
NPCM7xx Evaluation Board (EVB) BootBlock, u-boot and OpenBMC Programming Guide

March 2018
Revision 1.0

REVISION RECORD

REVISION	RELEASE DATE	SUMMARY OF CHANGES
1.0	June 2018	First revision

PREFACE

This Evaluation Board Programming Guide is intended for board designers using the Nuvoton NCTM750x.

The information contained in this document is for reference only and is subject to change without prior notice. No part of this document may be reproduced in any form without the prior written consent of Nuvoton Technology Corporation.

CONTENTS

Table of Contents

CHAPTER 2 OVERVIEW	1
2.1 Requirements	1
CHAPTER 3 PROGRAMMING BOOTBLOCK AND U-BOOT	2
3.1 Conectors, Switches and Jumpers on the EVB Board.....	2
3.2 Programming Procedure	2
CHAPTER 4 OPENBMC PROGRAMMING	7
4.1 OpenBMC Programming From SD card or USB key to SPI Flash Device	7
4.1.1 USB Example Log.....	9
4.2 Program the openBMC image from the Host machine via TFTP server	10
4.2.1 FTP Example Log	11

Chapter 1 Overview

This guide describes how to program the BootBlock + u-boot and/or The entire OpenBMC image on the NPCM7xx Evaluation Board (EVB).

1.1 Requirements

- For BootBlock and u-boot programming:
 - Programming script package file, '**Image_Generation_and_Programming_x.y.w.zip**'. (**x.y.w** represents the version number, e.g., 1.8.7)
- For OpenBMC programming:
 - Linux files: "**image-bmc**"
 - Environment file: "**uboot_env_parameters.txt**"

Note: Check with Nuvoton support for the most recent versions of the BootBlock, u-boot and OpenBMC.

Chapter 2 Programming BootBlock and u-boot

Note: if u-boot is already running on your EVB you can skip to [OpenBMC Programming](#)

