

Universidad Complutense de Madrid

Facultad de Informática



Museo de Informática García Santesmases

Departamento de Videojuegos

Máquina Arcade

Jaime Cornejo Pérez

Índice

1

Objetivos	03
Historia	04
Datos Técnicos	06
Apéndices	1?
Apéndice A: Desmontar y acceder a todos los rincones	1?
Apéndice B: Manuales y Documentación específica	1?
Apéndice C: Conector JAMMA	??
Apéndice D: Software, configuración y uso	??
Apéndice E: Documentos externos	??
Referencias	5?

Objetivos

2

Los objetivos que persigue el proyecto son:

Objetivos primarios:

- Conseguir una máquina arcade para el departamento de videojuegos del MIGS.
- Introducir un PC en el interior de la máquina recreativa permitiéndonos interactuar con esta.
- Instalar, configurar y dejar a punto en el PC todo el software necesario para poder “crear la ilusión” de que tenemos una máquina recreativa con todos los juegos clásicos que se han convertido en cultura popular.

Objetivos secundarios:

- Dar la posibilidad de mostrar videos informativos o de divulgación.
- Hacer que sea lo más vistoso posible, dando posibilidad de consultar la historia de cada videojuego, como eran su marquesina, panel de control originales, etc...

Historia

3

Todo esto empezó cuando uno de mis profesores del master de videojuegos Federico Peinado ("Fede") nos propuso a todos los alumnos hacer un proyecto para su clase, cada uno era libre de elegir lo que quisiera. Cuando fui a hablar con él, me propuso varios trabajos, y hablando surgió la conversación de la máquina recreativa, yo ya había restaurado una y le había introducido un PC, y esa idea le entusiasmó a Fede, y me propuso hacer eso mismo para el museo de informática de la facultad.

Esto no me suponía ningún problema, todo lo contrario, me gustaba, así que nos pusimos manos a la obra, lo primero que hicimos fue reunirnos. En esta reunión (cuya asistencia fue escasa, debido a varios motivos), conocí al director del Museo, José Manuel Mendías "Mendi" y entre Fede, Mendi y yo decidimos que ordenador necesitábamos, que tipo de máquina queríamos, y los componentes que necesitábamos para poder conectar la máquina a un PC. Toda esta información yo la conocía gracias a una web que me ayudó a construir y restaurar la máquina anterior (Marcianitos).

Se nos planteaban diferentes problemas a la hora de decidir como conectar un PC con una recreativa, el primer problema era como conectar un monitor de Recreativa (que tiene una estructura similar a un TV) con el PC, una característica importante del monitor de la recreativa, es que funciona con un sincronismo vertical (V-Sync) de 15.7 KHz. (los hay que trabajan en frecuencias superiores, 25KHz o 31KHz) cuando lo normal de un monitor de PC son como mínimo 31 KHz., por lo que se nos planteaban varias formas de resolverlo:

- Conexión directa: Es decir, construir un cable desde el conector VGA del ordenador hasta el conector RGB del monitor arcade, esto supone un riesgo, ya que si le enviamos señales con una frecuencia superior a 15.7KHz podemos dañar el monitor arcade.
- Utilizando un J-Pac: Suponiendo que la máquina tuviese interfaz JAMMA (Interfaz creado por la industria japonesa para facilitar la actualización de las máquinas, ya que antes de esto, cada fabricante utilizaba uno diferente), podíamos utilizar esta circuitería especial para conectarlo.

Nos decidimos por lo más sencillo utilizar un J-Pac, además estaba el tema de la frecuencia que envía la tarjeta gráfica al monitor (y el J-Pac nos ayudaría a cortar frecuencias superiores a 15.7KHz que pueden ser perjudiciales), por lo que teníamos nuevamente varias opciones:

- Utilizar drivers específicos: Utilizar un driver especial que nos proporcione una señal de salida de 15.7KHz.
- Utilizar un emulador especial: Que realizan esa tarea por nosotros (pej: AdvanceMAME).
- Utilizar una tarjeta gráfica especial: Usar una tarjeta gráfica modificada por hardware para que emita señal de 15.7KHz.

Ya que teníamos que comprar el J-Pac, nos decidimos por lo más fácil, comprar la ArcadeVGA, que es una ATI Radeon 9200 modificada (ambos elementos se venden en

www.ultimarc.com).

Ya habíamos solucionado la cuestión de como conectar la máquina al PC, enviarle señal de video y recibir la entrada de los controles (gracias al J-Pac). Pero nos faltaba lo más importante, conseguir una máquina recreativa en un estado decente.

Después de varios contactos fallidos, y otra vez gracias a la gente del foro de [Marcianitos](#), conseguimos una información fiable, la dirección de **Recreativos Líder** en Moraleja de Enmedio, Madrid.

Hicimos unas llamadas, y quedamos para ir a echar un vistazo a su almacén (justo antes de navidades, el 22 o 23 de diciembre). En principio íbamos a ir Fede, Mendi y yo, finalmente fuimos Fede y yo, después de un viaje movido (nos perdimos un poco, y luego con las obras de Madrid y Fuenlabrada, pues...), conseguimos llegar a nuestro destino.

Estuvimos hablando con el dueño, y finalmente después de echar un vistazo a todo lo que podíamos nos decidimos por esta:



Una vez habíamos decidido la máquina que queríamos, solo necesitábamos algo de tiempo y que Mendi mandara el dinero de la máquina a la gente de R.Líder. Además de la máquina, ya que estábamos allí, se nos ocurrió que podría ser interesante traerse algún juego, no para jugar con él, ya que el objetivo es meter un ordenador dentro, sino para mostrar como eran las placas de juegos de las máquinas recreativas (Printed Circuit Board - PCB). Finalmente nos decidimos por un clásico que todo el mundo reconociese, el Tetris.

Foto PCB

A fecha de 1 de Febrero, por fin llega a la facultad la ansiada máquina, necesitamos hacer otra reunión para organizar el trabajo. En esta reunión nos organizamos de forma que pudiera ir al despacho donde estaba la máquina siempre que quisiera, con lo que tenía bastante libertad para hacer lo que quisiera con la máquina, finalmente después de horas y horas conseguimos tener la máquina funcionando.

Datos Técnicos _____ 4

SOBRE LA MÁQUINA ARCADE:

La máquina es una Video Líder con monitor Antares MTC 9110 de 25", fuente de alimentación Hantarex US250, 2 altavoces, conector JAMMA estándar y palanca de mandos y 4 botones por jugador de IL (Industrias Lorenzo).

Monitor.- Este monitor no solo es grande sino que soporta múltiples resoluciones de pantalla, actualmente esta puesta a 640x480, aunque puede ser mayor (800x600 o incluso 1024x768, no he probado resoluciones superiores pero igual las aguanta), pero eso nos obliga a aumentar las fuentes de tamaño para poder leer bien.



Estos son los controles de la pantalla que pueden ser manipulados para conseguir una mayor nitidez, colocación de pantalla etc...



Estos son los mandos a distancia, muy útiles porque nos permiten ver lo que estamos haciendo, para acceder a ellos es necesario levantar el Panel de Mandos (consultar *Apéndice A: Desmontar y acceder a todos los rincones*). Para más información sobre estos controles y sus funciones consultar el manual (*Apéndice B: Manuales y Documentación específica*).

Fuente de Alimentación.- Esta fuente de alimentación transforma la corriente de 220V a diferentes voltajes, 5V – 0V para los botones y palancas, -5V – 12V para el conector JAMMA y por lo tanto para alimentar a las placas de juego o para el monitor, podemos regular el voltaje que le llega a la placa de juego mediante este regulador:



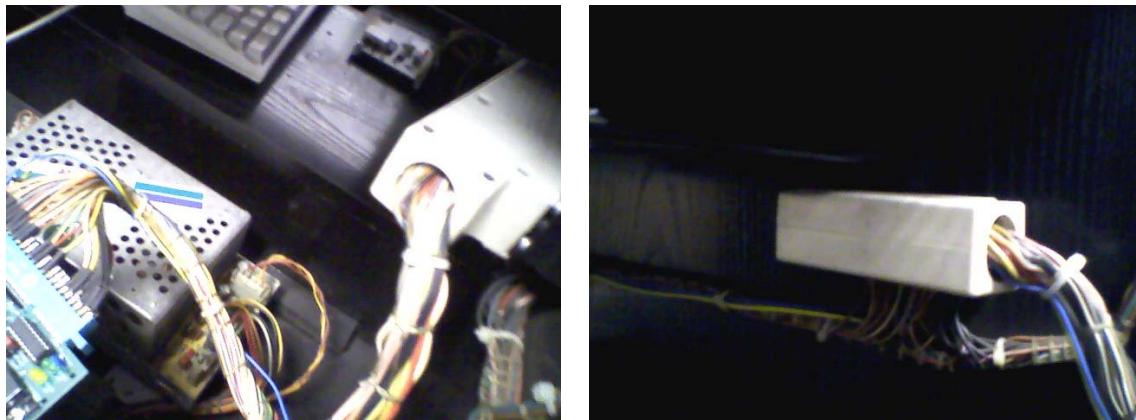
y podemos comprobar el voltaje actual en el display (foto de abajo) situado a la izquierda, cuando abrimos la puerta del monedero (más información *Apéndice A: Desmontar y acceder a todos los rincones*).



Altavoces.- Estos altavoces situados detrás de la marquesina estan puenteados de forma que la señal que le llega al altavoz izquierdo, es la misma que le llega al altavoz derecho, esto (supongo) que esta hecho porque el JAMMA soporta sonido mono y no estereo.

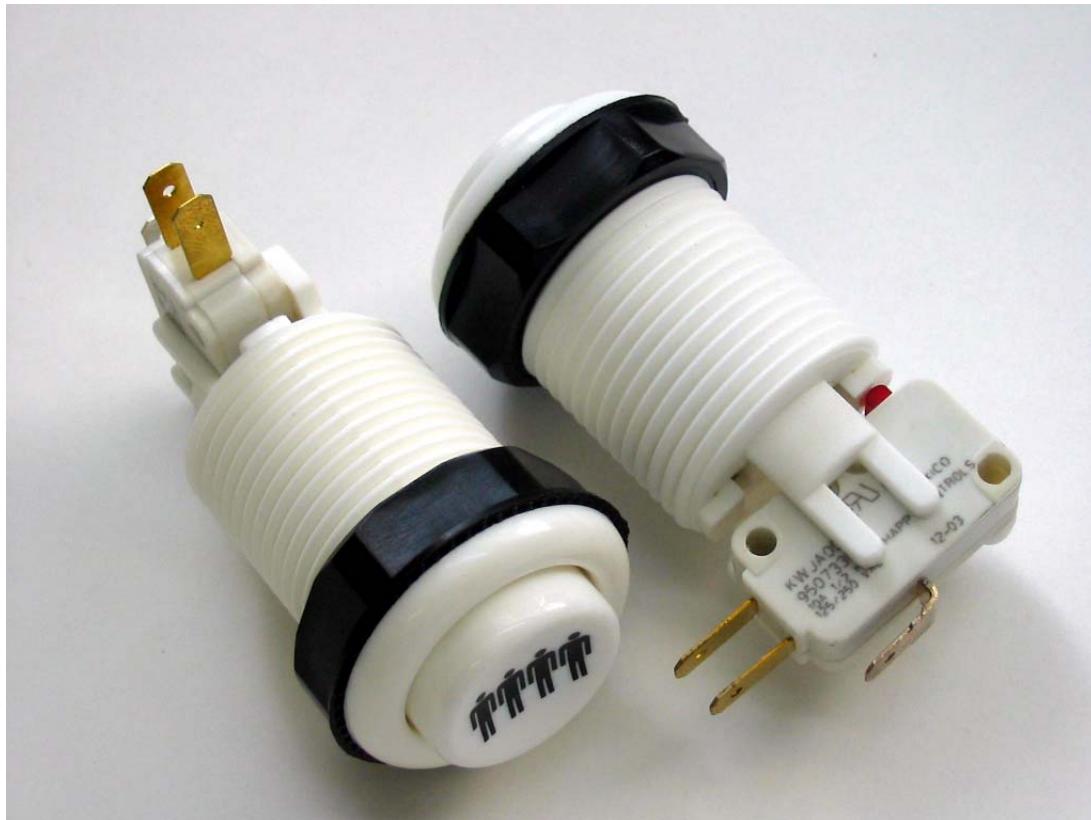


Conector JAMMA.- Este conector nos permite la comunicación de la máquina hardware con el software de los juegos, es decir nos permite interactuar con los juegos, ya que todo esta conectado al JAMMA, sonido, video, controles, monedero, incluso alimentación.



Aprovecharemos esta capacidad del conector JAMMA para conectar nuestro PC con el hardware de la máquina y poder comunicarnos en ambas direcciones. Para ello necesitaremos el **J-Pac** que explicaremos más abajo. (Para más información referente al conector JAMMA consultar el *Apéndice C: Conector JAMMA*).

Palancas y botones:- No hace falta decir mucho sobre estos, la máquina tiene “de serie” 4 botones por jugador además del de “start”. Tanto los botones como la palanca tienen un funcionamiento muy sencillo, de hecho es tan simple como que al mover la palanca, o pulsar un botón lo que hace es activar un microswitch, este microswitch está conectado a 5V y a 0V, de forma que al pulsarlo se cierra el circuito y el conector JAMMA detecta que el botón está pulsado, se puede hacer que la señal se envíe al pulsar o al soltar, dependiendo del microswitch.



Más información en el *Apéndice B: Manuales y Documentación específica*.

Monedero.- Una cosa curiosa de este modelo de máquina es que la no tiene monedero instalado dentro de la máquina, era un tanto especial en ese sentido, ya que se componía de 2 máquinas y entre medias un pequeño mueble con un monedero (se pueden observar las marcas de los agujeros por los que se conectaba el cableado) que permitía mandar los créditos a una u otra máquina. Para solucionar esta falta de monedero, se ha instalado un botón extra que al pulsarlo envía al interfaz JAMMA la señal de moneda introducida.



SOBRE EL HARDWARE INTERMEDIO:

J-PAC.- Como ya hemos dicho antes este es un hardware específico fabricado por Ultimarc (www.ultimarc.com) que nos permite la conexión entre máquina recreativa y PC, esta conexión es gracias a que la máquina tiene todas las conexiones con sus diferentes partes centralizadas en un interfaz JAMMA. Este hardware intermedio, además de proveernos de comunicación con la máquina nos permite añadirle funcionalidad extra, como por ejemplo ponerle más botones por jugador (hasta 8 botones) cosa que con el interfaz JAMMA no es posible (solo permite 4 botones, más información en *Apéndice C: Conector JAMMA*)



Referente al video, nos puede cortar señales peligrosas para el monitor arcade, de forma que si emitimos una señal con un sincronismo vertical superior al deseado, lo filtra y no lo deja pasar, simplemente necesitamos conectar la salida

de la tarjeta de video al conector del J-Pac, y con los “jumpers” seleccionamos las frecuencias que queremos dejar pasar.

