ferramenta de seleção retangular* e clique na imagem do botão, na barra à esquerda já deve aparecer as informações do tamanho e a posição na tela que seria x 1105 e y 3314 com o tamanho de 190 x 190.



Com todos os botões posicionados e com os valores em mãos, temos então a seguinte configuração para o botão esquerda, direita e disparo:

```
<element ref="esquerda">
    <bounds x="867" y="3313" width="190" height="190" />
</element>
<element ref="direita">
    <bounds x="1105" y="3313" width="190" height="190" />
</element>
<element ref="disparo">
    <bounds x="1819" y="3313" width="190" height="190" />
</element>
```

Outra maneira de se posicionar os botões na tela é utilizar os valores centralizados, no modo descrito anteriormente a referência utilizada como coordenadas é o limite da imagem do lado **esquerdo** ou **x** e o limite do **topo** da imagem ou **y**. Os valores centralizados utilizam exatamente a posição do ponteiro do mouse na tela e por isso tais coordenadas são definidas como **xc** e **yc** como mostra a figura abaixo:



Posicione o ponteiro do mouse bem em cima onde as linhas se cruzam para ver as coordenadas na parte de baixo da tela do *Gimp*, fica mais fácil fazer um zoom com 300% ou mais, assim os valores podem ser encontrados de forma mais precisa, para isso, posicione o ponteiro no ponto desejado, mantenha pressionado Ctrl e movimente a roda do mouse para cima para aplicar o zoom na região do ponteiro do mouse:

```
<element ref="esquerda">
  <bounds xc="962" yc="3407" width="190" height="190" />
</element>
<element ref="direita">
  <bounds xc="1201" yc="3407" width="190" height="190" />
</element>
<element ref="disparo">
  <bounds xc="1913" yc="3407" width="190" height="190" />
</element>
```

Conectando os botões e ativando as suas funções lógicas

A referência "ref" esquerda, direita e disparo são os nomes dos conjuntos das imagens definidos lá no começo, durante o início da emulação o MAME identifica o defstate (condição/estado inicial) inicial, como o seu valor é 0 (zero), a primeira imagem que aparece será aquela que estiver definida como state=0. Quando acionarmos o botão e a sua condição mudar para state=1 o MAME carregará a imagem definida como state=1, a mecânica para todo este processo é bem simples:

```
<element name="disparo" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>
```

Para conectar os botões e para dar as suas funções lógicas é necessário encontrar os valores para inputtag e inputmask onde inputtag é o nome da porta usada pelo sistema para os controles e botões do jogador 1, 2, etc. Já inputmask é o valor usado pelo sistema para definir os valores hexadecimais dos comandos, botões, etc.

Para encontrar estes valores, inicie o sistema Galaxian:

```
mame galaxian
```

Pressione Tab e vá em *Atribuições da entrada (este sistema)*. Selecione *P1 Left*, no teclado clique na tecla Del para excluir o valor e usando o seu controle, joystick ou teclado, clique no direcional para o lado esquerdo, faça o mesmo para *P1 Right* e *P1 Button 1*, quando terminar, pressione Tab seguido de Esq para encerrar a emulação.

Será criado dentro do diretório **cfg** um arquivo chamado galaxian.cfg, abra-o num editor de texto e veja que para cada configuração feita para os controles e para o botão há um valor específico para eles, aqui um exemplo usando um controle de *Playstation 2*:

```
<input>
  <port tag=":IN0" type="P1_JOYSTICK_LEFT" mask="4" defvalue="0">
    <newseq type="standard">
      JOYCODE_1_XAXIS_LEFT_SWITCH
    </newseq>
  </port>
  <port tag=":IN0" type="P1_JOYSTICK_RIGHT" mask="8" defvalue="0">
    <newseq type="standard">
      JOYCODE_1_XAXIS_RIGHT_SWITCH
    </newseq>
  </port>
  <port tag=":IN0" type="P1_BUTTON1" mask="16" defvalue="0">
    <newseq type="standard">
      JOYCODE_1_BUTTON3
    </newseq>
  </port>
</input>
```


Para o botão de disparo *P1_BUTTON1* por exemplo, temos o valor da porta `tag="IN0"` que usaremos em `inputtag` e o valor `mask="16"` que usaremos em `inputmask`, assim a nossa configuração fica assim:

```
<element ref="esquerda" inputtag="IN0" inputmask="4">
    <bounds x="867" y="3313" width="190" height="190" />
</element>
<element ref="direita" inputtag="IN0" inputmask="8">
    <bounds x="1105" y="3313" width="190" height="190" />
</element>
<element ref="disparo" inputtag="IN0" inputmask="16">
    <bounds x="1819" y="3313" width="190" height="190" />
</element>
```

Na primeira definimos que vamos associar a imagem **esquerda** na entrada `IN0` e que seu código (máscara) para este botão é `4` e assim sucessivamente, com isso nós conectamos e damos funções para as imagens na parte lógica do sistema fazendo com que o MAME passe a interpretá-las de forma animada na tela quando o botão for pressionado no seu joystick ou seja acionado na tela quando for clicado pelo mouse.

Observe que não é preciso copiar os dois pontos **iniciais** existentes em `tag="IN0"`, copie apenas o seu valor `IN0`. Cada sistema possui a sua configuração específica, no caso dos sistemas Neo Geo por exemplo a **tag** aparece como `tag="edge:joy:JOY1"`, apenas ignore os dois pontos iniciais e copie **todo** o resto, ou seja, a nossa `inputtag` para os sistemas Neo Geo ficaria `inputtag="edge:joy:JOY1"`.

Alertas e retornos externos

O sistema *Galaxian* possui uma porta que sinaliza através de um sinal luminoso quando 1 crédito é inserido, o sinal da lâmpada passa a piscar indicando que o jogador deve clicar nele para iniciar a partida para o jogador 1 e o mesmo ocorre quando dois créditos são inseridos indicando o início da partida para o jogador 1 ou para o jogador 2. A versão do sistema *Galaxian Italiana* não possui tais luzes porém vamos adicioná-las mesmo assim para fins didáticos.

Usando o *Gimp*, posicione os botões para o jogador 1 e para o jogador 2, escolhemos o tamanho de `200 x 200 px` e então chegamos na seguinte configuração:

```
<element ref="J1" inputtag="IN1" inputmask="1">
    <bounds xc="2670" yc="3408" width="200" height="200" />
</element>
<element ref="J2" inputtag="IN1" inputmask="2">
    <bounds xc="2790" yc="3753" width="200" height="200" />
</element>
```

Execute o MAME no terminal ou prompt de comando com a opção abaixo:

```
mame -output console galaxian
```

Insira 2 créditos na sistema, repare no terminal que é possível ver os valores `lamp0` e `lamp1` se alternando entre `0` (desligado) e `1` (ligado), são estes sinais que usaremos para fazer piscar uma imagem na tela simulando uma lâmpada e indicando os créditos para os respectivos jogadores.

É possível montar este layout de duas maneiras, posicionando a nossa lâmpada uma a uma ou utilizando cont em conjunto com os *Parâmetros*. Imagine o seguinte cenário, você está montando um layout onde precisa colocar `lamp0` até `lamp100` na tela, é possível fazer isso de forma manual ou usar parâmetros para facilitar escrevendo poucas linhas e deixando que o MAME lide com o resto.

Dada a simplicidade do sistema *Galaxian* não é preciso usar qualquer parâmetro porém usaremos mesmo assim para que fique fácil a compreensão de como isso funciona.

São duas lâmpadas lamp0 e lamp1 então contamos até 2:

```
<repeat count="2">
```

No parâmetro nós definimos que a variável i inicie uma contagem, que comece a partir do 0 (zero) e termine em 1.

```
<param name="i" start="0" increment="1" />
```

A nossa "lâmpada" será um anel azul que vai aparecer e sumir na tela simulando um pisca-pisca. Definimos os parâmetros para a posição deste primeiro anel azul (pisca) no eixo x (horizontal), x é o nome da nossa variável e esta pode ter qualquer nome, escolhemos x para facilitar a identificação do eixo horizontal, start define a posição inicial do anel que fica bem em cima da posição do botão J1 (Jogador 1) como descrito na configuração alguns parágrafos atrás, não precisamos fazer nenhum deslocamento no eixo x, portanto o valor de increment é zero.

```
<param name="x" start="2790" increment="0" />
```

Fazemos o mesmo para o eixo y (vertical), para que o segundo anel azul possa ficar alinhado exatamente onde queremos realizado o cálculo abaixo. O cálculo serve para descobrir quantos pixels a mais eu preciso para deslocar o anel azul da primeira coordenada no eixo y a partir da posição do J1 (3408) até a segunda coordenada vertical do eixo y (3753) do J2. A conta foi organizada para que o minuendo tenha o maior valor que o subtraendo para que se obtenha um número inteiro positivo:

```
increment = J2 y - J1 y
increment = 3753 - 3408
increment = 345
```

Definimos uma variável y para facilitar a identificação do eixo y (vertical), nós começamos na coordenada 3408 e para que o anel azul fique em cima do botão J2 precisamos deslocá-lo 345 pixels.

```
<param name="y" start="3408" increment="345" />
```

Aqui é onde toda a mágica acontece, usaremos as variáveis que definimos para que elas sejam substituídas automaticamente pelos seus respectivos valores, aqui o valor da variável ~i~ será automaticamente substituído por 0 e 1, então lamp~i~ se transforma em lamp0 e lamp1 respectivamente, repare que utilizamos apenas 1 linha para isso, se fosse o caso para 100 lamps seria a mesma coisa:

```
<element name="lamp~i~" ref="pisca">
```

Para ficar fácil a compreensão do resultado, a linha acima se traduziria desta forma para o MAME:

```
<element name="lamp0" ref="pisca">
<element name="lamp1" ref="pisca">
```

Pisca está definido com os parâmetros:

```
<element name="pisca" defstate="0">
    <image file="pisca1.png" state="1" />
</element>
```

O valor de lamp0 sempre retorna 0 (zero) quando estiver desligado e 1 (um) quando estiver ligado, o mesmo acontece com lamp1, como não é preciso definir nada para quando lamp0 e lamp1 estiverem desligados basta definir apenas a condição de ligado, portanto defstate="0" estabelece a sua condição inicial é 0 ou não faça nada, o anel azul só vai aparecer caso a sua condição mude para 1 através do state="1".

As variáveis `~x~` e `~y~` são substituídas pelos seus respectivos valores que foram definidos anteriormente fazendo com que a imagem do anel azul seja posicionada corretamente no eixo **x** (horizontal) e no eixo **y** (vertical), assim como definimos que o seu tamanho será pouca coisa maior que o desenho do botão para que ele cubra um pouco a mais dos limites do botão, assim o valor seria de 250 **pixels** de largura por 250 **pixels** de altura.

```
...  
    <bounds xc=~x~ yc=~y~ width="250" height="250" />  
</element>
```

O atributo abaixo encerra as definições para repeat

```
</repeat>
```

Finalizamos o nosso layout com os dois últimos itens:

```
</view>  
</mamelayout>
```

Este é o nosso arquivo completo:

```
<?xml version="1.0"?>
<!-- Layout file created by: Wellington Terumi Uemura
      Bezel design by: Etienne MacGyver
      Round Buttons design by: u/cd4053
      License: CC by 4.0
      https://mamedoc.readthedocs.io/pt/latest/techspecs/layout_files.html
-->
<mamelayout version="2">
<element name="Italiano">
  <image file="arte.png" />
</element>
<element name="J1" defstate="0">
  <image file="btn0.png" state="0" />
  <image file="btn1.png" state="1" />
</element>
<element name="J2" defstate="0">
  <image file="btn0.png" state="0" />
  <image file="btn1.png" state="1" />
</element>
<element name="disparo" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>
<element name="esquerdo" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>
<element name="direito" defstate="0">
  <image file="vermelho0.png" state="0" />
  <image file="vermelho1.png" state="1" />
</element>
<element name="pisca" defstate="0">
  <image file="pisca1.png" state="1" />
</element>
<view name="Galaxian Italiano" showpointers="no">
<element ref="Italiano">
  <bounds x="0" y="0" width="3296" height="4093" />
</element>
<screen index="0">
  <bounds x="817" y="575" width="1660" height="2213" />
</screen>
<element ref="J1" inputtag="IN1" inputmask="1">
  <bounds xc="2790" yc="3408" width="200" height="200" />
</element>
<element ref="J2" inputtag="IN1" inputmask="2">
  <bounds xc="2790" yc="3753" width="200" height="200" />
</element>
<element ref="esquerdo" inputtag="IN0" inputmask="4">
  <bounds xc="962" yc="3407" width="190" height="190" />
</element>
<element ref="direito" inputtag="IN0" inputmask="8">
```

(continua na próxima página)

(continuação da página anterior)

```

        <bounds xc="1201" yc="3407" width="190" height="190" />
</element>
<element ref="disparo" inputtag="IN0" inputmask="16">
    <bounds xc="1913" yc="3407" width="190" height="190" />
</element>
<repeat count="2">
    <param name="i" start="0" increment="1" />
    <param name="x" start="2790" increment="0" />
    <param name="y" start="3408" increment="345" />
    <element name="lamp~i~" ref="pisca">
        <bounds xc="~x~" yc="~y~" width="250" height="250" />
    </element>
</repeat>
</view>
</mamelayout>

```

Colocando para funcionar e adicionando efeitos na tela

Para ver a nossa criação funcionar, dentro do diretório do MAME existe o diretório **artwork**, dentro deste diretório crie um novo chamado **galaxian**, dentro deste diretório coloque o arquivo acima com o nome `default.lay` e todos os arquivos ***.png**. Rode o mame com o comando:

```
mame -window galaxian
```

A tela deve aparecer com a arte e todos os botões em seus respectivos lugares, experimente clicar nos botões com o mouse e verá que eles reagem ao seu clique, ao adicionar 2 créditos o nosso anel azul deve aparecer simulando o piscar de uma luz.



O MAME aceita trabalhar com o diretório como está ou então comprimindo-o em formato .zip ou .7zip. Ao usar a arte como um diretório permite que você altere alguma coisa enquanto realiza testes, caso altere o arquivo ou alguma imagem, é possível recarregar todos eles sem precisar encerrar o MAME pressionando Shift + F3.

Caso queira aplicar shaders na tela para dar um efeito de uma tela CRT como mostra a foto, baixe [este arquivo](#), descompacte-o dentro de um diretório chamado **glsl**. Vá até o diretório **ini** e crie o arquivo **galaxian.ini** e dentro dele adicione estas opções:

```
video                opengl
gl_glsl              1
gl_glsl_filter        1

# No Windows use
glsl_shader_mame0 glsl\osd\CRT-geom

# No Linux use
glsl_shader_mame0 glsl/osd/CRT-geom
```

Ao rodar o *galaxian* novamente a tela terá uma aparência de um CRT, caso queira manter a aparência porém eliminando um pouco a curvatura da tela, edite o arquivo **CRT-geom.vsh** e use estas configurações:

```
// START of parameters

// gamma of simulated CRT
CRTgamma = 2.2;
// gamma of display monitor (typically 2.2 is correct)
monitorgamma = 2.2;
// overscan (e.g. 1.02 for 2% overscan)
overscan = vec2(1.01,1.01);
// aspect ratio
aspect = vec2(1.0, 0.75);
// lengths are measured in units of (approximately) the width of the monitor
// simulated distance from viewer to monitor
d = 2.0;
// radius of curvature
R = 10.0;
// tilt angle in radians
// (behavior might be a bit wrong if both components are nonzero)
const vec2 angle = vec2(0.01,0.01);
// size of curved corners
cornersize = 0.03;
// border smoothness parameter
// decrease if borders are too aliased
cornersmooth = 1000.0;

// END of parameters
```

Altere o valor do overscan de `vec2(1.01,1.01)` para `vec2(1.00,1.00)` para que a tela apareça completa na tela.

Adicionando diagonais nos controles com 8 direções usando o inputraw

Alguns sistemas possuem controles simples como a própria **Galaxian** tem apenas 2 direções, **esquerda** e **direita**, outras como **Pacman** possuem 4 direções, **cima**, **baixo**, **esquerda**, **direita**. Nestes casos a criação de um layout animado para o controle por exemplo é bem simples, basta adicionar uma imagem para cada posição associando-a com os seus respectivos inputtag e inputmask para que seja possível ver a sua animação na tela quando cada uma das posições forem acionadas, no entanto a coisa muda um pouco quando se trata de um sistema com um joystick com 8 direções (ou mais).

Nestes casos, quando qualquer uma das diagonais é acionada o MAME sobrepõem outras imagens uma em cima da outra, caso a diagonal superior direita seja acionada no controle, o MAME irá exibir a imagem para a diagonal superior direita em cima da imagem **cima** e em cima da imagem **direita** ou seja, em vez de aparecer apenas uma imagem para a diagonal escolhida irão aparecer outras duas. Na internet diferentes pessoas encontraram diferentes soluções para o problema, algumas criaram imagens na diagonal com um tamanho suficiente para cobrir as duas outras imagens dentre diversas outras soluções.

Outra maneira de lidar com o problema é utilizar o inputraw, com esta opção o MAME faz a leitura dos dados diretamente da entrada do controle, vamos aproveitar o layout que acabamos de criar para o sistema **Galaxian** e ver como isso ficaria modificando a lógica para os controles e adicionando 3 imagens para simular um joystick:

```
<element name="controle" defstate="0xc">
  <image file="esquerda.png" state="0x1" />
  <image file="direita.png" state="0x2" />
  <image file="centro.png" state="0x0" />
</element>
```

Assim como foi feito anteriormente, usamos o *Gimp* para posicionar o joystick na região que queremos que ele apareça, com as coordenadas em mãos nós montamos toda a lógica dele:

```
<element ref="controle" inputtag="IN0" inputmask="0xc" inputraw="yes">
  <bounds x="1010" y="3260" width="250" height="250" />
</element>
```

Agora em vez de 2 botões vermelhos nós temos um joystick preto na posição que escolhemos e se move conforme nós mexemos nos controles. Fica claro que a quantidade de linhas necessárias para que esta animação aconteça se reduz deixando o layout mais enxuto e simplificado.

Note que ao clicar com o mouse em cima do controle ele faz um movimento indesejado, para evitar essa anomalia adicionamos uma camada de "nada" com o mesmo tamanho da imagem do controle e exatamente na mesma posição:

```
<element name="nada" defstate="0">
  <text string=" " />
</element>
```

E aplicamos isso **antes** do controle com inputmask="0x00" pois o MAME interpreta estes elementos na ordem que eles aparecem no layout, assim temos a camada "nada" sendo construída primeiro seguida do controle logo abaixo desta camada, bloqueando qualquer interação do mouse:

```
<element ref="nada" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
  <bounds x="1010" y="3260" width="250" height="250" />
</element>
<element ref="controle" inputtag="IN0" inputmask="0xc" inputraw="yes">
  <bounds x="1010" y="3260" width="250" height="250" />
</element>
```


A partir da versão **0.265** do MAME prefira fazer desta maneira, exemplo:

```
<element name="cobertura" defstate="0">
    <rect><color alpha="0" /></rect>
</element>
...
...
<element ref="cobertura" clickthrough="no">
    <bounds x="1010" y="3260" width="250" height="250" />
</element>
<element ref="controle" inputtag="IN0" inputmask="0xc" inputraw="yes">
    <bounds x="1010" y="3260" width="250" height="250" />
</element>
```

Ambos funcionam mas o atributo `clickthrough` deixam as coisas mais fáceis. Para obter mais detalhes consulte *Itens que podem ser clicados*.

Ao executar o sistema novamente o controle não mais responde aos cliques do mouse.










Como estamos trabalhando com dados vindos diretamente dos controles, é necessário encontrar os valores para `defstate`, `inputmask` e para `state`. O `defstate` e o `inputmask` utilizam o mesmo valor, este valor **precisa ser calculado**, para isso acessamos o [código-fonte do driver Galaxian](#), bem na linha L3011 de cara já temos o nosso `inputtag` com o valor `IN0` que utilizamos acima, observe quem nem sempre o valor do `inputtag` está disponível assim tão fácil, o driver **Playstaion** (`psx`) por exemplo utiliza `port1:digital_pad:PSXPAD0`, já o sistema **Neo Geo** usa `edge:joy:JOY1` e assim por diante. Para casos como estes é preferível neste caso é preferível utilizar a maneira descrita no capítulo *Conectando os botões e ativando as suas funções lógicas*.

A **Galaxian** tem apenas duas direções, esquerda e direita, preste atenção ao valor `0x04` para `IPT_JOYSTICK_LEFT` e `0x08` para `IPT_JOYSTICK_RIGHT`. Abra a calculadora do seu celular ou do seu sistema operacional em modo programador ou qualquer função que consiga somar valores em **hexadecimais** e some os valores `4 + 8` para obter `c` ou `0xc`, este é o valor do nosso `defstate` e `inputmask` **específicos para este sistema**, dependendo do sistema este valor pode mudar, é possível ter uma ideia vendo a tabela logo abaixo.

Já os valores para `state` vão de `0x0` até `0xf` nos sistemas mais comuns, porém em outros sistemas mais complexas como os sistemas de corrida de carro e de moto que usam um volante ou guidão, os valores vão muito além disso. O assunto foge ao escopo do que estamos apresentando aqui então recomendo o belo trabalho do [Mr.Do](#) e de todas as pessoas que colaboram com aquele projeto, baixe o arquivo **hangon.ZIP** e veja um exemplo de até onde o nível de complexidade pode chegar.

Até o presente momento não há uma ferramenta que ajude a obter os valores para `state` e portanto eles devem ser encontrados de forma **manual** o que dá um certo trabalho, para facilitar, veja o exemplo da tabela abaixo com os valores para alguns sistemas:

Tabela 1: Códigos dos direcionais de alguns drivers

Driver	defstate									
cps1	0xf	0x7	0x6	0xe	0xa	0xb	0x9	0xd	0x5	0xf
cps2	0xf	0x7	0x6	0xe	0xa	0xb	0x9	0xd	0x5	0xf
cps3	0xf	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
ddragon	0xf	0xb	0xa	0xe	0x6	0x7	0x5	0xd	0x9	0xf
dkong	0xf	0x4		0x1		0x8		0x2		0x0
galaxian	0xc			0x1				0x2		0x0
kinst	0x3c0	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
m72/m92	0xf	0x7	0x6	0xe	0xa	0xb	0x9	0xd	0x5	0xf
midtunit	0xf	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
midwunit	0xf	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
model1	0xf0	0xd	0x9	0xb	0xa	0xe	0x6	0x7	0x5	0xf
nemesis	0xf	0xb	0x9	0xd	0x5	0x7	0x6	0xe	0xa	0xf
neogeo	0xf	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
pacman	0xf	0xe		0xb		0x7		0xd		0xf
psx	0xf0	0xe	0xc	0xd	0x9	0xb	0x3	0x7	0x6	0xf
segas16a	0xf0	0xd	0x9	0xb	0xa	0xe	0x6	0x7	0x5	0xf
segas18	0xf0	0xd	0x9	0xb	0xa	0xe	0x6	0x7	0x5	0xf
seibuspi	0xf	0xe	0x6	0x7	0x5	0xd	0x9	0xb	0xa	0xf
simpsons	0xf	0xb	0x9	0xd	0x5	0x7	0x6	0xe	0xa	0xf
stv	0xf0	0xd	0x9	0xb	0x6	0xe	0x6	0x7	0x5	0xf
tmnt	0xf	0xb	0x9	0xd	0x5	0x7	0x6	0xe	0xa	0xf
toaplan2	0xf	0x1	0x9	0x8	0xa	0x2	0x6	0x4	0x5	0x0

Aqui vai a **dica de ouro** para quem for criar os seus próprios layouts com o método inputraw, se ao carregar um layout a imagem usada para o centro (posição neutra do controle) **não aparecer** ou se durante o acionamento do direcional uma das posições sumir da tela, significa que:

- O valor do inputtag está errado.
- O valor do state está errado.
- O valor do defstate não foi calculado corretamente.

Todos estes valores variam de sistema para sistema, em determinados sistemas por exemplo, ainda que o valor inputtag esteja incorreto a imagem do controle relacionada ao ponto neutro ou **centro** poderá aparecer ou não e em outras vezes apesar da imagem do ponto neutro aparecer, não haverá qualquer animação do controle na tela. Para a maioria dos sistemas avaliadas o state para o ponto neutro funciona com o valor 0xf, contudo há alguns sistemas onde o valor precisa ser alterado para 0x0 para que funcione ou algum outro valor onde este irá depender do tipo do sistema emulado. Quando você movimenta os direcionais e nota que o controle fica piscando na tela, este é um bom indicativo que você está no caminho certo pois indica que a combinação inputtag e inputmask está correta e apenas os valores para o state estão errados.

Nos casos onde a animação aparece invertida na tela como por exemplo se ao clicar para cima o controle na tela aparece para baixo e assim por diante, basta inverter os valores dos seus respectivos state e clicar em Shift + F3 para recarregar o arquivo e ver como ficou.

Há casos onde nenhum dos valores da tabela mostrada acima vai funcionar com o sistema que você estiver desenvolvendo o layout, mesmo que os valores estejam corretos para inputtag, defstate e inputmask. Nestes casos inicie a sua configuração com todos os valores state zerados:

```
<element name="controle"                defstate="0xf">
  <image file="cima.png"                 state="0x0" />
  <image file="baixo.png"                 state="0x0" />
  <image file="esquerda.png"             state="0x0" />
  <image file="direita.png"              state="0x0" />
  <image file="centro.png"               state="0xf" />
</element>
```

Comece alterando o valor state para o comando **cima**, comece com 0x1 e vá subindo, clique Shift + F3 a cada alteração para ver se alteração surtiu algum efeito. É muito comum encontrar um determinado valor onde você aciona o controle para cima e ocorre uma animação para qualquer outra direção, supondo que a animação que apareceu foi para baixo, altere o valor do state relacionado com **baixo.png** e continue incrementando o valor até chegar em 0xf, depois comece novamente a partir de 0x1. Observe que alguns valores já testados e que não retornaram nenhuma animação agora passam a responder, continue definindo os valores encontrados para as suas respectivas posições até encontrar todos os valores.

Sistemas com dois ou mais controles

Em sistemas com dois ou mais controles como a **Neo Geo** por exemplo, nós fazemos assim:

```
<element name="controle" defstate="0xf">
  <image file="baixodireito.png" state="0x5" />
  <image file="cimadireita.png" state="0x6" />
  <image file="direita.png" state="0x7" />
  <image file="baixoesquerdo.png" state="0x9" />
  <image file="cimaesquerda.png" state="0xa" />
  <image file="esquerda.png" state="0xb" />
  <image file="baixo.png" state="0xd" />
  <image file="cima.png" state="0xe" />
  <image file="centro.png" state="0xf" />
</element>

</element>
<element ref="cobertura" clickthrough="no">
  <bounds x="158" y="794" width="150" height="150" />
</element>
<element ref="controle" inputtag="edge:joy:JOY1" inputmask="0xf" inputraw="yes">
  <bounds x="158" y="794" width="150" height="150" />
</element>
<element ref="cobertura" clickthrough="no">
  <bounds x="665" y="794" width="150" height="150" />
</element>
<element ref="controle" inputtag="edge:joy:JOY2" inputmask="0xf" inputraw="yes">
  <bounds x="665" y="794" width="150" height="150" />
</element>
```

Ambos os controles utilizam os mesmos valores para state e defstate, o que muda é o valor do inputtag onde edge:joy:JOY1 define que este é o controle do **jogador 1** e que edge:joy:JOY2 é o controle do **jogador 2**.

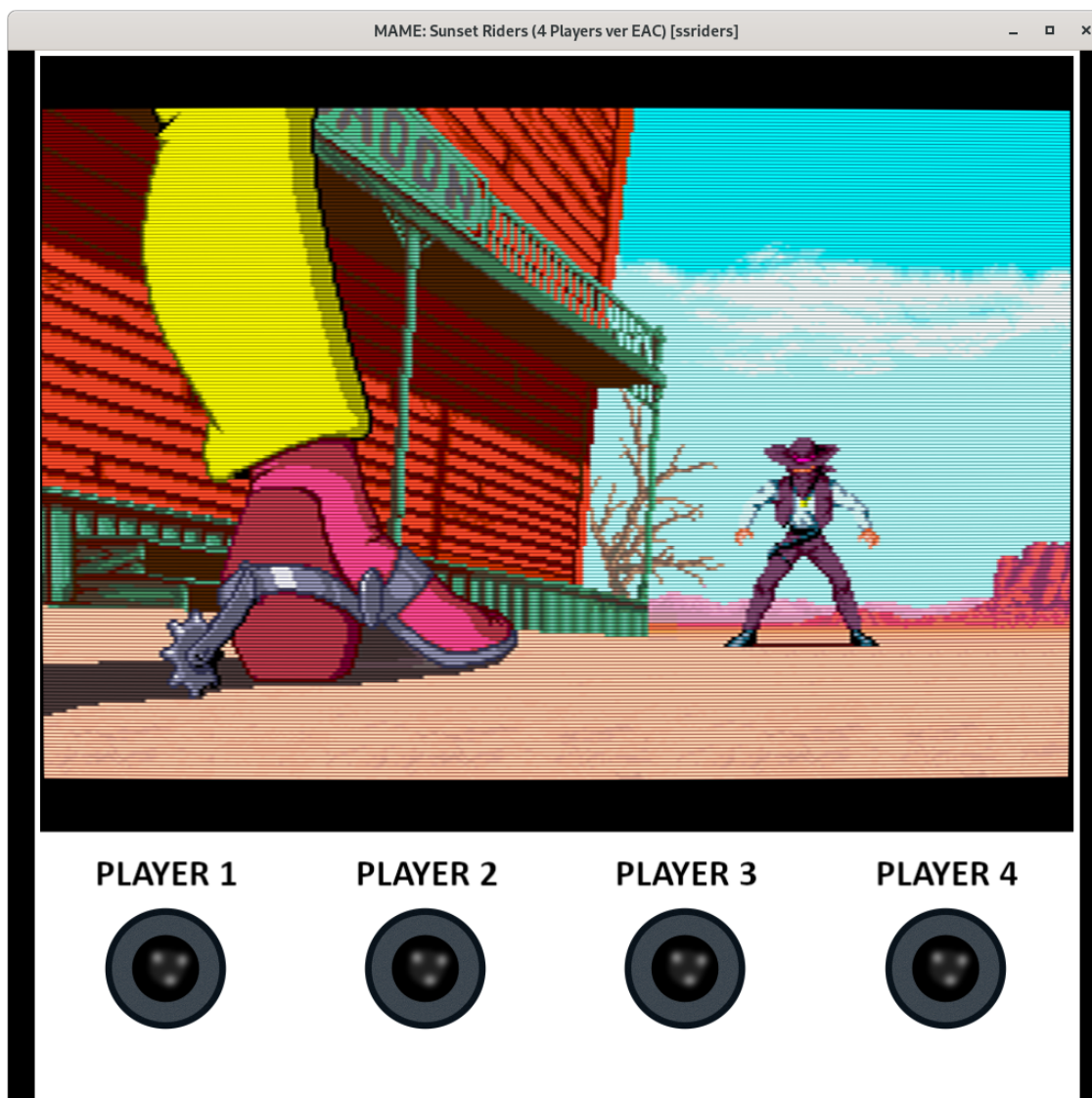
Contudo, há sistemas na **CPS2** que apresentam o mesmo valor IN0 no inputtag para ambos os jogadores, nestes casos devemos recorrer novamente ao código-fonte do MAME para este driver, observando a linha #1501 nós temos os 4 valores para a porta do 2º jogador PORT_PLAYER(2), temos o valor 100 para a direita, 200 para a esquerda, 400 para baixo e 800 para cima. Assim foi feito anteriormente, some todos os valores hexadecimais com uma calculadora compatível para obter o valor f00 ou seja $100 + 200 + 400 + 800 = f00$.

Este é o valor que deve ser usado no inputmask da configuração do controle do 2º jogador como mostra o exemplo abaixo que também funcionam para todas as outros sistemas existentes no driver **CPS2** do MAME:

```
<element name="controle" defstate="0xf">
  <image file="cimaesquerda.png" state="0x5" />
  <image file="cimadireita.png" state="0x6" />
  <image file="cima.png" state="0x7" />
  <image file="baixoesquerda.png" state="0x9" />
  <image file="baixodireita.png" state="0xa" />
  <image file="baixo.png" state="0xb" />
  <image file="esquerda.png" state="0xd" />
  <image file="direita.png" state="0xe" />
  <image file="centro.png" state="0xf" />
</element>

<element ref="cobertura" clickthrough="no">
  <bounds x="158" y="794" width="150" height="150" />
</element>
<element ref="controle_J1" inputtag="IN0" inputmask="0xf" inputraw="1">
  <bounds x="158" y="794" width="150" height="150" />
</element>
<element ref="cobertura" clickthrough="no">
  <bounds x="665" y="794" width="150" height="150" />
</element>
<element ref="controle_J2" inputtag="IN0" inputmask="0xf00" inputraw="1">
  <bounds x="665" y="794" width="150" height="150" />
</element>
```

Utilize o **layout modelo** disponibilizado em [Ferramentas necessárias](#) para realizar testes em diferentes sistemas e obter uma visualização simples e rápida dos controles, depois de baixar e extrair o arquivo, copie-o para o diretório **artwork** e o renomeie com o nome do sistema que será testada, para o exemplo usado na foto o nome do diretório é **ssriders**. Rodando o comando `mame -window ssriders` irá aparecer uma tela com um design branco genérico, será possível escolher um modelo para 2 ou 4 jogadores através do menu *Opções do Vídeo* (TAB --> *Opções do vídeo*), faça as alterações necessárias no arquivo de layout e para visualizar na tela pressione Shift + F3.



Abaixo as teclas predefinidas do MAME para os 4 jogadores, para mais informações consulte [Predefinições para o jogador 1](#).

Tabela 2: Teclas predefinidas do MAME para o controle dos jogadores

1º Jogador	Direcional cima	Direcional baixo	Direcional esquerda	Direcional direita
2º Jogador	R	F	G	D
3º Jogador	I	K	J	L
4º Jogador	Teclado num. 8	Teclado num. 2	Teclado num. 4	Teclado num. 6

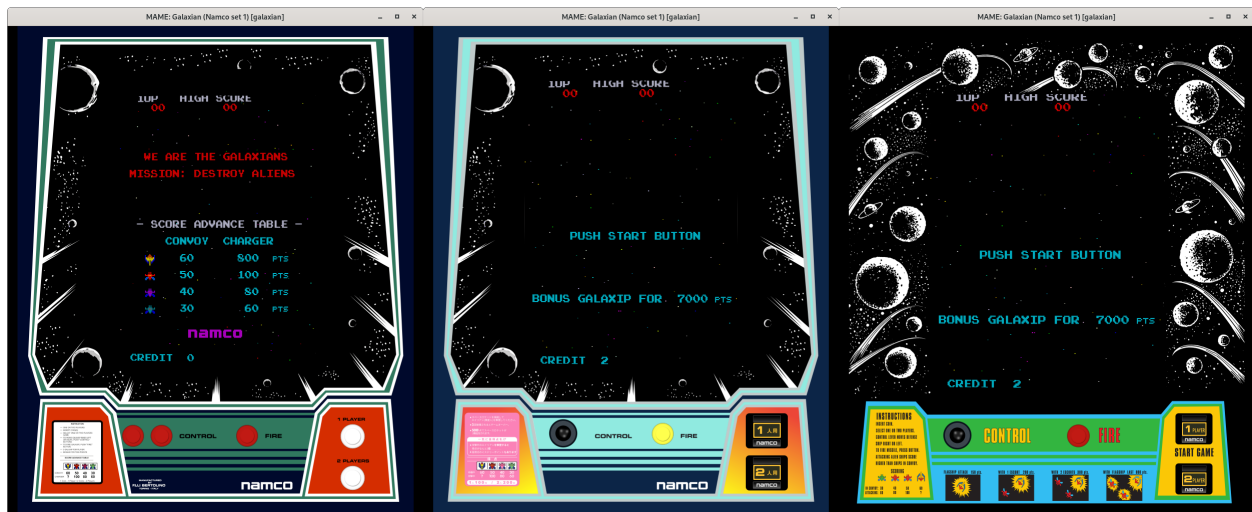
Um único arquivo para diferentes visualizações

Com os layouts também é possível montar diferentes visualizações não ficando limitado a apenas uma, ou seja, é possível dar a opção ao jogador para escolher aquilo que ele possa querer ver na tela como por exemplo, ter diferentes versões de visualização para o sistema **Galaxian**. No arquivo da versão completa (disponível em [Ferramentas necessárias](#)) temos o exemplo da **Galaxian** Italiana, Americana e a Japonesa com diferentes configurações para os botões para o 1º e 2º jogador, algumas utilizam botões simples e outras usam os botões iluminados da **NAMCO**.

As configurações para as diferentes visualizações devem ficar entre os elementos view. Assim como é feito com um layout simples, primeiro **todas as imagens** que serão utilizadas são definidas, depois cada visualização fica separada através dos elementos view e cada um com seus respectivos nomes:

```
<view name="Nome da visualização 1" showpointers="no">
    ...
    ...
</view>
<view name="Nome da visualização 2" showpointers="no">
    ...
    ...
</view>
<view name="Nome da visualização 3" showpointers="no">
    ...
    ...
</view>
```

Estas visualizações ficam acessíveis através do menu *Opções do Vídeo* (TAB --> *Opções do vídeo*).



Desativando objetos na tela

Assim como foi descrito em *Coleções* isso é possível organizando os objetos da tela dentro dos elementos collections, quando a visualização tiver esta opção, o jogador poderá desligar qualquer objeto na tela desde que tenha sido organizado desta maneira pelo autor da arte. Use visible yes ou no (também funciona com 1 ou 0) dentro do elemento collection caso queira que ele já inicie **ligado** ou **desligado**. Baixe este arquivo para ver como funciona na prática com o sistema *mspacman*.

```
<collection name="Botão para o inicio da partida para o 1º Jogador" visible="no">
<element ref="btn-namco">
    <bounds x="2605" y="3217" width="350" height="291" />
</element>
<element name="lamp0" ref="J1JP" inputtag="IN1" inputmask="1">
    <bounds x="2666" y="3299" width="230" height="126" />
</element>
</collection>
```




Criando as posições de um controle arcade

Para animar os movimentos de um joystick é preciso antes definir os limites de onde ele vai operar para que quando tudo estiver pronto a sua animação se comporte como a movimentação de um controle arcade tradicional. Usando as referências deste limite, nós podemos definir a posição de cada movimento do controle.

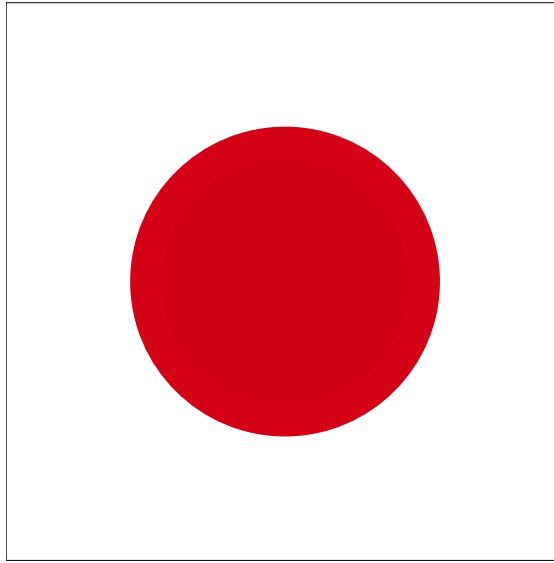
- No *Inkscape*, crie um novo arquivo, pressione **Ctrl + Shift + D** ou *File --> Document properties*, em *Page* defina *Display units* como **PX**.
- Crie um círculo preto, defina o seu tamanho em *W*: e *H*: com **7,700 px**, este será aquela peça preta que fica no fim do eixo do joystick, ele vai ajudar a dar a ilusão de movimento.
- Vá em *Layers* ou **Shift + Ctrl + L**, altere o nome deste primeiro layer para **centro base** clique no **+** para adicionar um layer chamado **centro** e um outro chamado **quadrado**, este layer **quadrado** deve ficar abaixo do layer **centro base**.
- Clique no círculo preto para selecioná-lo, pressione **Ctrl + D** para duplicá-lo, escolha uma cor vermelha qualquer, defina o seu tamanho para **9,700 px**. Clique com o botão direito em cima deste círculo vermelho e selecione a opção *Move to Layer...* e mova este círculo para o layer **centro**.

Exportando arquivos SVG

Para definir o limite eu multiplico **1.8 vezes** o valor do diâmetro círculo do controle para determinar o tamanho total desse quadrado. Este valor funciona bem ao fazer um joystick genérico, dependendo do design pode ser que um valor multiplicado por **1.5** ou por **2** funcione melhor dependendo do modelo/tipo do joystick que você está querendo imitar, ajuste e faça testes para ver qual se adapta melhor ao seu design e veja se a animação fica boa **na tela**.

- Fazendo os cálculos, $9,700 * 1,8 = 17,46$, vá em *File --> Document Properties* (**Shift + Ctrl + D**), em *Custom size* defina as unidades para **PX** e defina também o *Width/Height* para **17,46**, pressione **Enter** para aplicar e feche a janela.
- Vá em *Object --> Align and distribute*, clique no círculo **preto**, na aba *Align --> Relative To*: escolha *Page*, clique no botão logo abaixo (*Center on vertical axis*), depois (*Center on horizontal axis*) para centralizar o objeto. Clique no círculo vermelho e faça o mesmo com ele. * Pressione o **3** no teclado numérico (lado direito do teclado) para que o círculo apareça no centro da tela.
- Clique 2x no sinal de **-** no teclado numérico ou pressione **Ctrl** e gire a roda do mouse para traz para recuar um pouco.
- Crie um quadrado com **17,46 x 17,46 px**.
- Pressione **Ctrl + Shift + F** para abrir o *Fill and Stroke*, ainda com o quadrado selecionado, em *Fill* clique no **X** para remover o preenchimento, mantenha o **Shift** pressionado e com o mouse, clique na cor preta na parte de baixo da tela.
- Novamente na aba *Fill and Stroke*, defina *Width*: para **0,005**, isso cria uma borda bem fina no nosso quadrado, fina o suficiente para que ele possa ser selecionado, isso será útil mais adiante.
- Selecione o quadrado e mova ele para o layer **quadrado**.
- Na aba *Align and distribute*, clique nos mesmos botões para centralizar o quadrado na página.

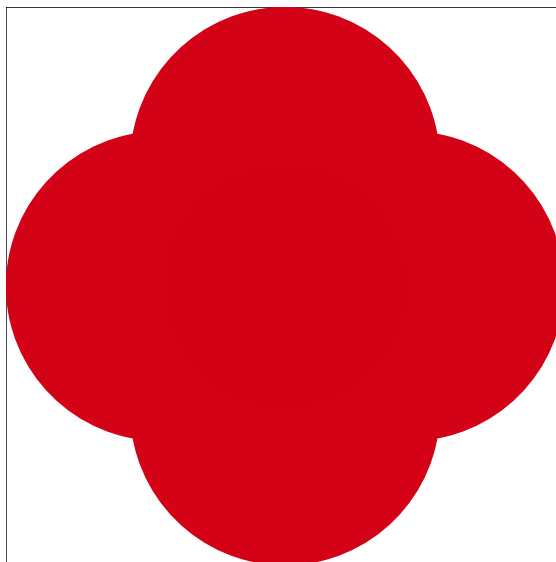
Até aqui nós temos uma imagem como esta:



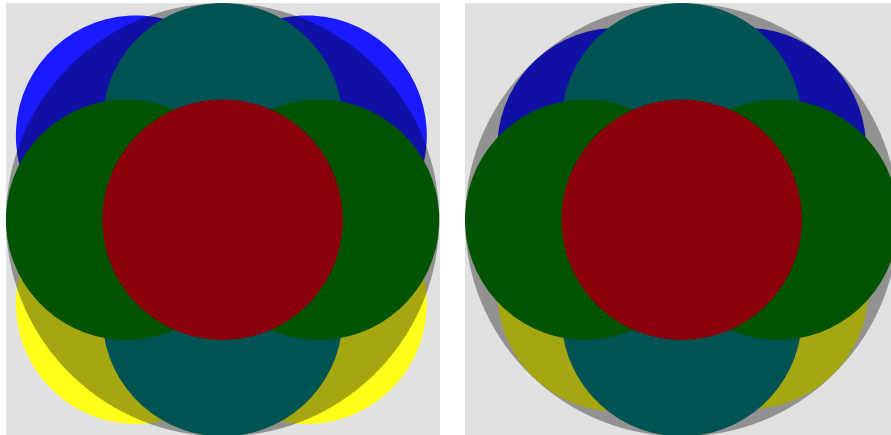
Em caso de problema, basta baixar a imagem acima (caso esteja lendo o documento no formato HTML) e abra ele no *Inkscape* para seguir os próximos passos e recrie todos layers **quadrado**, **centro base** e **centro**, depois mova círculo vermelho para o layer **centro** (clique no ícone do olho para escondê-lo) e faça o mesmo com o círculo preto e mova ele para o layer **centro base**, faça o mesmo com o quadrado, sempre deixando o layer do quadrado abaixo de todos os outros layers.

