

Manual



**ANTONIO
VILLENA**

Índice

Introducción	1
Agradecimientos.....	1
Modelos y complementos.....	2
MiSTer Mini.....	3
MiSTer Slim.....	4
MiSTer Plus	5
Puertos y conectores.....	6
Decapod	8
Puertos y conectores.....	9
mt32-pi.....	10
Jamma	11
Splitters	11
Configuración Inicial.....	12
Formato de la tarjeta SD	12
Windows.....	13
MacOS	13
Linux	13
Controles.....	14
Teclas especiales y botones	14
Joystick	14
Configuración de joystick	15
Red	16
Bluetooth	17
Wiimote	17
Consola de Linux	18
Scripts	18
Actualizaciones	19
Configuración avanzada	20
Modificar el contenido de una imagen de disco VHD	20
Windows	20
MacOS	20
Linux	20
Crear una imagen VHD.....	21
Windows	21
MacOS	22
Linux.....	23
Otros	25
DB9	25

Acceso remoto	25
Cores	26
Acorn Archimedes	26
Configuración de BIOS	26
Formato de disco duro virtual	26
Teclado	26
Teclas especiales y botones	26
Guía Básica	27
Acorn Atom	28
Formato de disco duro virtual	29
Teclado	30
Teclas especiales y botones	30
Guía básica	30
Acorn Electron	31
Formato de Tarjeta SD	31
Teclado	32
Teclas especiales y botones	32
Guía básica	32
Adventure Vision	34
Alice MC10	35
Altair 8800	36
Amiga	37
Amstrad CPC 6128	38
Formato de Tarjeta SD	38
Teclado	38
Teclas especiales y botones	38
Guía básica	39
Amstrad PCW	40
ao486 (PC 486)	41
Apogee	42
Apple I	43
Formato de Tarjeta SD	43
Teclado	43
Guía básica	44
Apple][.....	46
Formato de Tarjeta SD	46
Windows	46
MacOS y Linux	47
Teclado	47
Teclas especiales y botones	47
Apple Macintosh Plus	48

Aquarius	49
Arcade	50
Arduboy	51
Astrocade	52
Atari 800XL	53
Formato de Tarjeta SD	53
Teclado	53
Teclas especiales y botones	53
Guía básica	54
Atari 2600	55
Formato de Tarjeta SD	55
Teclado	55
Teclas especiales y botones	55
Personalización de los controles	56
Guía básica	56
Atari 5200	57
Atari 7800	58
Atari Lynx	59
Atari ST/STe	60
BBC Micro B,Master	61
Formato de Tarjeta SD	61
Teclado	62
Teclas especiales y botones	62
Guía básica	63
AY-3-8500	65
BK0011M	66
Computers Lynx48,96	67
Formato de Tarjeta SD	67
Teclado	67
Teclas especiales y botones	67
Guía básica	68
Chess	69
CHIP-8	70
Formato de Tarjeta SD	70
Teclado	70
Teclas especiales y botones	70
Guía Básica	71
Colecovision, SG-1000	72
Formato de Tarjeta SD	72
Teclado	72
Teclas especiales y botones	72

Guía básica	73
Color Computer 2, Dragon 32	74
Color Computer 3	75
Commodore 16, Plus/4	76
Formato de Tarjeta SD	77
Windows	77
MacOS y Linux	77
Teclado	78
Teclas especiales y botones	78
Guía básica	79
Commodore 64, Ultimax	80
Formato de Tarjeta SD	81
Windows	81
MacOS y Linux	81
Teclado	82
Teclas especiales y botones	82
Guía básica	83
Commodore PET	84
Formato de Tarjeta SD	84
Teclado	84
Teclas especiales y botones	84
Guía básica	85
Commodore VIC-20	86
Formato de Tarjeta SD	86
Teclado	86
Teclas especiales y botones	86
Guía básica	87
Compukit UK101	88
DEC PDP-1	89
EDSAC	90
Einstein TC01	91
Emerson Arcadia 2001	92
Epoch Galaxy II	93
Flappy Bird	94
Formato de Tarjeta SD	94
Teclado	94
Teclas especiales y botones	94
Fairchild Channel F	95
Game of Life	96
Gameboy, Gameboy Color	97
Gameboy, Gameboy Color 2P	98

Gameboy Advance	99
Gameboy Advance 2P	100
Galaksija	101
Formato de Tarjeta SD	101
Teclado	102
Teclas especiales y botones	102
Guía básica	103
Genesis/Megadrive	104
Intellivision	105
Interact Home Computer	106
Jupiter Ace	107
Formato de Tarjeta SD	107
Teclado	107
Teclas especiales y botones	107
Guía básica	108
Laser 310	109
MegaCD	110
MSX	111
Formato de Tarjeta SD	111
Teclado	112
Teclas especiales y botones	112
Guía básica	113
MSXCTRL	114
Otros	114
MultiComp	115
NEC PC8801	116
NeoGeo	117
NES	118
Formato de Tarjeta SD	118
Teclado	119
Teclas especiales y botones	119
Personalización de los controles	119
Guía básica	120
Odyssey2	121
Formato de Tarjeta SD	121
Teclado	121
Teclas especiales y botones	121
Guía básica	122
Cambio del charset de la ROM de VDC	123
Ondra SPO 186	124
Formato de Tarjeta SD	124

Teclado	124
Teclas especiales y botones	124
Guía básica	124
Orao	126
Oric 1 & Atmos	127
Formato de Tarjeta SD	127
Teclado	128
Teclas especiales y botones	128
Guía básica	129
PMD 85	130
RX-78	131
SAM Coupe	132
Formato de Tarjeta SD	132
Teclado	132
Teclas especiales y botones	132
Guía básica	133
Sega Master System, Game Gear	134
Formato de Tarjeta SD	134
Teclado	134
Teclas especiales y botones	134
Guía básica	135
Sharp MZ Series	136
Sinclair QL	137
Slug Cross	138
SNES	139
Sord M5	140
Specialist/MX	141
SV-328	142
TI-99/4A	143
TomyTronic Scramble	144
TRS-80 Model 1	145
TSConf	146
TurboGrafx 16 / PC Engine	147
VC4000, Interton	148
Vector 06C	149
Vectrex	150
Formato de Tarjeta SD	150
Teclado	150
Teclas especiales y botones	150
Guía básica	151
WonderSwan	152

X68000	153
ZX Spectrum	154
ZX Spectrum Next	155
Formato de Tarjeta microSD.....	156
Teclado.....	156
Teclas especiales y botones	156
Guía básica	157
ZX81.....	159
Formato de Tarjeta microSD.....	159
Teclado.....	160
Teclas especiales y botones	160
Guía básica	161
Referencias	163
Enlaces	163

Introducción

MiSTer es un proyecto abierto que intenta recrear varios ordenadores clásicos, consolas de juegos y máquinas recreativas, utilizando hardware moderno. Esto permite que el software y las imágenes de los juegos se ejecuten como lo harían en el hardware original, utilizando periféricos como ratones, teclados, joysticks y otros mandos de juego.

MiSTer es una evolución del conocido proyecto MiST a una FPGA (Terasic DE10-Nano) más grande y a un procesador ARM más rápido. MiSTer proporciona una salida de vídeo moderna a través de HDMI. El hardware puede ampliarse con varios complementos (como USB, SDRAM, audio, o salida VGA).

El software y sistema operativo de MiSTer puede descargarse libremente y cualquiera puede contribuir a su desarrollo. De hecho, MiSTer depende de las contribuciones de muchos desarrolladores para los distintos sistemas (conocidos como "cores") que replica.

La página oficial del proyecto es https://github.com/MiSTER-devel/Main_MiSTER/wiki.

Agradecimientos

La mayor parte del contenido de este documento se basa en información compartida anteriormente:

- En la [wiki oficial](#)
- En los [foros oficiales](#)
- En el [servidor oficial de Discord](#)

Sin el trabajo previo de todas esas personas (y más), este manual no podría existir.

Modelos y complementos

El componente principal de un sistema MiSTER es una placa DE10-Nano de Terasic, que incluye una FPGA Intel® Cyclone® V SoC, además de un procesador ARM, cabezales de expansión de propósito general de 40 pines y un cabezal Arduino®, memoria DDR3, Ethernet, etc.

Sin embargo, también se pueden añadir distintas placas complementarias, como las siguientes:

- Placa de SDRAM. Algunos cores necesitan esta expansión para funcionar correctamente
- Concentrador USB: proporciona un concentrador OTG, añadiendo puerto USB de sólo alimentación en la parte trasera y 6 puertos USB 2.0 en los otros 3 lados.
- Placa de E/S digital: incorpora varios botones, LEDs de estado, ranura de microSD secundaria, puerto de audio digital óptico con TOSLINK, puerto E/S de usuario (para la comunicación directa en serie con varios periféricos y adaptadores como los adaptadores SNAC y MT32-Pi) y un sistema de refrigeración con ventilador.
- Placa de E/S analógica: con varios botones, LEDs de estado, interruptor de encendido, jack de barril con el que se puede alimentar el DE10-Nano, ranura de microSD secundaria, puerto VGA para salida de vídeo analógico, puerto de audio analógico de 3,5 mm que también es un puerto de audio digital óptico Mini-TOSLINK, puerto de entrada de audio de 3,5 mm para ADC-in, puerto E/S de usuario (para la comunicación directa en serie con varios periféricos y adaptadores como los adaptadores SNAC y MT32-Pi) y un sistema de refrigeración con ventilador.
- Placa MT32-Pi Lite MiSTER: esta placa incorpora un emulador del módulo de sintetizador Roland MT-32, y que permite escuchar audio Midi actualizado en ciertos cores y juegos que lo soporten.
- Tarjeta RTC: añade un reloj en tiempo real, que utilizan algunos cores de ordenador.

A continuación, se describen varias unidades que se venden ya ensambladas incluyendo varias de estas placas complementarias, y otros periféricos externos también disponibles.

MiSTER Mini

Unidad con placas de expansión alternativas a las oficiales. La placa IO incluye un DAC RGB888 R-2R con salida VGA, salida de audio jack (analógica y óptica), entrada de audio jack (para carga de cintas), joystick DB9, leds y pulsadores.



Características:

- Placa Terasic DE10-Nano (salida HDMI)
- DAC de audio 112dB SNR (PCM5122)
- DAC de vídeo RGB888 video DAC
- 128MB de SDRAM
- Placa IO (para salida VGA y Scart)
- Disipador de 6mm y ventilador de 40x40mm
- Hub USB con 7 puertos (uno interno)
- Puerto microSD secundario
- Puerto de joystick DB9, protegido para SNAC
- Salida de audio óptica SPDIF a través de conector de 3.5mm

MiSTer Slim

Incluye placas de expansión alternativas a las oficiales alternativas a las oficiales. Compatible con los módulos SDRAM horizontales y verticales oficiales. La placa IO incluye un DAC R-2R con salida VGA, Scart o YPbPr, salida de audio jack (analógica y óptica), entrada de audio jack (para carga de cintas), ranura secundaria para microSD, joystick DB9, leds y pulsadores.

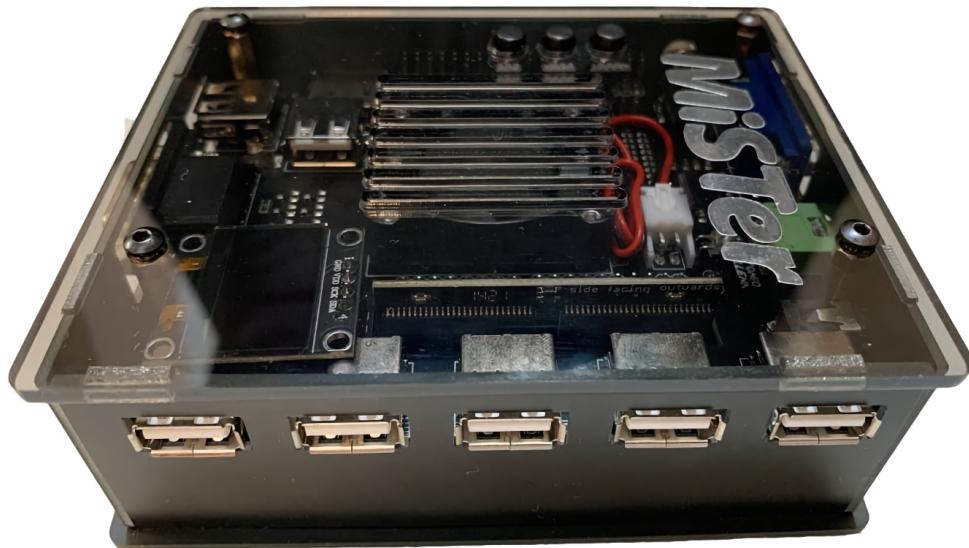


Características principales:

- Placa Terasic DE10-Nano (salida HDMI)
- 128MB de SDRAM
- Placa IO (para salida VGA y Scart)
- Disipador de 6mm y ventilador de 40x40mm
- Hub USB con 7 puertos
- Extensor microSD
- RTC
- Altavoz interno
- Puerto de joystick DB9, protegido para SNAC
- Salida de audio óptica SPDIF a través de conector de 3.5mm

MiSTER Plus

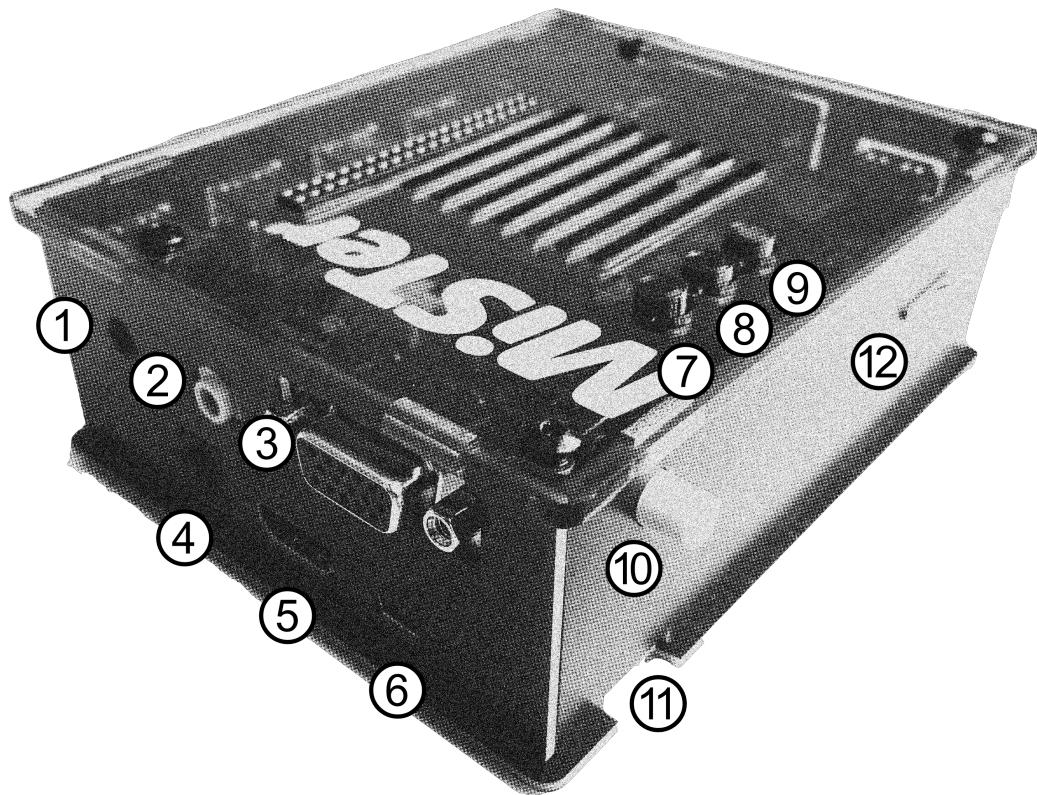
Unidad completamente que incluye varias placas de expansión alternativas a las oficiales. La placa IO incluye un DAC RGB888 R-2R con salida VGA, salida de audio jack (análogica y óptica), RTC, pantalla OLED, entrada de audio jack (para carga de cintas), joystick DB9, leds y pulsadores.



