Das U-Boot on the Efika MX53

Chris Jenkins - Genesi USA, Inc

Terminology

- `Bootloader`: Small program living in fixed location of non-volatile memory that initializes hardware and starts the operating system.
- `Boot sector`: The partition of a non-volatile memory device containing the bootloader. Often marked as such in a master boot record (MBR)
- `Das U-Boot`: Short for “Universal Bootloader”; free and open-source primary bootloader often used in embedded systems, and in use on the Efika MX53 and related hardware designs.
- We also assume you are somewhat familiar with terms associated with serial device connections (e.g. baudrate, software flow control, etc)

Prerequisites

- Linux / OS X workstations
  - You will need to install the program `screen` which will allow you to open a serial console
  - If you are using a serial to USB converter, the serial device will be named[1] something like `/dev/ttyUSB0` on an Ubuntu system and like `/dev/tty.usbserial-ftDXJ1BC` on OS X
- Windows workstations
  - We recommend installing the program PuTTY
  - The device will be named[1] something like `COM0`, and can be looked up in the device manager menu.

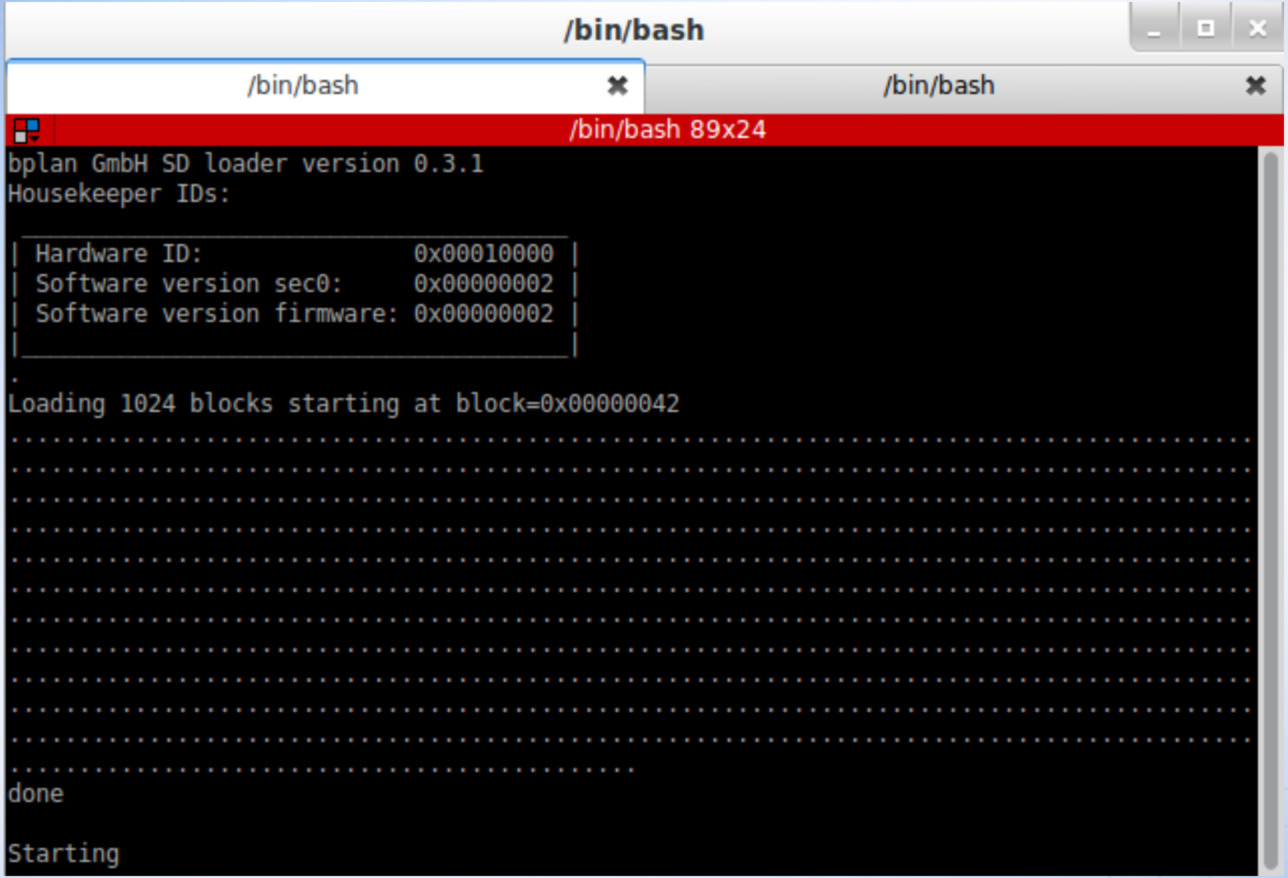
You will need a naked board or one with removable housing so you can solder the serial RX / TX; there is no other way to access the U-Boot console. You may also wish to use a serial to USB converter.

