i.MX31 PDK 1.5 Linux
User's Guide

Document Number: 926-77208
Rev. 1.5
02/2009

freescale™
semiconductor

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 010 5879 8000

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

About This Book

This guide explains how to build and install the Linux® BSP to the i.MX31 3-Stack board.

For more information about installing the BSP and toolchain for the 3-Stack board, and building a root file system and a FLASH root file system, see the *i.MX PDK 1.5 Linux Reference Manual*.

Audience

This document is intended for software, hardware, and system engineers who are planning to use the product and for anyone who wants to understand more about the product.

Organization

This document contains the following chapters.

Chapter 1 Introduces the contents and organization of the BSP Release Package.

Chapter 2 Explains how to build the kernel and driver from source.

Chapter 3 Explains how to download the image to target.

Chapter 4 Explains how to run the image using the downloaded flash and NFS.

References

The following documents were referenced to build this document:

1. i.MX PDK 1.5 Linux Reference Manual
2. i.MX Advanced Toolkit Standard Version User's Guide
3. RedBoot documentation for each platform

Chapter 1
Introduction

The i.MX 3-Stack Linux BSP is a collection of binary, code, and support files that you can use to create a Linux kernel image and a root file system image for the i.MX 3-Stack board.

1.1 RedBoot Utilities

For information about RedBoot utilities, see `redboot_mx31_mx32.pdf`.

1.2 Flash Boot Loader

The i.MX 3-Stack Linux delivery package contains RedBoot Flash bootloader binaries for the 3-Stack board. The RedBoot Flash bootloader will be loaded to RAM memory from flash when the system is powered on. The bootloader initializes the i.MX board, loads the kernel image from flash, and then jumps to the kernel startup entry.

The `redboot_200845.zip` file contains bootloader binaries for the i.MX 3-Stack board. This is the RedBoot binary that supports NOR and NAND boot.

1.3 Linux Kernel and Driver

The Freescale 3-Stack BSP contains the Freescale Linux 2.6.26 3-Stack kernel, driver source code, and a pre-built kernel image. You can obtain the 3-Stack kernel image from the following location:

```
imx31_3stack/zImage
```

1.4 Root File System

The root file system package provides busybox, common libraries, and other fundamental elements. The 3-Stack BSP contains the original root file system package:

```
imx31_3stack/rootfs.jffs2
imx31_3stack/rootfs.ext2.gz
```

Chapter 2
Building the Linux Platform

This chapter explains how to set up the build environment, install and build LTIB, set rootfs for NFS, and set up the host environment.

2.1 Setting Up the Build Environment

Setting up the build environment includes installing Linux OS and LTIB.

2.1.1 Install Linux OS using Linux Builder

Install a Linux distribution such as Fedora™ 4/5 or RedHat on one computer.

2.1.2 Installing/Building LTIB

To install and build LTIB, use these steps.

NOTE

In some Linux systems, the following procedure must be done with "root" permissions. However, these instructions are for performing the procedure "not as root".

1. Install the LTIB package not as root:

```
tar zxf <ltib_release>.tar.gz
./<ltib_release>/install
```

   This command installs LTIB.

NOTE

During certain steps in this process you will configure the LTIB directory. The specified LTIB directory is referenced in this and other documents as <LTIB dir>.

2. Build LTIB:

```
cd <LTIB dir>
./ltib -m config
```

3. Select the "Freescale iMX reference boards" platform, and then exit, saving the changes.
4. At the next menu, select the platform type and package profile. Exit saving changes.
5. Run the following command:

```
./ltib
```

When complete, you can obtain the kernel image from:

```
<LTIB dir>/rootfs/boot/zImage
```

NOTE

You must set the network parameters in LTIB in order to boot via NFS:

```
./ltib -c
```

Set the network parameters in the following path:

Target System Configuration

Options--->

start networking

Network setup

IP address

netmask

broadcast address

gateway address

nameserver IP address

2.2 Setting rootfs for NFS

There are two ways to set the rootfs for NFS on this package.

- Use the `ext2` format `rootfs` package provided in the distribution OR
- Use the `rootfs` that is created after making the build of the kernel.

Method 1: Using the `rootfs` package in the distribution

Use the following commands to set the `rootfs` directory for NFS (you must be the root user for this operation):

```
mkdir /mnt/rootfs
cp imx31_3stack/rootfs.ext2.gz  /tools
cd /tools
gunzip rootfs.ext2.gz
mount -o loop -t ext2 rootfs.ext2 /mnt/rootfs
cp -r /mnt/rootfs .
export ROOTFS_DIR=/tools/rootfs
```

Method 2: Using the rootfs created after the kernel build

Instead of using the rootfs.ext2.gz, you might use the root file system in <LTIB dir>.

```
export ROOTFS_DIR=/<LTIB dir>/rootfs
```

Other methods: For other ways to make a file system image file, see the *i.MX PDK 1.5 Linux Reference Manual.*

2.3 Setting up the Linux Host

To set up the Linux host system, use these steps.

