Release Notes for the QNX Neutrino 6.4.0 BSP for Freescale MPC5121e ADS 1.0.0#

System requirements#

Target system#

- QNX Neutrino RTOS 6.4.0
- Board version: Freescale MPC 5121e ADS reference board
- ROM Monitor version: UBoot or QNX IPL
- AUDIO: AC'97-compliant onboard audio controller
- AUDIO: I2S audio controller
- SERIAL: Programmable Serial Controller
- GRAPHICS: Integrated Display Unit + Power VR MBX Lite
- NETWORK: 10/100 BASE-T Ethernet controller
- NAND FLASH: 512MB HYNIX HY27UG088G5M NAND FLASH (Only support CE1 at this point)
- FLASH: 64MB Spansion GL256N11FFI02
- I2C: Inter-Integrated Circuit protocol module
- SPI: Programmable Serial Controller
- USB: EHCI USB Host Controller
- CAN: MSCAN controllers
- PCI: PCI controller
- BLOCK: ATA/ATAPI Interface

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
- Latest version of the QNX Momentics Character Driver Development Kit (DDK)
- Terminal emulation program (Qtalk, Momentics IDE Terminal, tip, HyperTerminal, etc.)
- RS-232 serial port
- NULL-modem serial cable
- Ethernet link

System layout#

The tables below depict the memory layout for the image and for the flash.

Item	Address
OS image loaded at:	0x80200000
OS image begins execution at:	0x80200000
NOR Flash base address	0xfc000000
NAND Flash base address	0x40000000

Getting Started#

Step 1: Connect your hardware#

- Connect the serial cable to the first serial port of the Freescale MPC5121e ADS reference board and to the first serial port of your host machine (e.g. "ser1" on a Neutrino host, "/dev/ttyS*" on Linux, "COM1" on Windows, etc).

> **Note**:
> If you have a Neutrino host with a serial mouse,you may have to move it 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.

Step 3: Transfer the OS image to the target#

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.Apply power to the board.

3.Configure the board for a TFTP transfer.

```
>setenv ipaddr <ipaddress>
>setenv serverip <serverip>
>setenv netmask <netmask>
>setenv bootfile ads5121e.raw
>saveenv
```

4.Copy the image from your host to the target using this command:

tftpboot 0x200000
go 0x200000

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

```
Welcome to QNX Neutrino 6.4.0 on the Freescale ADS5121e board
```

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

Creating a flash partition#

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

devf-mgt5200 -s0xfc000000,64M -r

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

flashctl -p/dev/fs0 -o7m -l50m -ve

3. Format the partition:

```
flashctl -p/dev/fs0p0 -o7m -l50m -vf
```

4. Slay, then restart the driver:

```
slay devf-mgt5200
devf-mgt5200 -s0xfc000000,64M -r
```

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 MPC5121e ADS reference board.See the online docs for each driver for additional command-line options and other details.

Serial#

Command:

```
devc-sermpc512x -c 66000000 -b 115200 -p 3 -e -F -S -x 40
```

> **Note**:
> For MPC5121e ADS HW Rev 2D and later, the clock frequency is 66MHz; For MPC5121e ADS HW Rev 2, the clock frequency is 99MHz

Required libraries:

- devc-sermpc512x

CAN#

Command:

```
dev-can-mpc5121 -u 0 can0
```

Required libraries:

- dev-can-mpc5121
- canctl

PCI#

Command:

```
pci-mpc5121
```

Required libraries:

- pci-mpc5121
- pci

Network#

Command:

```
io-pkt-v4 -d mpc5121 mac=662200043039 verbose -ptcpip
```

Required libraries:

- devn-mpc5121.so
- libsocket.so
- npm-tcpip.so
- devnp-shim.so
- io-pkt-v4

NOR Flash#

Command:

```
devf-mgt5200 -s0xfc000000,64M -r
```

Required libraries:

- devf-mgt5200
- flashctl

NAND Flash#

Command:

```
fs-etfs-mpc5121-e -m /fs/etfs
fs-etfs-mpc5121 -m /fs/etfs
```

Required libraries:

- fs-etfs-mpc5121
- etfsctl

USB#

Command:

```
io-usb -d ehci-mpc512x ioport=0x80004000,irq=44,verbose=1 -vvv &
```

> **Note**:
> Please use the mini-ATX power supply and remove P1 jumper to obtain the enough power for USB, otherwise the system may reset unexpected.