The baud rate is 115200, and on my workstation (Linux Mint 16) this is the only option that needs to be specified to access the U-Boot console.

[1] Names subject to some variation.

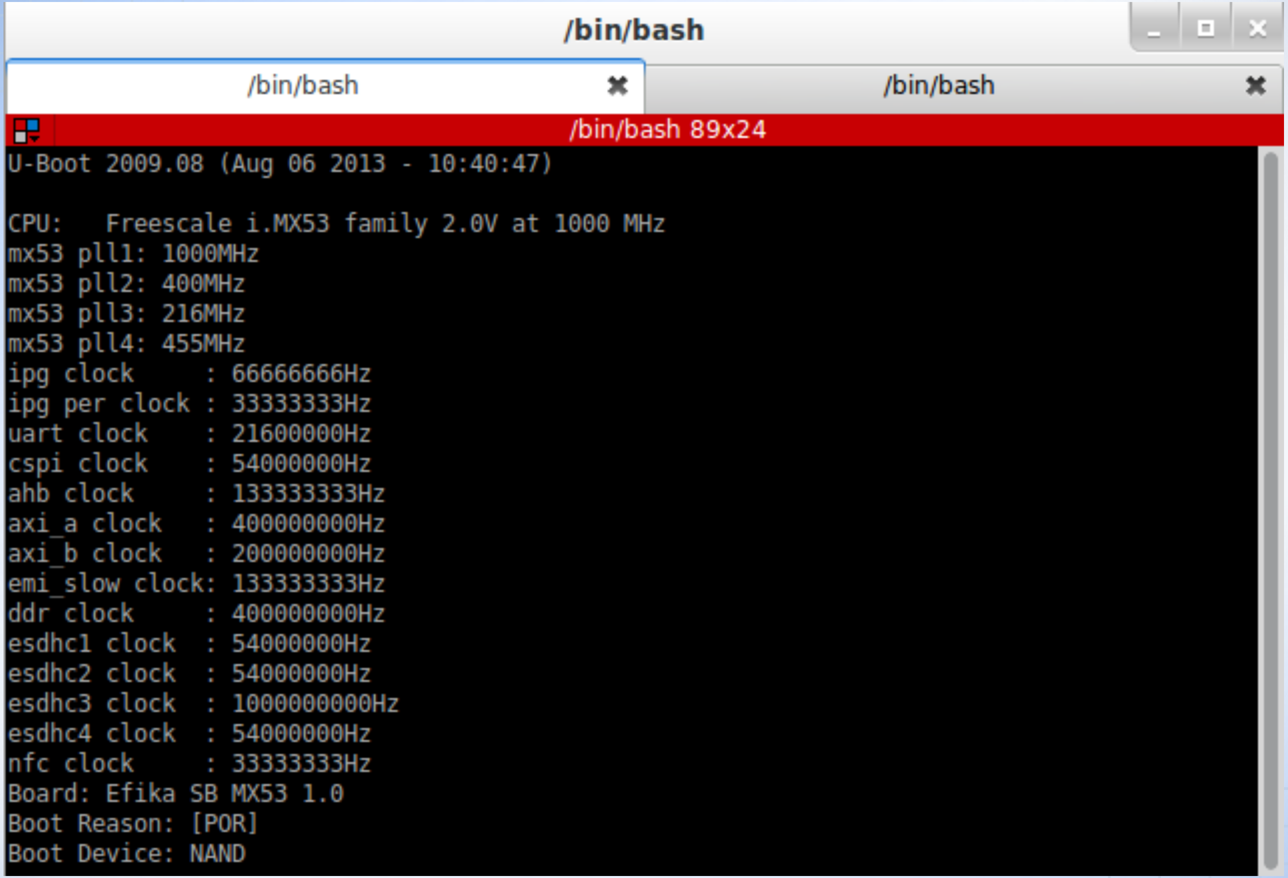
Entering the U-Boot console

- With the Efika MX53 powered off, connect device serial to your workstation. Open screen / PuTTY session for the device
  - e.g. `screen /dev/ttyUSB0 115200`
- Power on Efika MX53. After a few moments, you should see the bplan STM bootloader splash, followed quickly by the U-Boot splash