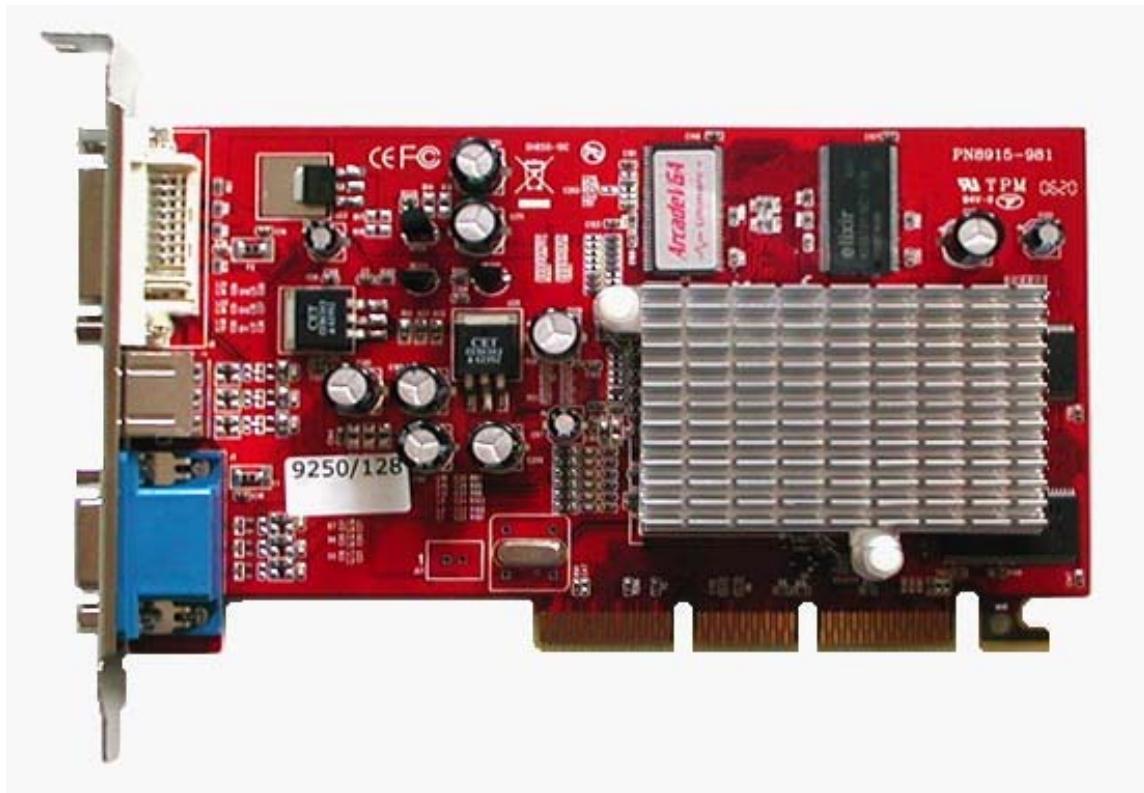
Referente a los controles, debemos conectar el teclado al ordenador utilizando el J-Pac como puente intermedio, es decir, conectamos el teclado al J-Pac y luego un cable del J-Pac al PC para conectarlos (en nuestro caso es de PS/2 a USB), esto se hace porque las pulsaciones de botones y palancas se “mapean” a comandos de teclado para el PC y además el J-Pac “elimina” la restricción de los teclados de pulsar varias teclas a la vez, de forma que podemos pulsar todas las que queramos. También referente a los controles, decir que podemos aumentar el número de botones por jugador como ya hemos comentado antes. Existe la posibilidad de configurar ese “mapeo” de teclas a botones con un software especial incluido en el CD de Ultimarc. (winipac.exe)

Referente al sonido, se puede usar el J-Pac para conectar los altavoces de la máquina con la tarjeta de sonido del PC, para ello simplemente se usa un cable que tenga un conector minijack para conectarse al PC y luego se conecta mediante las “clemas” que tiene el J-Pac para ello. Según el fabricante se obtiene peor calidad que conectándolo directamente, ya que el J-Pac no tiene ninguna circuitería específica de sonido, además el estándar JAMMA proporciona un sonido mono, por lo que nos interesa más conectar directamente los 2 altavoces a la salida de la tarjeta de audio.

Para más información referente al J-Pac consultar el *Apéndice B*.

SOBRE EL PC:

Hardware.- El PC es un Intel Pentium III a 650MHz. con una memoria SDRAM de 572 Mb (**que número más raro... hay pasa algo...? (como saca esos 60Mb...??)**). La placa base es una **nosequé** con tarjeta de audio integrada en la propia placa, lector de DVD “Pioneer 500M” en IDE2-slave, tarjeta de red “EtherLink XL” de “3Com” y dos discos duros, uno un “Maxtor” de 10Gb con el sistema operativo instalado en él, en IDE1-master y el otro disco duro, un “Seagate” de 40Gb. que tiene el resto de datos en IDE2-master. Por último la tarjeta de video es una “ArcadeVGA”, que es una “ATI Radeon 9250” con el “firmware” modificado para que directamente nos saque una señal de video con un v-sync de 15.7Khz. (más información en *Apéndice B.*)



Software.- En el primer disco duro tenemos instalado un Windows XP Profesional con el SP2 sobre un sistema de ficheros NTFS, además tenemos el “Partition Magic 8” para tareas de mantenimiento de discos.

En el segundo disco, tenemos el resto de los datos, esto es: el emulador que vamos a utilizar, el front-end, y todos los datos, ya sea las propias roms de los juegos, los snapshots, las fotos de las marquesinas y de los paneles de control, todos los ficheros de información referente a las roms: controles, historia, etc....

Sistema Operativo: El sistema operativo, como ya hemos comentado Windows XP, tiene una instalación básica, pero tiene modificado el arranque, para ocultar su presencia, utilizando un software adicional (instant sheller), este nos cambiará unas claves del registro de Windows (exactamente: [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon] la clave “shell” y la clave “userinit”). Además nos permite hacer otros cambios interesantes. (Para más información consultar el *Apéndice D: Software, configuración y uso*).

Además de esto se han hecho modificaciones en los servicios que se cargan al arrancar el Windows, eliminando servicios que no nos interesan y no vamos a utilizar, dejando así a Windows un poco más “ligero” y ahorrando memoria principal. Para saber qué servicios se han suprimido en la carga de Windows, ejecutar “msconfig.exe” e ir a la pestaña de “servicios”.

Emulador: El emulador que vamos a utilizar es el MAME 0.90 con optimizaciones para Pentium Pro (disponible en www.mame.net). La configuración es estándar con la excepción de la resolución que en lugar de

automática, esta forzada a 640x480, esto está hecho por falta de tiempo, pero debería de estar en auto, para aprovechar todas las ventajas de la “ArcadeVGA” y de las resoluciones que nos ofrece, pero para hacerlo bien necesitamos software adicional (avres, este software nos crea un fichero ini por cada rom que tengamos en el sistema, de forma que le asigna la resolución soportada por la AVGA más cercana a la resolución original del juego). Además necesitamos ejecutar MAME con las siguientes opciones “–ddraw –nohwstreich –switchres”. Para más información mirar *Apéndice B*, documentación de la ArcadeVGA. Esta aproximación no es totalmente exacta a la original, pero se acerca mucho y en muchos casos no se nota diferencias.

Es importante decir que se está utilizando la línea de comando para ejecutar el emulador, con las siguientes opciones activadas:

-skip_disclaimer : que nos ocultará la pantalla negra que aparece cada vez que cargamos un juego por primera vez, diciéndonos que debemos tener el juego original, etc....

-skip_gameinfo : que nos ocultará la pantalla negra que aparece siempre que cargamos una rom en MAME y nos informa de las características del juego, resolución original, etc....

Front-end: El front-end que vamos a usar es Mamewah v1.62b13, se trata de un software bastante probado, desarrollado en Visual Basic 6 y que aporta una funcionalidad bastante decente y lo más importante, es muy ligero en memoria, dejándonos más memoria para la emulación, como desventaja cabe destacar, que la configuración es un poco engorrosa, debido a que esta se hace en ficheros de inicialización a parte (ficheros “.ini”). Este front-end además aporta software adicional para creación de “skins” o “layouts” (“MameWah Layout Designer.exe”).

Además de este, se encuentra instalado otro front-end (MaLa), que tiene más funcionalidad, una configuración mucho más sencilla y la posibilidad de hacer layouts con animaciones que resultan más atractivos, pero la versión actual (v1.0 RC8) al ser una Release Candidate todavía no esta pulida y tiene algunos fallos, concretamente en este PC tenemos un fallo grave, que hace que el programa deba terminar al generar la lista de juegos general (más concretamente al hacer el “collecting names”), desconozco porque lo hace, puesto que he probado en otro ordenador y lo hace todo correctamente y funciona muy bien, si finalmente se consigue hacer que funcione, se utilizará este.

Se ha creado una lista de juegos con los “clásicos destacados” se pueden añadir o borrar juegos a esta lista de forma sencilla, estos son los que hay actualmente (esta lista es la que en teoría se mostraría por defecto, luego se pueden crear más listas con los juegos que se quieran):

Para más información consultar el *Apéndice D*, y la documentación que se encuentra en el mismo directorio que el front-end: “D:\mamewah\docs\”

(NOTA: tener en cuenta que alguna de esa documentación, tomada directamente de la página web del proyecto “Mamewah” – <http://mamewah.mameworld.net/> - se encuentra desfasada debido a que la versión que utilizamos no tiene toda la documentación y los ficheros de configuración pueden no coincidir).

Roms: Poco hay que decir de estas roms, se encuentran en la ruta “D:\mame090\roms” y se trata del romset completo de la versión 0.90 de MAME.

Otros ficheros: Además de estas roms existen otros ficheros de datos que pueden ser interesantes:

- **Snapshots.**- se trata de las capturas que se muestran en el front-end, correspondientes a cada juego, también se puede usar de salva-pantallas. Ubicadas en “D:\mame090\snap”.
- **Marquesinas.**- fotografías de las marquesinas originales de las máquinas, cada juego solía tener una marquesina diferente, aunque muchas veces los salones recreativos directamente no las cambiaban. Ubicadas en “D:\mame090\marquee”.
- **Paneles de Control.**- fotografías de los paneles de control originales, solo usadas en el front-end MaLa (al menos de momento). Están en “D:\mame090\cpansels”.
- **Controls.ini.**- este fichero contiene información de los controles de cada juego, es decir si utiliza spinner, joystick, para que se utiliza cada botón, etc... (solo en MaLa de momento). Ubicado en “D:\mame090”.
- **History.dat.**- este fichero tiene información de la historia de cada juego, clones, juegos recomendados por ser parecidos, trivia (curiosidades), historial de cambios con las diferentes versiones de MAME (solo se usa en MaLa, esta información se puede consultar online en www.mameworld.net/maws). Ubicado en “D:\mame090”.
- **Hiscore.dat.**- este fichero se utiliza para configuración de salvado de records, especifica como se almacena en la memoria de la placa original (y por lo tanto donde va a buscarla la rom) las puntuaciones más altas, permitiéndonos salvarlas. Se encuentra en “D:\mame090” y la información de los hi-scores se guarda en “D:\mame090\hi”.

Apéndices

5

Apéndice A: Desmontar y acceder a todos los rincones

Apéndice B: Manuales y Documentación específica

Apéndice C: Conector JAMMA

Apéndice D: Software, configuración y uso.

Apéndice E: Documentos Externos

Apéndice A: Desmontar y acceder a todos los rincones

En este apéndice vamos a explicar como se accede a cada parte de la máquina, desde acceso al monitor, o a la fuente de alimentación, hasta acceso a los altavoces o a la parte interna de los botones.

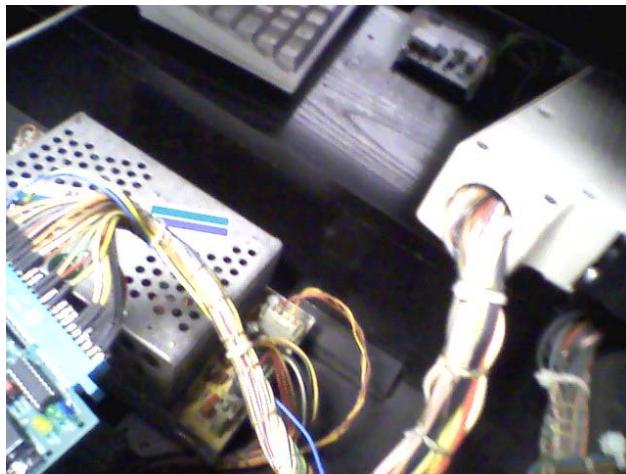


Empezaremos por lo más sencillo, quitar la puerta trasera de la máquina, para ello algo tan sencillo como una llave, desde esta puerta tendremos acceso a:

1. La fuente de alimentación.



2. Conector JAMMA.



3. Monitor y sus controles.

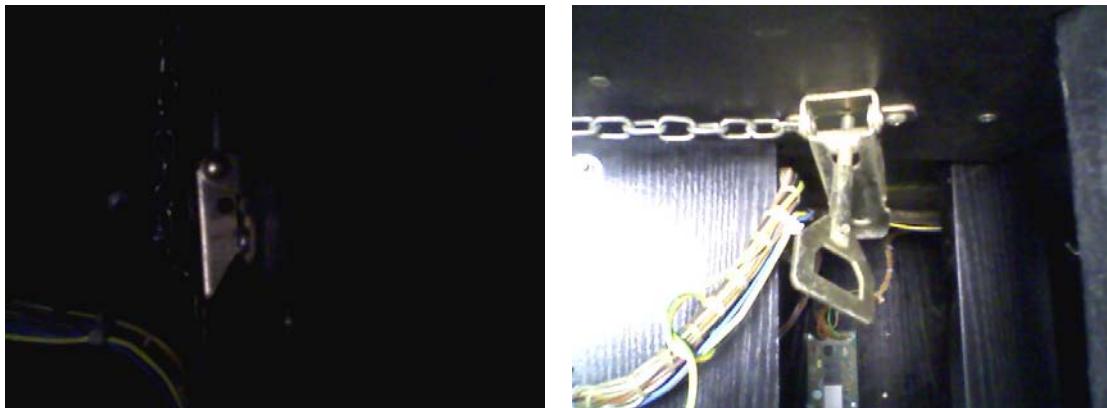


4. PC. Foto19

Para acceder al teclado y al ratón del PC sugerimos acceder por delante de la máquina, abriendo la puerta del monedero (la del botón azul), además podremos ver el display de voltaje de la fuente de alimentación.

Para manipular los controles del monitor se recomienda que se haga por medio de los mandos a distancia. Para ello tenemos que levantar el panel de control, con lo que además podremos acceder a las conexiones de los botones y microswitches.

Para hacer eso, necesitamos abrir la puerta del monedero, y situado a los lados del panel de control encontramos unos enganches, simplemente hay que levantarlos y desplazarlos hacia el centro del panel de control para desengancharlos.

