Programming the Operating System on the S3730

Applies to firmware versions 3.0.0 and above.

03/31/09

Table of Contents

1. Introduction

This document describes the procedures for updating the contents of the S3730 built-in Flash memory.

2. What you need

1. RS-232 cable
2. Terminal Emulator e.g. Windows Hyperterminal

If you are transferring files via Ethernet, you will also need:

3. TFTP server
4. Ethernet switch or cross-over cable

Most Linux distributions include a TFTP server (e.g. atftp). On Windows, several FTP servers are available, for example http://www.solarwinds.net/products/freetools/.

The S3730 built-in flash is divided into several partitions: bootloader, configuration, Linux kernel, root and user files-systems. Each of these partitions may be updated

independently of the other. Updating a single partition requires to first transfer the corresponding partition data, a file, to RAM on the S3730. Once in RAM, the partition data is written to Flash.

Readers that are already familiar with the U-Boot bootloader may proceed directly to section 3, and skim the remainder of this document.

3. Important Addresses and Commands

Address	Description
0x00100000	RAM location where the transferred file is stored
0x40000000	Flash location of the bootloader code
0x40040000	Flash location of the bootloader environment
0x40060000	Flash location of the configuration partition (minix)
0x40080000	Flash location of the Linux Operating System (cramfs)
0x40380000	Flash location of the /home partition (jffs2)
0x42380000	Flash location of the /mnt/local partition (jffs2)

The following bootloader commands are relevant to the flash update process.

```
cp      - memory copy
erase   - erase FLASH memory
iminfo  - print header information for application image
loadb   - load binary file over serial line (kermit mode)
loady   - load binary file over serial line (ymodem mode)
protect - enable or disable FLASH write protection
tftpboot- boot image via network using TFTP protocol
setenv  - set environment variables
```

4. Connecting to the Console Port

The console port is located on the 10-pin header. Using the ribbon adapter cable, connect to a terminal (e.g. Windows Hyperterminal) using the following settings: 115200 bps, 8 data bits, 1 stop bits, no flow control

After powering up the S3730, press any key to obtain the bootloader's prompt.

```
U-Boot 1.2.0-gf5fd6232-dirty (Mar 18 2008 - 18:34:13)

CPU:   MPC885ZPnn at 133 MHz [15.0...133.0 MHz]
       8 kB I-Cache 8 kB D-Cache FEC present
Board: SETRIX S3730
       Watchdog enabled
I2C:   ready
DRAM:  64 MB
FLASH: 128 MB
In:    serial
Out:   serial
Err:   serial
FEC ETHERNET
=>
```

5. Transfering files to the S3730

Partition data that is transferred to the S3730 is stored at memory address 0x100000. It is from this memory location that the data will be copied to Flash.

There are two methods for transferred partition data to the S3730: either via the console serial port, or via Ethernet. File transfer via Ethernet allows for a faster file transfer, but it slightly more complex to set-up.

5.1. Via the Console Serial Port

The bootloader includes support for the Kermit file transfer protocol. Support for this protocol is also provided in many terminal emulation tools, such as Windows Hyperterminal.

1. Start the Kernel protocol on the S3730

=> loadb
Ready for binary (kermit) download to 0x00100000 at 115200 bps...

2. Initiate the file transfer from your terminal Emulator

5.2. Via Ethernet

1. Install and configure a TFTP server on your LAN
2. Copy the partition data files to the TFTP server directory.
3. Configure the IP addresses in the bootloader

```
=> setenv ethaddr 00:06:5B:4B:93:79
=> setenv ipaddr 192.168.15.99      <- any un-used IP address
=> setenv serverip 192.168.15.4     <- IP address of the TFTP server
=> setenv gatewayip 192.168.15.1    <- IP address of the Gateway
```

4. Specify the name of the file to transfer

=> *setenv bootfile <filename on TFTP server>*

5. Transfer the file

=> *tftpboot*

6. Writing the Firmware to Flash

The latest firmware version is available for download from:

https://s3730.net/downloads/FIRMWARE/

Filename	Description
linux-s3730.bin	Linux operating system for the standalone S3730 board
linux-fast.bin	Linux operating system for FAST II.

Once copied to RAM, the firmware if written to Flash using the following commands:

```
=> erase 40080000 +$filesize
=> cp.b $fileaddr 40080000 $filesize
```

To complete the process, reset the board:

```
=> reset

[ wait until you see "ready" and press <return> two times ]

s3730 console login: root
Password: setrix
#
```

7. Installing the bootloader (for experts)

The bootloader may also be updated in a manner similar to firmware. The bootloader file should first be transferred to the S3730 using either Kermit or TFTP (see section 5 ”Transfering files to the S3730” above). Once the bootloader file has been uploaded to RAM, it is copied to Flash using the following commands:

```
=> protect off 40000000 +$filesize
=> erase 40000000 +$filesize
=> cp.b $fileaddr 40000000 $filesize
=> protect on 40000000 +$filesize
=> reset
```

Alternatively, the bootloader may also be updated from within Linux, using the following commands:

```
# mknod /dev/mtdblock0 b 31 0
# dd if=u-boot.bin of=/dev/mtdblock0 bs=64k
# reboot
```

Important: power should not be removed from the S3730 while the Flash during the bootloader update process.

8. Running bootloader scripts (for experts)

The bootloader includes the ability to run a series of commands packaged into a script. Such scripts are first transferred to the S3730 via `tftpboot` or `loadb` and then executed with the `autoscr` command.

```
=> setenv bootfile <name of script in TFTP server>
=> tftpboot && autoscr
```

A number of prebuilt scripts available for download and are described below.

Script name	Description
prog_uboot.scr	Program the bootloader onto flash. Please copy the bootloader binary file to your TFTP server as file name `u-boot.bin` before running this script.
prog_linux.scr	Program the linux operating system onto flash. Please copy the linux binary file to your TFTP server as file name `root.cramfs` before running this script.

9. Booting into single-user mode

The following bootloader commands may be used to boot the system and bypass all boot scripts.

```
=> setenv bootargs ${bootargs} init=/bin/sh
=> boot
```

10. Ethernet Configuration

When running Linux, the S3730 will obtain an IP address via DHCP. If no DHCP server is available, the IP address can be configured manually:

```
# ifconfig eth0 192.168.15.9
```

11. Add-on Packages

A number of add-on packages are available for download from https://s3730.net/downloads/IPK/.These packages provide additional functionality such as device drivers and open-source utilities and applications.

11.1. Package Management

All add-ons are provided in the ipk format. Package management (installation, removal) is provided by the built-in *ipkg* command. Since firmware version 2.2.4, *ipkg* is built into the firmware, but requires a configure file /home/etc/ipkg.conf. This configuration file can be created by running the following commands:

```
# mkdir /home/etc
# echo dest home /home > /home/etc/ipkg.conf
```

To install a package, transfer the *ipk* file to the S3730 and run the following command:

```
# ipkg install <package>
```