2.1 Conectors, Switches and Jumpers on the EVB Board

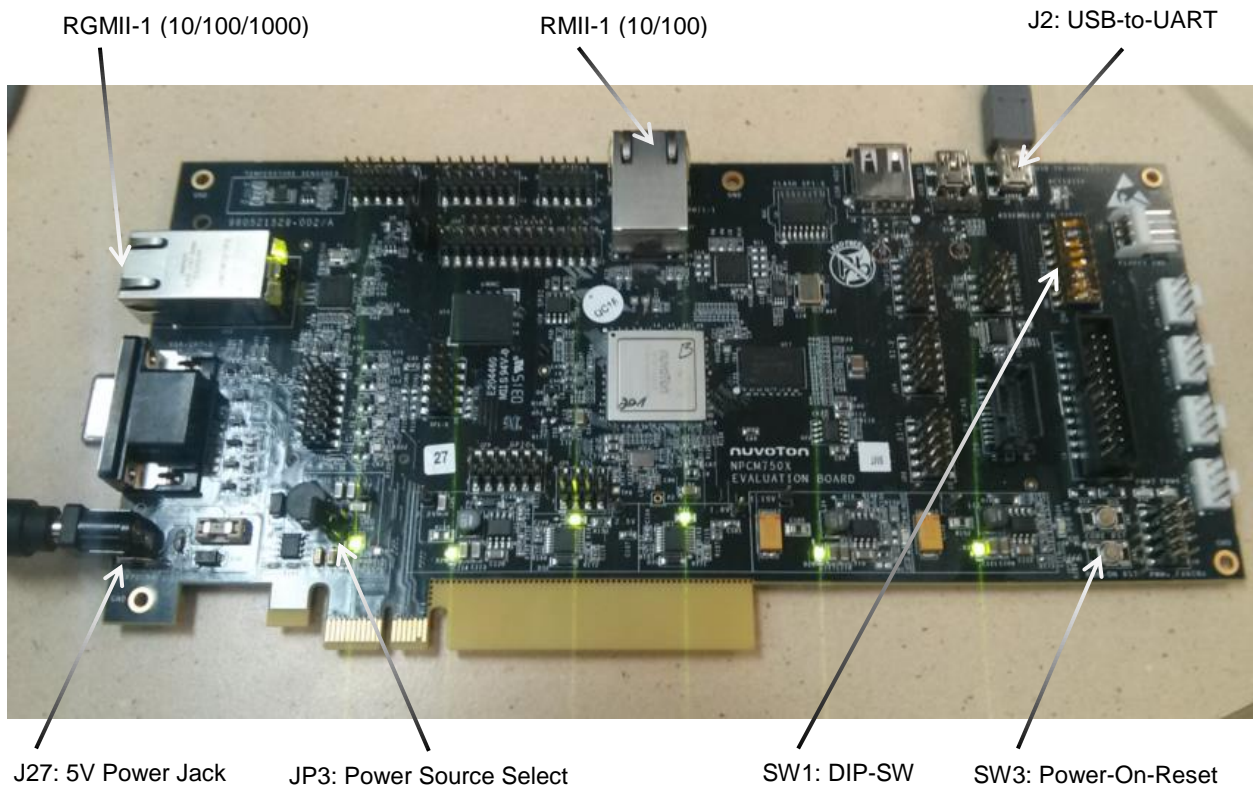


Figure 1 - Connectors, Switches and Jumpers on the EVB Board

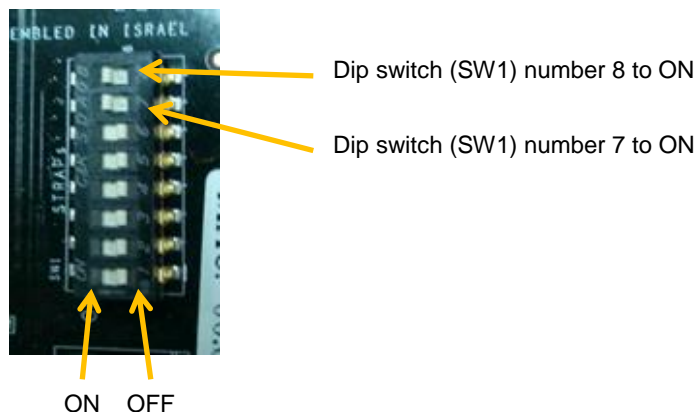
2.2 Programming Procedure

This procedure refers to Figure 1, above.

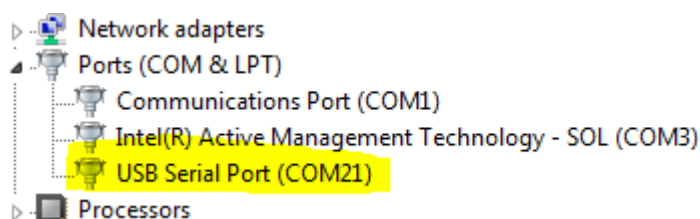
1. Hardware Setup:

- a. Verify that jumper JP3 (power source selector) is in the 1-2 position (to select the 5V power jack).
- b. Connect the 5V power supply to power jack J27. The power supply should be 5V with at least 2.5A; the jack should be 2.5 mm in diameter (with '+' in the center core).

- c. Set dip switch (SW1) number 7 and 8 to the ON position and all other switches to the OFF position. This places the NPCM7xx into Flash UART Programming (FUP) mode using Serial Interface 2 (SI2) on the next Power-Up reset. In the NPCM7xx EVB, SI2 is connected, via the FTDI USB-to-UART IC, to mini-USB connector J2.



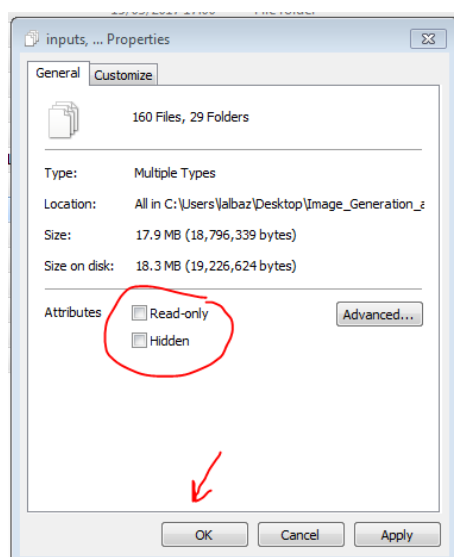
- d. Connect a mini-USB cable between the PC host and J2. J2 is connected to a USB-to-UART device, which is itself connected to SI2.
- e. Wait for the FTDI driver to be installed automatically. The COM port number is assigned automatically. If the FTDI driver does not get installed automatically, download and install the USB-to-UART driver from: <http://www.ftdichip.com/Drivers/VCP.htm>.