- Quickly press <Enter> when you see the prompt “Hit any key to stop autoboot”. You only have 3 seconds! If you did this correctly you will be at the U-Boot prompt: `Slimbook>`
- Enter the command `help` to verify



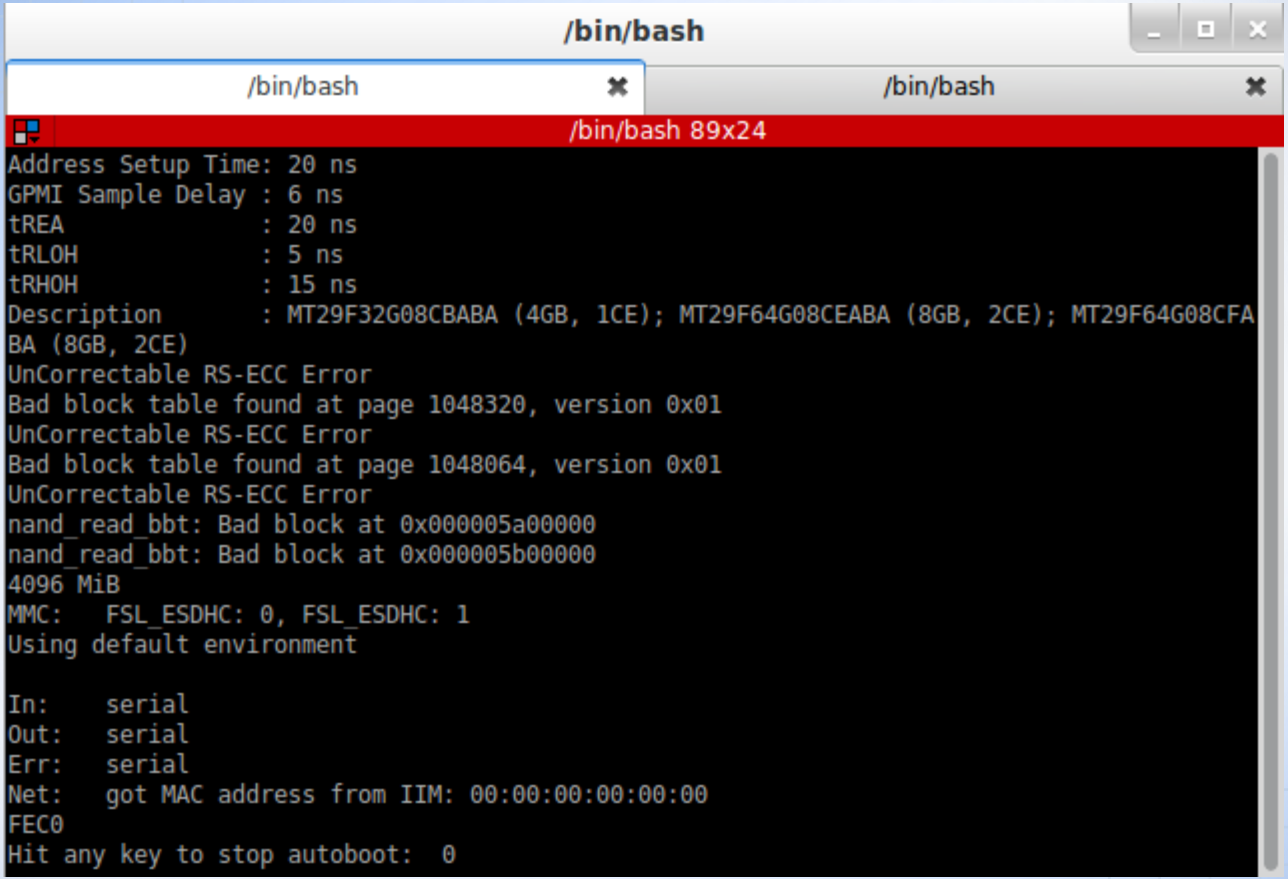
```
bplan GmbH SD loader version 0.3.1
Housekeeper IDs:
 _______________________________________
| Hardware ID:               0x00010000 |
| Software version sec0:     0x00000002 |
| Software version firmware: 0x00000002 |
|_______________________________________|
.
Loading 1024 blocks starting at block=0x00000042
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.............................................
done

Starting
```