1. Turn off the firewall, to enable the `tftp` to work:

   ```
   iptables -F
   ```

   OR, at the command line, type:

   ```
   setup
   ```

2. Install the **tftp** server, and then verify the installation using the following command:

   ```
   rpm –q tftp
   ```

3. Install the **nfs** server, and then verify the installation by confirming that the `/etc/init.d/nfs` file exists.

4. Create the `tftboot` directory.

   The kernel images and anything that needs to be uploaded by `tftp` (such as the **zImage** kernel image) will be stored in this directory.

   ```
   mkdir /tftpboot
   ```

5. As root, edit `/etc/xinetd.d/tftp` to enable tftp as follows:

   ```
   {
   disable  =  no
   socket_type = dgram
   protocol = udp
   wait = yes
   user = root
   server = /usr/sbin/in.tftpd
   server_args = /tftpboot
   }
   ```

6. As root, run the following command on your Linux host machine:

   ```
   vi /etc/exports
   ```

7. Add this line in the file: `/tools/rootfs *(rw,sync,no_root_squash)` and save the file.

8. As root, restart the **nfs** and **tftp** servers on your host:

   ```
   /etc/init.d/xinetd restart
   /etc/init.d/nfs restart
   ```

9. Copy zImage and rootfs.jffs2 in the release package or LTIB to the tftp directory.

```
cp imx31_3stack/zImage /tftpboot
cp imx31_3stack/rootfs.jffs2 /tftpboot
```

Or

```
cp /<LTIB dir>/rootfs/boot/zImage /tftpboot
cp /<LTIB dir>/rootfs.jffs2 /tftpboot
```

NOTES

These instructions specify using an **nfs** server. Some Linux systems use **nfsserver**, rather than **nfs**. Use these instructions for either server type.

The command lines below make the rootfs directory for the i.MX31 PDK. The mkfs.jffs2 file must be installed in order to make the jffs2 rootfs. For detailed information, see the *i.MX PDK 1.5 Linux Reference Manual*.

mx31 3-Stack (Nand, Samsung K9K2G08R0A, 2k page size, erase size:0x20000,SLC):

```
mkfs.jffs2 -r rootfs -e 0x20000 -s 0x800 –n -o rootfs.jffs2
```

A Windows tftp program `tftp.zip` is located under LTIB release package `Common/` folder. You can install it in the Windows OS to set up the Windows tftp server for downloading images.

Chapter 3
Downloading the Image to Target

This chapter explains how to install the ATK, erase the NAND flash, download RedBoot using the ATK, boot from RedBoot, and download **zImage** and **rootfs** to the board, using RedBoot.

NOTE

Obtain the ATK tool install package from the Freescale release web site or from Freescale support.

3.1 Erasing the NAND Flash

If your board does not boot up or identifies an excessive number of bad blocks when booting up, erase NAND flash and power up the boards again. This problem may occur because a different NAND bad block handling mechanism had been used previously.

To erase the NAND Flash, use these steps:

1. Set the boot mode jumper and/or dip switches on the Debug board to internal boot mode (to program flash). For settings, see the *i.MX Advanced Toolkit User's Guide*.
2. Power on the 3-Stack board, run the ATK, and select the options shown in Figure 3-1.