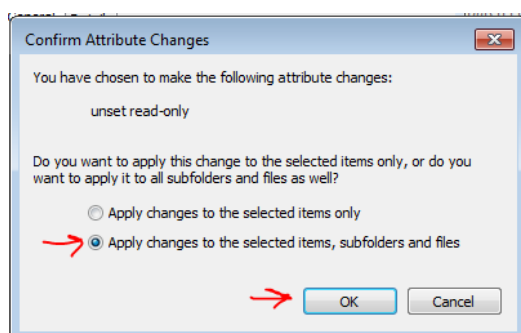
- f. Press and release the PWR-ON-RST (SW3) button to issue a Power-Up reset.
2. Prepare the Programming Tool:
 - a. Unzip the **Image_Generation_and_Programming_x.y.w.zip** file to a local directory.
 - b. Select all files in the **Image_Generation_and_Programming_x.y.w** folder, right click the mouse and select '**Properties**'.

Name	Date modified	Type	Size
docs	15/03/2017 17:00	File folder	
inputs	15/03/2017 17:00	File folder	
inputs_reference	15/03/2017 17:00	File folder	
output_binaries	15/03/2017 17:00	File folder	
programming_tools	15/03/2017 17:00	File folder	
tools	15/03/2017 17:00	File folder	
versions	15/03/2017 17:00	File folder	
FormatFlash.bat	09/01/2017 08:07	Windows Batch File	1 KB
Generating_Images.bat	09/01/2017 08:07	Windows Batch File	5 KB
ImageGenerationAndProgrammingScripts_RelLetter_1.5.7....	09/01/2017 08:07	TXT File	7 KB
ProgramAll_Basic.bat	09/01/2017 08:07	Windows Batch File	1 KB
ProgramAll_Secure.bat	09/01/2017 08:07	Windows Batch File	1 KB
ProgramBootblock_Basic.bat	09/01/2017 08:07	Windows Batch File	1 KB
ProgramBootblock_Secure.bat	09/01/2017 08:07	Windows Batch File	1 KB
UpdateInputsBinaries_DRB.bat	09/01/2017 08:07	Windows Batch File	2 KB
UpdateInputsBinaries_EB.bat	09/01/2017 08:07	Windows Batch File	2 KB
UpdateInputsBinaries_Horizon.bat	09/01/2017 08:07	Windows Batch File	1 KB
UpdateInputsBinaries_SVB.bat	09/01/2017 08:07	Windows Batch File	2 KB

- c. Make sure the 'Read-only' Attributes check box is clear (see below) and press '**OK**' .