- Pressione Shift + Ctrl + A para abrir a aba *Align and distribute*, em *Relative To*: escolha *Last Selected* para alinhar com último objeto que for selecionado.
- Na aba *Layers* crie as camadas **cima**, **baixo**, **esquerda** e **direita**, clique no olho para escondê-las.
- Deixe o círculo do centro visível, clique nele e pressione Ctrl + D para duplicar.
- Com o Shift pressionado clique no quadrado na borda da página, solte o Shift e na aba de alinhamento escolha o ícone com a seta para a direita (*Align right sides*), isso alinha o nosso círculo para os limites da página do lado direito.
- Clique em qualquer ponto vazio da tela para remover a seleção, clique no círculo vermelho e com o botão direito do mouse, escolha a opção para movê-lo para a camada **direita**, o círculo do lado direito deve desaparecer logo em seguida.
- Clique no círculo vermelho ao centro, pressione Ctrl + D para duplicar, mantenha Shift pressionado, clique no quadrado, solte o Shift, na aba de alinhamento clique no ícone a seta para a esquerda (*Align left sides*) para alinhar o círculo para o lado esquerdo, repita o passo anterior e faça o mesmo para as posições **cima** e **baixo**.

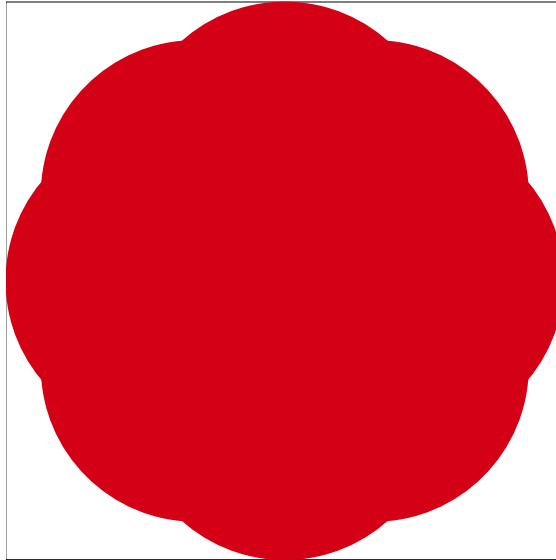
Temos agora as quatro primeiras posições:



Na questão das diagonais do joystick, seguindo a mesma lógica, elas seriam alinhadas bem nos cantos do quadrado assim com foi feito com as outras posições, porém a animação fica estranha, parece que o eixo oculto do controle permite que o círculo vá muito além do que deveria. Na imagem abaixo temos duas opções, na esquerda as diagonais foram alinhadas no limite do quadrado e depois recuados cerca de 5 pontos (clicando no + e - do X: e Y: trazendo o círculo mais para dentro). Já na imagem da direita, foi criado um círculo nas mesmas dimensões do quadrado (17,46 x 17,46 px) e as diagonais foram alinhadas no limite do círculo:



A decisão de usar um ou outro vai depender de como ele se comporta **no seu design e na tela**, o alinhamento com base num círculo pode parece "*curto*" em determinados designs e assim por diante, por isso que **não há uma regra**, cabe ao artista/designer testar e avaliar o que fica melhor. No nosso exemplo usamos o alinhamento com base no círculo.



Caso use a imagem acima, é preciso criar todos os layers novamente e mover cada posição do círculo para cada um dos seus respectivos layers, isso é importante para isolar o elemento que será exportado depois.

Depois que todas as posições já estiverem nas suas respectivas camadas (layers), começamos o processo de exportação:

- Clique em qualquer um dos layers com o botão direito do mouse e escolha a opção *Hide all layers* para ocultar tudo.
- Ative (torne visível) apenas os layers, **centro base** e **centro**.
- Vá em *File --> Save a Copy* ou Shift + Ctrl + Alt + S, escolha o caminho **MAME\Artwork\sfa3**, ou a mesma pasta onde está o `default.lay` que criamos, no nome insira **centro.svg** e na parte debaixo desta janela troque a opção *Inkscape SVG* por *Optimized SVG*, clique em *Save* e clique em *OK* na próxima janela que aparecer.
- Esconda o layer **centro** e deixe apenas o layer **direita** visível e repita o procedimento anterior, faça o mesmo para cada uma das posições.

As imagens das diagonais têm os seguintes nomes:

- **Diagonal superior direita** cimad
- **Diagonal superior esquerda** cimae
- **Diagonal inferior direita** baixod
- **Diagonal inferior esquerda** baixoe

Exportando arquivos PNG

Existe um pequeno truque que precisa ser feito para exportar imagens como estas que precisam estar dentro de algo invisível, pois diferente da exportação em SVG, não basta simplesmente deixar visível o que se deseja exportar, também é preciso exportar o quadrado junto e aqui está o "*pulo do gato*", definimos o *Stroke* (como descrito no capítulo anterior) com 0,005 ou até mesmo 0,001 para que este quadrado possa ser selecionável no *Inkscape* e que quando exportado, a borda não apareça quando for carregado pelo MAME.

No final, a imagem exportada ficará dentro de uma área invisível onde ocorrerá a animação dos movimentos. É preciso fazer assim pois caso contrário, em vez de uma animação de movimentos nós teremos o controle sendo jogado de um lado para o outro.

Usando o mesmo arquivo que você usou acima:

- Pressione Shift + Ctrl + E para abrir a aba *Export PNG image*.
- Deixe apenas o layout **centro** e **quadrado** visível.
- Clique no círculo, mantenha o Shift pressionado e clique no quadrado fazendo com que o círculo e o quadrado fiquem selecionados.
- Em *Export PNG image*, na aba *Export area* o *Selection* vai estar selecionado.
- Em *Image size* defina ambos para 174.
- Em *Filename* defina o caminho completo e o nome do arquivo (centro.png), no caso, exportaremos essa imagem para dentro da pasta **artwork\sfa3**.
- Faça o mesmo com as outras posições e respeitando o mesmo nome usado anteriormente, porém, com a extensão **.png**, centro.png, cima.png, baixo.png, etc.

Trabalhando com grupos

Trabalhar com grupos é como trabalhar com camadas em programas como o *Adobe Illustrator* ou o *Inkscape*, você desenha, posiciona os seus objetos onde eles precisam ficar e no final você **agrupa** tudo para que o seu design composto de diferentes pedaços fiquem fixos e você possa movimentá-los livremente.

O sistema de layout do MAME trabalha da mesma maneira e possui também a mesma vantagem, primeiro você define o tamanho da sua área de trabalho, define o tamanho e a posição de todos os seus elementos dentro deste grupo e faz o que quiser com ele depois.

Sem organizar o seu layout dentro de um grupo, você precisaria posicionar todos os seus elementos num ponto da tela e mais tarde caso queira movê-los para um outro ponto, você teria que literalmente ajustar cada elemento na nova posição da tela.

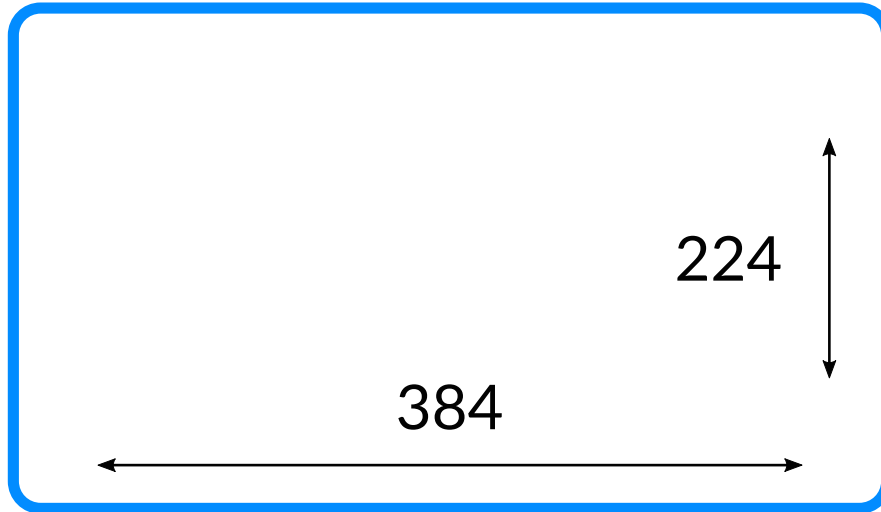
Exportaremos os gráficos tanto em **.PNG** (gráficos matriciais/pixels) quanto em **.SVG** (gráficos vetoriais), o suporte o MAME ao formato **.SVG** é simples, ele consegue carregar arquivos simples porém ele não aceita designs complexos e cheio de efeitos. Por isso dependendo da complexidade do design é preferível exportar tais gráficos no formato **.PNG**.

Os gráficos vetoriais por serem coordenadas em texto e cálculos matemáticos para criar as formas geométricas, tamanho, cor, etc; tais gráficos não se deterioram quando eles são expandidos, eles mantêm as suas bordas lisas e o seu arquivo final é extremamente leve, já os gráficos matriciais quando a aproximação ultrapassa o seu tamanho original ele se deteriora, os pixels que compõem essa imagem começa a ficar mais evidente, nas bordas começa a aparecer um serrilhamento, etc. Para compensar tal limitação é preciso exportar um gráfico cada vez maior e com isso tais imagens ficam cada vez mais pesadas. Quando for possível, use SVG, caso o seu design seja mais complexo, use PNG.

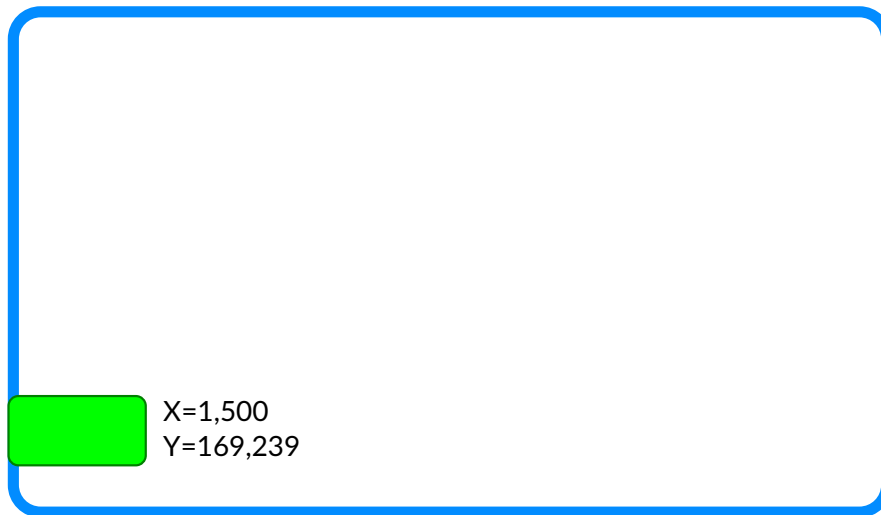
Colocaremos um pequeno joystick animado sobreposto num canto da tela, porém antes precisamos saber de duas coisas, o tamanho da tela e o tamanho do objeto que será colocado nela. Para saber o tamanho da tela, inicie um sistema qualquer como a **sfa3** (mame sfa3), pressione Tab --> *Informação do sistema*, na parte de **Vídeo** vai estar listado 384 x 224 (valor em **pixels**).

- Abra o *Inkscape*, vá em *File --> Document properties...* ou Shift + Ctrl + D, em *Page --> Units* alterne de *mm* para *px* (pixels), em *Width* defina 384,0 e em *Height* defina 224,0, feche a janela.

Abaixo temos uma imagem para ilustrar o tamanho, a área em azul é a área de segurança de quem usa shaders para deixar a tela curvada e com scanlines.



Com a nossa área de trabalho definida, precisamos definir o tamanho e a posição do nosso joystick na tela, como é um controle animado, ele não precisa ser muito grande pois a região já é pequena. No nosso exemplo, este controle só vai nos servir como uma referência para o que está acontecendo na tela, assim, ele não precisa de muitos detalhes. Na imagem abaixo tem uma ilustração da posição que foi escolhida e do tamanho.



Para o joystick da nossa tela, foi criado um objeto com 60,017 **px** de largura por 30,760 **px** de altura com uma cor qualquer e com a mesma cor na porém porém mais escura.



Dentro dessa área posicionaremos o joystick, os botões, a arte de fundo (se for o caso), etc. Depois obtemos as coordenadas **X** (horizontal), **Y** (vertical), **W** (largura) e **H** (altura), como o tamanho da tela e dos elementos são pequenos, todos os elementos e as suas bordas tem um tamanho um pouco exagerado e com bordas bem grossas, porém pois quando tudo isso for projetado na tela o resultado final ficará dentro do desejado.



Com todos os elementos posicionados nós temos:

- Joystick, **X** = 2,397, **Y** = 6,378, **W** = 18, **H** = 18
- Botão 1, **X** = 21,201, **Y** = 4,337, **W** = 9,6, **H** = 9,6
- Botão 2, **X** = 33,641, **Y** = 4,292, **W** = 9,6, **H** = 9,6
- Botão 3, **X** = 46,126, **Y** = 4,292, **W** = 9,6, **H** = 9,6
- Botão 4, **X** = 21,201, **Y** = 16,820, **W** = 9,6, **H** = 9,6
- Botão 5, **X** = 33,641, **Y** = 16,820, **W** = 9,6, **H** = 9,6
- Botão 6, **X** = 46,126, **Y** = 16,820, **W** = 9,6, **H** = 9,6

Usando os botões gerados pelo MAME e os arquivos SVG

Na pasta do MAME vá em **Artwork**, crie uma pasta chamada **sfa3**, dentro dela crie um novo arquivo texto chamado `default.lay`, tenha certeza de estar criando um arquivo `default.lay` e não `default.lay.txt`, no Windows é preciso alterar as configurações da pasta para [ver a extensão dos arquivos](#).

Abra o arquivo num editor de texto, iniciamos o nosso layout pelo cabeçalho e colocando todas as informações que achamos relevantes:

```
<?xml version="1.0"?>
<mamelayout version="2">
<!--
    Joystick for CPS2
    Created by: Wellington Terumi Uemura
    License: CC by 4.0
    https://mamedoc.readthedocs.io/
    Date: October 02, 2021
    Download: https://www.mediafire.com/file/lk1veez63081xy2/sfa3-v1.zip
-->
```

Aqui nós definimos os tipos (círculos) e as cores de todos os objetos que usaremos neste design, vamos criar os círculos coloridos (azul, amarelo, vermelho) e um círculo branco ao centro para indicar um botão na sua condição normal, quando o jogador pressionar um destes botões definidos em hit azul, hit amarelo e hit vermelho a cor vai se alterar para um tom mais escuro das cores do botão:

```
<!-- elementos gerados -->
<!-- Aqui definimos a cor azul e o mesmo é feito com as outras cores -->
<element name="azul">
    <disk><color red="0" green="0.549" blue="0.831" /></disk>
</element>
<!-- definimos um tom mais escuro da cor do botão, o mesmo é feito com as outras cores -->
<element name="hit azul" defstate="1">
    <disk state="1"><color red="0" green="0.369" blue="0.6" /></disk>
</element>
<element name="amarelo">
    <disk><color red="1" green="0.933" blue="0" /></disk>
</element>
<element name="hit amarelo" defstate="1">
    <disk state="1"><color red="0.639" green="0.596" blue="0" /></disk>
</element>
<element name="vermelho">
    <disk><color red="1" green="0.007" blue="0.105" /></disk>
</element>
<element name="hit vermelho" defstate="1">
    <disk state="1"><color red="0.6" green="0" blue="0.058" /></disk>
</element>

<!-- Este é o elemento branco que fica em cima dos círculos coloridos -->
<element name="branco">
    <disk><color red="0.925" green="0.925" blue="0.925" /></disk>
</element>

<!-- Definimos um elemento vazio que será usado para evitar os cliques do mouse -->
<element name="tampa" defstate="0">
```

(continua na próxima página)

(continuação da página anterior)

```

        <text string=" " />
</element>

<!-- Agrupamos as imagens que formam o movimento do controle num único elemento -->
<element name="comandos"          defstate="0xf">
    <image file="cima.svg"          state="0x7" />
    <image file="cimad.svg"          state="0x6" />
    <image file="direita.svg"         state="0xe" />
    <image file="baixod.svg"          state="0xa" />
    <image file="baixo.svg"           state="0xb" />
    <image file="baixoe.svg"          state="0x9" />
    <image file="esquerda.svg"        state="0xd" />
    <image file="cima.e.svg"          state="0x5" />
    <image file="centro.svg"          state="0xf" />
</element>

<!-- Carregamos a imagem da base do nosso controle -->
<element name="base">
    <image file="base.svg" />
</element>

```

Organizamos tudo isso dentro de um grupo e usamos o count para duplicar os nossos botões no eixo vertical. Aqui nós também conectamos os respectivos botões ao driver, no caso, a lógica vai funcionar com qualquer sistema dentro do driver CPS2. Alguns valores aparecem diferente daqueles que nós obtemos, é um ajuste fino necessário para alinhar os objetos na tela:

```

<!-- Nome do grupo, este é o joystick do jogador 1 -->
<group name="Joystick J1">

<!-- Definimos o tamanho do grupo -->
<bounds x="0" y="0" width="60.017" height="30.760" />

<!-- Usamos a nossa base do joystick e definimos o seu tamanho -->
<element ref="base">
    <bounds x="0" y="0" width="60.017" height="30.760" />
</element>

<!-- São 2 botões no eixo Y (vertical) iniciando na posição 4.292.-->
<!-- Incremente 12.528 para definir a posição do segundo botão. -->
<!-- Para encontrar o valor 12.528, use o Inkscape para posicionar o botão onde deseja -->
<!-- pegue o valor de Y do segundo botão (16,82) e subtraia com o valor -->
<!-- de Y do primeiro botão (4,292), ou seja, 16,82 - 4,292 = 12,528. -->
<!-- No eixo X inicie em 21.202 e use o valor calculado para definir Y. -->
<!-- Crie o nosso elemento com 9.7px x 9.7px -->
<!-- O mesmo é feito com os outros botões -->
<repeat count="2">
    <param name="y" start="4.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="azul"><bounds x="21.202" y="~y" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">

```

(continua na próxima página)

(continuação da página anterior)

```

    <param name="y" start="5.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="branco"><bounds x="22.202" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
    <param name="y" start="4.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="amarelo"><bounds x="33.641" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
    <param name="y" start="5.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="branco"><bounds x="34.641" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
    <param name="y" start="4.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="vermelho"><bounds x="46.129" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
    <param name="y" start="5.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="branco"><bounds x="47.129" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>

<!-- Aqui a tampa cobre toda a área do joystick para evitar o click do mouse -->
<element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
    <bounds x="0" y="0" width="60.017" height="30.760" />
</element>

<!-- Invocamos os nossos comandos, definimos a sua posição e tamanho na base do joystick -->
<element ref="comandos" inputtag="IN0" inputmask="0xf" inputraw="yes">
    <bounds x="2.397" y="6.378" width="18" height="18" />
</element>

<!-- Aqui conectamos toda a lógica dos botões -->
<element ref="hit azul" inputtag="IN0" inputmask="0x10">
    <bounds x="22.202" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit azul" inputtag="IN1" inputmask="0x1">
    <bounds x="22.202" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="hit amarelo" inputtag="IN0" inputmask="0x20">
    <bounds x="34.641" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit amarelo" inputtag="IN1" inputmask="0x2">
    <bounds x="34.641" y="17.820" width="7.7" height="7.7" />

```

(continua na próxima página)

(continuação da página anterior)

```

</element>
<element ref="hit vermelho" inputtag="IN0" inputmask="0x40">
    <bounds x="47.129" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN1" inputmask="0x40">
    <bounds x="47.129" y="17.820" width="7.7" height="7.7" />
</element>
</group>

```

Como já explicado em *A Posição e a proporção de tela*, as telas CRT da época tinham uma proporção de 4:3, assim sendo, precisamos ajudar a resolução de tela. Use a planilha disponibilizada em *Ferramentas necessárias*, no campo *Digite a altura* insira 224, note que o cálculo retorna um DAR com 298,666666666667, isso não é bom pois causa *aliasing*, então, precisamos de um valor inteiro.

Substitua 224 por 225, note que os cálculos retornam valores inteiros, é este valor que usaremos para definir o tamanho da nossa tela 300 x 225, procure sempre utilizar valores pares no eixo horizontal. Logo abaixo nós definimos a posição do nosso joystick e o seu respectivo tamanho.

```

<!-- Definimos o nome que vai aparecer na seleção -->
<view name="Controle" showpointers="no">

<!-- Não precisamos da posição, só do tamanho da tela -->
<screen index="0">
    <bounds x="0" y="0" width="300" height="225" />
</screen>

<!-- Aqui definimos um nome que vai aparecer nas opções para -->
<!-- tirar o joystick da tela, será possível ligar e desligar esta opção. -->
<!-- Já fica predefinido que quando a emulação começar, o joystick já apareça na tela. -->
<collection name="Joystick do jogador 1" visible="yes">

<!-- Invocamos o nome do grupo, definimos a posição e o seu tamanho -->
<group ref="Joystick J1">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
</group>
</collection>

<!-- Fim do layout -->
</view>
</mamelayout>

```

Salve o arquivo, execute o comando no terminal/prompt de comando `mame sfa3` e veja que todos os botões e o controle reagem ao que você fizer no seu controle. Pressione Tab --> *Opções do vídeo* --> *Tela #0*, veja que há a opção *Joystick do jogador 1* que pode ser ligada ou desligada.

Para adicionar o controle para o jogador 2, basta agora adicionar a lista abaixo **depois** do grupo **Joystick J1** e adaptar a lógica para o controle 2.

```

<!-- Nome do grupo, este é o joystick do jogador 2 -->
<group name="Joystick J2">
<bounds x="0" y="0" width="60.017" height="30.760" />
<element ref="base">
    <bounds x="0" y="0" width="60.017" height="30.760" />

```

(continua na próxima página)

(continuação da página anterior)

```

</element>
<repeat count="2">
  <param name="y" start="4.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="azul"><bounds x="21.202" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="22.202" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="4.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="amarelo"><bounds x="33.641" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="34.641" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="4.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="vermelho"><bounds x="46.129" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="47.129" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
  <bounds x="0" y="0" width="60.017" height="30.760" />
</element>
<element ref="comandos" inputtag="IN0" inputmask="0xf00" inputraw="yes">
  <bounds x="2.397" y="6.378" width="18" height="18" />
</element>
<!-- Aqui conectamos toda a lógica dos botões para o J2-->
<element ref="hit azul" inputtag="IN0" inputmask="0x1000">
  <bounds x="22.202" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit azul" inputtag="IN1" inputmask="0x10">
  <bounds x="22.202" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="hit amarelo" inputtag="IN0" inputmask="0x2000">
  <bounds x="34.641" y="5.292" width="7.7" height="7.7" />

```

(continua na próxima página)

(continuação da página anterior)

```

</element>
<element ref="hit amarelo" inputtag="IN1" inputmask="0x20">
  <bounds x="34.641" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN0" inputmask="0x4000">
  <bounds x="47.129" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN2" inputmask="0x4000">
  <bounds x="47.129" y="17.820" width="7.7" height="7.7" />
</element>
</group>

```

Para que funcione, nós adicionamos a opção "Controle" e adicionamos a opção para o jogador 2 logo depois das configurações do jogador 1.

```

<collection name="Joystick do jogador 1" visible="yes">
  <group ref="Joystick J1" blend="add">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
  </group>
</collection>

<collection name="Joystick do jogador 2" visible="no">
  <group ref="Joystick J2" blend="add">
    <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
  </group>
</collection>

```

Salve e rode o **sfa3** novamente, parece que nada mudou, porém ao entrar nas opções do vídeo novamente, verá que está disponível a opção para o *Joystick do jogador 2* e ele está *desligado*. Quem define isso é a opção `visible="no"`, porém para deixar ele sempre ativo sem ter que mexer no arquivo layout, basta deixar o segundo joystick visível e encerrar a emulação, isso salva as definições este sistema e na próxima vez o joystick vai aparecer na tela.

Simplificando o layout

Uma outra maneira de se lidar com grupos e evitar a duplicidade de configuração como foi feito para o controle do jogador 1 e 2, seria organizar a parte visual primeiro num grupo que poderia ser usado por ambos e deixar a parte lógica dos botões de fora, invocando a lógica individual de cada controle separadamente.

A parte visual do nosso controle ficaria assim:

```

<group name="Joystick">
  <bounds x="0" y="0" width="60.017" height="30.760" />
  <element ref="base">
    <bounds x="0" y="0" width="60.017" height="30.760" />
  </element>
  <repeat count="2">
    <param name="y" start="4.292" increment="12.528" />
    <param name="i" start="0" increment="1" />
    <element ref="azul"><bounds x="21.202" y="~y~" width="9.7" height="9.7" />
  </element>
</repeat>

```

(continua na próxima página)

(continuação da página anterior)

```

<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="22.202" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="4.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="amarelo"><bounds x="33.641" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="34.641" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="4.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="vermelho"><bounds x="46.129" y="~y~" width="9.7" height="9.7" />
</element>
</repeat>
<repeat count="2">
  <param name="y" start="5.292" increment="12.528" />
  <param name="i" start="0" increment="1" />
  <element ref="branco"><bounds x="47.129" y="~y~" width="7.7" height="7.7" />
</element>
</repeat>
<element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
  <bounds x="0" y="0" width="60.017" height="30.760" />
</element>
</group>

```

Aqui a parte lógica para o jogador 1.

```

<group name="Logic J1">
  <bounds x="0" y="0" width="60.017" height="30.760" />
  <element ref="hit azul" inputtag="IN0" inputmask="0x10">
    <bounds x="22.202" y="5.292" width="7.7" height="7.7" />
  </element>
  <element ref="hit azul" inputtag="IN1" inputmask="0x1">
    <bounds x="22.202" y="17.820" width="7.7" height="7.7" />
  </element>
  <element ref="hit amarelo" inputtag="IN0" inputmask="0x20">
    <bounds x="34.641" y="5.292" width="7.7" height="7.7" />
  </element>
  <element ref="hit amarelo" inputtag="IN1" inputmask="0x2">
    <bounds x="34.641" y="17.820" width="7.7" height="7.7" />
  </element>
  <element ref="hit vermelho" inputtag="IN0" inputmask="0x40">

```

(continua na próxima página)

(continuação da página anterior)

```

        <bounds x="47.129" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN1" inputmask="0x4">
    <bounds x="47.129" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="comandos" inputtag="IN0" inputmask="0xf" inputdraw="yes">
    <bounds x="2.397" y="6.378" width="18" height="18" />
</element>
</group>

```

Aqui a parte lógica para o jogador 2.

```

<group name="Logic J2">
    <bounds x="0" y="0" width="60.017" height="30.760" />
<element ref="hit azul" inputtag="IN0" inputmask="0x1000">
    <bounds x="22.202" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit azul" inputtag="IN1" inputmask="0x10">
    <bounds x="22.202" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="hit amarelo" inputtag="IN0" inputmask="0x2000">
    <bounds x="34.641" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit amarelo" inputtag="IN1" inputmask="0x20">
    <bounds x="34.641" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN0" inputmask="0x4000">
    <bounds x="47.129" y="5.292" width="7.7" height="7.7" />
</element>
<element ref="hit vermelho" inputtag="IN2" inputmask="0x4000">
    <bounds x="47.129" y="17.820" width="7.7" height="7.7" />
</element>
<element ref="comandos" inputtag="IN0" inputmask="0xf00" inputdraw="yes">
    <bounds x="2.397" y="6.378" width="18" height="18" />
</element>
</group>

```

Aqui alteramos a parte final do layout, invocamos primeiro a parte visual e depois a parte lógica.

```

<view name="Controle" showpointers="no">
<screen index="0">
    <bounds x="0" y="0" width="300" height="225" />
</screen>

<collection name="Joystick do jogador 1" visible="yes">
<group ref="Joystick" blend="add">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
</group>
<group ref="Logic J1">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
</group>
</collection>

```

(continua na próxima página)

(continuação da página anterior)

```

<collection name="Joystick do jogador 2" visible="no">
<group ref="Joystick" blend="add">
  <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
</group>
<group ref="Logic J2">
  <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
</group>
</collection>

```

Ao rodar o sistema novamente verá que toda a parte funcional e das opções continuam os mesmos, porém, o nosso arquivo layout está mais organizado.

Baixe [este arquivo completo aqui](#).

Usando imagens PNG para o mesmo tipo de layout

Ao trabalhar com imagens PNG é preciso exportar **CADA** elemento do nosso joystick, além de todas as posições do controle é preciso exportar todo o resto, além disso, é preciso levar em consideração o seu tamanho final pois como foi explicado antes, a imagem matricial se deteriora caso ela seja expandida para dimensões maiores do que ela foi projetada.

Neste exemplo usaremos design abaixo como referência:



- Abra a imagem no *Inkscape*, pressione Ctrl + A para selecionar todos os elementos, em seguida faça Shift + Ctrl + G para separar todos os elementos do grupo.
- Faça Shift + Ctrl + E ou File --> *Export PNG Image* para abrir a aba de exportação.
- Clique nos números e pressione Del para excluí-los.
- Clique no círculo azul com o centro escuro, na aba *Export PNG Image* em *Image size* defina ambos *Width/Height* como 100 (poderia ser maior mas serve para o nosso exemplo).
- Em *Filename* exporte para a pasta **artwork/sfa3** com o nome azul1.png.
- Clique no outro botão azul com o centro claro e faça o mesmo, porém exporte com o nome azul0.png.
- Faça o mesmo com os outros botões, no final você deverá ter os arquivos azul0.png, azul1.png, amarelo0.png, amarelo1.png, vermelho0.png e vermelho1.png.

- Clique na base do nosso joystick, exporte ele com o tamanho 575 x 294 e defina o seu nome como base.png.

Com todas as imagens em mãos podemos começar a montar o nosso layout:

```
<?xml version="1.0"?>
<mamelayout version="2">
<!--
    Joystick for CPS2
    Created by: Wellington Terumi Uemura
    License: CC by 4.0
    https://mamedoc.readthedocs.io/
    Date: October 02, 2021
    Download: https://www.mediafire.com/file/3dd1dtn114ochi2/sfa3-v3.zip
-->
```

Em seguida definimos as condicionais dos botões e a nossa tampa:

```
<element name="painel">
    <image file="base.png" />
</element>
<element name="azul" defstate="0">
    <image file="azul0.png" state="0" />
    <image file="azul1.png" state="1" />
</element>
<element name="amarelo" defstate="0">
    <image file="amarelo0.png" state="0" />
    <image file="amarelo1.png" state="1" />
</element>
<element name="vermelho" defstate="0">
    <image file="vermelho0.png" state="0" />
    <image file="vermelho1.png" state="1" />
</element>
<element name="tampa" defstate="0">
    <text string=" " />
</element>
```

Aqui nós definimos todas as posições do joystick:

```
<element name="comandos" defstate="0xf">
    <image file="cima.png" state="0x7" />
    <image file="cimad.png" state="0x6" />
    <image file="direita.png" state="0xe" />
    <image file="baixod.png" state="0xa" />
    <image file="baixo.png" state="0xb" />
    <image file="baixoe.png" state="0x9" />
    <image file="esquerda.png" state="0xd" />
    <image file="cimaes.png" state="0x5" />
    <image file="centro.png" state="0xf" />
</element>
```

Neste primeiro momento, usamos a área que nós definimos para o controle 60,017 x 30,760 px e dentro dessa área posicionamos todas as partes do controle.

```

<group name="controle J1">
  <bounds x="0" y="0" width="60.017" height="30.760" />
  <element ref="painel">
    <bounds x="0" y="0" width="60.017" height="30.760" />
  </element>
  <!-- aqui cobrimos toda a área do joystick para não interagir com o mouse -->
  <element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
    <bounds x="0" y="0" width="60.017" height="30.760" />
  </element>
  <!-- Botão 1 -->
  <element ref="azul" inputtag="IN0" inputmask="16">
    <bounds x="21.201" y="4.292" width="9.6" height="9.6" />
  </element>
  <!-- Botão 4 -->
  <element ref="azul" inputtag="IN1" inputmask="1">
    <bounds x="21.201" y="16.820" width="9.6" height="9.6" />
  </element>
  <!-- Botão 2 -->
  <element ref="amarelo" inputtag="IN0" inputmask="32">
    <bounds x="33.641" y="4.292" width="9.6" height="9.6" />
  </element>
  <!-- Botão 5 -->
  <element ref="amarelo" inputtag="IN1" inputmask="2">
    <bounds x="33.641" y="16.820" width="9.6" height="9.6" />
  </element>
  <!-- Botão 3 -->
  <element ref="vermelho" inputtag="IN0" inputmask="64">
    <bounds x="46.126" y="4.292" width="9.6" height="9.6" />
  </element>
  <!-- Botão 6 -->
  <element ref="vermelho" inputtag="IN1" inputmask="4">
    <bounds x="46.126" y="16.820" width="9.6" height="9.6" />
  </element>
  <element ref="comandos" inputtag="IN0" inputmask="0xf" inputraw="yes">
    <bounds x="2.397" y="6.378" width="18" height="18" />
  </element>
</group>
<!-- Aqui os parâmetros para o joystick para o segundo jogador -->
<group name="controle J2">
  <bounds x="0" y="0" width="60.017" height="30.760" />
  <element ref="painel">
    <bounds x="0" y="0" width="60.017" height="30.760" />
  </element>
  <element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
    <bounds x="0" y="0" width="60.017" height="30.760" />
  </element>
  <!-- Botão 1 -->
  <element ref="azul" inputtag="IN0" inputmask="0x1000">
    <bounds x="21.201" y="4.337" width="9.6" height="9.6" />
  </element>
  <!-- Botão 4 -->
  <element ref="azul" inputtag="IN1" inputmask="0x10">

```

(continua na próxima página)

(continuação da página anterior)

```

        <bounds x="21.156" y="16.820" width="9.6" height="9.6" />
    </element>
    <!-- Botão 2 -->
    <element ref="amarelo" inputtag="IN0" inputmask="0x2000">
        <bounds x="33.641" y="4.292" width="9.6" height="9.6" />
    </element>
    <!-- Botão 5 -->
    <element ref="amarelo" inputtag="IN1" inputmask="0x20">
        <bounds x="33.641" y="16.820" width="9.6" height="9.6" />
    </element>
    <!-- Botão 3 -->
    <element ref="vermelho" inputtag="IN0" inputmask="0x4000">
        <bounds x="46.126" y="4.292" width="9.6" height="9.6" />
    </element>
    <!-- Botão 6 -->
    <element ref="vermelho" inputtag="IN2" inputmask="0x4000">
        <bounds x="46.126" y="16.820" width="9.6" height="9.6" />
    </element>
    <element ref="comandos" inputtag="IN0" inputmask="0xf00" inputraw="yes">
        <bounds x="2.397" y="6.378" width="18" height="18" />
    </element>
</group>

```

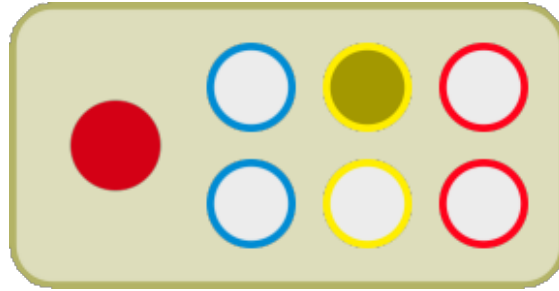
O mesmo caso anterior, definimos o nome e uma resolução 4:3, no caso, 300 x 225, posicionamos os controles na tela e encerramos o layout.

```

<view name="Controles" showpointers="no">
    <screen index="0">
        <bounds x="0" y="0" width="300" height="225" />
    </screen>
    <collection name="Joystick do jogador 1" visible="yes">
    <group ref="controle J1">
        <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
    </group>
    </collection>
    <collection name="Joystick do jogador 2" visible="no">
    <group ref="controle J2">
        <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
    </group>
    </collection>
</view>
</mamelayout>

```

Ao testar o layout com as imagens o controle funciona normalmente, contudo, repare que nos botões **amarelos** e nos botões **vermelhos**, parece que há uma borda preta neles.



Parece que outra pessoa já identificou este problema e ele ainda *não foi corrigido*, a solução parece que é criar um quadrado do mesmo tamanho do círculo (ou outra cor qualquer menos preto), alinhe com a página, preencha com o mesmo amarelo do círculo ffee00ff, baixe o **alpha** para zero ficando ffee0000 e exporte a imagem **.png** novamente.

Simplificando o layout novamente

Usando o conhecimento aprendido em *Usando os botões gerados pelo MAME e os arquivos SVG*, podemos fazer o mesmo com imagens **PNG**, dando a mesma funcionalidade porém deixando o layout mais enxuto, separamos o que é visual do lógico e depois montamos.

```
<?xml version="1.0"?>
<mamelayout version="2">
<!--
    Joystick for CPS2
    Created by: Wellington Terumi Uemura
    License: CC by 4.0
    https://mamedoc.readthedocs.io/
    Date: October 02, 2021
    Download: https://www.mediafire.com/file/8w6fk6ogo0z17q2/sfa3-v4.zip
-->

<element name="painel">
    <image file="base.png" />
</element>
<element name="azul" defstate="0">
    <image file="azul0.png" state="0" />
    <image file="azul1.png" state="1" />
</element>
<element name="amarelo" defstate="0">
    <image file="amarelo0.png" state="0" />
    <image file="amarelo1.png" state="1" />
</element>
<element name="vermelho" defstate="0">
    <image file="vermelho0.png" state="0" />
    <image file="vermelho1.png" state="1" />
</element>
<element name="tampa" defstate="0">
    <text string=" " />
</element>

<element name="comandos" defstate="0xf">
    <image file="cima.png" state="0x7" />
```

(continua na próxima página)

(continuação da página anterior)

```

        <image file="cimad.png"          state="0x6" />
        <image file="direita.png"       state="0xe" />
        <image file="baixod.png"        state="0xa" />
        <image file="baixo.png"         state="0xb" />
        <image file="baixoe.png"        state="0x9" />
        <image file="esquerda.png"      state="0xd" />
        <image file="cimae.png"         state="0x5" />
        <image file="centro.png"        state="0xf"/>
    </element>

    <group name="Joystick">
        <bounds x="0" y="0" width="60.017" height="30.760" />
    <element ref="painel">
        <bounds x="0" y="0" width="60.017" height="30.760" />
    </element>
    <element ref="tampa" blend="add" inputtag="IN0" inputmask="0x00" inputraw="yes">
        <bounds x="0" y="0" width="60.017" height="30.760" />
    </element>
</group>

<group name="Logic J1">
    <bounds x="0" y="0" width="60.017" height="30.760" />
    <element ref="azul" inputtag="IN0" inputmask="0x10">
        <bounds x="21.202" y="4.292" width="9.7" height="9.7" />
    </element>
    <element ref="azul" inputtag="IN1" inputmask="0x1">
        <bounds x="21.202" y="16.820" width="9.7" height="9.7" />
    </element>
    <element ref="amarelo" inputtag="IN0" inputmask="0x20">
        <bounds x="33.641" y="4.292" width="9.7" height="9.7" />
    </element>
    <element ref="amarelo" inputtag="IN1" inputmask="0x2">
        <bounds x="33.641" y="16.820" width="9.7" height="9.7" />
    </element>
    <element ref="vermelho" inputtag="IN0" inputmask="0x40">
        <bounds x="46.129" y="4.292" width="9.7" height="9.7" />
    </element>
    <element ref="vermelho" inputtag="IN1" inputmask="0x4">
        <bounds x="46.129" y="16.820" width="9.7" height="9.7" />
    </element>
    <element ref="comandos" inputtag="IN0" inputmask="0xf" inputraw="yes">
        <bounds x="2.397" y="6.378" width="18" height="18" />
    </element>
</group>

<group name="Logic J2">
    <bounds x="0" y="0" width="60.017" height="30.760" />
    <element ref="azul" inputtag="IN0" inputmask="0x1000">
        <bounds x="21.202" y="4.292" width="9.7" height="9.7" />
    </element>
    <element ref="azul" inputtag="IN1" inputmask="0x10">
        <bounds x="21.202" y="16.820" width="9.7" height="9.7" />
    </element>

```

(continua na próxima página)

(continuação da página anterior)

```

</element>
<element ref="amarelo" inputtag="IN0" inputmask="0x2000">
    <bounds x="33.641" y="4.292" width="9.7" height="9.7" />
</element>
<element ref="amarelo" inputtag="IN1" inputmask="0x20">
    <bounds x="33.641" y="16.820" width="9.7" height="9.7" />
</element>
<element ref="vermelho" inputtag="IN0" inputmask="0x4000">
    <bounds x="46.129" y="4.292" width="9.7" height="9.7" />
</element>
<element ref="vermelho" inputtag="IN2" inputmask="0x4000">
    <bounds x="46.129" y="16.820" width="9.7" height="9.7" />
</element>
<element ref="comandos" inputtag="IN0" inputmask="0xf00" inputraw="yes">
    <bounds x="2.397" y="6.378" width="18" height="18" />
</element>
</group>

<view name="Controle" showpointers="no">
<screen index="0">
    <bounds x="0" y="0" width="300" height="225" />
</screen>

<collection name="Joystick do jogador 1" visible="yes">
<group ref="Joystick" blend="add">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
</group>
<group ref="Logic J1">
    <bounds x="1.5" y="169.239" width="60.017" height="30.760" />
</group>
</collection>

<collection name="Joystick do jogador 2" visible="no">
<group ref="Joystick" blend="add">
    <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
</group>
<group ref="Logic J2">
    <bounds x="238.4" y="169.239" width="60.017" height="30.760" />
</group>
</collection>
</view>
</mamelayout>

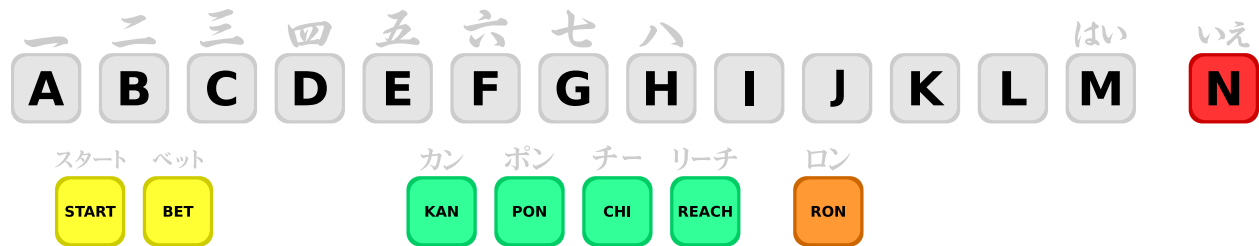
```

Dividindo a tela com um controle ou outro tipo de informação

Usar uma tela dividida com um controle pode ser útil para jogos do tipo *Mahjong* por exemplo, o uso do teclado do PC ou até mesmo um joystick comum em tais jogos pode ser confuso, deixando o controle visível e com a possibilidade de poder clicar nos botões na tela, assim você se preocupa com o jogo e não em lidar com o mapeamento do teclado.

Neste exemplo usaremos o sistema *Lovely Pop Mahjong JangJang Shimasho (Japan)*, ela usa o driver `ssv.cpp`, precisamos acessá-lo para identificar todas as suas entradas e posteriormente, mapear todos os botões.

Para os controles usaremos uma variação do controle do capítulo *Teclas predefinidas para Mahjong e Hana-fuda*:



Crie a pasta `artwork\janjans1`, dentro dela copie o arquivo `.svg` acima, crie também o arquivo `default.lay` e abra-o num editor de texto.

