Release Notes for the QNX Neutrino 6.4.0 BSP for Freescale MPC8360E MDS trunk#

System requirements#

Target system#

- QNX Neutrino RTOS 6.4.0
- Board version: Freescale MPC8360E MDS
- ROM Monitor version U-Boot 1.3.0-rc2
- 32M flash
- 256MB DDR RAM

Host development system#

- QNX Momentics 6.4.0, one of the following host systems:
  - QNX Neutrino 6.4.0
  - Microsoft Windows Vista, XP SP2 or SP3, 2000 SP4
  - Linux Red Hat Enterprise Workstation 4 or 5, Red Hat Fedora Core 6 or 7, Ubuntu 6.0.6 LTS or 7, or SUSE 10
- Terminal emulation program (Qtalk, Momentics IDE Terminal, tip, HyperTerminal, etc.)
- RS-232 serial port
- NULL-modem serial cable
- Ethernet link

System layout#

The table below depicts the memory layout for the image and for the flash.

Memory layout

Item	Address
OS image loaded at:	0xfe700000
IPL location:	0xfef00000
Flash base address	0xfe000000

Getting Started#

Step 1: Connect your hardware#

Connect the serial cable to the first serial port of the MPC8360E MDS board to the first serial port of your host machine (e.g. ser1 on a Neutrino host).

- If you have a Neutrino host with a serial mouse, you may have to move the mouse to the second serial port on your host, because some terminal programs require the first serial port.

Step 2: Build the BSP#

You can build a BSP OS image from the source code or the binary components contained in a BSP package. For instructions about building a BSP OS image, please refer to the chapter Working with a BSP in the Building Embedded Systems manual.

Step 3: Transfer the OS image to the target#

Boot OS images by ROM monitor#

1. On your host machine, start your favorite terminal program with these settings:

- Baud: 115200
- Bits: 8
- Stop bits: 1
- Parity: none
- Flow control: none

2. Setting up the environment

- Apply power to the target board. You should see the U-Boot ROM monitor output.
- You can use TFTP download (the default) to transfer an OS image to the board, as described below.

This method requires a raw image, which the buildfile creates by default.

On your target, type the following, filling in the appropriate IP addresses and `ifs` file:

```
=> setenv ipaddr 192.168.200.2
=> setenv serverip 192.168.200.1
=> setenv bootfile ifs-mpc8360mds.raw
=> setenv loadaddr 0x200000
=> setenv bootcmd 'tftpboot $loadaddr $bootfile; go $loadaddr'
=> setenv bootdelay 2
=> saveenv
Saving Environment to Flash...
Un-Protected 1 sectors
Erasing Flash...
flash erase done
Erased 1 sectors
Writing to Flash... done
Protected 1 sectors
=> boot
```

At this point you should see the ROM monitor download the boot image, indicated by a series of number signs. You'll also see output similar to this when it completes downloading:

```
geth: PHY is Marvell 88E11x1 (1410cc2)
FSL GETH0: Full Duplex
FSL GETH0: Speed 1000BT
FSL GETH0: Link is up
Using FSL GETH0 device
TFTP from server 192.168.200.1; our IP address is 192.168.200.2
Filename 'ifs-mpc8360emds.raw'.
Load address: 0x200000
Loading: ######################################################################    #############################
#################################
done
Bytes transferred = 819804 (c825c hex)
## Starting application at 0x00200000 ...
```

You should see the QNX Neutrino welcome message on your terminal screen:

```
Welcome to QNX Neutrino trunk on the Freescale MPC8360E MDS board
```

Using the QNX IPL code#

1. Flashing the IPL and the OS image

- The images directory contains a mkflash script that creates a 32 MB binary flash image that contains the IPL and the OS image. The IPL is stored at flash offset 0x0 and 0x1F000000, so that the board can boot low and high. Th OS-image is stored at at offset 0x1000000. The IPL's default behavior is to scan the flash strarting at flash offset 0x1000000 for an OS image. The flash image can be loaded to the flash using a flash programmer.
- If your MPC8360E MDS board has u-boot installed, then both the IPL and the OS image can be loaded to RAM using tftpboot, then copied to the flash. For example, you could have u-boot stored at the bottom of the flash and the IPL at the top. Here is an example of how to use u-boot to write the IPL and OS image to the flash:

```
erase fff00000 fffffff
tftpboot 200000 ipl-mpc8360e
cp.b 200000 fff00000 8000
erase ff000000 ff1fffff
tftpboot 200000 ifs-mpc8360emds.raw
cp.b 200000 ff000000 200000
```

Note: High/Low boot is configured via the BMS bit in the Reset Configuration Word (RCW). The RCW can be read from offset 0 of the flash or from the on-board FPGA. Please read the board documentation for more details.