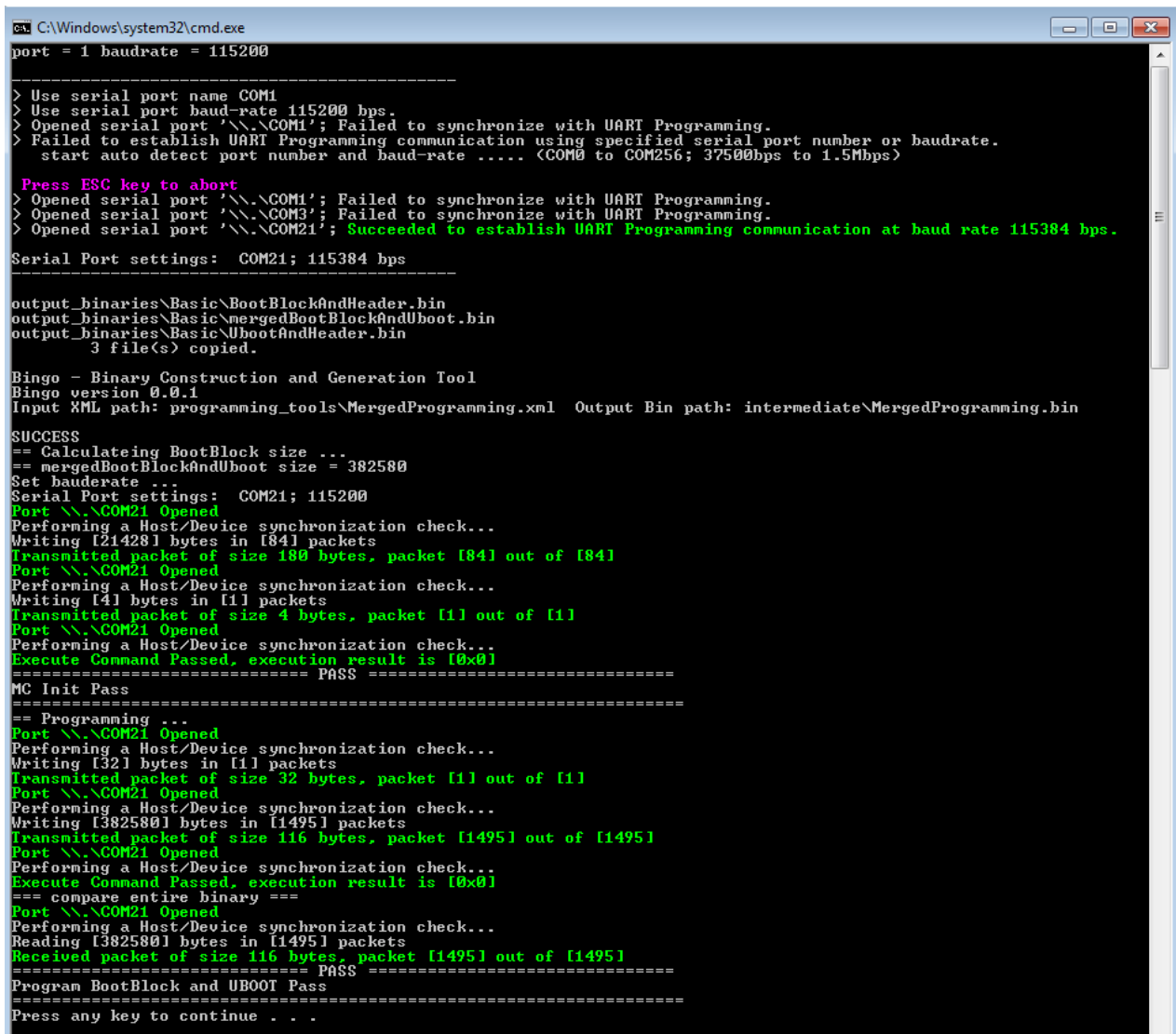
- d. Select "**Apply changes to the selected items, subfolders and files**" and press '**OK**'.



3. Programming:

Note: Steps **a** and **b** may be skipped if they were already performed for the correct platform and firmware version.