Iniciamos o layout com o básico:

```
<?xml version="1.0"?>
<mamelayout version="2">

<!--
    Joystick Mahjang for janjans1
    Created by: Wellington Terumi Uemura
    License: CC by 4.0
    https://mamedoc.readthedocs.io/
    Date: October 25, 2021
    Download: https://www.mediafire.com/file/4m6di0smdiqj8wf/janjans1.zip
-->
<!-- Aqui carregamos o nosso controle -->
<element name="base">
    <image file="mahjongpanel2.svg" />
</element>

<!--
    Aqui deixamos reservado para uso futuro, como iluminar as teclas
    depois de serem clicadas por exemplo, porém, distrai demais.
    Assim deixamos ele vazio.
-->
<element name="btn" defstate="1">
    <text string=" ">
        <color red="0.0" green="0.0" blue="0.0" />
        <bounds x="0" y="0" width="58.038" height="58.038" />
    </text>
</element>
```

Agora precisamos ver no driver como os botões estão definidos para poder conectá-los na imagem do nosso controle, na data de criação deste texto, essa informação começa na [linha 1028](#):


```

<!-- Definimos o tamanho da nossa área de trabalho dentro de um grupo -->
<group name="Controle_mah">
    <bounds x="0" y="0" width="1136" height="216.63" />

<!-- Informamos o tamanho original da imagem do nosso controle -->
<element ref="base">
    <bounds x="0" y="0" width="1136" height="216.63" />
</element>

<!--
    Aqui mapeamos todos os botões segundo o driver, abrindo a nossa
    imagem no Inkscape e desagrupando todos os elementos
    (Shift + Ctrl + G), ao clicar na região amarelo claro do botão
    "Start" por exemplo, nós temos as coordenadas e o seu respectivo
    tamanho no topo da tela (lembrando que os valores devem estar em
    px!).
    É assim que obtemos os valores de posição e tamanho que usamos
    abaixo.
-->

<!-- PON -->
<element ref="btn" inputtag="KEY0" inputmask="0x4">
    <bounds x="442.969" y="159.961" width="58.038" height="58.038" />
</element>
<!-- L -->
<element ref="btn" inputtag="KEY0" inputmask="0x8">
    <bounds x="882.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- H -->
<element ref="btn" inputtag="KEY0" inputmask="0x10">
    <bounds x="562.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- D -->
<element ref="btn" inputtag="KEY0" inputmask="0x20">
    <bounds x="242.979" y="47.960" width="58.038" height="58.038" />
</element>

<!-- RON -->
<element ref="btn" inputtag="KEY1" inputmask="0x2">
    <bounds x="715.014" y="159.961" width="58.038" height="58.038" />
</element>
<!-- CHI -->
<element ref="btn" inputtag="KEY1" inputmask="0x4">
    <bounds x="522.965" y="159.961" width="58.038" height="58.038" />
</element>
<!-- K -->
<element ref="btn" inputtag="KEY1" inputmask="0x8">
    <bounds x="802.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- G -->
<element ref="btn" inputtag="KEY1" inputmask="0x10">
    <bounds x="482.979" y="47.960" width="58.038" height="58.038" />
</element>

```

(continua na próxima página)

(continuação da página anterior)

```

<!-- C -->
<element ref="btn" inputtag="KEY1" inputmask="0x20">
    <bounds x="162.979" y="47.960" width="58.038" height="58.038" />
</element>

<!-- BET -->
<element ref="btn" inputtag="KEY2" inputmask="0x1">
    <bounds x="122.981" y="159.961" width="58.038" height="58.038" />
</element>
<!-- REACH -->
<element ref="btn" inputtag="KEY2" inputmask="0x2">
    <bounds x="602.965" y="159.961" width="58.038" height="58.038" />
</element>
<!-- N -->
<element ref="btn" inputtag="KEY2" inputmask="0x4">
    <bounds x="1075" y="47.961" width="58.038" height="58.038" />
</element>
<!-- J -->
<element ref="btn" inputtag="KEY2" inputmask="0x8">
    <bounds x="722.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- F -->
<element ref="btn" inputtag="KEY2" inputmask="0x10">
    <bounds x="402.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- B -->
<element ref="btn" inputtag="KEY2" inputmask="0x20">
    <bounds x="82.979" y="47.960" width="58.038" height="58.038" />
</element>

<!-- START -->
<element ref="btn" inputtag="KEY3" inputmask="0x1">
    <bounds x="42.980" y="159.961" width="58.038" height="58.038" />
</element>
<!-- KAN -->
<element ref="btn" inputtag="KEY3" inputmask="0x2">
    <bounds x="362.965" y="159.961" width="58.038" height="58.038" />
</element>
<!-- M -->
<element ref="btn" inputtag="KEY3" inputmask="0x4">
    <bounds x="962.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- I -->
<element ref="btn" inputtag="KEY3" inputmask="0x8">
    <bounds x="642.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- E -->
<element ref="btn" inputtag="KEY3" inputmask="0x10">
    <bounds x="322.979" y="47.960" width="58.038" height="58.038" />
</element>
<!-- A -->
<element ref="btn" inputtag="KEY3" inputmask="0x20">

```

(continua na próxima página)

(continuação da página anterior)

```

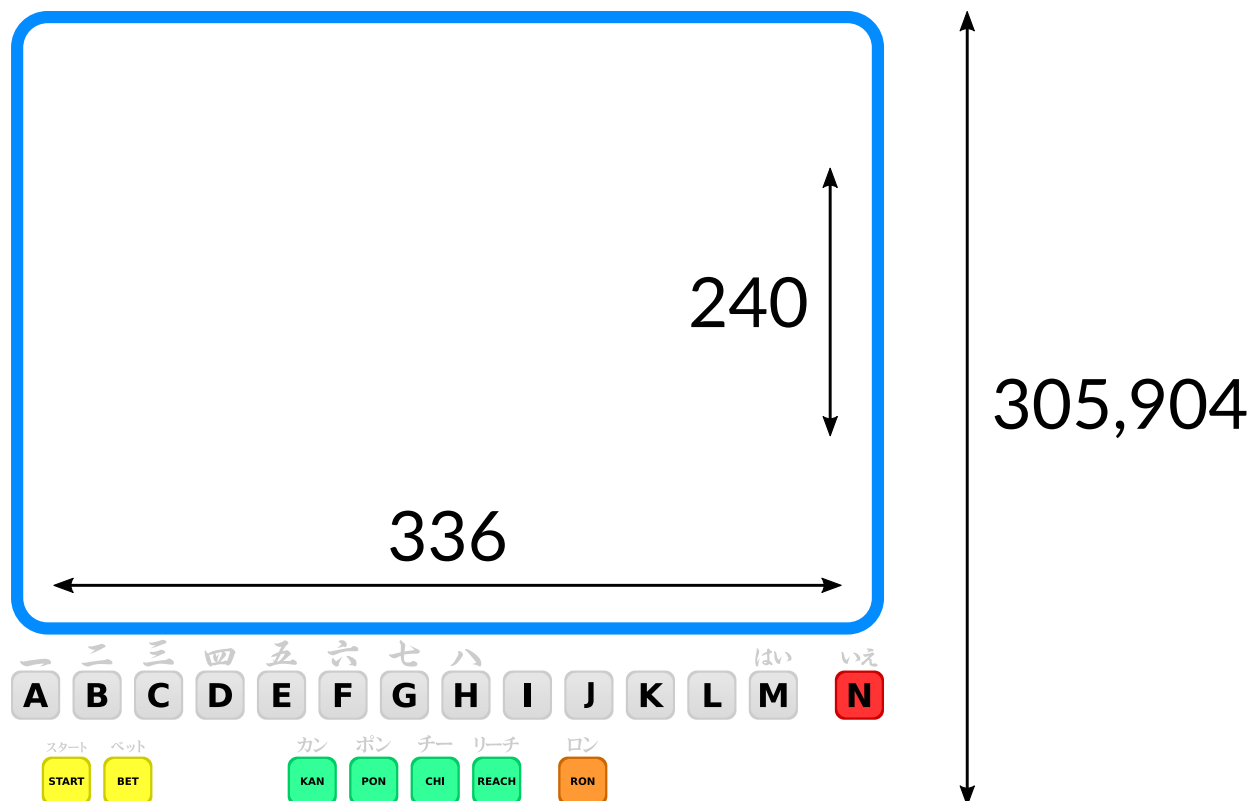
        <bounds x="2.979" y="47.960" width="58.038" height="58.038" />
    </element>
</group>

```

Resolvida a questão dos controles, agora é preciso definir como vamos disponibilizar os objetos na tela. Ao iniciar o sistema com `mame janjans1` e ao clicar em *Tab --> Informação do sistema*, o driver informa que o tamanho da tela tem 336 x 240.

- Inicie o *Inkscape*, defina o *Units* e a página em **px**, defina o tamanho da página como 336 por 240 **px**.
- Importe para esta página a imagem do controle fornecida acima, durante a importação da imagem deve aparecer uma tela, em *DPI for rendered SVG* mantenha 96 DPI.
- Clique no cadeado entre os valores *W*: e *H*: para travar a proporção.
- Em *W*: coloque 336,0 e pressione Enter para aplicar, agora temos um objeto na tela com 336 por 63,904.
- Somando a altura do objeto com a altura da tela 63,904 + 240 nós temos uma altura total de 303,904.
- Pressione Shift + Ctrl + D e na altura da página coloque 303,904 **px**.
- Desenhe um quadrado que vai simular a nossa tela com 336 por 240 alinhado no **Topo da página**.
- Posicione a imagem do controle logo abaixo e alinhe, isso lhe dará uma visão geral daquilo que pretende fazer.
- Como o controle fica muito colado com a tela, descemos o controle cerca de um toque do direcional para baixo.
- Faça Ctrl + A para selecionar tudo seguido de Ctrl + G para agrupar ambos os objetos, agora temos uma altura final com 305,904.
- Faça novamente Shift + Ctrl + D e atualize a altura da página em **px**, isso é importante para obter as devidas coordenadas para o posicionamento dos elementos na tela.

Se deu tudo certo até aqui, você deve ter algo semelhante ao exemplo abaixo:

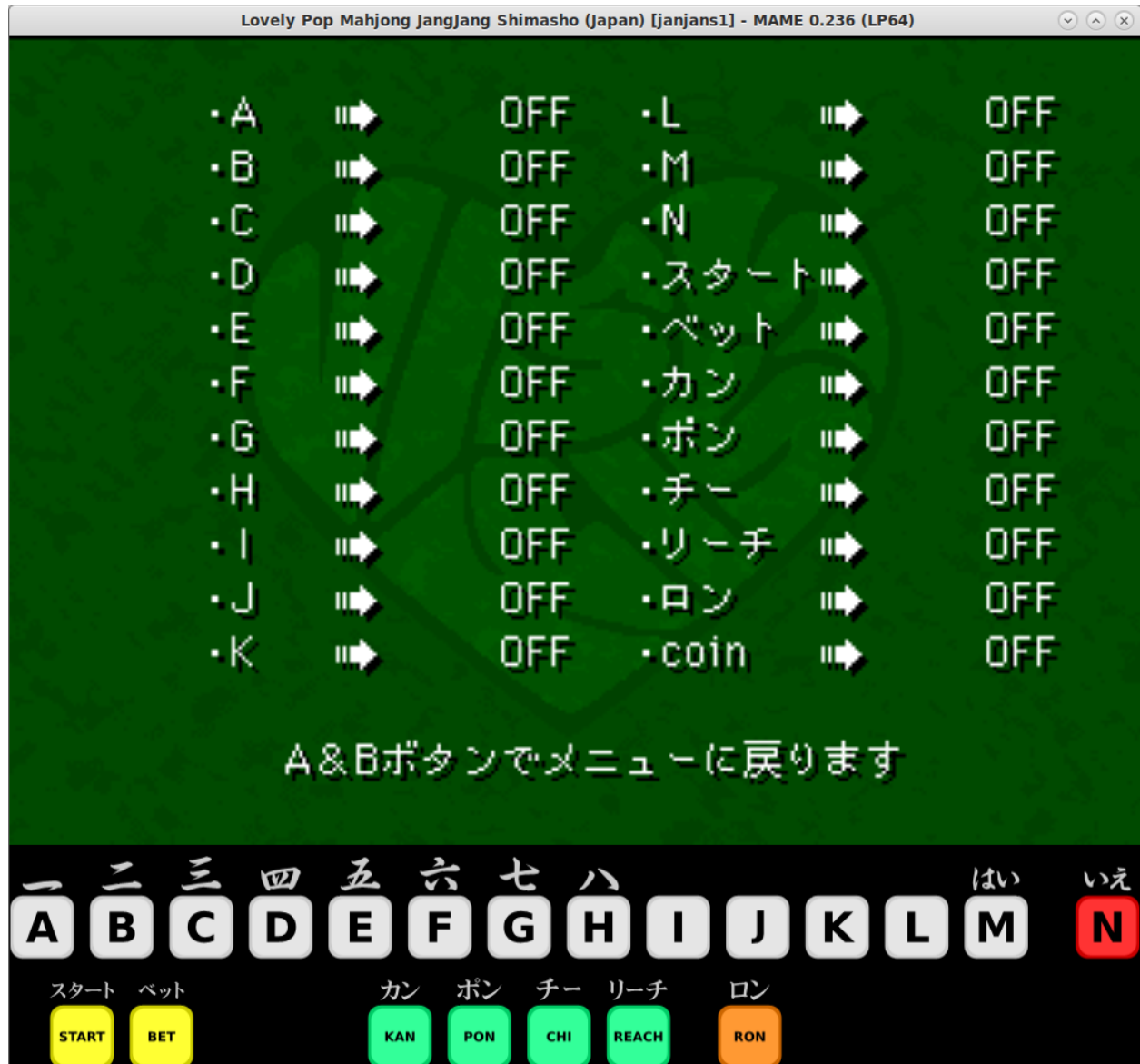


Agora faça Ctrl + A seguido de Shift + Ctrl + G para desagrupar o objeto tela do controle, repare que ao clicar no objeto tela e depois no controle, nós obtemos as coordenadas e seus respectivos tamanhos, é com estas informações que usamos para concluir o nosso layout:

```
<!-- Nome -->
<view name="Controle" showpointers="no">
  <!-- Delimitamos a nossa área de exibição -->
  <bounds x="0" y="0" width="336" height="305.904" />
<screen index="0">
  <!-- O tamanho original da tela -->
  <bounds x="0" y="0" width="336" height="240" />
</screen>
  <!-- Invocamos o nosso controle -->
  <group ref="Controle_mah">
    <bounds x="0.4" y="242" width="335" height="63.904" />
  </group>
</view>
</mamelayout>
```

Observe que os valores **X**, **Y** e **width** de Controle_mah estão um pouco diferentes do que obtivemos com o *Inkscape*, não é algo comum, porém, algumas vezes é necessário para fazer um ajuste fino de alinhamento para que os elementos apareçam em seus devidos lugares na tela do MAME.

Com tudo pronto, rodamos o sistema novamente com `mame janjans1` e entramos no modo de serviço (Tab --> Chaves DIP --> Service Mode --> On, depois Redefine. Clique no botão A na tela uma vez para pular o teste de cores, clique novamente para entrar no modo de teste e pressione Tab para fechar o quadro de informação na tela).



Ao clicar nos botões na tela, os botões da emulação deverão se alternar entre ON e OFF indicando que todos os botões estão funcionando. Retorne ao *Service Mode*, mude a chave para *Off* e clique em *Redefine* para reiniciar o sistema.

12.2.9 Exemplos de outros arquivos layout

Estes arquivos layout demonstram as várias características do sistema de visualização, todos eles estão embutidos no MAME.

- [sstrangr.layout](#)

Um caso simples da utilização de transparências coloridas para visualizar a separação e o destaque dos elementos numa tela preto e branco.

- [seawolf.layout](#)

Este sistema utiliza lâmpadas para os elementos importantes do jogo. Os modos de mesclagem são utilizados para a transparência colorida que é colocada na frente do monitor. Também utiliza collections permitindo que partes do layout sejam desativadas seletivamente.

- [armora.layout](#)

A tela deste jogo é vista diretamente através de uma transparência colorida em vez de ser refletida a partir de dentro do gabinete. Isso significa que a transparência reflete a luz ambiente assim como afeta a cor da imagem do vídeo.

- [tranz330.layout](#)

Uma tela e um teclado alfanumérico com vários segmentos. As teclas são clicáveis e fornecem feedback visual quando pressionadas.

- [esq2by16.layout](#)

Constrói uma matriz de caracteres com múltiplas linhas. As repetições são usadas para evitar a repetição das linhas num caractere, dos caracteres numa linha e das linhas numa página. As cores de grupo permitem que um único elemento seja usado para todas as quatro cores da tela.

- [cgang.layout](#)

Anima a posição dos elementos para simular um jogo de tiro eletromecânico. Também demonstra o uso eficaz dos componentes para a construção de gráficos complexos.

- [minspace.layout](#)

Exibe a posição de um controle deslizante com LEDs.

- [md6802.layout](#)

Usa grupos de forma efetiva como uma linguagem de programação para construir a imagem de um protoboard.

- [beena.layout](#)

Faz o uso de scripts com base em eventos para posicionar dinamicamente os elementos e automaticamente desenhar o conteúdo dos elementos.

12.3 Usando scripts no layout do MAME

- *Introdução*
- *Exemplos práticos*
 - *Espial: joystick dividido entre as portas*
 - *Star Wars: animação com dois eixos*
- *O ambiente do script layout*
- *Os eventos do layout*
 - *Os eventos do arquivo layout*
 - *Os eventos de visualização do layout*
 - *O evento de visualização dos itens do layout*
 - *Eventos de elementos de layout*

12.3.1 Introdução

Os arquivos de layout do MAME podem incorporar scripts Lua para aprimorar a sua funcionalidade. Embora haja muito que você possa fazer com componentes desenhados de forma condicional e a animação dos parâmetros, algumas coisas só podem ser feitas através de scripts. O MAME usa um modelo baseado em eventos. Os scripts podem fornecer funções que serão chamadas após determinados eventos ou quando determinados dados forem necessários.

É preciso que o *layout plugin* esteja ativo. Por exemplo, use este comando para rodar **BWB Double Take** com o script Lua ativo no layout é possível usar este comando:

```
mame -plugins -plugin layout v4dblak
```

Adicione esta configuração num arquivo `v4dblak.ini` dentro do diretório **ini** para que a configuração do layout e do plug-in sempre seja utilizada quando o sistema for iniciado. Consulte *PLUG-INS* para mais informações.

Consulte também o capítulo *Diversos arquivos de configuração*.

12.3.2 Exemplos práticos

Antes de entrarmos de cabeça nos detalhes técnicos do seu funcionamento, começaremos com alguns exemplos usando os aprimoramentos do script Lua. É importante que você tenha conhecimento prévio de como sistema de ilustrações (*artwork*) do MAME funciona, assim como, um conhecimento básico de como criar scripts Lua. Para mais obter mais informações sobre os arquivos de layout consulte [Os arquivos layout](#), para descrições detalhadas das classes Lua usadas pelo MAME consulte [INTERFACE PARA SCRIPTS LUA](#).

Espial: joystick dividido entre as portas

Vejamos estas definições das entradas para o **Espial**:

```
PORT_START("IN1")
PORT_BIT( 0x01, IP_ACTIVE_HIGH, IPT_START1 )
PORT_BIT( 0x02, IP_ACTIVE_HIGH, IPT_START2 )
PORT_BIT( 0x04, IP_ACTIVE_HIGH, IPT_JOYSTICK_LEFT ) PORT_8WAY PORT_COCKTAIL
PORT_BIT( 0x08, IP_ACTIVE_HIGH, IPT_JOYSTICK_RIGHT ) PORT_8WAY PORT_COCKTAIL
PORT_BIT( 0x10, IP_ACTIVE_HIGH, IPT_JOYSTICK_UP ) PORT_8WAY PORT_COCKTAIL
PORT_BIT( 0x20, IP_ACTIVE_HIGH, IPT_JOYSTICK_DOWN ) PORT_8WAY
PORT_BIT( 0x40, IP_ACTIVE_HIGH, IPT_JOYSTICK_DOWN ) PORT_8WAY PORT_COCKTAIL
PORT_BIT( 0x80, IP_ACTIVE_HIGH, IPT_BUTTON2 ) PORT_COCKTAIL

PORT_START("IN2")
PORT_BIT( 0x01, IP_ACTIVE_HIGH, IPT_UNKNOWN )
PORT_BIT( 0x02, IP_ACTIVE_HIGH, IPT_COIN1 )
PORT_BIT( 0x04, IP_ACTIVE_HIGH, IPT_UNKNOWN )
PORT_BIT( 0x08, IP_ACTIVE_HIGH, IPT_JOYSTICK_RIGHT ) PORT_8WAY
PORT_BIT( 0x10, IP_ACTIVE_HIGH, IPT_JOYSTICK_UP ) PORT_8WAY
PORT_BIT( 0x20, IP_ACTIVE_HIGH, IPT_BUTTON1 ) PORT_COCKTAIL
PORT_BIT( 0x40, IP_ACTIVE_HIGH, IPT_BUTTON1 )
PORT_BIT( 0x80, IP_ACTIVE_HIGH, IPT_JOYSTICK_LEFT ) PORT_8WAY
```

Há dois joysticks, um usado por ambos os jogadores num gabinete vertical ou o primeiro jogador num gabinete tipo coquetel e um usado para o segundo jogador num gabinete tipo coquetel. Observe que os interruptores para o primeiro joystick está dividido entre as duas portas de E/S.

Não há sintaxe no arquivo de layout para construir o estado do elemento usando bits das diversas portas de E/S. Também é inconveniente se cada joystick precisar ser definido como um elemento a parte porque os bits para os interruptores não estão dispostos da mesma maneira.

Podemos superar estas limitações usando um script Lua para ler as entradas do jogador e definir o estado dos elementos nos itens:

```
<?xml version="1.0"?>
<mamelayout version="2">
  <!-- o elemento para desenhar um joystick -->
  <!-- cima = 1 (bit 0), baixo = 2 (bit 1), esquerda = 4 (bit 2), direita = 8 (bit 3) -->
  <element name="stick" defstate="0">
    <image state="0x0" file="stick_c.svg" />
    <image state="0x1" file="stick_u.svg" />
    <image state="0x9" file="stick_ur.svg" />
    <image state="0x8" file="stick_r.svg" />
    <image state="0xa" file="stick_dr.svg" />
    <image state="0x2" file="stick_d.svg" />
    <image state="0x6" file="stick_dl.svg" />
    <image state="0x4" file="stick_l.svg" />
    <image state="0x5" file="stick_ul.svg" />
  </element>
  <!-- caso o plug-in do layout não esteja ativo, nós avisaremos o usuário -->
  <!-- desenha apenas quando o seu estado for 1, define o seu estado predefinido para 1 -->
  <!-- assim o aviso fica visível inicialmente -->
  <element name="warning" defstate="1">
    <text state="1" string="Esta ilustração precisa que o plug-in do layout esteja ativo." />
  </element>
  <!-- exibindo a tela e o joystick num gabinete tipo coquetel -->
  <view name="Joystick Display">
    <!-- desenha a tela com a proporção correta -->
    <screen index="0">
      <bounds x="0" y="0" width="4" height="3" />
    </screen>
    <!-- primeiro joystick, o atributo id permite que o script encontre o item -->
    <!-- sem vínculos, o estado será definido pelo script -->
    <element id="joy_p1" ref="stick">
      <!-- posição abaixo da tela -->
      <bounds xc="2" yc="3.35" width="0.5" height="0.5" />
    </element>
    <!-- segundo joystick, o atributo id permite que o script encontre o item -->
    <!-- sem vínculos, o estado será definido pelo script -->
    <element id="joy_p2" ref="stick">
      <!-- a tela é invertida em 180º para o segundo jogador -->
      <orientation rotate="180" />
      <!-- posição acima da tela -->
      <bounds xc="2" yc="-0.35" width="0.5" height="0.5" />
    </element>
    <!-- item com texto de aviso que também possui um atributo id para que o script o encontre -->
    <element id="warning" ref="warning">
      <!-- posição fora da tela próximo da parte de baixo -->
      <bounds x="0.2" y="2.6" width="3.6" height="0.2" />
    </element>
  </view>
  <!-- o conteúdo do elemento do script que será invocado como uma função pelo plug-in do -->
```

(continua na próxima página)

(continuação da página anterior)

```

↪layout -->
    <!-- use um bloco CDATA para evitar a necessidade da utilização dos símbolos "maior que",
↪ "menor que" e sinais tironianos -->
    <script><![CDATA[
        -- o arquivo é um objeto do arquivo do layout
        -- define uma função que será invocada depois de resolver as tags
        file:set_resolve_tags_callback(
            function ()
                -- file.device é o dispositivo que causou a leitura do layout
                -- neste caso, é o principal controlador do sistema espial
                -- consulta as duas portas E/S que precisamos ler
                local in1 = file.device:ioport("IN1")
                local in2 = file.device:ioport("IN2")

                -- consulta os itens view para exibir o estado do joystick
                local p1_stick = file.views["Joystick Display"].items["joy_p1"]
                local p2_stick = file.views["Joystick Display"].items["joy_p2"]

                -- consulte a função que será chamada antes de adicionar os itens que_
↪serão exibidos no destino
                file.views["Joystick Display"]:set_prepare_items_callback(
                    function ()
                        -- faz a leitura da entrada das portas E/S dos dois jogadores
                        local in1_val = in1:read()
                        local in2_val = in2:read()

                        -- define a condição do elemento para o primeiro joystick
                        p1_stick:set_state(
↪IN2 com bit 4 para bit 0
                                ((in2_val & 0x10) >> 4) |    -- muda cima a partir do_
↪do IN1 com bit 5 para bit 1
                                ((in1_val & 0x20) >> 4) |    -- muda baixo a partir_
↪partir do IN2 com bit 7 para bit 2
                                ((in2_val & 0x80) >> 5) |    -- muda esquerda a_
↪com bit 3
                                (in2_val & 0x08))                -- direita está em IN2_

                        -- define a condição do elemento para o primeiro joystick
                        p2_stick:set_state(
↪IN1 com bit 4 para bit 0
                                ((in1_val & 0x10) >> 4) |    -- muda cima a partir do_
↪do IN1 com bit 6 para bit 1
                                ((in1_val & 0x40) >> 5) |    -- muda baixo a partir_
↪com bit 2
                                (in1_val & 0x04) |                -- esquerda está em IN1_
↪com bit 3
                                (in1_val & 0x08))                -- direita está em IN1_

                        end)

                        -- se estivermos com o script rodando, esconde o aviso
                        file.views["Joystick Display"].items["warning"]:set_state(0)
                    end)
    end)

```

(continua na próxima página)

(continuação da página anterior)

```
]]></script>
</mamelayout>
```

O layout tem um elemento script contendo o script Lua que é invocado como uma função através do plug-in **Layout** durante o carregamento do arquivo do layout. A visualização do layout foi construída neste ponto, porém o sistema emulado ainda não terminou de ser iniciado. Não é seguro acessar as entradas e as saídas neste momento. A variável chave no ambiente do script é `file` que dá ao script o acesso ao seu *arquivo de layout*.

Nós fornecemos uma função que será invocada depois que as tags no arquivo de layout tiverem sido resolvidas. Neste ponto, o sistema emulado terá concluído a sua inicialização. Esta função realiza as seguintes tarefas:

- Monitora a entrada das duas *portas E/S* do jogador. As portas E/S podem ser monitoradas através das tags relacionadas com o dispositivo que fizer com que o arquivo de layout seja carregado.
- Monitora os *dois itens* usados pela tela exibindo o estado do joystick. As visualizações podem ser monitoradas através do nome (o valor do atributo `name` por exemplo), e os itens que estiverem entre `view` e que possuam um ID (o valor do atributo `id` por exemplo).
- Fornece uma função que será invocada antes que os itens sejam renderizados na tela.
- Oculta o aviso que lembra o usuário para ativar o plug-in do layout ao definir o estado do elemento para o item com 0 (o componente do texto só é desenhado quando o estado do elemento for 1).

A função que é invocada antes dos itens de visualização são renderizados na tela, lê as entradas do jogador e embaralha os bits na ordem necessária pelo elemento joystick.

Star Wars: animação com dois eixos

Faremos um layout que mostra a posição do manche de voo para o Star Wars da Atari. As portas de entrada são simples, cada eixo analógico produz um valor na faixa entre 0x00(0) a 0xff(255), inclusive:

```
PORT_START("STICKY")
PORT_BIT( 0xff, 0x80, IPT_AD_STICK_Y ) PORT_SENSITIVITY(70) PORT_KEYDELTA(30)

PORT_START("STICKX")
PORT_BIT( 0xff, 0x80, IPT_AD_STICK_X ) PORT_SENSITIVITY(50) PORT_KEYDELTA(30)
```

E aqui temos o nosso layout:

```
<?xml version="1.0"?>
<mamelayout version="2">

  <!-- um quadrado com uma borda branca com 1% da sua largura -->
  <element name="outline">
    <rect><bounds x="0.00" y="0.00" width="1.00" height="0.01" /></rect>
    <rect><bounds x="0.00" y="0.99" width="1.00" height="0.01" /></rect>
    <rect><bounds x="0.00" y="0.00" width="0.01" height="1.00" /></rect>
    <rect><bounds x="0.99" y="0.00" width="0.01" height="1.00" /></rect>
  </element>

  <!-- um retângulo com 10% da linha vertical da sua largura até o meio -->
  <element name="line">
```

(continua na próxima página)

(continuação da página anterior)

```

<!-- use um retângulo transparente para impor as dimensões do elemento -->
<rect>
  <bounds x="0" y="0" width="0.1" height="1" />
  <color alpha="0" />
</rect>
<!-- está é a linha branca que está visível -->
<rect><bounds x="0.045" y="0" width="0.01" height="1" /></rect>
</element>

<!-- o traçado de um quadrado com uma borda interna com 20% e com linhas com 10% do_
↳ comprimento e da largura do elemento -->
<element name="box">
  <!-- use um retângulo transparente para impor as dimensões do elemento -->
  <rect>
    <bounds x="0" y="0" width="0.1" height="0.1" />
    <color alpha="0" />
  </rect>
  <!-- desenha o traçado de um quadrado -->
  <rect><bounds x="0.02" y="0.02" width="0.06" height="0.01" /></rect>
  <rect><bounds x="0.02" y="0.07" width="0.06" height="0.01" /></rect>
  <rect><bounds x="0.02" y="0.02" width="0.01" height="0.06" /></rect>
  <rect><bounds x="0.07" y="0.02" width="0.01" height="0.06" /></rect>
</element>

<!-- caso o plug-in do layout não esteja ativo, nós avisaremos o usuário -->
<!-- desenha apenas quando o seu estado for 1, define o seu estado predefinido para 1_
↳ assim o aviso fica visível inicialmente -->
<element name="warning" defstate="1">
  <text state="1" string="Esta ilustração precisa que o plug-in do layout esteja ativo.
↳ " />
</element>

<!-- visualização exibindo o manche e a sua posição na tela -->
<view name="Analog Control Display">
  <!-- desenha a tela com a correta relação de aspecto -->
  <screen index="0">
    <bounds x="0" y="0" width="4" height="3" />
  </screen>

  <!-- desenha o traçado de um quadrado branco do lado inferior direito da tela -->
  <!-- o script utiliza o tamanho deste item para determinar os limites do seu_
↳ movimento -->
  <element id="outline" ref="outline">
    <bounds x="4.1" y="1.9" width="1.0" height="1.0" />
  </element>

  <!-- linha vertical para exibir os dados recebidos do eixo X -->
  <element id="vertical" ref="line">
    <!-- o elemento desenha uma linha vertical, sem a necessidade de rotacioná-lo -->
    <orientation rotate="0" />
    <!-- centralize horizontalmente no quadrado usando toda a sua altura -->
    <bounds x="4.55" y="1.9" width="0.1" height="1" />

```

(continua na próxima página)

(continuação da página anterior)

```

</element>

<!-- linha horizontal para exibir os dados recebidos do eixo Y -->
<element id="horizontal" ref="line">
    <!-- rotaciona o elemento em 90º para obter uma linha horizontal -->
    <orientation rotate="90" />
    <!-- centraliza verticalmente no quadrado, usando toda a largura -->
    <bounds x="4.1" y="2.35" width="1" height="0.1" />
</element>

<!-- desenhar uma pequena caixa na intersecção das linhas verticais e horizontais -->
<element id="box" ref="box">
    <bounds x="4.55" y="2.35" width="0.1" height="0.1" />
</element>

<!-- desenha um texto de aviso próximo da parte de baixo da tela -->
<element id="warning" ref="warning">
    <bounds x="0.2" y="2.6" width="3.6" height="0.2" />
</element>
</view>

<!-- o conteúdo do elemento do script será invocado como uma função pelo plug-in layout -
->
<!-- use um bloco CDATA para evitar a necessidade da utilização dos símbolos "maior que",
-> "menor que" e sinais tironianos -->
<script><![CDATA[
    -- o arquivo é o objeto do arquivo de layout
    -- define a função para ser invocada depois de resolver as tags
    file:set_resolve_tags_callback(
        function ()
            -- file.device é o dispositivo que fez com que o layout fosse carregado
            -- neste caso, é o condutor principal do sistema starwars
            -- localize as entradas dos eixos analógicos
            local x_input = file.device:ioport("STICKX")
            local y_input = file.device:ioport("STICKY")

            -- localize o esboço do item
            local outline_item = file.views["Analog Control Display"].items["outline
->"]

            -- variáveis para manter o estado através das chamadas
            local outline_bounds    -- a delineação do esboço do quadrado
            local width, height    -- largura e altura dos itens animados
            local x_scale, y_scale -- relação das unidades dos eixos para_
->renderizar as coordenadas
            local x_pos, y_pos     -- exibe as posições para os itens animados

            -- define uma função que será invocada quando as dimensões da_
->visualização forem recalculadas
            -- isso pode acontecer quando a janela for redimensionada ou as opções_
->de escala forem alteradas
            file.views["Analog Control Display"]:set_recomputed_callback(

```

(continua na próxima página)

(continuação da página anterior)

```

function ()
    -- obtém a delineação do esboço do quadrado
    outline_bounds = outline_item.bounds
    -- animação dos itens, use 10% da largura e altura do_

↪quadrado

    width = outline_bounds.width * 0.1
    height = outline_bounds.height * 0.1
    -- calcula as proporções das unidades do eixo para_

↪renderizar as coordenadas

    -- animação dos itens, deixe 90% da largura e altura para o_

↪limite do movimento

    -- o limite do percurso de cada eixo fica em 0xff
    x_scale = outline_bounds.width * 0.9 / 0xff
    y_scale = outline_bounds.height * 0.9 / 0xff
end)

-- define uma função para ser invocada antes de adicionar a visualização_
↪dos itens no destino renderizado
file.views["Analog Control Display"]:set_prepare_items_callback(
    function ()
        -- lê os eixos analógicos, eixo Y invertido como zero está_

↪na parte de baixo

        local x = x_input:read() & 0xff
        local y = 0xff - (y_input:read() & 0xff)
        -- converte os valores recebidos para as coordenadas do_

↪layout

        -- usa a quina superior esquerda do quadrado delineado como_

↪a sua origem

        x_pos = outline_bounds.x0 + (x * x_scale)
        y_pos = outline_bounds.y0 + (y * y_scale)
    end)

-- define uma função para fornecer os limites da linha vertical
file.views["Analog Control Display"].items["vertical"]:set_bounds_
↪callback(
    function ()
        -- renderize a delineação de um novo objeto (começando como_

↪uma unidade quadrada)

        local result = emu.render_bounds()
        -- define esquerda, cima, largura e altura
        result:set_wh(
            x_pos,                -- posição X calculada para_

↪os itens animados

            outline_bounds.y0,    -- delineação do topo do_

↪quadrado

            width,                -- 10% da largura do_

↪quadrado delineado

            outline_bounds.height) -- altura total do quadrado_

↪delineado

        return result
    end)

```

(continua na próxima página)

(continuação da página anterior)

```

-- define uma nova função para informar a delineação da linha horizontal
file.views["Analog Control Display"].items["horizontal"]:set_bounds_
↪callback(
    function ()
        -- renderize a delineação de um novo objeto (começando como_
↪uma unidade quadrada)
        local result = emu.render_bounds()
        -- define esquerda, cima, largura e altura
        result:set_wh(
            outline_bounds.x0,      -- esquerda do quadrado_
↪delineado
            y_pos,                  -- posição Y calculada para_
↪os itens animados
            outline_bounds.width,   -- largura total do quadrado_
↪delineado
            height)                -- 10% da altura do quadrado_
↪delineado
        return result
    end)

-- define uma nova função para informar a delineação da caixa entre a_
↪interseção das linhas
file.views["Analog Control Display"].items["box"]:set_bounds_callback(
    function ()
        -- renderize uma nova delineação de objeto (começando como_
↪uma unidade quadrada)
        local result = emu.render_bounds()
        -- define esquerda, cima, largura e altura
        result:set_wh(
            x_pos,                  -- posição X calculada para_
↪os itens animados
            y_pos,                  -- posição Y calculada para_
↪os itens animados
            width,                  -- 10% da largura do_
↪quadrado delineado
            height)                -- 10% da altura do quadrado_
↪delineado
        return result
    end)

-- oculta o aviso uma vez que se chegamos até aqui, o script já está_
↪rodando
file.views["Analog Control Display"].items["warning"]:set_state(0)
end)
]]></script>
</mamelayout>

```

O layout possui um elemento script contendo o script Lua que será invocado como uma função através do plug-in **Layout** quando o arquivo de layout for carregado. Isto ocorre após a construção das visualizações do layout, mas antes que o sistema emulado tenha concluído a sua inicialização. O objeto do *arquivo do layout* é fornecido ao script através da variável `file`.

Nós oferecemos uma função que será invocada depois que as tags no arquivo do layout forem resolvidas. Esta função faz o seguinte:

- Monitora o recebimento de dados da *entrada* do eixo analógico.
- Monitora o *item visualizado* que traça o contorno da área onde a posição do manche é exibido.
- Declara algumas variáveis para manter os valores calculados através das chamadas das funções.
- Fornece a função para ser invocada quando a visualização das dimensões tenham sido recalculadas.
- Fornece a função para ser invocada antes de adicionar os itens visíveis ao contêiner durante a renderização do quadro.
- Fornece as funções que definirão os limites para os itens animados.
- Esconde o aviso que alerta o usuário para ativar o plug-in **Layout** ao definir a condição do elemento para o item como 0 (o componente do texto só é desenhado quando o estado do elemento for 1).

A visualização é monitorada através do nome (pelo valor do seu atributo name) e os itens dentro da visualização são monitoradas através do ID (com o valor dos seus respectivos atributos id).

As dimensões de visualização do layout são recalculadas em resposta a vários eventos, incluindo o redimensionamento da janela, entrando ou saindo do modo de tela cheia, alternando a visibilidade das coleções dos itens e mudando o zoom para a configuração da área da tela. Quando isso acontece, precisamos atualizar os nossos fatores de tamanho e da escala da animação. Obtemos os limites do quadrado onde a posição do manche é exibido, calculamos o tamanho dos itens animados e calculamos as proporções das unidades do eixo para renderizar as coordenadas do alvo para cada direção. É mais eficiente fazer estes cálculos somente caso os resultados mudem.

Antes dos itens de visualização serem adicionados no destino da renderização, lemos as entradas do eixo analógico e convertemos os valores da posição em coordenadas para a animação dos os itens. A entrada do eixo Y usa valores maiores para apontar para cima, então precisamos inverter o valor subtraindo-o de 0xff (255). Adicionamos nas coordenadas do canto superior esquerdo do quadrado onde estamos exibindo a posição do manche. Fazemos isso uma vez cada vez que o layout for desenhado por questões de eficiência já que podemos usar os valores para todos os três itens animados.

Finalmente, fornecemos limites para a animação dos itens quando necessário. Estas funções precisam retornar os objetos "render_bounds" dando a posição e o tamanho dos itens como coordenadas do alvo que serão renderizados.

Como os elementos da linha vertical e da linha horizontal movem-se cada um apenas num único eixo, seria possível animá-los usando os recursos de animação do arquivo de layout. Na verdade apenas a caixa na interseção da linha precisa de um script. É feito totalmente com script para fins ilustrativos.

12.3.3 O ambiente do script layout

O ambiente Lua é oferecido pelo plug-in **Layout**. É bem reduzido, oferecendo apenas o mínimo necessário:

- O file oferecendo o objeto do *arquivo de layout* do script. Possui uma propriedade device para saber quem foi qual foi o *dispositivo* responsável para que o layout fosse carregado e uma propriedade views para conseguir as *exibições do layout* (indexadas através do nome).
- A função machine que oferece ao MAME a informação sobre a *máquina em execução* no momento.
- As funções emu.device_enumerator, emu.palette_enumerator, emu.screen_enumerator, emu.cassette_enumerator, emu.image_enumerator e emu.slot_enumerator para obter as interfaces de dispositivos específicos.
- As funções emu.attotime, emu.render_bounds e emu.render_color que criam os objetos *attotime*, *bounds* e *cores*.
- emu.bitmap_ind8, emu.bitmap_ind16, emu.bitmap_ind32, emu.bitmap_ind64, emu.bitmap_yuy16, emu.bitmap_rgb32 e objetos emu.bitmap_argb32 para criar *bitmaps*.
- As funções emu.render_bounds e o emu.render_color criam os limites e as cores dos objetos.
- As funções emu.print_verbose, emu.print_error, emu.print_warning, emu.print_info e o emu.print_debug são usadas para diagnósticos.
- Padrão Lua, funções tonumber, tostring, pairs e ipairs, assim como objetos table string para manipular strings, tabelas e os outros contêineres.
- Função Lua print para gerar texto no console.

12.3.4 Os eventos do layout

O script do layout do MAME usa um modelo com base em eventos. Os scripts podem fornecer funções que serão invocadas após a ocorrência dos eventos ou quando os dados forem solicitados. Há três níveis de eventos: do arquivo do layout, da visualização do layout e do item de visualização de layout.

Os eventos do arquivo layout

Os eventos do arquivo do layout é aplicado no arquivo como um todo e não numa visualização individualmente.

Resolve as tags

```
file:set_resolve_tags_callback(cb)
```

É invocado após o sistema que está sendo emulado ter terminado a sua inicialização, as tags do layout que forem recebidas tenham sido resolvidas e as invocações retornadas tenham sido configuradas. Este é um bom momento para consultar as entradas e configurar os manipuladores dos eventos do item de visualização.

O retorno de chamada não retorna nenhum valor e também não aceita parâmetros. Use nil como um argumento para remover o manipulador do evento.

Os eventos de visualização do layout

Os eventos da visualização do Layout sem aplicam para uma visualização individual.

Prepara os itens

`view:set_prepare_items_callback(cb)`

É invocado antes que a renderização de visualização dos itens sejam adicionados no destino em preparação para conceber o quadro de vídeo.

O retorno de chamada não retorna nenhum valor e também não aceita parâmetros. Use `nil` como um argumento para remover o manipulador do evento.

Carga prévia

`view:set_preload_callback(cb)`

É invocado após a carga prévia dos elementos visíveis da visualização. Isso pode acontecer quando a visualização é selecionada pela primeira vez durante a seção ou caso o usuário alterne a exibição da coleção de um elemento. Esteja ciente que isto pode ser invocado várias vezes durante uma seção, evite a repetição de tarefas onerosas ao sistema.

O retorno de chamada não retorna nenhum valor e não também aceita nenhum parâmetro. Use `nil` como um argumento para remover o manipulador do evento.

O recálculo das dimensões

`view:set_recomputed_callback(cb)`

É invocado quando as visualizações forem recalculadas. Isso acontece em várias situações, inclusive quando a janela for redimensionada, entrando ou saindo do modo de tela cheia, alternando as visualizações de um item numa coleção e alterando as configurações de rotação e zoom da tela. Caso esteja animando a posição dos itens visualizados, este é um bom momento para calcular os fatores de escala e posição.

O retorno de chamada não retorna nenhum valor e não também aceita nenhum parâmetro. Use `nil` como um argumento para remover o manipulador do evento.

Atualizações do ponteiro

`view:set_pointer_updated_callback(cb)`

É invocado quando o ponteiro ingressar, se mover ou altera o estado do botão na exibição.

São passados nove argumentos para a função de retorno de chamada:

- O tipo do ponteiro como uma *string*. Os valores válidos são `mouse`, `pen`, `touch` ou `unknown` que não se alterará durante a vida do ponteiro.
- A ID do ponteiro. Este será um inteiro não negativo que não se alterará durante a vida do ponteiro. Os valores da ID podem ser recicladas de maneira "agressiva".
- A ID do dispositivo. Para grupos de ponteiros para reconhecer gestos multitoque.
- A posição horizontal nas coordenadas do layout.
- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização.

- Uma máscara de bits que representa os botões pressionados no momento. O botão primário é o bit menos importante.
- Uma máscara de bits que representa os botões que foram pressionados nesta atualização. O botão primário é o bit menos importante.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização. O botão primário é o bit menos importante.
- A contagem de cliques. É positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)).

O retorno de chamada não retorna nenhum valor. Use `nil` como um argumento para remover o manipulador do evento.

O ponteiro esquerdo

`view:set_pointer_left_callback(cb)`

É invocado quando um ponteiro deixa a exibição normalmente. Após receber este evento, a ID do ponteiro pode ser reutilizada com um novo ponteiro.

São passados sete argumentos para a função de retorno de chamada:

- O tipo do ponteiro como uma *string*. Os valores válidos são `mouse`, `pen`, `touch` ou `unknown` que não se alterará durante a vida do ponteiro.
- A ID do ponteiro. Este será um inteiro não negativo que não se alterará durante a vida do ponteiro. Os valores da ID podem ser recicladas de maneira "agressiva".
- A ID do dispositivo. Para grupos de ponteiros para reconhecer gestos multitoque.
- A posição horizontal nas coordenadas do layout.
- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização. O botão primário é o bit menos importante.
- A contagem de cliques. É positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)). Se aplica apenas ao botão primário.

O retorno de chamada não retorna nenhum valor. Use `nil` como um argumento para remover o manipulador do evento.

Abortando o ponteiro

`view:set_pointer_aborted_callback(cb)`

É invocado quando um ponteiro deixa a exibição de maneira anormal. Após receber este evento, a ID do ponteiro pode ser reutilizada com um novo ponteiro.

São passados sete argumentos para a função de retorno de chamada:

- O tipo do ponteiro como uma *string*. Os valores válidos são `mouse`, `pen`, `touch` ou `unknown` que não se alterará durante a vida do ponteiro.
- A ID do ponteiro. Este será um inteiro não negativo que não se alterará durante a vida do ponteiro. Os valores da ID podem ser recicladas de maneira "agressiva".
- A ID do dispositivo. Para grupos de ponteiros para reconhecer gestos multitoque.
- A posição horizontal nas coordenadas do layout.

- A posição vertical nas coordenadas do layout.
- Uma máscara de bits que representa os botões que foram liberados nesta atualização. O botão primário é o bit menos importante.
- A contagem de cliques. É positivo para ações com clique simultâneo ou negativo se um clique for transformado numa retenção (clicar e manter) ou arraste (clicar e arrastar)). Se aplica apenas ao botão primário.

O retorno de chamada não retorna nenhum valor. Use `nil` como um argumento para remover o manipulador do evento.

Esquecendo os ponteiros

```
view:set_forget_pointers_callback(cb)
```

É invocado quando a visualização deve parar de processar a entrada do ponteiro. Isso pode ocorrer em várias situações, incluindo:

- Quando o usuário ativar um menu.
- Quando a configuração da visualização for alterada.
- Quando a visualização for desativada.