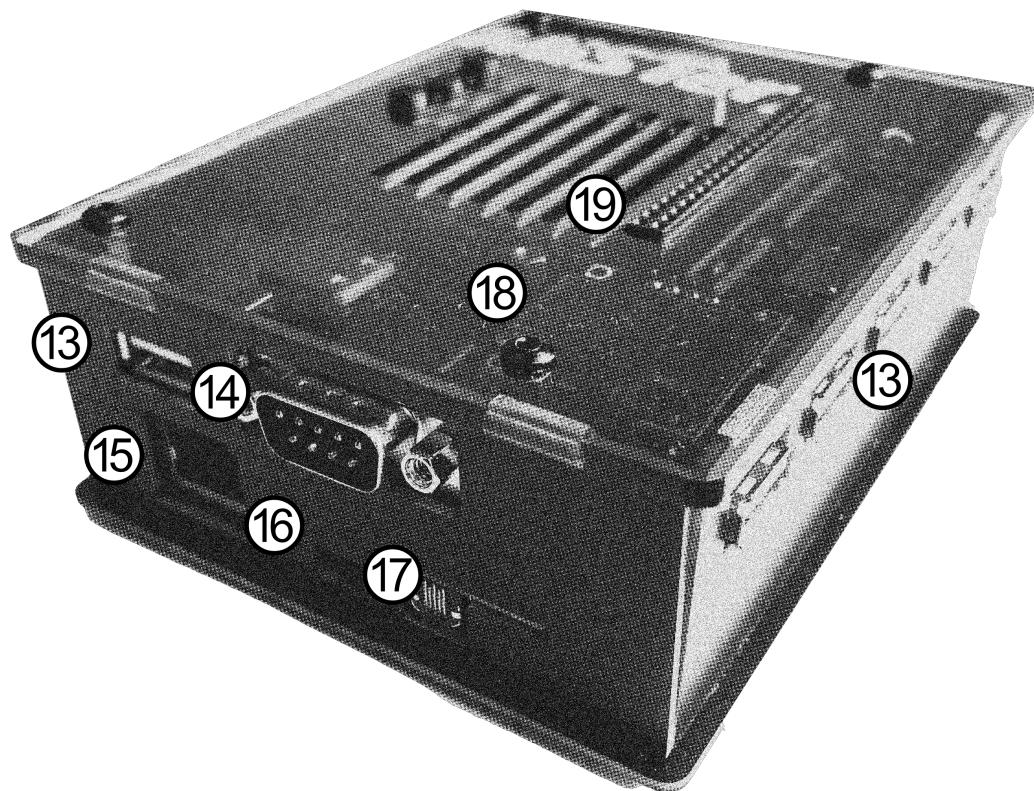
Características:

- Placa Terasic DE10-Nano (salida HDMI)
- DAC de audio 112dB SNR (PCM5122)
- DAC de vídeo RGB888 video DAC
- 128MB de SDRAM
- Placa IO (para salida VGA y Scart)
- Disipador de 6mm y ventilador de 40x40mm
- Hub USB con 7 puertos (uno interno)
- Opcionalmente, Wifi y BlueTooth
- Puerto microSD secundario
- RTC
- Pantalla OLED para mostrar logos de los cores
- Puerto de joystick DB9, protegido para SNAC
- Salida de audio óptica SPDIF a través de conector de 3.5mm

Puertos y conectores



1	Entrada de sonido
2	Salida de sonido
3	Salida de vídeo VGA
4	Alimentación
5	HDMI
6	Puerto USB (USB-Blaster II)
7	Botón de reset
8	Botón de usuario
9	Botón de OSD
10	Botón de encendido
11	microSD principal
12	microSD secundaria



13	Puertos USB
14	Puerto DB9 (Usuario/Joystick/SNAC)
15	Puerto Ethernet
16	Puerto USB (UART)
17	Puerto USB OTG (utilizado por el hub USB)
18	Pantalla OLED
19	Puerto de expansión

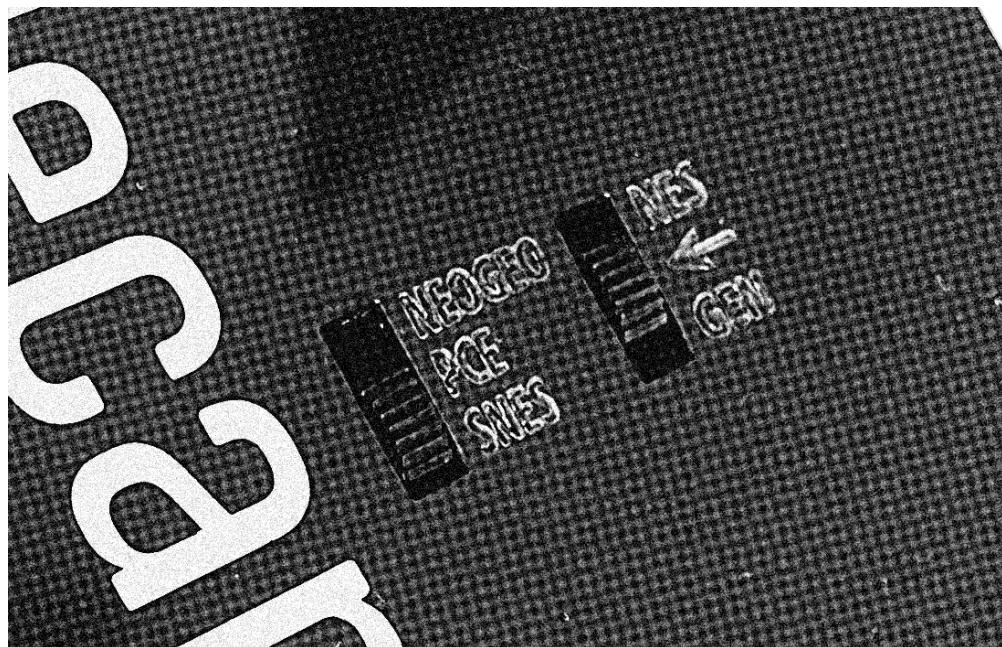
Decapod

El adaptador decapod, basado, en parte, en los [adaptadores retro DeamonBite de MickGyver](#), permite conectar mandos de SNES, NES, Mega Drive/Genesis (3/6 botones), Master System, Atari, NEO GEO y PC Engine/TurboGrafX a USB (a MiSTER o a un ordenador), y/o SNAC vía un conector DB9. Además, la versión con DB9 puede utilizar una placa [mt32-pi](#).

Se pueden conectar todos los puertos de controlador a la vez, pero sólo los 2 puertos seleccionados por los interruptores superiores funcionarán cada vez.



Nunca se ha de conectar DB9 y USB al mismo tiempo



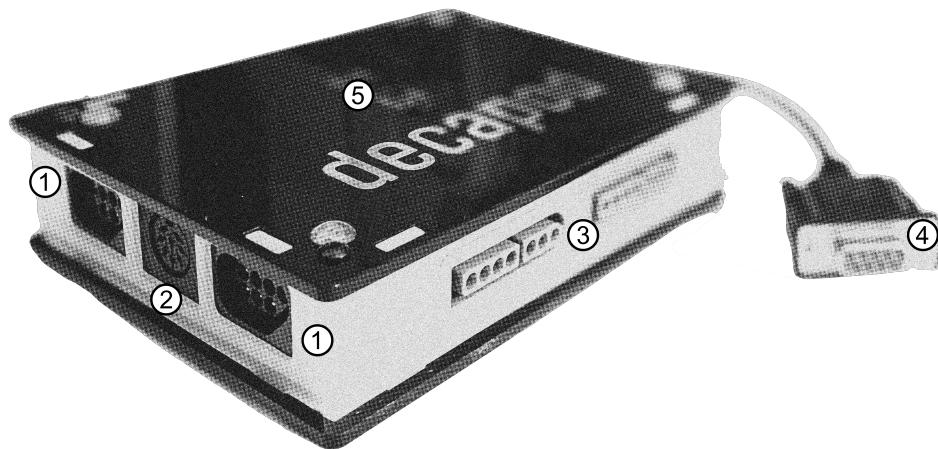
Estas son las combinaciones necesarias, usando los interruptores superiores, para elegir los puertos a utilizar

Puerto	Izquierdo	Derecho
NES		Arriba (NES)
Genesis		Abajo (GEN)
Neo Geo	Arriba (NEOGEO)	Centro(←)
PC Engine o TurboGrafX	Centro (PCE)	Centro (←)
SNES	Abajo (SNES)	Centro(←)

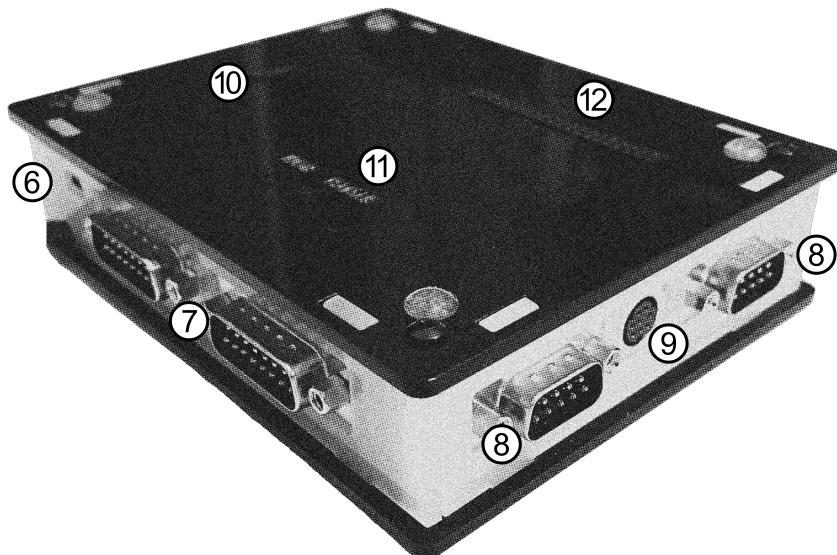


Para activar la funcionalidad mt32-pi en la parte inferior, se ha de seleccionar SNES en los interruptores, con ningún controlador SNES conectado y con una placa mt32-pi en los conectores de la parte inferior. Conectar entonces el Decapod a través del puerto DB9 a la MiSTER, y ejecutar un core compatible con [mt32-pi](#).

Puertos y conectores



1	Puertos de NES
2	Puerto DIN-8 de TurboGrafx-16
3	Puertos de SNS
4	Conector DB9 para MiSTER*
5	Botones de selección de puertos



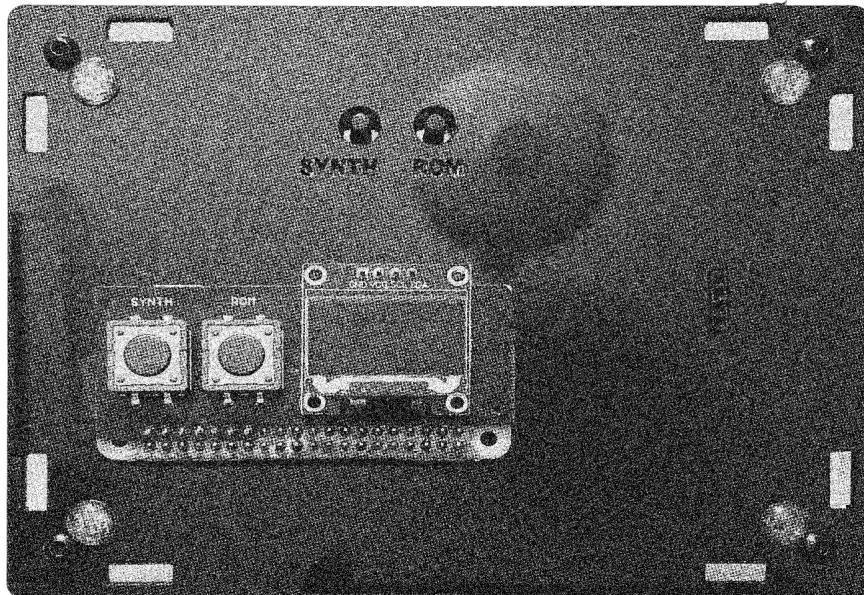
6	Puerto USB para PC o MiSTER
7	Puertos de Neo Geo
8	Puertos DB9 de Atari/Mega Drive/Genesis
9	Puerto Mini-Omer 8 de PC Engine
10	Conector para LCD (mt32)*
11	Botones para mt32*
12	Conector para Raspberry Pi (mt32)*

mt32-pi

mt32-pi es un sintetizador MIDI (baremetal) para la Raspberry Pi, que emula el módulo MT-32 de Roland. Conectado a la MiSTer, ofrece sonido, por ejemplo, para el <#_ao486,core AO486>.



La forma de conectar la Raspberry Pi Zero con mt32-pi es de forma que los pines estén en el lado más cercano a nosotros y mirando hacia abajo, orientando también el Decapod para que el lado con la etiqueta del decapod esté en la parte inferior, y los puertos SNES estén en el lado más cercano. En el lado del Decapod que está orientado hacia arriba, conectar los pines de la cabecera de la Raspberry Pi en el puerto de la parte inferior izquierda.



Una vez que se ha conectado la Raspberry Pi con mt32-pi, se puede utilizar el Decapod como una mt32-pi siguiendo estos pasos:

1. Cable USB desconectado.
2. Desconectar todo lo que esté conectado a los puertos de SNES.
3. Configurar los interruptores del Decapod para que el modo SNES esté activado.
4. Conectar el cable DB9 al puerto DB9 de la tarjeta I/O de la MiSTer.
5. Iniciar un core (por ej. [AO486](#)) con soporte para mt32-pi, y comenzar a utilizarlo



El core [AO486](#) requiere que la opción de E/S del USER esté configurada como MIDI.

Jamma

Este adaptador permite meter la MiSTER dentro de una caja de máquina arcade y conectar los mandos correspondientes. Es compatible con MiSTER, PC y πCRT. Sólo hay que conectar el VGA y un cable a un puerto USB. En caso de PC (o MiSTER oficial) es necesario conectar también un cable jack para el audio y un adaptador para utilizar un cable VGA.

Basado en una placa Atmega32u4, permite 2 controles con una sola conexión USB con baja latencia (0,74ms). El firmware se puede actualizar desde [este repositorio](#).

Además, dispone de un amplificador estéreo con control de volumen para alimentar directamente los altavoces de la caja. La versión con caja tiene un arnés de CPS2. La versión sin caja sólo tiene la huella, no el conector.

Splitters

Adaptadores varios para la versión de MiSTER con puerto DB9. Permiten conectar dos controladores nativos para el jugador 1 y el jugador 2, para distintos mandos clásicos (SNES, NES, Genesis/Megadrive, NeoGeo o TurboGrafx/PCE).

Configuración Inicial

Para poder poner en marcha una MiSTer hace falta, al menos, lo siguiente:

- Un cargador USB, una TV u otro dispositivo que ofrezca alimentación USB. Para las versiones con varios puertos USB donde se conecten varios dispositivos, se recomienda, como mínimo una fuente de 4A.
- Un cable y un monitor o TV con HDMI (o VGA para los modelos de MiSTer que lo soporten)
- Un modelo de MiSTer con varios puertos USB, o un adaptador USB OTG
- Un teclado USB
- Una tarjeta microSD con el formato adecuado

Para poder aprovechar todo su potencial, es útil tener también (según el core a usar):

- Conexión de red ethernet (o un adaptador USB WiFi compatible)
- Mandos de juegos (USB, o clásicos con un adaptador adecuado, como splitters en el puerto DB9, un decapod, etc.)
- Un ratón USB

Formato de la tarjeta SD

MiSTer utiliza una versión de Linux optimizada para que tener un arranque muy rápido (menos de 5 segundos), y que se ha de instalar la tarjeta microSD principal (la incorporada en la placa principal Terasic DE10-Nano).

Si no se dispone ya de una tarjeta preparada, el método más sencillo de prepararla es utilizar [la imagen de Mr Fusion](#).

Mr Fusion se basa en una pequeña imagen compacta que se aplica sobre la tarjeta microSD. Cuando se arranque esta tarjeta la placa DE10-nano, se expandirá automáticamente a la máxima capacidad y se instalará una configuración básica de MiSTer.

Después de esto, usando los diferentes "scripts" incorporados, se puede configurar el acceso de red y ejecutar el script MiSTer Updater dejar la instalación actualizada a la versión más reciente de software.



Este proceso borra el contenido que hubiera antes en la tarjeta SD, y no se puede deshacer.

El primer paso consiste en descargar la imagen más reciente de Mr Fusion desde el [el repositorio oficial de GitHub](#).

Windows

Volcar la imagen en la SD, por ejemplo, usando [HDD Raw Copy Tool](#) o [balena Etcher](#).

MacOS

Se puede utilizar una herramienta como [balena Etcher](#) o la línea de comandos.

En el caso de querer usar la línea de comandos, lo primero es determinar el disco a formatear:

```
diskutil list
```

En este ejemplo sería el disco 6:

```
(...)
/dev/disk6 (external, physical):
 #:          TYPE NAME      SIZE    IDENTIFIER
 0:  FDisk_partition_scheme        *15.9 GB   disk6
 1:            DOS_FAT_32 UNKNOWN     15.9 GB   disk6s1
```

Pasos a seguir:

1. Expulsar el disco:

```
diskutil unmountDisk /dev/disk6
```

2. Volcar la imagen en la SD, usando **dd**:

```
sudo dd if=mr-fusion.img of=/dev/rdisk6 bs=8m
```

Linux

Volcar la imagen en la SD, usando **dd** (por ejemplo, si la tarjeta SD es el disco **sdc**):

```
sudo dd if=mr-fusion.img of=/dev/sdc
```

Controles

Teclas especiales y botones

- **F12**: Mostrar u ocultar el menú OSD de MiSTer
- **Alt+F12**: Mostrar u ocultar el menú OSD del core actual
- Flechas de dirección: Navegar por los menús
- **F1**: Cambiar la imagen de fondo del menú principal
- **F9**: Shell de Linux de MiSTer (**F12** para volver al menú principal)
- **F11**: Menú de enlace **bluetooth** (sólo con el menú de MiSTer activo y 000si hay conectado un dispositivo bluetooth compatible)
- **Mayús.Izda+Ctrl Izda+Alt Izda+Alt Dcha**: Reiniciar la MiSTer
- **Win+Impr.Pant**: Guardar una captura de pantalla en

Joystick

En general, antes de utilizar un mando con un core, tiene que configurarse desde el menú principal de MiSTer.

