

IMPORTANT: Upgrading your firmware is potentially dangerous. If something goes wrong during the update, it might leave your system unusable. There is usually not much reason to upgrade your firmware unless

- The update is flagged as 'required'
- You need a feature of the new firmware that was not in your previous firmware or did not work correctly.
- You have other issues with your machine that are due to the firmware.

If your system is working, think twice before you do an update.

U-Boot Firmware Update for AmigaOne/Teron boards

1. Introduction

This program updates the firmware in your computer. The firmware (also called the 'BIOS' for very historical reasons) is a piece of software stored in a fixed storage in your machine (usually a “flashrom”, i.e. an erasable Read-Only Memory) that receives control as soon as you switch on your computer. The firmware usually initializes the machine, sets up a suitable environment and then boots an operating system.

U-Boot is an open-source firmware that is licensed under the GNU General Public License. A copy of the GPL is included in this archive. U-Boot is under active development and can be found at <http://u-boot.sourceforge.net/>.

2. Reasons to update

As was mentioned above, upgrading your firmware is potentially dangerous, and therefore only recommended under certain circumstances. If the update is not required to run a new revision of the operating system, you should think twice before you change a working system. It is only recommended to upgrade if there is a new feature/hardware support or a general problem with your machine that would be cured by the update.

3. How to update

This update comes in one of two forms. The 'Floppy' edition contains a standalone U-Boot bootable image that can be directly written to a floppy disk and booted, or loaded via a network boot. The 'ISO' edition contains a bootable ISO image that can be written to a blank CD-R.

3.1 Floppy Edition update

The file “updater.image” needs to be written to a floppy as raw data. Under GNU/Linux, the easiest way to do this is to use the 'dd' command. Insert a blank floppy into your floppy drive, open a shell then type the following at the shell prompt:

```
dd if=updater.image of=/dev/floppy
```

The string '/dev/floppy' denotes the special device node that represents your floppy disk. Most Linux distributions use a name like /dev/floppy or /dev/fd0. You should consult the manual of your Linux distribution to find the correct name.

On AmigaOS, a number of programs exist to write raw data to a floppy. One such example is the TrackDOS program, or the FLAT file handler. Consult the documentation of these programs for details on how to write the image.

If you want to use Windows, a program `rawrite.exe` can be used. This is frequently used to write boot disks, for example for Linux installations.

Once the floppy is prepared, reboot your AmigaOne/Teron and escape from the menu (by pressing ESC twice and then selecting “Go to U-Boot shell”). On the U-Boot command line enter the following:

```
fdcbboot ; bootm
```

After some time, there should be a message about “AmigaOne Firmware Updater” or similar on screen. Continue to read in section 3.4 below.

3.2 Floppy Edition over a network

The 'updater.image' program can be booted via network. You must have a tftpboot server accessible from your AmigaOne/Teron machine (the setup of such a server is beyond the scope of this README; also, there are other methods to boot via BOOTP or DHCP which are also not covered here).

A bit of setup is required on the AmigaOne/Teron side. First of all, verify that the variable `ethaddr` is set up to your machine's internal hardware ethernet address (U-Boot should have done that automatically already). Secondly make sure that you have a correct server IP address set up in the `serverip` variable. Third, your own machine's IP address should be set up in the `ipaddr` variable.

To boot the image, from the U-Boot prompt enter

```
setenv autostart yes
tftpboot 500000 updater.image
```

After some rows of '#' characters, a message about “AmigaOne Firmware Updater” or similar should appear on screen. Continue to read in section 3.4 below.

3.3 ISO/CD Edition

You need to have access to a machine with a CD-Burner. The ISO archive contains a plain bootable ISO-CDROM-Image that can be written to a CD-ROM directly (i.e. don't select “bootable CD” or similar when creating the CD – the ISO already contains all info that makes it bootable). For this purpose the burner program should have a function to write a raw image.

If you are using GNU/Linux, you can use the `xcdroast` program. On AmigaOS, a program like `FryingPan` or `MakeCD` can be used. Under Windows, any program that can write raw images can be used.

Once you have transferred the image to a CD-ROM, reboot your AmigaOne and enter the U-Boot prompt as described in 3.1.

The next step depends on the firmware version you are *currently* using. If your current firmware is from a date prior to February 2005, boot the CD by entering

```
diskboot 500000 <ide_device>:0
```

Replace `<ide_device>` with the number that corresponds to your CD-ROM drive (the the table 'ide device mappings' below). Your machine might or might not be set up for automatic booting. If the machine returns to the prompt after a second or two, you need to explicitly start the image by entering

```
bootm
```

from the prompt. A message about “AmigaOne Firmware Updater” should appear on the screen shortly; continue to read in section 3.4.

If your current firmware is from after February 2005, the command line to boot the image looks slightly different:

```
diskboot 500000 <ide_device>:0 <ide_controller>
```

Here, `<ide_device>` is again the number that corresponds to your CD-ROM drive (as described in the table 'ide device mappings' below). The `<ide_controller>` number specifies which controller is used for booting. The mapping is outlined in the table below ('ide controller mappings'). Also note that if your machine is not set up for automatic booting, you need to give the 'bootm' command to start the boot process.

<i>CD-Rom location</i>	<i>Device number</i>
Primary Master	0
Primary Slave	1
Secondary Master	2
Secondary Slave	3

Table 1 IDE device mappings

<i>Controller</i>	<i>ID-Number</i>
On-Board VIA controller	0
PCI Silicon Images Parallel ATA controller	1
PCI Silicon Images Serial SATA controller	2

Table 2 IDE controller mappings

3.4 The updater

Once the image has booted, you will be presented with a warning message. On the first page press return to continue. On the second page, you are required to enter the letters 'o' and 'k'. **If you do so you will accept all responsibilities if the process fails!** Read the warnings above before you press 'o' and 'k'. Immediately after you pressed 'k' the re-flashing process starts. This usually only takes about 15 seconds.

You will be notified if the updater finds a problem. After the update is finished, the updater will output a few lines of information and ask you to reset the machine (it will usually reset itself after 5 seconds). You should verify the firmware date is correct after the boot by entering the 'version' command from the U-Boot prompt.

4. What is new in this release

The following is a short run-down of the changes in this release of the firmware:

1. Numerous fixes in the x86 emulator (thanks to the folks at Scitech).
2. Completely reworked IDE code (thanks to Andrea Valinotto). Boot devices can now be selected in the boot setup menu according to your needs. Note that if you have no devices connected to the internal VIA controller you might want to set the variable `ide_maxbus` to 0 to speed up booting.
3. New 'video' options page in the menu. Right now, the only choice is to boot from either a PCI or AGP video card. If "PCI" is selected, any AGP card (or the internal Radeon® chip in a MicroA1/TeronMini) is ignored (albeit initialized) and the primary display/firmware menu will open on a PCI graphics card (if inserted). Otherwise, an AGP card (or internal graphics chip) is used. If only one card is inserted, this card is used in any case.
4. (Preliminary) support for the Catweasel Mark IV controller by Individual Computers. For more info on the Catweasel, visit <http://www.ami.ga/>.
The Catweasel Amiga keyboard support can be enabled by setting the variable "stdin" to "amikbd", i.e. from the U-Boot prompt enter (you need a PS/2 or USB keyboard connected

before doing this):
setenv stdin amikbd
saveenv

5. Numerous internal bug fixes

5. Support

If you have questions concerning U-Boot, write a mail to u-boot@hyperion-entertainment.biz.