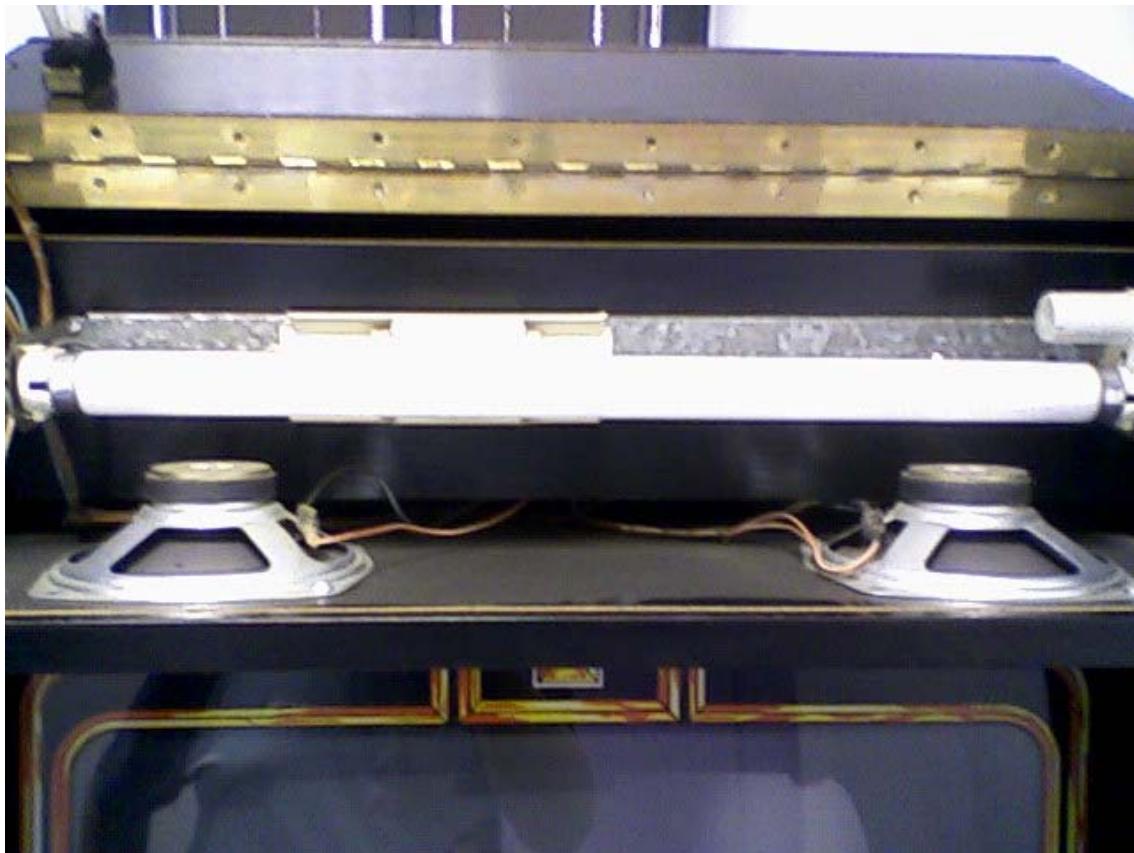


Una vez liberados los enganches, tiramos hacia nosotros de las palancas de mando y todo el panel se vendrá hacia nosotros, ya tenemos acceso a los mandos a distancia desde la parte delantera, y podemos manipular los controles del monitor viendo el efecto que producimos (nota, en estos controles no se incluyen el brillo ni el contraste, solo accesibles desde detrás). Además podemos sacar el cristal del bezel simplemente se coge con cuidado de la base del cristal, se levanta un poco y se tira hacia nosotros con cuidado de no romperlo.

En la puerta inferior (debajo de la del monedero) podemos abrirla, hay normalmente va el cajón de las monedas, en nuestro caso como ya hemos explicado lo especial del monedero, no hay nada, esta vacío y se puede emplear para guardar alguna cosa (por ejemplo el cable de alimentación para evitar que se pueda encender la máquina o lo que sea).

Finalmente para acceder a la marquesina y a los altavoces basta con quitar los 2 tornillos de la parte superior de la máquina y levantar la tapa.





Como se puede observar ya podemos acceder a los altavoces, y al tubo fluorescente de la marquesina, además de la propia marquesina que simplemente podemos sacar tirando hacia arriba.

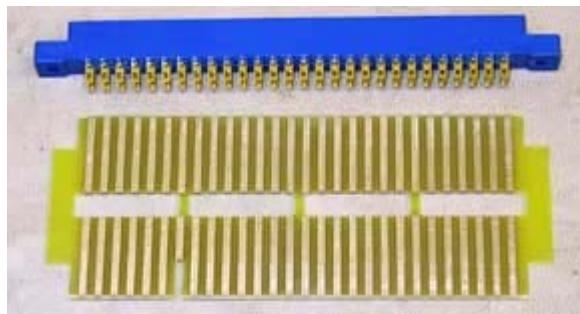
Apéndice B: Manuales y Documentación específica

Debido a que estos manuales se encuentran en formato “PDF” o incluso en formato “HTML” no están directamente incluidos en este documento y simplemente vamos ha reverenciarlos desde aquí (Si se desea tener impresa una guía completa se recomienda que se impriman también los documentos aquí referenciados).

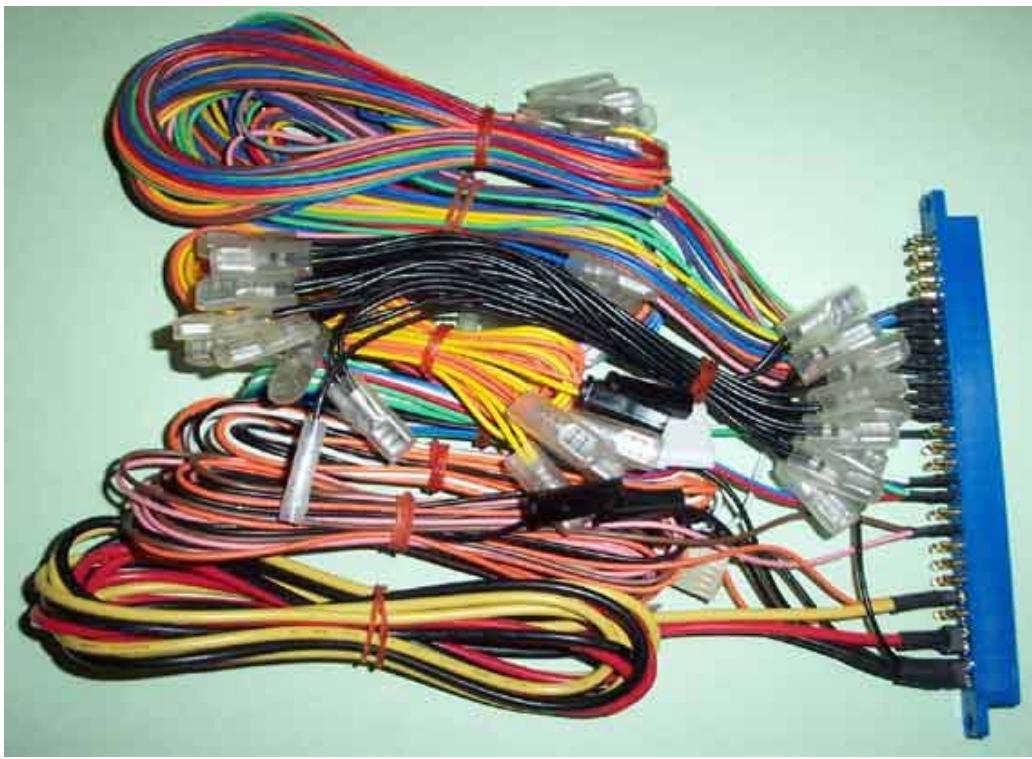
- [Botones de Industrias Lorenzo](#) [PDF]
- [Mandos de Industrias Lorenzo](#) [PDF] < Este documento no se corresponde con el que tenemos realmente instalado en la máquina, pero es el más parecido, y su montaje es similar, por lo que puede sernos útil.
- [Documentación del J-Pac](#) [HTML]
- [Documentación de la ArcadeVGA](#) [HTML]
- [Monitor Hantarex MTC 9110](#) [PDF]
- Fuente de alimentación Hantarex **US250** [PDF]
- **Mas?**

Apéndice C: Conector JAMMA

Cuando la industria de las máquinas recreativas comenzó su andadura (como suele pasar en casi todos los campos) cada fabricante tenía su propia circuitería, su manera particular de conectar los datos que estaban en los diferentes chips con los mandos que lo controlaban o con el video, etc. Esto suponía un problema para el mantenimiento de los mismos y obligaba a crear arquitecturas específicas para cada nuevo modelo, para solucionar esto, se necesitaba un estandar, el estandar creado por el JAMMA. El **JAMMA (Japanese Amusement Machine Manufacturers' Association)** es una asociación de fabricantes de máquina recreativas japoneses, que en 1985 crearon el “Standards for Elimination of Unhealthy Amusement Machinery” de lo que surgió el Interfaz JAMMA, un conector que unía todas las partes hardware de la máquina, las



centralizaba y las estandarizaba de forma que todos los fabricantes crearan sus juegos acogiéndose a este estándar, facilitando el mantenimiento y la explotación de las máquinas, permitiendo cambiar los videojuegos, manteniendo un mismo mueble.

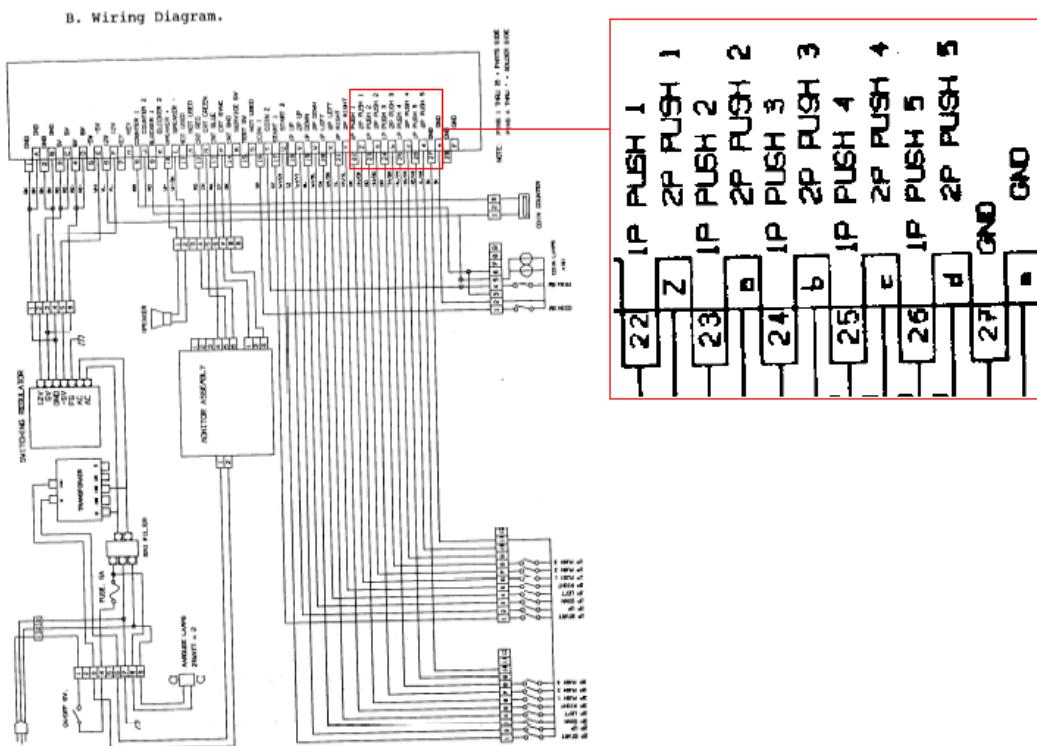


El conector JAMMA tiene el siguiente esquema de conexiones:

Standard JAMMA Edge Connector

SOLDER SIDE		PARTS SIDE	
Ground	A	1	Ground
Ground	B	2	Ground
+5V	C	3	+5V
+5V	D	4	+5V
-5V	E	5	-5V
+12V	F	6	+12V
KEY	H	7	KEY
Coin Meter 2	J	8	Coin Meter 1
Lockout 2	K	9	Lockout 1
Speaker -	L	10	Speaker +
Audio Ground	M	11	Audio +
Video Green	N	12	Video Red
Video Sync	P	13	Video Blue
Service Switch	R	14	Video Ground
Tilt Switch	S	15	Test Switch
Coin 2	T	16	Coin 1
2 Player start	U	17	1 Player start
Player 2 Up	V	18	Player 1 Up
Player 2 Down	W	19	Player 1 Down
Player 2 Left	X	20	Player 1 Left
Player 2 Right	Y	21	Player 1 Right
Player 2 Button 1	Z	22	Player 1 Button 1
Player 2 Button 2	Aa	23	Player 1 Button 2
Player 2 Button 3	Ab	24	Player 1 Button 3
Not used	Ac	25	Not used
Not used	Ad	26	Not used
Ground	Ae	27	Ground
Ground	Af	28	Ground

Como se puede observar, el estándar solo admite 3 botones por jugador, por lo que por ejemplo en el caso de los juegos MVS de NEOGEO que utilizan 4 botones (al igual que la consola NeoGeo) o en nuestro caso con nuestra máquina recreativa se tiene que usar conectores no estándar, en el caso de SNK se usaban el conector ‘25’ y el ‘c’ o incluso el ‘26’ y el ‘d’ para poner otro botón más como es el caso del “Ikari 3: The Rescue” como se puede observar en el gráfico de cableado del manual de instrucciones del juego:



En nuestro caso se están utilizando los conectores ‘25’ y ‘c’ para introducir el cuarto botón.

En nuestra máquina el interfaz JAMMA llega hasta un conector de 40 vías, a la que llegan todas las conexiones de la máquina, para conocer los detalles de este conector se puede ver la pegatina que hay dentro de la máquina:

MACHO LADO SOLDAD			
SA - ACCION	5	TH - CONTADOR	2
SA - STAR	2	2H - ACCION	3
SA - ACCION	1	3H - STAR	1
SA - ABAJO	3	4H - DERECHA	1
SA - ARRIBA	1	5H - CONTADOR	1
A - MONEDERO	1	6H - VERDE	1
A -	- 12V.	7H - ROJO	
	SV	8H	SV
ALTAVOZ	7	9H - MASA	
A - MASA		10H - MASA	
<hr/>			
ACCION	4	IV - ARRIBA	2
ACCION	2	2V - ABAJO	2
MONEDERO	2	3V - DERECHA	2
DERECHA	1	4V - IZQUIERDA	2
ALTAVOZ	1	5V - ACCION 1 (I)	
IZQ.		6V - ACCION 2 (S)	
OPCION		7V - ACCION 3 (D)	
RECREATIVOS LIDER, S.A.			
O	O	O	O
1A 1B 1H			
2A 2B 2H			
3A 3B 3H			
4A 4B 4H			
5A 5B 5H			
6A 6B 6H			
7A 7B 7H			
8A 8B 8H			
9A 9B 9H			
10A 10B 10H			

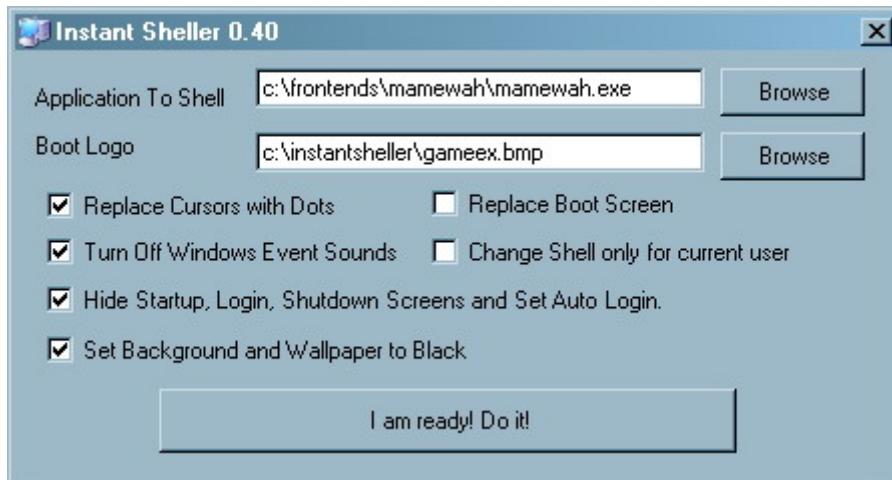
Más sobre el [estándar JAMMA \(en inglés\)](#).