- The programming tool supports several different platforms. To set the EVB platform, run the **UpdateInputsBinaries_EB.bat** batch file. If there is no error, the console window will open and close immediately.
- Run the **Generating Images.bat** batch file. Verify that there are no errors and then press any key to close the console window.
- Close all terminals and programs associated with the COM port number assigned by the FTDI driver for SI2.
- Run the **ProgramAll_Basic.bat** batch file to start programming BootBlock and u-boot. The utility auto-detects the COM port number and starts the programming.
- Verify that the programming completes with no errors.



```
C:\Windows\system32\cmd.exe
port = 1 baudrate = 115200

-----
> Use serial port name COM1
> Use serial port baud-rate 115200 bps.
> Opened serial port '\\.\COM1'; Failed to synchronize with UART Programming.
> Failed to establish UART Programming communication using specified serial port number or baudrate.
  start auto detect port number and baud-rate ..... <COM0 to COM256; 37500bps to 1.5Mbps>

Press ESC key to abort
> Opened serial port '\\.\COM1'; Failed to synchronize with UART Programming.
> Opened serial port '\\.\COM3'; Failed to synchronize with UART Programming.
> Opened serial port '\\.\COM21'; Succeeded to establish UART Programming communication at baud rate 115384 bps.

Serial Port settings: COM21; 115384 bps
-----

output_binaries\Basic\BootBlockAndHeader.bin
output_binaries\Basic\mergedBootBlockAndUboot.bin
output_binaries\Basic\UbootAndHeader.bin
  3 file(s) copied.

Bingo - Binary Construction and Generation Tool
Bingo version 0.0.1
Input XML path: programming_tools\MergedProgramming.xml  Output Bin path: intermediate\MergedProgramming.bin

SUCCESS
== Calculating BootBlock size ...
== mergedBootBlockAndUboot size = 382580
Set baudrate ...
Serial Port settings: COM21; 115200
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Writing [21428] bytes in [84] packets
Transmitted packet of size 180 bytes, packet [84] out of [84]
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Writing [4] bytes in [1] packets
Transmitted packet of size 4 bytes, packet [1] out of [1]
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Execute Command Passed, execution result is [0x0]
===== PASS =====
MC Init Pass
=====
== Programming ...
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Writing [32] bytes in [1] packets
Transmitted packet of size 32 bytes, packet [1] out of [1]
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Writing [382580] bytes in [1495] packets
Transmitted packet of size 116 bytes, packet [1495] out of [1495]
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Execute Command Passed, execution result is [0x0]
=== compare entire binary ===
Port \\.\COM21 Opened
Performing a Host/Device synchronization check...
Reading [382580] bytes in [1495] packets
Received packet of size 116 bytes, packet [1495] out of [1495]
===== PASS =====
Program BootBlock and UBOOT Pass
=====
Press any key to continue . . .
```

4. Testing:

- a. Open a terminal (e.g., Tera Term version 4.9) and set the correct COM port number assigned by the FTDI driver (in Step 1.e).

The COM port should be configured as follows:

115 Kbps, 8 bit, 1 stop-bit, no parity no flow control.

- b. Verify that all first 7 dip switches (SW1) are set to the OFF and dip switch number 8 is set to ON, position for normal operation (800 MHz CPU and 800 MHz DDR4).
- c. Press and release the PWR-ON-RST (SW3) button to issue a Power-Up-reset.
- d. Verify that boot-block and u-boot are running and that the versions are up-to-date.

Notes:

- If, after Power-Up reset, the terminal displays only a '**Z**', dip switch (SW1) number 7 is probably at ON position. Set it to OFF position for normal operation and try again.
- Further information can be found in the '**How_to_generate_and_program_images.html**' file. (in the **Image_Generation_and_Programming_x.y.w.zip** file in the 'docs' folder).

```
>Boot block run for 1376261 us.
>Jump to uboot at 0x8000
U-Boot 201510.5.9 <Sep 14 2016 - 16:33:03 +0300> by Nuvoton Technology Corp.
HAL ver : v0.3.2
Board: PolegSUB
Chip : npcm750
Core : Cortex-A9
CPU Freq: 800MHz
Memory Freq: 800MHz
SPI0 Freq: 50MHz
SPI3 Freq: 20MHz
APB1 Freq: 25MHz
APB2 Freq: 50MHz
APB3 Freq: 50MHz
APB4 Freq: 25MHz
APB5 Freq: 100MHz
CP Freq: 200MHz
RomCode : v00.01.02.06
BootBlock : v00.10.06.03
PolegSUB Power-On Straps = 0x1ff7
Security: DISABLED, running in non-secure mode
DRAM: 464 MiB
Flash:
Reducing flash chip size from 32 MB to 16 MB
SPI_Flash0: Found CS0 dev#0 Name[W25Q256] ChipSize[0x1000000]
16 MiB
In: serial
Out: serial
Err: serial
MMC: npcmx50_sdhci: 0, npcmx50_sdhci: 1
Net: RGMII-Phy Addr: 0x1 Phy 0 not found
ETH0, ETH1, ETH2, ETH3
Hit any key to stop autoboot: 0
=>
```

Figure 2: Sample Terminal Log File

Chapter 3 OpenBMC Programming

3.1 OpenBMC Programming From SD card or USB key to SPI Flash Device

This section describes how to program the entire OpenBMC image to the flash from either an SD card or a USB storage device.