Required libraries:

- devu-ehci-mpc512x.so
- libusbdi.so
- io-usb
- usb

SPI#

Command:

```
spi-master -d mpc5121
```

Required libraries:

- spi-mpc5121.so
- spi-master

Audio AC97#

Command:

```
io-audio -d mpc512x
```

> **Note**:
> DMA mutex infrastructure need to be created before starting this driver.

Required libraries and binaries:

- deva-ctrl-mpc512x.so
- deva-mixer-ac97.so
- libasound.so.2
- deva-util-restore.so
- libdma-mpc512x.so
- io-audio

Audio I2S#

Command:

```
io-audio -d mpc5121_i2s
```

> **Note**:
> DMA mutex infrastructure need to be created before starting this driver.

Required libraries and binaries:

- deva-ctrl-mpc5121_i2s.so
- libasound.so.2
- deva-util-restore.so
- libdma-mpc512x.so
- io-audio

I2C#

Command:

```
i2c-mpc5200 -o1 -p0x8001700 -c100000000 -i9
```

Required libraries:

- i2c-mpc5200

Graphics#

Command:

```
io-display -dvid=0,did=0
```

Required libraries:

- devg-mpc5121.so
- libgf.so.1
- libffb.so.1
- io-display
- display.conf
- mpc5121.conf

EIDE Block#

Command:

PIO mode:
```
devb-eide-mpc5121 eide nobmstr
```

DMA mode:
```
devb-eide-mpc5121
```

Required libraries:

- devb-eide-mpc5121
- libdma-mpc512x.so
- libcam.so
- io-blk.so
- cam-xxx.so
- fs-xxx.so

DMA Mutex#

Command

```
mpc512x_dma_cfg -c
```

Required libraries and binaries:

- mpc512x_dma_cfg

RTC#

Command:

```
rtc m41t6x
rtc -s m41t6x
```

> **Note**:
> The rtc command must be run after the I2C driver is active

Required binaries:

- rtc
- date
- i2c-mpc5200

Known issues for this BSP#

- The current NAND Flash driver works on MPC5121e ADSRev3.2 boards. The NAND interface was modified (significantly) for the MPC5121e ADS Rev3.5 and Rev 4 boards. A new NAND Flash driver will be delivered in a follow-on release of the MPC5121e ADS BSP.
- The current EIDE driver only works in both PIO mode and DMA mode. The DMA libary libdma-mpc512x.so is required for both modes.
- There are no real audio codec connect to the I2S interface on the board. The I2S audio driver deva-ctrl-mpc5121_i2s.so was tested by loopback mode. An invalid clock rate issue is observed on the hardware.
- The resolution of the playback/capture positional information returned by the audio drivers deva-ctrl-mpc512x.so and deva-ctrl-mpc5121_i2s.so to the client is limited to the fragment size.
- Some Open GLES conformance tests fail with devg-mpc5121.so. (QNX Ref# 62594)
- In those instances where the the ROM monitor's MAC address is different from the one you pass in when running io-net and/or io-pkt, the host can cache the ROM monitor's address. This can result in a loss of connectivity. **`Workaround`**: If you need to specify a MAC address to io-net and/or io-pkt, we recommend that you use the same MAC address that the ROM monitor uses. This will ensure that if the host caches the ROM monitor's MAC address, you'll still be able to communicate with the target. Otherwise you might need to delete the target's arp entry on your host.