Apéndice D: Software, configuración y uso

- Instant Sheller 2?
- M.A.M.E (Multiple Arcade Machine Emulator) 2?
- MameWah 2?
- MaLa (MAME and More Launcher) 2?
- WinIPac IPD (Interactive Panel Designer) 2?

Instant Sheller

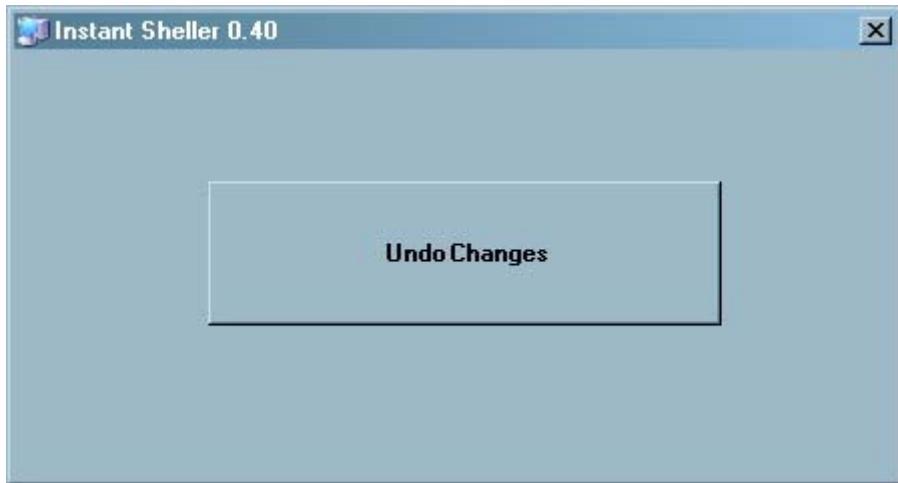
Esta utilidad, en realidad forma parte del conjunto de utilidades creadas para el front-end “GameEx” desarrollado por Tom Speirs y nos facilita el ocultar todo rastro de Windows.



Esta es la apariencia del programa, como podemos observar nos permite cambiar la aplicación de shell (que por defecto en Windows es explorer.exe) por la que queramos nosotros. Vamos a explicar que es cada opción y la configuración actual en el PC de la máquina.

- **Replace Cursor with dots** Oculta el puntero del ratón cambiando la típica flecha por defecto, por un punto negro que apenas se ve.
- **Turn Off Windows Event Sounds** Deshabilita los sonidos típicos de arranque y parada de Windows.
- **Hide Startup, Login, Shutdown Screens and Set Auto Login** Oculta las pantalla de bienvenida, Login, Cierre del sistema, y activa el autologin, es decir que el usuario actual entrará por defecto y sin password.
- **Set Background and Wallpaper to Black** Quita el fondo de escritorio y pone el fondo a negro.
- **Replace Boot Screen** Reemplaza la pantalla de carga por la que le digamos en "Boot Logo".
- **Change Shell only for current user** Si tenemos varios usuarios en el equipo, el usuario actual será el único al que se le modifique el shell por defecto.

Si por algún motivo no hemos conseguido lo que pretendíamos en un principio y queremos deshacer los cambios, bastará con volver a ejecutar el programa y nos aparecerá lo siguiente:



Bastará con pulsar el botón y nos dejará todo como lo teníamos antes.

Actualmente en nuestro PC no se está utilizando.



El emulador por excelencia, de sobra conocido por todo el mundo y que pretende una emulación lo más fidedigna posible, sin importar ni utilizar optimizaciones hardware, aunque estas supongan mejor rendimiento.

MAME is strictly a non-profit project. Its main purpose is to be a reference to the inner workings of the emulated arcade machines. This is done for educational purposes and to prevent many historical games from sinking into oblivion once the hardware they run on stops working. Of course to preserve the games, you must also be able to actually play them; you can consider that a nice side effect.

It is not our intention to infringe on any copyrights or patents on the original games. All of MAME's source code is either our own or freely available. To operate, the emulator requires images of the original ROMs from the arcade machines, which must be provided by the user. No portions of the original ROM codes are included in the executable.

Este es un proyecto de software libre y código abierto, para más información sobre licencias etc. consultar el documento “D:\mame090\docs\mame.txt”.

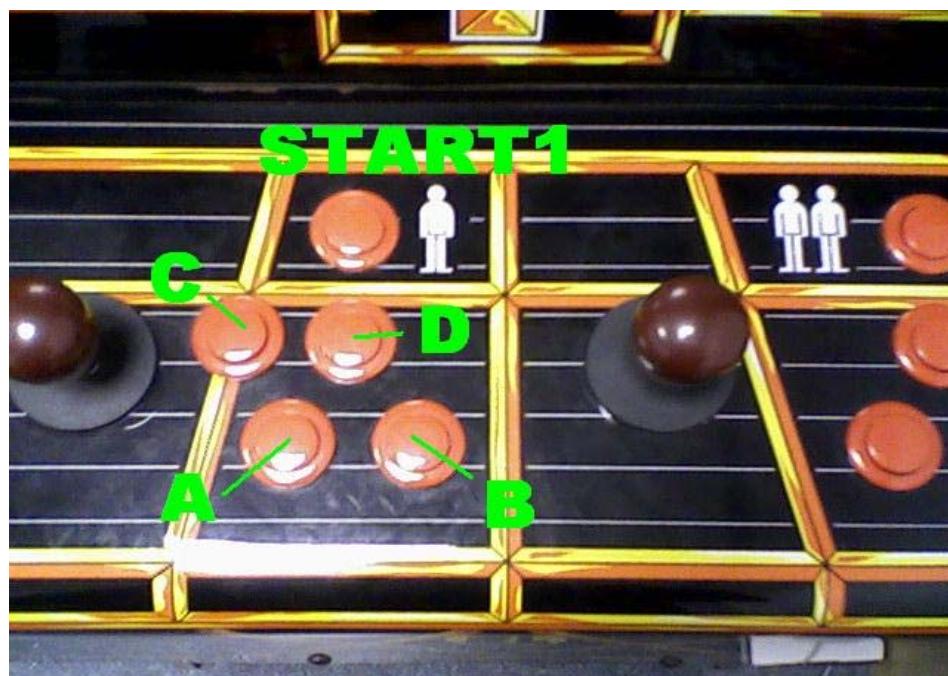
A día de hoy este emulador está por la versión 0.114, aunque nosotros usamos una versión mucho anterior, la 0.90, ya que disponemos del romset completo para la versión 0.90 y con esta versión conseguimos perfectamente el objetivo que se planteó inicialmente de conseguir la emulación en la máquina arcade de juegos clásicos

(anteriores a 1986).

La configuración del emulador se hace a través de un archivo .ini situado en la misma carpeta que el ejecutable, es decir en “D:\mame090\mame.ini”. Se recomienda la lectura de los siguientes documentos para la configuración y el uso del emulador:

- D:\mame090\docs\mame.txt Explica el propósito del proyecto MAME, su posición ante las licencias, y mucho más.
- D:\mame090\docs\windows.txt Explica las diferentes opciones que podemos utilizar cuando lanzamos el emulador desde línea de comandos, como ya hemos dicho anteriormente actualmente utilizamos las opciones “-skip_disclaimer” y “-skip_gameinfo”.

Los controles:



Botón	Funciones	Tecla
Start1 + Start2	- Salir del emulador	[Escape]
Start1 + 1P_ARRIBA		
Start1 + 1P_ABAJO	- Pausa en MAME	[P]
Start1 + 1P_IQZ		
Start1 + 1P_DCHA		
Start1 + 1P_A		
Start1 + 1P_B		
Start1 + 1P_C		
Start1 + 1P_D		
Start1 + 2P_ARRIBA		
Start1 + 2P_ABAJO		
Start1 + 2P_IQZ		
Start1 + 2P_DCHA		
Start1 + 2P_A		
Start1 + 2P_B		
Start1 + 2P_C		
Start1 + 2P_D		



Este es el front-end que usaremos por defecto y que se ejecutará nada más arrancar el Sistema Operativo. La configuración se hace a través de ficheros “.ini” como en el caso del propio MAME. La configuración principal del programa se encuentra en el fichero “mamewah.ini” se recomienda la lectura del documento “D:\mamewah\docs\ini.txt” ya que aunque la documentación de mamewah esta obsoleta, muchas de sus opciones son iguales o parecidas, aun así vamos a intentar explicar como están configurados los “inis” de la versión actual.

```
### mamewah.ini (mamewah v1.62) ###

### (Global) Screen-Saver Settings #####
delay 300          # Tiempo en segundos para activarse
slide_duration 3    # Tiempo en seg para cambiar de img

### Audio Settings #####
enable_sounds 1     # 1 - Habilitado; 0 - Deshabilitado
sound_volume 100    # Volumen maestro
music_path      # Ruta para música de fondo del f-e
enable_music 1      # 1 - Si; 0 - No
music_volume 100    # Volumen de la música

### Movie Settings #####
intro_movie_file   # Cuando arranca podemos iniciar un
                   # video, aquí se especifica donde está
exit_movie_file    # Lo mismo pero cuando cerramos
keep_movie_preview_aspect game_aspect # Podemos querer un video
para enseñar el juego, como los snaps, aquí se decide su aspecto, posibles
valores: no / video_aspect / game_aspect
delay_before_movie_preview 2
music_movie_mix    mute_movies # mute_movies/merge/pause_music
movie_volume 100
```

```

### Misc Settings ###
emulator_root_path          # Para todas 1 - Si; 0 - No
auto_setup_emulators         0      # Path donde tenemos nuestro MAME
automáticamente la configuración para cada emulador que tenemos.
theme                         # El layout que queremos poner
wrap_list                     0
remember_current_game        1      # Cuando volvamos a iniciar la
lista estará en el punto en el que la dejamos.
keep_image_aspect              1      # Para los snaps de preview
refresh_images                 0      # se recomienda que esté a 0
delay_before_history_dat     2      # Carga el history despues de 2 sec
safe_mode                      1
show_cursor                    1      # Para ocultar el ratón
keyboard_led_states           -,-- # Podemos encender y apagar los
leds del teclado, valores: --leave, 0=off, 1=on
lcd_com_port                   1
lcd_settings
exit_action                    choice # Cuando salimos podemos: choice /
windows / run_app / reboot / shutdown
maxlogsize                     100000 # Tamaño máximo del fichero de log

### External Application Settings ###
startup_app_commandlines       # Al iniciar mamewah podemos lanzar
una aplicación externa en línea de comando.
exit_app_commandlines          # Igual pero al salir
exit_and_run_app_commandlines

### Settings used by MAMEWAH ###
current_group                  # Esto lo modifica Mamewah
current_emulator                mame
locked                           0

```

Para configurar las opciones del emulador, debemos modificar otro archivo .ini situado en “D:\mamewah\config\mame\mame.ini”, vamos a explicar para que sirve cada cosa ya que la documentación esta obsoleta, como ya hemos dicho.

```

### mame.ini (mamewah v1.62) ###

emulator_title           M.A.M.E.      # Nombre que queremos
                           darle

### List Generation Settings ###
rom_path                  # Rutas de de roms, etc.
rom_extension             c:\Emuladores\mame090\roms
dat_file                  zip
nms_file                  c:\Emuladores\mame090\mameinfo.dat
catver_ini_file           c:\Emuladores\mame090\catver.ini
list_generation_method    rom_folder_vs_listxml # Metodo de
generación de la lista principal de juegos: rom_folder /
rom_folder_subfolders / dat_file / rom_folder_vs_dat_file /
rom_folder_vs_listinfo / verifysets_vs_listinfo / rom_folder_vs_listxml /
verifysets_vs_listxml

post_emulator_app_commandlines
general_app_commandlines

### Artwork Settings ###
artwork_image_paths        # Paths para las imágenes y el history
                           c:\Emuladores\mame090\snap # Aquí
podemos poner hasta 10 paths que se corresponden con los 10 "artwork image"
que aparecen en nuestro layout (tener en cuenta que 2 de estos artworks
serán usados para el movie_preview y para el history_dat
movie_preview_path         c:\Emuladores\mame090\previews
movie_artwork_no           1 # Igual que history_dat_artwork
enable_fuzzy_search        0
history_dat_file           # La ruta de nuestro history.dat
history_dat_artwork_no     2 # En el layout se define como un
                           artwork que esta numerado del 1 al 10, hay que decirle el que queremos.

### Execution Settings ###
pre_emulator_app_commandlines   # Si queremos ejecutar una aplicación
                                 antes que se ejecute el emulador
emulator_commandline         c:\Emuladores\mame090\mamepp.exe
[name]{autodosbox}            # Línea de comando para ejecutar
nuestro emulador, además de las opciones de MAME, tenemos variables de
mamewah como
[description] - full game name,
[name] - romname,
[year] - year made,
[manufacturer] - manufacturer,
[cloneof] - parent romname,
[romof] - bios/parent romname,
[screen] - screen type,
[orientation] - screen orientation,
[input] - input type,
[status] - driver status,
[color] - color status,
[sound] - sound status,
[category] - game category/genre.

Note that if '-clone [cloneof]' is present in the cmdline_format, and
the selected game is not a clone, then '-clone [cloneof]' will be ignored.
In order to alter the cmdline format used, the following flags can also
be used:

```

```
[rompath] - rom path excluding trailing '\\',
[romext] - rom extension excluding '.', 
{8.3string} - can be used to convert 'string' to a short (8.3) filename
              (note that quotes if applicable must be used outside of the
              {}'s),
{dosbox} - this specifies dos window is to be visible upon launching
            emulator/app,
{nodosbox} - this specifies dos window is to be hidden upon launching
            emulator/app,
{autodosbox} - this specifies dos window is to be visible only if
              necessary upon launching emulator/app (this should be used
              for M.A.M.E. only!!),
{safelaunch} - this 'safe launches' the emulator/app, use this if the
            emulator/app is not launching properly from MAMEWAH,
{nosafelaunch} - this launches the emulator/app as normal (use if
              possible),
{nomusic} - this pauses music playback while the emulator/app is
            running,
{music} - this allows music playback to continue while emulator/app is
            running,
{nocursor} - this hides the cursor while the emulator/app is running,
{cursor} - this makes the default cursor visible while the emulator/app
            is running,
{cursor filename} - this makes the cursor specified by 'filename'
            visible while the emulator/app is running.
```

Note: If no 'dosbox' flag is set then '{dosbox}' will be assumed,
 If no 'safelaunch' flag is set then '{safelaunch}' will be assumed,
 If no 'music' flag is set then '{nomusic}' will be assumed,
 If no 'cursor' flag is set then '{nocursor}' will be assumed.