O evento de visualização dos itens do layout

O retorno da visualização dos itens do layout se aplicam aos itens individuais dentro da visualização. Eles são usados para sobrescrever a condição predefinida do elemento do item, o estado de animação, limites e o comportamento da cor.

Obtém o estado do elemento

```
item:set_element_state_callback(cb)
```

Define um retorno de chamada para obter o estado dos itens. Este controla como o elemento do item é desenhado, para componentes que mudam a aparência dependendo do seu estado para desenhar os componentes de forma condicional e o limite, cor da animação dos componentes. Não tente acessar o `element_state` dos itens a partir do retorno de chamada pois isso resultará numa repetição infinita.

O retorno de chamada não retorna nenhum valor e também não aceita parâmetros. Use `nil` como um argumento para restaurar o estado do manipulador do evento (com base nos atributos XML dos itens).

Obtém o estado da animação

```
item:set_animation_state_callback(cb)
```

Define um retorno de chamada para obter o estado de animação do item. É utilizado para as animações dos limites e das cores da animação. Não tente acessar o `animation_state` do item a partir do retorno de chamada pois ocorrerá uma recorrência infinita.

O retorno de chamada deve retornar um número inteiro e também não aceita nenhum parâmetro. Use `nil` como um argumento para restaurar o estado original do manipulador do evento de animação (com base nos atributos XML dos itens e do sub-elemento `animate`).

Obtém os limites do item

```
item:set_bounds_callback(cb)
```

Define um retorno de chamada para obter os limites do item (a sua posição e o seu tamanho). Não tente acessar o `bounds` do item a partir do retorno de chamada pois ocorrerá uma recorrência infinita.

O retorno de chamada deve retornar os limites da renderização do objeto representando os limites do item em coordenadas do seu destino (geralmente criado ao invocar o `emu.render_bounds`) e também não aceita nenhum parâmetro. Use `nil` como um argumento para restaurar o limite original do manipulador do evento (com base no estado de animação do item e do sub-elemento `bounds`).

Obtém a cor do item

```
item:set_color_callback(cb)
```

Define um retorno de chamada para obter a cor de um item (a textura da cor do elemento multiplicado por esta cor)

O retorno de chamada deve retornar a renderização da cor do objeto representando a cor ARGB (geralmente criado ao invocar o `emu.render_color`) e também não aceita parâmetros. Use `nil` como um argumento para restaurar a cor original do manipulador do evento (com base no estado de animação do item e do sub-elemento `color`).

Obtém o tamanho da rolagem horizontal do item da janela

```
item:set_scroll_size_x_callback(cb)
```

Define um retorno de chamada para obter o tamanho da rolagem horizontal do item da janela. Isto permite que o script controle o quanto do elemento será exibido pelo item. Não tente acessar a propriedade `scroll_size_x` do item a partir do retorno de chamada, pois isso resultará numa repetição infinita.

O retorno de chamada deve retornar um número de ponto flutuante representando o tamanho horizontal da janela como uma proporção da largura dos elementos associados e não aceita quaisquer parâmetros. Um valor `1.0` exibirá a largura total do elemento; valores menores exibem as partes com uma proporção menor do elemento. Use `nil` como um argumento para restaurar o tamanho padrão da rolagem horizontal da janela (com base no sub-elemento `xscroll`).

Obtém o tamanho da rolagem vertical do item da janela

```
item:set_scroll_size_y_callback(cb)
```

Define um retorno de chamada para obter o tamanho da rolagem vertical do item da janela. Isto permite que o script controle o quanto do elemento será exibido pelo item. Não tente acessar a propriedade `scroll_size_y` do item a partir do retorno de chamada, pois isso resultará numa repetição infinita.

O retorno de chamada deve retornar um número de ponto flutuante representando o tamanho vertical da janela como uma proporção da altura dos elementos associados e não aceita quaisquer parâmetros. Um valor `1.0` exibirá a altura total do elemento; valores menores exibem as partes com uma proporção menor do elemento. Use `nil` como um argumento para restaurar o tamanho padrão da rolagem vertical da janela (com base no sub-elemento `yscroll`).

Obtém a posição da rolagem horizontal do item

```
item:set_scroll_pos_x_callback(cb)
```

Define um retorno de chamada para obter a posição da rolagem horizontal do item. Isto permite que o script controle qual parte do elemento seja exibido pelo item. Não tente acessar a propriedade `scroll_pos_x` do item a partir do retorno de chamada, pois isso resultará numa repetição infinita.

O retorno de chamada deve retornar um número de ponto flutuante e não aceita parâmetros. Um valor `0.0` alinha a borda esquerda do elemento com a borda esquerda do item; valores maiores deslocam para à direita. Use `nil` como um argumento para restaurar o manipulador da posição da rolagem horizontal padrão (com base nas ligações no sub-elemento `xscroll`).

Obtém a posição da rolagem vertical do item

```
item:set_scroll_pos_y_callback(cb)
```

Define um retorno de chamada para obter a posição da rolagem vertical do item. Isto permite que o script controle qual parte do elemento seja exibido pelo item. Não tente acessar a propriedade `scroll_pos_y` do item a partir do retorno de chamada, pois isso resultará numa repetição infinita.

O retorno de chamada deve retornar um número de ponto flutuante e não aceita parâmetros. Um valor `0.0` alinha a borda superior do elemento com a borda superior do item; valores maiores deslocam para baixo. Use `nil` como um argumento para restaurar o manipulador da posição da rolagem vertical padrão (com base nas ligações no sub-elemento `yscroll`).

Eventos de elementos de layout

Os eventos de elementos de layout se aplicam à definição de um elemento visual individual.

Draw

```
element:set_draw_callback(cb)
```

Defina um retorno de chamada para um desenho adicional após os componentes do elemento serem desenhados. Isso oferece controle direto ao script sobre a textura final ao desenhar o elemento.

O retorno de chamada recebe dois argumentos, o estado do elemento (um número inteiro) e um bitmap ARGB com 32 bits no tamanho desejado. O retorno de chamada não deve tentar redimensionar o bitmap. Use nil como um argumento para remover o manipulador do evento.

12.4 Os localizadores de objetos

- *Introdução*
- *Os tipos do localizador dos objetos*
- *Localizando os recursos*
- *As conexões entre os dispositivos*
- *As arrays para a localização dos objetos*
- *Localizadores opcionais de objetos*
 - *Componentes opcionais do sistema*
 - *Recursos Opcionais*
- *Os tipos dos localizadores de objetos com mais detalhes*
 - *Os localizadores dos dispositivos*
 - *O sistema de memória dos localizadores de objetos*
 - *O localizador das portas E/S*
 - *Os localizadores dos espaços nos endereços*
 - *Os localizadores dos ponteiros da memória*
- *Os localizadores das saídas*

12.4.1 Introdução

Os localizadores de objetos são uma parte importante oferecida pelo MAME como uma junção na união dos dispositivos que formam a emulação de um sistema. São usados para definir as conexões entre os dispositivos acessando os seus recursos de forma eficiente e também confirmando se os mesmos estão disponíveis durante a validação.

Os localizadores procuram um objeto alvo através de uma etiqueta através de um dispositivo base. Para alguns tipos são necessários parâmetros adicionais.

A maioria dos localizadores possuem versões obrigatórias e também versões opcionais. As versões obrigatórias geram um erro caso o objeto não seja encontrado no destino. Isso prevenirá que um dispositivo seja iniciado inicie ou gere um erro de validação. As versões opcionais irão registrar uma mensagem detalhada no caso do objeto não ter sido encontrado no destino ou então, forneça membros de teste caso o objeto no destino tenha ou não sido encontrado.

As classes do localizador são declaradas no cabeçalho `src/emu/devfind.h` e tem a sua API documentada em formato Doxygen.

12.4.2 Os tipos do localizador dos objetos

required_device<DeviceClass>, optional_device<DeviceClass>

Encontra um dispositivo. O modelo do argumento DeviceClass deve ser uma classe derivada a partir do device_t ou do device_interface.

required_memory_region, optional_memory_region

Encontra uma região da memória, geralmente a partir das definições das ROMs. O alvo é o objeto memory_region.

required_memory_bank, optional_memory_bank

Encontra um banco de memória instanciado num mapa dos endereços. O alvo é o objeto memory_bank.

memory_bank_creator

Encontra um banco de memória instanciado num mapa dos endereços ou o cria caso ele ainda não exista. O alvo é o objeto memory_bank. Não há versão opcional, porque o objeto no destino sempre será encontrado ou será criado.

required_ioport, optional_ioport

Encontra uma porta de E/S nas definições da porta de entrada num dispositivo. O alvo é o objeto ioport_port.

required_address_space, optional_address_space

Encontra um espaço dos endereços de um dispositivo. O alvo é o objeto address_space.

required_region_ptr<PointerType>, optional_region_ptr<PointerType>

Encontra o ponteiro base de uma região da memória, geralmente a partir das definições das ROMs. O argumento do modelo PointerType é o tipo do destino (geralmente um tipo inteiro e sem assinatura). O alvo é o primeiro elemento na região compartilhada da memória.

required_shared_ptr<PointerType>, optional_shared_ptr<PointerType>

Encontra o ponteiro base de um compartilhamento da memória instanciado num mapa dos endereços. O argumento do modelo PointerType é o tipo do destino (geralmente um tipo inteiro e sem assinatura). O alvo é o primeiro elemento na região compartilhada da memória.

memory_share_creator<PointerType>

Encontra o ponteiro base de um compartilhamento da memória instanciado num mapa dos endereços, ou o cria caso um ainda não exista. O argumento do modelo PointerType é o tipo do destino (geralmente um tipo inteiro sem assinatura). O alvo é o primeiro elemento no compartilhamento da memória. Não há versão opcional porque o objeto alvo sempre será encontrado ou será criado.

12.4.3 Localizando os recursos

Começaremos com um exemplo simples de um dispositivo que utiliza os localizadores dos objetos para acessar a ramificação dos seus próprios dispositivos, as entradas e a região da ROM. Os exemplos do código aqui presente tem base na placa da interface da impressora paralela do Apple II, porém visando a simplificação dos exemplos, muita coisa foi removida.

Os localizadores são declarados como membros da classe do dispositivo:

```
class a2bus_parprn_device : public device_t, public device_a2bus_card_interface
{
public:
    a2bus_parprn_device(machine_config const &mconfig, char const *tag, device_t *owner,
↳ u32 clock);

    virtual void write_c0nx(u8 offset, u8 data) override;
    virtual u8 read_cnxx(u8 offset) override;

protected:
    virtual tiny_rom_entry const *device_rom_region() const override;
    virtual void device_add_mconfig(machine_config &config) override;
    virtual ioport_constructor device_input_ports() const override;

private:
    required_device<centronics_device>      m_printer_conn;
    required_device<output_latch_device>    m_printer_out;
    required_ioport                          m_input_config;
    required_region_ptr<u8>                  m_prom;
};
```

Queremos encontrar um centronics_device, um output_latch_device, uma porta E/S e uma região da memória 8 bits.

No construtor, definimos o alvo inicial para os localizadores:

```
a2bus_parprn_device::a2bus_parprn_device(machine_config const &mconfig, char const *tag,
↳ device_t *owner, u32 clock) :
    device_t(mconfig, A2BUS_PARPRN, tag, owner, clock),
    device_a2bus_card_interface(mconfig, *this),
    m_printer_conn(*this, "prn"),
    m_printer_out(*this, "prn_out"),
    m_input_config(*this, "CFG"),
    m_prom(*this, "prom")
{
}
```

Cada localizador recebe um dispositivo base e uma etiqueta com argumentos do construtor. O dispositivo base informado na construção atende a dois propósitos. O mais óbvio, a etiqueta, que é definida em relação a este dispositivo e provavelmente o mais importante, o objeto se registra com este dispositivo para que seja chamado para realizar a validação e a resolução dos objetos.

Observe que os localizadores *não* copiam as strings das etiquetas. O solicitante deve garantir que a string da tag se mantenha válida até o final da validação e/ou até que a resolução seja concluída e que a região da memória e da porta de E/S venha a partir da definição e da entrada da ROM respectivamente:

```
namespace {

ROM_START(parprn)
    ROM_REGION(0x100, "prom", 0)
    ROM_LOAD( "prom.b4", 0x0000, 0x0100, BAD_DUMP_CRC(00b742ca)_
    ↪SHA1(c67888354aa013f9cb882eeeed924e292734e717) )
ROM_END

INPUT_PORTS_START(parprn)
    PORT_START("CFG")
    PORT_CONFNAME(0x01, 0x00, "Acknowledge latching edge")
    PORT_CONFSETTING( 0x00, "Falling (/Y-B)")
    PORT_CONFSETTING( 0x01, "Rising (Y-B)")
    PORT_CONFNAME(0x06, 0x02, "Printer ready")
    PORT_CONFSETTING( 0x00, "Always (S5-C-D)")
    PORT_CONFSETTING( 0x02, "Acknowledge latch (Z-C-D)")
    PORT_CONFSETTING( 0x04, "ACK (Y-C-D)")
    PORT_CONFSETTING( 0x06, "/ACK (/Y-C-D)")
    PORT_CONFNAME(0x08, 0x00, "Strobe polarity")
    PORT_CONFSETTING( 0x00, "Negative (S5-A-/X, GND-X)")
    PORT_CONFSETTING( 0x08, "Positive (S5-X, GND-A-/X)")
    PORT_CONFNAME(0x10, 0x10, "Character width")
    PORT_CONFSETTING( 0x00, "7-bit")
    PORT_CONFSETTING( 0x10, "8-bit")
INPUT_PORTS_END

} // anonymous namespace

tiny_rom_entry const *a2bus_parprn_device::device_rom_region() const
{
    return ROM_NAME(parprn);
}

ioport_constructor a2bus_parprn_device::device_input_ports() const
{
    return INPUT_PORTS_NAME(parprn);
}
```

Observe que as etiquetas "prom" e o "CFG" correspondem às etiquetas passadas ao objeto em construção.

A ramificação dos dispositivos são instanciados na função do membro de configuração do sistema do dispositivo:

```
void a2bus_parprn_device::device_add_mconfig(machine_config &config)
{
    CENTRONICS(config, m_printer_conn, centronics_devices, "printer");
    m_printer_conn->ack_handler().set(FUNC(a2bus_parprn_device::ack_w));

    OUTPUT_LATCH(config, m_printer_out);
    m_printer_conn->set_output_latch(*m_printer_out);
}
```

Os localizadores são passados para os tipos dos dispositivos para fornecer as etiquetas ao instanciar os dispositivos herdados. Depois de instanciar um dispositivo ramificado desta forma, o objeto pode ser utilizado como um ponteiro para o dispositivo até o final da configuração da função do membro de configuração do sistema. Observe que para usar um localizador como este, o seu dispositivo base deve ser o mesmo que o dispositivo que está sendo configurado (o ponteiro `this` da função do membro de configuração do sistema).

Após a inicialização do sistema emulado, os localizadores podem ser usados da mesma maneira que os ponteiros:

```
void a2bus_parprn_device::write_cnxx(u8 offset, u8 data)
{
    ioport_value const cfg(m_input_config->read());

    m_printer_out->write(data & (BIT(cfg, 8) ? 0xffU : 0x7fU));
    m_printer_conn->write_strobe(BIT(~cfg, 3));
}

u8 a2bus_parprn_device::read_cnxx(u8 offset)
{
    offset ^= 0x40U;
    return m_prom[offset];
}
```

Por questão de conveniência, os localizadores que visam o ponteiro base das regiões da memória e os compartilhamentos podem ser indexados como arrays.

12.4.4 As conexões entre os dispositivos

Os dispositivos precisam estar conectados num sistema. No Sun SBus por exemplo, o dispositivo precisa de acesso à CPU do host e ao espaço do endereço. É assim que declaramos os localizadores na classe do dispositivo (com todas as distrações removidas):

```
DECLARE_DEVICE_TYPE(SBUS, sbus_device)

class sbus_device : public device_t, public device_memory_interface
{
    template <typename T, typename U>
    sbus_device(
        machine_config const &mconfig, char const *tag, device_t *owner, u32_
        ↪clock,
            T &&cpu_tag,
            U &&space_tag, int space_num) :
        sbus_device(mconfig, tag, owner, clock)
    {
        set_cpu(std::forward<T>(cpu_tag));
        set_type1space(std::forward<U>(space_tag), space_num);
    }

    sbus_device(machine_config const &mconfig, char const *tag, device_t *owner, u32_
        ↪clock) :
        device_t(mconfig, SBUS, tag, owner, clock),
        device_memory_interface(mconfig, *this),
        m_maincpu(*this, finder_base::DUMMY_TAG),
        m_type1space(*this, finder_base::DUMMY_TAG, -1)
    {
    }

    template <typename T> void set_cpu(T &&tag) { m_maincpu.set_tag(std::forward<T>
        ↪(tag)); }
    template <typename T> void set_type1space(T &&tag, int num) { m_type1space.set_
        ↪tag(std::forward<T>(tag), num); }

protected:
    required_device<sparc_base_device> m_maincpu;
    required_address_space m_type1space;
};
```

Há algumas coisas que podem ser observadas aqui:

- Os membros do localizador são declarados para tudo que o dispositivo precisar acessar.
- O dispositivo não sabe como se encaixará num sistema maior, o localizadores são construídos com argumentos fictícios.
- As funções do membro da configuração são providas para definir a tag para o host da CPU e a tag e o índice para o espaço do endereço tipo 1.
- Além do construtor do dispositivo padrão é provido um construtor com parâmetros para definir a CPU e espaço do endereço tipo 1.

A constante `finder_base::DUMMY_TAG` é garantida como sendo inválida e não será resolvida para um objeto. Isso torna mais fácil detectar as configurações que forem incompletas ao relatar um erro. Os espaços dos endereços são numerados a partir do zero, logo, haverá um erro caso o mesmo seja negativo.

As funções do membro para configurar os localizadores dos objetos tomam uma referência universal a um objeto semelhante a uma etiqueta (um tipo modelado com o qualificador &&), bem como qualquer outros parâmetros necessários para o tipo específico do localizador de objeto. Um localizador do espaço de endereço precisa de um número além de um objeto semelhante a uma etiqueta.

Então o que é seria um objeto semelhante a uma etiqueta?

Há suporte para três coisas:

- Um ponteiro de string C (char const *) representando uma etiqueta relativa ao dispositivo que estiver sendo configurado. Observe que o localizador de objetos não copiará a string. O chamado deve garantir que continuará válido até a sua resolução e/ou que a validação seja concluída.
- Um outro localizador de objetos, o localizador de objetos assumirá o seu alvo atual.
- Para os localizadores dos dispositivos, uma referência para uma instância do tipo de dispositivo do destino, definindo o alvo para este dispositivo. Observe que não irá funcionar caso o dispositivo seja posteriormente substituído na configuração do sistema. Em geral é mais utilizado com *this.

O construtor adicional que define a configuração inicial delega para o construtor padrão e em seguida chama as funções do membro da configuração apenas por conveniência.

Quando queremos instanciar este dispositivo e conectá-lo, fazemos o seguinte:

```
SPARCV7(config, m_maincpu, 20'000'000);  
  
ADDRESS_MAP_BANK(config, m_type1space);  
  
SBUS(config, m_sbus, 20'000'000);  
m_sbus->set_cpu(m_maincpu);  
m_sbus->set_type1space(m_type1space, 0);
```

Nós fornecemos os mesmos localizadores de objetos para instanciar a CPU, o espaço do endereço dos dispositivos e para configurar o dispositivo SBus.

Observe que também podemos usar strings literais C para configurar o dispositivo SBus ao custo de precisar atualizar as etiquetas em diferentes lugares caso elas se alterem:

```
SBUS(config, m_sbus, 20'000'000);  
m_sbus->set_cpu("maincpu");  
m_sbus->set_type1space("type1", 0);
```

Caso queira utilizar o construtor por questão de conveniência, fornecemos apenas os argumentos ao instanciar o dispositivo:

```
SBUS(config, m_sbus, 20'000'000, m_maincpu, m_type1space, 0);
```

12.4.5 As arrays para a localização dos objetos

Diversos sistemas possuem dispositivos semelhantes, portas de E/S ou outros recursos que podem ser organizados de forma lógica como uma array. Para simplificar estes casos, são oferecidos tipos do localizador de objetos da array. Os nomes dos tipos da array do localizador de objetos `_array` são adicionado a eles:

<code>required_device</code>	<code>required_device_array</code>
<code>optional_device</code>	<code>optional_device_array</code>
<code>required_memory_region</code>	<code>required_memory_region_array</code>
<code>optional_memory_region</code>	<code>optional_memory_region_array</code>
<code>required_memory_bank</code>	<code>required_memory_bank_array</code>
<code>optional_memory_bank</code>	<code>optional_memory_bank_array</code>
<code>memory_bank_creator</code>	<code>memory_bank_array_creator</code>
<code>required_ioport</code>	<code>required_ioport_array</code>
<code>optional_ioport</code>	<code>optional_ioport_array</code>
<code>required_address_space</code>	<code>required_address_space_array</code>
<code>optional_address_space</code>	<code>optional_address_space_array</code>
<code>required_region_ptr</code>	<code>required_region_ptr_array</code>
<code>optional_region_ptr</code>	<code>optional_region_ptr_array</code>
<code>required_shared_ptr</code>	<code>required_shared_ptr_array</code>
<code>optional_shared_ptr</code>	<code>optional_shared_ptr_array</code>
<code>memory_share_creator</code>	<code>memory_share_array_creator</code>

Um caso comum para um localizador da array do objeto é a chave da matriz:

```
class keyboard_base : public device_t, public device_mac_keyboard_interface
{
protected:
    keyboard_base(machine_config const &mconfig, device_type type, char const *tag,
↳device_t *owner, u32 clock) :
        device_t(mconfig, type, tag, owner, clock),
        device_mac_keyboard_interface(mconfig, *this),
        m_rows(*this, "ROW%u", 0U)
    {
    }

    u8 bus_r()
    {
        u8 result(0xffU);
        for (unsigned i = 0U; m_rows.size() > i; ++i)
        {
            if (!BIT(m_row_drive, i))
                result &= m_rows[i]->read();
        }
        return result;
    }

    required_ioport_array<10> m_rows;
};
```

Construir um objeto localizador da array é o mesmo que construir um localizador de objetos exceto que em vez de apenas uma etiqueta você fornece uma string com o formato da tag e um offset do índice. Neste caso as etiquetas das portas de E/S no array serão ROW0, ROW1, ROW2, . . . e ROW9. Observe que a matriz

do localizador de objetos aloca o armazenamento de forma dinâmica para as etiquetas para que as mesmas permaneçam válidas até a sua destruição.

O localizador é utilizado da mesma forma que um `std::array` do tipo do localizador de objeto subjacente. Ele suporta a indexação, os iteradores e com base nos intervalos de loop `for`.

Por ter um offset do índice definido, as etiquetas não precisam utilizar os índices com base zero. É comum utilizar a indexação com base 1 como mostra o exemplo abaixo:

```
class dooyong_state : public driver_device
{
protected:
    dooyong_state(machine_config const &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_bg(*this, "bg%u", 1U),
        m_fg(*this, "fg%u", 1U)
    {
    }

    optional_device_array<dooyong_rom_tilemap_device, 2> m_bg;
    optional_device_array<dooyong_rom_tilemap_device, 2> m_fg;
};
```

Isso faz com que `m_bg` encontre os dispositivos com as etiquetas `bg1` e `bg2` enquanto `m_fg` encontra os dispositivos com as etiquetas `fg1` e `fg2`. Observe que os índices nos localizadores ainda tem base zero como qualquer outra array C.

Também é possível que haja outras conversões do formato como hexadecimais (`%x` e `%X`) ou caractere (`%c`):

```
class eurit_state : public driver_device
{
public:
    eurit_state(machine_config const &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_keys(*this, "KEY%c", 'A')
    {
    }

private:
    required_ioport_array<5> m_keys;
};
```

Neste caso as portas da matriz chave usam as etiquetas `KEYA`, `KEYB`, `KEYC`, `KEYD` e `KEYE`.

É possível usar uma lista das etiquetas do inicializador fechado-as entre colchetes quando as etiquetas não seguirem uma sequência ascendente simples:

```
class seabattl_state : public driver_device
{
public:
    seabattl_state(machine_config const &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_digits(*this, { "sc_thousand", "sc_hundred", "sc_half", "sc_unity", "tm_
↪half", "tm_unity" })
    {
    }

private:
    required_device_array<dm9368_device, 6> m_digits;
};
```

Se os localizadores subjacentes dos objetos exigirem argumentos adicionais do construtor, forneça-os após o formato da etiqueta e o deslocamento do índice (os mesmos valores serão usados para todos os elementos da array):

```
class dreamwld_state : public driver_device
{
public:
    dreamwld_state(machine_config const &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_vram(*this, "vram_%u", 0U, 0x2000U, ENDIANNESS_BIG)
    {
    }

private:
    memory_share_array_creator<u16, 2> m_vram;
};
```

Isso localiza ou cria uma memória compartilhada com as etiquetas vram_0 e vram_1, cada uma com 8 KiB organizadas com 4,096 words big-Endian 16-bit.

12.4.6 Localizadores opcionais de objetos

Os localizadores opcionais de objetos não exibem um erro caso o objeto alvo não seja encontrado. Isto é útil em duas situações: implementações `driver_device` (classes do estado) que representam uma família dos sistemas onde alguns componentes não estão presentes em todas as configurações e os dispositivos que podem utilizar um recurso de maneira opcional. Também são fornecidas funções adicionais dos membros para testar se o objeto alvo foi encontrado ou não.

Componentes opcionais do sistema

Muitas vezes uma classe é usada para representar uma família relacionada de sistemas. Caso um componente não esteja presente em todas as configurações, pode ser conveniente usar um localizador opcional para obter acesso a ele. Como exemplo, usaremos o dispositivo Sega X-board:

```
class segaxbd_state : public device_t
{
protected:
    segaxbd_state(machine_config const &mconfig, device_type type, char const *tag,
↳device_t *owner, u32 clock) :
        device_t(mconfig, type, tag, owner, clock),
        m_soundcpu(*this, "soundcpu"),
        m_soundcpu2(*this, "soundcpu2"),
        m_segaic16vid(*this, "segaic16vid"),
        m_pc_0(0),
        m_lastsurv_mux(0),
        m_adc_ports(*this, "ADC%u", 0),
        m_mux_ports(*this, "MUX%u", 0)
    {
    }

    optional_device<z80_device> m_soundcpu;
    optional_device<z80_device> m_soundcpu2;
    required_device<mb3773_device> m_watchdog;
    required_device<segaic16_video_device> m_segaic16vid;
    bool m_adc_reverse[8];
    u8 m_pc_0;
    u8 m_lastsurv_mux;
    optional_ioport_array<8> m_adc_ports;
    optional_ioport_array<4> m_mux_ports;
};
```

Os membros `optional_device` e `optional_ioport_array` são construídos e declarados de maneira comum. Antes de acessar o objeto alvo, chamamos o membro da função `found()` para verificar a sua presença no sistema (o operador "cast-to-Boolean" pode ser utilizado de maneira explícita com a mesma finalidade):

```
void segaxbd_state::pc_0_w(u8 data)
{
    m_pc_0 = data;
    m_watchdog->write_line_ck(BIT(data, 6));
    m_segaic16vid->set_display_enable(data & 0x20);
    if (m_soundcpu.found())
        m_soundcpu->set_input_line(INPUT_LINE_RESET, (data & 0x01) ? CLEAR_LINE :
↳ASSERT_LINE);
    if (m_soundcpu2.found())
        m_soundcpu2->set_input_line(INPUT_LINE_RESET, (data & 0x01) ? CLEAR_LINE :
↳ASSERT_LINE);
}
```

As portas opcionais de E/S oferecem de maneira conveniente a função do membro chamado `read_safe` que lê o valor da porta caso esta esteja presente ou em vez disso retorna o valor padrão:

```
u8 segaxbd_state::analog_r()
{
    int const which = (m_pc_0 >> 2) & 7;
    u8 value = m_adc_ports[which].read_safe(0x10);

    if (m_adc_reverse[which])
        value = 255 - value;

    return value;
}

u8 segaxbd_state::lastsurv_port_r()
{
    return m_mux_ports[m_lastsurv_mux].read_safe(0xff);
}
```

Na ausência, as portas ADC retornam 0x10 (decimal 16) enquanto na ausência das portas digitais multiplexadas retornam 0xff (decimal 255). Observe que o `read_safe` é um membro do próprio `optional_ioport` e não um membro do objeto alvo `ioport_port` (o `optional_ioport` não perde a sua referência durante o uso).

Há algumas desvantagens durante o uso dos localizadores opcionais:

- Não há como distinguir entre o alvo não estar presente ou não ser encontrado por questões de erros nas etiquetas tornando-as mais propensos a erros.
- Verificando caso o alvo esteja presente para poder utilizar os recursos do prognóstico do núcleo da CPU prejudicando potencialmente seu desempenho caso isso aconteça com muita frequência.

Avalie se os localizadores opcionais são a melhor solução ou se seria mais apropriado a criação de uma classe derivada para o sistema com componentes adicionais.

Recursos Opcionais

Alguns dispositivos podem utilizar certos recursos de maneira opcional. O dispositivo ainda funcionará caso o sistema host não os forneça, embora algumas funcionalidades possam não estar disponíveis. Por exemplo, o slot do cartucho do **Virtual Boy** responde em três espaços de endereço chamados EXP, CHIP e ROM. O sistema host jamais utilizará um ou mais deles, não é necessário fornecer um lugar para que o cartucho instale os manipuladores correspondentes. (Por exemplo, uma copiadora só pode utilizar apenas o espaço da ROM).

Vejamos como isso é implementado. O dispositivo de slot do cartucho do **Virtual Boy** declara os membros `optional_address_space` para os três espaços dos endereços, os membros do `offs_t` para os espaços nestes endereços e as funções dos membros em linha para configurá-los:

```
class vboy_cart_slot_device :
    public device_t,
    public device_image_interface,
    public device_single_card_slot_interface<device_vboy_cart_interface>
{
public:
    vboy_cart_slot_device(machine_config const &mconfig, char const *tag, device_t_
    ↪,*owner, u32 clock = 0U);

    template <typename T> void set_exp(T &&tag, int no, offs_t base)
    {
        m_exp_space.set_tag(std::forward<T>(tag), no);
        m_exp_base = base;
    }
    template <typename T> void set_chip(T &&tag, int no, offs_t base)
    {
        m_chip_space.set_tag(std::forward<T>(tag), no);
        m_chip_base = base;
    }
    template <typename T> void set_rom(T &&tag, int no, offs_t base)
    {
        m_rom_space.set_tag(std::forward<T>(tag), no);
        m_rom_base = base;
    }

protected:
    virtual void device_start() override;

private:
    optional_address_space m_exp_space;
    optional_address_space m_chip_space;
    optional_address_space m_rom_space;
    offs_t m_exp_base;
    offs_t m_chip_base;
    offs_t m_rom_base;

    device_vboy_cart_interface *m_cart;
};

DECLARE_DEVICE_TYPE(VBOY_CART_SLOT, vboy_cart_slot_device)
```

Os localizadores de objetos são construídos com valores fictícios para as etiquetas e os números do espaço matemático (finder_base::DUMMY_TAG e -1):

```
vboy_cart_slot_device::vboy_cart_slot_device(machine_config const &mconfig, char const *tag,
↳ device_t *owner, u32 clock) :
    device_t(mconfig, VBOY_CART_SLOT, tag, owner, clock),
    device_image_interface(mconfig, *this),
    device_single_card_slot_interface<device_vboy_cart_interface>(mconfig, *this),
    m_exp_space(*this, finder_base::DUMMY_TAG, -1, 32),
    m_chip_space(*this, finder_base::DUMMY_TAG, -1, 32),
    m_rom_space(*this, finder_base::DUMMY_TAG, -1, 32),
    m_exp_base(0U),
    m_chip_base(0U),
    m_rom_base(0U),
    m_cart(nullptr)
{
}
```

Para ajudar na detecção dos erros de configuração, verificaremos os casos onde os espaços dos endereços foram configurados mas não estão presentes:

```
void vboy_cart_slot_device::device_start()
{
    if (!m_exp_space && ((m_exp_space.finder_tag() != finder_base::DUMMY_TAG) || (m_exp_
↳ space.spacenum() >= 0)))
        throw emu_fatalerror("%s: Address space %d of device %s not found (EXP)\n",
↳ tag(), m_exp_space.spacenum(), m_exp_space.finder_tag());

    if (!m_chip_space && ((m_chip_space.finder_tag() != finder_base::DUMMY_TAG) || (m_
↳ chip_space.spacenum() >= 0)))
        throw emu_fatalerror("%s: Address space %d of device %s not found (CHIP)\n",
↳ tag(), m_chip_space.spacenum(), m_chip_space.finder_tag());

    if (!m_rom_space && ((m_rom_space.finder_tag() != finder_base::DUMMY_TAG) || (m_rom_
↳ space.spacenum() >= 0)))
        throw emu_fatalerror("%s: Address space %d of device %s not found (ROM)\n",
↳ tag(), m_rom_space.spacenum(), m_rom_space.finder_tag());

    m_cart = get_card_device();
}
```

12.4.7 Os tipos dos localizadores de objetos com mais detalhes

Todos os localizadores de objetos oferecem a funcionalidade de configuração:

```
char const *finder_tag() const { return m_tag; }
std::pair<device_t &, char const *> finder_target();
void set_tag(device_t &base, char const *tag);
void set_tag(char const *tag);
void set_tag(finder_base const &finder);
```

Os membros das funções `finder_tag` e `finder_target` oferecem acesso ao alvo que está sendo configurado no momento. Observe que a tag retornada por `finder` é relativa a base do dispositivo e por si só não é suficiente para identificar o alvo.

As funções do membro `set_tag` fazem a configuração do alvo do localizador de objetos. Estes membros não devem ser invocados depois que o localizador de objetos seja resolvido. O primeiro formulário configura a base do dispositivo e a etiqueta relativa a ele. Já o segundo formulário configura a etiqueta relativa como também configura de forma implícita a base do dispositivo que atualmente está sendo configurado, este formulário só deve ser invocado a partir das funções de configuração do sistema. O terceiro formulário configura a base do objeto base e a etiqueta relacionada para o alvo atual de um outro localizador de objetos.

Observe que a função do membro `set_tag` **não** copia a etiqueta relacionada ao objeto. É responsabilidade de quem invoca assegurar que a string C permaneça válida até que o localizador do objeto seja resolvido (ou reconfigurado com uma etiqueta diferente). No momento da resolução a base do dispositivo também deve ser válido. Este pode não ser o caso se posteriormente o dispositivo puder ser removido ou substituído.

Todos os localizadores de objetos oferecem a mesma interface para acessar o objeto alvo:

```
ObjectClass *target() const;
operator ObjectClass *() const;
ObjectClass *operator->() const;
```

Todos estes membros dão acesso ao objeto-alvo. Caso o alvo não tenha sido encontrado, a função do membro `target` e o operador "cast-to-pointer" retornará `nullptr`. O operador do membro de acesso do ponteiro garante que o alvo tenha sido encontrado.

Os localizadores de objetos opcionais fornecem aos membros de maneira adicional os testes para saber se o objeto-alvo tenha foi encontrado:

```
bool found() const;
explicit operator bool() const;
```

Estes membros retornam `true` caso o alvo tenha sido encontrado assumindo que o ponteiro do alvo não seja nulo no momento da sua localização.

Os localizadores dos dispositivos

Os localizadores dos dispositivos exigem um modelo de argumento para a classe prevista do dispositivo. Isto deve ser proveniente a partir do `device_t` ou do `device_interface`. O objeto do dispositivo alvo deve ser entre uma instância desta classe ou de uma instância de uma classe que se derive dela. Uma mensagem de aviso é registrada caso um dispositivo correspondente seja encontrado, porém mas não é uma instância da classe prevista.

Os localizadores dos dispositivos oferecem uma sobrecarga `set_tag` adicional:

```
set_tag(DeviceClass &object);
```

Seria o mesmo que invocar `set_tag(object, DEVICE_SELF)`. Observe que o objeto do dispositivo não deve ser removido ou substituído antes que o localizador do objeto seja resolvido.

O sistema de memória dos localizadores de objetos

Os localizadores dos objetos do sistema de memória, `required_memory_region`, `optional_memory_region`, `required_memory_bank`, `optional_memory_bank` e o `memory_bank_creator`, não possuem qualquer funcionalidade especial. São frequentemente utilizados no lugar das etiquetas literais durante a instalação dos bancos da memória no espaço dos endereços.

Um exemplo do uso do localizador do banco de memória num endereço no mapa:

```
class qvt70_state : public driver_device
{
public:
    qvt70_state(machine_config &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_rombank(*this, "rom"),
        m_rambank(*this, "ram%d", 0U),
    { }

private:
    required_memory_bank m_rombank;
    required_memory_bank_array<2> m_rambank;

    void mem_map(address_map &map);

    void rombank_w(u8 data);
};

void qvt70_state::mem_map(address_map &map)
{
    map(0x0000, 0x7fff).bankr(m_rombank);
    map(0x8000, 0x8000).w(FUNC(qvt70_state::rombank_w));
    map(0xa000, 0xbfff).ram();
    map(0xc000, 0xdfff).bankrw(m_rambank[0]);
    map(0xe000, 0xffff).bankrw(m_rambank[1]);
}
```

Um exemplo de um criador do banco da memória para instalação dinâmica:

```
class vegaeo_state : public eolith_state
{
public:
    vegaeo_state(machine_config const &mconfig, device_type type, char const *tag) :
        eolith_state(mconfig, type, tag),
        m_qs1000_bank(*this, "qs1000_bank")
    {
    }

    void init_vegaeo();

private:
    memory_bank_creator m_qs1000_bank;
};

void vegaeo_state::init_vegaeo()
{
    // Set up the QS1000 program ROM banking, taking care not to overlap the internal RAM
    m_qs1000->cpu().space(AS_IO).install_read_bank(0x0100, 0xffff, m_qs1000_bank);
    m_qs1000_bank->configure_entries(0, 8, memregion("qs1000:cpu")->base() + 0x100,
    0x10000);

    init_speedup();
}
```

O localizador das portas E/S

Um localizador opcional de portas E/S para fornecer uma função adicional e conveniente ao membro:

```
ioport_value read_safe(ioport_value defval);
```

Isso vai ler o valor da porta caso o valor alvo da porta E/S tenha sido encontrado ou então retorne defval. É útil em situações onde certos dispositivos de entrada não estejam sempre presentes.

Os localizadores dos espaços nos endereços

Os localizadores de espaço nos endereços aceitam um argumento adicional para o número do espaço do endereço que será localizado. Pode ser que uma largura de dados possa ser fornecida de forma opcional ao construtor.

```
address_space_finder(device_t &base, char const *tag, int spacenum, u8 width = 0);
void set_tag(device_t &base, char const *tag, int spacenum);
void set_tag(char const *tag, int spacenum);
void set_tag(finder_base const &finder, int spacenum);
template <bool R> void set_tag(address_space_finder<R> const &finder);
```

A base do dispositivo e sua etiqueta devem identificar um dispositivo que implemente `device_memory_interface`. O número do espaço do endereço é um índice baseado em zero para um dos espaços nos endereços do dispositivo.

Caso a largura não seja zero ela deve corresponder à largura dos dados do espaço do endereço do alvo em bits. Caso o espaço de endereço exista no destino mas tenha uma largura de dados diferente, uma mensagem de aviso será registrada e será tratada como não tendo sido encontrada. No caso da largura ser zero (o valor predefinido do argumento), não haverá verificação na largura dos dados no espaço dos endereços do alvo.

As funções dos membros também são fornecidas visando obter o número do espaço dos endereços configurados e definir a largura necessária dos dados:

```
int spacenum() const;
void set_data_width(u8 width);
```

Os localizadores dos ponteiros da memória

Todos os localizadores `required_region_ptr`, `optional_region_ptr`, `required_shared_ptr`, `optional_shared_ptr` e `memory_share_creator` do ponteiro da memória precisam de um argumento modelo para o tipo do elemento da região da memória. Este normalmente deve ser um tipo explicitamente inteiro não assinado (`u8`, `u16`, `u32` or `u64`). O tamanho deste tipo é comparado com a largura da região da memória. Caso não corresponda, uma mensagem de aviso é registrada e a região ou o compartilhamento é tratado como não encontrado.

Os localizadores do ponteiro da memória providenciam um operador de acesso ao array e aos membros para que tenham acesso ao tamanho da região da memória:

```
PointerType &operator[](int index) const;
size_t length() const;
size_t bytes() const;
```

O operador de acesso ao array retorna uma referência `const` \ `non-const` para um elemento da região da memória. O índice está em unidades do tipo elemento; deve ser positivo e menor que o comprimento da região da memória. O membro `length` retorna a quantidade dos elementos na região da memória. O membro `bytes` retorna o tamanho da região da memória em bytes. Estes membros não devem ser invocados caso a região ou compartilhamento alvo não tenha sido encontrada.

O `memory_share_creator` necessita de argumentos adicionais do construtor para o tamanho e a extremidade¹ do compartilhamento da memória:

```
memory_share_creator(device_t &base, char const *tag, size_t bytes, endianness_t endianness);
```

O tamanho é definido em bytes. Caso um compartilhamento da memória seja encontrado, será considerado como um erro caso seu tamanho não corresponda ao tamanho especificado. No caso da largura ser maior que 8 bits e seja encontrado um compartilhamento existente na memória, será considerado um erro caso o seu *Endianness* não corresponda ao *Endianness* definido.

O `memory_share_creator` proporciona membros adicionais para acessar as propriedades do compartilhamento da memória:

```
endianness_t endianness() const;
u8 bitwidth() const;
u8 bytewidth() const;
```

Estes membros retornam o *Endianness*, a largura em bits e a largura em bytes da parte da memória respectivamente. Eles não devem ser invocados caso o compartilhamento da memória não tenha sido encontrado.

¹ Propriedade daquilo que é ENDIAN, extremidade (ordenação)

12.4.8 Os localizadores das saídas

Os localizadores das saídas são usados para expor as saídas que podem ser aproveitadas pelo sistema de arte-final ou por programas externos. Uma aplicação comum que utilize um programa externo é um painel de controle ou controlador da iluminação do gabinete.

Os localizadores da saída não são realmente localizadores de objetos porém eles são descritos aqui porque são usados de maneira similar. Há uma série de diferenças importantes a se considerar:

- Os localizadores das saídas sempre criam saídas caso elas não existam.
- Os localizadores devem ser resolvidos de forma manual, eles não são automaticamente resolvidos.
- Os localizadores não podem ter o seu alvo alterado depois da construção.
- Os localizadores são do tipo array e suportam uma quantidade indeterminado de dimensões.
- Os nomes das saídas são globais, não há influência do dispositivo de base (Isto irá mudar no futuro).

Os localizadores da saída aceitam uma quantidade variável de argumentos do modelo correspondente a quantidade das dimensões que você quiser da array. Vejamos um exemplo que utiliza os localizadores zero-, unidimensionais e bidimensionais:

```
class mmd2_state : public driver_device
{
public:
    mmd2_state(machine_config const &mconfig, device_type type, char const *tag) :
        driver_device(mconfig, type, tag),
        m_digits(*this, "digit%u", 0U),
        m_p(*this, "p%u_%u", 0U, 0U),
        m_led_halt(*this, "led_halt"),
        m_led_hold(*this, "led_hold")
    { }

protected:
    virtual void machine_start() override;

private:
    void round_leds_w(off_t, u8);
    void digit_w(u8 data);
    void status_callback(u8 data);

    u8 m_digit;

    output_finder<9> m_digits;
    output_finder<3, 8> m_p;
    output_finder<> m_led_halt;
    output_finder<> m_led_hold;
};
```

Os membros `m_led_halt` e `m_led_hold` são localizadores com saída zero-dimensional. Eles encontram uma única saída cada um. O membro `m_digits` é um localizador unidimensional, ele encontra nove saídas organizadas como uma array unidimensional. O membro `m_p` é um localizador bidimensional, ele encontra 24 saídas organizadas em três filas de 8 colunas cada. Uma quantidade maior de dimensões são suportadas.

O construtor do localizador obtém uma referência do dispositivo de base, um formato string e um deslocamento do índice para cada dimensão. Neste caso todos os deslocamentos são zero. O localizador unidimensional `m_digits` encontrará as saídas `digit0`, `digit1`, `digit2`, ... `digit8`. O localizador bidimensional `m_p`

encontrará as saídas p0_0, p0_1, ... p0_7 para a primeira linha, p1_0, p1_1, ... p1_7 para a segunda e p2_0, p2_1, ... p2_7 para a terceira.

