



# Mega Speedy 1050 Drive Upgrade

This web page was last updated 12/14/2014 by SJC

## ***Mega Speedy Usage and Programming Information (c) 2014 Matthias Reichl***

### 1. Configuring the drive mode

#### 1.1 Configuring mode using display and encoder

When powering up the 1050 or when pressing the reset button the Mega Speedy enters configuration mode.

The user-configurable default mode is shown on the display and the rotary encoder lights up to indicate that the mode can be changed by turning the knob.

After approximately 3 seconds of idle time the selected mode is automatically activated.

If you press the rotary encoder down the selected mode is activated immediately and will become the new default mode.

You can choose between 10 drive modes, each having 4 separate ROM slots, and the flasher mode where you can boot the flasher directly from the Mega Speedy (just boot the Atari with the drive door open).

The flasher mode is indicated by showing "FL" on the display, for the other modes the first digit indicates the drive mode and the second digit the ROM slot (1-4):

- "E" Mega Speedy
- "S" Super Speedy
- "y" Speedy
- "o" original 1050
- "t" 1050 Turbo
- "H" Happy
- "U" US Doubler
- "A" Super Archiver
- "d" 1050 Duplicator
- "u" SuperMax

*Note: The flasher mode is just another Mega Speedy ROM slot that you are usually not supposed to touch so that you'll always be able to use your drive, even if you screwed up with all the other drive slots.*

In the stock ROM the flasher is also bootable from the 4 Mega Speedy ROM slots.

By using the optional switches you can select between 2-4 default configurations. As long as the encoder lights up you can toggle the switches to select one of the default configurations. Pressing the encoder down will update only the currently selected configuration, the other configurations are not changed.

## 1.2 Configuring mode using the flasher

Just use the "Switch drive mode" option in the flasher and select the drive mode and ROM slot. Note: this option doesn't change the stored default configuration.

## 2. Using the flasher

The flasher allows you to switch the drive mode, program drive ROMs to the flash and test drive ROMs by uploading them to the Mega Speedy RAM.

If you set the Mega Speedy to flasher mode you can boot the flasher directly from your Mega Speedy ROM - just open the drive door and power up your Atari.

You can also load the flasher from the megaspeedy.atr, it is the FLASH.COM file. But note that the Mega Speedy has to be either in "Mega Speedy" or "Flasher" mode.

If you load the flasher from disk you'll be prompted to enter the drive number on startup. You can also later switch to another drive number by using the "restart" option in the flasher.

After selecting the drive the flasher checks if the drive is in Mega Speedy / Flasher mode and uploads code to the drive to enable the flasher functionality. If the drive isn't in Mega Speedy / Flasher mode you'll see an error message and can select another drive.

Note: pressing BREAK aborts the currently selected option and brings you back to the main menu. Use this if you accidentally selected the wrong option.

### 2.1 Programming drive ROMs

Choose the "Program drive ROM" option, enter the desired drive mode and ROM slot number, and then enter the filename of the ROM image. The ROM image is uploaded to the Mega Speedy and then flashed to the selected slot.

The ROM image has to be a plain 4096, 8192, 16384 or 32768 bytes file, without a COM header! Uploading a file with a COM header will render the slot unusable unless you re-flash it. In doubt, first test by running the drive ROM from RAM.

Note: you can't update the currently active ROM slot. Switch to a different ROM slot before, for example by using the "Switch drive mode" option, followed by "Restart".

### 2.2 Running a drive ROM from Mega Speedy RAM

The option "Run drive code from RAM" allows you test drive ROMs by running them from the Mega Speedy RAM. The contents of the flash memory aren't altered by this option.

Choose the mode and enter the filename. After uploading the file to the Mega Speedy RAM the Mega Speedy is automatically configured to use this ROM.

*Note: if you want to continue using the flasher you have to use the "Restart" option so that the drive is reconfigured to support the flasher function (this only applies to uploading Mega Speedy mode ROMs to the RAM). Otherwise just exit the flasher or reboot your Atari.*

### 2.3 Switching drive mode

With this option you can configure the Mega Speedy drive mode. Select the mode and slot number that should be activated.

*Note: use "Restart" if you want to continue using the flasher.*

## **2.4 Programming the special flasher and config ROM slots**

The config and flasher ROM slots work like the Mega Speedy slots, but if you screw up here you'll brick your Mega Speedy. If this happens you have to use an EPROM/flash programmer to reprogram the flash with the megaspeedy-stock.rom file.

The config slot is particularly critical; it is the one that is activated when powering up. If this ROM doesn't work your 1050/Mega Speedy won't function at all.

The flasher slot is a little bit less critical. As long as you have a working Mega Speedy / flasher ROM in one of the 4 Mega Speedy ROM slots, you can still use one of these to reprogram the flasher slot. If none of these work, you won't be able to use the flasher anymore and have to dig out your EPROM/flash programmer. So, be careful!