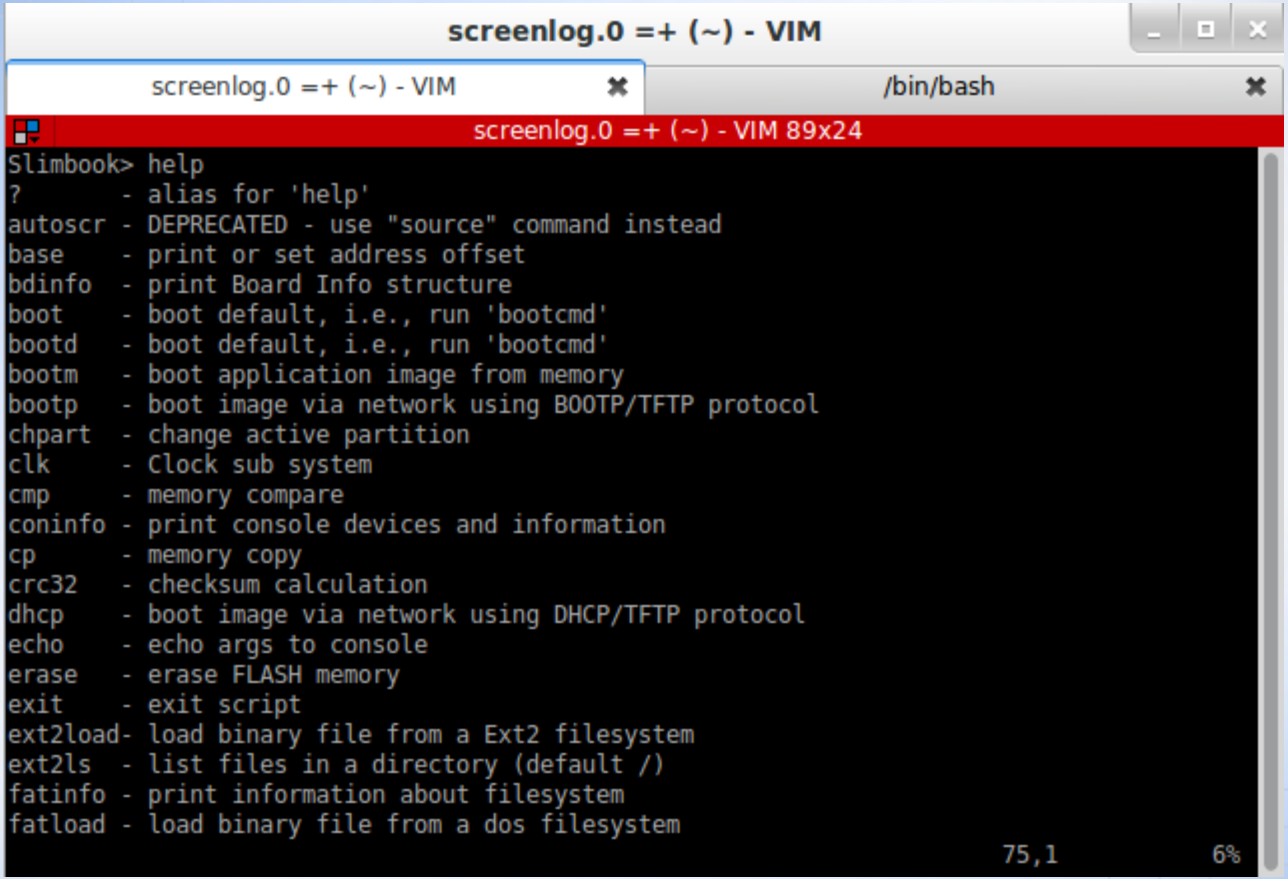
```
U-Boot 2009.08 (Aug 06 2013 - 10:40:47)

CPU:   Freescale i.MX53 family 2.0V at 1000 MHz
mx53 pll1: 1000MHz
mx53 pll2: 400MHz
mx53 pll3: 216MHz
mx53 pll4: 455MHz
ipg clock     : 66666666Hz
ipg per clock : 33333333Hz
uart clock    : 21600000Hz
cspi clock    : 54000000Hz
ahb clock     : 133333333Hz
axi_a clock   : 400000000Hz
axi_b clock   : 200000000Hz
emi_slow clock: 133333333Hz
ddr clock     : 400000000Hz
esdhc1 clock  : 54000000Hz
esdhc2 clock  : 54000000Hz
esdhc3 clock  : 1000000000Hz
esdhc4 clock  : 54000000Hz
nfc clock     : 33333333Hz
Board: Efika SB MX53 1.0
Boot Reason: [POR]
Boot Device: NAND
```



```
Address Setup Time: 20 ns
GPMI Sample Delay : 6 ns
tREA              : 20 ns
tRLOH             : 5 ns
tRHOH             : 15 ns
Description       : MT29F32G08CBABA (4GB, 1CE); MT29F64G08CEABA (8GB, 2CE); MT29F64G08CFA
BA (8GB, 2CE)
UnCorrectable RS-ECC Error
Bad block table found at page 1048320, version 0x01
UnCorrectable RS-ECC Error
Bad block table found at page 1048064, version 0x01
UnCorrectable RS-ECC Error
nand_read_bbt: Bad block at 0x000005a00000
nand_read_bbt: Bad block at 0x000005b00000
4096 MiB
MMC:   FSL_ESDHC: 0, FSL_ESDHC: 1
Using default environment

In:    serial
Out:   serial
Err:   serial
Net:   got MAC address from IIM: 00:00:00:00:00:00
FEC0
Hit any key to stop autoboot:  0
```