Você deve solicitar o resolve em cada localizador antes de serem utilizados. Isto deve ser feito na inicialização para que os valores da saída sejam incluídos nos estados de salvamento:

```
void mmd2_state::machine_start()
{
    m_digits.resolve();
    m_p.resolve();
    m_led_halt.resolve();
    m_led_hold.resolve();

    save_item(NAME(m_digit));
}
```

Os localizadores da saída proporcionam operadores que permitem que eles sejam designados ou fundidos aos inteiros assinados com 32 bits. O operador da atribuição enviará uma notificação caso o novo valor seja diferente do valor da saída atual.

```
operator s32() const;
s32 operator=(s32 value);
```

Para definir os valores da saída, atribua através dos localizadores da mesma maneira que seria feito com uma array do mesmo nível:

```
void mmd2_state::round_leds_w(offst offset, u8 data)
{
    for (u8 i = 0; i < 8; i++)
        m_p[offset][i] = BIT(~data, i);
}

void mmd2_state::digit_w(u8 data)
{
    if (m_digit < 9)
        m_digits[m_digit] = data;
}

void mmd2_state::status_callback(u8 data)
{
    m_led_halt = (~data & i8080_cpu_device::STATUS_HLTA) ? 1 : 0;
    m_led_hold = (data & i8080_cpu_device::STATUS_WO) ? 1 : 0;
}
```

12.5 Entrada do sistema

- *Introdução*
- *Componentes*
 - *Entrada do dispositivo*
 - *Item da entrada do dispositivo*
 - *Faixa da porta de E/S*
 - *Gerenciamento da entrada*
 - *Gerenciador da porta de E/S*
- *Estruturas e os tipos dos dados*
 - *Código de entrada*
 - *Sequência de entrada*
- *Módulos do gerenciador de entrada*
- *Posições do jogador*
- *Atualizando as faixas de E/S da porta*
 - *Atualizando as faixas digitais*
 - *Atualizando as faixas analógicas absolutas*
 - *Atualizando as faixas analógicas relativas*

12.5.1 Introdução

A variedade dos sistemas que o MAME emula, bem como a variação da hospedagem dos sistemas e dos periféricos, exige um sistema de entrada flexível e configurável.

Observe que o sistema da entrada tem a preocupação com a entrada de baixo nível do usuário. A interação de alto nível do usuário, envolvendo itens como a entrada de texto e dos dispositivos apontadores, é tratada de forma separada.

12.5.2 Componentes

Do ponto de vista do sistema que está sendo emulado, o conceito da entrada do sistema tem os seguintes componentes.

Entrada do dispositivo

A entrada dos dispositivos disponibilizam informações sobre os valores da entrada. A entrada geralmente corresponde a um dispositivo físico hospedado pelo sistema, por exemplo, um teclado, um mouse ou um controle de jogo. Entretanto, nem sempre há uma correspondência um-a-um entre as entradas dos dispositivos e os dispositivos físicos. Por exemplo, o módulo que provê o teclado SDL, agrega todos os teclados numa única entrada do dispositivo, já o módulo que provê o *lightgun Win32*, pode apresentar dois dispositivos usando a entrada de um único mouse.

A entrada dos dispositivos são identificados através da sua classe (teclado, mouse, joystick ou revólver) e pela quantidade dos dispositivos dentro da sua classe. Os provedores dos módulos de entrada também podem fornecer um identificador dependente da implementação visando permitir que o usuário possa configurar um valor numérico estável do dispositivo.

Observe que a entrada dos dispositivos não têm relação com os dispositivos emulados (implementações do tipo `device_t`) apesar da terem uma nomenclatura semelhante.

Item da entrada do dispositivo

Também conhecido como **controle**, o item da entrada do dispositivo corresponde a uma fonte de entrada que gera um valor único. Isso geralmente corresponde a um controle físico ou a um sensor, por exemplo, um eixo do joystick, um botão ou um acelerômetro.

O MAME suporta três tipos de controles: **interruptores**, **eixos absolutos** e **eixos relativos**:

- Os interruptores geram um valor 0 quando estão inativos (liberados ou desligados).
- Os eixos absolutos produzem um valor normalizado na faixa entre -65.536 até 65.536 com o 0 correspondendo à posição neutra.
- Os eixos relativos produzem um valor correspondente ao movimento a partir da atualização da entrada anterior. Os dispositivos do tipo mouse, escalam os valores nominais para aproximadamente 512 com 100 DPI por pixel.

Os valores negativos dos eixos devem corresponder as direções para cima, para a esquerda, distante do jogador ou no sentido anti-horário. Para eixos com uma extremidade (pedais, gatilhos e botões sensíveis ao deslocamento por exemplo), apenas o 0 e a parte negativa da faixa deve ser usada.

Os interruptores são usados para representar os controles que naturalmente têm dois estados bem distintos, como botões e os interruptores de comutação.

Os eixos absolutos são usados para representar controles com uma faixa definida ou uma posição neutra. Os exemplos incluem volantes com final de curso, eixos de joystick e gatilhos sensíveis ao deslocamento.

Os eixos relativos são usados para representar controles com um alcance efetivamente infinito. Os exemplos incluem os eixos do mouse, de um *trackball*, mostradores de codificação incremental e giroscópios.

Os acelerômetros e os eixos de detecção de força do joystick devem ser representados como eixos absolutos, ainda que o alcance seja teoricamente aberto. Na prática, há um limite para o alcance que os transdutores podem relatar, geralmente é substancialmente maior do que o necessário para uma operação normal.

Os itens da entrada do dispositivo são identificados pela classe e pelo número associado ao dispositivo, juntamente com uma **identificação da entrada do item**. O MAME fornece IDs para os tipos de itens comuns dos controles. Os controles adicionais ou os controles que não correspondam a um tipo comum, as IDs são dinamicamente atribuídas para eles. O MAME suporta centenas de itens por entrada de dispositivo.

Faixa da porta de E/S

Uma faixa da porta de E/S representa uma fonte de entrada num dispositivo ou num sistema emulado. A maioria dos tipos das faixas da porta de E/S pode ser atribuída a uma ou mais combinações de controles, permitindo ao usuário controlar a entrada para o sistema que está sendo emulado.

Da mesma forma que os itens da entrada do dispositivo, existem diversos tipos de faixas da porta de E/S:

- **As faixas digitais:**

Funcionam como interruptores que produzem um de dois valores distintos. Eles são usados para as teclas de um teclado, interruptores de direção do joystick com oito direções, comutadores, foto interruptores e outras entradas emuladas que funcionam como interruptores com duas posições.

- **As faixas analógicas absolutas:**

Têm uma faixa com posições mínimas, máximas e neutras definidas. São usadas para os eixos de um joystick analógico, pedais sensíveis ao deslocamento, comandos rotativos e outras entradas emuladas com uma faixa definida.

- **As faixas analógicas relativas:**

Têm uma faixa inicial com posições mínimas, máximas definidas. Em cada atualização, o valor se acumula e se enrola quando passa por qualquer uma das extremidades do intervalo. Funcionalmente, isto é como a saída de um contador acima/abaixo conectado a um codificador incremental. Eles são usados para eixos do mouse/trackball, volantes sem limites de parada (fim de curso) e outras entradas emuladas que não têm limites de alcance.

- **A chave DIP:**

A configuração e as faixas de ajuste permitem que o usuário defina um valor através da interface de usuário do MAME.

- **Outros tipos especiais da faixa:**

São usados para produzir valores fixos ou gerados de forma programática.

Uma faixa digital é exibido para o usuário como uma única entrada atribuível, que aceita valores de comutação.

Uma faixa analógica é exibida ao usuário como três entradas atribuíveis: uma **entrada do eixo**, que aceita valores do eixo; uma **entrada de incremento** e uma **entrada de decremento**, que aceita valores de comutação.

Gerenciamento da entrada

O gerenciador da entrada possui diversas responsabilidades, que incluem:

- Fazer o rastreamento da entrada nos dispositivos disponíveis no sistema.
- Ler o valor das entradas.
- Fazer a conversão entre os valores dos identificadores internos, das sequências dos símbolos de configuração e das sequências de exibição.

Na prática, os dispositivos e os sistemas emulados raramente interagem diretamente com o gerenciador de entrada. A razão mais comum para acessar o gerenciador é implementar controles especiais de depuração que devem ser desativados nas versões de lançamento. Os plug-ins que respondem à necessidade de entrada, precisam invocar o gerenciador para fazer a leitura das entradas.

Gerenciador da porta de E/S

O gerenciador de E/S possui diversas responsabilidades, que incluem:

- Fazer o gerenciamento das atribuições das faixas dos controles da porta de E/S e as ações da interface com o usuário.
- Fazer Leitura dos valores da entrada através do gerenciador e fazer a atualização dos valores da faixa das portas de E/S.

Assim como o gerenciador da entrada, o gerenciador das portas de E/S é amplamente transparente com os dispositivos e com os sistemas emulados. Só é preciso configurar as suas portas e as faixas de E/S para que o gerenciador cuide do resto.

12.5.3 Estruturas e os tipos dos dados

Os seguintes tipos de dados são usados para lidar com a entrada.

Código de entrada

Um código de entrada determina um item de entrada do dispositivo e como ele deve ser interpretado. É um **énuplo**¹ que consiste basicamente nos seguintes valores, classe do dispositivo (**device class**), número do dispositivo (**device number**), classe do item (**item class**), modificador do item (**item modifier**) e a ID do item (**item ID**):

- **A classe do dispositivo:**

O valor numérico do dispositivo junto com a ID do item identificam o item do dispositivo que será lido na entrada.

- **A classe do item:**

Determina o tipo desejado do valor a ser gerado de um interruptor, de um eixo absoluto ou de um eixo relativo. Os valores dos eixos podem ser convertidos para valores de comutação ao definir um modificador adequado.

- **O modificador:**

Determina como um valor deve ser interpretado. As opções válidas dependem do tipo do item do dispositivo na entrada e da classe definida do item.

Caso o item da entrada especificada seja um interruptor, ele só poderá ser lido usando a classe do interruptor e nenhum outro modificador será suportado. O item sempre retornará 0 caso haja a tentativa de ler um comutador como um eixo absoluto ou como um eixo relativo.

No caso de ser especificado como um eixo absoluto, ele poderá ser lido como um eixo absoluto ou como um comutador:

- A leitura de um item com um eixo absoluto retorna para o seu estado atual de controle, potencialmente se transformado num modificador caso um seja especificado. Os modificadores suportados são **invertidos** para inverter a faixa de alcance do controle, **positivo** para fazer o mapeamento da faixa de alcance positivo do controle na saída (o 0 corresponde a -65.536 e 65.536 corresponde a 65.536) e **negativo** para mapear a faixa do alcance negativo do controle na saída (o 0 corresponde a -65.536 e -65.536 corresponde a 65.536).
- A leitura de um item com um eixo absoluto como um interruptor, retorna 0 ou 1 dependendo caso o controle passe de um limite na direção determinada pelo modificador. Use o modificador **negativo** para retornar 1 quando o controle estiver além do limite na direção negativa (para cima ou para a esquerda) ou um modificador **positivo** para retornar 1 quando o controle estiver além do limite na direção positiva (para baixo ou para a direita). Há dois pares especiais de modificadores, **esquerda/direita** e **cima/baixo** que são aplicáveis somente aos eixos primários X/Y dos dispositivos de joystick. O usuário pode definir um mapa para o joystick para controlar como estes modificadores interpretam a movimentação do joystick.
- O valor sempre retornará 0 ao tentar ler um item com eixo absoluto como se fosse um eixo relativo.

¹ Tuple do Inglês, também conhecido como tupla, é uma estrutura de dados ordenada por elementos.
<https://www.wikiwand.com/pt/Énuplo>

Caso o item informado da entrada seja um eixo relativo, ele pode ser lido como um eixo relativo ou como um interruptor:

- A leitura de um item com um eixo relativo como um eixo relativo, retorna a alteração do valor desde a última atualização da entrada. O único modificador suportado é o **inverso**, que negativa o valor, revertendo a direção.
- A leitura de um eixo relativo como sendo um interruptor, retorna 1 caso o controle se mova na direção determinada pelo modificador da entrada desde a sua última atualização. Use os modificadores **negativo/esquerda/cima** para retornar 1 quando o controle tiver sido movido na direção negativa (para cima ou para a esquerda) ou use os modificadores **positivo/direita/baixo** para retornar 1 quando o controle tiver se movido na direção positiva (para baixo ou para a direita).
- A tentativa de fazer a leitura de um item com eixo relativo como se fosse um eixo absoluto, sempre retorna 0.

Há também os códigos especiais da entrada que são usados para especificar como os vários controles devem ser combinados na entrada de uma sequência.

O lugar mais comum que será possível encontrar os códigos de entrada no dispositivo e no código do driver do sistema é ao especificar atribuições iniciais para as faixas da porta de E/S que não possuam atribuições padrão fornecidas pelo seu núcleo. A macro `PORT_CODE` é utilizada para esta finalidade.

O MAME oferece macros e funções auxiliares para produzir os códigos de entrada mais usados, incluindo as teclas padrão do teclado, os eixos e os botões do mouse/joystick/arma de luz.

Sequência de entrada

Uma sequência de entrada determina uma combinação dos controles que podem ser atribuídos a uma entrada. O nome diz respeito ao fato de ser implementado como um recipiente desta sequência com os códigos de entrada como elementos. É um tanto enganador, pois, as sequências de entrada são interpretadas utilizando valores instantâneos de controle. As sequências de entrada são interpretadas de maneira diferente para a entrada dos comutadores (*switches*) e dos eixos.

A entrada das sequências vindas de comutadores devem conter apenas os códigos de entrada com a classe do item sendo definida para chavear juntamente com os códigos especiais de entrada `or` e `not`. A sequência é interpretada usando a lógica de soma dos produtos. Um código `not` faz com que o valor retornado seja invertido imediatamente. A conjunção dos valores retornados pelos sucessivos códigos é avaliado até que um código `or` seja encontrado. Caso o valor atual for 1 quando um código `or` for encontrado, ele retorna, caso contrário a avaliação prossegue.

Sequências de entrada nos eixos podem conter códigos de entrada com a classe do item definido para comutar um eixo absoluto ou um eixo relativo juntamente com os códigos especiais `or` e `not`. É útil pensar na sequência de entrada como contendo um ou mais grupos com códigos de entrada separados por códigos `or`:

- Um código `not` faz com que haja uma inversão imediata no valor seguinte do código retornado pela comutação. Ele não tem efeito sobre os códigos absolutos ou sobre os códigos relativos do eixo.
- Dentro de um grupo, é avaliado a conjunção dos valores retornados pelos códigos de comutação. O grupo é ignorado quando o valor for zero.
- Dentro de um grupo, são somados diversos valores do eixo que sejam o mesmo tipo. São somados os valores dos códigos retornados pelo eixo absoluto, assim como, pelos valores do eixo relativo.
- Caso qualquer código de eixo absoluto num grupo retornar um valor diferente de zero, a soma dos eixos relativos no grupo será ignorada. Qualquer valor absoluto do eixo diferente de zero, tem precedência sobre os valores relativos do mesmo.

- A mesma lógica é aplicada ao combinar os valores de um grupo: os valores do grupo produzidos a partir de um eixo do mesmo tipo são somados, já os valores produzidos a partir dos eixos absolutos têm precedência sobre os valores produzidos a partir dos eixos relativos.
- Após a soma dos valores do grupo, caso o valor seja produzido a partir dos eixos absolutos, ele é fixado no intervalo entre -65.536 até 65.536 (os valores produzidos a partir de eixos relativos não são fixados).

O código de emulação raramente precisa lidar diretamente com as sequências de entrada, pois elas são tratadas internamente entre o gerenciador das portas de E/S e o gerenciador das entradas. Este gerenciador também converte as sequências da entrada a partir de e para o token das sequências armazenadas nos arquivos de configuração e gera um texto para exibir a entrada das sequências para usuários.

Para permitir a configuração, os plug-ins com controles ou com as teclas de atalho precisam usar sequências de entrada. As classes utilitárias são fornecidas para permitir que as sequências sejam inseridas pelo usuário de forma consistente, assim como, o gerenciador da entrada possa ser usado para realizar conversões de e para a configuração e a exibição das sequências dos caracteres. É muito raro haver a necessidade de manipular diretamente estas sequências.

12.5.4 Módulos do gerenciador de entrada

Os módulos do gerenciador de entrada faz parte da camada dependente do SO (OSD) e não são expostos diretamente à emulação, ou, ao código da interface do usuário. Os módulos são responsáveis por detectar os dispositivos disponíveis de entrada no host, para configurar os dispositivos de entrada para o gerenciador e fornecer chamadas de retorno (*callbacks*) para ler o estado atual dos itens do dispositivo de entrada. Estes módulos também podem fornecer atribuições predefinidas adicionais de entrada que sejam adequadas para os dispositivos presentes no host.

O usuário pode escolher quais os módulos de entrada pode ser utilizados. Um módulo que gerencie a entrada é utilizado para cada uma das quatro classes de dispositivos de entrada (teclado, mouse, joystick e lightgun). Os módulos disponíveis dependem do sistema operacional do host e da implementação do OSD. Diferentes módulos podem usar *APIs* diferentes, podem se compatíveis com diferentes tipos de dispositivos ou apresentar dispositivos de maneiras diferentes.

12.5.5 Posições do jogador

O MAME usa um conceito chamado *posições do jogador* para ajudar a gerenciar as atribuições da entrada. A quantidade de posições dos jogadores suportados depende do tipo da faixa da porta de E/S:

- Dez posições dos jogadores são suportadas para as entradas comuns do jogo, incluindo o joystick, o controlador, o remo, o mostrador, o trackball, o lightgun e o mouse.
- Quatro posições dos jogadores são suportadas para as entradas para mahjong e para hanafuda.
- Uma posição do jogador é compatível com a entrada usadas pelos sistemas de apostas.
- Outras entradas não utilizam as posições dos jogadores. Isto inclui slots para moedas, botões de início do jogo de arcade, interruptores de inclinação, interruptores de serviço e teclas do teclado/keypad.

O usuário pode configurar as atribuições predefinidas da entrada por posição do jogador para os tipos da porta de E/S compatíveis que são salvos no arquivo `default.cfg`. Estas atribuições são usadas para todos os sistemas, a menos que o driver do dispositivo/sistema forneça as suas próprias atribuições predefinidas ou que o usuário faça a configuração das atribuições da entrada para este sistema em específico.

Visando facilitar o desenvolvimento dos dispositivos de entrada reutilizáveis da emulação, particularmente dispositivos de slot, o gerenciador da porta de E/S renumera automaticamente as posições dos jogadores ao configurar o sistema que está sendo emulado no momento:

- O gerenciador da porta de E/S inicia na posição 1 do jogador e começa a iterar a árvore da emulação do dispositivo em profundidade na primeira ordem e a partir do dispositivo raiz.
- Se um dispositivo tem faixas de E/S na porta que sejam compatíveis com as posições do jogador, eles são renumerados para iniciar a partir da posição atual do gerenciador da porta de E/S do jogador.
- Antes de avançar para o próximo dispositivo, o gerenciador define a posição atual do jogador para a sua última posição mais uma.

Para um exemplo simples, considere o que acontece quando você administra um console Mega Drive da Sega com dois controles conectados:

- O gerenciador da porta de E/S começa na posição 1 do dispositivo raiz do jogador.
- O primeiro dispositivo encontrado na porta de E/S que sejam compatíveis as posições dos jogadores é o primeiro controle. As entradas são renumeradas para começar na posição 1 do jogador. Isto não tem efeito visível, pois as faixas da porta de E/S são inicialmente numerados a partir da posição 1 do jogador.
- Antes de passar para o próximo dispositivo, o gerenciador da porta de E/S define a sua posição atual do jogador para 2 (a última posição do jogador mais um).
- O próximo dispositivo encontrado com as faixas da porta de E/S que sejam compatíveis com as posições dos jogadores é o segundo controle. As entradas são renumeradas para começar na posição 2 do jogador. Isto evita conflitos do tipo da faixa da porta de E/S com o primeiro controle.
- Antes de passar para o próximo dispositivo, o gerenciador da porta de E/S define a sua posição atual do jogador para 3 (a última posição de jogador mais um).
- Não são mais encontrados dispositivos na porta de E/S que suportem as posições dos jogadores.

12.5.6 Atualizando as faixas de E/S da porta

O gerenciador da porta de E/S atualiza as faixas da porta de E/S uma vez para cada quadro de vídeo produzido pela primeira tela emulada do sistema. A maneira como uma faixa é atualizada depende se fica numa faixa digital ou numa faixa analógica.

Atualizando as faixas digitais

A atualização das faixas das portas digitais de E/S é simples:

- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída da sequência na entrada (através do gerenciador da entrada).
- Se o valor for zero, um valor padrão é definido.
- Caso o valor seja diferente de zero, o complemento binário do valor da faixa padrão será definida.

Atualizando as faixas analógicas absolutas

A atualização da faixa absoluta de E/S das portas analógicas são mais complexas devido à necessidade de ser compatível com uma variedade de configurações para os controles:

- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída ao eixo da sequência na entrada (através do gerenciador da entrada).
- Caso o valor atual mude desde a sua última atualização e o item do dispositivo da entrada que produziu o valor atual tenha sido um eixo absoluto, o valor da faixa é definido como o valor atual dimensionado para o intervalo correto e nenhum processamento adicional é executado.
- Quando este valor for diferente de zero e o item do dispositivo da entrada que produziu o valor atual for um eixo relativo, o valor atual será adicionado ao valor do intervalo, dimensionado pela configuração da sensibilidade da faixa.
- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída ao aumento sequencial da entrada (através do gerenciador da entrada); caso este valor seja diferente de zero, o valor da configuração de sensibilidade da velocidade de aumento/redução da faixa será adicionado e dimensionado através da configuração de sensibilidade.
- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída a redução sequencial da entrada (através do gerenciador da entrada); caso este valor seja diferente de zero, o valor da configuração de sensibilidade da velocidade de aumento/redução da faixa será reduzido e dimensionado através da configuração de sensibilidade.
- Se a entrada do eixo atual e todos os outros valores forem zero, porém, um ou ambos os valores da entrada aumentarem e a redução da entrada forem diferentes de zero na última vez que o valor faixa tenha sido alterado em resposta à entrada do usuário, a centralização automática do valor da velocidade da faixa é adicionado a configuração ou subtraído do seu valor para movê-lo para o seu valor predefinido.

Observe que o valor de sensibilidade da configuração para as faixas analógicas absolutas, afeta a resposta aos itens do dispositivo de entrada com eixo relativo e as entradas com aumento/redução, mas não afeta a resposta aos itens de dispositivo da entrada com eixo absoluto ou com a velocidade de centralização automática.

Atualizando as faixas analógicas relativas

As faixas relativas de E/S da porta analógica também precisam de um tratamento especial para atender as várias configurações do controle, mas são um pouco mais simples do que as faixas analógicas absolutas:

- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída ao eixo da sequência na entrada (através do gerenciador da entrada).
- Quando o valor atual for diferente de zero e o item do dispositivo da entrada que produziu o valor atual tenha sido um eixo absoluto, o valor atual será adicionado ao valor da faixa, sendo dimensionado pela configuração da sensibilidade e nenhum processamento adicional será executado.
- Quando o valor atual for diferente de zero e o item do dispositivo da entrada que produziu o valor atual for um eixo relativo, o valor atual será adicionado ao valor da faixa e dimensionado pela configuração de sensibilidade da faixa.
- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída ao aumento sequencial da entrada (através do gerenciador da entrada); caso este valor seja diferente de zero, o valor da configuração de sensibilidade da velocidade de aumento/redução da faixa será adicionado e dimensionado através da configuração de sensibilidade.
- O gerenciador da porta de E/S lê o valor atual para a faixa atribuída a redução sequencial da entrada (através do gerenciador da entrada); caso este valor seja diferente de zero, o valor da configuração de sensibilidade da velocidade de aumento/redução da faixa será reduzido e dimensionado através da configuração de sensibilidade.

Observe que o valor de configuração da sensibilidade para as faixas analógicas relativas afeta a resposta a todas as entradas do usuário.

12.6 O dispositivo da interface de memória

- *Das capacidades*
- *Configuração*
- *Associando os mapas aos espaços*
- *Acessando os espaços*
- *O suporte MMU para o desmontador*

12.6.1 Das capacidades

A interface do dispositivo da memória fornece aos dispositivos a capacidade de criar o mapeamento nos espaços dos endereços onde estes possam ser associados. É usado por qualquer dispositivo que forneça um endereço/barramento de dados (lógico) para que os outros dispositivos possam se conectar nele. É em essência, mas não apenas, às CPUs.

A interface permite um conjunto ilimitado nos espaços de endereços, numerados apenas com pequenos valores positivos. Os índices dos vetores das IDs devem permanecer pequenos visando manter uma pesquisa rápida. Os espaços numerados entre 0-3 possui o nome de uma constante associado e ele:

ID	Nome
0	AS_PROGRAM
1	AS_DATA
2	AS_IO
3	AS_OPCODES

Os espaços 0 e 3 como o AS_PROGRAM e o AS_OPCODE são especiais para o depurador e algumas CPU's por exemplo. o AS_PROGRAM é usado pelo depurador e CPUs como um espaço de onde a CPU lê as suas instruções para o desmontador. O AS_OPCODE quando está presente é utilizado pelo depurador e algumas CPUs para ler parte da instrução do 'opcode', isso significa que o 'opcode' é dependente do dispositivo. Para o Z80 por exemplo, o byte inicial é lido junto com o sinal declarado do M1 enquanto que para o 68000 significa cada instrução 'word' mais os acessos relativos ao PC. O principal, mas não o único uso do AS_OPCODE, serve para implementar a descriptografia de instruções através de um hardware de forma separada dos dados.

12.6.2 Configuração

```
std::vector<std::pair<int, const address_space_config \*>>\ **memory_space_config**\ *(int,
↳ spacenum) const*
```

O dispositivo deve sobrescrever esse método fornecendo um vetor de pares compreendendo um espaço numerado e seu descritor de configuração associado address_space_config. Alguns exemplos para pesquisar quando precisar:

- Vetor padrão two-space: `v60_device`
- Condicional AS_OPCODE: `z80_device`
- Configuração herdada e com um espaço adicionado: `hd647180x_device`
- Configuração herdada e com um patch no espaço: `tmpz84c011_device`

```
bool has_configured_map(int index = 0) const;
```

12.6.3 Associando os mapas aos espaços

A associação dos mapas aos espaços é feito no nível da configuração do sistema após a instanciação do dispositivo:

```
void set_addrmap(int spacenum, T &obj, Ret (U::*func)(Params...));
void set_addrmap(int spacenum, Ret (T::*func)(Params...));
void set_addrmap(int spacenum, address_map_constructor map);
```

Estas funções associam um mapa a um determinado espaço. A associação dos mapas dos endereços com espaços que não existem são ignorados sem qualquer aviso. A primeira forma toma uma referência num objeto e um método para chamar este objeto. A segunda forma assume um método para invocar o dispositivo atual que está sendo configurado. A terceira forma toma um `address_map_constructor` para ser copiado. Em cada caso a função deve poder ser invocada como argumento através da referência de um objeto `address_map`.

Como exemplo aqui está o mapa de configuração do endereço do mapa para a CPU principal nos sistemas Hana Yayoi e na Hana Fubiki com todas as distrações removidas:

```
class hnayayoi_state : public driver_device
{
public:
    void hnayayoi(machine_config &config);
    void hnfubuki(machine_config &config);

private:
    required_device<cpu_device> m_maincpu;

    void hnayayoi_map(address_map &map);
    void hnayayoi_io_map(address_map &map);
    void hnfubuki_map(address_map &map);
};

void hnayayoi_state::hnayayoi(machine_config &config)
{
    Z80(config, m_maincpu, 20000000/4);
    m_maincpu->set_addrmap(AS_PROGRAM, &hnayayoi_state::hnayayoi_map);
    m_maincpu->set_addrmap(AS_IO, &hnayayoi_state::hnayayoi_io_map);
}

void hnayayoi_state::hnfubuki(machine_config &config)
{
    hnayayoi(config);

    m_maincpu->set_addrmap(AS_PROGRAM, &hnayayoi_state::hnfubuki_map);
    m_maincpu->set_addrmap(AS_IO, address_map_constructor());
}
```

12.6.4 Acessando os espaços

```
address_space &space(int index = 0) const;
```

Retorna um espaço de endereço específico depois da inicialização e o endereço informado deve existir.

```
bool has_space(int index = 0) const;
```

Indica se um determinado espaço fornecido realmente existe.

12.6.5 O suporte MMU para o desmontador

```
bool translate(int spacenum, int intention, offs_t &address);
```

Faz uma tradução lógica para o endereço físico através do dispositivo MMU¹. O "*spacenum*" dá o número do espaço, a intenção para o tipo do acesso futuro (TRANSLATE_(READ\|WRITE\|FETCH)(_USER\|_DEBUG)) e o endereço é um parâmetro de entrada e saída (in/out) armazenando o endereço para tradução na entrada e a versão traduzida no retorno. Deve retornar true caso a tradução seja correta ou false caso o endereço não tenha sido mapeado.

Observe que por alguma razão histórica, o próprio dispositivo deve substituir o método virtual `memory_translate` com a mesma assinatura.

¹ Memory management unit ou Unidade de gerenciamento da memória. (Nota do tradutor)

12.7 O dispositivo da interface da ROM

- *Das capacidades*
- *Configuração*
- *Acesso a ROM*
- *O Banco da ROM*
- *Ressalvas*

12.7.1 Das capacidades

Esta interface foi concebida para dispositivos que esperam ter uma ROM conectada à ela através de um barramento dedicado sendo principalmente desenvolvido para CIs de áudio. Pode haver um interesse de outros tipos de dispositivos, no entanto há outros pontos a serem levados em consideração, que pode torná-lo impraticável (como a cache da decodificação dos gráficos, por exemplo). A interface informa a possibilidade da conexão entre a região da ROM com um mapa de endereço ou então configurar dinamicamente um bloco da memória como se fosse uma ROM. Nos casos da região e blocos, este banco de memória é tratado de forma automática.

12.7.2 Configuração

```
device_rom_interface<AddrWidth, DataWidth=0, AddrShift=0, Endian=ENDIANNESS_LITTLE>
```

A interface é um modelo que pega a largura do endereçamento do barramento dedicado como um parâmetro. Além disso é possível informar a largura do barramento de dados (caso não seja um byte), o deslocamento do endereço (caso não seja 0) e o endianness¹ (caso não seja little endian ou um barramento com tamanho em byte) pode ser fornecido. A largura do barramento de dados é 0 para byte, 1 para word, etc.

```
void set_map(map);
```

Use este método na configuração do tempo do sistema para que seja providenciado a conexão de um mapa de endereçamento ao barramento. Tem prioridade sobre uma região da ROM, caso uma esteja presente.

```
void set_device_rom_tag(tag);
```

Usado para selecionar uma região da ROM que será usada caso o mapa de endereços de um dispositivo não seja informado. A predefinição é DEVICE_SELF, por exemplo, uma tag do dispositivo.

```
ROM_REGION(length, tag, flags)
```

Caso a região da ROM definida com a tag set_device_rom_tag esteja presente, seja informada nas definições da ROM para o sistema ou seja idêntica a tag do dispositivo, esta será selecionada como a ROM correta. Um mapeamento do endereço terá prioridade sobre uma região da ROM caso uma esteja presente na configuração do sistema.

```
void override_address_width(u8 width);
```

¹ Para maiores explicações sobre os diferentes tipos de endianness, acesse [este link](#). (Nota do tradutor)

Este método permite sobrescrever a largura do endereço do barramento. Deve ser invocado de dentro do dispositivo antes da conclusão do `config_complete`.

```
void set_rom(const void *base, u32 size);
```

A qualquer momento publique `interface_pre_start` com este método, um bloco da memória pode ser configurado como se uma ROM estivesse conectada. Sobrescreve qualquer configuração prévia que possa ter sido fornecida. Pode ser feito mais de uma vez.

12.7.3 Acesso a ROM

```
u8 read_byte(off_t addr);  
u16 read_word(off_t addr);  
u32 read_dword(off_t addr);  
u64 read_qword(off_t addr);
```

Estes métodos fornecem o acesso de leitura para uma ROM que esteja conectada. O acesso fora dos limites retorna mensagens de erro `logerror` sobre o não mapeamento.

12.7.4 O Banco da ROM

Caso a região da ROM ou o bloco da memória no `set_rom` seja maior que o barramento de endereços possa acessar, o banco da ROM² será configurado de forma automática.

```
void set_rom_bank(int bank);
```

Este método seleciona o número do banco atual.

12.7.5 Ressalvas

Ao usar aquela interface, faz com que o dispositivo derive do `device_memory_interface`. Caso o dispositivo queira realmente utilizar a interface da memória para si mesmo, lembre-se que o espaço zero (0 ou `AS_PROGRAM`) é utilizado pela interface da ROM, por isso não se esqueça de invocar a base do método `memory_space_config`.

Para os dispositivos com saídas que possam ser utilizadas para endereçar as ROMs restrito apenas ao encaminhamento dos dados para outro dispositivo com a única finalidade de processamento, pode ser que seja de grande ajuda desativar a interface quando ela não estiver sendo utilizada. Isto pode ser feito sobrescrevendo o `memory_space_config` para que retorne um vetor vazio.

² Rom banking no texto original. (Nota do tradutor)

12.8 O device_disasm_interface e os desmontadores

- *Das capacidades*
- *Os desmontadores*
 - *Definição*
 - *opcode_alignment*
 - *disassemble*
 - *interface_flags*
 - *pc_linear_to_real e pc_real_to_linear*
 - *page_address_bits*
 - *page2_address_bits*
 - *decryptnn*
- *Interface do desmontador; device_disasm_interface*
 - *Definição*
 - *create_disassembler*
- *A comunicação e a configuração do desmontador*
- *Coisas que ainda faltam*

12.8.1 Das capacidades

Os desmontadores são classes que fornecem desmontagem e o opcode meta informações para os núcleos da CPU e unidasm. O device_disasm_interface conecta um núcleo da CPU com o seu desmontador.

12.8.2 Os desmontadores

Definição

Um desmontador é uma classe que deriva de util::disasm_interface. Em seguida, ele tem dois métodos necessários de implementação, opcode_alignment e disassemble assim como 6 opcionais, interface_flags, page_address_bits, pc_linear_to_real, pc_real_to_linear e uma com quatro variantes possíveis, decrypt8/16/32/64.

opcode_alignment

```
u32 \ **opcode_alignment**\ () const
```

Retorna o alinhamento de opcode requisitado pela CPU nas unidades PC. Em outras palavras o alinhamento necessário para os registros PC da CPU. Tende a ser 1 (quase todos), 2 (68000...), 4 (mips, ppc...), com um excepcional 8 (processador paralelo tms 32082) e 16 (tms32010, instruções são 16-bits aligned e o PC targets bits). Deve ser a potência de dois para evitar que as coisas se quebrem.

Note que processadores como o tms32031 que têm instruções em 32-bits onde os valores PC targets em 32-bits têm um alinhamento de 1.

disassemble

```
offs_t \ **disassemble**\ (std::ostream &stream, offs_t pc, const data_buffer &opcodes,   
↳ const data_buffer &params)
```

Este é o método onde o trabalho de fato é acontece. Esse comando desmonta uma instrução no endereço *PC* e escreve o resultado para *stream*. Os valores a serem decodificados são recuperados da memória intermediária *opcode*. Um objeto *data_buffer* oferecem quatro métodos de acesso:

```
u8  util::disasm_interface::data_buffer::\ **r8**\ (offs_t pc) const
u16 util::disasm_interface::data_buffer::\ **r16**\ (offs_t pc) const
u32 util::disasm_interface::data_buffer::\ **r32**\ (offs_t pc) const
u64 util::disasm_interface::data_buffer::\ **r64**\ (offs_t pc) const
```

Eles leem os dados num determinado endereço e pegam o endianness e os PCs não lineares por acessos maiores que a largura do barramento. A variante do depurador também armazena em cache os dados lidos num bloco, então por essa razão um não deve ler os dados muito longe da base pc (ficar entre de 16K ou então, ter cuidado ao tentar seguir acessos indiretos, por exemplo).

Uma quantidade de CPUs tem um sinal externo que divide as buscas em parte um opcode e parte um parâmetro. Este é, por exemplo o sinal M1 do z80 ou o sinal SYNC do 6502. Alguns sistemas apresentam diferentes valores para a CPU dependendo se esse sinal for ativo, em geral usado para fins de proteção. Nestes CPUs a parte do opcode deve ser lida a partir da memória intermediária do *opcode* e o parâmetro *part* vindo da memória intermediária *params*. Eles serão ou não a mesma memória intermediária, tudo vai depender do próprio sistema.

O método retorna o tamanho da instrução em unidades de PC, com um valor máximo de 65535. Além disso, caso seja possível o desmontador deve dar algumas informações meta sobre o opcode por "OR-ing" no resultado:

- **STEP_OVER** para chamadas de sub-rotina ou auto-decrementos de loops. Caso haja alguns slots com atraso, faça também OR com **step_over_extra**(n) onde n é o número da instrução.
- **STEP_OUT** para o retorno das instruções da sub-rotina

Além disso, para indicar que esses sinalizadores são compatíveis, OU o resultado com **SUPPORTED**. Uma quantidade chata de desmontadores mentem sobre essa compatibilidade (eles fazem um OR com **SUPPORTED** mesmo sem gerar o **STEP_OVER** ou **STEP_OUT**, por exemplo). Não faça isso, pois quebra a funcionalidade do *step over/step out* do depurador.

interface_flags

```
u32 **interface_flags**\ () const
```

Esse método opcional mostra detalhes do desmontador. O valor zero predefinido é o correto na maioria das vezes. As bandeiras possíveis e que precisam ser "OR-ed" juntas, são:

- **NONLINEAR_PC**: passar para o próximo opcode ou o próximo byte do opcode se não adicionar um ao pc. Usado para antigos PCs com base em LFSR.
- **PAGED**: o PC é envolvido com um limite de página
- **PAGED2LEVEL**: não apenas o PC envolve em algum tipo de limite de página, mas há dois níveis de paginação
- **INTERNAL_DECRYPTION**: há alguma descryptografia escondida entre a leitura de AS_PROGRAM e o desmontador atual
- **SPLIT_DECRYPTION**: há alguma descryptografia escondida entre a leitura do AS_PROGRAM e o desmontador atual, assim como essa descryptografia é diferente para os opcodes e os parâmetros

Note que, na prática, os sistemas de PC não lineares também são paginados, o PAGED2LEVEL implica no PAGED e o SPLIT_DECRYPTION implica em DECRYPTION.

pc_linear_to_real e pc_real_to_linear

```
offs_t **pc_linear_to_real**\ (offs_t pc) const
offs_t **pc_real_to_linear**\ (offs_t pc) const
```

Esses métodos devem estar presentes apenas quando NONLINEAR_PC estiver definido nos sinalizadores da interface. Eles devem converter o PC de e para um valor com destino a um domínio linear onde os parâmetros de instrução e a próxima instrução sejam alcançadas ao incrementar o valor. O pc_real_to_linear converte para aquele domínio, já o pc_linear_to_real é convertido de volta daquele domínio.

page_address_bits

```
u32 **page_address_bits**\ () const
```

Presente quando **PAGED** ou **PAGED2LEVEL** for definido, retorna a quantidade de endereços de bits na pagina inferior.

page2_address_bits

```
u32 **page2_address_bits**\ () const
```

Presente quando **PAGED2LEVEL** for definido, retorna a quantidade de endereços de bits na página superior.

decryptnn

```
u8  **decrypt8**\ (u8 value, offs_t pc, bool opcode) const
u16 **decrypt16**\ (u16 value, offs_t pc, bool opcode) const
u32 **decrypt32**\ (u32 value, offs_t pc, bool opcode) const
u64 **decrypt64**\ (u64 value, offs_t pc, bool opcode) const
```

Um destes deve ser definido quando INTERNAL_DECRYPTION ou SPLIT_DECRYPTION for configurado. O escolhido será aquele que leva o que opcode_alignment representa em bytes.

Esse método descriptografa um determinado valor do endereço PC (a partir do AS_PROGRAM) e retorna o que será passado para o desmontador. No caso da descriptografia dividida, o opcode indica se estamos no opcode (true) ou na parte da instrução do parâmetro (false).

12.8.3 Interface do desmontador, device_disasm_interface

Definição

Um núcleo de CPU deriva de device_disasm_interface através do cpu_device . Um método deve ser implementado, create_disassembler .

create_disassembler

```
util::disasm_interface \*\ **create_disassembler**\ ()
```

Esse método deve retornar um ponteiro para um novo objeto desmontado que foi recém-alocado. O solicitante apropria-se do objeto e lida com o seu tempo de vida.

Esse método será chamado no máximo uma vez durante a vida útil do objeto da CPU.

12.8.4 A comunicação e a configuração do desmontador

Alguns desmontadores precisam ser configurados. A configuração pode ser imutável (estático) duração da execução (como o modelo da CPU por exemplo) ou dinâmico (o estado de um sinalizador ou uma preferência de usuário). A configuração estática que pode ser feita seja por parâmetro(s) para o construtor do desmontador ou através da derivação da classe do desmontador principal. Caso a informação seja curta e sua semântica seja óbvia (como o nome do modelo), fique à vontade para usar um parâmetro. Caso contrário, deriva a classe.

A configuração dinâmica deve ser feita definindo primeiro uma estrutura de grupo público chamado "config" no desmontador, com o destruidor virtual e métodos virtuais puros para extrair as informações necessárias. Um ponteiro para essa estrutura deve ser passada para o construtor do desmontador. O núcleo da CPU deve então adicionar uma derivação dessa estrutura de configuração e implementar os métodos. O Unidasm terá que separar pequena classe da configuração de classes para que possa passar a informação.

12.8.5 Coisas que ainda faltam

Atualmente, não há como a GUI do depurador adicionar uma configuração para cada núcleo. Ela se faz necessária para o s2650 e os núcleos do saturn. É necessário também passar pela própria classe do núcleo da CPU uma vez que é retirado da estrutura de configuração.

Falta compatibilidade do unidasm para uma configuração individual dos núcleos da CPU. Isso se faz útil para muitas coisas, veja o código-fonte do unidasm para a um lista atual (comentários "Configuration missing").

12.9 A memória emulada do sistema e o gerenciamento dos espaços de endereçamento

- *Resumo*
- *Conceitos Básicos*
 - *O Endereçamento*
 - *Os espaços de endereçamento*
 - *As ações, os bancos e as regiões*
 - *Visualizações*
- *Os objetos da memória*
 - *Compartilhamentos - memory_share*
 - *Bancos - memory_bank*
 - *Regiões - memory_region*
 - *Visualizações - memory_view*
- *O API dos mapas de endereçamentos*
 - *A estrutura geral da API*
 - *As configurações globais*
 - * *Mascaramento global*
 - * *O valor retornado na leitura não mapeada/nop-ed*
 - *A configuração do manipulador*
 - * *O método no dispositivo atual*
 - * *O método num dispositivo diferente*
 - * *A função lambda*
 - * *O acesso direto à memória*
 - * *O acesso ao banco*
 - * *O acesso à porta*
 - * *Os acessos descartados*
 - * *O acesso não mapeado*
 - * *O mapeamento do sub-dispositivo*
 - *Os qualificadores de alcance*
 - * *O espelhamento*
 - * *O mascaramento*
 - * *A seleção*
 - * *A seleção da subunidade*
 - * *O manuseio da seleção do CI na subunidade*

- * *O sinalizador do usuário*
 - *Configuração da visualização*
- *O API do mapeamento dinâmico do espaço de endereçamento*
 - *A estrutura geral da API*
 - *O mapeamento do manipulador*
 - *O mapeamento direto da faixa do intervalo da memória*
 - *O mapeamento do banco*
 - *O mapeamento da porta*
 - *Os acessos abandonados*
 - *Os acessos não mapeados*
 - *A instalação do mapa do dispositivo*
 - *A instalação da visualização*

12.9.1 Resumo

O subsistema da memória (emumem e o addrmap) combina várias funções úteis para a emulação do sistema:

- a decodificação do barramento do endereço e o despacho com cache
- as descrições estáticas de um espaço de endereçamento
- a alocação da RAM e do registro para o registro do estado
- a interação com as regiões da memória para acessar a rom

Os dispositivos criam espaços de endereçamento, como por exemplo um barramento decodificável, através do `device_memory_interface`. A configuração do sistema define o endereço dos mapas para colocar nos espaços de endereçamento, assim o dispositivo pode ler e escrever através do barramento.

12.9.2 Conceitos Básicos

O Endereçamento

Um espaço de endereçamento implementado na classe **address_space**, representa um barramento endereçável com vários sub-dispositivos em potenciais conectados que exigem uma decodificação. Possui várias linhas de dados (8, 16, 32 ou 64) chamada largura dos dados, várias linhas de endereços (1 a 32) chamado largura do endereço e um **endianness**. Além disso uma mudança de endereço permite que barramentos que tenham uma granularidade atômica diferente de 1 byte.

Os objetos do espaço de endereço fornecem uma série de métodos para o acesso a leitura e a gravação, assim como uma segunda série de métodos para alterar dinamicamente a decodificação.

Os espaços de endereçamento

Um espaço de endereçamento é uma descrição estática da decodificação prevista quando um barramento for utilizado. Ele se conecta à memória, a outros dispositivos e a outros métodos, geralmente é instalado na inicialização de um endereçamento. Esta descrição é armazenada numa estrutura **address_map** que é preenchida programaticamente.

As ações, os bancos e as regiões

Os compartilhamentos da memória são regiões da memória que são alocadas e podem ser colocadas em vários lugares no mesmo espaço de endereço ou em um endereço diferente, eles também podem ser acessado de forma direta a partir dos dispositivos.

Os bancos de memória são regiões que acessam a memória de forma indireta dando a possibilidade de se alterar de forma dinâmica e eficiente a zona de uma região específica.

As regiões da memória são regiões da memória de somente leitura onde as ROMs são carregadas.

Todos eles têm nomes com permissão de acesso.

Visualizações

As visualizações são uma forma de misturar diferentes submapas numa faixa da memória com comutação rápida. É para ser usado quando diversos dispositivos mapearem nos mesmos endereços e forem comutados externamente. Elas devem ser criadas como um objeto do dispositivo e depois configuradas estaticamente num mapa de memória ou dinamicamente através das chamadas `install_*`.

