

# Flash Programming using the Atmel SAM-BA Utility

## WB50NBT

*Application Note*

*v1.0*

### INTRODUCTION

SAM-BA (Smart ARM Microcontroller Boot Assistant) is a programming utility provided by Atmel. This utility can program flash memory that is attached to the AT91 microcontroller through a USB port.

The goal of this document is to describe the process of programming an OS image to the WB50NBT.

### OVERVIEW

The following are the two images that may be flashed to the WB50NBT module:

- Bootstrap loader
- U-Boot boot loader

These images are programmed into specific locations in the flash memory of the WB50NBT module. Refer to the WB50 Reference Manual for descriptions of the locations for each binary image within flash memory. Pre-compiled binary images are available for download from Laird.

### REQUIREMENTS

To use SAM-BA with the WB50NBT, you will need the following:

- BB50NBT development board
- Serial-console cable
- USB-A to USB-B cable
- Tweezers for shorting module pins
- SAM-BA for the WB50 utility  
(available on the Software Downloads tab of the [WB50NBT product page](#))
- Images of Bootstrap loader and U-Boot boot loader for the WB50NBT, available from Laird

### USING SAM-BA

**Note:** To use SAM-BA, connect the BB50NBT to your PC via USB-A to USB-B cable. Use the DC power adapter provided with the BB50NBT. Do not rely on USB power.

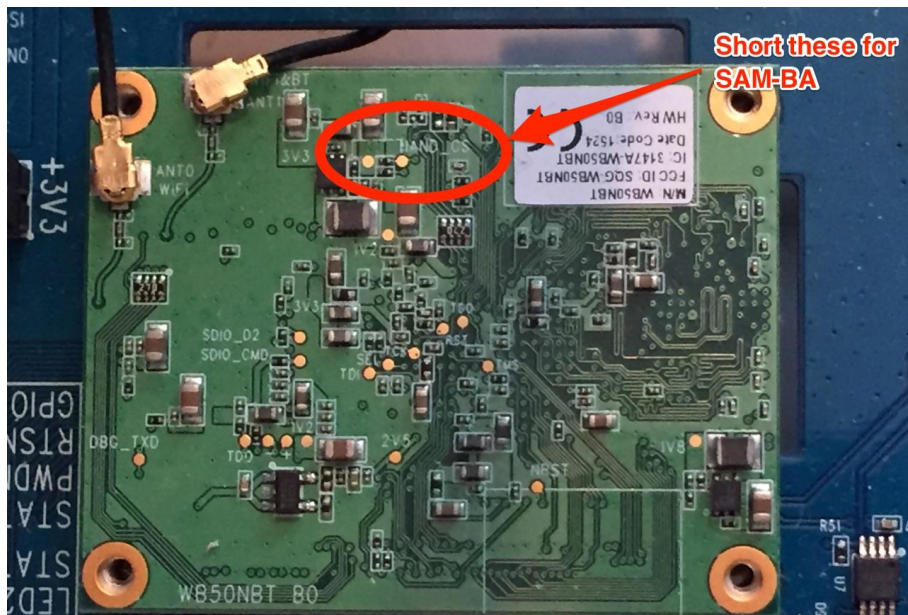
These instructions apply to the WB50NBT with 8-bit ECC.

Before beginning, note your Ethernet MAC address so that you can restore it after the flash process.

To use SAM-BA to flash images to the AT91, follow these steps:

1. Power off the WB50NBT.
2. Connect the serial-console cable to the DEBUG port on the BB50NBT development board.
3. Connect the USB device cable to the USB Device port (type B plug) on the BB50NBT (located on the corner opposite of power).
4. Short the WB50 flash memory jumper points (TP23 and TP24) as shown in [Figure 1](#).

**Note:** Disabling the flash in this way allows the AT91 to execute its own internal ROM bootloader when powered. The ROM bootloader must be running in order to attach to the WB50NBT using SAM-BA.



