I.MX Platform Advanced Toolkit Standard Version

User's Guide

Rev. 1.71
03/2010

freescale™
semiconductor

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

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)
support@freescale.com

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 Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-521-6274 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

About This Book

This guide explains how to use the Advanced Toolkit (ATK) to perform Flash operations on a chip application development board.

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	Describes installation and configuration.
Chapter 2	Describes the ATK tools.
Chapter 3	Illustrates and describes the operational procedures.
Appendix	Provides code and parameter information.

Conventions

This document uses the following conventions:

`Courier`	Is used to identify commands, explicit command parameters, code examples, expressions, data types, and directives.
Italic	Is used for emphasis, to identify new terms, and for replaceable command parameters.

All source code examples are in C.

Definitions, Acronyms, and Abbreviations

The following list defines the abbreviations used in this document.

ATK	Advanced ToolKit
Device Program	A program that includes the RAM kernel and Flash library. The device program runs in external RAM and communicates with the host machine to execute specific operations

Chapter 1
Installation and Setup

The Advanced ToolKit (ATK) is a graphical user interface application for use in an i.MX platform for development and validation.

The ATK provides the following features:

- A Flash tool for downloading, programming, dumping, and erasing images in Flash memory to the i.MX board
- An image conversion tool for converting the following image file formats: binary to S-record, S-record to binary, and ELF to binary

The ATK runs on Microsoft® Windows platforms. You will need a PC with Windows 2000 SP4 or Windows XP.

NOTE

The ATK is used for application development and is not a mass production tool.

During software installation, you will install the software tools indicated in Table 1-1. To use a USB connection, you must also install the USB driver.

Table 1-1 Installation Requirements for Tools in ATK

Installations / Tools	Standard ATK Tool installation package	USB Driver (Optional for USB connection)
Flash Tool	√	√
Image Convert	√	

Table 1-2 identifies the supported flash types for each board.

Table 1-2 Supported Flash Types

HW	NOR	NAND	MMC/SD	SPI-NOR
i.MX25 TO1/TO1.1 3-Stack board	Not supported	K9LAG08U0M K9LBG08U0D	Supported	Not Supported
i.MX27 ADS board	S71WS256ND0	K9K1G08U0B	Not Supported	Not Supported
i.MX27 3-Stack board	Not Supported	K9F2G08R0A	Not Supported	Not Supported
i.MX31 ADS board	S71WS256ND0	K9K1G08U0B	Supported	Not Supported
i.MX31 3-Stack board	Not Supported	K9F2G08R0A	Supported	Not Supported

i.MX32 ADS board	S71WS256ND0	K9K1G08U0B	Supported	Not Supported
i.MX35 TO1/TO2/TO2.1 3-Stack board	SG29GL215	K9LAG08U0M K9LBG08U0D	Supported	Supported
i.MX37 3-Stack board	Not Supported	K9LBG08U0M	Supported	Supported
i.MX51 TO1 3-Stack board	Not Supported	K9G8G08U0M	Supported	SST25VF016B
i.MX51 TO2 3-Stack board	Not Supported	MT29F32G08QAA	Supported	SST25VF016B
i.MX51 EVK board	Not Supported	Not Supported	Supported	AT45DB321D

1.1 Installing the Standard ATK Package

Installation requires extracting the files to a temporary folder, and then executing a setup file.

Note

Before installation, you must first remove any previous installation, and then reboot your PC. To do so:

From the Start Menu, point to **Programs ->AdvancedToolkit**, and then click **Uninstall AdvancedToolKit**.

OR

From the Start Menu, point to **Settings->Control Panel-> Add or Remove Programs**, and then select to uninstall the ATK.

If you do not reboot the PC after removing the previous installation, the remainder of this procedure will fail after the installation is complete and the PC is rebooted. If the operation fails, perform the uninstall again and reboot.

To install the standard ATK tool, use these steps:

1. Extract the ATK package and place it in a temporary folder.
2. From the top level folder, execute FSL_ATK_TOOL_WINS_STD_INSTALL_<version_number>.exe.

   After executing FSL_ATK_TOOL_WINS_STD_INSTALL_<version_number>.exe, you do not need to reboot.
3. From the Start Menu, point to **Programs**, then to **Advanced ToolKit**, and then click **Advanced ToolKit**.

   The Advanced ToolKit configuration screen is displayed. At this point, The USB Drivers will also be installed. Then go to the next chapter to configure the hardware.

1.2 (Optional) Installment and uninstallment of USB Drivers

During software installation, you will install the software tool indicated in Tabl 1-1. To use a USB connection, you must also install the USB driver.

In install procedure, the USB Drivers are installed automatically after the software tools are installed. If not, you can go to `<TOOL_INSTALL_PATH>/windriver/ and run wd_install_4NT.bat to install it manually.`

`To uninstall USB driver, you can uninstall ATK or go to <TOOL_INSTALL_PATH>/windriver/ and run wd_uninstall_4NT.bat manually.`

`The old version of USB drivers can also be uninstalled with this program.`

To verify that the USB driver is ready to use, navigate to the Device Manager, Jungo.

You should see a folder similar to that in Figure 1-2.

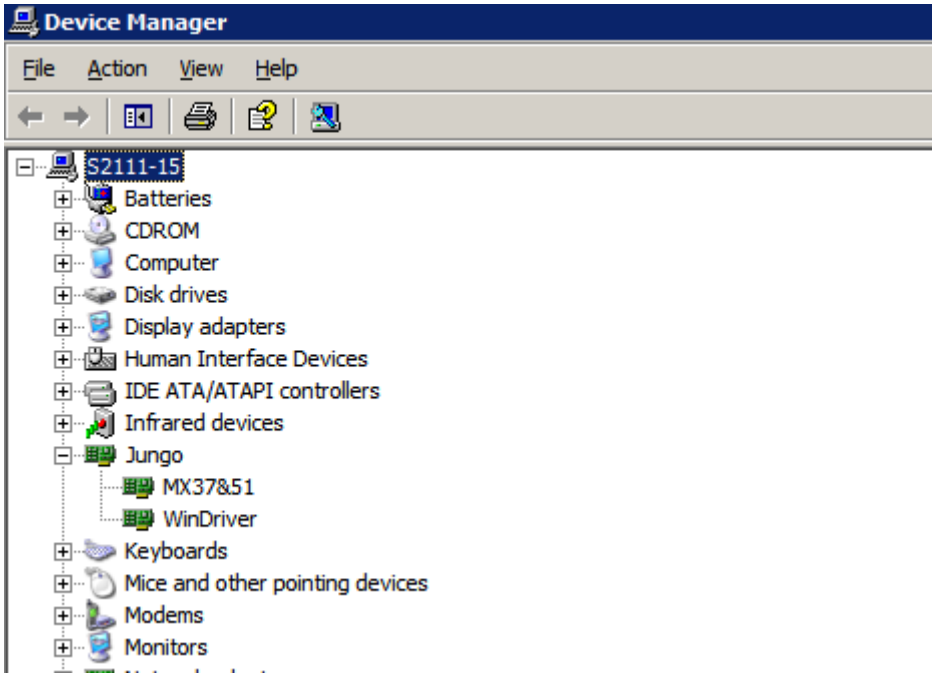

Figure 1.2 USB Interface

Chapter 2
Configuring the Hardware

Hardware configuration requires setting the board switches, and then connecting the board to the host computer.

2.1 Setting the Board Switches

Follow the instructions below for your board type.

Note

J28 should be open if programming redboot on pilot board.

2.1.1 i.MX31/i.MX32 ADS Board

To set the jumper pins, use these steps:

1. Change the JP22 jumper (**CLK SEL**) on the CPU board to select **FPM clock** as the **PLL reference clock**.

   Note

   Do not select CKIH as the PLL reference clock.

2. By default, jumper pins 1 and 2 are connected together; change the jumper state to connect pins 2 and 3 together.
3. Enable the UARTC PORT on the base board, by setting switch 4 on the SW1 switch box to ON.
4. Connect the PC COMx port to the UARTC port (the top one).
5. Use the information in Table 2-1 to set the SW2 switches for the desired boot mode.

Table 2-1 Configure Boot Mode on SW2 Switch

SW2	1	2	3	4	5
Internal boot (bootstrap mode for programming)	ON	ON	ON	ON	ON
External boot from Flash	ON	ON	OFF	ON	OFF

NOTE

The MX31 T01 chip does not support connecting the COM port using a USB UART cable. This is because the UART is

set to use Odd parity with two stop bits, which is not supported when using the USB UART cable in ROM codes.

2.1.2 i.MX31 3-Stack Board

Use Table 2-2 to select the settings on the Debug board for the desired boot mode.

