SAM-BA for the WB45NBT

WB45NBT

Application Note *v2.0*

1 Flash Programming Using the Atmel SAM-BA Utility

This section describes the process of programming the OS images into a WB45NBT module. SAM-BA (Smart ARM Microcontroller Boot Assistant) is a programming utility provided by Atmel. This utility can program flash memory that is attached to an AT91 microcontroller through either a serial port or through a USB port. It is highly recommended to avoid the serial port method because it is too slow for transferring large OS images.

1.1 Binary Image Description

The following are the four binary images that are programmed into the WB45NBT module:

- Bootstrap loader
- U-Boot boot loader
- Linux kernel
- File system image

These images are programmed into four specific locations in the 128 MB flash memory of the WB45NBT module. Refer to the WB45 Reference Manual for descriptions of the locations for each binary image within flash memory. Pre-compiled binary images are available for download from Ezurio.

1.2 Preparing the Ezurio WB45NBT for Programming

If the flash memory has been previously programmed, then it is necessary to temporarily disable the flash memory by shorting out the chip select to the flash memory chip. This can be done by shorting TP23 and TP24 (Figure 1) using a pair of tweezers before power is applied and until two or three seconds after power up.

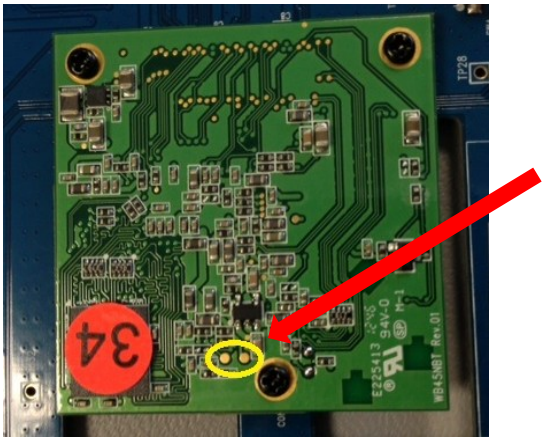

Figure 1: TP23 and TP24

By disabling the flash in this way, the AT91 processor executes its own internal ROM Boot Loader during the application of power. The ROM Boot Loader must be running in order to attach to the WB45NBT using SAM-BA. Roughly three seconds after power is applied, the short across R16 may be removed.

1.3 Installing the Atmel AT91 Driver

If this is the first time you are programming the board you will need to install the AT91 USB CDC driver for SAM-BA to communicate with it.

To install the driver, follow these steps:

1. Prepare the board for programming by shorting R16 on power up. See: Preparing the Ezurio WB45NBT for Programming section.
2. Connect the USB cable to the computer and then to the WB45NBT.
3. Install the AT91 USB CDC driver by following these steps:
    a. When the *Found New Hardware* dialog pops up and asks if Windows may connect to windows update choose ***No, not this time***. Click Next.
    b. Select ***Install from a list or specific location (Advanced)***. Click **Next**.
    c. Select ***Don't search. I will choose the driver to install***. Click **Next**.
    d. Select ***Have Disk***. Click **Next**.
    e. Select ***Browse***.
    f. Navigate to **C:\Program Files\ATMEL Corporation\SAM-BA v2.10\drv** (Figure 2).

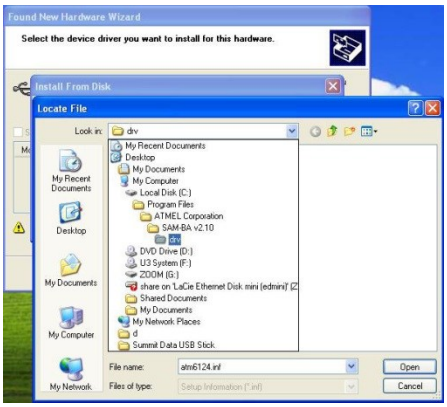

Figure 2: Installing the AT91 Driver

    g. Select the file named **atm6124_cdc.inf** (Summit's driver file) and click **Open**.
    h. Select **OK**. A dialog appears, noting the driver is not signed.
    i. Select **Continue Anyway**.
    j. Wait for the driver to install and then select **Finish**.