The stock flasher and config ROM images are included in the megaspeedy.atr and as separate files in the "roms" directory and are named FLASHER.ROM and CONFIG.ROM, respectively.

As a security measure you have to enter the sentence "PROGRAM FLASHER ROM" or "PROGRAM CONFIG ROM" before flashing starts.

*Note: programming the flasher ROM slot won't work if you set the Mega Speedy to "flasher mode" - the flasher forbids programming the active ROM slot.*

In this case you have 2 options:

- - Switch to one of the Mega Speedy modes, and then restart the flasher.
- - Run the flasher ROM code from Mega Speedy RAM, and then restart the flasher.

In both cases the Mega Speedy isn't using the flasher slot anymore so you can reprogram it.

## **2.5 Exiting the flasher**

Always use the "Quit" option of the flasher, as that will soft-reset the Mega Speedy and disable the uploaded flasher support code in the Mega Speedy.

## **3. Programming information**

The Mega Speedy mode is based upon the Super Speedy mode and adds several enhancements. This section only lists the items that were added or enhanced. For general Speedy programming information like RAM and ROM memory locations please read the original Speedy docs. At the time of writing this README they were available here:

<https://atariwiki.org/wiki/Wiki.jsp?page=Das%20SPEEDY-System>

### **3.1 1050 / 6502 memory layout**

\$0000-\$0FFF zero page, stack, FDC, RIOT

\$1000-\$1FFF unused

\$2000-\$3FFF same as \$0000-\$1FFF

\$4000-\$5FFF track display and density LEDs  
\$6000-\$6FFF \$A000 RAM bank select  
\$7000-\$7FFF Mega Speedy config registers  
\$8000-\$9FFF fixed 8k RAM bank  
\$A000-\$BFFF switchable 8k RAM bank  
\$C000-\$DFFF switchable 8k ROM bank  
\$E000-\$FFFF fixed 8k ROM bank with Speedy drive ROM

### **3.2 512k RAM chip memory layout**

\$00000-\$73FFF free  
\$74000-\$75DFF reserved  
\$75E00-\$75EFF 6810 replacement RAM  
\$75F00-\$75FFF second 6810 replacement RAM for US Doubler  
\$76000-\$77FFF fixed 8k RAM bank for Happy, Speedy etc  
\$78000-\$7FFFF reserved for running drive code from RAM  
The flasher uses \$30000-\$3FFFF as temporary storage when programming the flash ROM.

### **3.3 512k Flash ROM memory layout**

\$00000-\$0FFFF Speedy ROM slots (4x16k)  
\$10000-\$1FFFF Super Speedy ROM slots (4x16k)  
\$20000-\$27FFF 1050 Turbo ROM slots (4x8k)  
\$28000-\$2FFFF Happy ROM slots (4x8k)  
\$30000-\$33FFF original 1050 ROM slots (4x4k)  
\$34000-\$37FFF US Doubler ROM slots (4x4k)  
\$38000-\$3BFFF SuperMax ROM slots (4x4k)  
\$3C000-\$3FFFF Super Archiver ROM slots (4x4k)  
\$40000-\$5FFFF Mega Speedy ROM slots (4x32k)  
\$60000-\$67FFF 1050 Duplicator ROM slots (4x8k)  
\$68000-\$6FFFF unused / reserved (32k)  
\$70000-\$77FFF Flasher ROM slot (32k)  
\$78000-\$7FFFF Config ROM slot (32k)

### **3.4 Configuration Registers**

(r) Indicates a read-only register

(w) Indicates a write-only register

(r/w) Indicates a read/write register

Unused bits should be set to 0 unless otherwise noted.

\$4000 (w) Display Lo / second digit

\$4001 (w) Display Hi / first digit

Bit 7: unused

Bit 6-0: Segments G-A (0=off, 1=on)

\$4002 (w) Density LEDS

Bit 7-3: unused

Bit 2-0: Density LEDs (DD, ED , SD ; 0=off, 1=on)

\$4003 (w) Buzzer

Bit 7-1 unused

Bit 0: Buzzer output, toggle to produce a tone

\$6000 (w) RAMBANK - 8k RAM bank at \$A000

Bit 7-6: unused

Bit 5-0: bank number (0-63)

*Note: RAM banks 58-63 are reserved see RAM memory layout.*

### **Super Speedy ignored bits 7-5 and supported 24 banks**

\$7000 (w) ROMBANK - 8k ROM bank at \$C000

Bit 7: 0=512k mode, 1=32k mode

Bit 6: unused

Bit 5-0: bank number (0-63, 0-3 in 32k mode)

In 512k mode (bit 7=0) the bits 5-0 are used to select one of the 64 8k flash ROM banks. This allows access to the whole flash memory.