Table 2-2 Switches on the i.MX31 3-Stack Board

Boot Mode Device	SW5	BOOT4 SW6	BOOT3 SW7	BOOT2 SW8	BOOT1 SW9	BOOT0 SW10
Internal boot (bootstrap mode for programming)	0	0	0	0	0	0
8-bit NAND Flash (2KB page) Ext	0	1	0	0	0	0

2.1.3 i.MX27 3-Stack Board

To configure the 1.MX27 3-Stack board, use the information in Table 2-4 to set the SW5-SW10 dips on the Debug board for the desired boot mode. The rest of the chip switches should remain as default.

Table 2-3 Switches on the i.MX27 3-Stack Board

Boot Mode Device	BOOT3 SW7	BOOT2 SW8	BOOT1 SW9	BOOT0 SW10
Internal boot (bootstrap mode for programming)	ON	ON	ON	ON

2.1.4 i.MX35 TO1 3-Stack Board

2.1.4.1 Preconditions

Ensure the chip has been fused as “non-security” or “engineering” mode.

2.1.4.2 Hardware Setting for Download Mode

To configure the i.MX35 TO1 3-Stack board, use the information in Table 2-4 to set the switches for the Debug and Personality boards.

Table 2-4 Switches on the i.MX35 TO1 3-Stack Board

Green Debug Board		Personality Board	
SW9	SW10	SW2	
		Dip1	Dip2
OFF	OFF	ON	ON

2.1.5 i.MX35 TO2/TO2.1 3-Stack Board

2.1.5.1 Preconditions

Ensure the chip has been fused as “non-security” or “engineering” mode.

2.1.5.2 Hardware Setting for Download Mode

To configure the i.MX35 TO2 3-Stack board, use the information in Table 2-5 to set the switches for the Debug and Personality boards. For Debug board, switches from SW5 to SW8 are set to OFF. For Personality board, other switch settings can be ignored.

Table 2-5 Switches on the i.MX35 TO2 3-Stack Board

Green Debug Board		Personality Board	
SW9	SW10	SW1	
		Dip1	Dip2
ON	ON	OFF	OFF

2.1.6 i.MX37 3-Stack Board

2.1.6.1 Preconditions

Check that these preconditions have been met:

1. Ensure the chip has been fused as “non-security” or “engineering” mode.
2. Ensure that the UART fix that is identified in MCIMX37CE.pdf (Errata Number 20) has been programmed to I2C EEPROM by the ICE program.

   To download the fix, use follow files and follow steps:

   — Modify the file path of `uart_dcd.bin` in `Marley_init_IIC.inc`: as follows:

```
readfile,raw,gui
    "D:\Project\Marley\Testcode\BIN\uart_dcd.bin"=0x41000000
```

— In RVDS, load `Marley_init_IIC.inc` in **Tools > Includes commands from files**.

— Load `IIC_PRO.axf` from **Target > Load Image**…, and then run this `.axf` file.

2.1.6.2 Hardware Settings

To configure the i.MX37 3-Stack board, use the information in Table 2-6 to set the switches for green debug board and CPU board. For Debug board, switches from SW5 to SW8 are set to OFF. For CPU board, other switch settings can be ignored.

Table 2-6 Switches on the i.MX37 3-Stack Board

Debug Board		CPU Board
SW9	SW10	SW21
ON	ON	Dip8
		OFF

2.1.7 i.MX51 TO1 3-Stack Board

2.1.7.1 Preconditions

Check that these preconditions have been met:

1. Ensure the chip has been fused as “non-security” or “engineering” mode.
2. If the chip is not “non-security” or “engineering” mode, use the iim_hab_type_key_fuse_pgm.axf and Elvis_init.inc to fuse the chip to “engineering” mode by following steps.
   — In RVDS, load `Elvis_init.inc` in **Tools > Includes commands from files**.
   — Load `iim_hab_type_key_fuse_prm.axf` from **Target > Load Image**…, and then run this `.axf` file.

2.1.7.2 Hardware Settings

To configure the i.MX51 TO1 3-Stack board, use the information in Table 2-7 to set the switches for green debug board and CPU board. Others are set to OFF.

Table 2-7 Switches on the i.MX51 TO1 3-Stack Board

Debug Board		CPU version 2 Board	
SW9	SW10	SW2	
		Dip1	Dip2
ON	ON	ON	ON

2.1.8 i.MX51 TO2 3-Stack Board

To configure the i.MX51 TO2 3-Stack board, use the information in Table 2-8 to set the switches for green debug board and CPU board. For Debug board, other switches from SW5 to SW8 are set to OFF. For CPU board, other switch settings can be ignored.

NOTE

.

Don't mount J28 on CPU board; otherwise it will fail to enter serial download mode on some boards.

Table 2- 8 switches on the i.MX51 TO2 3-Stack Board

Debug Board		CPU version 2 Board				
SW9	SW10	SW3	SW4			
		Dip1	Dip1	Dip2	Dip5	Dip6
ON	ON	OFF	ON	ON	OFF	OFF

2.1.9 i.MX25 TO1/TO1.1 3-Stack Board

To configure the 1.MX25 TO1/TO1.1 3-Stack board, use the information in Table 2-9 to set the switched for debug board and CPU board. For Debug board, other switches SW5 to SW8 are set to OFF. For CPU board, other switch settings can be ignored.

Table 2-9 Switches on the i.MX25 TO1/TO1.1 3-Stack Board

Debug Board		CPU Board
SW9	SW10	SW22
ON	ON	Dip1
		OFF

2.1.10 i.MX51 EVK Board

To configure the 1.MX51 EVK board for ATK, use the information in Table 2-10 to set the switches.

Table 2-10 Switches on the i.MX51 EVK Board

S-1	S-2	S-3	S-4	S-5	S-6	S-7	S-8	S-9	S-10
ON	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON

2.1.11 i.MX53 EVK Board

To configure the i.MX53 EVK board for ATK, use the information in Table 2-11 to set the switches.

Table 2-11 Switches on the i.MX53 EVK Board

SW3		SW2
DIP1	DIP2	DIP6
ON	ON	OFF(SD slot1)
		ON(SD slot3)

NOTE

Set SW2 dip6 to OFF if using SD SLOT1(on EVK top)
Set SW2 dip6 to ON if using SD SLOT3(on EVK bottom)

2.2 Connecting the Host to the Board

Follow the instructions below for your connector type.

2.2.1 USB Connection

To connect and use a USB, use these steps:

1. Connect the USB cable to the USB port of the board.
2. Select the i.MX CPU that you will use. Versions supporting USB include:
   — i.MX25 TO1 for the i.MX25 TO1 3-Stack board
   — i.MX25 TO1.1 for the i.MX25 TO1.1 3-Stack board
   — i.MX27-TO1 for the i.MX27 TO1 board

— i.MX27-TO2 for the i.MX27 TO2/i.MX27 3-Stack board
— i.MX31_TO2 for the i.MX31 TO2 3-Stack board
— i.MX31_TO2.0.1 for the i.MX31 TO2.0.1 3-Stack board
— i.MX32 for the i.MX32 ADS board
— i.MX35 TO1 for the i.MX35 TO1 3-Stack board
— i.MX35 TO2 for the i.MX35 TO2 3-Stack board
— i.MX37 for the i.MX37 TO1.1.1 3-Stack board
— i.MX51 TO1 for the i.MX51 TO1 3-Stack board
— i.MX51 TO2 for the i.MX51 TO2/TO3 3-Stack board
— i.MX51 TO2 for the i.MX51 EVK(TO2/TO3) board
— i.MX53 for the i.MX53 EVK board

NOTE

.

If your PC displays the message "USB not recognized" when the USB is insert in the i.MX31 boards, we recommend that you use the "Switch UART to USB" method

For i.MX31 TO1, ROM code does not support the USB download method.

For i.MX37 before TO1.1.1, USB is not supported for ROM code issue.