Flags in {curley brackets} are not used in the actual commandline sent to
 the emulator.

```
### Screen-Saver Settings ###
enable_music_in_screensaver      # Se puede poner un screensaver diferente
                                # para cada emulador
                                1
select_random_game               0 # solo para el saver_type=
saver_type                       gamelist_slideshow # Tipo de screensaver
                                blank_screen / gamelist_slideshow / slideshow / movie / run_selected_game /
                                run_random_game / run_random_game_and_quit / run_random_game_using_wrapper /
                                launch_scr / power_saving / shutdown
                                gamelist_slideshow      Muestra las imágenes de los juegos en la lista
                                slideshow                Muestra todas las imágenes en orden aleatorio
                                que tengamos en un directorio (este viene
                                definido en el layout y en el fichero mame.ini)
movie_path                        # El path para el video, saver_type=movie
movie_fullscreen                  1
quit_delay                         0 # El tiempo que tarda en quitarse el
                                # video al volver del screensaver
saver_commandline

### Additional Settings ###
music_path
lcd_display_file_path
description_display                [description]{scroll}

### Settings used by MAMEWAH ###
current_list                      # No modificar, interno de Mamewah
                                0
```

```
### Additional Settings ###
music_path
lcd_display_file_path
description_display [description]{scroll}

### Settings used by MAMEWAH ###      # No modificar, interno de Mamewah
current_list                      0
```

Para crear las diferentes listas de roms, debemos crear antes de nada un fichero de texto en “D:\mamewah\config\mame”, bastará con hacer una copia del fichero “mame-0.ini” y renombrarlo cambiando el 0 por el siguiente número no usado. El fichero debe tener simplemente estas pocas líneas, y solo cambiaremos el nombre de la lista. Para añadir juegos a la lista debemos hacerlo una vez arrancado el programa.

```
### mame-0.ini (mamewah v1.62) ###

list_title                         All Games    # El nombre que queramos

### Games List Settings ###
cycle_list                          1           # Si activo, al llegar a un
extremo de la lista de juegos, saltamos al otro extremo, hacemos la lista
circular.

### Execution Settings ###          # mirar documentación mamewah.ini
pre_emulator_app_commandlines
emulator_commandline
post_emulator_app_commandlines

### Settings used by MAMEWAH ###      # No modificar, interno de Mamewah
current_game                        1234
```

Uso del front-end:



Los controles:

Botón	Funciones	Tecla
Start1	- Lanzar un juego / aceptar en menus / efecto "Shift" si se deja pulsado (cambia función de otros botones)	[1]
Start2	- Abrir Menú Principal / Cerrar Menús / Audio info en el screensaver.	[2]
1P_ARRIBA	- Subir en menús y listas de 1 en 1	[F.Arriba]
1P_ABAJO	- Bajar en menús y listas de 1 en 1	[F.Aabajo]
1P_IZQ	- Ir directamente arriba en menús / Cambiar de letra en listas	[F.Izq]
1P_DCHA	- Ir directamente abajo en menús / Cambiar a siguiente letra en listas	[F.Dcha]
1P_A	-	[Ctrl Izq]
1P_B	- Muestra o esconde la información de History.dat	[Alt Izq]
1P_C	-	[Space]
1P_D	- Selecciona un juego al azar	[Mays !]
2P_ARRIBA	- Subir en menús 1 página	[R]
2P_ABAJO	- Bajar en menús 1 página	[F]
2P_IZQ	-	[D]
2P_DCHA	-	[G]
2P_A	-	[A]
2P_B	-	[S]
2P_C	-	[Q]
2P_D	-	[W]
<hr/>		
Teclas especiales	Función	
<hr/>		
Backspace	- Abre el menú / cierra el menú	
Windows Izq	- Sale de Mamewah al entorno Windows	
Fin	- Sale de Mamewah y apaga el ordenador	

Estos controles pueden cambiarse en el fichero de configuración “D:\mamewah\config\mamewah.cfg”, el nombre de los eventos de input son estandar de DirectX, se recomienda consultar el documento “D:\mamewah\docs\ctrlr.txt” para más información al respecto.

Los menús:

Select Platform	- use this to select the current emulator (if you have >1)
Select Games List	- use this to select a custom list (if you have >1)
Find Game	- use this to find a specific game, by entering the name letter-by-letter
Select Random Game	- select a random game
Games List Options	
> Add Game to List	- add selected game to a custom list (selectable)
> Remove Game from List	- remove selected game from current list
> Generate List	- generate the list using the List Generation Method & settings in emuname.ini (MAIN LIST ONLY)
> Generate Filtered List...	- generate list using available filters (CUSTOM LISTS ONLY)
Audio Options	
> Volume Down	- }
> Volume Up	- } control de volumen
> Volume Mute	- }
> Play/Pause	- }
> Next Track	- }
> Previous Track	- } control music playback (if setup in mamewah.ini or emuname.ini)
> Fast Forward	- }
> Rewind	- }
Launch External Application	- use this to launch an application (if setup in emuname.ini)
Exit MAMEWAH	
> Exit to Windows	- }
> Exit & Run App	- } pretty obvious!
> Exit & Reboot	- }
> Exit & Shut-Down	- }



Este front-end esta instalado también en el PC, aunque como ya hemos explicado debido a un fallo al generar la lista principal de roms no lo usamos, este programa tiene más funcionalidad que Mamewah y es más fácil de configurar, por lo que si es posible en el futuro puede convenir cambiarlo, cuando se arregle el fallo que hemos detectado. Este front-end se compone de varios ejecutables de ayuda:

- MaLa.exe El ejecutable del front-end.
- MalaLayout Una utilidad para crear “Layouts” o “skins”
- MalaGamelist Este programa nos facilita la creación de listas para nuestro front-end permitiéndonos añadir y borrar juegos de forma muy rápida y fácil.
- MalaTree Utilidad para crear menús.
- MalaLayoutConverter Utilidad para convertir layouts de Mamewah a MaLa.

Cuando ejecutamos por primera vez el MaLa nos saldrá el cuadro de configuración, se recomienda leer la documentación de MaLa (D:\MaLa 1.0 RC8\docs) si no se entiende alguna de las opciones, es muy intuitivo y sencillo por lo que no debería suponer problemas, una vez configurado nos dirá que si queremos generar la lista de roms principal, tardará un rato y una vez generada nos lanzará el front-end. Si queremos cambiar algo de la configuración (que se almacena en “MaLa.ini”) basta con ejecutar el “MaLa.exe” y pulsar con el botón derecho del ratón en cualquier parte, nos abrirá un menú y podremos modificar las opciones.

Nota: a salido un nueva versión del front-end, la Release Candidate 9 que puede solucionar estos problemas que hemos comentado, pero debido a falta de tiempo y a que el “Mamewah” esta funcionando perfectamente a gusto del museo, no se ha mirado.



WinIPAC IPD replaces the older WinIPAC program. It's primary function is to tell the I-PAC and J-PAC boards which keyboard character(s) to send to the PC for each button or joystick on the control panel. You configure this information using this program and then download the settings to the I-PAC board, where they are held in EEPROM even after power off.

This program can only be used on boards that were shipped with this software on the accompanying CD (Jan 15th 2004 onwards). An [upgrade](#) is available for older boards. Check the upgrade page for details of how to determine your board version.

You can also tell the program to download various keycode settings to the board on-the-fly if you often run PC programs that need different panel configurations. All of this programmable functionality is entirely optional on the I-PAC range. If you leave the I-PAC set to MAME defaults (which most people prefer to do until they have built and tested their panel) you don't need to use this program. But that would be missing the fun!

So what is different about this program from all other existing keyboard encoder configuration software? The answer is this: If you think about the processes you go through to build a panel, the first thing you do is assemble the panel, then you wire all of the switches into the keyboard encoder. Then you might want to configure the encoder to send specific keycodes for each switch. Before WinIPAC IPD, you would have to refer to a note of how you have wired the panel, for example if you wanted to have a dedicated "quit" button for MAME, you would need to remember, for example, you have wired a button into the Player1 Switch 8 input, and this needs to be set to the keycode for the "esc" key. With WinIPAC IPD, the program already knows which

button you have wired to this input, and where it is physically located on the panel, and you can simply click on this button and assign the code. What's more, you can tell WinIPAC IPD that this is a "quit" button, by entering text, and it will remember this and display the text on a picture of the panel.

So WinIPAC IPD defines the steps in creating a panel as these: (all of which are optional)

Once only (unless you have multiple panels):

Step 1) Create a pictorial view of your panel design

Step 2) Tell the program how you have wired it to the interface

Step 3) Save the panel.

Then as many times as you need:

Step 1) Assign keycodes by clicking on each control

Step 2) Optionally assign display text to each control

Step 3) Save configuration

Step 4) Associate a configuration with an application in the separate Launcher program or in the MAMEWAH front end.

Other features:

Macros:

You can assign macros to any button. These are cumulative, each key is pressed and held, so you can program, for example, CTRL/ALT/DEL to one button.

On-Screen Keyboard Launcher:

Brings up a keyboard layout on the screen so you can configure using only a mouse or trackball.

Application Launcher:

Start a separate program which can be used to assign keycode files to any application and launch the app, having sent the file to the I-PAC board. As well as the Ultimarc Launcher, the MAMEWAH front end can do this. The Ultimarc Launcher will shortly be downloadable from this site.

Test Mode:

Use this to check operation of all buttons and switches.

Download to RAM instead of EEPROM (command line function):

This feature will be enabled in the next release. The feature is already

implemented in the I-PAC chip.

This enables the codeset to be stored in temporary storage on the I-PAC board instead of permanent storage.

Temporary reset to MAME defaults (command-line function):

This feature will be enabled in the next release. The feature is already implemented in the I-PAC chip.

You can tell the board to copy it's fixed MAME codeset into RAM temporary storage.

Installation:

Ensure you have Direct-X ver 8.0 or later installed. The Ultimarc CD has an auto-run menu which has an entry for installing WinIPAC IPD. Or, navigate to the setup.exe program in the WinIPAC folder. Depending on which fonts you have installed already on your system, you may need to re-boot for the table view to display properly. Note this program is not compatible with PS/2 keyboard/mouse dual-usage ports as found on laptops.

Using the program:

WinIPAC IPD is really two programs in one. So these will be covered separately. The first time you start the program, a menu will appear. This will ask you whether you want to use a default panel layout or not. The two choices are:

"Use Default" This skips all of the panel layout tools and goes straight into a table view which allows you to assign keycodes to the I-PAC inputs. This mimics the operation of the older WinIPAC program and is useful for quickly setting up your board when you know how the panel is wired and you don't intend to change it in the future.

"Customize Panel" This choice is much more fun! You will enter the first of the two sections of the program, Panel Design Mode, below.

PANEL DESIGN MODE:



This is the first program of the two sections that make up WinIPAC IPD. Normally you would enter this mode only once, when you build and wire a panel. The only time you may want to go through this more than once is if you have multiple panels. This is what you can do from this screen:

Design your panel.

The box in the top left corner contains a selection of different control types, selectable by using the left/right arrows. Choose a control, select a colour, then drag it onto the panel area on the right. Repeat this until the correct number and type of controls have been added. Then you can move and arrange the controls on the panel into an approximation of your actual panel layout. You can add more controls or remove them by dragging off the panel. You can change the colour of existing controls by clicking on it and clicking on a colour box on the left.

"Wire up" the panel.

Now you can tell WinIPAC IPD how you have wired each switch to the I-PAC board. Keep in mind that we are thinking about actual I-PAC connections here, as marked on the board, not keycodes or anything related to any application program. For each switch, look at which input it's connected to, then click on the switch, click on the Player (1,2,3,4) and click underneath on the switch number. For example if the I-PAC input is marked 1 DOWN, click on "P1", the "DOWN".

Save the panel.

You can save the panel layout and wiring information into a "CPL" file and give it a name. WinIPAC IPD keeps a running copy of the panel layout so this is only really

necessary if you have multiple panels.

Now you are done configuring the actual panel layout and wiring, you can forget about how you wired everything as it's all remembered by the program. You can now assign keycodes.

Keycode Mode.

The program will actually start in this mode normally, unless you have not yet configured a panel as described already. This mode allows you to configure the assignment of keyboard characters sent for every button and switch on your panel. This is something you might want to do more than once, because you might have many PC applications, emulators or other apps, which need different code configurations.

There are two ways to assign a keycode to each I-PAC input.

KEY CODE MODE:

Table View:



This is similar to the old WinIPAC program. You can assign a keyboard code to each I-PAC input. You need to know how your panel is wired up. To assign, click on the input required, then press a key. Further options such as Macros are available by right-clicking on the input.

Panel View:



This is much cleverer than table view. As you have already created a panel (see "panel design mode") the program knows what your panel looks like and how it's wired up. So you can simply click on a button on the picture of the panel and assign a keycode to it! The program looks at its records (actually the current CPL file) to see which I-PAC input you mean, so you don't need to know this! Just click on a button and press a key. Right-clicking gives more options such as macros.

Text View:



If you select this view, you can enter text against a button, and it's saved in the current layout file (IPC file) not sent to the I-PAC at all. Why would you want to do this? Well, think about it, let's say you have a flight simulator. When you load the correct IPC file, you don't have to remember that the middle button on the top row is "start engines", because you can type this into the panel layout and it will be remembered and displayed! Who wants to have to remember that "ctrl-S" is start engines? You don't have to.

The Launcher program can be used to start this type of PC application and automatically load the correct IPC file so the panel is displayed and configured correctly. You can save the current IPC file and give it a name, which could be the same name as the related application, but with an IPC extension. Even if you forget to save, the program keeps a running copy of the current configuration.

OTHER FUNCTIONS:

The three icons on the top of the window are:

Test mode:

For testing that every button and switch gives the correct keycode.

On-Screen Keyboard:

For launching on-screen keyboard (Windows XP and 2000 only)

Help:

Opens the current help screen.

Launcher:

This button simply starts a program installed in the WinIPAC folder, called "launcher.exe". The Ultimarc Launcher program would normally be installed here, but you could install (and rename) any other program such as a front end.

Reverse-lookup:

In table view, if you press any keyboard key or panel button, any input which is assigned to this key will be highlighted in green.

Understanding the I-PAC shift feature.

The I-PAC has a feature which allows you to select one button to be a shift button. This is normally "start1" but can be any button. You will want to limit the choice to buttons that are not used intensively during gameplay though. This button can still retain its normal use as well as being shift, or you can have a dedicated button just for shift. Pressing and holding this button invokes a different set of keycodes for all of the other buttons. These are normally used for control functions such as quit, coin, etc. That way you don't need special buttons for these functions and can retain the arcade-authenticity of your panel (have you ever seen a real arcade machine with a coin insert button on the panel!!??)

There is more in-depth information on how the shift function works [here](#).

Running from the command line.

You can run WinIPAC IPD from a command line giving an "invisible" way to download a specific keycode configuration to the I-PAC board. Just enter "WinIPAC nnn.IPC" where the correct IPC file is specified. This is how the Launcher program and MAMEWAH drives WinIPAC.

You can optionally tell WinIPAC IPD to leave a picture of the panel on the screen with user text which you have assigned to each control, by using the /t switch, ie "WinIPAC nnn.IPC /t"

Apéndice E: Documentos Externos

En este apéndice están reunidos todos los documentos externos de consulta que se han citado a lo largo de esta memoria. Esto está hecho así por comodidad para el lector, ya que no será necesario tener que buscar otros ficheros, o incluso tener que buscarlos en el PC de la máquina recreativa, si no se dispone de ellos en otro PC.

MAME:

mame.txt

```
M.A.M.E. - Multiple Arcade Machine Emulator
Copyright (C) 1997-2004 by Nicola Salmoria and The MAME Team
```

Many people have helped with this project--directly, or by releasing the source code for the drivers they have written. We are not trying to take credit that isn't ours. See the Acknowledgments section for a list of contributors. Please note, however, that the list is largely incomplete. Also see the comments in the source code to see the people who contributed to specific drivers. That list, too, may be incomplete. We apologize for any omission.