**Figure 1: TP23 and TP24 pins for SAM-BA preparation**

5. Power on the WB50NBT.
6. After power has been applied for about three seconds, the short may be removed.
7. The serial console should show *RomBOOT*. Run SAM-BA by executing the following script:

```
./sam-ba
```

8. The SAM-BA dialogue window displays. Select the appropriate connection and board as shown in [Figure 2](#).

Connection: /dev/ttyACM0

Board: at91sama5d3x-wb50nbt

**Note:** If you already have another ACM0 device, the BB50NBT may show as ACM1 or higher.

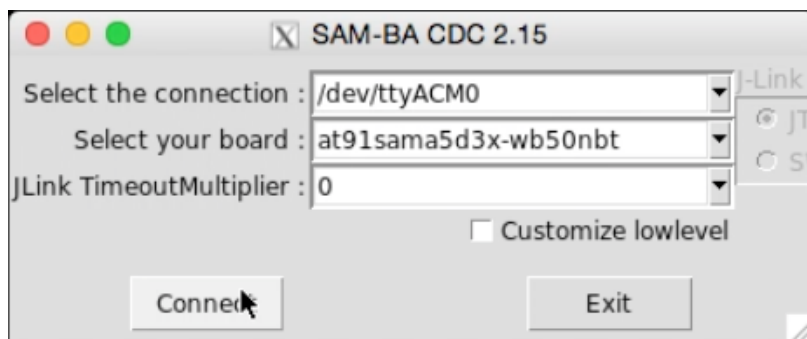


Figure 2: Selecting the correct connection and board in the SAM-BA utility

9. Click **Connect**.
10. In the main SAM-BA window, select the **NandFlash** tab.
11. Execute the following script:  

```
Enable NandFlash
```
12. Verify that PMECC is enabled and set up correctly by doing the following:

- a. Execute the following script:  

```
pmecc configuration
```

- b. Ensure that the configuration is set as shown in [Figure 3](#).

ECC bits required: 8

ECC offset: 12

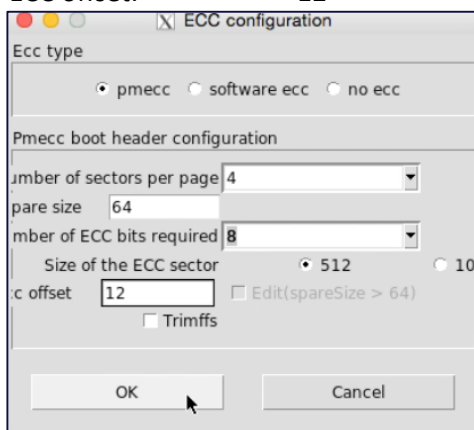


Figure 3: ECC configuration window

13. Execute the following scripts:

```
Erase All
List Bad Blocks
```

**Note:** It is typical for a small number of bad blocks (usually less than ten) to be detected in this process. The maximum supported number of bad blocks is twenty; if your device shows near or more than twenty bad blocks, please contact Laird support.

14. To flash bootstrap and U-Boot, do the following three steps identified in Figure 4.

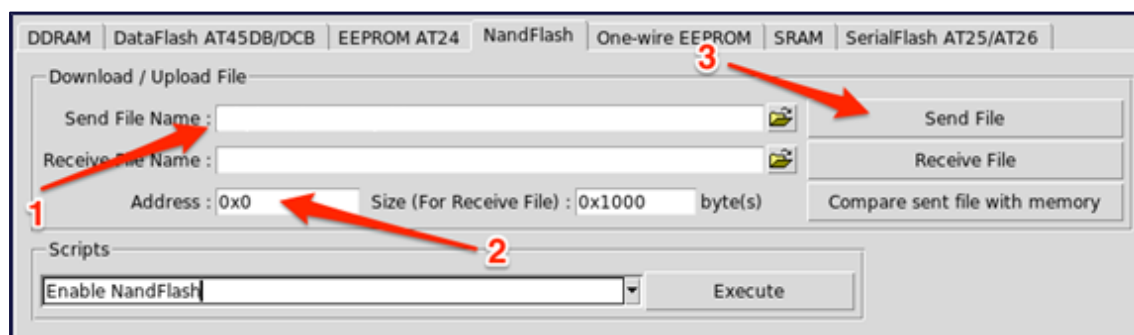


Figure 4: Flashing bootstrap and U-Boot

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

```
bootstrap:    at91bs.bin
u-boot:       u-boot.bin
```

- 2 Set the Address appropriately.

```
bootstrap:    0x0
u-boot:       0x20000
```

- 3 Click **Send File**.

15. Exit SAM-BA.  
16. Disconnect the USB device cable.  
17. Power the device off.

## REVISION HISTORY

Version	Date	Notes	Approver
1.0	6 April 2016	Initial Release	Steve deRosier