Prepare the SD card or USB storage device with a FAT file system (FAT16 or FAT32) and verify that there is at least 32 MB of free space available for the OpenBMC files.

1. Copy the "**image-bmc**" file into the root directory of either the SD card or USB storage device.
2. Plug the SD card or USB storage device into the EVB (see Figure 2 and Figure 3 below).
3. Power up the EVB and hit any key to stop at the u-boot shell.
4. On u-boot shell type those commands (the commands might already exist in your u-boot environment):

```
setenv sd_prog 'fatload mmc 0 10000000 image-bmc; cp.b 10000000 80000000  
${filesize}'  
setenv usb_prog 'usb start; fatload usb 0 10000000 image-bmc; cp.b 10000000  
80000000 ${filesize}'
```
5. If you wish to save those commands type:

```
saveenv
```
6. Type either "**run sd_prog**" (for an SD card) or "**run usb_prog**" (for USB storage) and press **<ENTER>**. This loads Linux kernel files to SDRAM and from SDRAM to SPI flash.
7. Reset the EVB to run Linux.

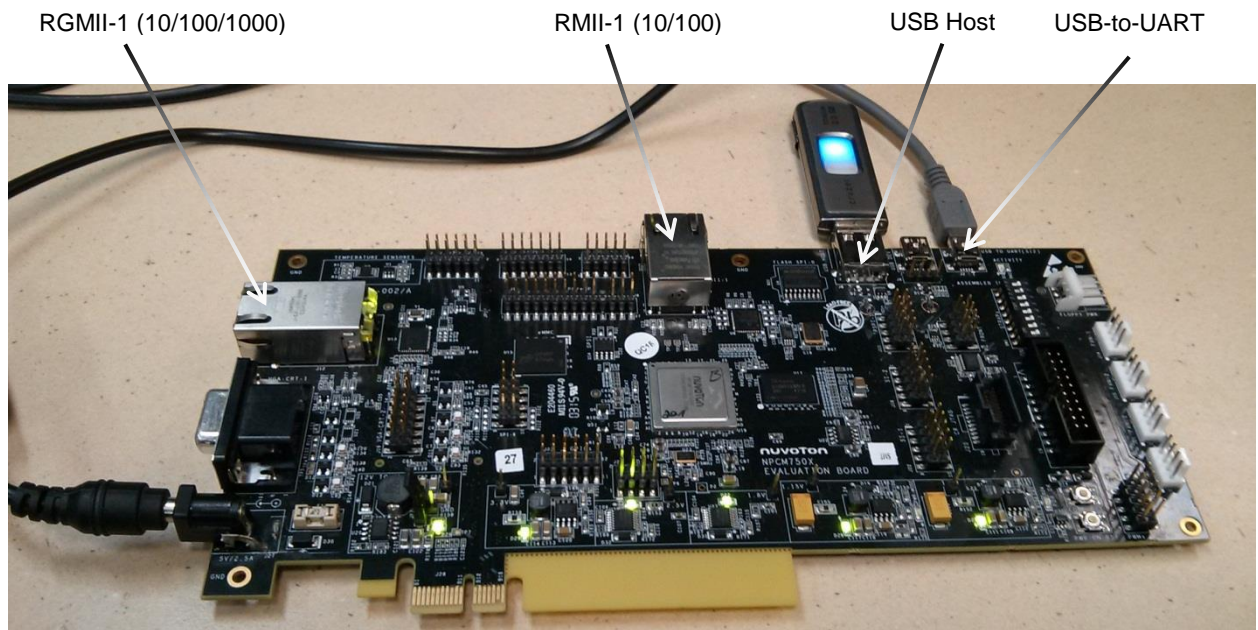


Figure 2: NPCM7xx EVB with USB Storage Device Plugged In



Figure 3: NPCM7xx EVB with SD Card Plugged In

3.1.1 USB Example Log

An example USB log is shown on the following page.

[illegible]

3.2 Program the openBMC image from the Host machine via TFTP server

This section describes how to load and boot the OpenBMC image from the TFTP server. This is very useful in the development stage when doing frequent builds.

These steps below should be performed one time per EVB.

After performing these steps, on each EVB reset, OpenBMC image is downloaded automatically from the TFTP to EVB SDRAM and then are booted.