Os submapas intercambiáveis, também conhecidos como variantes, são nomeados através de um número inteiro. Uma indexação interna através de um mapa garante que qualquer valor inteiro possa ser usado.

12.9.3 Os objetos da memória

Compartilhamentos - memory_share

```
class memory_share {
    const std::string &name() const;
    void *ptr() const;
    size_t bytes() const;
    endianness_t endianness() const;
    u8 bitwidth() const;
    u8 bytewidth() const;
};
```

Um compartilhamento da memória é uma zona nomeada da memória alocada que é automaticamente gravada como estados e podem ser mapeados nos endereçamentos. É o contêiner predefinido para a memória que é compartilhada entre os espaços, mas também compartilhado entre uma CPU emulada e um driver. Como tal, é fácil ter acesso ao seu conteúdo a partir da classe do driver.

```
required_shared_ptr<uNN> m_share_ptr;
optional_shared_ptr<uNN> m_share_ptr;
required_shared_ptr_array<uNN, count> m_share_ptr_array;
optional_shared_ptr_array<uNN, count> m_share_ptr_array;

[device constructor] m_share_ptr(*this, "name"),
[device constructor] m_share_ptr_array(*this, "name%u", 0U),
```

No nível do dispositivo, um ponteiro para a zona de memória pode ser facilmente recuperada através da construção de um destes quatro localizadores. Observe que como cada localizador chamando um target() no localizador dá a você o objeto memory_share.

```
memory_share_creator<uNN> m_share;

[device constructor] m_share(*this, "name", size, endianness),
```

Um compartilhamento da memória pode ser criado caso ele não exista num mapa da memória através dessa classe de criação. Caso já exista basta recupera-la. Esta classe se comporta como um ponteiro mas também tem o método target() para obter o objeto memory_share e os métodos de compartilhamento de informação bytes(), endianness(), bitwidth() e o bytewidth().

```
memory_share *memshare(string tag) const;
```

O método do dispositivo memshare recupera um compartilhamento da memória por nome. Cuidado pois a pesquisa pode ser dispendiosa, em vez disso prefira os localizadores.

Bancos - memory_bank

```
class memory_bank {
    const std::string &tag() const;
    int entry() const;
    void set_entry(int entrynum);
    void configure_entry(int entrynum, void *base);
    void configure_entries(int startentry, int numentry, void *base, offs_t stride);
    void set_base(void *base);
    void *base() const;
};
```

Um banco de memória é um desreferenciamento do nome da zona da memória que pode ser mapeada nos espaços de endereçamento. Ele aponta para nullptr quando é criado. O `configure_entry` permite definir uma relação entre um número da entrada e um ponteiro base. O `configure_entries` faz o mesmo através das diversas entradas consecutivas que abrangem uma zona da memória. Alternativamente o `set_base` define a base para a entrada 0 e a seleciona.

O `set_entry` permite selecionar de forma dinâmica e eficientemente a entrada ativa atual, o `entry()` obtém esta seleção de volta e `base()` obtém o ponteiro associado a base.

```
required_memory_bank m_bank;
optional_memory_bank m_bank;
required_memory_bank_array<count> m_bank_array;
optional_memory_bank_array<count> m_bank_array;

[device constructor] m_bank(*this, "name"),
[device constructor] m_bank_array(*this, "name%u", 0U),
```

No nível do dispositivo, um ponteiro para o objeto do banco da memória pode ser facilmente recuperado ao construir um destes quatro localizadores.

```
memory_bank_creator m_bank;

[device constructor] m_bank(*this, "name"),
```

Um banco de memória pode ser criado caso ele não exista num mapa de memória através dessa classe de criação. Caso já exista basta recuperá-la.

```
memory_bank *membank(string tag) const;
```

O método do dispositivo `membank` recupera um compartilhamento da memória por nome. Cuidado pois a pesquisa pode ser dispendiosa, em vez disso prefira os localizadores.

Regiões - memory_region

```
class memory_region {
    u8 *base();
    u8 *end();
    u32 bytes() const;
    const std::string &name() const;
    endianness_t endianness() const;
    u8 bitwidth() const;
    u8 bytewidth() const;
    u8 &as_u8(offset_t offset = 0);
    u16 &as_u16(offset_t offset = 0);
    u32 &as_u32(offset_t offset = 0);
    u64 &as_u64(offset_t offset = 0);
}
```

Uma região é usada para armazenar dados de somente leitura, como as ROMs ou o resultado das descrições fixadas. O seu conteúdo não são salvos, é por isso que eles não devem ser gravado a partir do sistema emulado. Eles na realidade não possuem uma largura intrínseca (base() sempre retorna um u8 *), que é histórico e praticamente impossível de consertar neste ponto. Os métodos as_* permitem acessá-los a partir de uma determinada largura.

```
required_memory_region m_region;
optional_memory_region m_region;
required_memory_region_array<count> m_region_array;
optional_memory_region_array<count> m_region_array;

[device constructor] m_region(*this, "name"),
[device constructor] m_region_array(*this, "name%u", 0U),
```

No nível do dispositivo, um ponteiro para o objeto da região da memória pode ser facilmente recuperado através da construção de um destes quatro localizadores.

```
memory_region *memregion(string tag) const;
```

O método do dispositivo memregion recupera um compartilhamento da memória por nome. Cuidado pois a pesquisa pode ser dispendiosa, em vez disso prefira os localizadores.

Visualizações - memory_view

```
class memory_view {
    memory_view(device_t &device, std::string name);
    memory_view_entry &operator[](int slot);

    void select(int entry);
    void disable();

    const std::string &name() const;
}
```

Uma visualização permite alternar parte de um mapa de memória entre diversas possibilidades ou mesmo desabilitá-lo completamente para ver o que estava lá antes. Ele é criado como um objeto do dispositivo.

```
memory_view m_view;

[device constructor] m_view(*this, "name"),
```

Então será configurado através da API do mapa de endereços ou dinamicamente. Durante a execução uma quantidade de variantes podem ser selecionadas utilizando o método `select` ou a visualização pode ser desativada utilizando o método `disable`. Uma visualização desativada pode ser reativada a qualquer momento.

12.9.4 O API dos mapas de endereçamentos

A estrutura geral da API

Um espaço de endereçamento é um método onde um dispositivo que preenche a estrutura de um **address_map** geralmente chamada de **mapa**, passada através de uma referência. O método então pode definir alguma configuração global através de métodos específicos e em seguida, oferecer as entradas orientadas para o intervalo de endereços que indicam o que deve acontecer quando um intervalo específico for acessado.

A sintaxe geral para as entradas utiliza um método de encadeamento:

```
map(start, end).handler(...).handler_qualifier(...).range_qualifier();
```

Os valores start e end definem o intervalo, o bloco handler() define como o acesso é tratado, o bloco handler_qualifier() determina alguns aspectos do manipulador (como o compartilhamento da memória, por exemplo) e o O bloco range_qualifier() refina o intervalo (o espelhamento, o mascaramento, o byte a seleção...).

O mapa segue um princípio do "o último ganha", onde o último é selecionado quando vários manipuladores correspondem a um determinado endereço.

As configurações globais

Mascaramento global

```
map.global_mask(off_t mask);
```

Permite indicar uma máscara que será aplicada em todos os endereços quando acessar o espaço onde o mapa estiver instalado.

O valor retornado na leitura não mapeada/nop-ed

```
map.unmap_value_low();  
map.unmap_value_high();  
map.unmap_value(u8 value);
```

Define o valor para retornar nas leituras para um endereço não mapeado ou sem saída. Low significa 0, high ~0.

A configuração do manipulador

O método no dispositivo atual

```
(...).r(FUNC(my_device::read_method))
(...).w(FUNC(my_device::write_method))
(...).rw(FUNC(my_device::read_method), FUNC(my_device::write_method))

uNN my_device::read_method(address_space &space, offs_t offset, uNN mem_mask)
uNN my_device::read_method(address_space &space, offs_t offset)
uNN my_device::read_method(address_space &space)
uNN my_device::read_method(offs_t offset, uNN mem_mask)
uNN my_device::read_method(offs_t offset)
uNN my_device::read_method()

void my_device::write_method(address_space &space, offs_t offset, uNN data, uNN mem_mask)
void my_device::write_method(address_space &space, offs_t offset, uNN data)
void my_device::write_method(address_space &space, uNN data)
void my_device::write_method(offs_t offset, uNN data, uNN mem_mask)
void my_device::write_method(offs_t offset, uNN data)
void my_device::write_method(uNN data)
```

Define um método do dispositivo ou driver atual para ler, escrever ou ambos na entrada atual. O protótipo do método pode levar diversas formas que tornam alguns elementos opcionais. uNN representa u8, u16, u32 ou u64 dependendo da largura dos dados do manipulador. O manipulador pode ser menos largo do que o próprio barramento (por exemplo, um dispositivo de 8 bits num barramento com 32 bits).

O offset informado é criado a partir do endereço de acesso. Começa com zero no início do intervalo com incrementos para cada unidade uNN. Um manipulador u8 obterá um offset em bytes, um u32 em words duplas. O mem_mask tem os seus bits definidos onde os acessadores de fato fazem a condução do bit. Geralmente é construído em unidades de byte, porém em alguns casos dos CIs das portas de E/S com os registros de direção por bit, a resolução pode estar no nível de bits.

O método num dispositivo diferente

```
(...).r(m_other_device, FUNC(other_device::read_method))
(...).r("other-device-tag", FUNC(other_device::read_method))
(...).w(m_other_device, FUNC(other_device::write_method))
(...).w("other-device-tag", FUNC(other_device::write_method))
(...).rw(m_other_device, FUNC(other_device::read_method), FUNC(other_device::write_method))
(...).rw("other-device-tag", FUNC(other_device::read_method), FUNC(other_device::write_
↪method))
```

Define um método de um outro dispositivo, designado através de um localizador (required_device ou optional_device) ou sua tag, para ler, escrever ou ambos na entrada atual.

A função lambda

```
(...).lr{8,16,32,64}(NAME([...])(address_space &space, offs_t offset, uNN mem_mask) -> uNN { .
↳ ... })))
(...).lr{8,16,32,64}([...])(address_space &space, offs_t offset, uNN mem_mask) -> uNN { ... },
↳ "name")
(...).lw{8,16,32,64}(NAME([...])(address_space &space, offs_t offset, uNN data, uNN mem_mask) ->
↳ void { ... })))
(...).lw{8,16,32,64}([...])(address_space &space, offs_t offset, uNN data, uNN mem_mask) ->
↳ void { ... }, "name")
(...).lrw{8,16,32,64}(NAME(read), NAME(write))
(...).lrw{8,16,32,64}(read, "name_r", write, "name_w")
```

Define um lambda que é chamada durante a leitura, a gravação ou em ambos. O protótipo lambda pode ser qualquer um dos 6 métodos disponíveis. Um pode ou utilizar FUNC() sobre toda a lambda ou informar um nome após a definição da lambda. O número é a largura de dados do acesso, como o NN por exemplo.

O acesso direto à memória

```
(...).rom()
(...).writeonly()
(...).ram()
```

Seleciona uma faixa do intervalo para acessar uma zona da memória como somente leitura (read-only), somente gravação (write-only) ou leitura e gravação (read-write) respectivamente. Qualificadores específicos do manipulador permite dizer onde esta zona da memória deveria estar. Existem dois casos onde não qualificador é aceitável:

- ram() dá uma zona ram anônima não acessível fora do espaço de endereçamento.
- rom() quando o mapa da memória é utilizado num AS_PROGRAM do espaço do dispositivo (CPU) cujos nomes também sejam o nome de uma região. Em seguida, a zona da memória aponta para essa região no offset correspondente ao início da zona.

```
(...).rom().region("name", offset)
```

O qualificador da região permite fazer um ponto somente leitura da zona para o conteúdo de uma determinada região num determinado offset.

```
(...).rom().share("name")
(...).writeonly().share("name")
(...).ram().share("name")
```

O qualificador de compartilhamento permite fazer o ponto da zona para uma região da memória compartilhada definida através do seu nome. Caso a região esteja presente em diversos espaços o endianness deve corresponder caso o tamanho, a largura do barramento e se o barramento tiver mais do que um byte de largura.

O acesso ao banco

```
(...).bankr("name")  
(...).bankw("name")  
(...).bankrw("name")
```

Define a faixa do intervalo para apontar para o conteúdo de um banco que é lido, escrito ou em modo de leitura e escrita.

O acesso à porta

```
(...).portr("name")  
(...).portw("name")  
(...).portrw("name")
```

Define a faixa do intervalo para apontar para uma porta de E/S.

Os acessos descartados

```
(...).nopr()  
(...).nopw()  
(...).noprw()
```

Define a faixa do intervalo para descartar o acesso sem registrar o log. Durante a leitura, um valor não mapeado é retornado.

O acesso não mapeado

```
(...).unmapr()  
(...).unmapw()  
(...).unmaprw()
```

Define a faixa do intervalo para descartar o acesso com registro no log. Durante a leitura, um valor não mapeado é retornado.

O mapeamento do sub-dispositivo

```
(...).m(m_other_device, FUNC(other_device::map_method))  
(...).m("other-device-tag", FUNC(other_device::map_method))
```

Inclui um submapa definido pelo dispositivo. O início da faixa do intervalo indica onde termina o endereço zero do submapa, e o fim do intervalo corta o submapa caso seja necessário. Observe que os qualificadores do intervalo (definidos posteriormente) se aplicam.

Atualmente, apenas manipuladores são permitidos nos submapas e não nas regiões da memória ou nos bancos.

Os qualificadores de alcance

O espelhamento

```
(...).mirror(mask)
```

Duplica o intervalo nos endereços que estiverem acessíveis, definindo qualquer um dos 1 bits presentes na máscara. Por exemplo, um intervalo 0-0x1f com um espelho 0x300 estará presente em 0-0x1f, 0x100-0x11f, 0x200-0x21f e 0x300-0x31f. Os endereços informados para o manipulador permanecem no intervalo 0-0x1f, os bits do espelho não são vistos.

O mascaramento

```
(...).mask(mask)
```

Válido apenas com os manipuladores, o endereço será mascarado com a máscara antes de ser passado para o manipulador.

A seleção

```
(...).select(mask)
```

Válido apenas com manipuladores, a faixa do intervalo será espelhado com espelho, mas os bits de endereçamento do espelho serão mantidos no offset informado para o manipulador quando for chamado. Isso é útil para os dispositivos como o CI de áudio onde os bits mais baixos do endereçamento selecionam uma função e os bits mais altos um número da voz.

A seleção da subunidade

```
(...).umask16(16-bits mask)
(...).umask32(32-bits mask)
(...).umask64(64-bits mask)
```

Válido apenas com manipuladores e submapas, seleciona quais as linhas dos dados do barramento estão realmente conectados ao manipulador ou aos dispositivos. O dispositivo atual deve ser um múltiplo de um byte, por exemplo, a máscara é uma série de 00 e ff. O offset será ajustado de acordo, de modo que a diferença de 1 significa a próxima unidade manuseada no acesso.

CASO a máscara seja mais estreita do que a largura do barramento, a máscara será replicada nas linhas superiores.

O manuseio da seleção do CI na subunidade

```
(...).cselect(16/32/64)
```

Quando um dispositivo está conectado na parte do barramento, como um byte num barramento de 16 bits, o manipulador do destino só é ativado quando essa parte for de fato acessada. Em alguns casos o acesso do byte num barramento de 16-bits 68000 o hardware atual verifica apenas o word do endereço e não se o byte correto é acessado. O `cswidth` permite informar a memória do sistema para acionar o manipulador caso uma parte mais ampla do barramento seja acessada. O parâmetro é a largura do gatilho (seria 16 no caso do 68000).

O sinalizador do usuário

```
(...).flags(16-bits mask)
```

Este parâmetro permite que o usuário defina os sinalizadores no manipulador e que podem então ser recuperadas através do acesso de um dispositivo, alterando o seu comportamento. Um exemplo da utilização do `i960` que marca dessa maneira as regiões de risco (elas têm um suporte específico a nível de hardware).

Configuração da visualização

```
map(start, end).view(m_view);  
m_view[0](start1, end1).[...];
```

Uma visualização é configurada num mapa de endereços com o método de visualização. O único qualificador aceito é o espelho. A versão "desativada" da visualização incluirá o que estava na faixa antes da configuração da visualização.

As diferentes variantes são configuradas através da indexação da visualização com o número da variante e da criação de uma entrada da maneira usual. As entradas dentro de uma variante devem permanecer dentro do limite. Não há outras restrições adicionais. O conteúdo de uma variante, por padrão é o que estava lá antes, por exemplo, o conteúdo da vista desabilitada, e então a configuração permite anular parte ou a totalidade dela.

As variantes só podem ser configuradas uma vez que a própria visualização tenha sido configurada com o método `view`.

Uma visualização só pode ser colocada num mapa de endereços e em apenas uma posição. Caso várias visualizações tenham o mesmo conteúdo ou similar, lembre-se que a criação de um mapa não é mais do que uma chamada do método e a criação de um segundo método para configurar uma visualização é perfeitamente razoável. Uma visualização é do tipo `memory_view` e uma entrada indexada (por exemplo, uma variante para configuração) é do tipo `memory_view::memory_view_entry &`.

Uma visualização pode ser instalada em outra visualização mas não se esqueça que uma visualização pode ser instalada apenas uma vez. Uma visualização também pode fazer parte do "que estava lá antes".

12.9.5 O API do mapeamento dinâmico do espaço de endereçamento

A estrutura geral da API

Uma série de métodos permite alterar a decodificação do barramento de um espaço de endereçamento em tempo real. Eles são poderosos, porém têm alguns problemas:

- Alterando os mapeamentos de forma repetida pode causar lentidão
- O estado do espaço do endereçamento não é registrado nos estados salvos, portanto, deve ser reconstruído após o carregamento do estado
- Podem ser ocultados em qualquer lugar, em vez de agrupados num mapa do endereçamento, que pode ser menos legível

Os métodos em vez de decompor as informações no manipulador, o qualificador do manipulador e a faixa do intervalo do qualificador os coloca todos juntos como parâmetros do método. Para tornar as coisas um pouco mais legíveis, muitos deles são opcionais, porém, os opcionais sendo escritos em *itálico*.

O mapeamento do manipulador

```
uNN my_device::read_method(address_space &space, offs_t offset, uNN mem_mask)
uNN my_device::read_method_m(address_space &space, offs_t offset)
uNN my_device::read_method_mo(address_space &space)
uNN my_device::read_method_s(offs_t offset, uNN mem_mask)
uNN my_device::read_method_sm(offs_t offset)
uNN my_device::read_method_smo()

void my_device::write_method(address_space &space, offs_t offset, uNN data, uNN mem_mask)
void my_device::write_method_m(address_space &space, offs_t offset, uNN data)
void my_device::write_method_mo(address_space &space, uNN data)
void my_device::write_method_s(offs_t offset, uNN data, uNN mem_mask)
void my_device::write_method_sm(offs_t offset, uNN data)
void my_device::write_method_smo(uNN data)

readNN_delegate (device, FUNC(read_method))
readNNm_delegate (device, FUNC(read_method_m))
readNNmo_delegate (device, FUNC(read_method_mo))
readNNs_delegate (device, FUNC(read_method_s))
readNNsm_delegate (device, FUNC(read_method_sm))
readNNsmo_delegate(device, FUNC(read_method_smo))

writeNN_delegate (device, FUNC(write_method))
writeNNm_delegate (device, FUNC(write_method_m))
writeNNmo_delegate (device, FUNC(write_method_mo))
writeNNs_delegate (device, FUNC(write_method_s))
writeNNsm_delegate (device, FUNC(write_method_sm))
writeNNsmo_delegate(device, FUNC(write_method_smo))
```

Para ser adicionado a um mapa, um método chama e o dispositivo é chamado para serem agrupados no tipo delegado de forma apropriada. São 12 tipos, para a leitura, para a escrita e para todos os seis protótipos possíveis. Observe que como todos os delegados, eles também podem envolver lambdas.

```
space.install_read_handler(addrstart, addrend, read_delegate, unitmask, cswidth, flags)
space.install_read_handler(addrstart, addrend, addrmask, addrmirror, addrselect, read_
↳delegate, unitmask, cswidth, flags)
space.install_write_handler(addrstart, addrend, write_delegate, unitmask, cswidth, flags)
space.install_write_handler(addrstart, addrend, addrmask, addrmirror, addrselect, write_
↳delegate, unitmask, cswidth, flags)
space.install_readwrite_handler(addrstart, addrend, read_delegate, write_delegate, unitmask, _
↳cswidth, flags)
space.install_readwrite_handler(addrstart, addrend, addrmask, addrmirror, addrselect, read_
↳delegate, write_delegate, unitmask, cswidth, flags)
```

Estes seis métodos permitem instalar manipuladores empacotados num espaço de endereçamento em tempo real, seja plano, com máscara, *mirror* (espelho) e *select* (seleção). No caso de leitura e escrita, ambos os delegados devem ter o mesmo tipo (coisa sm) para evitar uma explosão combinatória dos tipos dos métodos. Os argumentos *unitmask*, *cswidth* e *flags* são opcionais.

O mapeamento direto da faixa do intervalo da memória

```
space.install_rom(addrstart, addrend, void *pointer)
space.install_rom(addrstart, addrend, addrmirror, void *pointer)
space.install_rom(addrstart, addrend, addrmirror, flags, void *pointer)
space.install_writeonly(addrstart, addrend, void *pointer)
space.install_writeonly(addrstart, addrend, addrmirror, void *pointer)
space.install_writeonly(addrstart, addrend, addrmirror, flags, void *pointer)
space.install_ram(addrstart, addrend, void *pointer)
space.install_ram(addrstart, addrend, addrmirror, void *pointer)
space.install_ram(addrstart, addrend, addrmirror, flags, void *pointer)
```

Instala um bloco de memória num espaço do endereço com ou sem espelho e sinalização. A *_rom* é somente leitura, a *_ram* é leitura e escrita, *_writeonly* é somente gravação. O ponteiro não deve ser nulo, este método não aloca memória.

O mapeamento do banco

```
space.install_read_bank(addrstart, addrend, memory_bank *bank)
space.install_read_bank(addrstart, addrend, addrmirror, memory_bank *bank)
space.install_read_bank(addrstart, addrend, addrmirror, flags, memory_bank *bank)
space.install_write_bank(addrstart, addrend, memory_bank *bank)
space.install_write_bank(addrstart, addrend, addrmirror, memory_bank *bank)
space.install_write_bank(addrstart, addrend, addrmirror, flags, memory_bank *bank)
space.install_readwrite_bank(addrstart, addrend, memory_bank *bank)
space.install_readwrite_bank(addrstart, addrend, addrmirror, memory_bank *bank)
space.install_readwrite_bank(addrstart, addrend, addrmirror, flags, memory_bank *bank)
```

Num espaço de endereçamento, instala um banco já existente da memória para leitura, gravação ou ambos.

O mapeamento da porta

```
space.install_read_port(addrstart, addrend, const char *rtag)
space.install_read_port(addrstart, addrend, addrmirror, const char *rtag)
space.install_read_port(addrstart, addrend, addrmirror, flags, const char *rtag)
space.install_write_port(addrstart, addrend, const char *wtag)
space.install_write_port(addrstart, addrend, addrmirror, const char *wtag)
space.install_write_port(addrstart, addrend, addrmirror, flags, const char *wtag)
space.install_readwrite_port(addrstart, addrend, const char *rtag, const char *wtag)
space.install_readwrite_port(addrstart, addrend, addrmirror, const char *rtag, const char
↳ *wtag)
space.install_readwrite_port(addrstart, addrend, addrmirror, flags, const char *rtag, const
↳ char *wtag)
```

Instala portas através de um nome para leitura, a gravação ou ambas.

Os acessos abandonados

```
space.nop_read(addrstart, addrend, addrmirror, flags)
space.nop_write(addrstart, addrend, addrmirror, flags)
space.nop_readwrite(addrstart, addrend, addrmirror, flags)
```

Descarta os acessos para uma faixa do intervalo determinado com um espelho opcional.

Os acessos não mapeados

```
space.unmap_read(addrstart, addrend, addrmirror, flags)
space.unmap_write(addrstart, addrend, addrmirror, flags)
space.unmap_readwrite(addrstart, addrend, addrmirror, flags)
```

Desfaz o mapeamento dos acessos (por exemplo, faz o registro log do acesso como não mapeado) para uma determinada faixa do intervalo com espelho opcional e sinalização.

A instalação do mapa do dispositivo

```
space.install_device(addrstart, addrend, device, map, *unitmask*, *cswidth*)
```

Instala um endereço do dispositivo com um espaço de endereçamento num determinado espaço. Os argumentos unitmask, cswidth e flags são opcionais.

A instalação da visualização

```
space.install_view(addrstart, addrend, view)
space.install_view(addrstart, addrend, addrmirror, view)

view[0].install...
```

Instala uma visualização num determinado espaço. Isto só pode ser feito uma vez e em apenas um espaço e a visualização não deve ter sido configurada antes através da API do espaço de endereços. Uma vez instalada

a visualização pode ser selecionada através de indexação para chamar um método de mapeamento dinâmico sobre ela.

Uma visualização pode ser instalada numa variante de outra visualização sem problemas com a única restrição usual de uma única instalação.

12.10 O novo subsistema de disquete

12.10.1 Introdução

O novo subsistema de disquete visa emular o comportamento de disquetes e controladores de disquetes em nível baixo o suficiente a ponto de fazer com que as proteções também funcionem de forma transparente. O objetivo é alcançado ao seguir a configuração de um hardware real:

- uma classe de imagem de disquete que mantém na memória o estado magnético da superfície flexível e as suas características físicas.
- uma classe manipuladora de imagem fala com a classe de imagem de disquete visando simular o drive de disquete, fornecendo todos os sinais existentes num conector de disquete.
- dispositivos controladores que conversam com o manipulador de imagem e fornecem as interfaces de registo para o host que todos nós conhecemos e amamos.
- nas classes de manipulação de formato, lhes são dadas a tarefa de converter a origem e destino de forma neutra de uma imagem de disco físico para um estado de formato magnético do disco na memória, de forma que a classe gerenciadora do disquete possa geri-la.

12.10.2 O armazenamento de disquete para leigos

O disquete

O disquete é um disco que armazena as orientações magnéticas em sua superfície, dispostas numa série de círculos concêntricos chamado de faixas ou cilindros¹. As suas principais características são o seu tamanho que vai de um diâmetro em torno de 2.8 polegadas (63.5 milímetros) até 8 polegadas (200 milímetros), seu número de lados graváveis (1 ou 2) e sua resistividade magnética. A resistividade magnética indica o quão perto uma mudança na orientação magnética pode ocorrer e a informação mantida. Isso é um terço do que define o termo "densidade" que é usado com tanta frequência para disquetes (os outros dois são o tamanho da cabeça do drive de disquetes e a codificação a nível de bit (*bit-level encoding* no Inglês)).

As orientações magnéticas são sempre binárias, elas sempre apontam para um lado ou para o outro, não há nenhum estado intermediário. A sua direção pode estar na tangente da pista, na mesma direção, oposta a rotação ou no caso de uma gravação no sentido perpendicular, a direção é perpendicular (por isso o nome). A gravação no sentido perpendicular permite que os dados de gravação ocupem menos espaço permitindo uma maior densidade de gravação, porém chegou no final do tempo de vida da tecnologia. Os discos com 2.88 Mb e derivados dos disquetes como Zip Drives, etc; usavam gravação perpendicular. Para fins de emulação, a direção não importa, o que importa é o fato que duas orientações são possíveis. Além dessas orientações mais duas são possíveis: uma parte da trilha pode ser desmagnetizada (sem orientação) ou danificada (sem orientação ou não pode ser gravada).

Uma posição específica na rotação disco dispara um pulso de índice. Essa posição pode ser detectada através de um buraco na superfície (muito visível em disquetes 5.25 e 3 polegadas por exemplo) ou através de uma posição específica do centro de rotação (disquetes com 3.5 polegadas, talvez outros). Esse pulso de índice é usado para determinar o início da faixa, porém não é usado por todos os sistemas. Os disquetes mais antigos de 8 polegadas têm múltiplos buracos marcando o índice determinando o início dos setores (chamados de setor duro), no entanto um deles está numa posição diferente para ser reconhecido como um início de trilha, e os outros estão em posições fixas relativas à origem.

¹ O cilindro é um termo de disco rígido usado de forma inadequada para disquetes. Ele vem do fato que os discos rígidos são semelhantes aos disquetes, mas incluem uma série de discos empilhados com uma cabeça de leitura/gravação em cada um deles. As cabeças estão fisicamente ligadas e todas apontam para o mesmo círculo em cada disco num determinado momento, fazendo com que a área acessada pareça com um cilindro. Daí o nome. (Nota do tradutor)

Unidade de Disquete

Uma unidade de disquete é o aparelho que lê e grava um disquete. Inclui um conjunto capaz de girar o disco a uma velocidade fixa e uma ou duas cabeças magnéticas ligadas a um motor de posicionamento para acessar as trilhas.

A largura da cabeça e o tamanho do passo do motor de posicionamento determinam quantas trilhas estão escritas no disquete. O número total de trilhas varia entre 32 até 84 de acordo com o disquete e o drive, a trilha 0 ficando mais ao externo (mais longo) dos círculos concêntricos, e o maior com o menor círculo interno. Como resultado, as faixas com os números mais baixos têm a menor densidade física de orientação magnética, portanto, uma melhor confiabilidade. É por isso que estruturas importantes e/ou frequentemente alteradas, como o bloco de inicialização ou a tabela de alocação FAT, estão na trilha 0. É também aí que vem a terminologia "stepping in" para aumentar o número da faixa e "stepping out" para diminuí-lo. O número de faixas disponíveis é a segunda parte do que geralmente está por trás do termo "densidade".

Um sensor detecta quando a cabeça está na faixa 0 e o controlador não deve passar por ela. Além disso, bloqueios físicos impedem que a cabeça saia do alcance correto da pista. Alguns sistemas (Apple II, alguns C64) não levam em conta o sensor da trilha 0 fazendo com que a cabeça vá contra o limite físico do bloco, fazendo um ruído de impacto bem conhecido e eventualmente danificando o alinhamento da cabeça.

Além disso, alguns sistemas (Apple II e C64) têm acesso direto às fases do motor de posicionamento da cabeça, permitindo que a cabeça se posicione entre as pistas, no meio ou mesmo em posições intermediárias. Isso não era útil para escrever mais faixas, uma vez que a largura da cabeça não mudava, mas como a leitura confiável só era possível com a posição correta, ela era usada como proteção contra cópia por alguns sistemas.

O disco gira a uma velocidade fixa para uma determinada faixa. A velocidade mais comum é de 300 RPM para cada faixa, com 360 rpm encontrado para os disquetes de alta densidade com 5.25 polegadas e a maioria dos disquetes com 8 polegadas. A velocidade dos primeiros disquetes giravam em torno de 90 RPM ou até mesmo 150 RPM para um disquete de alta densidade num Amiga. Ter uma velocidade rotacional fixa para todo o disco é chamada de Velocidade Angular Constante (CAV em inglês) usada por quase todos ou Velocidade Angular Constante Zoneada (ZCAV em inglês, usado no C64), dependendo se a taxa de bits de leitura/gravação é constante ou depende da faixa. Alguns sistemas como Apple II e Mac variam a velocidade de rotação dependendo da faixa (algo como até 394 RPM) para terminar como uma Velocidade Linear Constante (*Constant Linear Velocity* ou CLV em Inglês). A ideia por trás do ZCAV/CLV é extrair mais bits da mídia mantendo o espaçamento mínimo entre transições de orientação magnética, oferecendo o melhor desempenho possível entre o espaço ocupado e a velocidade de transição da cabeça. Parece que a complexidade não foi considerada válida já que quase nenhum sistema faz.

Finalmente, após o disco girar e a cabeça estiver sob a posição adequada, a leitura correta da faixa acontece. A leitura é feita através de uma cabeça indutiva, que lhe dá a característica interessante de não ler a orientação magnética de forma direta, em vez disso, ser sensível o suficiente às inversões de orientação, chamadas de transições de fluxo. Esta detecção é fraca e pouco precisa, de modo que um amplificador com Ajuste de Ganho Automático (*Automatic Gain Control* ou AGC em Inglês) e um detector de pico são colocados de forma a trabalhar em conjunto com da cabeça para fornecer pulsos limpos. O AGC aumenta lentamente o nível de amplificação até que um sinal ultrapasse um limite pré determinado, em seguida ajusta seu ganho para que o dito sinal esteja estável num nível fixo dentro deste limite. Conforme a oscilação vai acontecendo o AGC entra em ação novamente. Isso faz com que o amplificador se calibre para os sinais lidos no disquete, desde que as transições de fluxo aconteçam com uma certa frequência. Numa zona muito longa, ocorre a captação de ruídos aleatórios do ambiente, fazendo com que a amplificação deste sinal ultrapasse o limite pré estabelecido, criando pulsos falsos onde não existem nenhum. Muito longa neste caso são aquelas que acontecem entre 16-20us sem nenhuma transição.

Isso significa que uma zona suficientemente longa com uma orientação magnética fixa ou nenhuma orientação (desmagnetizada ou danificada) será lida como uma série de pulsos aleatórios após um breve atraso. Isso é usado por proteções e é conhecido como "weak bits", que ao serem lidos os dados são diferente cada vez que são acessados.

Um segundo nível de filtragem ocorre após o detector de pico. Quando duas transições estão um pouco próximas (mas ainda acima do limiar da mídia), um efeito saltante acontece entre elas, dando dois pulsos muito próximos no meio, além dos dois pulsos normais. O drive de disquete consegue detectar quando os pulsos estão muito próximos e os elimina, deixando os pulsos normais novamente. Como resultado, se alguém escrever uma cadeia de pulsos de alta frequência para o disquete, eles serão lidos como um trem de pulsos muito próximos (fracos porque estão acima da tolerância da mídia, mas capturados pelo AGC de qualquer forma, apenas de forma pouco confiável) eles serão todos filtrados, dando uma grande quantidade de tempo sem qualquer pulso no sinal de saída. Isso é usado por algumas proteções uma vez que não é gravável usando o relógio normal do controlador.

A escrita é simétrica, com uma série de pulsos enviados que fazem a cabeça de gravação inverter a orientação do campo magnético cada vez que um pulso é recebido.

Então, para concluir, a unidade de disquete fornece insumos para disco de controle de rotação e a posição da cabeça (assim como a escolha quando é de dupla-face), os dados são enviados de duas maneiras como um trem de pulsos que representam inversões de orientação magnética. O valor absoluto da orientação em si nunca é conhecido.

Controlador de Disquete

A tarefa do controlador de disquete é transformar a comunicação da unidade de disquete em algo a CPU principal possa compreender. O nível de compatibilidade entre um controlador e outro varia aos extremos, vai de praticamente nada nos Apple II e C64, com alguma coisa no Amiga e para completar Circuitos Integrados da *Western Digital*, família **uPD765**). Funções comuns incluem a seleção da unidade, controle do motor, busca das trilhas e claro a leitura e gravação de dados. Destes somente os dois últimos precisam ser descritos pois o resto é óbvio.

Os dados são estruturados em dois níveis: como bits individuais (meio byte ou bytes) que são codificados na superfície e como estes são agrupados em setores endereçados individualmente. Existem dois padrões para eles chamados *Frequency Modulation* (sigla FM no inglês) e *Modified Frequency Modulation* (sigla MFM no inglês), além de uma série de outros sistemas e suas variantes. Além disso, alguns sistemas tais como o Amiga usa um padrão de codificação *bit-level encoding* (MFM) com uma organização de nível setorial local.

12.10.3 Codificação a nível de bit

Organização Celular

Todos os controladores de disquetes, até os mais esquisitos como o Apple II, começa dividindo a pista em células de igual tamanho. Eles são seções angulares no meio de onde uma inversão de orientação magnética pode estar presente. Do ponto de vista do hardware, as células são vistas como durações que combinada com a rotação do disquete determina a seção. Por exemplo o tamanho padrão de uma célula MFM para um disquete de dupla densidade com 3 polegadas é de 2us, também combinada com uma velocidade de rotação com 300 RPM, dá um tamanho angular de 1/100.000 por volta. Outra maneira de dizer a mesma coisa é que há 100K (cem mil) células numa pista de dupla densidade de um disquete de 3 polegadas.

Em cada célula pode ou não haver uma transição de orientação magnética, por exemplo, uma pulsação vindo de uma leitura ou ir para a escrita da unidade de disquete. Uma célula com um pulso é tradicionalmente conhecida como '1', e um sem '0'. Embora, duas restrições aplicam-se para o conteúdo da célula. Primeiro, os pulsos não devem ser muito juntos ou eles irão causar um borrão um ao outro, e/ou serão filtrados.

O limite é ligeiramente melhor do que 1/50.000 de uma volta para disquete com densidade simples e dupla, metade disso para disquetes de alta densidade e metade disso novamente para disquetes com densidade estendida (ED) com gravação perpendicular. Segundo, eles não devem ser muito longe um do outro, ou seja o AGC vai ficar instável e introduzir pulsos fantasmas ou o controlador vai perder sincronização e obter um

sincronismo errado sobre as células durante a leitura. Para via de regra geral, é melhor não ter mais de 3 células '0' consecutivas.

Certas proteções usam isso para tornar os formatos não reconhecíveis pelo controlador do sistema, quebrando a regra de três zeros ou brincar com as durações e tamanhos das células.

Bit encoding é a arte de transformar dados brutos numa célula de configuração 0/1 respeitando os dois limites.

Codificação FM

O primeiro método de codificação desenvolvido para disquetes é chamado de Frequência Modulada (*Frequency Modulation* ou FM), o tamanho da célula é definida um pouco além do limite físico, como 4us por exemplo. Isso significa que é possível ter '1' célula consecutiva de confiança. Cada bit é codificado em duas células:

- a primeira célula, chamada o clock bit é '1'
- a segunda célula, chamada de data bit, é o bit em si

Uma vez que todas as outras células seja pelo menos '1' não há nenhum risco de ir além de três zeros.

O nome Frequência Modulada simplesmente deriva do fato de que um 0 é codificado com um período de trem de pulsos em 125 Khz enquanto um 1 são dois períodos do trem de pulso em 250 Khz.

Codificação MFM

A codificação de FM foi substituída pela codificação *Modified Frequency Modulation (MFM)*, que pode empilhar exatamente o dobro de dados na mesma superfície, daí seu outro nome de "dupla densidade". O tamanho da célula é definido com um pouco mais de metade do limite físico, 2us normalmente. A restrição significa que duas células '1' devem ser separadas por pelo menos uma célula '0'. Cada bit é novamente codificado em duas células:

- a primeira célula, chamada de clock bit, é '1' se ambos os bits de dados anteriores e atuais forem 0, então será '0'
- a segunda célula, chamada de data bit, é o bit em si

A regra de espaço mínimo é respeitada uma vez que um '1' de clock bit é, por definição, rodeado por dois '0' de data bits e um '1' data bit é rodeado por dois '0' clock bits. A maior cadeia de célula 0 possível é quando ao codificar 101 que retorna x10001, respeitando o limite máximo de três zeros.

Codificação GCR

As codificações *Group Coded Recording*, ou GCR, são uma classe de codificações onde cadeias de bits com pelo menos tamanho de meio byte ou 4 bit são codificadas num determinado fluxo de células dado por uma tabela. Ele foi usado particularmente pelo Apple II, o Mac e o C64, e cada sistema tem sua própria tabela ou tabelas.

Outras codificações

Existem outras codificações como o M2FM, mas elas são muito raras e específicas para um determinado sistema.

Lendo os dados codificados

Escrever dados codificados é fácil, basta de um relógio na frequência apropriada e enviar ou não uma cadeia de pulsos ao redor do relógio. A diversão está em ler esses dados. As células são uma construção lógica e não uma entidade física mensurável.

As velocidades rotacionais variam ao redor dos valores definidos ($\pm 2\%$ não é raro) e perturbações locais (turbulência do ar, distância da superfície...) no geral, tornam a velocidade instantânea muito variável. Portanto, para extrair o fluxo de valores da célula, o controlador deve sincronizar dinamicamente com o trem de pulso que a cabeça do disquete seleciona. O princípio é simples: uma janela de duração do tamanho da célula é construída dentro da qual a presença de pelo menos um pulso indica que a célula é um '1' e a ausência de qualquer um '0'. Depois de chegar ao final da janela, a hora de início é movida apropriadamente para tentar manter o pulso observado no meio exato dessa janela. Isso permite corrigir a fase em cada célula '1', fazendo a sincronização funcionar se a velocidade de rotação não estiver muito fora.

Gerações subsequentes de controladores usaram um *Phase Locked Loop* (PLL) que varia a duração da fase e da janela para se adaptar melhor as velocidades erradas de rotação, geralmente com uma tolerância de $\pm 15\%$.

Depois que o fluxo de dados da célula é extraído, a decodificação depende da codificação. No caso de FM e MFM, a única questão é reconhecer os bits de dados dos bits de clock, enquanto no GCR a posição inicial do primeiro grupo deve ser encontrada. O segundo nível de sincronização é tratado num nível mais alto usando padrões não encontrados num fluxo normal.

12.10.4 Organização de nível no setor

Os disquetes foram concebidos para a leitura e gravação com acesso aleatório para blocos de dados de tamanhos razoáveis. Permite a seleção de faixas para um primeiro nível de acesso aleatório e dimensionamento, mas os 6 K de uma faixa de densidade dupla seria muito grande para ser lido por um bloco. 256/512 bytes são considerados um valor mais apropriado. Para o efeito, dados numa faixa são organizados como uma série de (cabeçalho do setor, dados do setor) pares onde o cabeçalho do setor indicam informações importantes, como o número do setor, tamanho, e os dados do setor que contém os dados. Os setores tem que ser quebrados em duas partes, porque enquanto a leitura é fácil, é lido o cabeçalho, depois os dados sem assim for necessário, para escrever requer a leitura do cabeçalho para encontrar o lugar correto, para só então ligar a cabeça de escrita para os dados. A escrita inicial não é instantânea e a fase não está perfeitamente alinhada com a cabeça de leitura, portanto, um espaço para a sincronização é necessária entre o cabeçalho e dados.

Somando a isso, em algum lugar no setor do cabeçalho e no sector dos dados, geralmente são adicionados algum tipo de checksum para permitir a verificação da integridade destes dados.

O FM e o MFM (nem sempre utilizaram) métodos de layout padrão do setor.

Layout do setor de FM

O layout padrão em FM de trilha/setor para um "PC" é assim:

- Uma quantidade de 0xff codificados em FM (40 geralmente)
- 6 0x00 codificados em FM (dando uma cadeia de pulso estável em 125 KHz)
- Um fluxo 1111011101111010 com 16 células (f77a, clock 0xd7, data 0xfc)
- Uma quantidade de 0xff codificados em FM (geralmente 26, muito volátil)

Então para cada setor: - 6 0x00 codificados em FM (dando uma cadeia de pulso estável em 125 KHz)

- Um fluxo 1111010101111110 com 16 células (f57a, clock 0xc7, data 0xfe)

Cabeçalho do setor, faixa codificada em FM, cabeça, setor, código de tamanho e dois bytes de crc por exemplo

- 11 0xff codificados em FM
- 6 0x00 codificados em FM (dando uma cadeia de pulso estável em 125 KHz)
- Um fluxo 1111010101101111 com 16 células (f56f, clock 0xc7, data 0xfb)
- Dados do setor codificado em FM seguido por dois bytes CRC
- Uma quantidade de 0xff codificados em FM (geralmente 48, muito volátil)

A trilha é terminada com um fluxo de células '1'.

Os trens de pulsos com 125 KHz são utilizados para travar o PLL ao sinal corretamente. Os fluxos específicos com 16 células permitem distinguir entre o clock e os data bits fornecendo um arranjo que não é comum ocorrer em dados codificados em FM. No cabeçalho do setor da trilha, os números começam em 0, cabeças são 0/1 dependendo do tamanho, os números do setor geralmente começam em 1 e o tamanho do código é 0 para 128 bytes, 1 para 256, 2 para 512, etc.

O CRC é uma verificação de redundância cíclica dos bits de dados, começando com uma marca logo após o trem de pulso usando o polinômio 0x11021.

Os controladores com base na Western Digital geralmente livram-se de tudo deixando alguns 0xff no primeiro setor e permitem um melhor uso do espaço como resultado.

Layout do setor de FM

O layout padrão de trilha/setor para MFM num "PC" é assim:

- Uma quantidade de 0x4e codificados em MFM (80 geralmente)
- 12 0x00 codificados em FM (dando uma cadeia de pulso estável em 125 KHz)
- Um fluxo 0101001000100100 com 16 células (5224, clock 0x14, data 0xc2)
- O valor 0xfc codificado em MFM
- Uma quantidade de 0x4e codificados em MFM (geralmente 50, muito volátil)

Então para cada setor:

- 12 0x00 codificados em FM (dando uma cadeia de pulso estável em 125 KHz)
- Três vezes um fluxo 0100010010001001 com 16 células (5224, clock 0x14, data 0xc2)
- Cabeçalho do setor, 0xfe codificado em MFM, trilha, cabeça, setor, código de tamanho e dois bytes de CRC por exemplo

- 22 0x4e codificado em MFM
- 12 0x00 codificados em MFM (dando uma cadeia de pulso estável em 125 KHz)
- Três vezes um fluxo 0100010010001001 com 16 células (5224, clock 0x14, data 0xc2)
- 0xfb codificado em MFM, dados do setor seguido por dois bytes CRC
- Uma quantidade de 0x4e codificados em MFM (geralmente 84, muito volátil)

A trilha é finalizada com um fluxo 0x4e codificado em MFM.