3. Power on the board and select USB for communications, as shown in Figure 2-1.

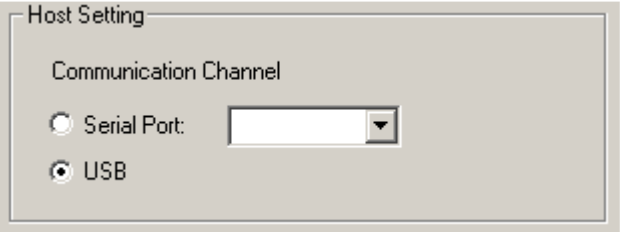

Figure 2-1 Select USB Connection

2.2.2 UART Connection

To connect and use a UART, follow these steps:

1. Connect a serial cable to the UART1 port of the ADS board.
2. Select the i.MX CPU that you will use.

   Supported versions include:
   - i.MX25 TO1 for i.MX25 TO1 3-Stack board
   - i.MX25 TO1.1 for i.MX25 TO1.1 3-Stack board
   - i.MX27-TO1 for i.MX27 TO1 board
   - i.MX27-TO2 for i.MX27 TO2 board and i.MX27 3-Stack board
   - i.MX31-TO1 for i.MX31 TO1 ADS
   - i.MX31-TO2 for i.MX31 TO2 ADS board
   - i.MX31-TO2.0.1 for i.MX31 TO2.0.1 ADS board
   - i.MX32 for i.MX32 ADS board
   - i.MX35 TO1 for i.MX35 TO1 3-Stack board
   - i.MX35 TO2 for i.MX35 TO1 3-Stack board
   - i.MX37 for i.MX37 3-Stack board
   - i.MX51 TO1 for i.MX51 TO1 3-Stack board
   - i.MX51 TO2 for i.MX51 TO2/TO3 3-Stack board
   - i.MX51 TO2 for the i.MX51 EVK(TO2/TO3) board
   - i.MX53 for the i.MX53 EVK board

NOTE

For i.MX25 TO1, UART baud rate should be set to 57600 limited by ROM code. The low UART baud rate will cause programming/dumping/erasing speed lower.

3. Power on the board.
4. Select the COM port that your PC will use (Figure 2-2).

Figure 2-2 Select UART Connection

2.2.3 Switch from UART to USB

NOTE

This feature is only available for Flash programming, dumping, and erasing on the i.MX31 boards. It works only for USB ULPI PHY, not for USB Serial and Atlas USB .

To switch from UART to USB, use these steps:

1. Connect a serial cable to the UART1 port of the ADS board and at the same time prepare for one USB cable; do not plug the USB cable in yet.
2. Select the i.MX CPU:
   — i.MX31-TO1 for the i.MX31 TO1 ADS board
   — i.MX31-TO2 for the i.MX31 TO2 ADS and MX31 3-Stack boards
3. Power on the board.
4. Select the Serial Port and COM port that your PC will use.
5. Select **COM > USB** (Figure 2-3).

Figure 2-3 Switch UART to USB

To erase, dump, and program using the Flash Tool, plug the USB cable using the information in Section 2.2.1, USB Connection.

Chapter 3
Configuring the ATK

When you launch the ATK, the configuration screen is displayed (Figure 3-1). For an example of a completed configuration, see Figure 3-2.

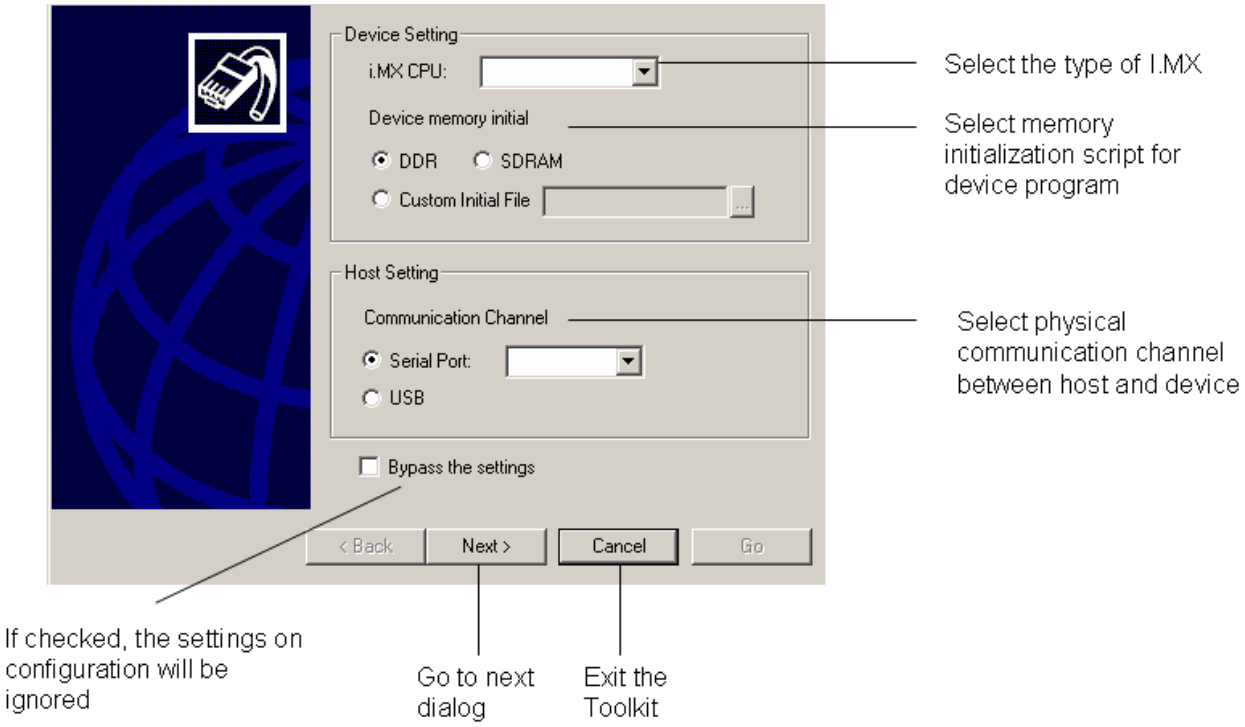

Figure 3-1 ATK Configuration Main Screen

To configure the ATK, use these steps:

1. In the Device Setting section, click the down arrow to select the i.MX CPU to use:
   - Select i.MX25 TO1 for the i.MX25 TO1 3-Stack board
   - Select i.MX25 TO1.1 for the i.MX25 TO1.1 3-Stack board
   - Select i.MX27-TO1 for the i.MX27 TO1 ADS board
   - Select i.MX27-TO2 for the i.MX27 TO2 board and i.MX27 3-Stack board
   - Select i.MX31-TO1 for the i.MX31 TO1 ADS board
   - Select i.MX31-TO2 for the i.MX31 TO2 board
   - Select i.MX31-TO2.0.1 for the i.MX31 TO2.0.1 board
   - Select i.MX32 for the i.MX32 ADS board
   - Select i.MX35 TO1 for the i.MX35 TO1 3-Stack board

— Select i.MX35 TO2 for the i.MX35 TO2 3-Stack board
— Select i.MX37 for the i.MX37 3-Stack board
— Select i.MX51 TO1 for the i.MX51 TO1 3-Stack board
— Select i.MX51 TO2 for the i.MX51 TO2/TO3 3-Stack board
— Select i.MX51 TO2 for the i.MX51 EVK(TO2/TO3) board

Note

From this version, **i.MX CPU** is mandatory even when **Bypass the settings** is checked

2. For **Device memory initial** (**DDR** or **SDRAM, DDR2 or MDDR**), select the memory initialization script to use for the device program.

   First check which DDR is used on your board:
   — **DDR**: Specifies the DDR initialization script.
   — **SDRAM**: Specifies the SDRAM initialization script.

   For the i.MX25 TO1 3-stack board:
   — **MDDR**: Specifies the MDDR initialization script.
   — **Custom Initial File**: If selected, click the button to select a custom memory initialization script. You can also use your own memory initialization file if it has the correct format. See the **Appendix** for details.

   For the i.MX25 TO1.1 3-Stack boards
   — **DDR2**: Specifies the DDR2 initialization script.

   Custom Initial File: If selected, click the button to select a custom memory initialization script. You can also use your own memory initialization file if it has the correct format. See the **Appendix** for details.

   For the i.MX35 TO1/TO2:
   — **DDR2**: Specifies the DDR2 initialization script.
   — **MDDR**: Specifies the MDDR initialization script.

   For the i.MX35 TO2.1:
   — **DDR2**: Specifies the DDR2 initialization script. The purple color of i.MX35 TO2.1 3stack CPU board is equipped with DDR2 memory.
   — **MDDR**: Specifies the MDDR initialization script. The white color of i.MX35 TO2.1 3stack CPU board is equipped with MDDR memory.

   For the i.MX51 TO1/TO2:
   — **DDR2**: Specifies the DDR2 initialization script.
   — **MDDR**: Specifies the MDDR initialization script.

   For the i.MX53:

— **DDR2**: Specifies the DDR2 initialization script.
— **MDDR**: Specifies the MDDR initialization script.