The driver installs itself as a virtual COM port on the PC. This COM port shows up in the device manager under Windows.

Note: Because this COM port will be needed when starting the SAM-BA utility, remember what COM port is assigned to this connection.

1.4 Software Requirements

- Operating System: Windows XP 32-bit
- Images listed in Boot Image Description section (available from Summit Data Communications)
- SAM-BA flash utility provided by Atmel. This utility is available from Atmel's web site: http://www.atmel.com/dyn/products/tools_card.asp?tool_id=3883

One modification to the SAM-BA utility configuration file is required. SAM-BA must be configured for the 16-bit hardware connection between the AT91 CPU and the SDRAM on the WB45NBT module. Because the configuration file specifies a 32-bit connection by default, this setting must be changed.

To change this setting, follow these steps:

1. Using a text editor, open the following file:

```
C:\Program Files\ATMEL Corporation\SAM-BAv2.10\tcl_lib\ at91sam9g20-ek\at91sam9g20-ek.tcl
```

2. On line 55, find the following text:

```
variable extRamDataBusWidth 32
```

3. Change the value from **32** to **16**.

4. Save and close this file.

Note: The binary images are loaded onto the PC used to program the module; this ensures that the SAM-BA utility can access them.

The SAM-BA utility is now ready to use.

1.5 Using SAM-BA

Note: To use SAM-BA, a mini-USB device cable, BB45, and power-adapter are required. Do not rely on USB power.

To use SAM-BA to program the microcontroller, follow these steps:

1. Connect the serial-console cable.
2. Connect USB device-cable to the Host/Device port on BB45 (on the corner opposite of power).
3. Power off the WB45.
4. Short the WB45 flashing-jumper points shown in Figure 3.

Figure 3: Flashing jumper pins

5. Power on (and wait 4 seconds). The serial console should show ***RomBOOT***.
6. Remove the WB45 flashing-jumper.
7. Run SAM-BA by executing the following script:

```
./sam-ba
```

The SAM-BA dialog window displays.

1. Select the appropriate connection and board (Figure 4).

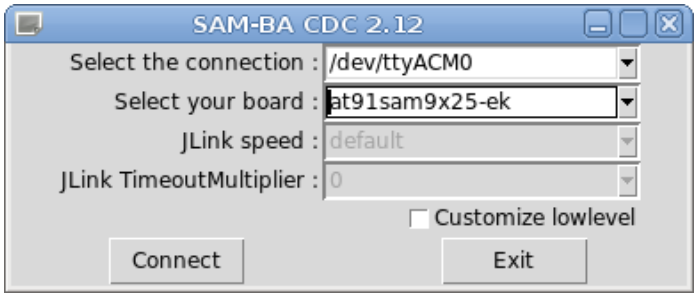

Figure 4: SAM-BA dialog window with connection and board selected

```
Connection:     /dev/ttyACM0
Board:          at91sam9x25-ek
```

2. Click **Connect**.
3. In the main SAM-BA window, select the **NAND_Flash** tab.
4. Execute the following script:

```
Enable NAND Flash
```

5. Verify that PMECC is enabled and set up correctly by doing the following:
   a. Execute the following script:

```
pmecc configuration
```

   b. Ensure that Ecc offset is set to 36 (Figure 5).

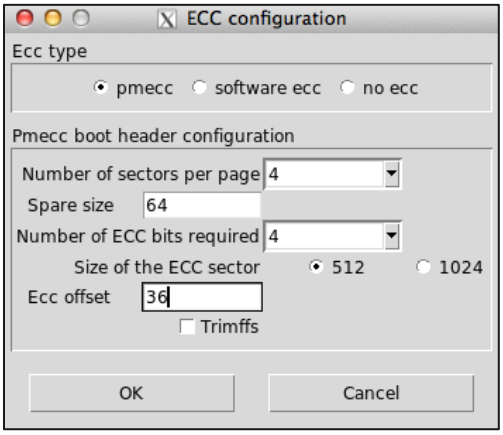

Figure 5: ECC configuration window

6. Execute the following scripts:

```
Erase All

List Bad Blocks
```

Note: No bad blocks should display.