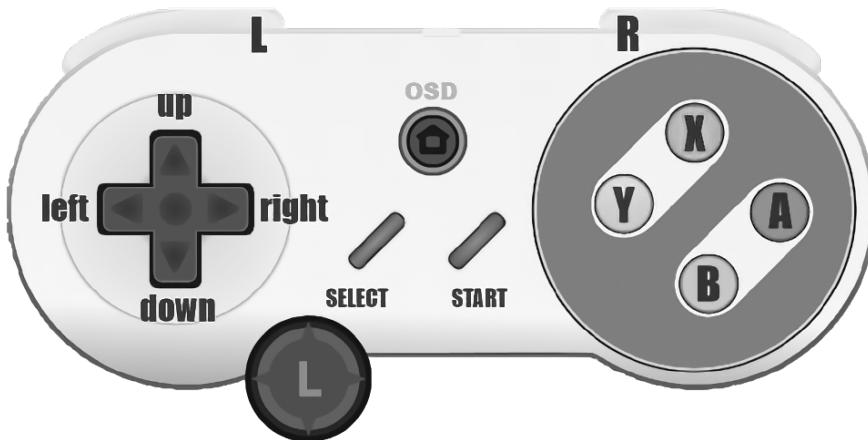
El sistema de configuración (mapeo de botones), se basa en tres pasos:

1. Asignar un mando físico a un "mando de MiSTer" virtual (+ extras)
2. El mando virtual se asigna automáticamente a los distintos cores
3. Opcinalmente, se puede modificar este mapeo de forma específica para cada núclo (y para cada mando), desde el menú OSD de cada core

La comunidad recopila algunos mapeos ya hechos, disponibles en [el repositorio oficial](#). Los ficheros descargados, se deben dejar en el directorio **Inputs** de la microSD (que se corresponde con **/media/fat/Inputs/** en la consola de Linux).

Configuración de joystick

La configuración básica del mando virtual de MiSTer se corresponde con el siguiente esquema:



Para crear una asignación (mapeo de botones), se han de seguir los siguientes pasos:

1. Con un teclado conectado, pulse **F12** para mostrar el menú y elija la opción "Define joystick buttons"
2. El sistema solicitará asignar varios botones al mando:
 - Comprobar el D-Pad y los mandos analógicos (si los hay)
 - Cuatro botones principales y dos traseros
 - Inicio (Start) y Seleccionar (Select)
 - Una pulsacion de un botón asociado al menú, o una combinación de dos (para usar en vez de **F12** en el teclado)
 - Otros botones extra



Se puede omitir una asignación pulsando la tecla **Espacio** del teclado

Existe información más detallada (en inglés) en [la wiki oficial del proyecto](#).

Red

La versión de Linux que utiliza MiSTer, está preparada para utilizar automáticamente la conexión ethernet con DHCP.

Para utilizar una dirección de red estática, se puede editar el archivo `/etc/dhcpcd.conf`, añadiendo datos similares a los siguientes al final del mismo:

```
interface eth0
static ip_address=<dirección ip>/<máscara de red> (ej. 192.168.0.15/24)
static routers=<dirección ip de la puerta de enlace> (ej. 192.168.0.1)
static domain_name_servers=<direcciones ip de servidores DNS, separadas por espacios>
(ej. 192.168.0.1 1.0.0.1)
```

También es posible utilizar algunos adaptadores WiFi. Para ello, se debe editar el fichero `/media/fat/linux/wpa_supplicant.conf`, y **guardar una copia como** `/media/fat/linux/wpa_supplicant.conf (sin el delante)`, modificando la parte que pone `put_your_SSID_here`, poniendo en su lugar el nombre de la red donde conectar, y modificando también `put_your_password_here`, sustituyéndolo por la contraseña de la red.

Después de esto, al reiniciar la MiSTer, se debería activar la conexión de red.



Es posible, en el caso de que no estén incluidos en la distribución de Linux de MiSTer, compilar controladores propios, siguiendo las instrucciones disponibles en [este enlace](#).

Bluetooth

Es posible enlazar dispositivos bluetooth, para ello se ha de conectar un único adaptador bluetooth (por ejemplo, en un puerto USB). No está soportado tener varios adaptadores bluetooth a la vez.

Sólo se permite la conexión de dispositivos bluetooth de entrada (teclados, mandos de juegos, etc.). No hay soporte, por ejemplo, para dispositivos bluetooth de audio. No se garantiza compatibilidad con trackpad o ratones.

Para emparejar un dispositivo bluetooth, desde el menú de MiSTER, pulsar la tecla **F11** de un teclado conectado, o bien el botón **OSD** de una placa de expansión durante varios segundos. Esto hará que se muestre la ventana de enparejamiento de bluetooth.

MiSTER intentará encontrar y emparejar los dispositivos que encuentre en modo de enparejamiento (los teclados requieren que se permita introducir el pin **0000** y pulsar **Enter**).



Después de reiniciar la MiSTER, muchos dispositivos no se reconectan automáticamente.



Para depurar el funcionamiento de bluetooth, se puede usar el comando **hcitool dev** para verificar si se ha detectado el adaptador bluetooth, y el comando **hcitool scan** sirve para hacer una búsqueda y mostrar todos los dispositivos bluetooth encontrados

Wiimote

Los mandos Wiimote funcionan de forma nativa desde la versión 20190510 de Linux. El método de enparejamiento es como el de cualquier otro dispositivo bluetooth, usando el botón rojo de sincronización en la parte trasera del mando (el enparejamiento mediante los botones 1+2 **no funciona**).



Se recomienda usar la primera versión del Wiimote (la que no tiene integrado WiiMotion Plus). La segunda versión del Wiimote (con WiiMotion Plus integrado) también es compatible, pero se conecta automáticamente y hay que emparejarlo cada vez que se vaya a utilizar. Los Wiimotes de terceros pueden o no funcionar, o funcionar con problemas. Así que se aconseja usar el Wiimote oficial.

El Nunchuck y el Classic Controller conectados al Wiimote son compatibles. El adaptador Mayflash Gamecube para Wiimote también es compatible.

Recordar que, para utilizar completamente el Wiimote hace falta una barra de infrarrojos.

Consola de Linux

La consola o shell (entorno de comandos) de Linux de MiSTer (a la que se puede acceder, por ejemplo, vía red -SSH-, o pulsando **F9** en el menú de MiSTer) utiliza bash. Bash es un intérprete de línea de comandos interactivo o shell. Algunos comandos útiles son

- **ls**: Muestra el contenido de un directorio
- **cd**: Cambia la ubicación actual a un nuevo directorio
- **mkdir**: Crea un directorio nuevo
- **rmdir**: Borra un directorio
- **mv**: Mueve o renombra ficheros
- **cp**: Copia ficheros
- **rm**: Borra ficheros
- **find**: Encuentra ficheros
- **less**: Muestra el contenido de un fichero de texto, sin modificarlo
- **nano**: Permite cambiar el contenido de un fichero de texto

Existen multitud de manuales, guías, cursos, etc. para aprender el uso de Linux, como por ejemplo, [los tutoriales](#) en la web de los [servicios editoriales para la documentación libre en español](#).

Scripts

En vez de tener que utilizar la consola, es posible, desde el menú de MiSTer, acceder a la opción "Scripts". Normalmente aparecerá una advertencia avisando del uso de scripts. Tras aceptarla, se podrán ver todos los ficheros de script que se encuentran en el directorio **Scripts** de la tarjeta microSD (que se corresponde con **/media/fat/Scripts** en la consola), y se puede seleccionar el que se desee para ejecutarlo.

Actualizaciones

La instalación básica estándar de MiSTer provee de un [script](#) llamado [update.sh](#) que permite obtener la última versión de cores, scripts, y otros ficheros útiles para el funcionamiento del sistema.

Una vez realizada la primera actualización, se añadirá otro [script](#) llamado [update_all.sh](#) que provee de funciones más avanzadas, permitiendo elegir distintos tipos de actualizaciones:

- Main Distribution: Distribución principal. Descarga la mayoría de los archivos esenciales, incluyendo cores y firmware. Se puede elegir entre dos versiones:
 - La distribución oficial de MiSTer en la organización MiSTer-devel. [Opción por defecto]
 - La Distribución MiSTer DB9 con compatibilidad ampliada de controladores nativos para los controladores Genesis y NeoGeo/Supergun. Requiere un adaptador compatible con SNAC8.
- JTCORES for MiSTer: JTCORES para MiSTer. Descarga muchos cores realizados por Jotego. Se puede seleccionar entre dos versiones:
 - JTSTABLE. Sólo cores públicos. [Opción por defecto]
 - jtbin. Cores públicos y Cores Beta en sus últimas versiones.
- thepsilon Unofficial Distribution: Distribución no oficial. Descarga algunos cores que aún no han sido liberados pero que son bastante estables. [Desactivado por defecto]
- LLAPI Folder: Descarga los cores LLAPI compatibles con BlisSTer y LLAMA. [Desactivado por defecto]
- Arcade Offset. Descarga arcade parcheados mantenidos por atrac17. [Desactivado por defecto]
- Names TXT: Descarga un archivo names.txt completo, mantenido por Threepwood, que permite mejorar los nombres de los cores en los menús. [Desactivado por defecto] Adicionalmente, este script también encadena los siguientes scripts:
 - BIOS Getter: descarga ficheros BIOS para los sistemas instalados desde <https://archive.org/>
 - MAME y HBMAME Getter: descarga ROMs de <https://archive.org/>
 - Arcade Organizer: Crea una estructura de directorios bajo _Arcade/_Organized para facilitar la navegación y el acceso a los archivos MRA.

[update_all.sh](#) puede tardar más de media hora la primera vez que se ejecuta, pero las ejecuciones posteriores suelen tardar mucho menos.

Para más información, ajustes, extra, etc. se puede consultar [el repositorio oficial](#).

Configuración avanzada

Modificar el contenido de una imagen de disco VHD

Windows

En Windows, existen múltiples utilidades como el propio sistema operativo (Windows 10 o posterior), [OSFMount](#) o [ImDisk](#) que permiten montar los archivos de imagen como si fuesen discos físicos.

También es posible hacerlo utilizando la utilidad de comandos del sistema [diskpart](#):

```
select vdisk file="ruta_al_fichero_vhd"
attach vdisk
```

MacOS

Montar la imagen desde Terminal con

```
hdiutil attach -imagekey diskimage-class=CRawDiskImage /ruta_al_fichero_vhd
```

Linux

El primer paso para poder acceder a una imagen de disco, es analizar el contenido, ya que, a veces, se trata de imágenes completas de disco, incluyendo particiones. Con el comando [fdisk](#) se puede averiguar la geometría de la imagen.

```
fdisk -lu /ruta_al_fichero_vhd

You must set cylinders.
You can do this from the extra functions menu.

Disk ...fichero_vhd: 0 MB, 0 bytes
nnn heads, mm sectors/track, 0 cylinders, total 0 sectors
Units = sectors of 1 * bbb = bbb bytes
Sector size (logical/physical): bbb bytes / bbb bytes
I/O size (minimum/optimal): bbb bytes / bbb bytes
Disk identifier: 0xdeffffff

      Device Boot      Start        End      Blocks   Id  System
fichero_vhd          offset_start    ....
```

Según esta información, la imagen tiene sectores de `bbb` bytes de tamaño y comienza en `offset_start`. Con esta información, se puede usar el comando `mount` para montar, por ejemplo, la imagen en la ruta `/mnt/vhd_disk`.

```
mount -o loop,offset=$((offset_start*bbb)) /ruta_al_fichero_vhd /mnt/vhd_disk/
```

Por ejemplo, para una imagen con sectores de `512` bytes de tamaño y comienzo en `63`:

```
mount -o loop,offset=$((63*512)) /ruta_al_fichero_vhd /mnt/vhd_disk/
```

Crear una imagen VHD

Windows

Se puede crear una imagen de disco RAW para que utilice un core. Por ejemplo, siguiendo estos pasos, se puede tener una imagen de 2GB FAT16

1. Crear archivo vacío (2G)

```
diskpart
create vdisk file="spectrum.vhd" type=fixed maximum=2000
select vdisk file="spectrum.vhd"
attach vdisk
convert mbr
create partition primary
```

2. Identificar la partición y formatearla (en este ejemplo, sería la número 8)

```
list partition
select partition 8
format /FS:FAT /Q
exit
```

3. Modificar el contenido de la imagen, y finalmente expulsar el disco al finalizar

MacOS

Se puede crear una imagen de disco RAW para que utilice un core. Por ejemplo, siguiendo estos pasos, se puede tener una imagen de 2GB FAT16

1. Crear archivo vacío (2G)

```
dd if=/dev/zero of=spectrum.vhd bs=8m count=256
```

2. Crear particiones en el archivo

```
fdisk -e spectrum.vhd
fdisk: could not open MBR file /usr/standalone/i386/boot0: No such file or directory
The signature for this MBR is invalid.
Would you like to initialize the partition table? [y] y
Enter 'help' for information
fdisk:*1> erase
fdisk:*1> edit 1
      Starting      Ending
 #: id cyl hd sec - cyl hd sec [     start -       size]
-----
 1: 00   0   0   0 -   0   0   0 [           0 -             0] unused
Partition id ('0' to disable) [0 - FF]: [0] (? for help) 6
Do you wish to edit in CHS mode? [n]
Partition offset [0 - 4194304]: [63] 128
Partition size [1 - 4194176]: [4194176]
fdisk:*1> flag 1
Partition 1 marked active.
fdisk:*1> w
Writing MBR at offset 0.
fdisk: 1> exit
```

3. Preparar el nuevo disco:

```
hdiutil attach -imagekey diskimage-class=CRawDiskImage -nomount spectrum.vhd
```

4. Tomar nota de cuál es el nuevo dispositivo (en este ejemplo `/dev/disk7s1`) y formatear en FAT16:

```
newfs_msdos -F 16 -v SPECTRUM -c 128 /dev/disk7s1
hdiutil detach /dev/disk7
```

5. Montar imagen para poder copiar los ficheros que se quiera:

```
hdiutil attach -imagekey diskimage-class=CRawDiskImage spectrum.vhd
```

Linux

Se puede crear una imagen de disco RAW para que utilice un core. Por ejemplo, siguiendo estos pasos, se puede tener una imagen de 2GB FAT16

- Crear archivo vacío (2G)

```
dd if=/dev/zero of=spectrum.vhd bs=8m count=256
```

- Crear particiones en el archivo

```
fdisk --compatibility=dos spectrum.vhd
(...)
Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (62-31116288, default 62):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (128-31116288, default 31116288):
Created a new partition 1 of type 'Linux'

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 6
Changed type of partition 'Linux' to 'FAT16'.

Command (m for help): a
Partition number (1, default 1): 1
The bootable flag on partition 1 is enabled now.

Command (m for help): w
```

- Montar el fichero de imagen

```
losetup /dev/loop0 spectrum.vhd
```

- Formatear la partición FAT

```
mkfs.vfat /dev/loop0p1
losetup -d /dev/loop0
```

5. Montar imagen para poder copiar los ficheros que se quiera:

```
mount -o loop,offset=$((62*512)) spectrum.vhd /mnt/vhd_disk/
```

Otros

DB9

https://github.com/antoniovillena/MiSTER_DB9/blob/master/Docs/db9-user-io-and-snac8.md

Acceso remoto

https://github.com/MiSTER-devel/Main_MiSTER/wiki/Samba

Montar CIFS

<https://misterfpga.org/viewtopic.php?t=3246>

Cores

Acorn Archimedes

El [Acorn Archimedes](#) fue el primer ordenador personal de propósito general de Acorn Computers basado en su propia CPU RISC ARM de 32 bits. La [versión para MiSTER](#) es una adaptación del [core Archimedes](#) de MiST, de Stephen Leary.

Características principales:

- RTC: Obtiene la configuración del reloj actual del subsistema Linux de MiSTER
- Soporte para dos unidades de disquete (sólo en formato ADF, y de exactamente 819200 bytes)
- Soporte para hasta dos unidades de disco duro con extensión HDF

Configuración de BIOS

Se necesita una versión de una [ROM de RISC OS](#) en la ruta `.../games/ARCHIE/riscos.rom`.

Formato de disco duro virtual

Se pueden montar hasta 2 imágenes de disco duro con extensión HDF.

Teclado

Teclas especiales y botones

- **Win+F12**: Mostrar u ocultar el menú OSD del core
- **Ctrl+Alt+Altg Gr** o el botón **User**: Reinicio en caliente (warm reset)
- Mantener pulsado **Ctrl+Alt+Altg Gr** o el botón **User**: Reinicio en frío (cold reset). Vuelve a cargar la ROM de RISC OS y se borra toda la memoria.

Guía Básica

[archie]

Si se reconoce la ROM, el core debería arrancar con RISC OS. Pulsar **Win+F12** para mostrar el menú OSD. El acceso otras pantallas de configuración es pulsando la tecla de flecha derecha.