Os trens de pulsos com 125 KHz são utilizados para travar o PLL ao sinal de forma correta. A célula com o arranjo 4489 não aparece numa codificação de dados MFM normal e é usada para a separação de clock/dados.

Já para FM, os controladores com base Western Digital geralmente livram-se de tudo menos alguns 0x4e antes do primeiro setor e permite um melhor uso do espaço como resultado.

Formatação e escrita

Para ser utilizável, um disquete deve ter os cabeçalhos do setor e os dados padrão escritos em cada trilha. O controlador começa a escrita em um determinado lugar, muitas vezes pelo pulso de índice, mas em alguns sistemas sempre que o comando é enviado ele grava até que seja feita uma volta completa. Isso é conhecido como formatação de disquete. No ponto onde a escrita termina, há uma perda de sincronização uma vez que não há nenhuma chance do relógio de fluxo da célula terminar a escrita de forma correta. Esta mudança de fase brutal é chamada uma gravação da tala, especificamente a faixa escrever da tala. É o ponto onde a escrita deve começar caso queira uma cópia raw da faixa para um novo disquete.

Igualmente duas junções de gravação são criadas quando um setor é escrito no início e no final da parte do bloco de dados. Não deveria acontecer num disco masterizado, mesmo que haja algumas raras exceções.

12.10.5 A nova implementação

Representação do disquete

O conteúdo do disquete é representado pela classe *floppy_image*. Contém informações do tipo de mídia e uma representação do estado magnético da superfície.

O tipo de mídia é dividido em duas partes. A primeira metade indica o fator de forma física, ou seja, todas as mídias com esse fator podem ser fisicamente inseridas num leitor que puder manuseá-lo. A segunda metade indicam as variantes que são geralmente detectáveis pelo leitor, tais como a densidade e o número de lados.

A trilha de dados consiste numa série valores lsb primários em 32-bits representando as células magnéticas. Os bits 0-27 indicam a posição absoluta do início da célula (não o tamanho) e os bits 28-31 indicam os tipos. Os tipos podem ser:

- 0, MG_A -> Orientação Magnética A
- 1, MG_B -> Orientação Magnética B
- 2, MG_N -> Zona não magnetizada (neutra)
- 3, MG_D -> Zona danificada, lê como neutra mas não pode ser alterada por escrita

A posição está em unidades angulares de 1/200,000,000 de uma volta. Corresponde a um nanossegundo quando a unidade gira a 300 RPM.

A última posição implícita da célula é 200,000,000.

As trilhas não formatadas são codificadas com um tamanho zero.

A informação de "junção de trilha" indica onde começar a escrever caso tente reescrever um disco físico com dados. Alguns formatos de preservação codificam essa informação, ela é adivinhada para os outros. A função de gravação da trilha do fdcs deve configurá-la. A representação é a posição angular relativa ao índice.

12.10.6 Convertendo de e para uma representação interna

Classe e interface

Precisamos ser capazes de converter para a representação interna os formatos de dados contidos no disquete. Isso é feito através de classes derivadas de *floppy_image_format_t*. A interface a ser implementada deve conter:

- **name()** fornece um nome abreviado ao formato no disco
- **description()** fornece uma breve descrição do formato
- **extensions()** fornece uma lista separada por vírgula das extensões dos nomes de arquivos encontrados para esse formato
- **supports_save()** retorna verdadeiro se houver compatibilidade com o formato externo
- **identify(file, form factor)** retorna uma pontuação entre 0-100 para o arquivo que for daquele formato:
 - 0 = esse formato não
 - 100 = provavelmente esse formato
 - 50 = formato identificado apenas pelo tamanho do arquivo
- **load(file, form factor, floppy_image)** carrega uma imagem e a converte para a representação interna
- **save(file, floppy_image)** (se implementado) convertido da representação interna e salva numa imagem

Todos estes métodos são previstos para serem sem estado.

Métodos auxiliares de conversão

Vários métodos são fornecidos para simplificar a gravação das classes do conversor.

Métodos de conversão orientados à leitura

```
generate_track_from_bitstream(track number,
    head number,
    UINT8 *cell stream,
    int cell count,
    floppy image)
```

Obtém um fluxo de tipos de células (0/1), primeiro o MSB, converte-o para o formato interno e armazena-o na devida trilha e cabeça de uma determinada imagem.

```
generate_track_from_levels(track number,
    head number,
    UINT32 *cell levels,
    int cell count,
    splice position,
    floppy image)
```

Pega uma variante do formato interno onde cada valor representa uma célula, a parte da posição dos valores é o tamanho da célula e a parte do nível é MG_0, MG_1 para os tipos de células normais, MG_N, MG_D para as células não formatadas ou danificadas e MG_W para os bits mais fracos no estilo *Dungeon-Master*. Converte para o formato interno. Os tamanhos são normalizados para que eles tenham uma volta completa no total.

normalize_times(UINT32 *levels,
int level_count)

Pega um buffer de formato interno onde a parte da posição representa o ângulo até a próxima mudança e o transforma num fluxo normal de posição, primeiro garantindo que o tamanho total seja normalizado para uma volta completa.

Métodos de conversão orientados a gravação

generate_bitstream_from_track(track number,
head number,
base cell size,
UINT8 *cell stream,
int &cell_stream_size,
floppy image)

Extraí um fluxo da célula 0/1 do formato interno usando uma configuração PPL com um tamanho de célula inicial definida para 'base cell size' e uma tolerância de +/- 25%.

```
struct desc_xs { int track, head, size; const UINT8 *data }
extract_sectors_from_bitstream_mfm_pc(...)
extract_sectors_from_bitstream_fm_pc(const UINT8 *cell stream,
int cell_stream_size,
desc_xs *sectors,
UINT8 *sectdata,
int sectdata_size)
```

Extraí os setores padrão MFM ou FM de um fluxo de células regeneradas. Os setores devem apontar para uma matriz com 256 ofdesc_xs.

Um setor existente é reconhecível por ter -> dados não nulos. Os dados do setor são escritos em sectdata até os bytes sectdata_size.

get_geometry_mfm_pc(...)
get_geometry_fm_pc(floppy image,
base cell size,
int &track_count,
int &head_count,
int §or_count)

Extraí a geometria (cabeças, trilhas, setores) de uma imagem de disquete tipo pc, verificando a trilha 20.

```
get_track_data_mfm_pc(...)
get_track_data_fm_pc(track number,
    head number,
    floppy image,
    base cell size,
    sector size,
    sector count,
    UINT8 *sector data)
```

Extraí o que seria obtido ao ler na ordem dos setores '*sector size*' do número 1 para o contador do setor e registra o resultado no setor de dados.

12.10.7 Unidade de Disquete

A classe *floppy_image_interface* simula a unidade de disquete. Isso inclui uma série de sinais de controle, leitura e escrita. Os sinais de controle de alterações devem ser sincronizadas, disparo do temporizador para assegurar que a hora atual seja a mesma para todos os dispositivos, por exemplo.

Sinais de controle

Devido à maneira de como estão ligados na CPUs (diretamente numa porta I/O por exemplo), o controlador de sinais trabalha com valores físicos em vez de lógicos. Em geral, o 0 significa ativo e 1 inativo. Alguns sinais têm também um retorno de chamada associado a eles quando mudam.

mon_w(state) / mon_r()

Sinal para ligar o motor, gira no 0

idx_r() / setup_index_pulse_cb(cb)

Sinal de indexação, vai a 0 no início da pista por aproximadamente 2ms. O retorno de chamada é sincronizado. Só acontece quando um disco está em funcionamento e o motor está funcionando.

ready_r() / setup_ready_cb(cb)

Sinal de pronto (*Ready*), vai a 1 quando o disco é removido ou o motor é parado. Vai a 0 depois de dois pulsos indexados.

wpt_r() / setup_wpt_cb(cb)

Sinal de proteção contra gravação (1 = somente leitura). O retorno de chamada não é sincronizado.

dskchg_r()

Sinal de mudança de disco, vai a 1 quando um disco é alterado, vai a 0 para a mudança de trilha.

dir_w(dir)

Seleciona a direção do passo da trilha (1 = fora = diminui o número da trilha).

stp_w(state)

Sinal de passo, move-se por uma trilha na transição 1->0.

trk00_r()

Sensor de trilha 0, retorna 0 quando estiver na trilha 0

ss_w(ss) / ss_r()

Seleciona um lado

12.10.8 Interface de leitura e gravação

A interface de leitura e gravação é projetada para trabalhar de forma assíncrona, de maneira independentemente da hora atual, por exemplo.

12.11 O novo subsistema SCSI

12.11.1 Introdução

O subsistema **nscsi** foi criado para permitir que uma implementação fique o mais próximo possível do hardware físico real, (na esperança de) facilitar a implementação de novos CIs controladores a partir de documentações.

12.11.2 Estrutura global

O SCSI paralelo é construído em torno de um barramento simétrico ao qual vários dispositivos estão conectados. O barramento é composto de 9 linhas de controle (no momento, as versões posteriores do SCSI podem ter mais) e até 32 linhas de dados (mas os chips atualmente implementados suportam apenas 8). Todas as linhas são coletores abertos, o que significa que um ou vários chips conectam a linha ao terra e a linha, óbvio, vai para o terra ou nenhum CI conduz nada e a linha continua no Vcc. Além disso, o barramento usa um lógica invertida, significa que o sinal ao ser aterrado equivale a 1. Os controladores SCSI tradicionalmente funcionam em níveis lógicos e não físicos, então o subsistema **nscsi** também funciona em níveis lógicos, assim todas as suas saídas para os dispositivos são lógicos.

Estruturalmente, a implementação é feita em torno de duas classes principais:

- **nscsi_bus_devices** representa o barramento
- **nscsi_device** representa um dispositivo individual

Um dispositivo só se comunica com o barramento e o barramento cuida da manipulação transparente da descoberta e a comunicação do dispositivo. Além disso a classe **nscsi_full_device** propõe um dispositivo SCSI com o protocolo SCSI implementado, facilitando a criação de dispositivos SCSI genéricos como se fossem discos rígidos ou leitores de CD-ROM.

12.11.3 Conectando um barramento SCSI num driver

O subsistema **nscsi** aproveita as interfaces de slot e a nomenclatura do dispositivo para permitir uma implementação e configuração de barramento de forma simples.

Primeiro é necessário criar uma lista de dispositivos aceitáveis para conectar ao barramento. Isso geralmente inclui **cdrom**, **hasrdisk** e o CI do controlador. Por exemplo:

```
static SLOT_INTERFACE_START( next_scsi_devices )
    SLOT_INTERFACE("cdrom", NSCSI_CDROM)
    SLOT_INTERFACE("harddisk", NSCSI_HARDDISK)
    SLOT_INTERFACE_INTERNAL("ncr5390", NCR5390)
SLOT_INTERFACE_END
```

A interface **_INTERNAL** indica um dispositivo que não é selecionável pelo usuário, o que é útil para o controlador.

Então na configuração do sistema (ou numa configuração de fragmento) primeiro é necessário adicionar o barramento e em seguida os dispositivos (potenciais) como dispositivos de sub-dispositivos do barramento com o SCSI ID como seu nome, exemplo:

```
MCFG_NSCSI_BUS_ADD("scsibus")
MCFG_NSCSI_ADD("scsibus:0", next_scsi_devices, "cdrom", 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:1", next_scsi_devices, "harddisk", 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:2", next_scsi_devices, 0, 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:3", next_scsi_devices, 0, 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:4", next_scsi_devices, 0, 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:5", next_scsi_devices, 0, 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:6", next_scsi_devices, 0, 0, 0, 0, false)
MCFG_NSCSI_ADD("scsibus:7", next_scsi_devices, "ncr5390", 0, &next_ncr5390_interface, 10000000,
true)
```

Essa configuração coloca como um leitor de CD-ROM padrão no SCSI ID 0 e um disco rígido com SCSI ID 1 forçando o controlador no ID 7. Os parâmetros para adição são:

- device tag, composto por bus-tag:scsi-id
- uma lista com os dispositivos aceitos
- um dispositivo conforme disposto na lista, se um já não estiver por padrão
- a configuração do dispositivo de entrada, caso haja (e geralmente não há)
- a estrutura de configuração do dispositivo, geralmente usado apenas para o controlador
- a frequência, geralmente usado apenas pelo controlador

O nome completo do dispositivo, para fins de mapeamento seria bus-tag:scsi-id:device-type, scsibus:7:ncr5390 para o nosso controlador aqui.

12.11.4 Criando um novo dispositivo SCSI usando `nscsi_device`

A classe base "`nscsi_device`" deve ser usado para os CIs do controlador SCSI. A classe fornece três variáveis e um método:

- A primeira variável, `scsi_bus`, é um ponteiro para o `nscsi_bus_device`.
- A segunda, `scsi_refid`, é uma referência opaca para passar algumas operação ao barramento.
- Finalmente, o `scsi_id` dá um SCSI ID individual por tag de dispositivo. É escrito uma vez na inicialização e nunca é lido ou gravado depois, o dispositivo pode fazer o que quiser com o valor ou a variável.
- O método virtual `scsi_ctrl_changed` é chamado quando for assistir as mudanças das linhas de controle. É através do barramento que são definidas quais as linhas serão monitoradas.

Para acessar as linhas existe uma proposta com cinco métodos:

- `ctrl_r()` e `data_r()` são os métodos de leitura. Os controles de bits são definidos dentro do enum¹ `S_*` de `nscsi_device`.
- Os três bits abaixo (`INP`, `CTL` e `MSG`) são configurações para que o "masking" com 7(`S_PHASE_MASK`) retorne os números para as fases, que também estão disponíveis com o enum `S_PHASE_*`.
- A escrita nas linhas de dados é feito com `data_w(scsi_refid, value)`.

¹ Assumo que o termo abreviado "*enum*" seja um enumerador. (Nota do tradutor)

- A escrita nas linhas de controle é feito com `ctrl_w(scsi_refid, value, mask-of-lines-to-change)`. Para alterar todas as linhas de controle com uma chamada use a máscara **S_ALL**.

Claro que o que é lido é a lógica de tudo o que é conduzido por todos os dispositivos.

- Finalmente, o método `ctrl_wait_w(scsi_id, value, mask of wait lines to change)`, permite selecionar quais as linhas de controle que são monitoradas. A máscara de monitoramento é individual para cada dispositivo, o **scsi_ctrl_changed** é chamado sempre que uma linha de controle da máscara for alterado, devido a uma ação de um outro dispositivo (não em si, para evitar uma recursão irritante e um tanto inútil).

A implementação do controle é apenas uma questão de seguir o estado descritivos dos sistemas, pelo menos se eles estiverem disponíveis. A única parte não descrita é a arbitragem/seleção que está documentada na norma do SCSI. Para um iniciador (o que é que o controlador sempre é essencialmente), funciona assim:

- espera o barramento ficar ocioso
- garante em qual número o seu **scsi_id** está na linha de dados ($1 \ll \text{scsi_id}$)
- espera o tempo de atribuição
- verifica se as linhas de dados ativas com o número maior é a sua
 - caso não seja, a atribuição é perdida, pare a condução de tudo e reinicie
- garante a linha selecionada (nesse ponto o barramento é seu)
- espera um pouco
- mantém a sua linha de dados garantida, garante que o número da linha de dados é o SCSI ID de destino
- espera um pouco
- garante que caso a linha **atn** seja necessária, retorne como sinal ocupado
- espera que o sinal ocupado seja garantido ou que acabe o tempo limite (timeout)
 - O tempo limite significa que ninguém está respondendo naquele ID, desocupe tudo e pare
- aguarda por um curto período até o **de-skewing**
- desocupa o barramento de dados e seleciona uma linha
- espera mais um pouco

E tudo pronto, a conexão com o dispositivo de destino está feita até que o alvo desocupe a linha ocupada, seja porque foi requisitado ou apenas para te aborrecer. O **de-assert** (desocupar) é chamado de desconexão.

O **ncr5390** é um exemplo de como usar um estado do sistema com dois níveis de estado para lidar com todos os eventos.

12.11.5 Criando um novo dispositivo SCSI usando o `nscsi_full_device`

A classe base "**nscsi_full_device**" é usada para criar dispositivos SCSI HLE-d destinados para uso genérico, como discos rígidos, CD-ROMs, scanners talvez, etc. A classe fornece a manipulação de protocolo SCSI, deixando somente a manipulação de comando e (opcionalmente) o tratamento de mensagens para a implementação.

A classe atualmente suporta apenas dispositivos de destino.

O primeiro método para implementar é **scsi_command()**. Esse método é chamado quando um comando chegar por completo. O comando está disponível em `scsi_cmdbuf[]` e seu comprimento fica em `scsi_cmdsize`

(porém o comprimento em geral é inútil ao primeiro byte de comando dado). A matriz de 4096-bytes `scsi_cmdbuf` pode então ser modificada livremente.

Em `scsi_command()`, o dispositivo pode lidar com o comando ou passá-lo com `nscsi_full_device::scsi_command()`.

Para lidar com o comando, vários métodos estão disponíveis:

- `get_lun(lua set in command)` lhe dará o LUN a ser trabalhado (o **in-command** um pode ser substituído por um nível de mensagem um).
- `bad_lun()` respostas para o host que o LUN específico não tiver suporte.
- `scsi_data_in(buffer id, size)` envia bytes com tamanho vindo da memória intermédia **buffer-id**
- `scsi_data_in(buffer id, size)` recebe bytes com o tamanho para a memória intermédia **buffer-id**
- `scsi_status_complete(status)` termina o comando com um determinado status.
- `sense(deferred, key)` prepara o senso da memória intermédia para um comando subsequente de solicitação, que é útil ao retornar um status de verificação da condição.

Os comandos `scsi_data_*` e `scsi_status_complete` são enfileirados, o manipulador de comandos deve chamá-los todos sem tempo de espera.

O **buffer-id** identifica a memória intermediária. 0 também conhecido como **SBUF_MAIN**, direciona a memória intermédia `scsi_cmdbuf`. Os outros valores aceitáveis são 2 ou mais. 2+ ids são manipulados pelo método `scsi_get_data` para leitura e `scsi_put_data` para gravação.

`UINT8 device::scsi_get_data(int id, int pos)` deve retornar o id da posição do byte na memória intermediária, chamando em `nscsi_full_device` por `id < 2`.

`void device::scsi_put_data(int id, int pos, UINT8 data)` deve escrever o id da posição do byte na memória intermediária, chamando em `nscsi_full_device` por `id < 2`.

O `scsi_get_data` e o `scsi_put_data` devem fazer as leituras e gravações externas quando for necessário.

O dispositivo também pode sobrescrever o `scsi_message` para lidar com mensagens SCSI diferentes daquelas tratadas de forma genérica e também pode substituir alguns dos tempos (mas muitos deles não são usados, cuidado).

Para facilitar as coisas uma certa quantidade de "enums" é definida:

- O enum `SS_*` dá retornos de status (como `SS_GOOD` para todos que em condições boas).
- O enum `SC_*` fornece os comandos SCSI.
- O enum `SM_*` fornece as mensagens SCSI, com exceção do identificador (que é 80-ff, realmente não se encaixa num enum).

12.11.6 O que falta no `scsi_full_device`

- **Suporte ao iniciador** Nesse momento, não temos nenhum dispositivo iniciador para o HLE.
- **Delays** Um comando `scsi_delay` ajudaria a dar tempos (*timings*) mais realistas, particularmente ao leitor de CD-ROM.
- **Operações desconectadas** Primeiro exigiria atrasos e além disso, um sistema operacional emulado que pudesse manipulá-lo.
- **Operação ampla em 16-bits** Precisa de um SO e de um iniciador que possam manipulá-lo.

12.11.7 O que falta no ncr5390 (e provavelmente em outros controladores futuros)

- **A detecção de um barramento livre** No momento, o barramento é considerado livre caso o controlador não esteja ocupado, o que é verdade. Isso pode mudar uma vez que a operação de desconexão esteja em ação.
- **Comandos alvo** Ainda não são emulados ainda (vs. HLE).

12.12 A implementação da nova família 6502

12.12.1 Introdução

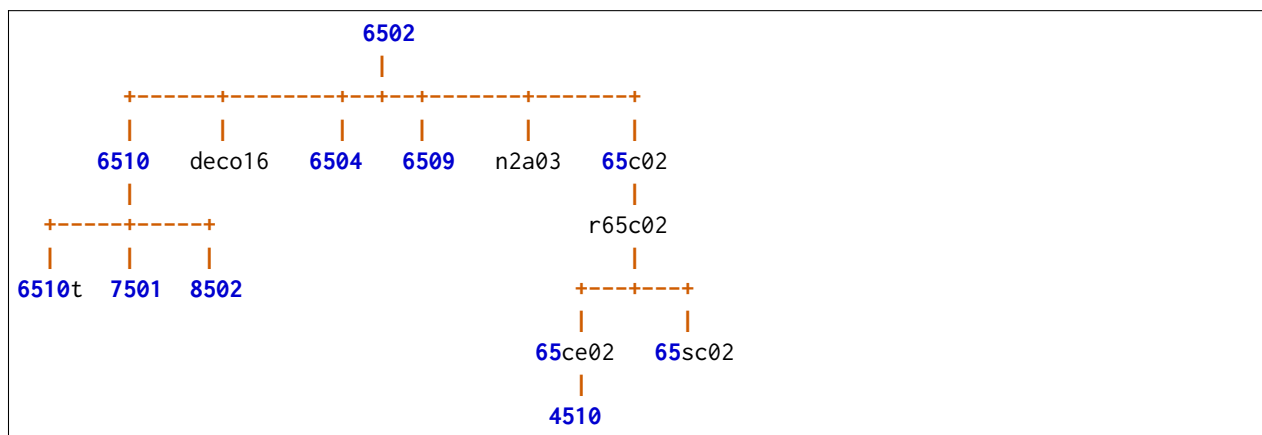
A implementação da nova família 6502 foi criada de maneira que as suas sub-instruções sejam observáveis de maneira precisa. Foi projetado visando 3 coisas:

- cada ciclo do barramento deve acontecer no exato momento que aconteceria numa CPU real assim como cada acesso.
- as instruções podem ser interrompidas a qualquer momento e depois reiniciado deste ponto de forma transparente
- para fins de emulação, as instruções podem ser interrompidas mesmo de dentro de um manipulador de memória para a contenção/espera do barramento.

O Ponto 1 foi garantido através de bi-simulação do *perfect6502* a nível de gate. O Ponto 2 foi garantido estruturalmente através de um gerador de código que será explicado com mais detalhes na seção 8. O Ponto 2 ainda não está pronto devido a falta de suporte nos subsistemas de memória, no entanto a seção 9 mostra como isso será tratado.

12.12.2 A família 6502

A família do MOS 6502 tem sido grande e produtiva. Existe um grande número de variantes, tamanhos de barramentos variados, I/O e até mesmo opcodes. Alguns coadjuvantes (g65c816, hu6280) até existem e estão perdidos em algum lugar dentro do código-fonte do MAME. A classe hierárquica final ficou assim:



O 6510 adiciona 8 bits na porta I/O, com o 6510t, o 7501 e o 8502 são variantes, compatíveis entre si a nível de software com uma quantidade de pinos diferente (quantidade de I/O), processo da die (NMOS, HNMOS, etc.) e suporte a clock.

O deco16 é uma variante do Deco, com um pequeno número de instruções adicionais ainda não compreendidas e alguns I/O.

O 6504 é uma versão reduzida de pinos e barramento de endereços.

O 6509 adiciona um suporte interno para paginação.

O n2a03 é uma variante NES com a bandeira D desativada e uma funcionalidade de som integrada.

O 65c02 é a primeira variante CMOS com algumas instruções adicionais, algumas correções e a maioria das instruções não documentadas se transformaram em *nops*. A variante R (*Rockwell*, mas eventualmente produzida pela WDC também dentre outras) adiciona várias instruções *bitwise* e também *stp* e *wai*. A variante

SC, usada pelo console portátil Lynx, parece idêntica à variante R. O 'S' provavelmente indica um processo estático de células de memória RAM, permitindo total controle de clock *DC-to-max*.

O 65ce02 é a evolução final do ISA nesta hierarquia, com instruções adicionais, registros e remoções de muitos acessos inertes que desacelerava o 6502 original em pelo menos 25%. O 4510 é o 65ce02 com suporte a MMU e GPIO integrados.

12.12.3 O uso das classes

Todas as CPUs são dispositivos de CPU modernos com toda a interação normal junto com a infraestrutura do dispositivo. Para incluir uma destas CPUs no seu driver, é necessário incluir "**CPU/m6502/<CPU>.h**" e então fazer um **MCFG_CPU_ADD("tag", <CPU>, clock)**.

Os callbacks da porta I/O das variantes do 6510 são configuradas através de:

```
MCFG_<CPU>_PORT_CALLBACKS(READ8(type, read_method), WRITE8(type, write_method))
```

E as linhas das máscaras **pullup** e **floating** são fornecidas através de:

```
MCFG_<CPU>_PORT_PULLS(pullups, floating)
```

Para ver todos os acessos de barramento nos manipuladores de memória, é necessário desativar os acessos através do mapa direto (ao custo de um processamento extra de CPU, é claro) com:

```
MCFG_M6502_DISABLE_DIRECT()
```

Nesse caso, o suporte à descritografia transparente também é desabilitada, tudo passa através de chamadas comuns de leitura/gravação no mapa de memória. O estado da linha de sincronização é dado pelo método da CPU **get_sync()**, possibilitando a implementação da descritografia no manipulador.

A cada dispositivo executável, o método de CPU **total_cycles()** dá o tempo atual em ciclos desde o início do sistema do ponto de vista da CPU. Ou em outras palavras, o que normalmente é chamado o número de ciclo da CPU quando alguém fala sobre contenção do barramento ou a espera do estado. A chamada é projetada para ser rápida (sem sincronização ampla do sistema, sem apelo à **machine.time()**) e é preciso. A quantidade de ciclos para cada acesso é exata a nível de sub-instruções.

A linha especial do nomap 4510 é acessível usando **get_nomap()**.

Além destes detalhes específicos, estas são classes normais de CPU.

12.12.4 Estrutura geral das emulações

Cada variante é emulada através de 4 arquivos:

- <CPU>.h = cabeçalho para a classe de CPU
- <CPU>.c = implementação para a maioria das classes de CPU
- d<CPU>.lst = tabelas de despacho para a CPU
- o<CPU>.lst = implementações opcode para a CPU

As duas últimas são opcionais. Eles são usados para gerar um arquivo **<CPU>.inc** no diretório de objeto que está incluso no arquivo fonte .c.

A classe deve incluir, no mínimo, um construtor e um enum, captando as identificações de linha de entrada correta. Veja o *m65sc02* para um exemplo minimalista. O cabeçalho também pode incluir macros de configuração específica (consulte o *m8502*) e também a classe pode incluir assessores específicos de memória (mais sobre estes mais tarde, exemplo simples no *m6504*).

Se a CPU tiver a sua própria tabela de expedição, a classe também deve incluir uma declaração (mas não uma definição) de **disasm_entries**, **do_exec_full** e **do_exec_partial**, a declaração e definição de **disasm_disassemble** (idêntico para todas as classes, mas refere-se a uma matriz classe específica **disasm_entries**) e inclui o arquivo `.inc` (que fornece as definições que faltarem). Suporte para a geração também deve ser adicionada ao `CPU.mak`.

Se a CPU possuir algo a mais do que seus opcodes, a sua declaração deve ser feita por meio de uma macro, veja por exemplo o `m65c02`. O arquivo `.inc` irá fornecer as definições.

12.12.5 Tabelas de despacho

Cada arquivo `d<CPU>.lst` é uma tabelas de despacho para a CPU. As linhas que começam com '#' são comentários. O arquivo deve conter 257 entrada, sendo as primeiras 256 sendo opcodes e o 257º dever ser a instrução que a CPU deve fazer durante um reset. Dentro do IRQ e mni do 6502 há uma chamada "mágica" para o opcode "brk", daí a falta de descrição específica para eles.

As entradas entre 0 e 255 por exemplo, os opcodes devem ter uma dessas estruturas:

- `opcode_addressing-mode`
- `opcode_middle_addressing-mode`

O opcode tradicionalmente é um valor com três caracteres. O modo de endereçamento devem ser um valor de 3 cartas correspondente a um dos `DASM_*` macros no `m6502.h`. O Opcode e modo de endereçamento são utilizados para gerar a tabela de desmontagem. O texto completo de entrada é usado na descrição do arquivo de opcode, os métodos de expedição permitem opcodes variantes por CPU que sejam aparentemente idênticos.

Uma entrada de "." era utilizável para opcodes não implementados ou desconhecidos, pois gera códigos "???" na desmontagem, não é uma boa ideia neste momento uma vez que vai realizar um *inffloop* numa função `execute()` caso seja encontrado.

12.12.6 Descrições de Opcode

Cada arquivo `o<CPU>.lst` inclui descrições de opcodes específicas para uma CPU. Uma descrição de opcode é uma série de linhas que começam por uma entrada de opcode por si mesmo e seguido por uma série de linhas recuadas com o código opcode a ser executando. Por exemplo, o opcode `asl <absolute address>` ficaria assim:

```
asl_aba
    TMP = read_pc();
    TMP = set_h(TMP, read_pc());
    TMP2 = read(TMP);
    write(TMP, TMP2);
    TMP2 = do_asl(TMP2);
    write(TMP, TMP2);
    prefetch();
```

A primeira parte baixa do endereço é a leitura, em seguida a parte alta (`read_pc` é incrementada automaticamente). Assim, agora que o endereço está disponível o valor a ser deslocado é lido, depois reescrito (sim, o 6502 faz isso), deslocado novamente e o resultado final é escrito (o `do_asl` cuida das bandeiras). A

instrução termina com um prefetch da próxima instrução, assim como todas as instruções que não quebram a CPU¹ fazem.

As funções de acesso ao barramento são:

read(adr)	leitura comum
read_direct(adr)	lê do espaço do programa
read_pc()	lê no endereço do PC e incrementa
read_pc_noinc()	lê no endereço do PC
read_9()	indexador y do depósito de leitura do 6509
write(adr, val)	escrita comum
prefetch()	instrução prefetch
prefetch_noirq()	instrução prefetch sem verificação de IRQ

A contagem dos ciclos é feita pelo gerador de código que detecta através de strings correspondentes os acessos e gera o código apropriado. Além das funções de acesso ao barramento, uma linha especial pode ser usada para aguardar o próximo evento (irq ou qualquer outro). o "**eat-all-cycles;**" numa linha fará essa espera para que só então continue. Para o m65c02 é usado um *wai_imp* e um *stp_imp*.

Devido às restrições da geração do código, algumas regras devem ser seguidas:

- no geral, fique com uma instrução ou expressão por linha
- não deve haver efeitos colaterais nos parâmetros de uma função de acesso ao barramento
- a vida útil das variáveis locais não deve ultrapassar a de um acesso ao barramento Em geral é melhor deixá-los para ajudar em métodos auxiliares (como o **do_asl**) que não fazem acesso ao barramento. Note que "TMP" e "TMP" não são variáveis locais, são variáveis da classe.
- então uma linha única ou então as construções devem ter chaves ao redor delas caso elas estejam chamando uma função de acesso ao barramento

O código gerado para cada opcode são métodos da classe da CPU. Como tal eles têm acesso completo a outros métodos da classe, variáveis, tudo.

12.12.7 Interface da Memória

Para uma melhor reutilização do opcode com as variantes MMU/banking, foi criada uma subclasse de acesso à memória. É chamado de **memory_interface**, que declarado num dispositivo *m6502_device* e provê os seguintes auxiliares:

UINT8 read(UINT16 adr)	leitura normal
UINT8 read_sync(UINT16 adr)	leitura com sync ativo para opcode (primeiro byte do opcode)
UINT8 read_arg(UINT16 adr)	leitura com sync inativo para opcode (resto do opcode)
void write(UINT16 adr, UINT8 val)	escrita normal

UINT8 read_9(UINT16 adr)	leitura especial para o 6509 com y-indexado, padrão para leitura()
void write_9(UINT16 adr, UINT8 val);	escrita especial para o 6509 com y-indexado, padrão para escrita()

¹ *non-CPU-crashing instructions* no original. (Nota do tradutor)

Por predefinição duas implementações são dadas, uma usual, **mi_default_normal**, uma desabilitando o acesso direto, **mi_default_nd**. Uma CPU que queira a sua própria interface como o 6504 ou o 6509 por exemplo, este deve substituir o `device_start`, inicializar o `mintf` e em seguida chamar a função `init()`.

12.12.8 O código gerado

Um gerador de código é usado para ser compatível com a interrupção durante o reinício de uma instrução. Isso é feito por meio de um estado do sistema com dois níveis e com atualizações apenas nos limites. Para ser mais exato, o `inst_state` informa qual o estado principal no momento. É igual ao byte opcode quando **0-255** e **0xff00** significarem um reset. É sempre válido e usado por instruções como `rmb`. O `inst_substate` indica em qual etapa estamos numa instrução, mas é definida somente quando uma instrução tiver sido interrompida. Vamos voltar ao código `asl <abs>`:

```
asl_aba
    TMP = read_pc();
    TMP = set_h(TMP, read_pc());
    TMP2 = read(TMP);
    write(TMP, TMP2);
    TMP2 = do_asl(TMP2);
    write(TMP, TMP2);
    prefetch();
```

O código completo que foi gerado é:

```
void m6502_device::asl_aba_partial()
{
    switch(inst_substate) {
    case 0:
        if(icount == 0) { inst_substate = 1; return; }
    case 1:
        TMP = read_pc();
        icount--;
        if(icount == 0) { inst_substate = 2; return; }
    case 2:
        TMP = set_h(TMP, read_pc());
        icount--;
        if(icount == 0) { inst_substate = 3; return; }
    case 3:
        TMP2 = read(TMP);
        icount--;
        if(icount == 0) { inst_substate = 4; return; }
    case 4:
        write(TMP, TMP2);
        icount--;
```

```

        TMP2 = do_asl(TMP2);
        if(icount == 0) { inst_substate = 5; return; }
case 5:
    write(TMP, TMP2);
    icount--;
    if(icount == 0) { inst_substate = 6; return; }
case 6:
    prefetch();
    icount--;
}
    inst_substate = 0;
}

```

Percebe-se que a inicial `switch()` reinicia a instrução no *substate* apropriado, que o *icount* é atualizado depois de cada acesso e após chegar a zero (0) a instrução é interrompida e o *substate* atualizado. Desde que a maioria das instruções são iniciadas desde o princípio, uma variante específica é gerada para quando o `inst_substate` for 0:

```

void m6502_device::asl_aba_full()
{
    if(icount == 0) { inst_substate = 1; return; }
    TMP = read_pc();
    icount--;
    if(icount == 0) { inst_substate = 2; return; }
    TMP = set_h(TMP, read_pc());
    icount--;
    if(icount == 0) { inst_substate = 3; return; }
    TMP2 = read(TMP);
    icount--;
    if(icount == 0) { inst_substate = 4; return; }
    write(TMP, TMP2);
    icount--;
    TMP2 = do_asl(TMP2);
    if(icount == 0) { inst_substate = 5; return; }
    write(TMP, TMP2);
    icount--;
    if(icount == 0) { inst_substate = 6; return; }
    prefetch();
    icount--;
}

```

Essa variante remove o interruptor, evitando um dispendioso custo de processamento e também uma gravação de `inst_substate`. Há também uma boa chance de que o decremento teste com um par zerado seja

compilado em algo eficiente.

Todas essas funções de opcode denominam-se através de dois métodos virtuais, **do_exec_full** e **do_exec_partial**, que são gerados numa declaração de chaveamento com 257 entradas. Uma função virtual que implemente um interruptor tem uma boa chance de ser melhor do que ponteiros para métodos de chamada, que custam caro.

A execução da chamada principal é muito simples:

```
void m6502_device::execute_run()
{
    if(inst_substate)
        do_exec_partial();

    while(icount > 0) {
        if(inst_state < 0x100) {
            PPC = NPC;
            inst_state = IR;
            if(machine().debug_flags & DEBUG_FLAG_ENABLED)
                debugger_instruction_hook(this, NPC);
        }
        do_exec_full();
    }
}
```

Caso uma instrução tenha sido parcialmente executada, termine-a (o *icount* então será zero caso ele ainda não tenha terminado). Em seguida, tente executar as instruções completas. A dança do NPC/IR é devido ao fato que o 6502 realiza funções de prefetching², então a instrução PC e opcode vem dos resultados deste prefetch.

12.12.9 Suporte a um slot de contenção/atraso de barramento futuro

O apoio a um slot de contenção e atraso de barramento no contexto do gerador de código requer que este seja capaz de anular um acesso de barramento quando não houver ciclos suficientes disponíveis em *icount* e reiniciá-lo quando os ciclos tornaram-se disponíveis novamente. O plano de implementação seria:

- Tem um método de **delay()** na CPU que remove os ciclos *icount*. Caso o *icount* torne-se menor ou igual à 0, faça com que lance uma exceção **suspend()**.
- Mude o gerador de código para gerar:

```
void m6502_device::asl_aba_partial()
{
    switch(inst_substate) {
    case 0:
        if(icount == 0) { inst_substate = 1; return; }
    case 1:
        try {
```

² Carga prévia de pesquisa, busca ou dado relevante. (Nota do tradutor)


```

    TMP = read_pc();
    } catch(suspend) { inst_substate = 1; return; }
    icount--;
    if(icount == 0) { inst_substate = 2; return; }
case 2:
    try {
        TMP = set_h(TMP, read_pc());
    } catch(suspend) { inst_substate = 2; return; }
    icount--;
    if(icount == 0) { inst_substate = 3; return; }
case 3:
    try {
        TMP2 = read(TMP);
    } catch(suspend) { inst_substate = 3; return; }
    icount--;
    if(icount == 0) { inst_substate = 4; return; }
case 4:
    try {
        write(TMP, TMP2);
    } catch(suspend) { inst_substate = 4; return; }
    icount--;
    TMP2 = do_asl(TMP2);
    if(icount == 0) { inst_substate = 5; return; }
case 5:
    try {
        write(TMP, TMP2);
    } catch(suspend) { inst_substate = 5; return; }
    icount--;
    if(icount == 0) { inst_substate = 6; return; }
case 6:
    try {
        prefetch();
    } catch(suspend) { inst_substate = 6; return; }
    icount--;
}
    inst_substate = 0;
}

```

Caso nenhuma exceção seja lançada, não custa nada tentar uma tentativa de captura mais moderna. Ao usar isso, o controle retorna para o *loop* principal conforme mostrado abaixo:

```

void m6502_device::execute_run()
{
    if(waiting_cycles) {
        icount -= waiting_cycles;
        waiting_cycles = 0;
    }
}

```

```
    }

    if(icount > 0 && inst_substate)
        do_exec_partial();

    while(icount > 0) {
        if(inst_state < 0x100) {
            PPC = NPC;
            inst_state = IR;
            if(machine().debug_flags & DEBUG_FLAG_ENABLED)
                debugger_instruction_hook(this, NPC);
        }
        do_exec_full();
    }

    waiting_cycles = -icount;
    icount = 0;
}
```

Um *icount* negativo significa que a CPU não poderá fazer nada por algum tempo no futuro, porque ela estará aguardando que o barramento seja liberado ou que algum periférico responda. Esses ciclos serão contados até que o processamento normal continue. É importante observar que o caminho da exceção só acontece quando o estado de contenção/espera para além da fatia de planejamento da CPU. O custo deverá ser mínimo, porém, este não é sempre o caso.

12.12.10 Múltiplas variantes de despacho

Algumas variantes estão em processo de serem compatíveis com as mudanças do conjunto de instruções dependam de um sinalizador interno, seja alternando para o modo 16-bits ou alterando alguns acessos de registro para o acessos à memória. Isso é feito tendo várias tabelas de despacho para a CPU, o *d<CPU>.lst* não tem mais 257 entradas e sim $256 \cdot n + 1$. A variável **inst_state_base** deve selecionar qual a tabela de instruções usar num determinado momento. Deve ser um múltiplo de, e é de fato simplesmente *OR* para o byte de primeira instrução visando obter o índice da tabela de despacho (*inst_state*).

12.12.11 Tarefas a serem concluídas

- Implementar os estados de contenção/espera do barramento, mas isso requer suporte no lado do mapa de memória primeiro.
- Integrar os subsistemas de I/O no 4510
- Possivelmente integrar o subsistema de som no n2a03
- Adicionar *hookups* decentes para a bagunça que está no Apple 3

12.13 Software de renderização 3D no MAME

- *Introdução*
- *O conceito*
 - *ObjectType*
 - *Primitivos 3D*
 - *O sincronismo*
- *A classe poly_manager*
 - *Tipos e Constantes*
 - * *vertex_t*
 - * *extent_t*
 - * *render_delegate*
 - *Metodologia*
 - * *poly_manager*
 - * *wait*
 - * *object_data*
 - * *register_poly_array*
 - * *render_tile*
 - * *render_triangle*
 - * *render_triangle_fan*
 - * *render_triangle_strip*
 - * *render_polygon*
 - * *render_extents*
 - * *zclip_if_less*
- *Exemplos do renderizador*
 - *Tipos*
 - *O construtor*
 - *swap_buffers*
 - *clear_buffers*
 - *draw_triangle*
 - *draw_triangle_flat*
 - *draw_triangle_gouraud*
- *Tópico avançado: A classe poly_array*
 - *Metodologia*
 - * *poly_array*

```
* count
* max
* itemsize
* allocated
* byindex
* contiguous
* indexof
* reset
* next
* last
```

12.13.1 Introdução

No final dos anos 80 muitos jogos arcade começaram a incorporar vídeo com gráficos 3D renderizados através de um hardware dedicado. Estes gráficos 3D são tipicamente renderizados a partir de elementos de baixo nível numa memória intermediária (buffer) (geralmente um buffer duplo ou triplo), depois talvez combinados com *tilemaps* ou sprites comuns antes de serem apresentados ao jogador.

Se tratando da emulação de jogos 3D no geral há duas abordagens. A primeira seria aproveitar o hardware 3D atual e traduzir os elementos de baixo nível para formatos equivalentes ao hardware mais modernos. No caso de um emulador de plataforma compartilhada como o MAME, é preciso ter uma API flexível o suficiente para descrever com precisão todos os elementos e todas e as suas respectivas ações. É preciso também que o emulador seja capaz de obter os dados do quadro do buffer que foi renderizado (uma vez que diversos jogos já fazem isso) para então fazer a combinação com outros elementos de uma maneira que a sincronização seja devidamente efetuada em conjunto com a renderização de fundo.

Alternativamente, seria possível renderizar os elementos de baixo nível diretamente através de software. A grande vantagem é poder simular praticamente qualquer comportamento, assim como é feito pelo hardware original ao custo da perda de desempenho. No MAME assim como em toda emulação, isso acontece numa única tarefa o que acaba penalizando o desempenho final. Entretanto, assim como é feita na abordagem com o hardware 3D físico, teoricamente, uma abordagem com base em software poderia ser distribuída para outras tarefas para aliviar o esforço extra desde que houvessem mecanismos para sincronizar tudo quando fosse preciso, por exemplo, ao ler e ao escrever os quadros diretamente a partir de e para o buffer.

O MAME por enquanto usa a segunda abordagem, aproveitando a assistência do modelo de uma classe chamada **poly_manager** para lidar com situações triviais.

12.13.2 O conceito

O princípio do **poly_manager** é ser um mecanismo para dar suporte à renderização multi-tarefa dos elementos 3D de baixo nível. As chamadas fornecem ao **poly_manager** um conjunto de *vértices* para um elemento e mais um retorno do renderizador. O **poly_manager** quebra o elemento em extensões ao longo da linha de varredura e distribui o trabalho entre um conjunto das *tarefas*. O renderizador então é designado na tarefa de cada extensão que está sendo trabalhada onde a lógica específica do jogo pode fazer o que for preciso para que os dados sejam renderizados.

A principal responsabilidade do **poly_manager** é garantir a ordem. Dado um conjunto das tarefas e da quantidade de trabalho dos objetos que serão concluídos, por isso a importância que ao menos dentro de uma determinada linha de varredura, todo o trabalho seja realizado em sequência e na ordem certa. A abordagem seria então atribuir cada extensão a um *contêiner* (*bucket*) com referência na coordenada Y. Assim o **poly_manager** garante que apenas uma tarefa seja realizada por vez e também seja responsável pelo processamento do trabalho num determinado contêiner.

Os vértices no **poly_manager** consistem em simples coordenadas 2D X e Y, incluindo zero ou mais parâmetros adicionais de iteração. Estes parâmetros podem ser qualquer coisa: os valores da intensidade de iluminação; as cores RGB(A) para o shader Gouraud; coordenadas U, V padronizadas para o mapeamento da textura; valores 1/Z para o buffer Z; etc. A iteração dos parâmetros independentemente do que eles representam, são interpolados de forma linear através do elemento no espaço da tela e disponibilizados como parte da extensão da chamada de retorno da tela.

ObjectType

Ao criar uma classe **poly_manager**, você deve definir um tipo especial denominado **ObjectType**.

Como a renderização acontece de forma assíncrona nas tarefas, a ideia é que a classe **ObjectType** tenha uma imagem com todas as informações relevantes para a renderização. Isto permite que a tarefa principal continue, alterando potencialmente alguns dos estados mais relevantes enquanto a renderização acontece em outro lugar.

Em teoria, poderíamos alocar uma nova classe **ObjectType** para cada primitivo da renderização 3D, contudo, seria bastante ineficiente. É bem comum definir o estado da renderização e depois renderizar vários primitivos 3D usando o mesmo estado.