3. In the **Host Setting** section, select a physical communication channel between PC host and device:
   — **Serial Port**: Selects the UART as the communication channel.
   — **USB:** Selects the USB as the communication channel.
   — **COM > USB**: Switching from UART to USB **Communication Channel**. If **Serial Port** is checked, select a COM port (COM1 is preferred).
4. If **Serial Port** is checked and you want to switch to USB for dumping, erasing and programming, select **COM > USB**.
5. For **Bypass the settings**, use the following information:
   — If selected, the settings on this configuration screen are ignored. Note that the ATK supports the tools independent of the configuration.
   — If cleared, the tool will check the configuration settings, including opening the USB/COM port, and others.
6. Click **Next.**

   The Select Advanced Tools dialog is displayed.

NOTE

The Select Advanced Tools dialog will not be displayed if the board is not connected successfully, and if Bypass the Settings is not selected.

Figure 3-2 illustrates setting an ATK configuration for the **i.MX31** with **DDR Memory init** and a **UART** connection.

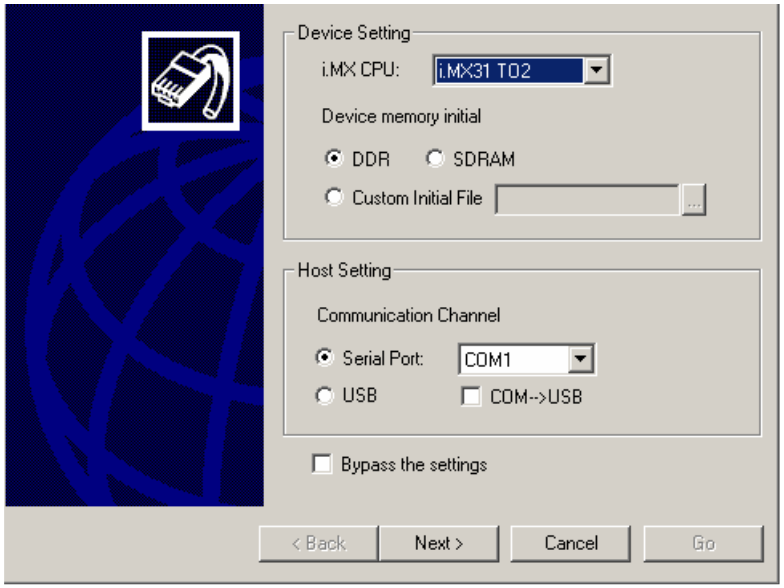

Figure 3-2 Example of an ATK Configuration

Chapter 4
Using the Flash Tool

You can use the Flash tool to download, program, dump, and erase images in Flash memory to the i.MX boards.

Figure 4-1 illustrates the Advanced ToolKit selection screen.

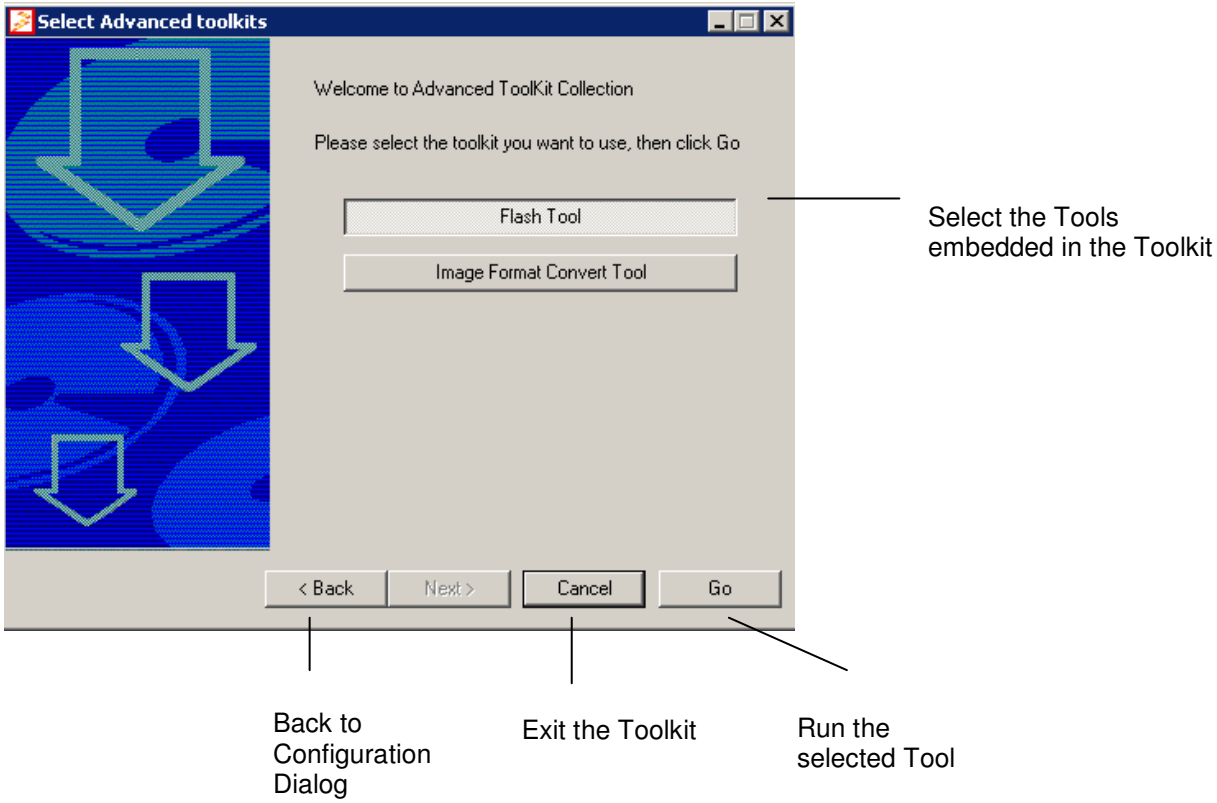

Figure 4-1 Select Advanced Tools

To use the Flash tool, follow these steps:

1. In the Advanced Tools screen, click **Flash Tool**, and then click **Go.**

   The Flash Tool screen is displayed (Figure 4-2).

   — For a detailed description of the options, see Table 4-1, which follows these instructions.

   — For examples of option choices, see Figure 4-3 Programming NOR Flash, Figure 4-4 Downloading an Image to RAM, Figure 4-5 Dumping from NOR Flash, and Figure 4-6 Erasing in NOR Flash.