Para configurar discos duros virtuales se han de seguir los siguientes pasos:

1. Borrar cualquier archivo **cmos.dat** existente de la carpeta **Archie**.
2. Volver a cargar el core
3. Abrir la línea de comandos de MiSTER (**F12** o **Ctrl+F12**) y escribir: **configure iedefsdiscs 1** (o **2** si utiliza dos discos duros)
4. Montar la(s) imagen(es) requerida(s)
5. Hacer un reinicio en frío.

Para guardar las imágenes actualmente seleccionadas, usar la opción el menú OSD del core "Save configuration"

Acorn Atom

El [Acorn Atom](#) era un computador casero hecho por Acorn Computers. El [core de MiSTer](#) es una adaptación del proyecto [AtomFPGA](#).

Características principales:

- 32KB de ram
- ROMs de sistema con mmc (para acceder a la tarjeta SD)
- Siete Roms adicionales y 1 ranura de ram para descargar Roms adicionales.
- Dos juegos de caracteres.
- Entrada/salida de cinta
- Sonido seleccionable, Atom,SID,Tape,off
- Fondo negro u oscuro
- Modo Atom o BBC Basic
- Paleta de colores
- Sonido SID
- Modos Turbo
- Cuatro modos de teclado seleccionables: UK(por defecto),USA,original,juego

Formato de disco duro virtual

Se necesita un archivo **boot.vhd** formateado como FAT y de alrededor de 100MB de tamaño (se puede obtener uno vacío en [el repositorio del proyecto](#)). Se puede [editar el contenido de la imagen de disco](#).

La mejor fuente es Atom Software Archive [en GitHub](#).

Se puede organizar la información en el disco duro virtual de dos maneras distintas:

1. Descomprimir todo el contenido del archivo en la raíz del disco duro virtual.
2. Organizar la información de una manera más reducida en la raíz, utilizando sólo dos directorios. Crear un directorio **ATOM** en la raíz de la tarjeta, y copiar en su interior todo el contenido del archivo, excepto el directorio **MANPAGES** que se tendrá que poner también en la raíz de la SD. Luego, copiar los ficheros del archivo **trick_ATOM_folder** (disponible [en el foro de ZX-Uno](#)), reemplazando todos los que se encuentren con el mismo nombre. Así, quedará una estructura como la siguiente:

```

/
+-ATOM/
|   +-AA/
|   | (... )
|   +-AGD/
|   |   +-SHOW2
|   |   +-SHOW3
|   | (... )
|   +-MENU
|   | (... )
|   +-TUBE/
|   |   +-BOOT6502
|   | (... )
|
+-MANPAGES/
|   +-CPM.MAN
|   +-FLEX.MAN
|   | (... )
|
+-MENU

```

Una vez preparado, copiar el fichero en la ruta […/games/AcornAtom/boot.vhd](#).

Teclado

Teclas especiales y botones

- **F12:** Mostrar u ocultar el menú OSD del core
- **Mayús+F10:** Muestra el menú de Atom Software Archive
- **F1:** Modo turbo 1Mhz
- **F2:** Modo turbo 2Mhz
- **F3:** Modo turbo 4Mhz
- **F4:** Modo turbo 8Mhz
- **F10: Shift+Break**
- **Mayús.: Shift Lock**
- **Alt Derecha: Repeat**
- **Tab: Copy**
- **Alt Izquierda: Shift**
- **Flecha Arriba: Shift+6**
- **Shift+8: (**
- **Shift+9:)**
- **Ctrl+F10: (Ctrl-Break)** desactiva la ROM de MMC

Guía básica

[atom]

Una vez iniciado, pulsar **F10**, o bien escribir ***MENU** y **Enter**, para mostrar el menú desde el que se pueden cargar los programas de Atom Software Archive del disco duro virtual.

Acorn Electron

El [Acorn Electron](#) fue una versión barata del BBC Micro de Acorn Computers. El core está basado en el [trabajo original de David Banks \(hoglet\)](#).

Características principales:

- Salida de vídeo compuesto/RGB y VGA 50Hz seleccionables vía teclado
- Soporte SD/MMC, vía archivos de imagen ".MMB"
- Carga de software vía el puerto de entrada de audio del ZX-Uno

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD con la primera partición en formato FAT16 o FAT32. El core usa una ROM especial (Smart SPI) que lee de la SD un archivo que contiene imágenes de disquete.

El archivo debe llamarse **BEEB.MMB** y estar en el directorio raíz. Se puede crear con la utilidad [MMBImager](#) para Windows, disponible en [el repositorio SVN de ZX-Uno](#) (Usuario **guest**, contraseña **zxuno**) o con la utilidades MMB/SSD en perl, disponibles en [GitHub](#).

El fichero debe estar sin fragmentar en la SD. Se puede utilizar algún programa que defragmente ficheros o sistemas de archivos FAT (como [Defraggler para Windows](#)), o bien usar el siguiente método:

1. Formatear la primera partición de la SD en FAT16 o FAT32, pero **NO en formato rápido** (en Windows, desmarcar la casilla de formato rápido).
2. Copiar a la SD el archivo **BEEB.MMB**, de modo que este sea el **PRIMER archivo** que se copia a la SD.
3. Si se desea, ya se pueden copiar otros archivos a la SD para usar con otros cores, pero **SIEMPRE debe mantenerse el BEEB.MMB** como el primero que se copió a la SD.



Como el fichero **MMB** se llama igual que el que utiliza el [core de BBC Micro](#), se puede usar otro core con acceso completo a la tarjeta SD (Como uno de ZX Spectrum con esxDOS), teniendo dos ficheros **MMB** distintos, y renombrar el de Acorn Electron como **BEEB.MMB** antes de ejecutar este core

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Ctrl+Bloq May+1 - Ctrl+Bloq May+4** (se necesita un teclado externo PS/2): Configurar otros modos gráficos (60Hz, etc)
- **F10 (Caps Shift+Symbol Shift+0 en ZXUnGo+)** y **Ctrl+F10 (Caps Shift+Symbol Shift+Z en ZXUnGo+)**: Soft Reset
- **Ctrl+Alt+Backspace (Caps Shift+Symbol Shift+F en ZXUnGo+)** y luego **B** en ZXUnGo+: Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**.

Guía básica

[electron]

Una vez introducida la SD en el ZX-Uno y arrancado el core, si el archivo de **BEEB.MMB** está correctamente creado, al arrancar debería aparecer:

```
Acorn Electron  
Smart SPI  
BASIC  
>
```

Automáticamente se monta el disco 0 de la imagen, y se puede ver su contenido con el comando:

```
*CAT
```

Para cargar, por ejemplo, el menú de juegos que viene en algunas imágenes disponibles en internet, usar el comando:

```
CHAIN"MENU"
```

Para cargar desde la entrada de audio:

```
*TAPE  
CHAIN""
```

Y entonces comenzar la reproducción.

Para ver una lista de los discos disponibles en el fichero **BEEB.MMB**:

```
*DCAT
```

Para insertar un disco concreto en una unidad virtual concreta:

```
*DIN numerodisco numerounidad
```

Adventure Vision

Alice MC10

Altair 8800

Amiga

Amstrad CPC 6128

El [Amstrad CPC 6128](#) fue un ordenador doméstico, sucesor del Amstrad CPC 664 (que solo duró 6 meses en el mercado), y este, a su vez, era sucesor del Amstrad CPC 464.

El core para ZX-Uno de Amstrad CPC 6128 está basado en el proyecto [FPGAmstrad](#) de Renaud Hélias.

Algunas de sus características son:

- VGA: 640x480 VGA centrado a 60Hz
- Selección de discos: El primer disco detectado se inserta en el arranque y la pulsación de una tecla hace reset y carga el siguiente

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD con la primera partición en formato FAT32 (Tipo de partición [0B](#) Win95 FAT-32), de 4GB de tamaño máximo y 4096 bytes por cluster.

Además son necesarios los ficheros ROM siguientes (se pueden obtener [en la wiki oficial del proyecto original](#) o en el [repositorio de GitHub](#)):

- [OS6128.ROM](#)
- [BASIC1-1.ROM](#)
- [AMSDOS.ROM](#)
- [MAXAM.ROM](#)

También se debe incluir uno o más ficheros con imágenes de disco ([DSK](#)) con el software que se quiera ejecutar.

Copiar tanto los ficheros [ROM](#) como los [DSK](#) a la raíz de la partición FAT32.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- [Re Pág](#) ([Caps Shift+Symbol Shift+J](#) en ZXUnGo+): Hace un Reset del Amstrad y carga el siguiente archivo [DSK](#) en orden alfabético.
- En un teclado PS/2, sólo funciona la tecla mayúsculas del lado izquierdo del teclado.

Guía básica

Escribir el comando **CAT** para ver el contenido del fichero DSK cargado actualmente.

[cpc]

Escribir el comando **RUN"<nombre>** para cargar un programa del disco

[cpc2]

Usar la tecla **Re Pág (Caps Shift+Symbol Shift+J** en ZXUnGo+) para hacer reset y cargar el siguiente archivo **DSK** en orden alfabético.

Amstrad PCW

ao486 (PC 486)

Apogee

Apple I

El [Apple I](#) fue uno de los primeros computadores personales, y el primero en combinar un microprocesador con una conexión para un teclado y un monitor.

El core de ZX-Uno ha sido creado por Subcritical, basado en el proyecto [Apple-One](#).

Algunas de sus características son:

- Sólo funciona con la salida VGA
- Incluye Integer Basic

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

El teclado está con la distribución en inglés.

Notar que, en Apple I, no existe tecla de borrado. Debido a la forma en que se muestra el vídeo, no es posible hacer que el cursor retroceda para borrar un carácter, así que se utiliza uno especial: [_](#). Este significa que se ha retrocedido para borrar en la entrada de datos (por ejemplo [PRN_INT](#) es interpretado como [PRINT](#)).

Guía básica

Según se inicia, el Apple I arranca con [Woz Monitor](#) (ROM cargada en `0xFF00`)

[apple1 1]

El intérprete BASIC incluido con el core de Apple I es [Integer BASIC](#). Para iniciararlo, se debe invocar la dirección de memoria correspondiente ([0xE000](#)):

```
\  
E000R
```

Una vez cargado el intérprete, la consola cambia para indicar que está lista para recibir [comandos de BASIC](#):

```
\  
E000R  
  
E000:4C  
>@
```

[apple1 2]

Apple][

La familia de computadores [Apple II](#) fue la primera serie de microcomputadoras de producción masiva hecha por la empresa Apple Computer.

El core de ZX-Uno está basado en el [original de Stephen A. Edwards](#) y en [la adaptación de vlast a las placas Papilio](#).

Algunas de sus características son:

- Soporta Joystick (hasta dos botones de disparo)
- Tarjetas de expansión de RAM. 128K Saturn RAM (slot 5) + 16K Language card (slot 0).
- Scanlines en modo VGA
- Cambio de tipo de monitor entre color y blanco y negro

Para más información consultar [el foro oficial de ZX-Uno](#).

Formato de Tarjeta SD

La tarjeta SD ha de ser en un formato exclusivo, y por tanto no puede ser utilizada con otros cores. Está basada en imágenes de disco **NIB** concatenadas.

Para convertir imágenes de disco en otro formato (**DSK** o **D0**), se puede utilizar la utilidad **dsk2nib**, disponible en [el repositorio SVN de ZX-Uno](#) (Usuario **guest**, contraseña **zxuno**) y en [GitHub](#).



Este proceso borra el contenido que hubiera antes en la tarjeta SD, y no se puede deshacer.

Windows

Concatenar las imágenes (máximo 20) usando **COPY**:

```
COPY /B imagen1.nib + imagen2.nib + (...) + imagen20.nib apple2_20discos.img
```

Volcar la imagen en la SD, por ejemplo, usando [HDD Raw Copy Tool](#).

MacOS y Linux

Concatenar las imágenes (máximo 20) usando **cat**:

```
cat imagen1.nib imagen2.nib (...) imagen20.nib > apple2_20discos.img
```

Volcar la imagen en la SD, usando **dd**:

```
sudo umount /dev/...
sudo dd if=apple2_20discos.img of=/dev/...
```

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- - (teclado numérico) (**Caps Shift+Symbol Shift+R** en ZXUnGo+): Activar o desactivar scanlines en salida VGA.
- * (teclado numérico) (**Caps Shift+Symbol Shift+E** en ZXUnGo+): Alternar entre monitor en color y monitor en blanco y negro.
- F1 a F10 (**Caps Shift+Symbol Shift+1** a **Caps Shift+Symbol Shift+0** en ZXUnGo+): Insertar la imagen de disco 1 a 10 en la tarjeta SD. Pulsar **F12** a continuación.
- Mayús+F1 (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Symbol Shift+1** en ZXUnGo+) a **Mayús+F10** (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Symbol Shift+0** en ZXUnGo+): Insertar la imagen de disco 11 a 20 en la tarjeta SD. Pulsar **F12** a continuación.
- Ctrl+Alt+Backspace (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**.
- F12 (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Soft reset y carga del disco insertado actualmente.



El core necesita tener insertada una tarjeta SD con el formato correcto, o si no, no finaliza el arranque. Esto es porque el Apple][necesita un disco de arranque para funcionar.

Apple Macintosh Plus

Aquarius

Arcade

Arduboy

Astrocade

Atari 800XL

El [Atari 800XL](#) era un ordenador personal lanzado por Atari en la década de los 80.

La versión para ZX-Uno tiene estas características:

- Memoria ampliada hasta 320K, seleccionable
- Soporte de unidad de disco (hasta 4 unidades), mediante tarjeta SD
- Soporte de cartuchos, también mediante carga por SD
- Soporte para vídeo compuesto y VGA
- Scanlines en modo VGA
- Soporte de joystick norma Atari

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT32. En la raíz de la tarjeta debe haber un directorio llamado `atari800`, con dos subdirectorios: `rom` con distintas ROM a utilizar (ej: `ATARIXL.ROM`), y `user` con ficheros de disco, cartucho, etc. (ej: `ManicMin.xex`)

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- - (`Caps Shift+Symbol Shift+R` en ZXUnGo+): Activar o desactivar scanlines en modo VGA
- * (`Caps Shift+Symbol Shift+E` en ZXUnGo+): Alternar el tipo de máquina entre PAL y NTSC
- `Ctrl+Alt+Backspace` (`Caps Shift+Symbol Shift+F` y luego `B` en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de `Enter`
- `F5` (`Caps Shift+Symbol Shift+5` en ZXUnGo+): `Help`
- `F6` (`Caps Shift+Symbol Shift+6` en ZXUnGo+): `Start`
- `F7` (`Caps Shift+Symbol Shift+7` en ZXUnGo+): `Select`
- `F9` (`Caps Shift+Symbol Shift+9` en ZXUnGo+): `Reset`
- `F11` (`Caps Shift+Symbol Shift+Q` en ZXUnGo+): Cargar un disco
- `F12` (`Caps Shift+Symbol Shift+W` en ZXUnGo+): Acceder al menú de opciones
- El teclado numérico emula un joystick. Las teclas `5` y `2` sirven indistintamente para la dirección *abajo* y `0` es el disparo

Guía básica

Pulsando **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+) se muestra el menú de configuración. Se usan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+) y **Enter** (o bien el joystick y el botón de disparo) para elegir y seleccionar en el menú.

[a800xl]

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Acelerar la velocidad de la CPU (CPU Turbo)
- Acelerar la velocidad de lectura de los discos (Drive Turbo)
- Cambiar el tipo de Ram
- Cargar una Rom desde la SD
- Cargar un disco en la unidad 1 (Drive 1)
- Cargar un disco en la unidad 2 (Drive 2)
- Cargar un disco en la unidad 3 (Drive 3)
- Cargar un disco en la unidad 4 (Drive 4)
- Cargar un cartucho (Cart)
- Cargar la memoria desde una grabación anterior (Load Memory)
- Guardar el estado de la memoria (Save memory)
- Salir (Exit)

Atari 2600

La [Atari 2600](#) es una videoconsola lanzada al mercado bajo el nombre de Atari VCS (Video Computer System).

La versión para ZX-Uno está desarrollada por Quest y DistWave..

Algunas de las características del core son:

- Dos modos de vídeo seleccionables: RGB y VGA
- Compatible con joystick

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los ficheros con las imágenes ROM que se deseen cargar.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **W, A, S, D** (personalizables) o el joystick 1: Controles de dirección del jugador 1
- **F** (personalizable) o el botón de disparo del joystick 1: Disparo del jugador 1
- **I, J, K, L** (personalizables) o el joystick 2: Controles de dirección del jugador 2
- **H** (personalizable) o el botón de disparo del joystick 2: Disparo del jugador 2
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Personalización de los controles

Los controles por defecto del core con soporte de teclado se pueden modificar. Para ello, se ha de crear en la raíz de la tarjeta SD un fichero llamado **KEYSP1** para el primer jugador y **KEYSP2** para el segundo (se puede descargar uno de ejemplo desde [el foro de ZX-Uno](#)).

Estos ficheros se pueden crear con un editor hexadecimal (como [HxD para Windows](#)) y están formados por una secuencia de bytes indicando en este orden: **Arriba**, **Abajo**, **Izquierda**, **Derecha**, **Botón de disparo**. Los códigos a utilizar en la secuencia se pueden consultar en la tabla [Scan Codes](#) al final del manual (Columna **MAKE**), pero, teniendo en cuenta que, si el código está emparejado con **0xE0**, se ha de usar el otro valor, sumándole **0x80** (por ejemplo, para **E0 14**, sería **0x94**).