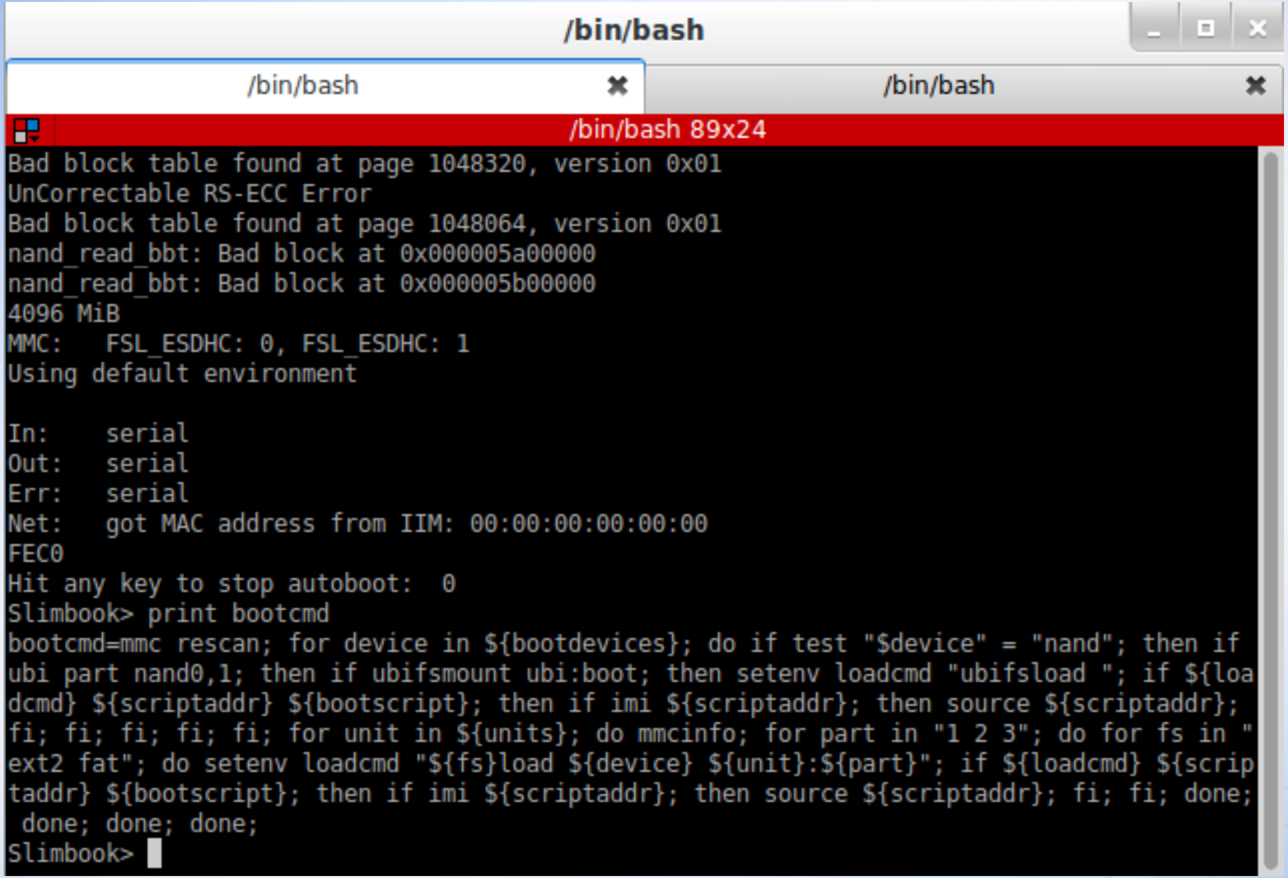
```
Slimbook> help
?       - alias for 'help'
autoscr - DEPRECATED - use "source" command instead
base    - print or set address offset
bdinfo  - print Board Info structure
boot    - boot default, i.e., run 'bootcmd'
bootd   - boot default, i.e., run 'bootcmd'
bootm   - boot application image from memory
bootp   - boot image via network using BOOTP/TFTP protocol
chpart  - change active partition
clk     - Clock sub system
cmp     - memory compare
coninfo - print console devices and information
cp      - memory copy
crc32   - checksum calculation
dhcp    - boot image via network using DHCP/TFTP protocol
echo    - echo args to console
erase   - erase FLASH memory
exit    - exit script
ext2load- load binary file from a Ext2 filesystem
ext2ls  - list files in a directory (default /)
fatinfo - print information about filesystem
fatload - load binary file from a dos filesystem
                                                                   75,1          6%
```