All trademarks cited in this document are property of their respective owners.

For usage instructions, please consult the file windows.txt

Usage and Distribution License

I. Purpose

MAME is strictly a non-profit project. Its main purpose is to be a reference to the inner workings of the emulated arcade machines. This is done for educational purposes and to prevent many historical games from sinking into oblivion once the hardware they run on stops working. Of course to preserve the games, you must also be able to actually play them; you can consider that a nice side effect.

It is not our intention to infringe on any copyrights or patents on the original games. All of MAME's source code is either our own or freely available. To operate, the emulator requires images of the original ROMs from the arcade machines, which must be provided by the user. No portions of the original ROM codes are included in the executable.

II. Cost

MAME is free. Its source code is free. Selling either is not allowed.

III. ROM Images

ROM images are copyrighted material. Most of them cannot be distributed freely. Distribution of MAME on the same physical medium as illegal copies of ROM images is strictly forbidden.

You are not allowed to distribute MAME in any form if you sell, advertise,

IV. Source Code Distribution

If you distribute the binary (compiled) version of MAME, you should also distribute the source code. If you can't do that, you must provide a link to a site where the source can be obtained.

V. Distribution Integrity

This chapter applies to the official MAME distribution. See below for limitations on the distribution of derivative works.

MAME must be distributed only in the original archives. You are not allowed to distribute a modified version, nor to remove and/or add files to the archive.

VI. Reuse of Source Code

This chapter might not apply to specific portions of MAME (e.g. CPU emulators) which bear different copyright notices.

The source code cannot be used in a commercial product without the written authorization of the authors. Use in non-commercial products is allowed, and indeed encouraged. If you use portions of the MAME source code in your program, however, you must make the full source code freely available as well.

Usage of the information contained in the source code is free for any use.

However, given the amount of time and energy it took to collect this information, if you find new information we would appreciate if you made it freely available as well.

windows.txt #**Using the program**

mame [name of the game to run] [options]

For example:

mame mspacman -nosound

...will run Ms Pac Man without sound

Configuration options

-createconfig / -cc

Creates the default MAME.INI file. All the following configuration options can be permanently changed by editing this configuration file.

-showconfig / -sc

Displays the current configuration settings. If you route this to a file, you can use it as an INI file. For example, the command:

```
mame -showconfig >mame.ini

is equivalent to -createconfig.

-showusage / -su

Displays a summary of all the command line options. For options that
are not mentioned here, the short summary given by "mame -showusage" is
usually sufficient.

-[no]readconfig / -[no]rc

Enables or disables the reading of the config files. When enabled
(which is the default), MAME reads the following config files in order:

- MAME.INI
- $MY_MAME.INI (i.e. if MAME was renamed MAME060.EXE, MAME
  parses MAME060.INI here)
- MAMED.INI (if this is a debug build, i.e. MAMED.EXE)
- VECTOR.INI (for vector games only)
- DRIVER.INI (based on the source filename of the driver)
- PARENT.INI (for clones only, may be called recursively)
- GAMENAME.INI

The settings in the later ini's override those in the earlier ini's.
So, for example, if you wanted to disable hardware stretch in the
vector games, you can create a VECTOR.INI with the "hwstretch 0" line
in it, and it will override whatever hwstretch value you have in your
MAME.INI.

-[no]verbose / -[no]v

Displays some diagnostic information at startup. IMPORTANT: when
reporting bugs, please run with mame -verbose and include the resulting
information. It can be very helpful in tracking down problems. The
default is off (-noverbose)

Windows path and directory options
-----
IMPORTANT: Please use the path, directory and file options ONLY in MAME.INI.
Otherwise, the outcome may be unpredictable and not consistent across
releases.

-rompath / -rp

You can give a path (list of directories separated by semicolons) of
directories to be searched for roms. The default is ROMS (that is, a
directory "roms" in the current directory).

-samplepath / -sp

You can give a path (list of directories separated by semicolons) of
directories to be searched for samples. The default is SAMPLES (that
is, a directory "samples" in the current directory).

-inipath

You can give a path (list of directories separated by semicolons) of
directories to be searched for INI-files. The default is .;ini (that
```

is, search in the current directory first, and then in the directory "ini" in the current directory).

-cfg_directory

After running a game, MAME stores some user changeable settings into cfg_directory/gamename.cfg. Additionally, on the first start of MAME a cfg_directory/default.cfg is created. The default is CFG.

-nvram_directory

The original hardware of some games use non-volatile ram chips to save their configuration. The contents of these are saved into this directory. The default is NVRAM.

-memcard_directory

The original hardware of some games supports so-called memory cards. The contents of these are stored here. The default is MEMCARD.

-input_directory

Input recordings created by "-record" will be saved here and loaded by "-playback". The default is INP.

-hiscore_directory

If there exists a file HISCORE.DAT in the MAME directory, highscores may also be saved for games that do not have the original hardware to support this. The quality of this feature depends on the accuracy of the externally available file HISCORE.DAT. The default is HI.

-state_directory

MAME supports state saving for some games. These states will be saved here. The default is STATE.

-artwork_directory

Some games used extra artwork not created by electrical circuits. MAME supports such artwork in PNG image format, located in this directory. The default is ARTWORK.

-snapshot_directory

Screenshots will go here. The default is SNAP.

-diff_directory

Directory for hard drive image difference files. The default is DIFF.

-ctrlr_directory

This directory holds controller-specific input port mapping .ini files. The default is CTRLR.

-cheat_file

The default is CHEAT.DAT. Modifying the default may not work at the moment.

-history_file

The default is HISTORY.DAT. Modifying the default may not work at the moment.

-mameinfo_file

The default is MAMEINFO.DAT. Modifying the default may not work at the moment.

Windows video options**-[no]autoframeskip / -[no]afs**

Automatically determines the frameskip level while you're playing the game, adjusting it constantly in a frantic attempt to keep the game running at full speed. Turning this on overrides the value you have set for -frameskip above. The default is ON (-autoframeskip).

-frameskip / -fs

Specifies the frameskip value. This is the number of frames out of every 12 to drop when running. For example, if you say -frameskip 2, then MAME will display 10 out of every 12 frames. By skipping those frames, you may be able to get full speed in a game that requires more horsepower than your computer has. The default value is -frameskip 0, which skips no frames.

-[no]waitvsync

Waits for the refresh period on your computer's monitor to finish before starting to draw video to your screen. If this option is off, MAME will just draw to the screen at any old time, even in the middle of a refresh cycle. This can cause "tearing" artifacts, where the top portion of the screen is out of sync with the bottom portion. Tearing is not noticeable on all games, and some people hate it more than others. However, if you turn this option on, you will waste more of your CPU cycles waiting for the proper time to draw, so you will see a performance hit. The default is OFF (-nowaitvsync). This option requires -ddraw.

-[no]triplebuffer / -[no]tb

Enables or disables "triple buffering". Normally, MAME just draws directly to the screen, without any fancy buffering. But with this option enabled, MAME creates three buffers to draw to, and cycles between them in order. It attempts to keep things flowing such that one buffer is currently displayed, the second buffer is waiting to be displayed, and the third buffer is being drawn to. Unfortunately, due to some DirectDraw issues, this doesn't always work out as well as it should. This is still being investigated. The default is OFF (-notriplebuffer). This option required -ddraw and -nowindow.

-[no]window

Run MAME in either a window or full screen. The default is OFF (-nowindow).

-[no]ddraw / -[no]dd

This is really just for testing. It disables the use of DirectDraw, which removes a lot of the features of the video system (-triplebuffer, -waitvsync, -resolution, -refresh, -switchres, -switchbpp, -resolution, -hwstretch all won't work). It will generally run really slowly. If anyone has a legitimate use for this option, we'd be curious to know. The default is ON (-ddraw).

-[no]direct3d / -[no]d3d

Use Direct3D to display the image. This does not render 3D games using 3D hardware, it just uses 3D hardware to display the image, and apply effects to it. -triplebuffer, -waitvsync, -resolution, -refresh, -switchres, -switchbpp, -resolution all work with this option. This option overrides -ddraw and implies -hwstretch. The default is OFF (-nodirect3d).

-[no]hwstretch / -[no]hws

MAME uses the hardware stretching abilities of modern graphic cards to scale the game image to the requested resolution. Depending on the quality of your graphic card and its drivers, this may be a fractional, antialiased scaling (nice) or an integer, blocky scaling (not so nice). In any case, you can disable this stretching altogether and let MAME do the scaling in software, which is probably slower, though. The default is ON (-hwstretch). Note: Vector games may actually look better with "-nohws".

-cleanstretch [option] / -cs [option]

Stretch the image to integer ratios only. This may leave a black border around the image in fullscreen mode. The options are:

none	disable. This will cause artifacts when using scanlines.
auto	let the blitter decide. The d3d module will select the best option.
full	always stretch to integer ratios both horizontally and vertically.
horizontal	always stretch to integer ratios horizontally.
vertical	always stretch to integer ratios vertically.

The default is AUTO (-cleanstretch auto). This option requires -ddraw or -direct3d.

-resolution wxh[xd] / -r wxh[xd]

Specifies an exact resolution to run in. In full screen mode, MAME will try to use the specific resolution you request. The width (w) and height (h) are required; the color depth (d) is optional. If omitted or set to 0, MAME will determine the mode automatically. For example, -resolution 640x480 will force 640x480 resolution, but MAME is free to choose the color depth. Similarly, -resolution 0x0x32 will force 32-bit color depth, but allows MAME to choose the resolution. The string "auto" is also supported, and is equivalent to 0x0x0. In window mode, this resolution is used as a maximum size for the window. The default is auto (-resolution auto). This option requires -ddraw for full screen resolution switching.

-refresh

Specifies a particular refresh rate to set your monitor to. If the refresh rate is not found, or if this parameter is 0, the default DirectDraw refresh rate is used. The default is -refresh 0. This option requires -ddraw and -nowindow.

-[no]scanlines / -[no]sl

Enables the classic MAME "scanlines" effect. The default is OFF (-noscanlines). This option requires -nohwstretch.

-[no]switchres

Enables resolution switching. This option is required for the -resolution option to switch resolutions in full screen mode. On many modern video cards with hardware stretching support, there is little performance penalty at higher resolutions, so it is nice to be able to get rid of the monitor resync time when you run in full screen mode. This is also useful on LCD displays with a fixed resolution. The default is ON (-switchres). This option requires -ddraw.

-[no]switchbpp

Enables color depth switching. This option is required for the -resolution option to switch color depths in full screen mode. This option is useful if you normally run at 16, 24, or 32 bit color depth on your desktop, and want to keep that color depth when you run MAME. The default is ON (-switchbpp). This option requires -ddraw.

-[no]maximize / -[no]max

Controls initial window size in windowed mode. If it is set on, the window will initially stretch to the maximum supported size when you start MAME. If it is turned off, the window will start out at the smallest supported size. The default is ON (-maximize). This option requires -window.

-[no]keepaspect / -[no]ka

Enables aspect ratio enforcement. When this option is on, the game's proper aspect ratio (generally 4:3 or 3:4) is enforced, so you get the game looking like it should. When running in a window with this option on, you can only resize the window to the proper aspect ratio, unless you are holding down the CONTROL key. By turning the option off, the aspect ratio is allowed to float. In full screen mode, this means that all games will stretch to the full screen size (even vertical games). In window mode, it means that you can freely resize the window without any constraints. The default is ON (-keepaspect).

-[no]matchrefresh

Enables refresh rate matching. When enabled, MAME will try to find the closest refresh rate match that is greater than the game's refresh rate. For example, if the game runs at 57fps, and you have 60, 70, 75Hz refresh rates, MAME will choose 60Hz. If the game runs at 61fps, then it will choose 70Hz. This is intended mainly for those who have tweaked their video card's settings to provide carefully matched refresh rate options. The default is OFF (-nomatchrefresh). This option requires

-ddraw and -nowindow.

-[no]syncrefresh

Enables speed throttling only to the refresh of your monitor. This means that the game's actual refresh rate is ignored; however, the sound code still attempts to keep up with the game's original refresh rate, so you may encounter sound problems. Again, this is intended mainly for those who have tweaked their video card's settings to provide carefully matched refresh rate options. The default is OFF (-nosyncrefresh). This option requires -ddraw.

-[no]throttle

Configures the default throttling setting. When throttling is on, MAME attempts to keep the game running at the game's intended speed. When throttling is off, MAME runs the game as fast as it can. The default is ON (-throttle).

-full_screen_brightness / -fsb

Some video cards adjust the brightness/gamma when they switch into full screen mode. To counteract this, you can specify the -full_screen_brightness value, which is a number between 0.1 and 2. 0.1 means 1/10th as bright as the default, and 2 means twice as bright. Note that the hardware support for this option is not present on all video cards. If you set a non-zero value, you may get a warning if MAME was unable to set the brightness on your card. The default is 0, which means that MAME will not attempt to adjust the brightness on your video card at all. This option requires -ddraw and -nowindow.

-frames_to_run / -ftr

This option can be used for benchmarking. It tells MAME to stop execution after a fixed number of frames. By combining this with a fixed set of other command line options, you can set up a consistent environment for benchmarking MAME performance.

-effect

Apply various blitting effects to emulate the look of Arcade monitors. Using these modes comes at the cost of reduced emulation speed, which is inherent to these modes. The current implementation is already as fast as possible, using self-modifying assembly code. For some of these, MMX is required. Possible values:

none	no effect (this is the default)
sharp	no effect, but gives a sharper image
scan25	25% scanlines
scan50	50% scanlines
scan75	75% scanlines
scan75v	75% scanlines (vertical)
rgb3	a certain "monitor mask"
rgb4	another one
rgb4v	another one (vertical)
rgb6	another one
rgb16	another one

```
rgbtiny      another one. Choose the one you like best.

-screen_aspect

Give a screen aspect ration in the form X:Y where X is the horizontal
and Y the vertical part. Examples are 4:3 for most resolution settings
on computer monitors, 5:4 for the 1280x1024 resolution or 3:4 for
turnable LCD's in the turned position. The default is 4:3.

Windows video options (Direct3D)
-----
-zoom [z] / -z [z]

Try to get a resolution that enlarges the emulated game z times. The
default is -zoom 2. MAME determines the resolution automatically. This
option only works with -direct3d.

-[no]d3dtexmanage

Let Direct3D handle conversion of the image to a texture (required to
display it with Direct3D). This is more compatible, but can be
(significantly) slower on some hardware. The default is ON
(-d3dtexmanage). Using -nod3dtexmanage can eliminate a loss of
performance that may occur with some graphics cards when using
-direct3d. This option requires -direct3d.

-d3dfilter [f] / -flt [f]

Select the type of filtering to apply to the image when stretching. 0
is point filtering, 1 is bi-linear filtering, 2 is bi-cubic filtering
(flat kernel), 3 is bi-cubic filtering gaussian kernel), 4 is
anisotropic filtering. Note that very few graphics cards support
bi-cubic filtering, and that older graphics cards may not support
anisotropic filtering. The default is bi-linear filtering (-d3dfilter
1). This option requires -direct3d.

-d3dfeedback [i]

Select the feedback effect intensity in percentages. 0 is disable,
higher values feed back the previous frame to the current one at the
specified intensity. The default is DISABLE (-d3dfeedback 0). This
option requires -direct3d.

-d3dscan [i]

Select the scanline effect intensity in percentages. 0 is black lines
between scanlines, 100 is disable scanline effect. The default is
DISABLE (-d3dscan 100). This option requires -direct3d.

-[no]d3deffectrotate

Apply rotation so that any effects are rotated along with the game
image. The default is ON. This option requires -direct3d.

-d3dprescale [option]

Pre-scale up the image with point filtering before fitting the image
to the screen. This gives an only slightly fuzzy image even at high
display resolutions. The options are:
```

```

none      disable.
auto     adaptively apply a moderate pre-scale effect,
          depending on other effects used.
full     adaptively apply an aggressive pre-scale effect.
n        (where 2 <= n <= 4) pre-scale the image n times.