7. To flash bootstrap/u-boot/kernel, do the following three steps identified in Figure 6:

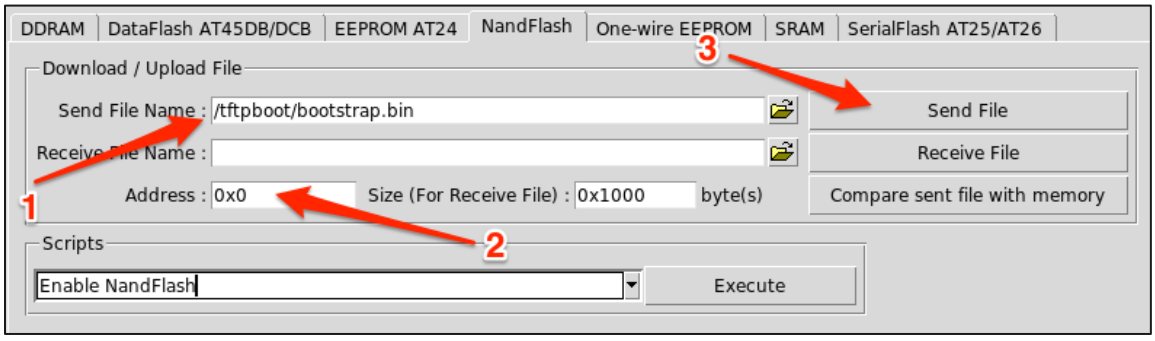

Figure 6: Flashing bootstrap, u-boot, and kernel

1 Use the Send File Name search option to locate the applicable .bin file.

bootstrap:	**bootstrap.bin**
u-boot:	**u-boot.bin**
kernel:	**kernel.bin**

2 Set the Address appropriately.

bootstrap:	**0x0**
u-boot:	**0x20000**
kernel:	**0xe0000**

3 Click **Send File**.

8. Change PMECC to enable Trimffs by doing the following (Figure 7):

   Execute the following script: **pmecc configuration**

   Select the **Trimffs** check box.

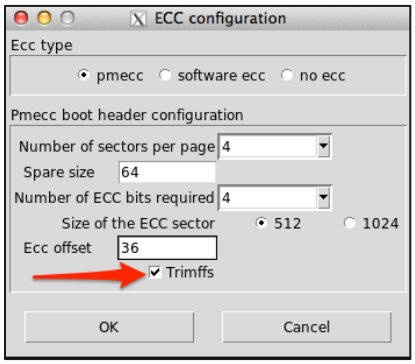

Figure 7: Enabling Trimffs

9. Flash rootfs by doing the following:

   a. Use Send File Name and select the **rootfs.bin** file.
   b. Set the address to **0x5e0000**.
   c. Click **Send**.

10. Exit SAM-BA.
    File: `Quit`
11. Disconnect the USB device-cable.
12. Power the device off.

Notes: Sometimes, operations in the main-window-gui appear to hang after a while.
A RAM initialize error message usually because at power-on, need to wait a bit.
The rootfs uses 'trimffs'.

---

2 Revision History

Revision	Date	Description	Contributor(s)	Approver
1.0	19 Aug 2013	Initial Release		John Imboden
1.2	21 Aug 2013	Grammatical edits. **Note:** v1.1 was unreleased		Dave Drogowski
1.4	26 Aug 2013	Significant additions (installation and preparation). **Note:** v1.3 was unchanged.		Sue White
1.5	19 Nov 2014	Fixed Cross Reference		Sue White
1.6	19 Oct 2015	Fixed links		Sue White
2.0	15 May 2025	Ezurio rebranding	Sue White	Dave Drogowski

Ezurio's products are subject to standard Terms & Conditions.