Efika MX53 Boot command

At the U-Boot console, you can type `print bootcmd` to see the boot command, `run bootcmd` to boot the Linux kernel

- Rescan MMC devices
- For every device in the list of recognized boot devices (mmc and nand, respectively)
  - If the device is NAND, has the right partitions, and the boot partition is successfully mounted, set the `loadcmd` variable to `ubifsload`[1]
    - Load the bootscript living on the boot partition into a preset location in RAM. If it is a valid script, run it.
  - If the device is MMC, check every partition (2 1 3 resp.) and every supported file system (fat ext2 resp.), set the `loadcmd` environment variable to the file-system specific load command on the selected partition.
    - Load the bootscript living on this partition into a preset location in RAM. If the file exists, is loaded successfully, and is a valid script, run it

1: UBIFS is the file system on the Efika MX53 NAND



```
Bad block table found at page 1048320, version 0x01
UnCorrectable RS-ECC Error
Bad block table found at page 1048064, version 0x01
nand_read_bbt: Bad block at 0x000005a00000
nand_read_bbt: Bad block at 0x000005b00000
4096 MiB
MMC:   FSL_ESDHC: 0, FSL_ESDHC: 1
Using default environment

In:    serial
Out:   serial
Err:   serial
Net:   got MAC address from IIM: 00:00:00:00:00:00
FEC0
Hit any key to stop autoboot:  0
Slimbook> print bootcmd
bootcmd=mmc rescan; for device in ${bootdevices}; do if test "$device" = "nand"; then if
ubi part nand0,1; then if ubifsmount ubi:boot; then setenv loadcmd "ubifsload "; if ${loa
dcmd} ${scriptaddr} ${bootscript}; then if imi ${scriptaddr}; then source ${scriptaddr};
fi; fi; fi; fi; fi; for unit in ${units}; do mmcinfo; for part in "1 2 3"; do for fs in "
ext2 fat"; do setenv loadcmd "${fs}load ${device} ${unit}:${part}"; if ${loadcmd} ${scrip
taddr} ${bootscript}; then if imi ${scriptaddr}; then source ${scriptaddr}; fi; fi; done;
 done; done; done;
Slimbook> 
```