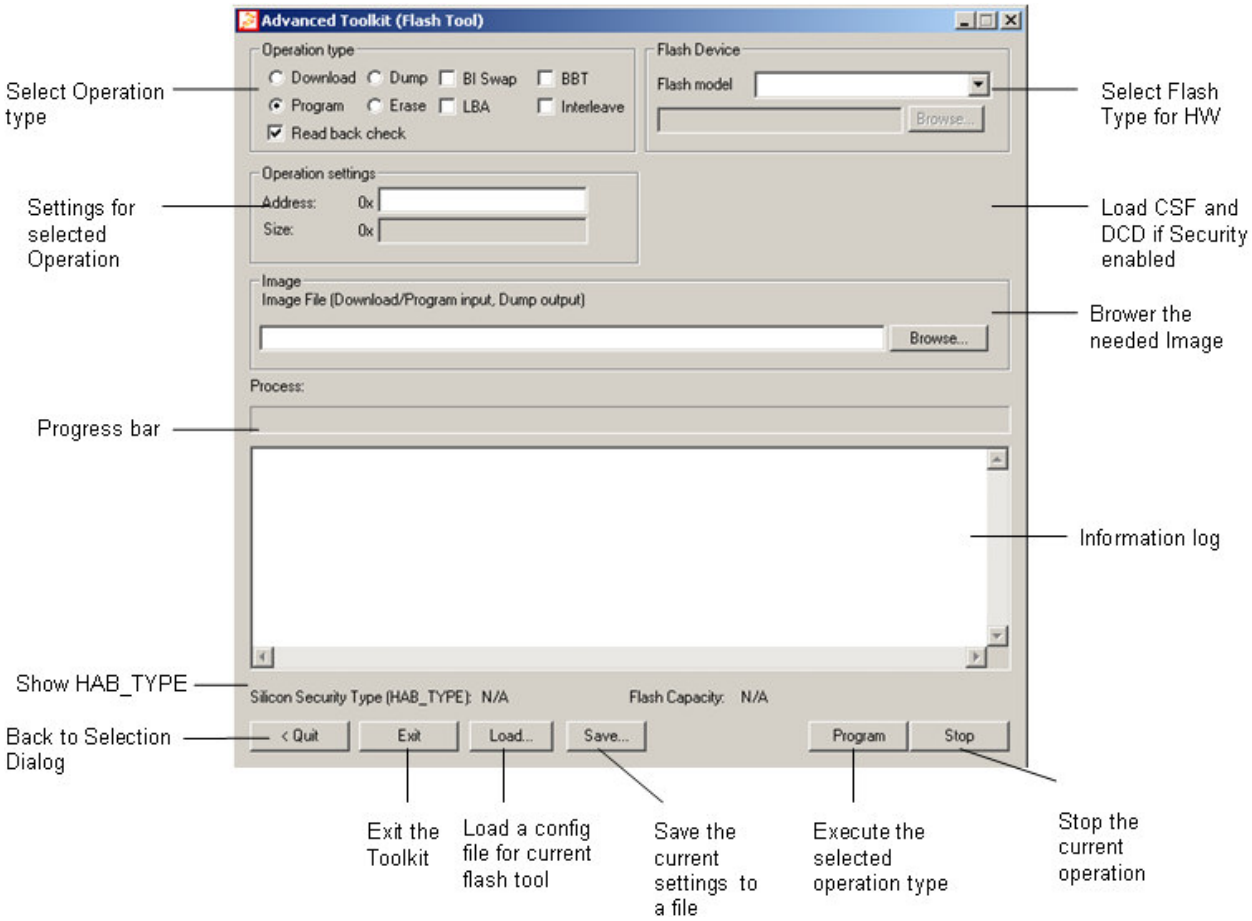

Figure 4-2 Flash Tool Screen

2. In the **Operation Type** field, select the Type of Operation to use.

   Generally, you should check the BBT for Linux images with a Flash Model type of NAND, and check Bi Swap for Windows Embedded CE images with a Flash Model type of NAND and a size exceeding 2k per page. In some specific Windows Embedded CE image cases, you will need to see to User's Guide to determine whether the Bi Swap is implemented in the software. If it is not implemented, Bi Swap should be unchecked. For

more information, see the Appendix Why and How to check BBT| Bi Swap in the Flash Tool.

3. In the **Flash Model** field, select the type (this option is unavailable if you selected **Download** as the **Operation Type**).

   For i.MX31 ADS boards, the supported Flash memories are:
   NOR (Spansion) Address from: 0xA0000000
   NAND Address from: 0x00000000

   For i.MX31 3-Stack boards, the supported Flash memories are:
   NAND Address from: 0x00000000

   NOR (Spansion) Address from: 0xA0000000

   SD Address from: 0x00000000

   MMC Address from: 0x00000000

   For i.MX32 3-Stack boards, the supported Flash memories are:
   NOR (Spansion) Address from: 0xA0000000

   NAND Address from: 0x00000000

   SD Address from: 0x00000000

   MMC Address from: 0x00000000

   For i.MX35 TO1/TO2 3-Stack boards, the supported Flash memories are:
   NOR (Spansion) Address from: 0xA0000000
   NAND Address from: 0x00000000

   MMC/SD Address from: 0x00000000

   For i.MX37 3-Stack boards, the supported Flash memories are:
   NAND Address from: 0x00000000

   MMC/SD Address from: 0x00000000

   For i.MX51 TO1/TO2 3-Stack boards, the supported Flash memories are:
   NAND Address from: 0x00000000

   MMC/SD Address from: 0x00000000

   For i.MX25 TO1/TO1.1 3-Stack boards, the supported Flash memories are:
   NAND Address from: 0x00000000

   MMC/SD Address from: 0x00000000

   SPI Address from: 0x00000000

   For i.MX51 EVK boards, the supported Flash memory is:

   MMC/SD Address from: 0x00000000

   SPI Address from: 0x00000000

   For i.MX53 EVK boards, the supported Flash memory is:

   MMC/SD Address from: 0x00000000

   SPI Address from: 0x00000000

NOTE

If you have built your own library for the target board, you can specify a different Flash Model.

Users don't have to know the specific type of NAND or NOR Spansion flash in use.

When you program a virgin SPI(Atmel) NOR flash which means it never has been programmed before, the first time, it will fail due to need change th page size. According to the SPI-NOR spec, it need one power down/up cycle. For such case, please power down the board first and then power up again for SPI-NOR flash programming.

4. Enter the address where the operation should start, and the size (which is entered only for dumping and erasing operations).
5. Browse to the file to be programmed, downloaded, dumped, or erased.
6. Select **program/download/dump/erase** to start the operation, and wait until it is complete. Error information is logged.

NOTE

If COM > USB is selected in configuration, follow any instructions that appear when performing any operations.

Table 4-1 Flash Tool Options

Section	Option	Description
Operation Type	Download	Downloads the image to the address specified, and then executes directly at the downloaded address.
	Dump	Dumps a selected area of Flash device to a file.
	Program	Programs a selected area of Flash from a file.
	Erase	Erases a selected area of Flash.
	BI Swap	Swaps Bad block identifier data, which is one solution for Bad block identifier data in the case that Nand Flash is incompatible with Nand Flash Controller.
	BBT	Bad block Table, which is the table recording all bad block information.
	Interleave	The are two kind of access mode when they are multiple nand flash chips on board, for example, 4 chips on board. (1) Interleave mode, which means paralleled access the nand flashes. The data is distributed into the multiple chips, each chip take one part of it equally. For example, the board has 4 chips, each chip is 2K page size. If we write 8K data to nand flash with interleave mode, the result is: the first 2K go to chip1, and the second 2K to chip 2, and last 2K goes to chip 4. So, each chip take 2K out of 8K total size. During interleave mode programming, the 4 chips can do programming at the same time, which improve the write performance. (2) None-interleave mode, which means exclusively access the nand flashes. Every time, only one of the multiple chips on board can be accessed at one time. When programming, the data will go to the selected nand flash not all the nand flashes on board compared with interleave mode. Note: Only I.MX 51 supports Interleaved mode.
	LBA	Logic Block Address, which indicates that the input address is a logic block address if it is checked. This option is only used on wince platform, for Linux, it should be ignored.
	Read Back Check	Comparing the programmed data with original data in programming process will not be done to speed up the programming speed if this option is un-checked. The option will be visible only when programming option is selected. By default, this option is checked.
Flash Device	Flash Model	Click the down arrow to select the Flash type for the associated board when the Program, Dump, or Erase Flash operations are selected: NOR (SPansion) for the i.MX31/i.MX27/i.MX32/i.MX35 boards NAND for the i.MX31//MX27/i.MX32/i.MX35/i.MX37/i.MX51/i.MX25 board MMC for the i.MX31/i.MX32/i.MX35 /i.MX37/i.MX51/i.MX25/i.MX53 board SD for the i.MX31/i.MX32/i.MX35/i.MX37/i.MX51/i.Mx25/ i.MX53 boards SPI for the i.MX51 3stack board and EVK boards, i.MX53 EVK board USER INPUT for a user input library
	Flash Lib (available when USER INPUT is selected)	Device program running on the board that handles the communication with the PC host through the protocol to program/dump/erase the flash.
Operation Settings	Address	The address in the physical memory of the target board where the operation will start.
	Size	Size in bytes for erase and dump operations.
Image	Image file	For downloading operations, indicates the image to be downloaded to RAM.

Section	Option	Description
		For Flash operations, indicates the image to be downloaded to Flash. For dumping operations, indicates the file that is used to store the dumped content.
Process	Progress bar	Displays the completed percent of the selected operation.
Information Log		Displays the execution log.

Figure 4-3 displays options for programming an image to 0xA0000000 in NOR flash.

