

minidvino

Manual



Índice

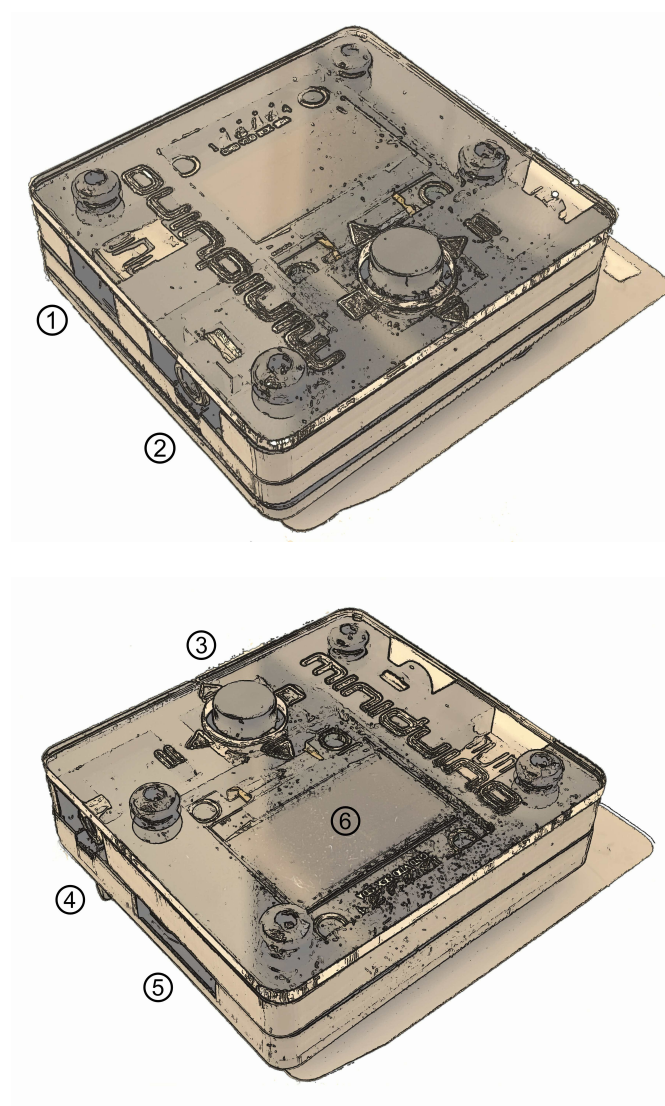
Introducción	1
Puertos y Botones	2
Preparación	3
Uso	4
Creación de ficheros TZX o TSX desde otros formatos	6
MakeTSX.....	6
RetroConverter	6
Actualización de firmware Maxduino.....	7
Modelo STM32.....	7
Preparación del entorno	7
Actualización	9
Modelo ATMega328P.....	10
Preparación del entorno	10
Actualización	10

Introducción

[Miniduino](#) es un sistema reproductor de archivos de cinta, basado en un microcontrolador STM32F103C8T6 o ATmega38P, y que trae instalado de serie el firmware [Maxduino](#).

Maxduino permite la reproducción, de una forma similar a como se manejaban las cintas de [cassette](#) originales, de archivos digitales de cinta en múltiples formatos como [TAP](#) y [TZX](#) (ZX Spectrum), [O](#) (ZX80), [P](#) (ZX81), [CDT](#) (Amstrad CPC), [CAS](#)(MSX) [TSX](#) (MSX, Acorn, etc). También es posible reproducir como cinta archivos de sonido AY, para cargarlos en [SpecAY](#) desde ZX Spectrum.

Puertos y Botones



1	Alimentación
2	Salida de sonido
3	Botón de control
4	Control de motor
5	Ranura de tarjeta SD
6	Pantalla

Preparación

Se necesita una tarjeta SD para almacenar los archivos de cinta que se quiere reproducir. Se recomienda que no sea de alta velocidad (Clase 10 o superior) porque puede haber problemas durante la lectura, ni de alta capacidad (SDXC o superior).

La tarjeta debe estar formateada con con la primera partición en formato FAT16 o FAT32.

Además de la tarjeta, recordar utilizar un cable adecuado para la [entrada de sonido del ZXUNO+](#), conectando el Miniduino.

Para los equipos que soporten control por motor, también se puede utilizar un cable con un jack de 2,6 mm.

En la primera partición de la tarjeta se han de copiar los archivos de cinta ([TAP](#), [TZX](#), [O](#), [P](#), [CAS](#), [TSX](#), etc), que se pueden organizar en carpetas o directorios.



El reproductor 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.

Uso

Una vez insertada la tarjeta SD con ficheros de datos, el Miniduino se enciende conectando el cable de alimentación USB incluido.