```

The default is AUTO (-d3dprescale auto). This option requires -direct3d.

-d3deffect [preset]

Select an effect preset. Valid presets are:

```

none      no effect (this is the default)
sharp    no effect, but gives a sharper image
scan25   25% scanlines
scan50   50% scanlines
scan75   75% scanlines
scan75v  75% scanlines (vertical)
rgbmicro a certain "monitor mask"
rgbtiny  another one
rgb3     another one
rgb4     another one
rgb4v    another one (vertical)
rgb6     another one
rgb16    another one
rgbminmask another one
dotmedmask another one
rgbmedmask another one
dotmedbright another one (extra bright)
rgbmaxbright another one (extra bright)
aperturegrille another one. Choose the one you like best.
auto     adaptively selects a special preset based on
          zoom level.

```

The default is none (-d3deffect none). This option requires -direct3d.

-d3dcustom [custom preset]

Supply a custom preset. The format is <-d3dcustom
1,0,0x0100,0xC0,0,0,0x22,-1,8x8_mame_rgbtiny.rgb>. The meaning
of the numbers is:

```

1      RGB effects mode: 0 is off, 1 is multiply mode, 2
          is add and multiply mode.
2,3,4,5  pattern preprocessing; use mask, white level, black
          level, desaturation, respectively.
6      image attenuation
7      feedback (0 is don't use)
8      prescale. bit 4: prescale horizontally, bit 0:
          prescale vertically.
9      maximum scanline intensity.
10     pattern filename, including path.

```

The filename must start with XxY, where X and Y are the dimensions of the pattern in pixels. Each pixel is represented by 4 bytes, red, green, blue, and mask, respectively. mask is an on/off value, where 0 is off. This option requires -direct3d.

```
-d3dexpert [settings]
```

Undocumented settings intended mostly for developers. This option requires -direct3d.

Windows misc options

```
-[no]sleep
```

Since MAME is running in a multitasking environment, it should be nice to its fellow processes. Therefore, if MAME does not need all processing power to emulate a game, it gives back already granted processor time to the operating system. This may not work in every case, so it can be disabled and MAME turned once again into a CPU hog. The default is ON (-sleep).

```
-[no]rdtsc
```

Prefer RDTSC over QueryPerformanceCounter for timing. The default is OFF (-nordtsc). -rdtsc sometimes causes problems on laptops, and -nordtsc may cause problems on desktop computers. If MAME does not operate smoothly (and you are sure you have enough CPU power for the emulation), try toggling this.

```
-[no]high_priority
```

Increases the thread priority so MAME runs better. The default is OFF (-nohigh_priority).

Windows sound options

```
-audio_latency
```

This controls the amount of latency built into the audio streaming. By default MAME tries to keep the DirectSound audio buffer between 1/5 and 2/5 full. On some systems, this is pushing it too close to the edge, and you get poor sound sometimes. The latency parameter controls the lower threshold. The default is 1 (meaning lower=1/5 and upper=2/5). Set it to 2 (-audio_latency 2) to keep the sound buffer between 2/5 and 3/5 full. If you crank it up to 4, you can definitely notice the lag.

```
-wavwrite <filename>
```

Writes the final mixer output to the given <filename> in .WAV format.

Input device options

```
-[no]mouse
```

Controls whether or not MAME looks for a mouse controller to use. When this is enabled, you will not be able to use your mouse in Windows while playing a game. If you want to get control of your computer

back, you will need to pause the game, or quit. The default is OFF (-nomouse).

-[no]joystick / -[no]joy

Controls whether or not MAME looks for joystick/gamepad controllers. When this is enabled, MAME will ask DirectInput about which controllers are connected. The default is OFF (-nojoystick).

-[no]lightgun/ -[no]gun

Controls whether or not MAME treats mouse inputs as lightgun inputs. Note that most lightguns map to the mouse, so using -lightgun and -mouse together may produce strange results. The default is OFF (-nolightgun).

-[no]dual_lightgun / -[no]dual

Controls whether or not MAME attempts to track two lightguns connected at the same time. This option requires -lightgun. The default is OFF (-nodual_lightgun).

-[no]offscreen_reload / -[no]reload

Controls whether or not MAME treats a second button input from a lightgun as a reload signal. In this case, MAME will report the gun's position as (0,0) with the trigger held, which is equivalent to an offscreen reload. The default is OFF (-nooffscreen_reload).

-[no]steadykey / -[no]steady

Some games require two or more buttons to be pressed at exactly the same time to make special moves. Due to limitations in the PC keyboard hardware, it can be difficult or even impossible to accomplish that using the standard keyboard handling. This option selects a different handling that makes it easier to register simultaneous button presses, but has the disadvantage of making controls less responsive. The default is OFF (-nosteadykey).

-[no]keyboard_leds / -[no]leds

Since MAME uses an input device (keyboard) for output, this sort of belongs here. Using this option enables/disables simulation of the game LEDs by the keyboard LEDs. This works fine, but can lead to problems after exiting MAME (i.e. CAPS LOCK remains ON), so you can disable it here. The default is ON (-keyboard_leds).

-led_mode <ps/2|usb>

For Windows NT and later systems, determines how LEDs on the keyboard are controlled. In PS/2 mode, MAME uses the low-level keyboard driver directly to control the LEDs. In USB mode, MAME attempts to post fake keypresses on the Caps Lock, Scroll Lock, and Num Lock keys in order to induce the LEDs to change. If you have a PS/2 keyboard, PS/2 mode is the more reliable approach. Default is PS/2.

-a2d_deadzone / -a2d

If you play with an analog joystick, but the game requires digital input, MAME needs to convert the signals. Here you can give the ratio

of movement along an axis that accounts for a digital signal. This option expects a float in the range of 0.0 to 1.0. The default is 0.3. Note: the current handling is not precise, as it does also implicitly set the size of the arc which is interpreted as diagonal movement. This is a known problem but it requires some major rework of the input handling code to fix it.

-ctrlr <controller>

Enables support for special controllers. Configuration files are provided for:

hotrod	HotRod
hotrodse	HotRod SE
slikstik	SlikStik
xarcade	X-Arcade

-paddle_device <keyboard|mouse|joystick|lightgun> / -paddle
-adstick_device <keyboard|mouse|joystick|lightgun> / -adstick
-pedal_device <keyboard|mouse|joystick|lightgun> / -pedal
-dial_device <keyboard|mouse|joystick|lightgun> / -dial
-trackball_device <keyboard|mouse|joystick|lightgun> / -trackball
-lightgun_device <keyboard|mouse|joystick|lightgun>

Each of these options controls autoenabling the mouse, joystick, or lightgun depending on the presence of a particular class of analog control for a particular game. For example, if you specify the option -paddle mouse, then any game that has a paddle control will automatically enable mouse controls just as if you had explicitly specified -mouse.

Note that these controls override the values of -[no]mouse, -[no]joystick, etc.

-digital <all|none|j<N>[a<M>[a<K>...]][,j<X>[a<Y>...]][,...]>

Controls which joystick axes are considered digital. If MAME knows that a given joystick axis is digital, it can make much better decisions about how to treat that axis. If you are using a digital gamepad with MAME, you will want to use this option. There are a number of ways to do this:

-digital all means that all axes of all connected joysticks will be treated as digital.

-digital none means that all axes of all connected joysticks will be treated as analog (this is the default behavior).

-digital j2 will treat all axes of joystick #2 as digital; axes on all other joysticks will be treated as analog.

-digital j1a0a1 will treat axis 0 and 1 on joystick #1 as digital; all other axes will be treated as analog.

-digital j1a0a1,j2a5 will treat axis 0 and 1 on joystick #1 as digital, as well as axis 5 on joystick #2; all other axes will be treated as analog.

Make use of the information provided by -verbose to determine which joysticks and axes you should be configuring this way.

MAME core video options

```
-----  
-[no]norotate  
-[no]ror  
-[no]rol  
-[no]autoror  
-[no]autorol  
-[no]flipx  
-[no]flipy
```

These are the standard MAME rotation options. They are all OFF by default.

-debug_resolution wxh / -dr wxh

Specifies the resolution of the debugger bitmap in the debug build. This controls how big the debugger window will be, and therefore, how many rows/columns it will display. The default value is 0x0, which sets the automatic 640x480 default debugger size. Note that the debugger is only available when running in a window.

-gamma

This controls the global gamma correction in the game. It is the same gamma that is applied when you bring up the on-screen-display within MAME. The default is 1.0.

-brightness / -bright

This controls the global brightness correction in the game. It is the same brightness that is applied when you bring up the on-screen-display within MAME. The default is 1.0.

-pause_brightness

This controls the brightness level when MAME is paused. The default value is 0.65.

MAME core vector options

```
-----  
-[no]antialias / -[no]aa
```

Antialiasing for vector games. The default is ON (-antialias).

-[no]translucency / -[no]tl

Enables or disables vector translucency. Colors of crossing vector beams will be merged. The default is ON (-translucency).

-beam

Sets the width in pixels of the vectors. This option expects a float in the range of 1.00 through 16.00 as argument. The default is 1 (1 pixel wide).

-flicker

Makes the vectors flicker. This option requires a float argument in the range of 0.00 - 100.00 (0=none, 100=maximum). The default is 0.

-intensity

Sets the intensity correction for the beam. Higher values give a brighter beam. This option expects a float in the range of 0.5 through 3.0 as argument. The default is 1.5.

MAME core sound options

-samplerate / -sr

Sets the audio sample rate. Smaller values (e.g. 11025) cause lower audio quality but faster emulation speed. Higher values (e.g. 44100) cause higher audio quality but slower emulation speed. The default is 44100.

-[no]samples

Use samples if available. The default is ON (-samples).

-[no]resamplefilter

If the samplerate of the original hardware does not match the samplerate actually used by an integer factor, apply a filter. This is more faithful to the original sound, however, the sound may not be as crisp. The default is ON (-resamplefilter).

-[no]sound

Enable or disable sound altogether. The default is ON (-sound).

-volume / -vol

Sets the startup volume. It can later be changed with the On Screen Display (see Keys section). The volume is an attenuation in dB: e.g., "-volume -12" will start with -12dB attenuation. The default is 0.

MAME core misc options

-[no]artwork / -[no]art

Enable or disable usage of additional artwork (backdrops, overlays, etc.). The default is ON (-artwork).

-[no]use_backdrops / -[no]backdrop

Enables/disables the display of backdrops. The default is ON (-use_backdrops).

-[no]use_overlays / -[no]overlay

Enables/disables the display of overlays. The default is ON (-use_overlays).

```
-[no]use_bezels / -[no]bezel
    Enables/disables the display of bezels. The default is ON
    (-use_bezels).

-[no]artwork_crop / -[no]artcrop
    This will crop the artwork to the game screen area only. The default
    is
    OFF (-noartwork_crop).

-artwork_resolution / -artres
    Scale the game by this factor to accommodate for higher resolution
    artwork. The default is 0 (auto).

-[no]cheat / -[no]c
    Cheats, like the speedup in Pac Man or the level-skip in many other
    games, are disabled by default. Use this switch to turn them on. The
    default is OFF (-nocheat).

-[no]debug
    Activates the integrated debugger. During emulation, press the tilde
    key (~) to activate the debugger. This is available only if the
    program
    is compiled with MAME_DEBUG defined. The default is OFF (-nodebug).

-playback / -pb
    Expects a filename as parameter. Play back all game inputs from from
    file INP/filename.inp. The gamename is contained in this file,
    therefore a gamename needs not be given on the commandline.

-record / -rec
    Expects a filename as parameter. All game inputs are written to the
    file INP/filename.inp.

-[no]log
    Creates a log of illegal memory accesses in ERROR.LOG. The default is
    OFF (-nolog).

-maxlogsize
    Expects a number as parameter. If the size of error.log (in kilobytes)
    hits that number, MAME will exit. By default, this is set fairly low
    (10 MB). This is in effect only if you are actually generating the
    error.log file.

-[no]oslog
    Outputs error.log data to the Windows debugger instead of to a file.
    Default is OFF (-nooslog).

-[no]skip_disclaimer
    Forces MAME to skip displaying the disclaimer screen. The default is
    OFF (-noskip_disclaimer).
```

```
-[no]skip_gameinfo
Forces MAME to skip displaying the game info screen. The default is OFF
(-noskip_gameinfo).

-[no]crconly
Tells MAME to only use CRCs for ROM integrity checks. This disables checking against the more reliable SHA1 hashes. Default is OFF (-nocrconly).

-bios
Expects a BIOS name as the parameter. -listxml will list the possible BIOS names for a game/system.

-state <slot>
Starts MAME and loads a saved state immediately from the specified slot.

Other MAME frontend options
-----
Note: By default, all the '-list' commands below write info to the screen. If you wish to write the info to a textfile instead, add this to the end of your command:
> filename
...where 'filename' is the textfile's path and name (e.g., c:\mame\list.txt).

-help / -?
Displays current MAME version and copyright notice

-[no]clones
Used together with the list commands, enables/disables listing of alternate ("cloned") versions of the same game. The default is ON.

-list / -ls
Displays a list of currently supported games.

-listfull / -ll
Displays a list of game directory names + descriptions.

-listgames
List the supported games, year, manufacturer.

-listdetails
Displays a detailed list of drivers and the hardware they use.
```

-listsourcefile

Displays the source file (in the drivers directory) containing the driver for the specified game. Useful for finding which driver a game runs on in order to fix bugs.

-listxml / -lx

List comprehensive details for all of the supported games. The output is quite long, so better redirect this into a file. The output is in XML format.

-listclones / -lc

Lists clones of the specified game. When no game is specified, this generates a list of all MAME-supported clones.

-listsamdir

List directory names for samples. Some clones can share their samples with their parents (no pun intended).

-listcrc

List CRC32 checksums of rom files

-listdupcrc

Find romfiles with same CRC32 (for developers).

-listroms

Displays ROMs required by the specified game.

-listsamples

Displays samples required by the specified game.

-verifyroms

Checks specified game(s) for missing and invalid ROMs. Adding "*" checks all available games.

-verifysets

Checks specified game(s) and reports its status. Adding "*" checks all available games. Terse output.

-vset

Checks specified game(s) and reports its status. Adding "*" checks all available games. Slightly more verbose output.

-verifysamples

Check selected game for missing samples. Adding "*" checks all available samples.

-vsam

Check selected game for missing samples. Adding "*" checks all available samples.

-romident

Attempts to identify ROM files, if they are known to MAME, in the specified .zip file. Can be used to try and identify ROM sets taken from unknown boards.

-isknown

Very terse romident.

-crconly

Uses only CRC-32 for all integrity checks. Disables the use of SHA-1 checks when loading the ROMs.