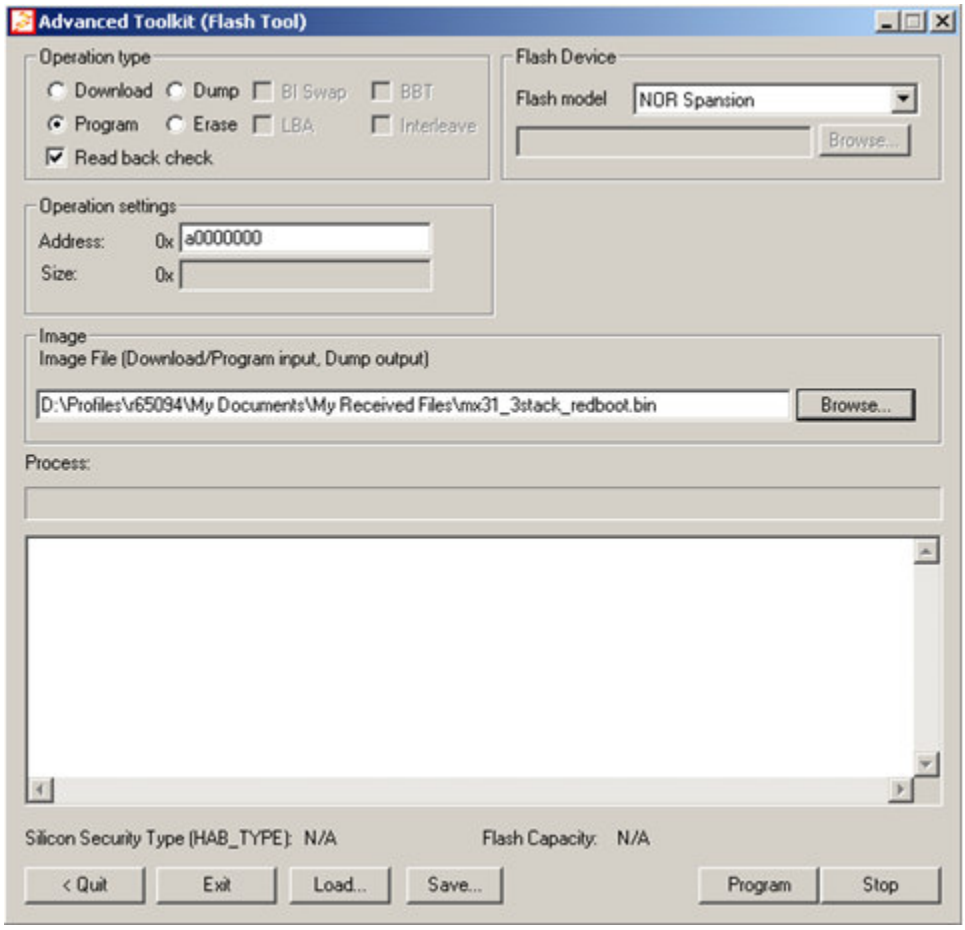

Figure 4-3 Programming NOR Flash

Figure 4-4 displays options for downloading an image to 0x80004000 in RAM.

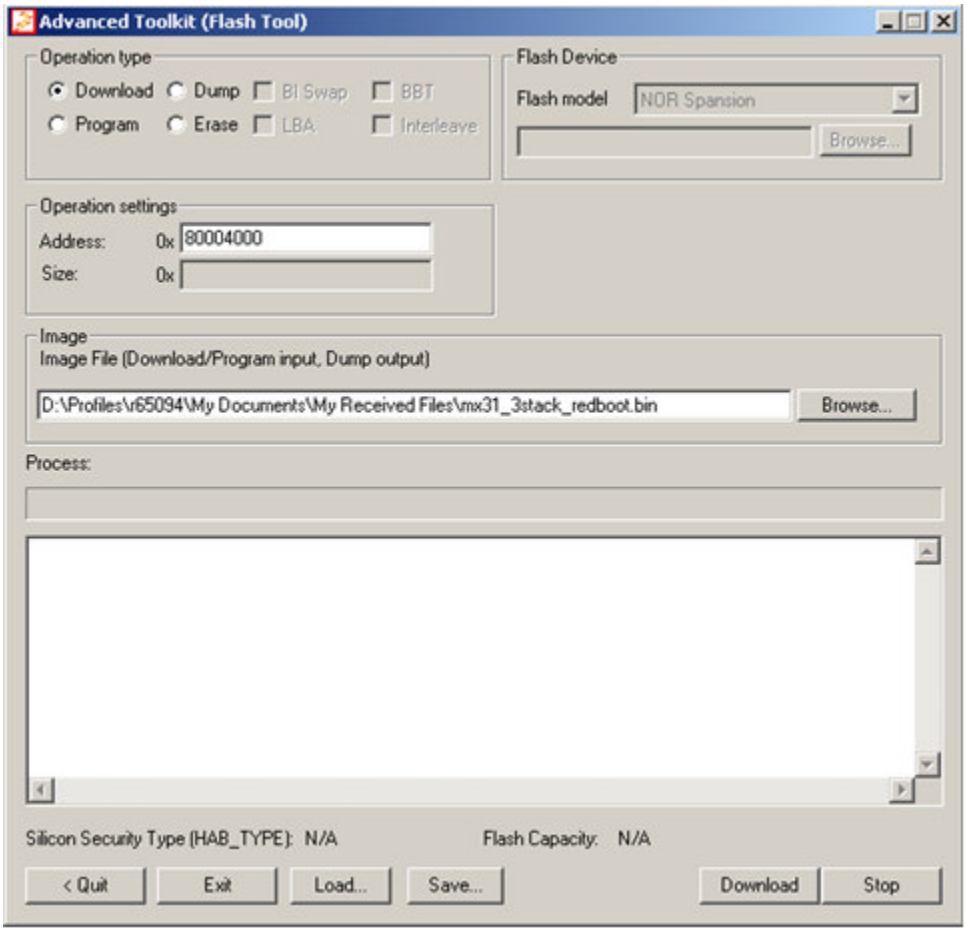

Figure 4-4 Downloading an Image to RAM

Figure 4-5 displays options for using the Flash Tool to dump from 0xA0000000 in NOR Flash with size 0x100.

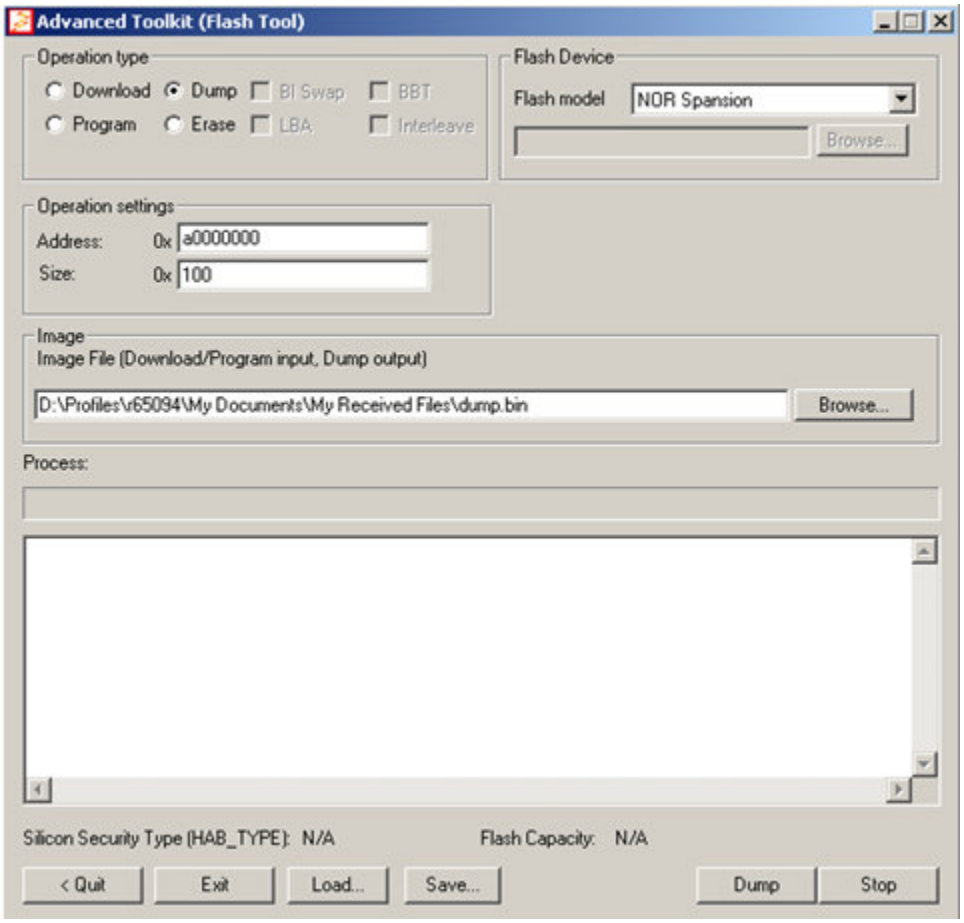

Figure 4-5 Dumping from NOR Flash

Figure 4-6 displays options for using the Flash Tool to erase 0x100 bytes from 0xa0000000 in NOR Flash.