Por ejemplo, la secuencia **15 1C 44 4D 29** se correspondería con **Q A O P Espacio**

Guía básica

Pulsando **Esc** ([Caps Shift+Espacio](#) en ZXUnGo+) o el botón 2 del joystick se muestra el menú de configuración. Se usan las teclas de cursor ([Caps Shift+6](#) y [Caps Shift+7](#) en ZXUnGo+) y **Enter** para elegir y seleccionar en el menú.

[a2600uno]

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Reiniciar el core (Reset)
- Línea de exploración (Scanlines)
- Modo RGB (PAL/NTSC)
- Color (Color)
- Dificultad A (Difficulty A)
- Dificultad B (Difficulty B)
- Select
- Start
- Cargar ROM (Load ROM)
- Salir del menú (Exit)

Atari 5200

Atari 7800

Atari Lynx

Atari ST/STe

BBC Micro B,Master

El [BBC Micro](#), fue uno de los primeros ordenadores domésticos, diseñado y construido por Acorn Computers para la British Broadcasting Corporation (BBC).

El core para ZX-Uno ha sido creado por Quest y mejorado posteriormente por azesmbog y hoglet.

Algunas de sus características son:

- Salida vídeo RGB y VGA (con scanlines opcionales)
- Soporte SD/MMC, vía archivos de imagen [.MMB](#)
- Soporte de joystick en el puerto 1 (Emula Joy analógico)
- Implementación del chip de sonido sn76489 desde el [proyecto PACE](#) (Programmable Arcade Circuit Emulation)

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD con la primera partición en formato FAT16 o FAT32. El core usa una ROM especial (MMFS en los cores más recientes, Smart SPI en los más antiguos) que lee de la SD un archivo [BEEB.MMB](#), que contiene imágenes de disquete.

El archivo debe llamarse exactamente [BEEB.MMB](#) y estar en el directorio raíz. Se puede crear con la utilidad [MMBImager](#) para Windows, disponible en [el repositorio SVN de ZX-Uno](#) (Usuario [guest](#), contraseña [zxuno](#)) o con la utilidades MMB/SSD en perl, disponibles en Github [aquí](#) o [aquí](#).

El fichero [MMB](#) debe estar sin fragmentar en la SD. Se puede utilizar algún programa que defragmente ficheros o sistemas de archivos FAT (como [Defraggler para Windows](#)), o bien usar el siguiente método:

1. Formatear la primera partición de la SD en FAT16 o FAT32, pero **NO en formato rápido** (en Windows, desmarcar la casilla de formato rápido).
2. Copiar a la SD el archivo [BEEB.MMB](#), de modo que este sea el **PRIMER archivo** que se copia a la SD.
3. Si se desea, ya se pueden copiar otros archivos a la SD para usar con otros cores, pero **SIEMPRE debe mantenerse el BEEB.MMB** como el primero que se copió a la SD.



Como el fichero [MMB](#) se llama igual que el que utiliza el [core de Acorn Electron](#), se puede usar otro core con acceso completo a la tarjeta SD (Como uno de ZX Spectrum con esxDOS), teniendo dos ficheros [MMB](#) distintos, y renombrar el de BBC Micro como [BEEB.MMB](#) antes de ejecutar este core

Teclado

[keyboardbbc]

Teclas especiales y botones

Durante la ejecución del core:

- - del teclado numérico (**Caps Shift+Symbol Shift+R** en ZXUnGo+): Activa o desactiva scanlines
- **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset
- **Mayús+F12** (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Symbol Shift+W** en ZXUnGo+): Soft Reset intentando cargar automáticamente el disco seleccionado en **BEEB.MMB**
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**.

Guía básica

[bbcmicro]

Una vez introducida la SD en el ZX-Uno y arrancado el core, si el archivo de **BEEB.MMB** está correctamente creado, al arrancar debería aparecer:

```
BBC Computer 32k  
Model B MMFS  
BASIC  
>
```

O bien, si es un core con Smart SPI

```
BBC Computer 32k  
Smart SPI  
BASIC
```

Automáticamente se monta el disco 0 de la imagen, y se puede ver su contenido con el comando:

```
*CAT
```

Para cargar, por ejemplo, un programa llamado **MENU**, y que esté en el disco usar el comando:

```
*MENU
```

Para ver una lista de los discos disponibles en el fichero **BEEB.MMB**:

*DCAT

Para insertar un disco concreto:

*DIN numerodisco



Recordar, tras insertar un disco, que si este tiene arranque automático, se puede iniciar pulsando **Mayús+F12**.

AY-3-8500

BK0011M

Computers Lynx48,96

El [Lynx](#) fue un ordenador doméstico británico de 8 bits lanzado a principios de 1983 por la compañía Computers. Se lanzaron en total tres modelos, con 48kB, 96kB o 128kB de RAM.

La versión para ZX-Uno ha sido [desarrollada por Kyp069](#) y tiene las siguientes características:

- Modos 48kB y 96 kB
- ROM Scorpion opcional
- Carga usando la entrada de audio
- Soporte para joystick

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **F6** ([Caps Shift+Symbol Shift+6](#) en ZXUnGo+): Alterna entre el modo 48kB y el modo 96kB (por defecto)
- **F7** ([Caps Shift+Symbol Shift+7](#) en ZXUnGo+): Alterna entre activar o desactivar la ROM Scorpio (activa por defecto)
- **F8** ([Caps Shift+Symbol Shift+8](#) en ZXUnGo+): Activar o desactivar el tener en cuenta los bits 2 y 3 del puerto \$80 (señal CAS del banco 2), para que se vean bien los juegos de Level 9.-
Ctrl+Alt+Supr: Reset
- **F11** ([Caps Shift+Symbol Shift+Q](#) en ZXUnGo+): Reset
- **Ctrl+Alt+Backspace** ([Caps Shift+Symbol Shift+F](#) y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[lynx]

Desde BASIC, se suele cargar desde una cinta (u otro dispositivo externo de audio) con la secuencia de comandos:

```
TAPE n  
LOAD "NOMBRE"
```

Donde **n** es un número (entre 1 y 5), según como se haya realizado la grabación, y **NOMBRE** es, obligatoriamente el nombre a cargar desde la cinta.

Si no se sabe el nombre a cargar, se puede averiguar con la misma secuencia de comandos, pero escribiendo **LOAD ""**.

Para ficheros binarios, se debe usar **MLOAD** en vez de **LOAD**.

Es posible utilizar programas como [Lynx2Wav](#) con los ficheros **TAP** de cinta de Lynx. Los ficheros de audio obtenidos se pueden embeber a su vez dentro de ficheros TSX o TZX con herramientas como MakeTSX o RetroConverter.

El script [lince](#) facilita todo este proceso, permitiendo crear directamente ficheros **TZX** desde ficheros **TAP** de Lynx.

Chess

CHIP-8

CHIP-8 es un lenguaje de programación interpretado, desarrollado por Joseph Weisbecker. Fue inicialmente usado en los microcomputadores de 8 bits COSMAC VIP y Telmac 1800 a mediados de 1970. CHIP-8 tiene un descendiente llamado SCHIP (Super Chip), presentado por Erik Bryntse.

El core de ZX-Uno está basada en una implementación ya existente [para FPGA](#) de la máquina virtual de SuperChip.

Existen múltiples sitios como [CHIP-8 Archive](#) o [CHIP-8 de Matthew Mikolay](#) donde se puede obtener software para este tipo de máquinas.

Formato de Tarjeta SD

Se puede utilizar una tarjeta con la primera partición en formato FAT16 o FAT32 para almacenar ficheros ROM en formato **BIN** o **CH8** para usar con el core.

Teclado

La máquina CHIP-8 utiliza un teclado hexadecimal como entrada. La asignación en el teclado es la siguiente:

Chip-8	Teclado
1 2 3 C	1 2 3 4
4 5 6 D	Q W E R
7 8 9 E	A S D F
A 0 B F	Z X C V

Teclas especiales y botones

Durante la ejecución del core:

- **Esc** (**Caps Shift+Espacio** en ZXUnGo+): para mostrar u ocultar el menú.
- **F11** (**Caps Shift+Symbol Shift+Q** en ZXUnGo+): Hard Reset
- **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset

Guía Básica

Pulsando **Esc** (**Caps Shift+Espacio** en ZXUnGo+) se muestra el menú de configuración. Se usan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+), o bien **W** y **S**, y **Enter** para elegir y seleccionar las distintas opciones.

[chip8uno]

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Reiniciar el core (Reset)
- Cambiar la velocidad de reloj del core (Clock Speed)
- Cargar un archivo de ROM desde la tarjeta SD (Load Rom)
- Activar o desactivar el sonido (Sound On/Off)
- Ayuda sobre el uso del teclado (Keyboard Help)
- Salir del menú (Exit)

ColecoVision, SG-1000

[ColecoVision](#) es una consola de videojuegos lanzada al mercado por la empresa Coleco.

La versión para ZUNO+ está basada el proyecto [de Fabio Belavenuto](#).

Algunas de las características del core son:

- La ROM de la BIOS se carga desde la tarjeta SD
- Soporta ROM multicartucho, que también se carga desde la SD
- Soporte para Joystick

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los ficheros con las imágenes ROM y otros archivos necesarios. Los archivos se pueden descargar desde la [web del proyecto original en GitHub](#).

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- Cursor ([Caps Shift+5](#), [Caps Shift+6](#), [Caps Shift+7](#) y [Caps Shift+8](#) en ZXUnGo+) o [Q](#), [A](#), [E](#), [R](#) o el joystick 1: Controles de dirección del jugador 1
- [Z](#) o el botón de joystick 1: Botón de disparo 1 del jugador 1
- [U](#), [J](#), [O](#), [P](#) o el joystick 2: Controles de dirección del jugador 2
- [M](#) o el botón de joystick 2: Botón de disparo 1 del jugador 2
- [X](#) o el botón secundario de joystick 1: Botón de disparo 2 del jugador 1 y del jugador 2
- [0](#) a [9](#): Botones del 0 al 9 del jugador 1 y el jugador 2
- [T](#): Botón '*'
- [Y](#): Botón '#'
- 'Esc' ([Caps Shift+Espacio](#) en ZXUnGo+): Soft Reset

Guía básica

Al iniciar, la ROM de la BIOS se carga desde la tarjeta SD, así como la ROM multicartucho.

[coleco]

En el menú multicartucho, usar los controles de dirección para elegir la ROM a cargar, y luego el botón de disparo 1 para cargar la ROM elegida. Pulsando **Esc** (**Caps Shift+Espacio** en ZXUnGo+) se reinicia el core y se vuelve a cargar el menú de selección de ROM.

Color Computer 2, Dragon 32

Color Computer 3

Commodore 16, Plus/4

El [Commodore 16](#) fue un ordenador doméstico fabricado por la empresa Commodore International en 1984.

El core de ZX-Uno está basado en el [proyecto FPGATED de István Hegedus](#), con algunos pequeños cambios y mejoras, como la lectura de cassetes reales por audio.

Características:

- Implementa un Commodore 16 PAL ampliado a 64K de RAM
- Disquetera 1541, redirigida a SD y en RAW. Sólo lectura (compatible con imágenes .D64)
- Chip TED de FPGATED
- Soporte de joystick, tanto real (conector DB9) como emulado en teclado numérico
- Switch VGA 50Hz / RGB-Compuesto
- Scanlines en modo VGA
- Carga de cinta usando la entrada de audio
- Cambio de polaridad de la señal de entrada de audio
- ROM Kernal PAL -5 modificada para evitar la espera de pulsación tras la cabecera en las cargas de cassette
- Mezcla de audio de salida compuesta por el feedback de audio del cassette (más bajo) y el audio principal del chip TED
- LED testigo de lectura de la 1541 y también del estado de la polaridad de EAR

Formato de Tarjeta SD

La tarjeta SD ha de ser en un formato exclusivo, y por tanto no puede ser utilizada con otros cores. Está basada en imágenes de disco [D64](#) concatenadas, en bloques de 256K. Descargar el fichero [dummyto256.bin](#) disponible en [el repositorio oficial de ZX-Uno](#).

Para incluir varios ficheros [PRG](#) en una imagen de disco [D64](#), se puede utilizar el [programa DirMaster](#) para Windows, creando una imagen cuyo primer programa sea [FB16.PRG](#) (más detalles en [el foro de ZX-Uno](#)).



El formato RAW de SD es compatible con el que usa también el core de Commodore 64, así que se puede utilizar una misma tarjeta para los dos, incluyendo imágenes de disco con programas para los dos sistemas.



Este proceso borra el contenido que hubiera antes en la tarjeta SD, y no se puede deshacer.

Windows

Concatenar las imágenes usando [COPY](#):

```
COPY /B imagen1.d64 + dummyto256.bin + imagen2.d64 + dummyto256.bin + (...)  
c16rawsd.img
```

Volcar la imagen en la SD, por ejemplo, usando [HDD Raw Copy Tool](#).

MacOS y Linux

Concatenar las imágenes usando [cat](#):

```
cat imagen1.d64 dummyto256.bin imagen2.d64 dummyto256.bin (...) > c16rawsd.img
```

Volcar la imagen en la SD, usando [dd](#):

```
sudo umount /dev/...  
sudo dd if=c16rawsd.img of=/dev/...
```

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Esc** (**Caps Shift+Espacio** en ZXUnGo+): Esc
- **Tab**: RUN/STOP
- **Windows izquierda** (**Caps Shift+Symbol Shift+x** en ZXUnGo+): = Commodore
- Teclado Numérico: Joystick emulado
- **F1** (**Caps Shift+Symbol Shift+1** en ZXUnGo+) a **F3** (**Caps Shift+Symbol Shift+3** en ZXUnGo+): F1 a F3
- **Bloq. Num** (**Caps Shift+Symbol Shift+I** en ZXUnGo+) o **F4** (**Caps Shift+Symbol Shift+4** en ZXUnGo+): HELP
- **Insert** (**Caps Shift+Symbol Shift+G** en ZXUnGo+): Seleccionar el primer disco de la tarjeta SD
- **Re.Pag** (**Caps Shift+Symbol Shift+J** en ZXUnGo+): Seleccionar el siguiente disco
- **Av.Pag** (**Caps Shift+Symbol Shift+N** en ZXUnGo+): Seleccionar el disco anterior
- **Ctrl+Re.Pag** (**Caps Shift+Symbol Shift+Z** y luego **Caps Shift+Symbol Shift+J** en ZXUnGo+): Avanzar 10 discos
- **Ctrl+Av.Pag** (**Caps Shift+Symbol Shift+Z** y luego **Caps Shift+Symbol Shift+N** en ZXUnGo+): Retroceder 10 discos
- + del teclado numérico (**Caps Shift+Symbol Shift+D** en ZXUnGo+): Cambiar la polaridad de la entrada de audio
- - del teclado numérico (**Caps Shift+Symbol Shift+R** en ZXUnGo+): Activar o desactivar scanlines en modo VGA
- **F11** (**Caps Shift+Symbol Shift+Q** en ZXUnGo+): Seleccionar la asignación del joystick al puerto 0 o al puerto 1
- **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard Reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[c16]

Desde BASIC, se puede cargar desde una cinta (u otro dispositivo externo de audio) con el comando **LOAD**. Una vez finalizada la carga, escribir **RUN** y pulsar **Enter** si fuese necesario.



Es posible utilizar programas como [Audiotap](#) o [Comodoro](#) con los ficheros **TAP** de cinta de Commodore.

Para listar el directorio del disco seleccionado, pulsar **F3** ([Caps Shift+Symbol Shift+3](#) en ZXUnGo+) o introducir el comando **DIRECTORY**. Para cargar un fichero del disco, usar el comando **DLOAD"<nombre de fichero>"** y luego, normalmente, **RUN**.

Para cargar rápidamente el primer fichero **PRG** de un disco pulsar **Mayús+TAB** o usar el comando **DLOAD"***", y luego, normalmente, **RUN**.

Commodore 64, Ultimax

Commodore 64 (C64, CBM 64/CBM64, C=64,C-64, VIC-641) era una [computadora doméstica de 8 bits](#) que fue desarrollada por Commodore International.

El core de ZX-Uno está desarrollado por Quest.

Características:

- Commodore 64 PAL con 64K de RAM
- Disquetera 1541 completa, redirigida a SD y en RAW. Sólo lectura.
- JiffyDOS opcional tanto para la 1541 como para el Commodore 64. Esto acelera muy notablemente las cargas.
- Chip de sonido SID
- Soporte de joystick, tanto real (norma Atari, SMS, megadrive...) como emulado en pad numérico
- Switch VGA 50Hz / RGB-Compuesto
- Scanlines en modo VGA
- Carga de cinta usando la entrada de audio
- Cambio de polaridad de la señal de entrada de audio
- Kernel modificado para evitar la espera de pulsación tras la cabecera en las cargas de cassette.
- LED testigo de lectura de la disquetera 1541.

Formato de Tarjeta SD

La tarjeta SD ha de ser en un formato exclusivo, y por tanto no puede ser utilizada con otros cores. Está basada en imágenes de disco D64 concatenadas, en bloques de 256K. Descargar el fichero [dummyto256.bin](#) disponible en [el repositorio oficial de ZX-Uno](#).