Por esta razão, o **poly_manager** mantém uma matriz interna dos objetos **ObjectType** e mantém uma cópia do último **ObjectType** que foi utilizado. Antes de enviar um novo elemento, os responsáveis pela chamada podem ver se o estado da renderização se alterou. Em caso positivo, é possível pedir ao **poly_manager** para alocar uma nova classe **ObjectType** e preenchê-la. Quando o primitivo 3D é encaminhado para a renderização, a instância mais recente do **ObjectType** é capturada de forma implícita e disponibilizada para as chamadas de retorno da renderização.

Nos cenários mais complexos onde os dados podem se alterar de maneira menos constante, há um modelo semelhante ao **poly_array** que pode ser usado no gerenciamento dos dados. O **poly_manager** interno utiliza a classe **poly_array** no gerenciamento das suas alocações **ObjectType**. Mais informações sobre a classe **poly_array** serão fornecidas mais tarde.

Primitivos 3D

O **poly_manager** é compatível com diferentes tipos de primitivos 3D:

- O elemento mais utilizado pelo **poly_manager** é o *triângulo*, pois tem a propriedade onde os parâmetros iterativos têm deltas constantes através de toda a sua superfície. Ambos também são compatíveis, os *os leques do triângulo* com comprimento arbitrário e as *faixas do triângulo*.
- Em adição aos triângulos o **poly_manager** também é compatível com *polígonos* com uma quantidade arbitrária de vértices. É esperado que a lista dos vértices esteja em ordem horária e anti-horária. O **poly_manager** analisará os limites para computar os deltas através de cada extensão.
- Um caso especial da compatibilidade do **poly_manager** é o primitivo *tile* que é um único *quad* definido por dois vértices, um vértice na diagonal superior esquerda e outro na diagonal inferior direita. Assim como os triângulos, os *tiles* possuem parâmetros iterativos constantes ao longo de toda a sua superfície.
- E concluindo, o **poly_manager** é compatível com um mecanismo totalmente personalizado onde o requerente fornece uma lista das extensões que são aproximadamente alimentadas diretamente nas tarefas. Isso é útil ao emular um sistema com primitivos 3D incomuns onde seja necessário um tipo de comportamento bem específico nas suas bordas.

O sincronismo

A sincronização é um dos principais requisitos para proporcionar um mecanismo assíncrono de renderização. A sincronização no **poly_manager** é muito simples: basta chamar a função `wait()`.

Há diversos motivos para usar um *wait*:

- No momento da exibição, os dados dos pixels devem ser copiados para a tela. Caso algum primitivo 3D seja enfileirado e este toque a parte da tela onde será exibida, será preciso esperar a conclusão da renderização antes de continuar com a cópia. Observe que esta espera pode não ser totalmente necessária em alguns casos (num sistema com *buffer* triplo por exemplo).
- Caso o sistema que esteja sendo emulado tenha um mecanismo para ler o retorno do *framebuffer* depois da renderização, assim um *wait* deve ser usado antes da leitura a fim de garantir que a assincronicidade da renderização seja concluída.
- Caso o sistema que esteja sendo emulado altere qualquer estado que não esteja no cache do **ObjectType** ou num outro lugar (na memória da textura por exemplo), assim um *wait* deve ser usado para garantir que o estado dos primitivos 3D sejam consumidos e que o seu trabalho seja finalizado.
- Caso o sistema que esteja sendo emulado possa usar a renderização de um objeto anterior como a origem da textura para um novo primitivo 3D, então a apresentação do segundo elemento primitivo deve aguardar até que o primeiro primitivo seja concluído. O **poly_manager** não dispõem de nenhum mecanismo interno para auxiliar nessa detecção, assim sendo, cabe àquele que faz a chamada determinar quando ou caso seja necessário.

Como a operação *wait* tem ciência quando acontece a conclusão de toda a renderização, o **poly_manager** também aproveita esta oportunidade para recuperar toda a memória que foi alocada para as suas estruturas internas, bem como a memória que foi alocada nas estruturas **ObjectType**. Por isso é importante que não seja mantido nenhum **ObjectType** após a invocação de um *wait*.

12.13.3 A classe `poly_manager`

Na maioria das aplicações o `poly_manager` não é usado de forma direta, em vez disso, serve como uma classe de referência para uma classe de renderização mais completa. A própria classe do `poly_manager` é um modelo:

```
template<typename BaseType, class ObjectType, int MaxParams, u8 Flags = 0>
class poly_manager;
```

E os parâmetros deste modelo são:

- **BaseType**

É o tipo utilizado internamente para coordenadas e para a iteração dos parâmetros, em geral, deve ser ou `float` ou `double`. Teoricamente, um ponto fixo inteiro também poderia ser utilizado, contudo, você pode se deparar com problemas pois a lógica matemática não foi projetada para isso.

- **ObjectType**

É a estrutura de dados definida por objeto pelo usuário, descrita acima. Internamente, o `poly_manager` vai gerenciar um destes `poly_array` e um ponteiro para a alocação mais recente no momento em que um primitivo 3D for submetido, este será implicitamente encaminhado para o retorno da chamada de cada extensão correspondente.

- **MaxParams**

É a quantidade máxima dos parâmetros iterados que podem ser definidos num vértice. Os parâmetros iterados são genéricos e tratados igualmente, de maneira que o mapeamento dos índices dos parâmetros está completamente alinhado com o vínculo entre a chamada e o seu retorno. É permitido que o **MaxParams** seja 0.

- **Flags**

Pode ser zero ou ser qualquer um dos sinalizadores abaixo:

- **POLY_FLAG_NO_WORK_QUEUE**

Defina este sinalizador para desativar a renderização assíncrona; pode ser útil para fazer depuração. Quando esta opção está ativa, todos os primitivos são enfileirados e depois processados em sequência nas tarefas quando um `wait()` for invocado a partir da classe `poly_manager`.

- **POLY_FLAG_NO_CLIPPING**

Especifique caso queira que o `poly_manager` ignore o corte (*clipping*) interno. Use isso caso o retorno do renderizador faça o seu próprio corte ou caso o solicitante sempre trate o corte antes de submeter os primitivos 3D.

Tipos e Constantes

vertex_t

Dentro da classe do **poly_manager** você encontrará o tipo **vertex_t** faz a descrição de um único vórtice. Todos os métodos de traçado primitivo aceitam 2 ou mais destes objetos **vertex_t**. O **vertex_t** inclui as coordenadas X e Y em conjunto com os valores dos parâmetros de uma matriz iteradas nele:

```
struct vertex_t
{
    vertex_t() { }
    vertex_t(BaseType _x, BaseType _y) { x = _x; y = _y; }

    BaseType x, y;                // coordenadas X, Y
    std::array<BaseType, MaxParams> p; // parâmetros iterados
};
```

Observe que o próprio **vertex_t** está definido dentro dos valores do modelo do **BaseType** e do **MaxParams** que tem posse da classe **poly_manager**.

Todos os primitivos do **poly_manager** operam na região da tela, onde (0,0) representa o canto superior esquerdo da diagonal superior esquerda do pixel, já (0,5,0,5) representa o centro deste pixel. Os valores dos pixels esquerdo e cima são inclusivos, enquanto os valores dos pixels direito e baixo são exclusivos.

Assim, um *tile* renderizado a partir de (2,2)-(4,3) ocupará 2 pixels: (2,2) e (3,2).

Ao invocar um método primitivo de desenho, a matriz dos parâmetros iterativos **p** não precisa ser completamente preenchida. A quantidade dos valores válidos dos parâmetros iterados é definido como base no parâmetro dos métodos de desenho primitivo, de maneira que apenas aquela quantidade de parâmetros precisem ser realmente preenchidos e repassados para as estruturas **vertex_t**.

extent_t

O **poly_manager** divide os primitivos em extensões, são intervalos horizontais contíguos mantidos dentro de uma única linha de varredura. Estas extensões então são distribuídas às tarefas que invocarão a chamada de retorno com as informações sobre como fazer a renderização de cada extensão. O tipo **extent_t** descreve uma dessas extensões, fornecendo as coordenadas X delimitadoras juntamente com uma matriz de valores iniciais dos parâmetros iterados e dos deltas em todo o intervalo:

```
struct extent_t
{
    struct param_t
    {
        BaseType start;                // o início do valor do parâmetro
        BaseType dpdx;                // dp/dx relativo ao início
    };
    int16_t startx, stopx;            // iniciando (inclusivo)/encerrando (exclusivo)_
    ↪ extremidades (endpoints)
    std::array<param_t, MaxParams> param; // matriz de parâmetros início/deltas
    void *userdata;                  // dados personalizados por intervalo
};
```

Para cada parâmetro iterado, o valor **start** contém o valor no lado esquerdo do intervalo. Já o valor **dpdx** contém a alteração do valor do parâmetro de cada coordenada X.

Também há um campo **userdata** na estrutura **extent_t**, que normalmente não é utilizada a não ser durante a execução de uma renderização personalizada.

render_delegate

Ao renderizar um primitivo, além dos vértices, você também deve informar uma chamada de retorno do formulário **render_delegate**:

```
void render(int32_t y, extent_t const &extent, ObjectType const &object, int threadid)
```

Este retorno de chamada é responsável pela renderização propriamente dita. Ela provavelmente será chamada mais tarde para cada extensão numa tarefa de trabalho diferente. Os parâmetros repassados são:

- **y**
É a coordenada Y da scanline da extensão atual.
- **extent**
É a referência à estrutura **extent_t** descrita acima, nesta extensão ela define o início/encerramento do valor X junto com os valores dos parâmetros de cada iteração dos valores do início/delta.
- **object**
É a referência da alocação mais recente do **ObjectType** no momento onde o primitivo foi enviado para ser renderizado; teoricamente deveria ter a maioria, se não todos os dados necessário para realizar a renderização.
- **threadid**
É a identificação única que indica o índice da tarefa de trabalho sendo executada no momento; este valor é útil caso esteja mantendo qualquer tipo de estatística e não queira acrescentar argumentos sobre os valores que são compartilhados. Nesta situação, é possível alocar as instâncias dos dados do **WORK_MAX_THREADS** e atualizar a instância que for passada para o **threadid**. Quando quiser exibir as estatísticas, a principal tarefa de trabalho pode acumular e redefinir os dados de todas as tarefas quando for seguro fazê-lo (após um *wait* por exemplo).

Metodologia

poly_manager

```
poly_manager(running_machine &machine);
```

O construtor do **poly_manager** aceita apenas um parâmetro, uma referência ao **running_machine**. Isso concede ao **poly_manager** o acesso às filas de trabalho necessárias para executar os trabalhos em multi-tarefa.

wait

```
void wait(char const *debug_reason = "general");
```

Invocando o `wait()` suspende as tarefas até que toda a renderização pendente seja concluída:

- **debug_reason**

É um parâmetro opcional que determina o motivo da espera. É útil caso a constante de tempo da compilação **TRACK_POLY_WAITS** esteja ativada, pois ela emitirá um resumo dos tempos de espera e as razões no final da execução.

Retorna: Nada.

object_data

```
objectdata_array &object_data();
```

Este método apenas devolve uma referência ao **poly_array** interno do **ObjectType** que foi definido ao criar o **poly_manager**. Para a maioria das aplicações a única coisa mais interessante a ser feita com ele é invocar o método `next()` para alocar um novo objeto à ser preenchido.

Retorna: Uma referência ao **poly_array** do **ObjectType**.

register_poly_array

```
void register_poly_array(poly_array_base &array);
```

Em aplicações avançadas, é possível optar pela criação dos seus próprios objetos **poly_array** para administrar grandes pedaços de dados alterados com pouca frequência, assim como as paletas. Após cada `wait()`, o **poly_manager** redefine todos os objetos **poly_array** conhecidos a fim de recuperar a pendência de toda a memória que foi alocada. Ao registrar aqui os seus objetos **poly_array** é possível garantir que as suas matrizes também sejam reinicializadas após uma invocação do `wait()`.

Retorna: Nada.

render_tile

```
template<int ParamCount>
uint32_t render_tile(rectangle const &cliprect, render_delegate callback,
                    vertex_t const &v1, vertex_t const &v2);
```

Este método enfileira um único *tile* primitivo para a renderização:

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido no instanciamento do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **v1**

Contém as coordenadas e os parâmetros de iteração para o canto superior esquerdo do tile.

- **v2**

Contém as coordenadas e os parâmetros de iteração para o canto superior direito do tile.

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

render_triangle

```
template<int ParamCount>
uint32_t render_triangle(rectangle const &cliprect, render_delegate callback,
                        vertex_t const &v1, vertex_t const &v2, vertex_t const &v3);
```

Este método enfileira um único *triângulo* primitivo para a renderização:

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido no instanciamento do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **v1, v2, v3**

Contém as coordenadas e os parâmetros de iteração para cada vértice do triângulo.

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

render_triangle_fan

```
template<int ParamCount>
uint32_t render_triangle_fan(rectangle const &cliprect, render_delegate callback,
                             int numverts, vertex_t const *v);
```

Este método enfileira um ou mais *triângulos* primitivos para a renderização, definido pela sua sequência:

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido na instânciação do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **numverts**

A quantidade total dos vértices fornecidos; deve ser pelo menos 3.

- **v**

É um ponteiro para uma matriz de objetos **vertex_t** contendo as coordenadas e os parâmetros iterados para todos os triângulos em leque. Significa que o primeiro vértice é fixo. Portanto, caso sejam apresentados 5 vértices, indicando 3 triângulos, os vértices utilizados serão: (0,1,2) (0,2,3) (0,3,4)

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

render_triangle_strip

```
template<int ParamCount>
uint32_t render_triangle_strip(rectangle const &cliprect, render_delegate callback,
                               int numverts, vertex_t const *v);
```

Este método enfileira um ou mais *triângulos* primitivos para a renderização, definido em ordem de tiras:

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido na instânciação do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **numverts**

A quantidade total dos vértices fornecidos; deve ser pelo menos 3.

- **v**

É um ponteiro para uma matriz de objetos **vertex_t** contendo as coordenadas e os parâmetros iterados para todos os triângulos, definido em ordem de tiras: Portanto, caso sejam apresentados 5 vértices, indicando 3 triângulos, os vértices utilizados serão: (0,1,2) (1,2,3) (2,3,4)

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

render_polygon

```
template<int NumVerts, int ParamCount>
uint32_t render_polygon(rectangle const &cliprect, render_delegate callback, vertex_t const_
↳*v);
```

Este método enfileira um único *polígono* primitivo para a renderização:

- **NumVerts**

É a quantidade dos vértices num polígono.

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido no instanciamento do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **v**

É um ponteiro para uma matriz de objetos **vertex_t** contendo as coordenadas e os parâmetros iterados para o polígono. É esperado que os vértices sejam ou em ordem horária ou em ordem anti-horária.

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

render_extents

```
template<int ParamCount>
uint32_t render_extents(rectangle const &cliprect, render_delegate callback,
↳int startscanline, int numscanlines, extent_t const *extents);
```

Este método enfileira as extensões personalizadas diretamente:

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido no instanciamento do modelo do **poly_manager**.

- **cliprect**

É uma referência ao recorte de um retângulo. Todos os pixels e todos os valores dos parâmetros são recortados para ficar dentro destes limites antes de serem adicionados nas filas de trabalho para a sua renderização, a menos que **POLY_FLAG_NO_CLIPPING** tenha sido definido como um parâmetro de sinalização para o **poly_manager**.

- **callback**

É o responsável pelo retorno da chamada que será feita para renderizar cada extensão.

- **startscanline**

É a coordenada Y da primeira extensão fornecida.

- **numscanlines**

É a quantidade das extensões fornecidas.

- **extents**

É um ponteiro para um conjunto de objetos **extent_t** contendo o início/parada (start/stop) das coordenadas X e os parâmetros iterados. O campo **userdata** da origem da extensão também é copiado para o destino (este campo não é usado para todos os outros tipos de renderização).

Retorna: A quantidade total dos pixels que foram recortados representado pelas extensões consultadas.

zclip_if_less

```
template<int ParamCount>
int zclip_if_less(int numverts, vertex_t const *v, vertex_t *outv, BaseType clipval);
```

Este método é um método de auxílio para cortar um polígono contra um valor Z informado. Ele assume que o primeiro parâmetro iterado em **vertex_t** representa a coordenada Z. Caso alguma borda cruze o plano Z representado por **clipval**, esta borda é aparada.

- **ParamCount**

É a quantidade dos valores ativos na matriz de parâmetros iterados dentro de cada **vertex_t** apresentado; não deve ser maior que o valor de **MaxParams** definido na instânciação do modelo do **poly_manager**.

- **numverts**

É a quantidade dos vértices na entrada da matriz.

- **v**

É um ponteiro para a matriz de entrada dos objetos **vertex_t**.

- **outv**

É um ponteiro para a saída da matriz dos objetos **vertex_t**. O **v** e o **outv** não podem se sobrepor ou apontar para a mesma memória.

- **clipval**

É o valor de recorte que deve ser comparado com o parâmetro 0.

Retorna: A quantidade dos vértices gerados escritos em **outv**.

Observe que desde a concepção, é possível que este método produza mais vértices do que a matriz de entrada, portanto, aqueles que forem invocá-lo devem garantir que haja espaço suficiente na saída do buffer para acomodar isso.

12.13.4 Exemplos do renderizador

Aqui está um exemplo completo de como criar um software renderizador 3D através do **poly_manager**. O nosso renderizador de exemplo só manuseará triângulos planos e com shaders Gouraud com buffer de profundidade (Z).

Tipos

A primeira coisa que precisamos definir é o formato da nossa vértice *visível externamente* * *que é diferente do* `**vertex_t*` interno que vai definir o **poly_manager**. Em teoria é possível usar **vertex_t** diretamente, porém a natureza genérica dos parâmetros iterados do **poly_manager** torna as coisas estranhas:

```
struct example_vertex
{
    float x, y, z;      // Coordenadas X,Y,Z
    rgb_t color;        // a cor neste vértice
};
```

Em seguida, definimos o **ObjectType** necessário para **poly_manager**. Para o nosso simples exemplo, nós definimos uma estrutura **example_object_data** que consiste em ponteiros para os nossos buffers de renderização, mais um par de valores fixos que em alguns casos são consumidos. Renderizadores mais complexos normalmente têm muitos mais parâmetros de objeto definidos aqui:

```
struct example_object_data
{
    bitmap_rgb32 *dest;    // ponteiro para o renderizador do bitmap
    bitmap_ind16 *depth;  // ponteiro de profundidade do bitmap
    rgb_t color;          // coloração geral (para a limpeza e o caso de sombreado plano)
    uint16_t depthval;    // valor da profundidade fixa (para limpeza)
};
```


Agora definimos a nossa classe do renderizador que derivamos do **poly_manager**. Como parâmetros do modelo float que definimos como o tipo base para os nossos dados, uma vez que isso será preciso o suficiente para este exemplo e também fornecemos os nossos **example_object_data** como a classe **ObjectType**, mais a quantidade máxima dos parâmetros iterados que o nosso renderizador precisará (4 neste caso):

```
class example_renderer : public poly_manager<float, example_object_data, 4>
{
public:
    example_renderer(running_machine &machine, uint32_t width, uint32_t height);

    bitmap_rgb32 *swap_buffers();

    void clear_buffers(rgb_t color, uint16_t depthval);
    void draw_triangle(example_vertex const *verts);

private:
    static uint16_t ooz_to_depthval(float ooz);

    void draw_triangle_flat(example_vertex const *verts);
    void draw_triangle_gouraud(example_vertex const *verts);

    void render_clear(int32_t y, extent_t const &extent, example_object_data const &object,
↳int threadid);
    void render_flat(int32_t y, extent_t const &extent, example_object_data const &object,
↳int threadid);
    R

    int m_draw_buffer;
    bitmap_rgb32 m_display[2];
    bitmap_ind16 m_depth;
};
```

O construtor

O construtor do nosso renderizador de exemplo inicializa apenas o **poly_manager** e aloca os buffers de renderização e de profundidade:

```
example_renderer::example_renderer(running_machine &machine, uint32_t width, uint32_t
↳height) :
    poly_manager(machine),
    m_draw_buffer(0)
{
    // aloca dois buffers para a exibição e um buffer de profundidade
    m_display[0].allocate(width, height);
    m_display[1].allocate(width, height);
    m_depth.allocate(width, height);
}
```

swap_buffers

O primeiro método interessante em nosso renderizador é o `swap_buffers()` que retorna um ponteiro ao buffer para onde estamos desenhando e configura o outro buffer como um novo alvo para ser desenhada. A ideia é que o manipulador da atualização da tela chamará este método para obter o bitmap que será mostrado ao usuário:

```
bitmap_rgb32 *example_renderer::swap_buffers()
{
    // aguarde pela conclusão de qualquer renderização antes de devolver o buffer
    wait("swap_buffers");

    // devolva o buffer do desenho atual e em seguida, alterne para o outro
    // para desenho futuro
    bitmap_rgb32 *result = &m_display[m_draw_buffer];
    m_draw_buffer ^= 1;
    return result;
}
```

O mais importante a ser observado aqui é a chamada para o **poly_manager** `wait()` que bloqueará a linha atual até que toda a renderização esteja concluída. Isto é importante pois, caso contrário, o requerente pode receber um bitmap que ainda está sendo desenhado levando a visuais quebrados ou corrompidos.

clear_buffers

Uma das operações mais comuns a serem realizadas ao fazer a renderização 3D é inicializar ou limpar a tela e os buffers de profundidade para um valor conhecido. O método abaixo alavanca a *tela* primitiva para renderizar um retângulo na tela passando em (0,0) e (largura,altura) para os dois vértices.

Como os valores de cor e de profundidade para limpar o buffer são constantes, eles são armazenados num objeto recentemente alocado **example_object_data**, juntamente com um ponteiro para os buffers em questão. A chamada `render_tile()` é feita com um sufixo `<0>` indicando que não há parâmetros iterados para se preocupar:

```
void example_renderer::clear_buffers(rgb_t color, uint16_t depthval)
{
    // aloque os dados do objeto e preencha-os com as informações necessárias
    example_object_data &object = object_data().next();
    object.dest = &m_display[m_draw_buffer];
    object.depth = &m_depth;
    object.color = color;
    object.depthval = depthval;

    // topo, a coordenada esquerda sempre é (0,0)
    vertex_t topleft;
    topleft.x = 0;
    topleft.y = 0;

    // inferior, a coordenada direita é (largura,altura)
    vertex_t botright;
    botright.x = m_display[0].width();
    botright.y = m_display[0].height();
}
```

(continua na próxima página)

(continuação da página anterior)

```

// renderize como um bloco com 0 parâmetros iterados
render_tile<0>(m_display[0].cliprect(),
               render_delegate(&example_renderer::render_clear, this),
               topleft, botright);
}

```

O retorno de chamada de renderização fornecido para `render_tile()` também é definido (privadamente) na nossa classe e lida com uma única extensão. Observe como os parâmetros da renderização são extraídos da estrutura fornecida **example_object_data**:

```

void example_renderer::render_clear(int32_t y, extent_t const &extent, example_object_data_
↳const &object, int threadid)
{
    // obtém os ponteiros para o início do buffer de profundidade e linhas de varredura do_
↳destino
    uint16_t *depth = &object.depth->pix(y);
    uint32_t *dest = &object.dest->pix(y);

    // faz um loop em toda a extensão e apenas armazena os valores constantes do objeto
    for (int x = extent.startx; x < extent.stopx; x++)
    {
        dest[x] = object.color;
        depth[x] = object.depthval;
    }
}

```

Outro ponto importante a ser ressaltado é que as coordenadas X fornecidas pela extensão da estrutura são inclusiva da *startx*, porém, exclusivas do *stopx*. O recorte é feito antes do tempo para que o retorno da chamada possa se concentrar na disposição dos pixels o mais rápido possível com o mínimo de sobrecarga.

draw_triangle

A seguir, temos a nossa função real de renderização triangular, que desenhara um único triângulo dado um conjunto de três vértices disponibilizados no formato externo **example_vertex**:

```

void example_renderer::draw_triangle(example_vertex const *verts)
{
    // caixa plana sombreada
    if (verts[0].color == verts[1].color && verts[0].color == verts[2].color)
        draw_triangle_flat(verts);
    else
        draw_triangle_gouraud(verts);
}

```

Como é mais simples e mais rápido renderizar um triângulo plano sombreado, o código verifica se as cores são as mesmas em todos os três vértices. Caso sejam, invocamos para um caso com shader plano, caso contrário o processamos como um triângulo shader Gouraud completo.

Esta é uma técnica comum para otimizar o desempenho da renderização: identificar casos especiais que reduzem o trabalho por pixel, e encaminhá-los para separar o retorno da renderização que são otimizados para aquele caso especial.

draw_triangle_flat

Aqui está o código de configuração para renderizar um triângulo com um shader plano:

```
void example_renderer::draw_triangle_flat(example_vertex const *verts)
{
    // aloque os dados do objeto e preencha-os com as informações necessárias
    example_object_data &object = object_data().next();
    object.dest = &m_display[m_draw_buffer];
    object.depth = &m_depth;

    // neste caso, a cor é constante e definida nos dados do objeto
    object.color = verts[0].color;

    // copie X, Y e 1/Z para os vértices poly_manager
    vertex_t v[3];
    for (int vertnum = 0; vertnum < 3; vertnum++)
    {
        v[vertnum].x = verts[vertnum].x;
        v[vertnum].y = verts[vertnum].y;
        v[vertnum].p[0] = 1.0f / verts[vertnum].z;
    }

    // renderiza o triângulo com 1 parâmetro iterado (1/Z)
    render_triangle<1>(m_display[0].cliprect(),
                      render_delegate(&example_renderer::render_flat, this),
                      v[0], v[1], v[2]);
}
```

Primeiro, colocamos diretamente a cor fixa no **example_object_data**, depois preenchemos três objetos **vertex_t** objetos com as coordenadas X e Y no local habitual, e 1/Z como o nosso único e único parâmetro iterado. (Usamos aqui 1/Z porque os parâmetros de iteração são interpolados linearmente no espaço da tela. Z não é linear no espaço da tela, já 1/Z é por causa da correção de perspectiva).

No caso do nosso shader plano então invoca o `render_triangle` definido o parâmetro iterado `<1>` para interpolar e apontando para um caso especial do retorno de chamada do renderizador plano:

```
void example_renderer::render_flat(int32_t y, extent_t const &extent, example_object_data_t
↳const &object, int threadid)
{
    // obtém os ponteiros para o início do buffer de profundidade e linhas de varredura do
↳destino
    uint16_t *depth = &object.depth->pix(y);
    uint32_t *dest = &object.dest->pix(y);

    // obtenha o valor inicial 1/Z e o delta por X
    float ooz = extent.param[0].start;
    float doozdx = extent.param[0].dpdx;

    // itera sobre a extensão
    for (int x = extent.startx; x < extent.stopx; x++)
    {
        // converta o valor 1/Z num valor de profundidade integral
        uint16_t depthval = ooz_to_depthval(ooz);

        // se mais próximo do que o pixel atual, copie o valor da cor e da profundidade
        if (depthval < depth[x])
        {
            dest[x] = object.color;
            depth[x] = depthval;
        }

        // independentemente, atualize o valor de 1/Z para o próximo pixel
        ooz += doozdx;
    }
}
```

Isto torna a chamada de retorno é um pouco mais envolvente do que o caso da liberação.

Primeiro, temos que lidar com um parâmetro iterado ($1/Z$) cujos valores iniciais e do X-delta nós extraímos da extensão antes do início do loop interno.

Em segundo lugar, realizamos testes no buffer de profundidade utilizando o `ooz_to_depthval()` como um auxiliar para transformar o valor de ponto flutuante $1/Z$ num inteiro com 16 bits. Comparamos este valor com o valor atual do buffer de profundidade, e só armazenamos o valor do pixel/profundidade caso seja inferior.

Ao final de cada iteração, avançamos o valor $1/Z$ pelo delta X, em preparação para o próximo pixel.

draw_triangle_gouraud

Concluimos agora com o código para completar o caso Gouraud-shaded:

```
void example_renderer::draw_triangle_gouraud(example_vertex const *verts)
{
    // aloque os dados do objeto e preencha-os com as informações necessárias
    example_object_data &object = object_data().next();
    object.dest = &m_display[m_draw_buffer];
    object.depth = &m_depth;

    // copie X, Y, 1/Z e R,G,B para os vértices do poly_manager
    vertex_t v[3];
    for (int vertnum = 0; vertnum < 3; vertnum++)
    {
        v[vertnum].x = verts[vertnum].x;
        v[vertnum].y = verts[vertnum].y;
        v[vertnum].p[0] = 1.0f / verts[vertnum].z;
        v[vertnum].p[1] = verts[vertnum].color.r();
        v[vertnum].p[2] = verts[vertnum].color.g();
        v[vertnum].p[3] = verts[vertnum].color.b();
    }

    // renderize o triângulo com 4 parâmetros iterados (1/Z, R, G, B)
    render_triangle<4>(m_display[0].cliprect(),
                      render_delegate(&example_renderer::render_gouraud, this),
                      v[0], v[1], v[2]);
}
```

Aqui temos 4 parâmetros iterados: o valor da profundidade 1/Z, mais o vermelho, verde e azul, armazenados como valores de ponto flutuante. Invocamos o `render_triangle()` com <4> já que é a quantidade de parâmetros iterados para serem processados e apontamos para o retorno completo da chamada do renderizador do Gouraud:

```
void example_renderer::render_gouraud(int32_t y, extent_t const &extent, example_object_data_
↳const &object, int threadid)
{
    // obtém os ponteiros para o início do buffer de profundidade e linhas de varredura do_
↳destino
    uint16_t *depth = &object.depth->pix(y);
    uint32_t *dest = &object.dest->pix(y);

    // obtenha o valor inicial 1/Z e o delta por X
    float ooz = extent.param[0].start;
    float doozdx = extent.param[0].dpx;

    // obtenha os valores iniciais de R,G,B e o delta por X como 8,24 com valores de ponto_
↳fixo
    uint32_t r = uint32_t(extent.param[1].start * float(1 << 24));
    uint32_t drdx = uint32_t(extent.param[1].dpx * float(1 << 24));
    uint32_t g = uint32_t(extent.param[2].start * float(1 << 24));
    uint32_t dgdx = uint32_t(extent.param[2].dpx * float(1 << 24));
    uint32_t b = uint32_t(extent.param[3].start * float(1 << 24));
    uint32_t dbdx = uint32_t(extent.param[3].dpx * float(1 << 24));

    // itera sobre a extensão
    for (int x = extent.startx; x < extent.stopx; x++)
    {
        // converta o valor 1/Z num valor de profundidade integral
        uint16_t depthval = ooz_to_depthval(ooz);

        // caso esteja mais próximo do que o pixel atual, monte a cor
        if (depthval < depth[x])
        {
            dest[x] = rgb_t(r >> 24, g >> 24, b >> 24);
            depth[x] = depthval;
        }

        // independentemente, atualize os valores 1/Z e R,G,B para o próximo pixel
        ooz += doozdx;
        r += drdx;
        g += dgdx;
        b += dbdx;
    }
}
```

Isto segue o mesmo padrão de retorno da chamada com o sombreado plano, exceto que temos 4 parâmetros de iteração para prosseguir.

Observe que ainda que os parâmetros iterados sejam do tipo "flutuante", convertemos os valores das cores em inteiros de ponto fixo quando iteramos sobre eles. Isto nos poupa de fazer 3 conversões *float-to-int* em cada pixel. Os valores RGB originais eram entre 0-255, portanto a interpolação só pode produzir valores na faixa entre 0-255. Assim, podemos usar 24 bits de um inteiro com 32 bits como fração, que é bastante preciso neste caso.

12.13.5 Tópico avançado: A classe `poly_array`

A classe **`poly_array`** tem como modelo uma classe que é utilizada para gerenciar um vetor dinamicamente dimensionado de objetos cuja vida útil começa na alocação e termina quando o `reset()` for invocado. A classe **`poly_manager`** utiliza internamente vários objetos **`poly_array`**, incluindo um para os dados **`ObjectType`** que foram alocados, um para cada renderização primitiva e um para manter todas as extensões alocadas.

Ela também tem uma propriedade adicional onde após um *reset* retém uma cópia do objeto alocado mais recentemente. Isto assegura que quem invoca sempre pode invocar o `last()` e obter imediatamente um objeto válido mesmo após um *reset*.

A classe **`poly_array`** exige dois modelos de parâmetros:

```
template<class ArrayType, int TrackingCount>
class poly_array;
```

Estes parâmetros são:

- **`ArrayType`**

É o tipo do objeto que se deseja alocar e administrar.

- **`TrackingCount`**

É a quantidade dos objetos que se deseja manter após um *reset*. Normalmente este valor ou é 0 (não importa rastrear nenhum objeto) ou 1 (só é necessário um objeto); entretanto, caso esteja usando **`poly_array`** para gerenciar uma coleção compartilhada de objetos entre vários consumidores independentes, ele pode ser maior. Veja abaixo um exemplo onde isto pode ser útil.

Note que os objetos alocados por **`poly_array`** são propriedade do **`poly_array`** e serão automaticamente liberados mediante o seu encerramento.

A classe **`poly_array`** é otimizado para uso em sistemas multi-tarefa de alta frequência. Portanto, uma característica adicional da classe em questão é o arredondamento do tamanho da alocação do **`ArrayType`** para o limite da linha de cache mais próxima, na suposição onde as entradas vizinhas poderiam ser acessadas por núcleos diferentes simultaneamente. Manter cada objeto do **`ArrayType`** na sua própria linha de cache assegura que não haja falsos impactos no desempenho do compartilhamento.

Atualmente, a classe **`poly_array`** não possui nenhum mecanismo para determinar o tamanho da linha de cache em tempo real, portanto, presume-se que 64 bytes seja um tamanho típico para a linha de cache, que é verdadeiro para a maioria dos chips x64 e ARM desde 2021. Este valor pode ser modificado alterando-se a constante **`CACHE_LINE_SHIFT`** definido no topo da classe.

Os objetos alocados pela classe **`poly_array`** são criados em blocos de 64k. No momento da construção, um pedaço dos objetos é antecipadamente alocado. O tamanho do pedaço é controlado pela constante **`CHUNK_GRANULARITY`** definido no topo da classe.

Como mais objetos são alocados, caso a classe **`poly_array`** fique sem espaço, mais será alocado de forma dinâmica. Este processo produzirá pedaços separados dos objetos até a próxima chamada `reset()`, quando a **`poly_array`** realocará todos os objetos num vetor contíguo novamente.

No caso onde a **`poly_array`** seja utilizada para gerenciar um pool compartilhado dos objetos, a classe pode ser configurada para reter vários itens alocados mais recentemente utilizando uma **`TrackingCount`** maior que 1. Caso a **`poly_array`** esteja gerenciando objetos para duas unidades de textura por exemplo, será possível definir o **`TrackingCount`** igual à 2 e passar o índice da unidade da textura em chamadas para `next()` e `last()`. Após um *reset*, a classe **`poly_array`** lembrará do objeto alocado mais recentemente para cada uma das unidades de forma independente.

Metodologia

poly_array

```
poly_array();
```

O construtor **poly_array** não precisa de parâmetros e basicamente pré-aloca um pedaço dos objetos preparando-os para futuras alocações.

count

```
u32 count() const;
```

Retorna: A quantidade dos objetos atualmente alocados.

max

```
u32 max() const;
```

Retorna: A quantidade máxima dos objetos já alocados em algum momento.

itemsizes

```
size_t itemsizes() const;
```

Retorna: O tamanho de um objeto, arredondado para o limite da linha do cache mais próxima.

allocated

```
u32 allocated() const;
```

Retorna: A quantidade dos objetos que cabem no que foi alocado atualmente.

byindex

```
ArrayType &byindex(u32 index);
```

Retorna uma referência de um objeto na matriz por índice. Equivale ao **[index]** numa matriz normal:

- **index**

É o índice do item que você deseja consultar.

Retorna: Uma referência ao objeto em questão. Como uma referência é devolvida, a sua responsabilidade é garantir que o **index** seja inferior a **count()**, pois não há qualquer mecanismo para devolver um resultado inválido.

contiguous

```
ArrayType *contiguous(u32 index, u32 count, u32 &chunk);
```

Retorna um ponteiro para a base de uma seção contígua dos itens **count** iniciando no **index**. Como **poly_array** se redimensiona dinamicamente, pode não ser possível acessar todos os objetos **count** de forma contígua, então a quantidade dos objetos realmente contíguos é devolvido no **chunk**:

- **index**

É o índice do primeiro item que se deseja acessar de forma contígua.

- **count**

É a quantidade dos itens que se deseja acessar de forma contígua.

- **chunk**

É uma referência a uma variável que será definida para a quantidade real dos itens contíguos disponíveis a partir do **index**. Caso o **chunk** seja inferior ao **count**, então o solicitante deverá processar os itens **chunk** devolvidos, então invoque novamente `contiguous()` no **(index + chunk)** para ter acesso ao restante.

Retorna: Um ponteiro ao primeiro item no pedaço contíguo. Nenhuma verificação do intervalo é feito, portanto a sua responsabilidade é garantir que **index + count** seja menor ou igual à `count()`.

indexof

```
int indexof(ArrayType &item) const;
```

Retorna o índice dentro da matriz do item em questão:

- **item**

É uma referência a um item na matriz.

Retorna: O índice do item. Deve sempre ser o caso onde:

```
array.indexof(array.byindex(index)) == index
```

reset

```
void reset();
```

Redefine o **poly_array**, desalocando semanticamente todos os objetos. Caso as alocações anteriores tenham criado uma matriz não contígua, um novo vetor é alocado neste momento para que as alocações futuras até o mesmo nível permaneçam contíguas.

Observe que o **ArrayType** destruidor *não* é invocado nos objetos pois eles são desalocados.

Retorna: Nada.

next

```
ArrayType &next(int tracking_index = 0);
```

Atribui um novo objeto e devolve uma referência a ele. Caso não haja espaço suficiente para um novo objeto na matriz atual, uma nova matriz não contígua é criada para mantê-lo:

- **tracking_index**

É o índice de rastreamento que se deseja atribuir um novo item. Neste caso comum, isto é 0, mas poderia ser diferente de zero caso se utilize um **TrackingCount** maior que 1.

Retorna: Uma referência ao objeto. Observe que o posicionamento do novo operador é invocado sobre este objeto, portanto o construtor predefinido **ArrayType** será invocado aqui.

last

```
ArrayType &last(int tracking_index = 0) const;
```

Retorna uma referência ao último objeto que foi alocado:

- **tracking_index**

É o índice de rastreamento do objeto desejado. Neste caso comum, isto é 0, mas poderia ser diferente de zero caso se utilize um **TrackingCount** maior que 1. O **poly_array** recorda o objeto recentemente alocado de forma independente para cada **tracking_index**.

Retorna: Uma referência ao último objeto alocado.

O MAME E A PREOCUPAÇÃO COM A SEGURANÇA

O MAME não foi desenvolvido e tão pouco é destinado para uso em ambientes seguros, muito menos foi auditado para tanto. Caso o MAME venha a ser executado por um usuário com poderes administrativos, já é conhecido que no passado houveram falhas que poderiam ser exploradas com fins maliciosos.

NÓS NÃO RECOMENDAMOS, TÃO POUCO INCENTIVAMOS, QUE O MAME SEJA USADO POR USUÁRIOS COM PODERES DE ADMINISTRADOR OU ROOT. TODOS OS DANOS QUE ISSO VENHA A CAUSAR SERÁ POR SUA CONTA E RISCO.

Os relatórios de problemas,¹² no entanto, são sempre bem vindos.

¹ Pedimos a gentileza de relatar os problemas encontrados em Inglês. (Nota do tradutor)

² Você pode colaborar reportando os erros encontrados no site [MAME Testers](#). (Nota do tradutor)

O projeto MAME como um todo é distribuído sob os termos do [GNU General Public License, versão 2 ou posterior](#) (GPL-2.0 +), uma vez que contém código disponibilizado sob várias licenças compatíveis com a GPL. A grande maioria dos arquivos (mais de 90% incluindo arquivos de núcleo) estão sob a [BSD-3-Clause License](#) e nós gostaríamos de incentivar os novos colaboradores para distribuir os arquivos sob esta licença.

MAME é uma marca registrada de Gregory Ember. Outras marcas registradas são propriedade dos seus respectivos proprietários. É preciso uma permissão para que se possa usar o nome, o logo e a marca "MAME".

Direitos autorais (c) 1997-2024 MAMEDev e colaboradores.

Este programa é um software livre; você pode redistribuí-lo e/ou modificá-lo sob os termos da Licença Pública Geral GNU como publicada pela Free Software Foundation; na versão 3 da Licença, ou (a seu critério) qualquer versão posterior.

Este programa é distribuído na esperança de que possa ser útil, porém SEM NENHUMA GARANTIA; sem uma garantia implícita de ADEQUAÇÃO a qualquer MERCADO ou APLICAÇÃO EM PARTICULAR. Veja a Licença Pública Geral GNU para mais detalhes.

Você deve ter recebido uma cópia da Licença Pública Geral GNU juntos com este programa. Na falta, obtenha em [Licenças GNU](#). 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Para mais informações acesse [license information](#).

LICENCIAMENTO DESTE DOCUMENTO E SUA TRADUÇÃO

A **Atribuição 4.0 Internacional (CC BY 4.0)** em resumo, lhe garantem os seguintes direitos:

- **Compartilhar**, copiar e redistribuir este material através de qualquer meio ou formato.
- **Adaptar**, misturar, transformar e criar a partir dele, para qualquer finalidade, inclusive comercial.

Desde que sejam respeitadas as seguintes condições:

- **Atribuição**, você deve dar o devido crédito, prover um link para a licença e indicar se mudanças foram feitas. Você deve fazê-lo em qualquer circunstância razoável, mas de nenhuma maneira que sugira que o licenciante apoia você ou o seu uso.
- **Sem restrições adicionais**, você não pode usar de meios jurídicos ou de qualquer outro meio de restrição, inclusive eletrônicos, digital ou qualquer outro meio restritivo que venha a aparecer, que impeçam outros de fazer legalmente o que o que esta licença já esteja permitindo.

Este trabalho de tradução está licenciado sob a Licença Atribuição 4.0 Internacional Creative Commons por Wellington T. Uemura.

Para visualizar uma cópia desta licença, visite <http://creativecommons.org/licenses/by/4.0/> ou mande uma carta para:

Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

CÓDIGO DE CONDUTA PARA COLABORADORES

16.1 Nossa promessa

Com o interesse de fomentar uma comunidade aberta e acolhedora, nós, como colaboradores e administradores deste projeto, comprometemo-nos a fazer a participação deste projeto uma experiência livre de assédio para todos, independentemente da aparência pessoal, deficiência, etnia, gênero, idade, identidade ou expressão de gênero, identidade ou orientação sexual, nacionalidade, nível de experiência, porte físico, raça ou religião.

16.2 Nossos padrões

Exemplos de comportamentos que contribuem a criar um ambiente positivo incluem:

- Usar linguagem acolhedora e inclusiva
- Respeitar pontos de vista e experiências diferentes
- Aceitar crítica construtiva com graça
- Focar no que é melhor para a comunidade
- Mostrar empatia com outros membros da comunidade

Exemplos de comportamentos inaceitáveis por parte dos participantes incluem:

- Uso de linguagem ou imagens sexuais e atenção ou avanço sexual indesejada
- Comentários insultuosos e/ou depreciativos e ataques pessoais ou políticos (Trolling)
- Assédio público ou privado
- Publicar informação pessoal de outros sem permissão explícita, como, por exemplo, um endereço eletrônico ou residencial
- Qualquer outra forma de conduta que pode ser razoavelmente considerada inapropriada num ambiente profissional

16.3 Nossas responsabilidades

Os administradores do projeto são responsáveis por esclarecer os padrões de comportamento e deverão tomar ação corretiva apropriada e justa em resposta a qualquer instância de comportamento inaceitável.

Os administradores do projeto têm o direito e a responsabilidade de remover, editar ou rejeitar comentários, commits, código, edições na wiki, erros ou outras formas de contribuição que não estejam de acordo com este Código de Conduta, bem como banir temporariamente ou permanentemente qualquer colaborador por qualquer outro comportamento que se considere impróprio, perigoso, ofensivo ou problemático.

16.4 Escopo

Este Código de Conduta aplica-se dentro dos espaços do projeto ou qualquer espaço público onde alguém represente o mesmo ou a sua comunidade. Exemplos de representação do projeto ou comunidade incluem usar um endereço de email oficial do projeto, postar por uma conta de mídia social oficial, ou agir como um representante designado num evento online ou offline. A representação de um projeto pode ser ainda definida e esclarecida pelos administradores do projeto.

16.5 Aplicação

Comportamento abusivo, de assédio ou de outros tipos pode ser comunicado contatando a equipe do projeto wellingtonuemura@gmail.com. Todas as queixas serão revistas e investigadas e resultarão numa resposta necessária e apropriada à situação. A equipe é obrigada a manter a confidencialidade em relação ao elemento que reportou o incidente. Demais detalhes de políticas de aplicação podem ser postadas separadamente.

Administradores do projeto que não sigam ou não mantenham o Código de Conduta em boa fé podem enfrentar repercussões temporárias ou permanentes determinadas por outros membros da liderança do projeto.

16.6 Atribuição

Este Código de Conduta é adaptado do **Contributor Covenant**, versão 1.4, disponível em <https://www.contributor-covenant.org/pt-br/version/1/4/code-of-conduct.html>