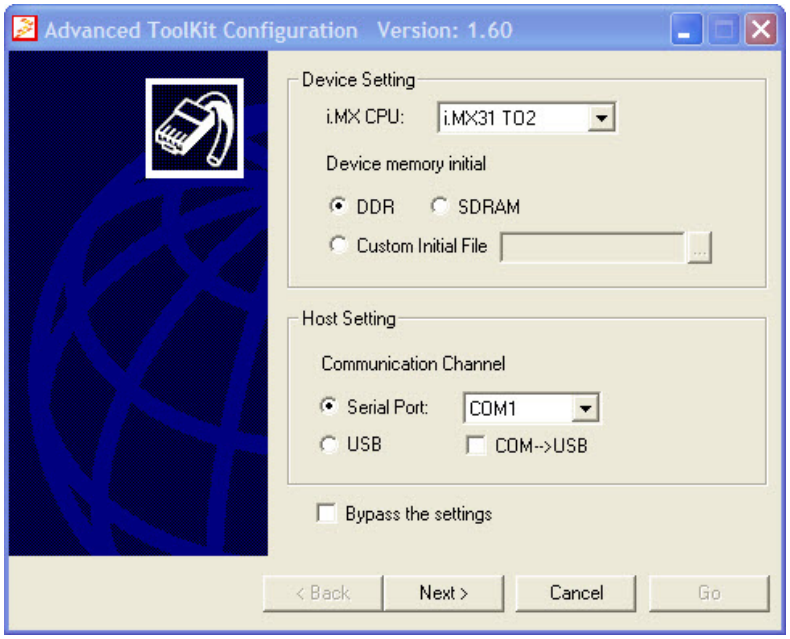

Figure 3-1 i.MX31 ATK Configuration Screen

3. Click **Next**.
4. Select **Flash Tool**, and then click **Go**.
5. Select the options shown in Figure 3-2, and click **Erase** to erase the flash.

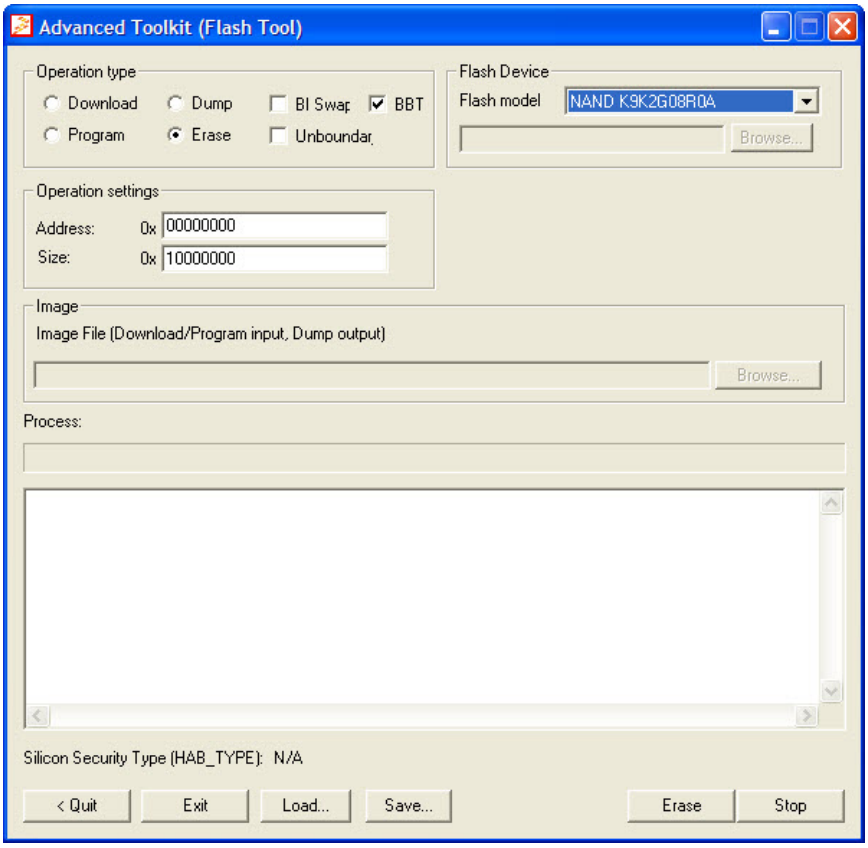

Figure 3-2 i.MX31 Erasing Memory

3.2 Downloading the RedBoot Bootloader

To download the RedBoot bootloader, use these steps:

1. Run the ATK.
2. Select the options shown in Figure 3-1, and then click **Next**.
3. Select **Flash Tool**, and then click **Go.**
4. Program RedBoot by selecting the options shown in Figure 3-3.