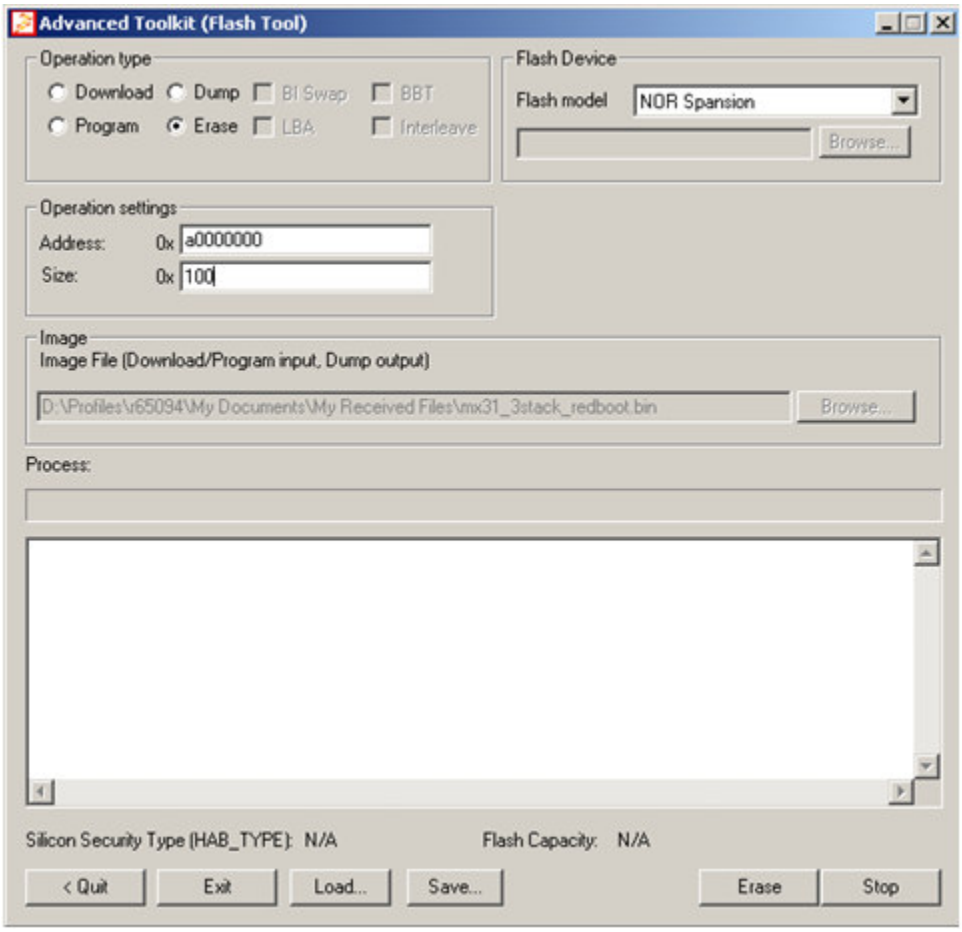

Figure 4-6 Erasing in NOR Flash

Chapter 5
Converting Image Formats

You can use the Image Format Convert Tool to convert the following image file formats: binary to S-record, S-record to binary, and ELF to binary.

Figure 5-1 displays the Advanced ToolKit selection screen.

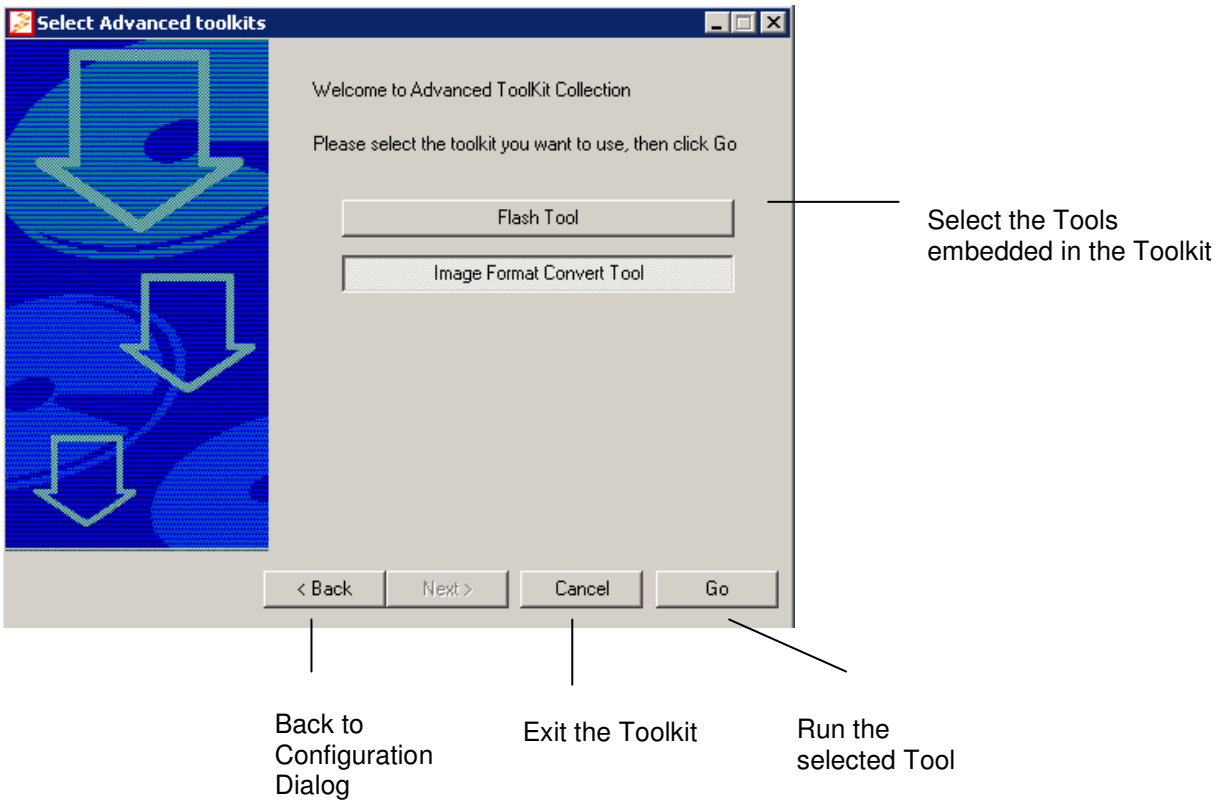

Figure 5-1 Select Advanced Tools

To use the Image Convert tool, follow these steps:

1. In the Advanced Tools screen, click **Image Format Convert Tool**, and then click **Go.**

   The Image Convert tool screen is displayed (Figure 5-2).
   - For a detailed description of the options, see Table 5-4, which follows these instructions
   - For examples of option choices, see Figure 5-3, Figure 5-4, and Figure 5-5.

.

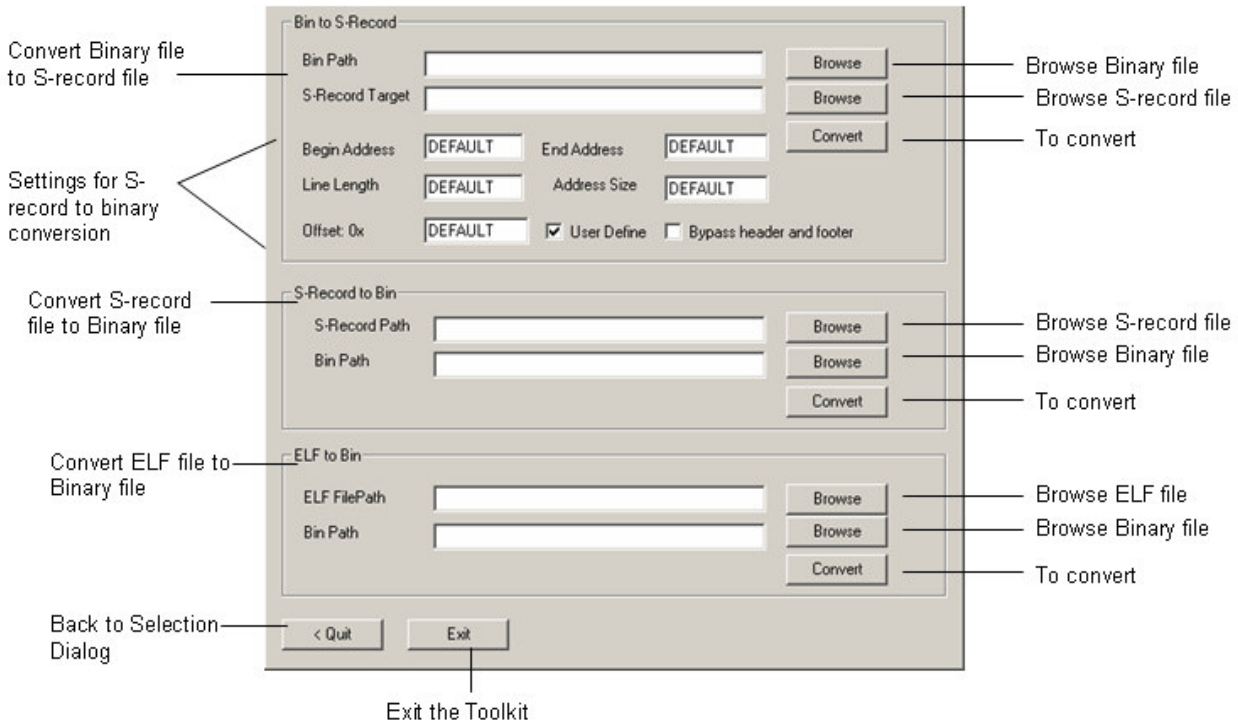

Figure 5-2 Image Conversion Tool

2. Select a file type to convert, enter the required information, and then click **Convert**.
   - For converting a **binary formatted file to an S-record format**, enter the location of the image file, the location for the converted file, address information, line length, address size, and offset parameters, and then click **Convert**.
   - For converting an **S-record formatted file to a binary file format**, enter the location of the file to convert and the location for the converted file, and then click **Convert**.
   - For converting an **ELF formatted file to a binary file format**, enter the location of the file to convert and the location for the converted file.