2. Using IPL

Note: This section assumes that the IPL has already been flashed into the CPU card.

Causion: This method requires an binary image. You have to modify the buildfile to create this format. Change this:

```
[virtual=ppcbe,raw]
```

to this:

```
[virtual=ppcbe,binary]
```

Rename **ifs-mpc8360emds.raw** to **ifs-mpc8360emds.bin** in images/Makefile, and then rebuild the image.

- After you apply power to the board, the IPL will run and you should see the following message on your host screen:

```
Welcome to QNX Neutrino on the Freescale MPC8360E MDS board
Hit 'd' to download a Neutrino image with sendnto
Press any other key to boot from flash
```

- Press **d** to transfer the OS image over the serial link.

Note: Pressing any other key would tell the IPL to boot OS image from flash. This assumes that the OS image has already been copied into the flash.

- Copy the OS image from your host to the target using this command:

On a Windows host:

```
sendnto -b115200 -dCOM1 ifs-mpc8360emds.bin
```

On a QNX Neutrino host:

```
sendnto -b115200 -d/dev/ser1 ifs-mpc8360emds.bin
```

On a Linux host:

```
sendnto -b115200 -d/dev/ttyS0 ifs-mpc8360emds.bin
```

You should now see the Neutrino welcome message on your terminal screen:

```
Welcome to QNX Neutrino 6.3 on the Freescale MPC8360E MDS board
```

Step 4: Start working with Neutrino OS#

You can now test the OS simply by executing any shell builtin command or any command residing within the OS image (e.g. l

Once the initial image is running, you can update the OS image using the network and flash drivers. For sample command lines

Creating a flash partition#

1. Enter the following command to start the flash filesystem driver:

```
devf-generic -s0xFE000000,32M
```

2. To prepare the area for the partition, enter the following command:

```
flashctl -p/dev/fs0 -o0 -l6M-ve
```

3. Format the partition:

```
flashctl -p/dev/fs0p0 -o0 -l6M-vf
```

4. Slay, then restart the driver:

```
slay devf-generic
devf-generic -s0xFE000000,32M;
```

You should now have a /fs0p0 directory where you can copy files to.

Driver Command Summary#

The driver command lines below are specific to the Freescale MPC8360E MDS board. See the online docs for each driver for additional command-line options and other details.

NOTE: Some of these drivers are commented out in the default buildfile. To use the drivers in the target hardware, you'll need to uncomment them in your buildfile, rebuild the image, and load the image into the board.

Component	Buildfile Command	Required Binaries	Required Libraries	Source Location
Startup	`startup-mpc8360e`	.	.	`src/hardware/startup/`

				boards/ mpc8360e
Serial	devc-ser8250 -e -F -c264000000 -b115200 0xe0004500,0x09 0xe0004600,0x0A	devc-ser8250	.	src/hardware/devc/ser8250
PCI	pci-mpc83xx	pci-mpc83xx pci	.	src/hardware/pci/mpc83xx
Network	io-pkt-v4 -ducc_ec-mds mac=00e02991234c,channel=1,phy=0,mode=GMII -ptcpip	io-pkt-v4 ifconfig	devn-ucc_ec-mds.so libsocket.so devn-speedo.so	src/hardware/devn/ucc_ec/ppc/mds.dll.be
I2C	i2c-mpc5200 -p0xe0003000 -c88000000 -i14 --u1 (Channel 1) i2c-mpc5200 -p0xe0003100 -c88000000 -i15 --u0(Channel 2)	i2c-mpc5200	.	src/hardware/i2c/mpc5200

Some of the drivers are commented out in the default buildfile. To use the drivers in the target hardware, you'll need to uncomment them in your buildfile, rebuild the image, and load the image into the board.

Note: PCI driver must be started after the I2C driver is active.

Known Issues for this BSP#

- The IPL included in this BSP release is compatible only with the MPC8360EA-MDS-PB (board revision Pilot) version of the development platform. It has been tested under the following configurations:

MPC8360E silicon rev. 2.1
Board assembly rev. 0.51
Factory preset switches
256 MB DDR2 SO-DIMM pugged into U33

- The BSP's src/hardware/devn/ucc_ec directory contains a README file outlining the specific limitations of the network driver.
- You can not use the second network interface. (QNX Ref# 63723)
- All QNX BSPs ship with an images/Makefile whose purpose is to build an image filesystems (ifs) file using drivers built from source. Currently, it's set up to copy buildfiles from within the source directory structure (source/hardware/startup) to the images directory so that it can generate an ifs. This happens automatically when you do a top-level make, but this mechanism doesn't work when you're making an binary BSP. (QNX Ref# 38242)

Workaround: For a binary-only BSP, change these lines in the root Makefile:

all: install links $(if $(wildcard images/*),images) @echo done