Para incluir varios ficheros PRG en una imagen de disco D64, se puede utilizar el [programa DirMaster](#) para Windows, creando una imagen cuyo primer programa sea FB64.PRG.



El formato RAW de SD es compatible con el que usa también el core de Commodore 16, así que se puede utilizar una misma tarjeta para los dos, incluyendo imágenes de disco con programas para los dos sistemas.



Este proceso borra el contenido que hubiera antes en la tarjeta SD, y no se puede deshacer.

Windows

Concatenar las imágenes usando COPY:

```
COPY /B imagen1.d64 + dummyto256.bin + imagen2.d64 + dummyto256.bin + (...)  
c64rawsd.img
```

Volcar la imagen en la SD, por ejemplo, usando [HDD Raw Copy Tool](#).

MacOS y Linux

Concatenar las imágenes usando cat:

```
cat imagen1.d64 dummyto256.bin imagen2.d64 dummyto256.bin (...) > c64rawsd.img
```

Volcar la imagen en la SD, usando dd:

```
sudo umount /dev/...  
sudo dd if=c64rawsd.img of=/dev/...
```

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Esc** (**Caps Shift+Espacio** en ZXUnGo+): Start/Stop
- **Tab**: RUN/STOP
- **Alt** (**Caps Shift+Symbol Shift+C** en ZXUnGo+): = Commodore
- Teclado Numérico: Joystick emulado
- **F1** (**Caps Shift+Symbol Shift+1** en ZXUnGo+) a **F8** (**Caps Shift+Symbol Shift+8** en ZXUnGo+): F1 a F8
- **F9** (**Caps Shift+Symbol Shift+9** en ZXUnGo+): Tecla del símbolo de la libra
- **F10** (**Caps Shift+Symbol Shift+0** en ZXUnGo+): Tecla del símbolo +
- **Insert** (**Caps Shift+Symbol Shift+G** en ZXUnGo+): Seleccionar el primer disco de la tarjeta SD
- **Re.Pag** (**Caps Shift+Symbol Shift+J** en ZXUnGo+): Seleccionar el siguiente disco
- **Av.Pag** (**Caps Shift+Symbol Shift+N** en ZXUnGo+): Seleccionar el disco anterior
- **Ctrl+Re.Pag** (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Symbol Shift+J** en ZXUnGo+): Avanzar 10 discos
- **Ctrl+Av.Pag** (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Symbol Shift+N** en ZXUnGo+): Retroceder 10 discos
- + del teclado numérico (**Caps Shift+Symbol Shift+D** en ZXUnGo+): Cambiar la polaridad de la entrada de audio
- - del teclado numérico (**Caps Shift+Symbol Shift+R** en ZXUnGo+): Activar o desactivar scanlines en modo VGA
- **Fin** (**Caps Shift+Symbol Shift+B** en ZXUnGo+) = Cambiar paleta de colores
- **F11** (**Caps Shift+Symbol Shift+Q** en ZXUnGo+): Rotar entre puertos de joystick
- **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset modo JiffyDOS
- **Ctrl+F12** (**Caps Shift+Symbol Shift+Z** y luego **Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset modo ROM C64 original (para carga de cintas)
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[c64]

Desde BASIC, se puede cargar desde una cinta (u otro dispositivo externo de audio) con el comando **LOAD**. Una vez finalizada la carga, escribir **RUN** y pulsar **Enter** si fuese necesario.



Para cargar desde cinta se debe estar usando la ROM de C64 original que se activa con **Ctrl+F12** (**Caps Shift+Symbol Shift+Z** y luego **Caps Shift+Symbol Shift+W** en ZXUnGo+).



Es posible utilizar programas como [Audiotap](#) o [Comodore](#) con los ficheros **TAP** de cinta de Commodore.

Para cargar desde un disco, habitualmente, se puede pulsar **Mayus+Esc** (**Caps Shift+Symbol Shift+A** y luego **Caps Shift+Espacio** en ZXUnGo+) o bien escribir **LOAD "*",8,1** y pulsar **Enter**. Una vez aparezca **READY** en la pantalla, escribir **RUN** y pulsar **Enter** para ejecutar el programa.

Si el disco tuviera varios programas, usar **F1** (**Caps Shift+Symbol Shift+1** en ZXUnGo+) para listar el contenido, o bien escribir **LOAD "\$"** y pulsar **Enter**. A continuación, escribir **LIST**, y pulsar **Enter**, para ver una lista con los archivos dentro del disco.

Ahora, para cargar el archivo deseado, escribir **LOAD "<nombre>",8** (donde **<nombre>** es el nombre del archivo a cargar) y pulsar **Enter**. Una vez aparezca **READY** en la pantalla, escribir **RUN** y pulsar **Enter** para ejecutar el programa. Si esto no funcionase, probar con el comando **LOAD "<nombre>",8,1**.



Para cargar desde disco se debe estar usando la ROM de C64 con JiffyDOS que se activa con **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+).

Commodore PET

El [Commodore PET \(Personal Electronic Transactor\)](#) fue una línea de ordenadores domésticos producida por Commodore a partir de 1977.

La versión para ZX-Uno ha sido [desarrollada por Jepalza](#), basada en el proyecto [pet2001fpga](#).

Características:

- Carga usando la entrada de audio
- Sólo funciona en VGA

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

El teclado está con la distribución en inglés.

Teclas especiales y botones

Durante la ejecución del core:

- **F1** ([Caps Shift+Symbol Shift+1](#) en ZXUnGo+): Cancelar carga desde cinta
- **Alt** ([Caps Shift+Symbol Shift+C](#) en ZXUnGo+): Permite escribir en modo gráfico

Guía básica

[pet]

Se puede cargar desde una cinta (u otro dispositivo externo de audio) con el comando **LOAD**. Una vez finalizadada la carga, normalmente, se ha de escribir **RUN** y pulsar **Enter**.



Es posible utilizar programas como [Audiotap](#) o [Comodoro](#) con los ficheros **TAP** de cinta de Commodore.

Commodore VIC-20

El **Commodore VIC-20** (Commodore VC20 en Alemania, VIC-1001 en Japón) fue un ordenador doméstico de 8 bits fabricado y diseñado por Commodore Business Machines, lanzado en junio de 1980, cerca de 3 años después del Commodore PET.

La versión para ZX-Uno ha sido desarrollada por McLeod y Quest

Características principales:

- Expansión de 32K de RAM. Para mayor compatibilidad se puede desactivar.
- Carga usando la entrada de audio
- Soporte para video compuesto y VGA
- Soporte de joystick

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **F9** (`Caps Shift+Symbol Shift+9` en ZXUnGo+) a **F11** (`Caps Shift+Symbol Shift+Q` en ZXUnGo+): Cargar cartuchos incorporados al core (si los hubiera)
- - del teclado numérico (`Caps Shift+Symbol Shift+R` en ZXUnGo+): Cambiar la polaridad de la entrada de audio. El LED rojo se enciende cuando la polaridad está invertida
- **Bloq. Num** (`Caps Shift+Symbol Shift+I` en ZXUnGo+): Reset desactivando la ampliación de memoria de 32K (activa por defecto)
- **F12** (`Caps Shift+Symbol Shift+W` en ZXUnGo) : Reset activando la ampliación de memoria
- **Ctrl+Alt+Backspace** (`Caps Shift+Symbol Shift+F` y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[vic20]

Se puede cargar desde una cinta (u otro dispositivo externo de audio) con el comando **LOAD**. Una vez finalizadada la carga, normalmente, se ha de escribir **RUN** y pulsar **Enter**.



Es posible utilizar programas como [Audiotap](#) o [Comodoro](#) con los ficheros **TAP** de cinta de Commodore.

Compukit UK101

DEC PDP-1

EDSAC

Einstein TC01

Emerson Arcadia 2001

Epoch Galaxy II

Flappy Bird

[Flappy Bird](#) fue un juego para móviles de gran éxito entre 2013 y 2014. El objetivo consiste en controlar un pájaro intentando volar entre filas de tuberías verdes sin tocarse con estas.

Este core ha sido realizado por azesmbog basado en el [trabajo de Max Veerapat Kumchom](#), adaptando su código para la placa AEON Lite al ZX-Uno.

Características principales:

- Sólo salida VGA
- Sólo se puede controlar usando un joystick

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

Este core no utiliza el teclado. Necesita un joystick para usarse.

Teclas especiales y botones

Durante la ejecución del core:

- **Joystick Arriba**: Reinicia el juego
- **Joystick Abajo**: Master Reset
- **Disparo**: Volar

Fairchild Channel F

Game of Life

Gameboy, Gameboy Color

Gameboy, Gameboy Color 2P

Gameboy Advance

Gameboy Advance 2P

Galaksija

[Galaksija](#) (Galaxia en español) es un ordenador creado en 1983 por el inventor serbio Voja Antonić. Los esquemas fueron publicados en la primera revista de ordenadores yugoslava, "Računari u vašoj kući" (Ordenadores en Casa), en su primer número de Enero de 1984. De este primer número se vendieron 100.000 ejemplares, y posteriormente se supo que al menos 8.000 personas habían construido su propio ordenador Galaksija.

El core de ZX-Uno es una versión del [proyecto µGalaksija](#), mejorada por azesmog, añadiendo el modo Galaksija Plus, y la posibilidad de cargar programas desde una tarjeta SD.

Características del core de ZX-Uno:

- Sólo salida VGA
- Sonido opcional
- Carga de programas desde la tarjeta SD
- Modos Galaksija Clásico (minus) y Galaksija Plus

Formato de Tarjeta SD

Se puede utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los programas. Estos se pueden conseguir, por ejemplo, en la web de [retrospec.sgn.net](#).

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **F1** (**Caps Shift+Symbol Shift+1** en ZXUnGo+): Velocidad de ejecución normal
- **F2** (**Caps Shift+Symbol Shift+2** en ZXUnGo+): Velocidad de ejecución 2x
- **F3** (**Caps Shift+Symbol Shift+3** en ZXUnGo+): Velocidad de ejecución 4x
- **F9** (**Caps Shift+Symbol Shift+9** en ZXUnGo+): Habilitar o deshabilitar sonido
- **Esc** (**Caps Shift+Espacio** en ZXUnGo+): Congela la ejecución de un programa BASIC o Código Máquina. Volviendo a pulsar, la ejecución continua
- **Supr** (**Caps Shift+Symbol Shift+V** en ZXUnGo+): **CLR**: Detiene la ejecución de un programa BASIC mientras se mantenga pulsada
- **Fin** (**Caps Shift+Symbol Shift+B** en ZXUnGo+): Si se pulsa durante la ejecución de un programa BASIC realiza un BREAK
- **Mayús.+M**: Habilita el modo plus más avanzado en aquellos programas que estén preparados para usarlo (por ejemplo **TETRIS.GTP**)
- **M**: Vuelve al modo normal desde el modo plus
- **Ctrl+Alt+F5**: NMI
- **Ctrl+Alt+Supr** (**Caps Shift+Symbol Shift+F** y luego **N** en ZXUnGo+): Soft Reset
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[galaksija]

Desde BASIC, se puede habilitar la carga de programas desde la tarjeta SD, ejecutando la siguiente instrucción:

```
A=USR(&F000)
```

Ahora, con instrucciones como **CD <directorio>** y **DIR** se puede llegar hasta el directorio donde se encuentre el fichero a cargar. Para cargar un fichero se tiene que escribir sólo el nombre, sin la extensión (por ejemplo, para un fichero **JUEGO.GTP** se debe escribir **JUEGO**). Una vez cargado en memoria, el programa se ejecuta con la instrucción **RUN**.

El core se inicializa en modo Galaksija Clásico (minus), pero es posible pasar al modo Galaksija Plus con la instrucción:

```
A=USR(&E000)
```

Se puede consultar [en Wikipedia](#) los comandos de BASIC disponibles.

Genesis/Megadrive

Intellivision

Interact Home Computer

Jupiter Ace

El [Jupiter ACE](#) fue un microordenador británico lanzado al mercado en 1982 por la empresa Jupiter Cantab.

El core para ZX-Uno ha sido desarrollado por McLeod y mejorado por Azesmbog, y tiene, entre otras, las siguientes características:

- Soporte para carga desde dispositivos de audio externo
- Soporte de acceso a la tarjeta SD (implementando [SDOS](#))

Formato de Tarjeta SD

Se puede utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar ficheros en formato [.TAP](#) de Jupiter ACE. Estos se pueden conseguir, por ejemplo, en la web de [Jupiter ACE Archive](#).

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **F1** (`Caps Shift+Symbol Shift+1` en ZXUnGo+) a **F4** (`Caps Shift+Symbol Shift+4` en ZXUnGo+): Modos Turbo
- **Ctrl+Alt+F5** (`Caps Shift+Symbol Shift+Y` en ZXUnGo+): NMI
- **Ctrl+Alt+Supr** (`Caps Shift+Symbol Shift+F` y luego **N** en ZXUnGo+): Reset
- **Ctrl+Alt+Backspace** (`Caps Shift+Symbol Shift+F` y luego **B** en ZXUnGo+): Hard Reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

Desde Forth, se puede cargar desde una cinta (u otro dispositivo externo de audio) con el comando `LOAD <nombre>`.



Los ficheros **TAP** de Jupiter ACE son diferentes de los de ZX Spectrum, y, para poder utilizarse con algunos reproductores, han de ser convertidos previamente a **TZX** con la utilidad `acetap2zx` disponible en el [repositorio de ZX-Uno](#) (Usuario `guest`, contraseña `zxuno`).

Para cargar ficheros de cinta desde la tarjeta SD, se debe acceder a SDOS escribiendo

```
61440 call
```

Ahora, con instrucciones como **CD**, **DIR** se puede llegar hasta el directorio donde se encuentre el fichero **TAP** a cargar. Para cargar un fichero de cinta **.TAP** se tiene que escribir sólo el nombre, sin la extensión (por ejemplo, para un fichero **JUEGO.TAP** se debe escribir **JUEGO**).



Todos los comandos de SDOS se deben escribir en letras mayúsculas.

Una vez la cinta ha cargado, escribir el comando necesario para iniciar el programa.



En lugar de BASIC como otros sistemas, el Jupiter ACE utiliza **Forth**. Para ver los comandos disponibles (incluyendo los que inician un programa cargado desde cinta), se tiene que utilizar **vlist**.

```
[ace vlist]
```

Laser 310

MegaCD

MSX

MSX1FPGA es un proyecto para clonar MSX1 en una FPGA. El desarrollo original es de Fabio Belavenuto y se encuentra disponible [en GitHub](#).

Algunas de sus características son:

- MSX1 a 50Hz o 60Hz;
- Utiliza Nextor ROM con un controlador para SD
- Mapa de teclado configurable
- Simulación de línea de exploración (Scanlines)
- Soporte para joystick

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD con la primera partición en formato FAT16 con [código 0x06 \(16-bit FAT\)](#). Es también posible utilizar una segunda partición FAT16 para albergar todo el software, dejando la primera sólo para arrancar el sistema.

Se ha de obtener lo siguiente:

- Ficheros básicos del proyecto para la SD [desde GitHub](#)
- Controlador ([NEXTOR.SYS](#)) y ROM ([NEXTOR.ROM](#)) de Nextor [también desde GitHub](#)
- ROM de MSX1 ([MSX_INT.rom](#), [MSX_JP.rom](#) o [MSX_USA.rom](#)) [en el mismo repositorio](#)

Una vez descargado, copiar el contenido del [directorio SD](#) en la raíz de la primera partición de la tarjeta SD.

Copiar [NEXTOR.SYS](#) en el mismo lugar.

Copiar [NEXTOR.ROM](#) en el directorio [MSX1FPGA](#).

Copiar la ROM deseada de MSX1 ([MSX_INT.rom](#), [MSX_JP.rom](#) o [MSX_USA.rom](#)) en el directorio [MSX1FPGA](#), pero usando el nombre [MSX1BIOS.ROM](#).