-gamelist

Used internally to create gamelist.txt

-sortname

-sortdriver

Used internally for creating gamelist.txt

-listcpu

-listcpuclass

-listgamespersourcefile

-listnosound

-listnvram

-listpalettesize / -lps

-listromdistribution

-listromnumber

-listromsize / -lrs

-listsound

Lists various statistical data.

-listwrongmerge / -lwm

-wrongfps

-wrongorientation

Attempts to catch careless developers.

Keys

Tab Toggles the configuration menu

Tilde Toggles the On Screen Display. Use the up and down arrow keys to select the parameter (global volume, mixing level, gamma correction etc.), left and right arrow keys to modify it.

P Pauses the game

Shift+P While paused, advances to next frame

F1 toggle raster effects in some games

F2 Service Mode

F3 Resets the game

F4	Shows the game palette, decoded GFX, and any tilemaps Cursor left/right changes between palette, GFX and tilemaps Cursor up/down cycle through valid palettes Page up/down scroll through the palette and GFX Page up/down, D, G scroll the tilemap display Ctrl & Shift are modifiers to change movement speed. F4 or Esc returns to the emulation. Note: Not all games have decoded graphics and/or tilemaps.
F5	Toggle debugger (only in debug build)
F6	Toggle cheat mode (if started with "-cheat")
F7	Load a 'Save State'. You will be requested to press a key to determine which Save State you wish to load. Note that the 'Save State' feature is for developing drivers and not intended for users. It is incomplete and works only on a number of drivers.
Use	at own risk.
Shift+F7 identify	Create a 'Save State'. Requires an additional keypress to the state.
F8	Decrease frame skip on the fly
F9	Increase frame skip on the fly
F10	Toggles speed throttling
F11	Toggles speed display
Shift+F11	Toggles profiler display (debug builds only)
F12	Saves a screen snapshot. The default target directory is SNAP.
ESC	Exits emulator

MAMEWAH

ctrlr.txt

MAMEWAH v1.5+ Ctrlr System

This is how the ctrlr system works: MAMEWAH will check for the following .ini files in the order below. If a given file does not exist, then MAMEWAH will move onto the next file in the hierarchy.

All files are searched for within '\mame wah\ctrlr\Ctrlr Name\' unless otherwise specified, where 'Ctrlr Name' refers to the name specified in MAMEWAH. Note that 'emuname' should be replaced with the emulator name, as used by MAMEWAH's ini config system. The '?' refers to the game list number, again, as used the \ini config system.

Ctrlr Files:

emuname-?.ini	emulator & game list specific ctrlr file
emuname.ini	emulator specific ctrlr file
ctrlr.ini	general ctrlr

\mamewah\ctrlr\default.ini

MAMEWAH's default ctrlr file

(if none of the above exist, MAMEWAH will exit)

This is a list of key, mouse and joystick input codes which can be used in MAMEWAH's ctrlr ini config system. Note that keycodes are DirectInput standard codes, please consult

http://msdn.microsoft.com/archive/default.asp?url=/archive/en-us/dx81_vb/directx_vb/Input/VB_Ref/Enums/const_dikeyflags.asp for a more thorough explanation (some keycodes are not obvious which keys they are for!). SEE BELOW these codes for further settings...

'Main Keys:

DIK_0	DIK_NUMPAD4
DIK_1	DIK_NUMPAD5
DIK_2	DIK_NUMPAD6
DIK_3	DIK_NUMPAD7
DIK_4	DIK_NUMPAD8
DIK_5	DIK_NUMPAD9
DIK_6	DIK_NUMLOCK
DIK_7	DIK_NUMPADSLASH or DIK_DIVIDE
DIK_8	DIK_NUMPADSTAR or DIK_MULTIPLY
DIK_9	DIK_NUMPADMINUS or DIK_SUBTRACT
DIK_A	DIK_NUMPADPLUS or DIK_ADD
DIK_B	DIK_NUMPADEENTER
DIK_C	DIK_NUMPADPERIOD or DIK_DECIMAL
DIK_D	DIK_NUMPADCOMMA
DIK_E	DIK_NUMPADEQUALS
DIK_F	DIK_ESCAPE
DIK_G	DIK_F1
DIK_H	DIK_F2
DIK_I	DIK_F3
DIK_J	DIK_F4
DIK_K	DIK_F5
DIK_L	DIK_F6
DIK_M	DIK_F7
DIK_N	DIK_F8
DIK_O	DIK_F9
DIK_P	DIK_F10
DIK_Q	DIK_F11
DIK_R	DIK_F12
DIK_S	DIK_F13
DIK_T	DIK_F14
DIK_U	DIK_F15
DIK_V	
DIK_W	
DIK_X	
DIK_Y	
DIK_Z	

DIK_UP or DIK_UPARROW
DIK_DOWN or DIK_DOWNARROW
DIK_LEFT or DIK_LEFTARROW
DIK_RIGHT or DIK_RIGHTARROW

DIK_NUMPAD0
DIK_NUMPAD1
DIK_NUMPAD2
DIK_NUMPAD3

DIK_GRAVE
DIK_TAB
DIK_CAPSLOCK or DIK_CAPITOL
DIK_LSHIFT
DIK_BACKSLASH
DIK_MINUS
DIK_EQUALS
DIK_BACKSPACE or DIK_BACK
DIK_LBRACKET
DIK_RBRACKET
DIK_RETURN
DIK_SEMICOLON
DIK_APOSTROPHE
DIK_COMMA
DIK_PERIOD
DIK_SLASH
DIK_RSHIFT
DIK_LCONTROL
DIK_LWIN
DIK_LALT or DIK_LMENU
DIK_SPACE
DIK_RALT or DIK_RMENU
DIK_RWIN
DIK_APPS
DIK_RCONTROL
DIK_SYSRQ
DIK_SCROLL
DIK_PAUSE
DIK_INSERT
DIK_HOME
DIK_PGUP or DIK_PRIOR
DIK_DELETE
DIK_END
DIK_PGDN or DIK_NEXT

'Brazilian Keyboards:

DIK_ABNT_C1
DIK_ABNT_C2

'British/German Keyboards:

DIK_OEM_102

'Japanese Keyboards:

DIK_AT
DIK_AX
DIK_CIRCUMFLEX

```

DIK_COLON
DIK_CONVERT
DIK_KANA
DIK_KANJI
DIK_NOCONVERT
DIK_STOP
DIK_UNDERLINE
DIK_UNLABLED
DIK_YEN

'Joystick Inputs (?=device no.):
JOY?-LEFT
JOY?-RIGHT
JOY?-UP
JOY?-DOWN
JOY?-IN
JOY?-OUT
JOY?-ROTATELEFT
JOY?-ROTATERIGHT
JOY?-ROTATEUP
JOY?-ROTATEDOWN
JOY?-ROTATEIN
JOY?-ROTATEOUT
JOY?-SLIDER0INC
JOY?-SLIDER0DEC
JOY?-SLIDER1INC
JOY?-SLIDER1DEC
JOY?-POV0-3_LEFT
JOY?-POV0-3_RIGHT
JOY?-POV0-3_UP
JOY?-POV0-3_DOWN
JOY?-BUTTON0-31

```

Other Settings:

Various other settings are available for use using the ctrlr ini files (see the bottom of default.ini). Here is a brief explanation:

keyboard	- enable / disable keyboard
mouse	- enable / disable mouse
joystick	- enable / disable joysticks
digital_scroll_rate	- scroll rate for games / options lists (also minimum scroll rate for simulated analog, see below)
simulated_analog_max_rate	- maximum scroll rate for 'simulated analog' (accelerated) scrolling (games list only)
mouse_x_sensitivity	- mouse x-axis sensitivity
mouse_y_sensitivity	- mouse y-axis sensitivity
(The below are joystick device no. [?] and [axis] specific)	
joy?-axis_deadzone	- % of travel before joystick is engaged
joy?-axis_saturation	- % of travel before maximum joystick value is reached
joy?-axis_sensitivity	- joystick sensitivity
joy?-axis_analog	- specifies whether joystick is analog or digital (analog devices can be treated as digital)

```

'Multimedia/Web Keyboards:
DIK_CALCULATOR
DIK_MAIL
DIK_MEDIASELECT
DIK_MEDIASTOP
DIK_MUTE
DIK_MYCOMPUTER
DIK_NEXTTRACK
DIK_PLAYPAUSE
DIK_POWER
DIK_PREVTRACK
DIK_VOLUMEDOWN
DIK_VOLUMEUP
DIK_WAKE
DIK_WEBBACK
DIK_WEBFAVORITES
DIK_WEBFORWARD
DIK_WEBHOME
DIK_WEBREFRESH
DIK_WEBSEARCH
DIK_WEBSTOP

'Mouse Inputs:
MOUSE_LEFT
MOUSE_RIGHT
MOUSE_UP
MOUSE_DOWN
MOUSE_SCROLLUP
MOUSE_SCROLLDOWN
MOUSE_BUTTON0
MOUSE_BUTTON1
MOUSE_BUTTON2
MOUSE_BUTTON3

```

mamewah.cfg

```

# MAMEWAH 1.62b13+ Standard Control Configuration

# PLEASE USE INPUT CODES FROM \MAMEWAH\DOCS\CTRLR.TXT
# NOTE DIFFERENCES TO MAME's (EX!) CTRLR INI FILES:
# 'NOT' (!) LOGIC NOT SUPPORTED
# 'AND' LOGIC EXPRESSED USING '&' (NOT SPACE)
# 'OR' PIPE '||' CAN BE USED AS WELL AS '|'
# FILES ARE NOT PARSED IN ORDER...YOU SHOULD EXPLICITLY STATE ALL REQUIRED
SETTINGS IN ALL CTRLR INI FILES USED

### INPUT MAPPINGS ###

# main form
UP_1_GAME           "DIK_UP | JOY1_UP | MOUSE_UP"
DOWN_1_GAME          "DIK_DOWN | JOY1_DOWN | MOUSE_DOWN"
UP_1_PAGE            "DIK_R"
DOWN_1_PAGE          "DIK_F"
UP_1_LETTER          "DIK_LEFT | JOY1_LEFT"
DOWN_1_LETTER         "DIK_RIGHT | JOY1_RIGHT"

FIND_GAME            "DIK_TAB"
RANDOM_GAME          "DIK_LSHIFT"
SELECT_GROUP         
NEXT_GROUP            "
PREVIOUS_GROUP        "
SELECT_PLATFORM        "
NEXT_PLATFORM          "
PREVIOUS_PLATFORM      "
SELECT_GAMELIST        "
NEXT_GAMELIST          "
PREVIOUS_GAMELIST      "
ADD_GAME_TO_LIST       "
REMOVE_GAME_FROM_LIST   "
MENU_SHOW              "DIK_2"
MENU_LOCK               "
TOGGLE_MOVIE_PREVIEW_DISPLAY "DIK_M"
TOGGLE_HISTORY_DAT_DISPLAY "DIK_LALT"
LAUNCH_GAME             "DIK_1"
LAUNCH_APP               "
LAUNCH_APP_1            "
LAUNCH_APP_2            "
LAUNCH_APP_3            "
LAUNCH_APP_4            "
LAUNCH_APP_5            "
NEXT_THEME               "
PREVIOUS_THEME             "
EXIT_WITH_CHOICE          "
EXIT_TO_WINDOWS           "DIK_ESCAPE"
EXIT_AND_RUN_APP          "
EXIT_AND_REBOOT            "
EXIT_AND_SHUTDOWN          "

# options form
OP_UP_1_OPTION          "DIK_UP | JOY1_UP | MOUSE_UP"
OP_DOWN_1_OPTION          "DIK_DOWN | JOY1_DOWN | MOUSE_DOWN"
OP_UP_1_OPTION_PAGE        "DIK_LEFT | JOY1_LEFT"
OP_DOWN_1_OPTION_PAGE      "DIK_RIGHT | JOY1_RIGHT"

```

```
OP_MENU_SELECT           "DIK_1"
OP_MENU_BACK             "DIK_2"
OP_MENU_HIDE             

# screensaver form
SS_SHOW_TRACK_NAME       "DIK_2"
SS_FIND_N_SELECT_GAME    "DIK_1"

# any form
VOLUME_DOWN
VOLUME_UP
VOLUME_MUTE
PLAY_OR_PAUSE
PREVIOUS_TRACK
NEXT_TRACK
REWIND_TRACK
FF_TRACK

### SETTINGS ###

# 0 or 1 (off / on)
keyboard 1
mouse 0
joystick 0

# 1+ (milli-seconds)
digital_scroll_rate 125

# 0 = off, 1+ (milli-seconds)
simulated_analog_max_rate 20

# 1-100%
mouse_x_sensitivity 100
mouse_y_sensitivity 100

joy1_x_deadzone 10
joy1_x_saturation 95
joy1_x_sensitivity 100
joy1_x_analog 0

joy1_y_deadzone 10
joy1_y_saturation 95
joy1_y_sensitivity 100
joy1_y_analog 0

joy1_z_deadzone 10
joy1_z_saturation 95
joy1_z_sensitivity 100
joy1_z_analog 0

joy1_rx_deadzone 10
joy1_rx_saturation 95
joy1_rx_sensitivity 100
joy1_rx_analog 0

joy1_ry_deadzone 10
joy1_ry_saturation 95
joy1_ry_sensitivity 100
joy1_ry_analog 0
```

Referencias

6

- <http://www.fdi.ucm.es/> Página web de la facultad de informática.
- <http://www.fdi.ucm.es/migs> Página web oficial del museo de informática García Santesmases.
- <http://www.marcianitos.org> Página web de Marcianitos, gran comunidad española de gente a la que le gustan los videojuegos y las máquinas recreativas, con muchos tutoriales y un foro muy activo.
- <http://www.ultimarc.com> Página web del fabricante del J-Pac y de la ArcadeVGA, tiene documentación interesante.
- <http://www.mame.net>
- <http://www.mamedev.net> Páginas oficiales de MAME.
- <http://mamewah.mameworld.net/> Página oficial del front-end MameWah.
- <http://mala.arcadezentrum.com/index.html> Página oficial del front-end MaLa.
- <http://www.mameworld.net/maws> Página del proyecto “MAWS”, la gran base de datos sobre MAME. Se actualiza en cada versión.
- <http://www.klov.net> “The Killer List of Videogames” otra base de datos online sobre recreativas, incluye juegos aunque no estén en MAME.
- <http://www.tomspeirs.com/gameex/shell.htm> Página donde poder descargar el “Instant Sheller”.
- <http://www.tomspeirs.com/gameex> Otro buen front-end, muy vistoso y con efectos “chulos”, tiene la pega de que necesita una máquina muy potente para poder funcionar correctamente. Altamente **NO** recomendado para el PC usado actualmente.
- http://www.coinop.org/kb_dl.aspx/KB/gametech/jamma%20tech.html

Información de interes sobre el interfaz
JAMMA

- http://www.jammabboards.com/jcenter_jammaFAQ.html Un FAQ para iniciarse en lo que es el JAMMA.
- <http://en.wikipedia.org/wiki/JAMMA>
- <http://www.jamma.or.jp/english/index.htm> Página oficial de la organización JAMMA.

```
joy1_rz_deadzone 10
joy1_rz_saturation 95
joy1_rz_sensitivity 100
joy1_rz_analog 0

joy1_slider0_deadzone 10
joy1_slider0_saturation 95
joy1_slider0_sensitivity 100
joy1_slider0_analog 0

joy1_slider1_deadzone 10
joy1_slider1_saturation 95
joy1_slider1_sensitivity 100
joy1_slider1_analog 0
```