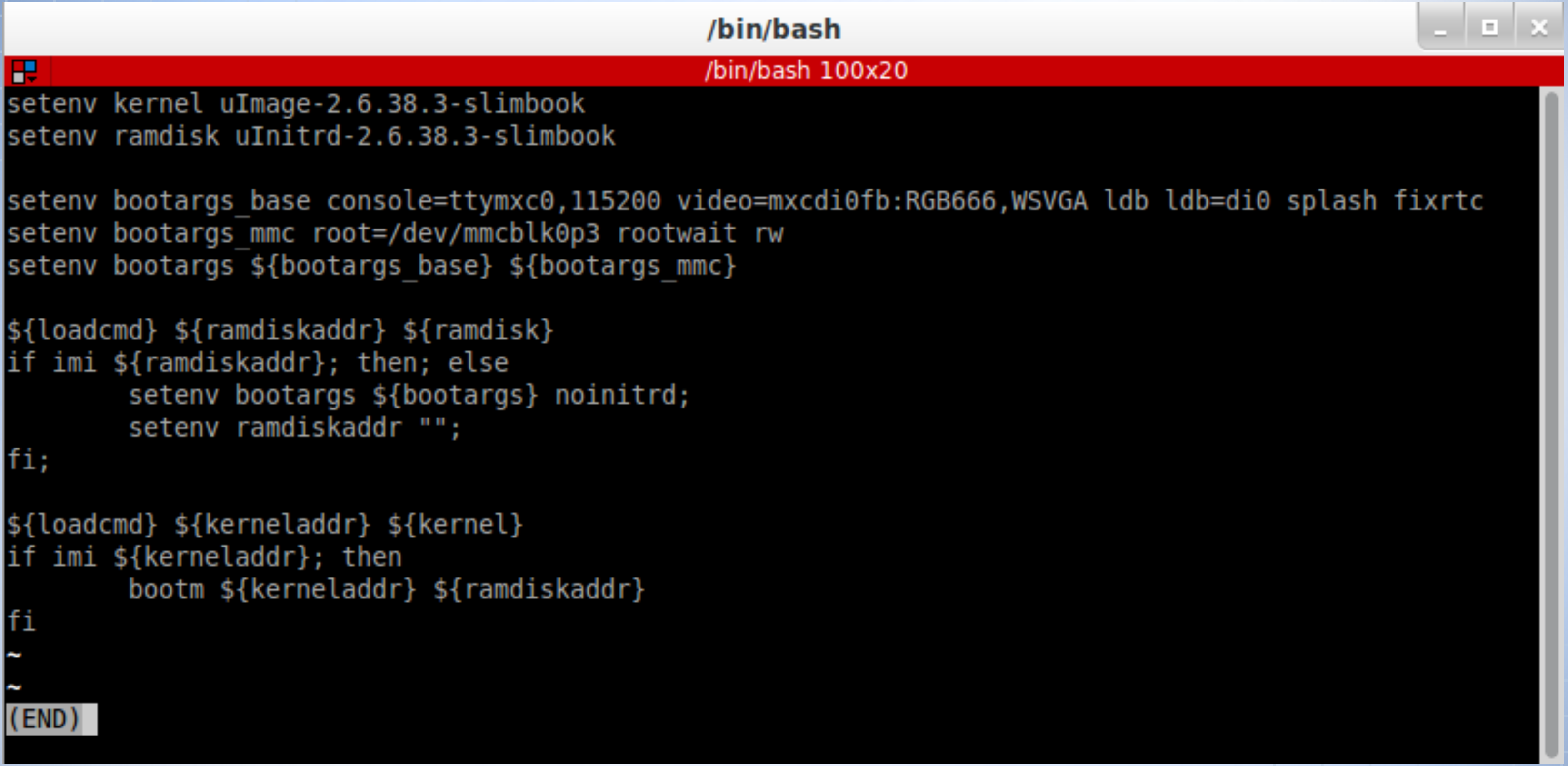
Boot scripts: Types

- Boot scripts are U-Boot shell scripts that boot from the supported boot device partitions. In Genesi software they can appear in one of 3 forms:
    - A “Meta-script”: a plain text file with variables like `%ROOTPARTITION%` and `%KERNELVERSION%` in place of appropriate values replaced during the Bamboo build. These end with the extension `.script`
    - Uncompiled boot script: a plain text file without the meta variables above, also ending in `.script`
    - Compiled boot script: file with a binary header and text body that is executable by the U-Boot console. These end with the extension `.scr`

Boot Scripts: General Workings

- Set environment variables for the name of the Linux kernel and initramfs files in the boot partition
- Set base kernel boot args
- Set medium-specific boot args
- Attempt to load[1] ramdisk into pre-set address in RAM
  - If this fails, specify “noinitrd” boot arg, wipe initramfs variable
- Attempt to load[1] kernel into pre-set address in RAM
  - If this succeeds, boot the kernel (and ramdisk if valid)

1: How does the boot script know how to load the kernel and ramdisk? The `loadcmd` environment variable is set prior to launching the boot script to the appropriate command.



```
setenv kernel uImage-2.6.38.3-slimbook
setenv ramdisk uInitrd-2.6.38.3-slimbook