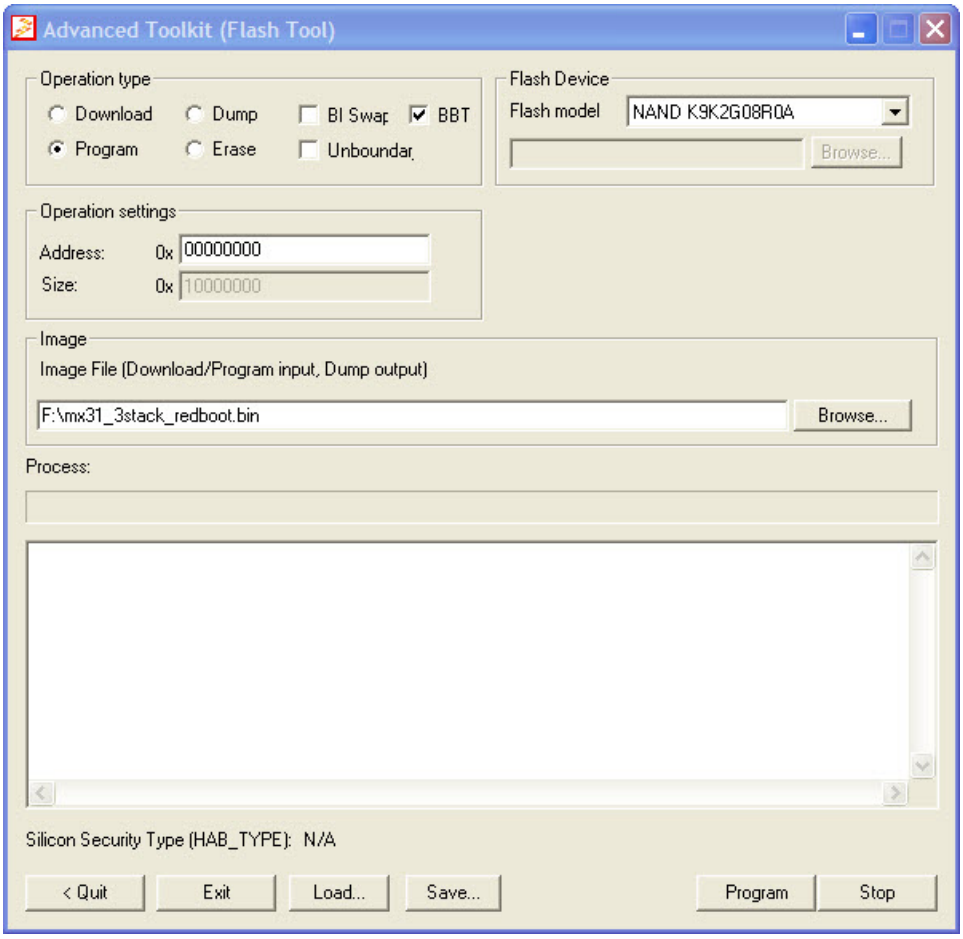

Figure 3-3 i.MX31 Programming RedBoot

5. Click **Browse** to select the bootloader.
6. Click **Program** to flash.

3.3 Running RedBoot

When RedBoot is installed, set the boot mode jumper and dip switches on the Debug board to boot from flash mode. When the board is powered up, RedBoot will run. You will first see RedBoot loading data, and then the RedBoot prompt. You can also run RedBoot from an SD or MMC card. For information about settings or SD and MMC card operation, see the RedBoot documentation.

3.4 Downloading the Kernel and rootfs Image

To program the kernel to NAND using the RedBoot bootloader, follow these steps:

1. Run RedBoot.
2. Initialize the flash, erasing any previous data:

```
RedBoot> fis init
About to initialize [format] FLASH image system - continue (y/n)? y
*** Initialize FLASH Image System
... Erase from 0x00080000-0x000a0000: .
... Program from 0x07ee0000-0x07f00000 at 0x00080000: .
```

3. Download the kernel image (zImage under /tftpboot) to RAM:

```
RedBoot> load -r -b 0x100000 zImage -h <host IP address>
```

4. Program the downloaded image into NAND Flash:

```
RedBoot> fis create –f 0x100000 kernel
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Erase from 0x00100000-0x002c0000: .............
... Program from 0x00100000-0x002a9c5c at 0x00100000: .............
... Erase from 0x00080000-0x000a0000: .
... Program from 0x07ee0000-0x07f00000 at 0x00080000: .
RedBoot>
```

5. Download the jffs2 format rootfs (**rootfs.jffs2** under `/tftpboot`) to RAM:

```
RedBoot> load -r -b 0x100000 rootfs.jffs2 -h <host IP address>
```

6. Program the downloaded rootfs into Flash:

```
RedBoot> fis create -f 0x600000 root
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Read from 0x07ee0000-0x07eff000 at 0x00080000: .
... Erase from 0x00600000-0x023a0000: ..........................................................
... Program from 0x00100000-0x01e9f5f4 at 0x00600000: ..........................................................
... Erase from 0x00080000-0x000a0000: .
... Program from 0x07ee0000-0x07f00000 at 0x00080000: .
RedBoot>
```

Chapter 4
Running an Image on the Target

This chapter explains how to run an image on the target from the downloaded flash and NFS.

4.1 Running the Image from the Downloaded NAND flash

These instructions assume that you have downloaded the kernel image and rootfs, using the instructions in Chapter 3.