1. Install and set up the TFTP server on your host machine
 - For Windows, use 'Tftpd32' or 'Tftpd64' from <http://tftpd32.jounin.net>
 - For Linux, set up the tftp server environment.
2. Copy the "deliverables" folder, which contains the "**image-bmc**" file, into the TFTP server root folder on the host machine.
3. Power up the EVB and hit any key to stop at the u-boot shell.
4. Open the "**uboot_env_parameters.txt**" file for editing and update the u-boot environment according to your network settings and working method. Pay attention to the following environment settings:
 - Update the MAC address (**mac_base** and **mac_offset**) as printed on your board.
 - Change the Host IP address (**gatewayip** and **serverip**) according to your network parameters
 - Change the EVB static IP address (**ipaddr**) in case the EVB uses static IP rather than dynamic IP.
 - Change the Linux file location in the FTP (**tftp_dir**).
 - Change the Ethernet connection (**eth_num**). The EVB supports an RMII-1 or RGMII-1 Ethernet connection.
5. Change the **bootcmd** environment to boot from the TFTP, as follows:
`setenv bootcmd 'run ftp_run'`
6. Copy and paste the contents of "**uboot_env_parameters.txt**" to the u-boot shell and press **<ENTER>**. This saves the u-boot environment to the flash.
7. Connect an Ethernet cable between the EVB (RMII-1 or RGMII-1, according to the **eth_num** configuration) and the host machine (the connection can be through your work network).
8. Type "**run bootcmd**" or reset the EVB, This loads the OpenBMC image file from the host machine through the TFTP server to EVB SDRAM, and performs the boot.

Note: To program the **OpenBMC** to the flash, type "**run usb_prog**" and press **<ENTER>**. This loads the OpenBMC image file from the TFTP to SDRAM and programs them into the flash device. Remember to update the **bootcmd** environment back to 'run **romboot**' and save the environment to the SPI flash device in order to be able to boot from the flash device after an EVB reset.

3.2.1 FTP Example Log

In this example log, we set up the TFTP server on host IP 10.191.10.155. The network supports DHCP. The following environments were changed:

- **mac_base** - changed to 00:00:F7:A0:01:38; **mac_offset** changed to 0138 as printed on the EVB (see picture below).