Si se pulsa directamente el botón de control, se accede al menú de opciones que permite modificar lo siguiente:

- Velocidad (Baud Rate): Para ajustar la velocidad turbo en bloques 4B en archivos de MSX (.CAS y .TSX)
- Control de motor (Motor Ctrl): Para indicar que se ha conectado un cable de control remoto a un equipo que lo soporte (Amstrad, CPC, MSX, etc.)
- Conversión (TSXCzxpUEFWS): Activa la carga turbo para los archivos .CAS y .TSX, cambiar la polaridad de la señal de audio de los archivos para Spectrum y Amstrad CPC y/o cambiar la paridad en los archivos .UEF de Acorn Electron y BBC Micro
- Saltar bloques (Skip BLK): Para deshabilitar (Skip ON) o habilitar la pausa automática al encontrar bloques del tipo 2A

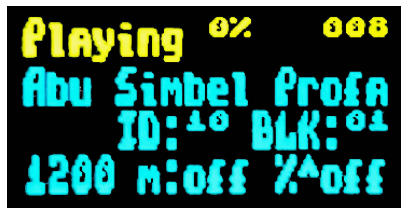
Estando fuera del menú de opciones, el botón de control se utiliza como una palanca de control de cuatro direcciones, que se comporta de dos maneras distintas, según esté la reproducción detenida (Stop) o en pausa (Pause).



Con la reproducción detenida (navegación por archivos y directorios):

- Arriba y abajo permiten desplazarse por la lista actual de ficheros y directorios
- Izquierda (Stop) retrocede un nivel en el árbol de directorios
- Derecha (Play/Pause) accede al contenido de un directorio o, si lo que hay seleccionado es un archivo, intenta reproducirlo

Una vez un archivo está en reproducción, el botón izquierdo (Stop), la detiene, y el botón derecho (Play/Pause) la pone en pausa.



Con la reproducción en pausa (navegación por bloques de cinta):

- Arriba y abajo permiten desplazarse por los bloques del archivo de cinta que ya hayan sido reproducidos (útil para juegos multicarga, para volver a cargar un bloque de un nivel, por ejemplo)
- Izquierda (Stop) cancela la reproducción y entra de nuevo en el modo de navegación de archivos y directorios
- Derecha (Play/Pause) reanuda la reproducción en el bloque seleccionado
- Presionar directamente el botón de control permite activar o desactivar el modo turbo para MSX



Para obtener información mucho más detallada, se puede consultar el manual oficial del firmware Maxduino, disponible en el [repositorio oficial](#).

Creación de ficheros TZX o TSX desde otros formatos

Existen algunos formatos de cinta (Commodore, Computers Lynx, etc.) que, por el momento, no están directamente soportados por Maxduino. No obstante, existen algunos programas que pueden permitir, con mayor o menor éxito, transformar ficheros de cinta desde [formato de audio](#) a [TSX](#) o [TZX](#), y así poder utilizarlos con Miniduino.

MakeTSX

Para usar [MakeTSX](#) de NataliaPC y crear un fichero [TSX](#) con audio embebido, se ha de usar un comando como el siguiente:

```
...MakeTSX -b15 -wav fichero_audio.wav -tsx fichero_nuevo.tsx
```

RetroConverter

Para utilizar [RetroConverter](#) de Jorge Fuertes, para crear un fichero [TZX](#) con audio embebido, usar un comando como:

```
...retroconv fichero_audio.wav fichero_nuevo.tzx
```


Actualización de firmware Maxduino

El firmware Maxduino es actualizado y mejorado periódicamente. Se puede hacer un seguimiento de los cambios y mejoras en el [foro Va de Retro](#) o bien en la [página del proyecto en GitHub](#). Para poder aprovechar estas mejoras se ha de actualizar la memoria flash del Miniduino con la versión de firmware correspondiente.

Existen dos modelos de Miniduino, uno basado en el microcontrolador STM32, y otro basado en ATmega328P.

Modelo STM32

Preparación del entorno

Para poder instalar el firmware, se ha de hacer desde un ordenador (Windows, Mac, Linux) con el entorno [Arduino IDE](#).

Una vez instalado, se ha de añadir la biblioteca de software SDFat (1.1.4) eligiendo la opción de menú Programa → incluir librería → administrar bibliotecas

También se ha de añadir el soporte para el microcontrolador del Miniduino. Esto se hace en dos pasos:

Primero añadiendo soporte para procesadores ARXM Cortex M3 en el menú Herramientas → placa → gestor de tarjetas Instalar "Arduino SAM boards (Cortex-M3)"

A continuación se ha de añadir el soporte para microcontroladores STM32, descargando el fichero disponible en [este enlace](#).

Descomprimir el contenido en la carpeta del usuario actual en

```
...Arduino/hardware/Arduino_STM32
```

En Windows instalar el controlador USB ejecutando con privilegios elevados:

```
...\\drivers\\win\\install_drivers.bat
```

En Linux instalar con privilegios de root las reglas de **udev** necesarias:

```
...tools/linux/install.sh
```

En MacOS, si no apareciese el Miniduino como dispositivo USB en Arduino IDE al conectarlo, puede que sea necesario instalar [libusb](#).