In 32k mode bits 5-2 are ignored. The selected bank number is constructed using bits 6-3 from ROM\_BASE and bits 1-0 from this register. Bit 7 of ROM\_BASE is respected; if it is set to 1 a read-only RAM bank is selected. This allows switching between the 4 8k blocks of the Mega Speedy ROM without having to care about the absolute memory position and whether it's located in flash or RAM.

\$7000 (r) ROMBANK - 8k ROM bank at \$C000

Bit 7: 0=512k mode, 1=32k mode

Bit 6: unused

Bit 5-0: bank number (0-63, 0-3 in 32k mode)

In 512k mode bits 5-0 are read back as written before.

In 32k mode bits 5-2 come from CFG\_ROM\_BASE bits 6-3, bits 1-0 read back as written before.

Default value on power up is bit 7=1, bit 1-0=2 (the 3rd of the 4 8k blocks in the Mega Speedy ROM).

\$7010 (rw) ROM\_BASE - drive ROM base address in 4k granularity

Bit 7: 0=ROM mode, 1=RAM mode

Bit 6-0: 4k bank (0-127)

Depending on the MODE bits 0-2 are ignored. Base address of the drive

ROM must always be a multiple of the drive ROM size.

In RAM mode the data is read from RAM instead of flash ROM. Writing to the RAM is not possible via the ROM area.

\$7011 (rw) MODE - select drive mode

Bit 7-4: unused

Bit 3-0: drive mode (0-15)

Drive modes:

0: Config (same as Mega Speedy)

1: Speedy

2: Super Speedy

3: Mega Speedy

4: original 1050

5: 1050 Turbo

6: Happy

7: US Doubler

8: Super Archiver

9: SuperMax

10: 1050 Duplicator

11-15: reserved, do not use

\$7018 (r) Encoder and switch inputs

Bit 7-5: unused

Bit 4: switch 2

Bit 3: switch 1

Bit 2: encoder button/pushed

Bit 1: encoder input B

Bit 0: encoder input A

All inputs are active-low (0=closed, 1=open).

\$7019 (w) Encoder LED

Bit 7-1: unused

Bit 0: Encoder LED (0=off, 1=on)

\$7020 (w) I2C data lo, clock lo

\$7021 (w) I2C data lo, clock hi

\$7022 (w) I2C data hi, clock lo

\$7023 (w) I2C data hi, clock hi

Bit 7-0: unused

Writing to these registers sets the I2C SDA and SCL lines.

\$7020 (r) I2C data input

Bit 7: I2C SDA input

Bit 6-0: unused

Reads back the status of the I2C SDA line

## **4. Programming hints and examples**

### **4.1 Bank Switching within the 32k Mega Speedy ROM**

The original Speedy and Super Speedy ROMs are 16k and are mapped to \$C000-\$FFFF.

The 32k Mega Speedy ROMs consists of 4 8k blocks. By default the 3<sup>rd</sup> block is mapped to \$C000-\$DFFF, this block can be bank switched. The 4th block is always mapped to \$E000-\$FFFF.

You can use an original Speedy ROM in one of the Mega Speedy slots by pretending some arbitrary 16k of data at the beginning – they won't be used unless you manually bank switch them in. An easy way to achieve this is to just store the original ROM twice in a file. Copy the original ROM to a new file, and then append the original ROM to the newly created file and you are done.

To switch the 8k ROM bank at \$C000 just write \$80-\$82 to ROMBANK (\$7000) to select one of the 3 banks. When you are finished always set ROMBANK back to the default \$82.

### **4.2 Booting code from the Speedy**

When you boot with the drive door open the Speedy sends the contents from \$C000-\$C07F when sector 1 is requested. Serving other sectors from the ROM is not

supported.

So your initial boot code has to fit within one sector. This code then must load additional data from the Speedy ROM via special Speedy commands.

If one sector is not large enough for your boot code, do 3-stage loading. The code in sector 1 loads the second-stage loader (which can be as large as you want) and then runs it. Use the second-stage loader to load additional data blocks and finally run the loaded code.

An easy way to load a (large) data block from ROM with a single SIO call is by having code present in the Speedy ROM (eg at \$C100) and executing it via the \$4D ("jump with complete") command.

Setup SIO with command=\$4D and DAUX=address of the code in the Speedy (eg \$C100). Set DBUF and DBYT according to the destination address and the length of the data block. Then do a normal SIO read call.

The code in the Speedy (at \$C100 in this example) then transmits a data block from the ROM (of course the length of the block should match the length you used in the SIO call).

That's all! By having multiple code-blocks in the Speedy ROM and issuing various SIO calls you can easily load multiple blocks of arbitrary size to different memory locations.

If you bank-switch the Speedy ROM block at \$C000 be sure to set it back to the default ROM block, otherwise a subsequent boot attempt might read the wrong data for the boot sector.

You can find an example how this is implemented in the "speedy-flash.src" file, that's the code to create the bootable flasher ROM image.

1 ONLINE

---

[About ILS](#) - [Contact us](#) - [Report Web Issues](#)

---