En el fichero [/MSX1FPGA/config.txt](#) se guarda la configuración del core, según este formato:

```
11SP01
|||||
|||||+-Modo de línea de exploración: 1=Activo, 0=Inactivo
|||||---Turbo: 1=Arrancar con el modo turbo activo
|||+---Sistema de color: N=NTSC, P=PAL
||+---Mapa de Teclado: E=Inglés, B=Brasileño, F=Francés, S=Castellano
|+---Scandoubler(VGA): 1=Activo, 0=Inactivo
+---Nextor: 1=Activo, 0=Inactivo
```

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **F1** (`Caps Shift+Symbol Shift+F1` en ZXUnGo+) a **F5** (`Caps Shift+Symbol Shift+F5` en ZXUnGo+): **F1** a **F5**
- **Impr Pant** (`Caps Shift+Symbol Shift+T` en ZXUnGo+): Cambia el modo entre VGA y RGB
- **Bloq. Desp** (`Caps Shift+Symbol Shift+Y` en ZXUnGo+): Cambia el modo de línea de exploración (Scanlines)
- **Pausa** (`Caps Shift+Symbol Shift+U` en ZXUnGo+): Cambia entre 50Hz y 60Hz
- **F11** (`Caps Shift+Symbol Shift+Q` en ZXUnGo+): Activa o desactiva el modo turbo
- **Ctrl+Alt+Supr** (`Caps Shift+Symbol Shift+F` y luego **N** en ZXUnGo+): Soft Reset
- **Ctrl+Alt+Backspace** (`Caps Shift+Symbol Shift+F` y luego **B** en ZXUnGo+): Hard Reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**
- **ALT Izquierdo** (`Caps Shift+Symbol Shift+C` en ZXUnGo+): MSX GRAPH
- **ALT Derecho** (`Caps Shift+Symbol Shift+M` en ZXUnGo+): MSX CODE
- **Re Pág** (`Caps Shift+Symbol Shift+J` en ZXUnGo+): MSX SELECT
- **Inicio** (`Caps Shift+Symbol Shift+H` en ZXUnGo+): MSX HOME (`Mayús+HOME` o `Caps Shift+Symbol Shift+A` y luego `Caps Shift+Symbol Shift+H` en ZXUnGo+): CLS)
- **Fin** (`Caps Shift+Symbol Shift+B` en ZXUnGo+): MSX STOP
- **Ñ o Windows** (`Caps Shift+Symbol Shift+X` en ZXUnGo+): MSX DEAD



En BASIC, se puede usar **CTRL + STOP** (`Ctrl+Fin` o `Caps Shift+Symbol Shift+Z`) y luego `Caps Shift+Symbol Shift+B` en ZXUnGo+ para detener la ejecución de un programa.



Para cambiar el modo de vídeo entre 50Hz y 60Hz (para ejecución correcta de programas PAL a través de VGA), se puede usar también **DISPLAY.COM**, que se puede obtener [en este hilo del foro de MSX](#).

Guía básica

Para acceder a BASIC desde MSX-DOS, ejecutar el comando **BASIC**.

[msx]

Desde BASIC, se puede cargar desde una cinta (u otro dispositivo externo de audio) con los comandos **RUN"CAS:", BLOAD"CAS:", R** o **CLOAD**.



Para que la carga desde audio funcione, el modo turbo tiene que estar desactivado.

Para acceder a MSX-DOS desde BASIC, ejecutar **CALL SYSTEM**.

MSXCTRL

Se trata de una utilidad exclusiva del core MSX1FPGA, que permite controlar todas las opciones del core que antes solo eran accesibles a través del fichero de configuración o pulsando determinadas teclas.

Al ejecutar **MSXCTRL** se muestran los parámetros de uso:

```
MSXCTRL.COM - Utility to manipulate MSX1FPGA core.
HW ID = 06 - ZX-Uno Board
Version 1.3
Mem config = 82
Has HWDS = FALSE
```

Use:

```
MSXCTRL -h -i -r -b -[5|6] -m<0-2>
-c<0-1> -d<0-1> -t<0-1>
[-w<filename> | -l<filename>]
-k<0-255> -e<0-255> -p<0-255>
-s<0-255> -o<0-255> -a<0-255>
```

MSXCTRL -h muestra ayuda para cada parámetro. Así, **MSXCTRL -i** presenta la configuración actual, los parámetros **-t 1** encienden el modo turbo, etc.

Otros

Existen múltiples sistemas para cargar los juegos dependiendo del tipo de archivo: .CAS, .DSK o ROM (ver [este hilo del foro de ZX-Uno](#) para más información).

El mapeo de para teclado español disponible con la distribución oficial se puede cambiar por otro más completo. Ver [aquí](#) para más información.

MultiComp

NEC PC8801

NeoGeo

NES

Nintendo Entertainment System (también conocida como Nintendo NES o simplemente NES) es la [segunda consola de sobremesa de Nintendo](#).

La versión para ZX-Uno ha sido creada para ZX-Uno por [DistWave y Quest](#), basado en el [core de NES de Ludde/Strigeus para la placa Nexys4](#).

Algunas de las características del core son:

- Sólo soporta salida VGA y utiliza timings poco rigurosos, por lo que es posible que de problemas en algunos monitores
- Filtro HQ2X que "despixeliza" la imagen
- Simulación de línea de exploración (Scanlines)
- Utiliza el reloj de la NES NTSC, por tanto funcionan correctamente las ROMs USA. Las ROMs PAL van más rápido de lo que deberían
- Permite cargar ROMS desde la SD
- El core básico utiliza los 512 KB de SRAM del ZX-Uno. Se ha dividido la memoria en 2 bancos de 256 KB, uno para PRG_ROM y otro para CHR_ROM. Por tanto, las ROMs que utilizan más de 256 KB para CHR o PRG no funcionan. Evidentemente cualquier ROM mayor de 512 KB tampoco.
- Existen otras versiones del core adaptadas para 2 MB de SRAM y que permiten la carga de juegos que usen hasta 1MB de CHR RAM y 1MB de PRG RAM (2MB en total).
- Compatible con 1 o 2 joysticks

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los ficheros con las imágenes ROM (extensión [.NES](#)) de los juegos que se desee cargar. Los ficheros pueden estar en subdirectorios.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Esc** (**Caps Shift+Espacio** en ZXUnGo+) o botón 2 del joystick: Mostrar u ocultar el menú de configuración
- Teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+), y **Enter** para usar el menú
- **W, A, S, D** (personalizables) o el joystick 1: Controles de dirección del jugador 1
- **F** y **R** (personalizables) o botones de disparo del joystick 1: Botones A y B (Jugador 1)
- **5** y **1**: **Select** y **Start** (Jugador 1)
- **I, J, K, L** (personalizables) o el joystick 2: Controles de dirección del jugador 2
- **H, Y** (personalizables) o botones de disparo del joystick 1: Botones A y B (Jugador 2)
- **6** y **2**: **Select** y **Start** (Jugador 1)
- **F4** (**Caps Shift+Symbol Shift+F** en ZXUnGo+): Intercambiar la asignación de teclas de jugador 1 y jugador 2
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard Reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Personalización de los controles

Los controles por defecto del core con soporte de teclado se pueden modificar. Para ello, se ha de crear en la raíz de la tarjeta SD un fichero llamado **KEYSP1`** para el primer jugador y **KEYSP2** para el segundo (se puede descargar uno de ejemplo desde [el foro de ZX-Uno](#)).

Estos ficheros se pueden crear con un editor hexadecimal (como [HxD para Windows](#)) y están formados por una secuencia de bytes indicando en este orden: **Arriba, Abajo, Izquierda, Derecha, Botón A, Botón B**. Los códigos a utilizar en la secuencia se pueden consultar en la tabla **Scan Codes** al final del manual (Columna MAKE), pero, teniendo en cuenta que, si el código está emparejado con **0xE0**, se ha de usar el otro valor, sumándole **0x80** (por ejemplo, para **E0 14**, sería **0x94**).



Por ejemplo, la secuencia **15 1C 44 4D 29 22** se correspondería con **Q A O P Espacio X**, y la secuencia **1D 1B 23 2B 1A 22** se correspondería con **W S D F Z X**

Guía básica

Pulsando **Esc** (**Caps Shift+Espacio** en ZXUnGo+) se muestra el menú de configuración. Para desplazarse por el menú y activar o elegir alguna opción, se utilizan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+) y **Enter**.

[nesuno]

En él se pueden activar o desactivar las siguientes opciones:

- Reiniciar la NES (Reset NES)
- Activar o desactivar línea de exploración (Scanlines)
- Encender o apagar el filtro que suaviza la imagen (HQ2X Filter)
- Simular la pulsación del botón Select del mando 1 (P1 Select)
- Simular la pulsación del botón Start del mando 1 (P1 Start)
- Elegir una ROM para cargar desde la SD (Load ROM)
- Salir del menú (Exit)



Se puede avanzar y retroceder más rápido por las opciones de menú usando **Re Pág** (**Caps Shift+Symbol Shift+J** en ZXUnGo+) y **Av Pág** (**Caps Shift+Symbol Shift+N** en ZXUnGo+).

Odyssey2

La [Philips Videopac](#), también conocida como Magnavox Odyssey², Philips Videopac G7000 o Philips Odyssey² es una videoconsola lanzada en 1978.

La versión para ZX-Uno está desarrollada por avlixa, basada en el core de ZX DOS de yomboprime.

Algunas de las características del core son:

- Dos modos de vídeo seleccionables: RGB/Video Compuesto y VGA
- Necesita Joystick(s) para funcionar
- Modos de color incluyendo monocromo o fósforo verde/naranja
- Soporte para modificar el tipo de letra (charset) de la ROM de VDC

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los ficheros con las imágenes ROM que se deseen cargar.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Bloq. Despl.** ([Caps Shift+Symbol Shift+Y](#) en ZXUnGo+): para cambiar entre modo de vídeo RGB/Vídeo Compuesto y VGA
- **F2** ([Caps Shift+Symbol Shift+2](#) en ZXUnGo+): Alternar entre RGB y vídeo compuesto cuando no se está usando el modo VGA
- **F3** ([Caps Shift+Symbol Shift+3](#) en ZXUnGo+): Reset
- **Ctrl+Alt+Backspace** ([Caps Shift+Symbol Shift+F](#) y luego [B](#) en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de [Enter](#)
- Tras cargar una ROM, la mayoría de los juegos mostrarán un mensaje "SELECT GAME". Pulsar una tecla entre **0** y **9** o el botón del joystick para jugar.
- **Esc** ([Caps Shift+Espacio](#) en ZXUnGo+) o el botón 2 del joystick para mostrar u ocultar el menú.
- **W, A, S, D** o las teclas de cursor ([Caps Shift+5](#), [Caps Shift+6](#), [Caps Shift+7](#) y [Caps Shift+8](#) en ZXUnGo+) y luego [Enter](#) para elegir y seleccionar en el menú.

Guía básica

Pulsando **Esc** (**Caps Shift+Espacio** en ZXUnGo+) o el botón 2 del joystick se muestra el menú de configuración. Se usan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+) y **Enter** para elegir y seleccionar en el menú.

[videopacuno]

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Reiniciar el core (Reset)
- Línea de exploración (Scanlines)
- Intercambiar el orden de los joysticks (Swap Joysticks)
- Unir los joysticks (Join Joysticks). Útil para ROMs que alternan el joystick en las partidas
- Cargar ROM (Load Cartridge ROM)
- Cargar un tipo de letra alternativo (Load VDC Font)
- Modo de vídeo (PAL VideoPac/NTSC Odyssey2)
- Modo de color (Color Mode)
- Salir del menú (Exit)



El sistema no tiene claramente definido qué joystick se corresponde a cada jugador, e incluso puede irse alternando para algunos juegos, así que puede ser necesario ir cambiando entre los dos, o bien (para partidas de un sólo jugador) usar la opción del menú que unifica los dos joysticks.



No suele haber información en pantalla de las opciones, así que es interesante consultar las instrucciones de cada juego (por ejemplo en [este enlace](#)) para saber qué hacer para jugar.



Si, al navegar por el directorio donde están las ROMS, no se ven todas, se puede dividir el contenido en varios subdirectorios con menos ficheros por directorio (por ejemplo, clasificando por letras) para solucionarlo.

Cambio del charset de la ROM de VDC

Es posible, para algunas ROM, cargar un archivo **CHR** con una fuente modificada, en vez de la original incluida en el chip Intel 8244/8245.

Estos archivos se pueden crear siguiendo las instrucciones y utilizando el editor disponible en el repositorio del proyecto, siguiendo [este enlace](#).

Ondra SPO 186

El [Ondra SPO 186](#) era un ordenador checoslovaco de 8 bits desarrollado en 1985 en Tesla Liberec como ordenador escolar y doméstico barato.

La [versión para ZX-Uno](#) ha sido desarrollada por PetrM1, basada en la versión anterior para MiSTer, y tiene estas características:

- 64 kB de RAM
- Soporte para teclado
- Sonido
- Carga de programas a través de la entrada de audio
- Carga de programas vía puerto serie

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

Ondra tiene teclado de 37 teclas, en el que el uso de teclas modificadoras permite que cada tecla sirva para 2 o 3 caracteres diferentes. Cada tecla modificadora debe ser presionada (y mantenida) antes de presionar la tecla de carácter a modificar.

[keyboradondra]

Teclas especiales y botones

- **Mayúsculas:** Shift
- **Alt:** Symbols
- **Tab:** Numbers
- **Control:** Ctrl
- **Bloq. Mayús:** ČS (se usa para los diacríticos checos)

Guía básica

[ondraspo1]

Una vez iniciado el core, se puede cargar desde una cinta (u otro dispositivo externo de audio) pulsando la tecla **Enter** (si se hubiera escrito algo, se ha de borrar todo utilizando **Ctrl+Cursor Izquierdo**). A continuación, la pantalla comenzará a parpadear y aparecerá el texto

.KÓD 1

[ondraspo2]

Inicie la reproducción de la cinta y el texto en pantalla irá cambiando durante la carga.

Orao

Oric 1 & Atmos

Otra versión distinta de core de [Oric Atmos](#), basada en la versión de Rampa para MiST, Mistica y SiDi.

Esta versión ha sido [desarrollada por Kyp069](#) y tiene las siguientes características:

- Salida de vídeo compuesto o VGA (Scandoubler)
- 64KB de RAM y 16KB de ROM. La ROM oculta la parte alta de la memoria, pero se puede acceder a todos los bancos de memoria mediante ensamblador
- ROM seleccionable entre la versión original (1.1), y la versión 1.22, ROM alternativa que corrige errores
- Las direcciones del joystick están mapeadas a las teclas del cursor ([Caps Shift+5](#), [Caps Shift+6](#), [Caps Shift+7](#) y [Caps Shift+8](#) en ZXUnGo+), y los dos botones de disparo a las teclas [Espacio](#) y [S](#).
- Carga de programas a través de la entrada de audio
- Grabación de programas a través de la salida de audio

Para más información consultar el [foro RetroWiki](#).

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

La disposición de las teclas en el teclado de PC intenta ser lo más parecida a la que tiene un Oric Atmos real.

La distribución del teclado está en inglés (así, por ejemplo, la tecla `-` se corresponde con `'`, al lado del número `0`).

[keyboardoric]

Teclas especiales y botones

Durante la ejecución del core:

- **F1 (Caps Shift+Symbol Shift+1 en ZXUnGo+)**: Seleccionar la ROM 1.1
- **F2 (Caps Shift+Symbol Shift+2 en ZXUnGo+)**: Seleccionar la ROM 1.22
- **F5 (Caps Shift+Symbol Shift+5 en ZXUnGo+)**: NMI
- **F10 (Caps Shift+Symbol Shift+0 en ZXUnGo+)**: Habilita la salida de audio al ejecutar `CSAVE`
- **Ctrl+Alt+Backspace (Caps Shift+Symbol Shift+F y luego B en ZXUnGo+)** o **F11 (Caps Shift+Symbol Shift+Q en ZXUnGo+)**: Hard reset. Backspace es la tecla de borrar hacia atrás, encima de `Enter`
- **Ctrl+Alt+Supr (Caps Shift+Symbol Shift+F y luego N en ZXUnGo+)** o **F12 (Caps Shift+Symbol Shift+W en ZXUnGo+)**: Soft reset
- **Bloq. Despl.** (Caps Shift+Symbol Shift+Y en ZXUnGo+): para cambiar entre modo de vídeo compuesto y VGA
- **Alt Gr.** (Caps Shift+Symbol Shift+M en ZXUnGo+): Tecla `FUNC`
- **Esc (Caps Shift+Espacio en ZXUnGo+)**: Tecla `ESC`
- **Supr (Caps Shift+Symbol Shift+V en ZXUnGo+)**: Tecla `DEL`
- **Ctrl (Caps Shift+Symbol Shift+Z en ZXUnGo+)**: Tecla `CTRL`
- **Mayus. (Caps Shift en ZXUnGo+)**: Tecla `SHIFT`
- **Windows Izq., Alt Izq, Windows Der., Menú**: Teclas de cursor `Izquierda, Abajo, Arriba, Derecha`

Guía básica

[oric]

Desde BASIC, se puede cargar desde una cinta (u otro dispositivo externo de audio) usando el comando **CLOAD ""** y **Enter**. A continuación, aparecerá en pantalla el mensaje

Searching...

Cuando se detecte un programa en la carga, el texto cambiará a

Loading...<nombre de programa>

En los [foros de RetroWiki](#) hay disponible un script de linux llamado **calorico.sh** que usa bash y PHP, junto con las utilidades [TSXphpclass](#) de [Natalia Pujol](#) para convertir ficheros **.TAP** de Oric en archivos **.TSX**.



Para grabar un programa BASIC usando la salida de audio, se puede utilizar el comando **CSAVE "NOMBRE"** y **Enter**. Si se desea poder escuchar durante la grabación, se debe haber habilitado previamente el sonido de grabación con la tecla **F10** (**Caps Shift+Symbol Shift+0** en ZXUnGo+).



No debe estar activo el sonido de grabación (tecla **F10** o **Caps Shift+Symbol Shift+Z** en ZXUnGo+) durante una carga desde audio externo. El motivo es que la activación del sonido de grabación provoca una retroalimentación con un ruido alto y agudo que impide la carga por audio.

PMD 85

RX-78

SAM Coupe

El [SAM Coupé](#) fue un ordenador doméstico desarrollado y comercializado a partir de 1989 por Miles Gordon Technology, basado en el Sinclair ZX Spectrum.