Finalmente, en el caso de Mac o Linux, el fichero `maple_upload` dentro de `Arduino_STM32` tiene que modificarse con un editor de texto. Estas líneas no funcionan bien:

```
if [ $# -eq 5 ]; then
    dfuse_addr="--dfuse-address $5"
else
    dfuse_addr=""
fi
```

Y se tienen que cambiar por

```
dfuse_addr=""
```

Actualización

Una vez preparado el entorno, descargar la versión deseada del proyecto desde el [repositorio oficial en GitHub](#)



El reproductor Miniduino con microcontrolador STM32 sólo está soportado a partir de la versión 1.65

Cargar el fichero del proyecto en Arduino IDE (por ejemplo `MaxDuino_v1.66.ino`).

Verificar en el fichero `userSTM32Config.h` que están comentadas todas las entradas de logo excepto la de Miniduino, y si no, cambiarlas.

```
...
//#define tanque4
//#define tanque1
//#define dostanques
//#define cablemax
//#define sony
#define miniduino
...
```

Conectar el Miniduino al equipo usando el cable USB e identificar el puerto asignado, normalmente con un nombre del tipo "Maple Mini" (por ejemplo: COM5 Maple Mini)

Configurar en el menú "Herramientas" las opciones

```
Placa: Generic STM32F103C Series
Variant: STM32F103C8 (20k RAM, 64k Flash)
Upload Method: STM32duino bootloader
CPU Speed: 72Mhz (Normal)
Optimize: Smallest (default)
Puerto: <Puerto identificado antes>
```

Finalmente, pulsar el botón de carga del firmware y esperar unos segundos mientras se compila el proyecto y se carga en el dispositivo.

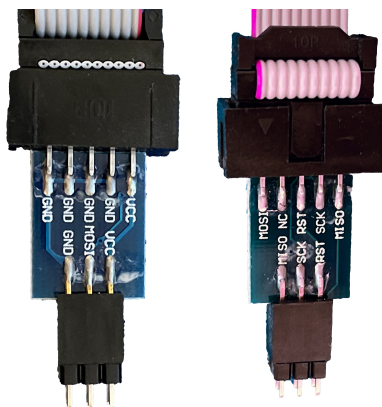
Si todo se ha hecho correctamente se verá cómo el Miniduino se reinicia y en la pantalla aparece la versión correspondiente de firmware.

Modelo ATmega328P

Preparación del entorno

Material necesario:

- Una [llave Allen](#) del tamaño adecuado para poder retirar la tapa
- Programador USBASP



Además, la instalación, se ha de hacer desde un ordenador (Windows, Mac, Linux) con el entorno [Arduino IDE](#).

Una vez instalado Arduino IDE, se ha de añadir la biblioteca de software SDFat (1.1.4) eligiendo la opción de menú Programa → incluir librería → administrar bibliotecas.

Actualización

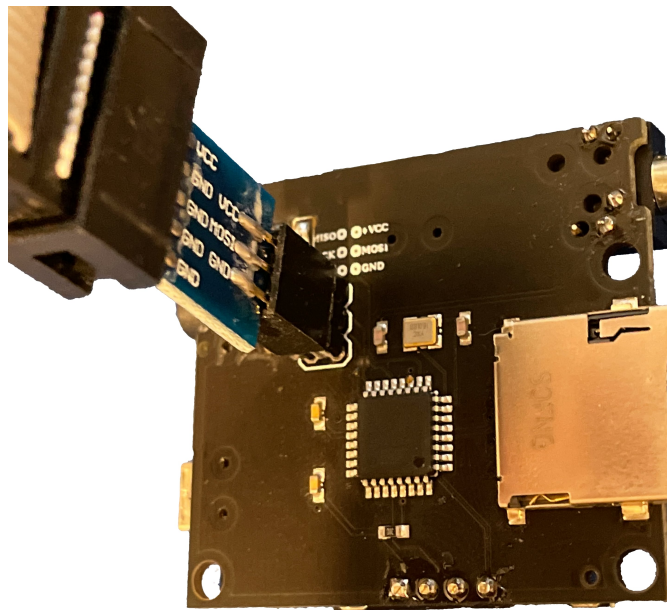
Una vez preparado el entorno, descargar la versión deseada del proyecto desde el [repositorio oficial en GitHub](#)

Cargar el fichero del proyecto en Arduino IDE (por ejemplo [MaxDuino_v1.68.ino](#)).

Verificar en el fichero [userconfig.h](#) que estén comentadas todas las entradas de logo excepto la de Miniduino, y si no, cambiarlas.

```
...
//#define tanque4
//#define tanque1
//#define dostanques
//#define cablemax
//#define sony
#define miniduino
...
```

Conectar el Miniduino al adaptador USBasp, asegurándose de que la serigrafía de los dos conectores se corresponde (ej: VCC con VCC, MOSI con MOSI, GND con GND, etc.), y conectar el adaptador USB al equipo.



Configurar en el menú "Herramientas" las opciones

Placa: Arduino Pro or Pro Mini
Procesador: ATmega328P (5V,16 Mhz)
Programador: "USBasp"

Finalmente, mantener pulsada la tecla mayúsculas mientras se hace clic en el botón de carga del firmware y esperar unos segundos hasta que se compile el proyecto y se cargue en el dispositivo.

Si todo se ha hecho correctamente se verá cómo el Miniduino se reinicia y en la pantalla aparece la versión correspondiente de firmware.