



Instruction Manual

Revision 1.4 21/1/2017

Atari 8-bit SD card Multi-cart

Quick Start

The UnoCart is an Atari cartridge emulator and XEX file launcher. It supports files up to 128k in size. First, copy XEX, CAR and ROM files to a micro SD card and insert it into the cartridge. Then insert the cartridge in the Atari 8-bit with the label/SD card slot facing forwards (Atari XL) or upwards (Atari XE). If the board is uncased, check you've got it the right way round. Power on the Atari. Use the cursor up and down keys to select an item and press Return to start it.

Menu

The menu allows you to navigate the files on the SD card and select a title to use. Use the cursor up, down and [Return] keys to select items. Press [B] to go back to the parent directory. Press [ESC] to search for a file. Press [X] to disable the cartridge and boot normally (e.g. from a SIO device) – this allows you to leave the cartridge plugged in and reduce wear.

Indicator Lights

The red led indicates the cartridge is functioning correctly and is powered. When the menu is active, the green light will indicate SD card access. After a title is selected from the menu, the green light indicates that the cartridge is active.

Reset Button

Press the reset button (next to the SD card slot) to reset the cartridge. This will generally cause the Atari to crash, and the reset button on the Atari can then be pressed to return to the cartridge menu.

ROM Files

Plain ROM files are supported, but since these files are raw cartridge dumps, the UnoCart will decide what type of cartridge they come from based on the file size only. The table below shows what types will be used.

ROM File Size	Cartridge Type
8k	Standard Atari 8k
16k	Standard Atari 16k
32k	Atari XEGS 32k
64k	Atari XEGS 64k
128k	Atari XEGS 128k

Supported Cartridge Types (CAR files)

Description	CAR Type(s)
Standard Atari 8k	1
Standard Atari 16k	2
Atari XEGS 32-128k	12-14
AtariMax 1mbit (128k)	41
Switchable XEGS 32-128k	33-35
Megacart 16-128k	26-29
Bounty Bob	18
Williams 32k, 64k	8, 22

OSS 8k	44
OSS 16k (034M)	3
OSS 16k (043M)	45
OSS 16k (type B)	15
SIC!Cart (128k only)	54
SDX 64k	11
Diamond 64k	10
Express 64k	9
SDX 128k	43
Blizzard 16k	40

Converting to CAR format

Utilities are available to convert ROM/BIN files into the CAR file format. The ROM2CAR utility available on the Ultimate cart website provides an easy drag and drop interface: <https://github.com/robinhedwards/UltimateCart/>

Problem Files

You may come across files with a .BIN extension. These are often just .ROM files and can be renamed. You may also find ROM files where the UnoCart menu gives the error "Unsupported ROM size". This happens when the ROM file is not an exact multiple of 1,024 bytes. In this case, the ROM dump may have already been converted to a CAR file (with a 16 byte header) but has been wrongly named and you can simply rename the file from .ROM to .CAR.

ATR Files (64k XL/XE only)

The cartridge also has simple support for reading and writing to ATR (floppy disk image) files. It supports both SD (128 byte sector) and DD (256 byte sector) images. DD images must be standard ATR files, with 128 bytes for the first 3 sectors and no padding.

When an ATR file is selected from the menu, a modified version of the Atari operating system (OS), based on the Altirra open-source LLE kernel, is installed as a "soft" OS in the 16k RAM under the OS – this allows the Atari's SIO requests to disk drive 1 (D1:) to be re-directed to the SD card in the cartridge.

The main use case for this feature is to allow you to boot a DOS formatted disk (e.g. DOS 2.5) and load and save BASIC programs to the disk. Note that the UnoCart cannot emulate a disk and cartridge at the same time.

In ATR mode, the LEDs on the cartridge indicate SIO reads (green) and writes (red) to the ATR file. Accesses to other disk drives (D2: upwards) are passed through to the SIO port as normal. This allows files to be copied between an ATR mounted on the cartridge (D1:) and other SIO devices (D2:+).

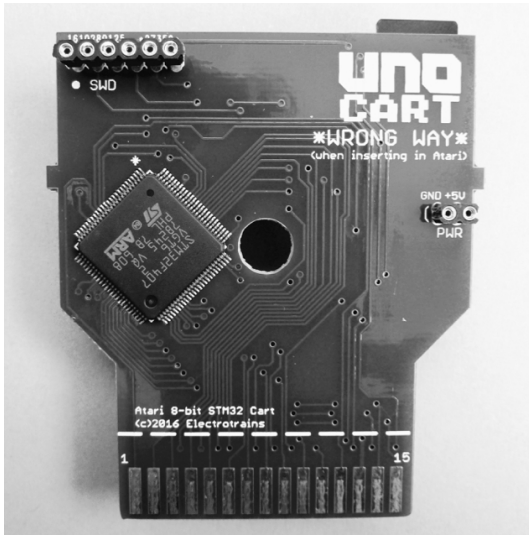
Hold down OPTION when you select the ATR file to disable BASIC. This is necessary for many games. MyPicoDOS ATRs (such as the 16 meg game compilations) will run, but SELECT should be pressed to disable high-speed SIO, and use the operating system SIO routines instead.

Important note – many ATRs will run correctly from the UnoCart. However, any programs that attempt to use the RAM under OS will not work, nor will any software that does not use the standard OS SIO calls.

The OS ROM is stored in the cartridge firmware, but an alternative UNO_OS.ROM file (16k) will be loaded and installed from the root directory of the SD card instead (if such a file is present).

Firmware re-programming (advanced)

The board can be re-programmed using an ST-Link device. The table below shows the pins of the SWD connector on the top left of the board (pin 1 is on the left, marked with a filled circle). Connections in bold are required for programming. The board must also be powered through the +5V pin on the right of the 2-pin power header, **not** through the +3V3 pin on the programming header to avoid back powering the on-board regulator. The latest firmware is available on the project website: <https://github.com/robinhedwards/UnoCart>



Pin	SWD	Function
1	+3V3	Output pin, not for supply
2	SWCLK	SWD Clock
3	GND	GND
4	SWDIO	SWD I/O
5	NRST	SWD Reset
6	SWO	Reserved

Microcontroller (advanced)

The UnoCart will be fitted with either a STM32F407VGT6 (1 meg flash) or a STM32F407VET6 (512k flash) microcontroller. Please refer to your PCB to determine which one is present on your device. If you are developing new firmware for the cartridge, the table below shows the pins connected to the cartridge port, LEDs & micro SD card slot.

STM32 Pin(s)	Cartridge Port	STM32 Pin(s)	Function
PD0...12	A0...A12	PB0, PB1	Green and Red LEDs (active high).
PE8...15	D0...D7	PB5	MicroSD CS
PC0	Phi2	PB13 (SPI2,2)	MicroSD CLK
PC1	/S5	PB14 (SPI2,2)	MicroSD DO
PC2	/S4	PB15 (SPI2,2)	MicroSD DI
PC4	CCTL		
PC5	RW		
PB2	RD5		
PB4	RD4		

Credits

Original idea, hardware and firmware by Robin Edwards (electrotrains at atariage). Altirra LLE OS used with permission from Avery Lee (phaeron at atariage). XEX loader, OS modifications and many helpful discussions thanks to Jonathan Halliday (flashjazzcat at atariage).