To boot the kernel, issue the following RedBoot commands using these steps:

1. Power up the target configured for External Boot from Flash mode, and wait until the system enters the RedBoot environment.
2. Run the following command in the RedBoot prompt.

```
fis load kernel
exec -c "noinitrd console=ttymxc0,115200 root=/dev/mtdblock2 rw rootfstype=jffs2 ip=dhcp"
```

4.2 Running the Image from NFS

To set up **nfs**, you will need a serial command console (HyperTerminal, minicom, etc.) and a Linux PC with the **nfs** server installed. One solution would be to have two PCs: one Linux PC to use as the **nfs** server, and one Windows PC to use as the command console. Another solution would be to have one Linux PC used as both **nfs** server and command console through minicom.

To set up **nfs**, use these steps, which illustrate the procedure using a Linux PC as nfs server and a Windows PC as command console:

1. Connect your target to your local network via the Ethernet port on the Debug board.
2. Connect the target and your Windows PC via a serial cable.

3. Open HyperTerminal on the Windows PC, and then select the settings shown in Figure 4-1.

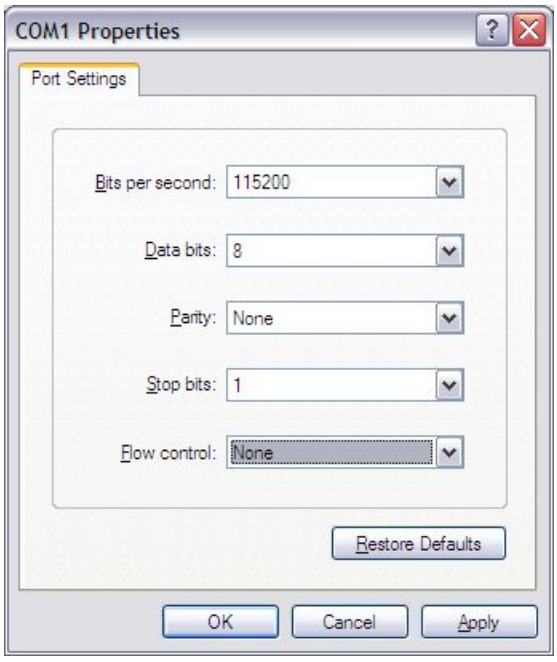

Figure 4-1 HyperTerminal Settings

4. Power up the target from external bootstrap and wait until the system enters the RedBoot environment.

5. Under the RedBoot environment, enter the following commands:

```
load -r  -b 0x100000 zImage
exec –c “noinitrd console=ttymxc0 root=/dev/nfsroot rootfstype=nfsroot nfsroot=<the IP address of the host machine>:/tools/rootfs rw ip=dhcp”
```

Another option may be to use static IPs rather than dhcp. In that case, the command line would be:

```
load -r  -b 0x100000 zImage
exec –c “noinitrd console=ttymxc0 root=/dev/nfsroot rootfstype=nfsroot nfsroot=<the IP address of the host machine>:/tools/rootfs/rootfs rw ip=<targetip>:<serverip>”
```

NOTE

The usage of static IPs applies in all sections of this document in which dhcp is mentioned.

The system will boot up with nfs mounted.

NOTE

You can have the target run automatically by configuring RedBoot via “fconfig” commands. Set “true” when prompted with “Run script at boot”. Enter the script that is to be auto executed after power on as indicated in the example below. The IP address of the tftp server may be entered at the “Default server IP address” prompt.

```
RedBoot> fconfig
Run script at boot: true
Boot script:
Enter script, terminate with empty line
>> load –r –b 0x100000 /tftpboot/zImage
>> exec –b 0x100000 –l 0x200000 -c "noinitrd console=ttymxc0 root=/dev/nfsroot rootfstype=nfsroot
nfsroot=<the IP address of the host machine>:/tools/rootfs rw ip=dhcp"
>>
Boot script timeout (1000ms resolution): 1
Use BOOTP for network configuration: true
Default server IP address: 10.192.221.167
Board specifics: 0
Console baud rate: 115200
Set eth0 network hardware address [MAC]: false
GDB connection port: 9000
Force console for special debug messages: false
Network debug at boot time: false
Default network device: lan92xx_eth0
Update RedBoot non-volatile configuration - continue (y/n)? y
... Read from 0x07ee0000-0x07eff000 at 0xeff80000: .
... Erase from 0xeff80000-0xeffa0000: .
... Program from 0x07ee0000-0x07f00000 at 0xeff80000: .
RedBoot>
... Program from 0x07ee0000-0x07f00000 at 0xeff80000: .
RedBoot
```