- **gatewayip** - changed to 10.191.20.254
- **serverip** - changed to 10.191.10.155
- **tftp_dir** - changed to “poleg/deliverables”

```
U-Boot 2015.10.7.0-00003-gb95e424a80 (Jun 08 2018 - 17:21:14 +0300)
Nuvoton Technology Corp.
```

```
HAL ver : v0.3.2
```

```
Board: PolegSVB
Chip : npc750
Core : Cortex-A9
```

```
CPU Freq: 800MHz
Memory Freq: 800MHz
SPI0 Freq: 50MHz
SPI3 Freq: 20MHz
APB1 Freq: 25MHz
APB2 Freq: 50MHz
APB3 Freq: 50MHz
APB4 Freq: 25MHz
APB5 Freq: 100MHz
CP Freq: 200MHz
```

```
RomCode : v00.01.02.06
BootBlock : v00.10.08.07
```

```
PolegSVB Power-On Straps = 0x1bf7
Security: DISABLED, running in non-secure mode
Model: Nuvoton npc750 Development Board (Device Tree)
```

```
DRAM: 464 MiB
Flash:
SPI_Flash0: Found CS0 dev#0 Name[MX66L512] ChipSize[0x4000000]
64 MiB
In: serial
Out: serial
Err: serial
MMC: npcmx50_sdhci: 0, npcmx50_sdhci: 1
Net: RGMII-Phy Addr: 0x1 Phy 0 not found
ETH0, ETH1, ETH2, ETH3
Hit any key to stop autoboot: 0
=> run ftp_run
Speed: 1000, full duplex
BOOTP broadcast 1
```

```

DHCP client bound to address 10.191.20.94 (2 ms)
Speed: 1000, full duplex
Using ETH2 device
TFTP from server 10.191.10.146; our IP address is 10.191.20.94; sending through gateway 10.191.20.254
Filename './image-bmc'.
Load address: 0x10000000
Loading: #####
#####
#####
#####
#####
#####
#####
3.4 MiB/s

done
Bytes transferred = 33554432 (2000000 hex)
## Loading kernel from FIT Image at 10200000 ...
Using 'conf@nuvoton-npcm750-evb.dtb' configuration
Verifying Hash Integrity ... OK
Trying 'kernel@1' kernel subimage
Description: Linux kernel
Type: Kernel Image
Compression: uncompressed
Data Start: 0x1020012c
Data Size: 3773448 Bytes = 3.6 MiB
Architecture: ARM
OS: Linux
Load Address: 0x00008000
Entry Point: 0x00008000
Hash algo: sha1
Hash value: ddcaccf733880d122a976e235229ccd6fb24d4d
Verifying Hash Integrity ... sha1 calc hash addr 0x1020012c size 0x399408
=>UBOOT last command run for 76648 us.
+ OK
## Loading ramdisk from FIT Image at 10200000 ...
Using 'conf@nuvoton-npcm750-evb.dtb' configuration
Trying 'ramdisk@1' ramdisk subimage
Description: obmc-phosphor-initramfs
Type: RAMDisk Image
Compression: lzma compressed
Data Start: 0x105a67d8
Data Size: 1844883 Bytes = 1.8 MiB
Architecture: ARM
OS: Linux
Load Address: unavailable
Entry Point: unavailable
Hash algo: sha1
Hash value: 11c9a7c95faa5afca5cceae1d318999a2091e1ac
Verifying Hash Integrity ... sha1 calc hash addr 0x105a67d8 size 0x1c2693
=>UBOOT last command run for 39692 us.
+ OK
## Loading fdt from FIT Image at 10200000 ...
Using 'conf@nuvoton-npcm750-evb.dtb' configuration
Trying 'fdt@nuvoton-npcm750-evb.dtb' fdt subimage
Description: Flattened Device Tree blob
Type: Flat Device Tree
Compression: uncompressed
Data Start: 0x1059963c
Data Size: 53468 Bytes = 52.2 KiB
Architecture: ARM
Hash algo: sha1
Hash value: b356e69630b97b8e209344c206df4dfb65e5c0a6
Verifying Hash Integrity ... sha1 calc hash addr 0x1059963c size 0xd0dc
=>UBOOT last command run for 4551 us.
+ OK
Booting using the fdt blob at 0x1059963c
Loading Kernel Image ... OK
Loading Ramdisk to 1cc14000, end 1cdd6693 ... OK
Loading Device Tree to 1cc03000, end 1cc130db ... OK

Starting kernel ...

```


*Nuvoton provides comprehensive service and support.
For product information and technical assistance, contact the nearest Nuvoton center.*

Headquarters

No. 4, Creation Rd. 3
Science-Based Industrial Park
Hsinchu, Taiwan, R.O.C
TEL: 886-3-5770066
FAX: 886-3-5665577
<http://www.nuvoton.com.tw>

Taipei Office

1F, No.192, Jingye 1st Rd.
Zhongshan District, Taipei, 104
Taiwan, R.O.C.
TEL: 886-2-2658-8066
FAX: 886-2-8751-3579

**Nuvoton Technology
Corporation America**

2727 North First Street
San Jose, CA 95134, U.S.A.
TEL: 1-408-5441718
FAX: 1-408-5441787

**Winbond Electronics
Corporation Japan**

NO. 2 Ueno-Bldg., 7-18, 3-chome
Shinyokohama Kohoku-ku
Yokohama, 222-0033
TEL: 81-45-4781881
FAX: 81-45-4781800

**Nuvoton Technology
(Shanghai) Ltd.**

27F, 2299 Yan An W. Rd.
Shanghai, 200336 China
TEL: 86-21-62365999
FAX: 86-21-62365998

Nuvoton Technology (H.K.) Ltd.

Unit 9-15, 22F, Millennium City 2
378 Kwun Tong Rd.
Kowloon, Hong Kong
TEL: 852-27513100
FAX: 852-27552064

For Advanced PC Product Line information contact: APC.Support@nuvoton.com

© 2018 Nuvoton Technology Corporation. All rights reserved

www.nuvoton.com