El core de ZX-Uno tiene las siguientes características:

- Soporte VGA (con scanlines) y RGB / Video compuesto
- Soporte de teclado completo
- Sonido SAA1099
- Carga desde dispositivos de audio externos

Formato de Tarjeta SD

Este core no utiliza la tarjeta SD.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- **Ctrl+Alt+F5**: NMI.
- **Ctrl+Alt+Supr(Caps Shift+Symbol Shift+F** y luego **N** en ZXUnGo+): Soft Reset
- **Ctrl+Alt+Backspace (Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard Reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**

Guía básica

[samcoupe]

Sólo es posible cargar programas desde una cinta (u otro dispositivo externo de audio) con el comando **LOAD""** (o pulsando **7** del teclado numérico, equivalente a pulsar **F7** en el teclado del SAM Coupé).



En el repositorio SVN de ZX-Uno ([aquí](#) - Usuario **guest**, contraseña **zxuno**) se pueden encontrar varios programas adaptados a fichero de cinta TZX.

Sega Master System, Game Gear

La [Master System](#) fue una consola de videojuegos de 8 bits basada en cartuchos y tarjetas, desarrollada por Sega.

El core de ZX-Uno está basado en el [trabajo original de Ben para Papilio Plus](#).

Algunas de sus características son:

- Soporte para VGA y Video compuesto o RGB, con sincro PAL (50Hz)
- Soporte de joysticks
- Emulación de joystick vía teclado
- Lectura de archivos desde la tarjeta SD

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16, para almacenar los ficheros con las imágenes (extensión [.SMS](#) o [.BIN](#)) de los juegos que se desee cargar. Los ficheros pueden estar en subdirectorios.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- Cursor ([Caps Shift+5](#), [Caps Shift+6](#), [Caps Shift+7](#) y [Caps Shift+8](#) en ZXUnGo+) o [Q](#), [A](#), [O](#), [P](#) o el joystick 1: Controles de dirección del jugador 1
- [Z](#), [Windows](#) o el botón de joystick 1: Botón de disparo 1 del jugador 1
- [X](#), [Espacio](#) o el botón secundario de joystick 1: Botón de disparo 2 del jugador 1
- [Pausa](#) ([Caps Shift+Symbol Shift+U](#) en ZXUnGo+): Tecla de botón de pausa de Master System
- [Bloq. Desp](#) ([Caps Shift+Symbol Shift+Y](#) en ZXUnGo+): para cambiar entre modo de vídeo RGB/Vídeo Compuesto y VGA
- [Ctrl+Alt+Backspace](#) ([Caps Shift+Symbol Shift+F](#) y luego [B](#) en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de [Enter](#)
- [F12](#) ([Caps Shift+Symbol Shift+W](#) en ZXUnGo+): Reset y mostrar el selector de ficheros

Guía básica

Pulsando **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+) se muestra el selector de archivos **SMS** o **BIN**. Para desplazarse por el menú y elegir el archivo a cargar, se utilizan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+) y **Enter**.

[sms]

Sharp MZ Series

Sinclair QL

Slug Cross

SNES

Sord M5

Specialist/MX

SV-328

TI-99/4A

TomyTronic Scramble

TRS-80 Model 1

TSConf

TurboGrafx 16 / PC Engine

VC4000, Interton

Vector 06C

Vectrex

La [Vectrex](#) era una consola con pantalla integrada y con los gráficos basados en vectores, en vez de píxeles.

El core para ZX-Uno está basado en el [trabajo original de DarFPGA](#).

Algunas de sus características son:

- Soporte VGA
- Menú OSD
- Controles vía teclado y vía joystick
- Tamaño máximo de la ROM a cargar: 8K

Formato de Tarjeta SD

Se debe de utilizar una tarjeta SD, con la primera partición en formato FAT16 o FAT32, para almacenar los ficheros con las imágenes (extensión **.BIN** o **.VEC**) de los juegos que se desee cargar. Los ficheros pueden estar en subdirectorios.

En [archive.org](#) se puede obtener software para Vectrex.

Teclado

Teclas especiales y botones

Durante la ejecución del core:

- Cursor (**Caps Shift+5**, **Caps Shift+6**, **Caps Shift+7** y **Caps Shift+8** en ZXUnGo+) o el joystick 1: Controles de dirección
- **Z**: Botón de disparo 1
- **X** o el botón principal de joystick: Botón de disparo 2
- **C** o el botón secundario de joystick: Botón de disparo 3
- **V**: Botón de disparo 4
- **Esc** (**Caps Shift+Espacio** en ZXUnGo+): para mostrar u ocultar el menú.
- **Ctrl+Alt+Backspace** (**Caps Shift+Symbol Shift+F** y luego **B** en ZXUnGo+): Hard reset. Backspace es la tecla de borrar hacia atrás, encima de **Enter**
- **F12** (**Caps Shift+Symbol Shift+W** en ZXUnGo+): Reset

Guía básica

Por defecto la Vectrex arranca con el juego que trae en memoria de fábrica (Mine Storm).

[vectrex]

Pulsando **Esc** se muestra el menú de configuración. Se usan las teclas de cursor (**Caps Shift+6** y **Caps Shift+7** en ZXUnGo+) y **Enter** para elegir y seleccionar en el menú.

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Reiniciar el core (Reset)
- Utilizar blanco neutro para los vectores (Disable bluish vectors)
- Cargar un fichero desde la SD (Load ROM)
- Salir del menú (Exit)

WonderSwan

X68000

ZX Spectrum

ZX Spectrum Next

<https://forum.tlienhard.com/phpBB3/viewtopic.php?t=4031>

ZX Spectrum Next es un proyecto, basado en FPGA, que aspira a ser la evolución de los ordenadores Sinclair ZX Spectrum, manteniendo la compatibilidad hardware y software con los modelos anteriores, pero añadiendo nuevas características.

Principalmente gracias a avlixa, existe una versión del core de ZX Spectrum Next sintetizada para usarse con ZXDOS+.

El core para ZXDOS+ no tiene, por el momento, implementada ninguna de las siguientes características:

- Beeper interno
- Conector de expansión EDGE
- Módulo RTC
- Teclado de membrana
- Grabación de cores adicionales o actualización del propio core Next desde el core Next
- Salida MIC
- vídeo HDMI
- Utilización de puerto de conexión joystick para comunicación UART

Por otra parte, puede disponer de las siguientes características, que no existen en el core original:

- Modos de color incluyendo monocromo o fósforo verde/naranja
- Salida de audio I²S (con el [addon RTC+I²S+Pizero](#))

El manual de uso se puede descargar desde [la página oficial](#).



Para poder utilizar una Raspberry Pi, se necesita una versión del Core con el soporte necesario, y el addon RTC+I²S+Pizero. Ver la [sección correspondiente](#) de otro hardware para más detalles.

Formato de Tarjeta microSD

Se debe de utilizar una tarjeta microSD con la primera partición en formato FAT16 o FAT32, y que tenga instalada la distribución de esxdos correspondiente a la configuración actual de BIOS (ver el apartado correspondiente de esxdos para más información).

Obtener la distribución de NextZXOS [en la página oficial](#).

Descomprimir el contenido de NextZXOS en la tarjeta microSD, pero modificando el archivo `config.ini` en `/machines/next` para que contenga (si no existiera ya) la línea `ps2=0` (para asegurar que se utiliza correctamente el puerto del teclado) y la línea `intbeep=0` para apagar el zumbador interno (este último paso no es necesario en el caso de un gomaDOS+).

Teclado

Teclas especiales y botones

Las combinaciones específicas de gomaDOS+ que se indican a continuación se corresponden con el modo de teclado ZX. Véase el apartado dedicado a los modos de teclado de gomaDOS+ para más información. También se pueden utilizar las equivalentes en el modo de teclado PC XT.

Notar que `Ctrl+Alt+backspace` no funciona con el core de Spectrum Next. Hay que apagar manualmente y volver a encender si se desea cambiar a otro core. Tampoco hay botón físico de Reset o Drive.

Durante la ejecución del core:

- **F1 (Caps Shift+Symbol Shift+1 en gomaDOS+)**: Hard Reset
- **F2 (Caps Shift+Symbol Shift+2 en gomaDOS+)**: Scandoubler. Dobra la resolución. Debería estar apagado para conexiones vía SCART
- **F3 (Caps Shift+Symbol Shift+3 en gomaDOS+)**: Alternar la frecuencia vertical entre 50Hz y 60Hz
- **F4 (Caps Shift+Symbol Shift+4 en gomaDOS+)**: Soft Reset
- **F7 (Caps Shift+Symbol Shift+7 en gomaDOS+)**: Scanlines
- **F9 (Caps Shift+Symbol Shift+9 en gomaDOS+)**: NMI
- **F10 (Caps Shift+Symbol Shift+0 en gomaDOS+)**: divMMC NMI. Simula la pulsación del botón Drive. Si se usa con mayúsculas, fuerza volver a buscar unidades de almacenamiento y cargar la pantalla de arranque en esxdos
- **F11 (Caps Shift+Symbol Shift+Q en gomaDOS+)**: En las versiones más recientes del core, activar distintos modos de color monocromo
- **F12 (`Caps Shift+Symbol Shift+W en gomaDOS+)**: Alternar entre la salida audio estándar y la salida I²S, si está conectado el [addon RTC+I²S+PIO](#). Notar que activar la salida I²S inhabilita parcialmente el sonido de la Raspberry Pi.

Guía básica

Al iniciarse la primera vez, aparecerán una serie de pantallas de ayuda. Tras pulsar la tecla **Espacio**, se mostrará el menú de inicio de NextZXOS.

[next]

Se puede navegar utilizando las teclas de cursor, las teclas **5**, **6**, **7** y **8**, o un joystick (si se ha configurado en modo Kempston, MD o cursor). **Enter** o el botón del joystick selecciona un elemento.

La opción **More...** muestra un segundo menú con más opciones.

[next2]

Si se elige **Browser**, se cargará el navegador de NextZXOS, desde el que es posible desplazarse viendo el contenido de la tarjeta microSD y cargar directamente diferentes tipos de archivo (TAP, NEX, DSK, SNA, SNX, Z80, Z8, etc.).



El navegador muestra las entradas de archivos y directorios en el orden de la tabla FAT interna, y no de manera alfabética. Si se desea ver esta información ordenada, se debe reorganizar la estructura de la tarjeta con una utilidad como **FAT Sorter** para Windows, **FATsort** para Linux y MacOS, **YAFS**, **SDSorter** u otros.



En el caso de utilizarse también la tarjeta con el core de **PC XT**, **no se debe utilizar ninguna utilidad de reordenación de FAT** ya que esto puede provocar que deje de arrancar correctamente DOS en dicho core.

[next3]

El core de ZX Spectrum Next para ZX DOS+ necesita el **acelerador basado en Raspberry Pi** para cargar ficheros TZX.



Por defecto, no es posible cargar ficheros TRD desde el navegador (se debe configurar NextZXOS para cargar una "personalidad" con esxdos).

Para más información, consultar el [manual de uso oficial](#).

ZX81

El computador personal [Sinclair ZX81](#), lanzado por Sinclair Research en 1981, fue el de menor precio de la época.

La versión para ZXDOS+ ha sido creada por avlixa, basada en la página de Grant Searle's sobre el ZX80

Características:

- Se puede elegir entre ZX80 y ZX81 (ZX80 sólo con salida RGB por el momento)
- 16k/32k/48k RAM packs
- 8KB con addon CHR\$128/UDG (sin probar)
- QS CHRS (sin probar)
- CHROMA81
- Turbo en modo lento: NoWait, x2, x8
- Chip de sonido YM2149 (compatible con ZON X-81)
- Múltiples joysticks (Cursor, Sinclair, ZX81, ZXpand)
- Timings PAL y NTSC
- Turbo carga de ficheros .o y .p
- Carga de ROM alternativa
- Carga de programas a través de la entrada de audio

Formato de Tarjeta microSD

Se debe de utilizar una tarjeta microSD, con la primera partición en formato FAT16 o FAT32, para almacenar los ficheros con archivos de cinta o ROM que se deseen cargar.

Se puede copiar, si se desea, un fichero [ZX8X.ROM](#) (disponible [the repositorio oficial](#) en el directorio [/zx81/roms](#): se trata de los ficheros ZX81 rom (8k) + ZX80 rom (4k) concatenados.

Teclado

El teclado no está mapeado a la distribución PS/2 estándar, y mantiene la distribución de la máquina original. Así, por ejemplo, para obtener " se ha de pulsar **Mayús+P** o para borrar, **Mayús+Ø**.

ZX80

[keyboardZX80]

ZX81

[keyboardZX81]

Teclas especiales y botones

Durante la ejecución del core:

- **F1 (Caps Shift+Symbol Shift+1 en gomaDOS+)**: Habilitar o deshabilitar los caracteres alternativos
- **F5 (Caps Shift+Symbol Shift+5 en gomaDOS+)** o el botón 2 del joystick para mostrar u ocultar el menú.
- **F9 (Caps Shift+Symbol Shift+9 en gomaDOS+)**: Deshabilita o habilita la salida MIC al conector de audio, porque algunos juegos generan un ruido molesto
- **F10 (Caps Shift+Symbol Shift+Ø en gomaDOS+)**: Habilita o deshabilita sonido de entrada por la salida de audio, para escuchar la cinta si se carga por audio
- **Bloq. Despl. (Caps Shift+Symbol Shift+G en gomaDOS+)**: para cambiar entre modo de vídeo RGB y VGA
- **Ctrl+Alt+Supr (Caps Shift+Symbol Shift+B en gomaDOS+)**: Reset
- **Ctrl+Alt+Backspace (Caps Shift+Symbol Shift+B en gomaDOS+)**: Hard reset.

Guía básica

Pulsando **F5** (**Caps Shift+Symbol Shift+5** en gomaDOS+) o el botón 2 del joystick se muestra el menú de configuración. Se usan las teclas de cursor (**Caps Shift+5**, **Caps Shift+6**, **Caps Shift+7** y **Caps Shift+8** en gomaDOS+, en modo de teclado **PC XT**) y **Enter** para elegir y seleccionar opciones del menú.

[zx81]

En él se pueden activar, desactivar o configurar las siguientes opciones:

- Reiniciar el core (Reset)
- Cargar un archivo de cinta (Load Tape)
- Cargar una ROM alternativa (Load ROM)
- Configurar opciones (Configuration Options)
- Salir del menú (Exit)

[zx81 2]

- Elegir el modelo de computador (Computer Model): ZX80/ZX81
- Tamaño de RAM (Main RAM): 16K/32K
- RAM baja (Low RAM): Off/8KB
- Joystick: Cursor/Sinclair/ZX81
- Habilitar o deshabilitar QS CHRS
- Configuración de CHR\$128/UDG: 128 chars/64 chars/Disabled
- Habilitar o deshabilitar Chroma81: Disabled/Enabled
- Vídeo inverso (Inverse Video): Off/On
- Borde negro (Black Border): Off/On
- Modo de velocidad (Slow mode speed): Original/No Wait/x2
- Frecuencia de vídeo (Video frequency): 50Hz/60Hz

Desde BASIC, se puede cargar desde un fichero cinta, que se haya seleccionado desde el menú, con el comando **LOAD""** y pulsando **Enter**.



Algunos monitores dejan de reproducir la entrada de audio si se desactiva la señal de vídeo. Se recomienda conectar unos auriculares o altavoces externos para poder oír el sonido durante la carga de una cinta.

Se pueden cargar archivos **.p** con color y caracteres alternativos.

Para que funcione el coloreado, se debe activar CHROMA81 antes de la carga. Igualmente, para los caracteres alternativos, QS CHRS debería estar activo antes de cargar.



Las opciones recomendadas para la mayoría de los juegos son:

Main RAM: 16KB Low RAM: 8KB CHR\$128: 128 chars QS CHRS: enabled
CHROMA81: enabled

Referencias

Enlaces

[Wiki oficial](#)

[Foros oficiales](#)

[Servidor oficial de Discord](#)

https://github.com/venice1200/MiSTER_i2c2oled_Pictures

https://github.com/antoniovillena/MiSTER_DB9/blob/master/Docs/db9-user-io-and-snac8.md

<https://www.antoniovillena.es/store/product/umt32pi/>

<https://github.com/dwhinham/mt32-pi#-quick-start-guide>

<https://github.com/dwhinham/mt32-pi/wiki/MiSTER-FPGA%3A-Setup-and-usage>

https://github.com/antoniovillena/MiSTER_DB9/blob/master/Docs/decapod.md

<https://github.com/dwhinham/mt32-pi/wiki/MiSTER-FPGA%3A-Setup-and-usage>

https://github.com/theyspsilon/Update_All_MiSTER

https://github.com/MiSTER-devel/Main_MiSTER/wiki/Main-Joystick-Mapping

https://boogermann.github.io/Bible_MiSTER/getting-started/inputs/joystick-mapping/

https://github.com/MiSTER-devel/ao486_MiSTER

<https://archive.org/details/top-300-final>

https://github.com/flynnnsbit/Top300_updates

<https://www.intel.com/content/www/us/en/developer/articles/guide/terasic-de10-nano-get-started-guide.html>