Table 5-1 Image Convert Tool Parameters

Item	Sub-Item	Description
Bin to S-record	Bin Path	Selects the binary file to be converted.
	S-record Target	Selects the location for the output S-record file.
	Begin Address	The beginning address of the file to be converted. The default address is 0, which means that the conversion starts from the beginning of the file.
	End Address	The end address of the file to be converted. The default End Address is the length of the file.
	Line Length	The S-record line length; the range is 8 – 32. By default, the tool calculates the length according to the address and the offset.
	Address Size	The address length in bytes; the range is 2 - 4. By default, the tool calculates the size according to the address and the offset.
	Offset	The offset of the part to be converted in the memory layout. The offset plus the beginning address become the first address in the memory layout.
	User Define	If selected, you can enter the info; if not, the tool calculates the values for you.
	Bypass header and footer	If selected, the tool will not generate the header and footer.
	Convert	Converts the binary file to the S-record file.
S-Record to Bin	S-Record Path	Selects the s-record file to be converted.
	Bin Path	Selects the output path where the output binary file will be put.
	Convert	Converts the S-record file to a binary file.
ELF to Bin	ELF File Path	Selects the ELF file to be converted.
	Bin Path	Selects the output path where the output binary file will be put.
	Convert	Converts the ELF file to a binary file.

Figure 5-3 illustrates converting a binary formatted file to an S-record format.

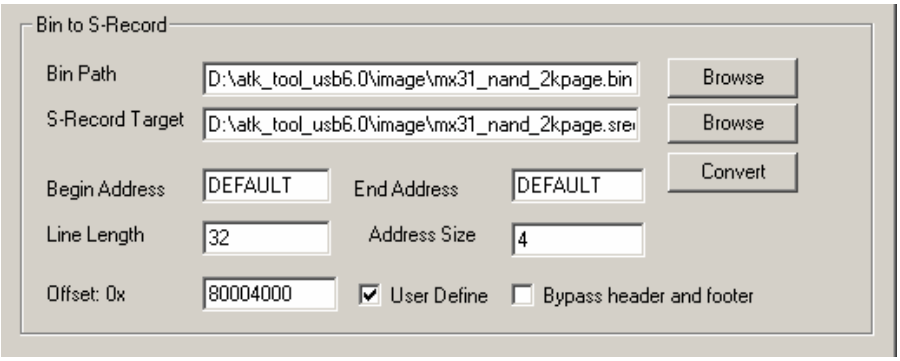

Figure 5-3 Bin to S-Record Example

Figure 5-4 illustrates converting an S-record formatted file to a binary file format.

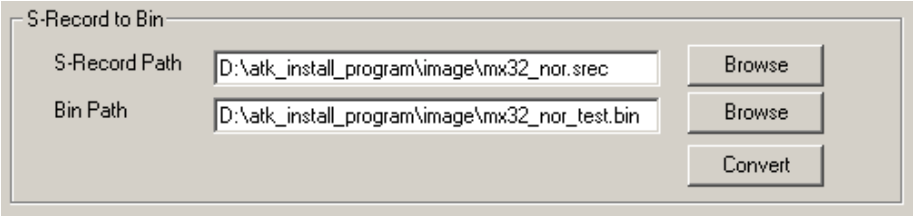

Figure 5-4 S-Record to Bin Example

Figure 5-5 illustrates converting an ELF formatted file to a binary file format.

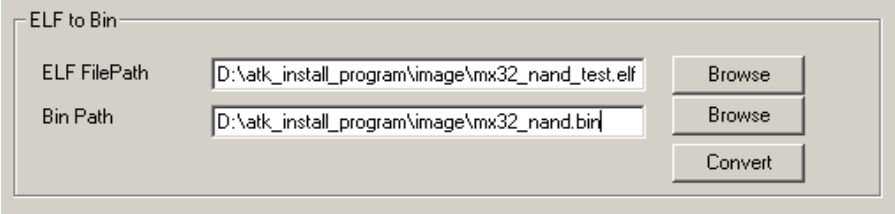

Figure 5-5 ELF to Bin Example

Appendix

Memory Init File Format

The memory initialization file is a text (.txt) file in which each line contains a Register Write operation to the MX31 device. The file columns contain information in the following format: **Address Data Format**, where:

- **Address** (hexadecimal) is the register address to be written to; it must be in hexadecimal and begin with `0x`.
- **Data** (hexadecimal) is the value written to the register address; it must be in hexadecimal and begin with `0x`.
- **Format** (decimal) indicates the data access type: 8-bit (8), 16-bit (16) or 32-bit (32).

The Address/Data/Format items are separated by a single space. End each line by pressing Enter, and the start on the next line. Use **'#'** to begin a comment line. The following example file is located in: `<tool_install_path>\example\memory_init\mx31_ddr_init.txt`.

In the first line of the example file image, the columns contain the following information: 0xB8002050 is the address; 0x0000DCF6 is the data value; and 32 indicates the format.

```
#MX31 DDR Memory init
0xB8002050 0x0000DCF6 32
0xB8002054 0x444a4541 32
0xb8002058 0x44443302 32
0xB6000000 0xCAFECAFE 32
0xb8002000 0x0000CC03 32
0xb8002004 0xa0330D01 32
0xb8002008 0x00220800 32
0xB8001010 0x00000004 32
0xB8001004 0x006ac73a 32
0xB8001000 0x92100000 32
0x80000f00 0x12344321 32
0xB8001000 0xa2100000 32
0x80000000 0x12344321 32
0x80000000 0x12344321 32
0xB8001000 0xb2100000 32
0x80000033 0xda 8
0x81000000 0xff 8
0xB8001000 0x82226080 32
0x80000000 0xDEADBEEF 32
```

Configuring ATK for High-Speed USB on MX27 with TO2

To support MX27 TO2 high-speed USB boot, you need to change HW/CPLD.

Use these steps:

1. Remove RP11 and RP12.
2. Remove R240.
3. Set “0” to bit 7 in CPLD_BCTRL3.

Where:

Bit 7 (OTG_HS_EN): USB OTG High Speed Enable, enables the USB OTH High Speed interface on the CPU.

0 = OTG High Speed Interface enabled.

1 = OTG High Speed Interface disabled.

Why and How to Check the BBT| Bi Swap in the Flash Tool

(1) Why do we need the BI swap?

The FSL NFC memory layout differs from the NAND flash data layout when it comes from the factory. The layout will cause the BI flag of the NAND flash to fail in the main area of the NFC memory. This means that the BI flag will be overwritten by the real data, which will cause the BI flag to be lost. In order to keep the BI flag, you must swap the data corresponding to the BI flag position to an unused spare area position when writing NAND flash and swap it back when it is read.

(2) There are two types of bad block management schema

a) **Scan mechanism**: In order to detect whether the block is bad, the software must scan the block to test the BI Flag value. If none are specified as 0xff, it is a bad block.

b) **Flash-based bad block table**: The bad block information for all blocks is stored in the NAND flash after the first scan out. The bad block table will be used to determine if the block is bad.

(3) How to set the BBT/Bi swap flag in the ATK

— **Windows** bad block management schema: If you are using the nb0 file format, which is the Windows Embedded CE image, select the Bi swap flag for the program/dump/erase operation. Do NOT select the BBT flag.

— **Linux** bad block management schema: Select the BBT flags for program/dump/erase operation. If the image that you programmed also uses the BI swap to maintain the NAND flash BI flag, then also select the BI swap. If you are not sure, contact the image provider.

(4) What do I do if I erased the bad block table on the NAND flash?

a) If you erased the partition that contains the BBT and the BBT flag was set, ATK will reconstruct the BBT and store it into the NAND flash through the scan mechanism.

b) If you erased the partition that contains the BBT and the BBT flag was NOT set, ATK will help you reconstruct the BBT and store it into NAND flash through the scan mechanism the next time you set the BBT flag for program/dump/erase operation.

NOTE

For Linux users, to make your work easier, we recommend that you always select the BBT flags.

If you not sure of the source or contents of the NAND flash, ATK suggests that you erase the entire NAND flash first, and then perform the other operations.