setenv bootargs_base console=ttymxc0,115200 video=mxcdi0fb:RGB666,WSVGA ldb ldb=di0 splash fixrtc
setenv bootargs_mmc root=/dev/mmcblk0p3 rootwait rw
setenv bootargs ${bootargs_base} ${bootargs_mmc}

${loadcmd} ${ramdiskaddr} ${ramdisk}
if imi ${ramdiskaddr}; then; else
        setenv bootargs ${bootargs} noinitrd;
        setenv ramdiskaddr "";
fi;

${loadcmd} ${kerneladdr} ${kernel}
if imi ${kerneladdr}; then
        bootm ${kerneladdr} ${ramdiskaddr}
fi
~
~
(END)
```

Editing Boot Scripts

When you are testing changes to the Linux kernel, you may find it useful to change the boot args to e.g. enable early prints, increase the ring buffer, or increase the log level at boot. To do this:

- Open `MMCSystem/home/installer` of installer image in a file browser
- Make a copy of file `boot.ubi.script` (e.g. `boot.ubi-debug.script`) and open in your favorite text editor
  - This is the boot script that, when living on an SD, boots the kernel from SD but has NAND as the root partition
- Make changes to variable `bootargs_base` (usually adding an argument)
- Open a terminal and type this command (requires `u-boot-tools`):
  - `mkimage -A arm -T script -C none -d boot.ubi-debug.script boot.scr`
  - Make sure you make a backup of your own `.scr` file (`boot.ubi-debug.scr`)
- Put the resulting `boot.scr` on the `MMCBoot` partition

genesi