High speed firmware downloader

This application demos download firmware with high speed UART.

Table of Contents

1 How it works 3
2 Procedure 3
2.1 Hardware Setting 3
2.2 Support UART ports 4
2.3 Prepare tool 4
2.4 DAP firmware upgrade 4
2.5 Start firmware download 5
2.5.1 Windows form mode 5
2.5.2 Command line mode 14
3 Check debug message 19
4 Trouble Shooting 19

1 How it works

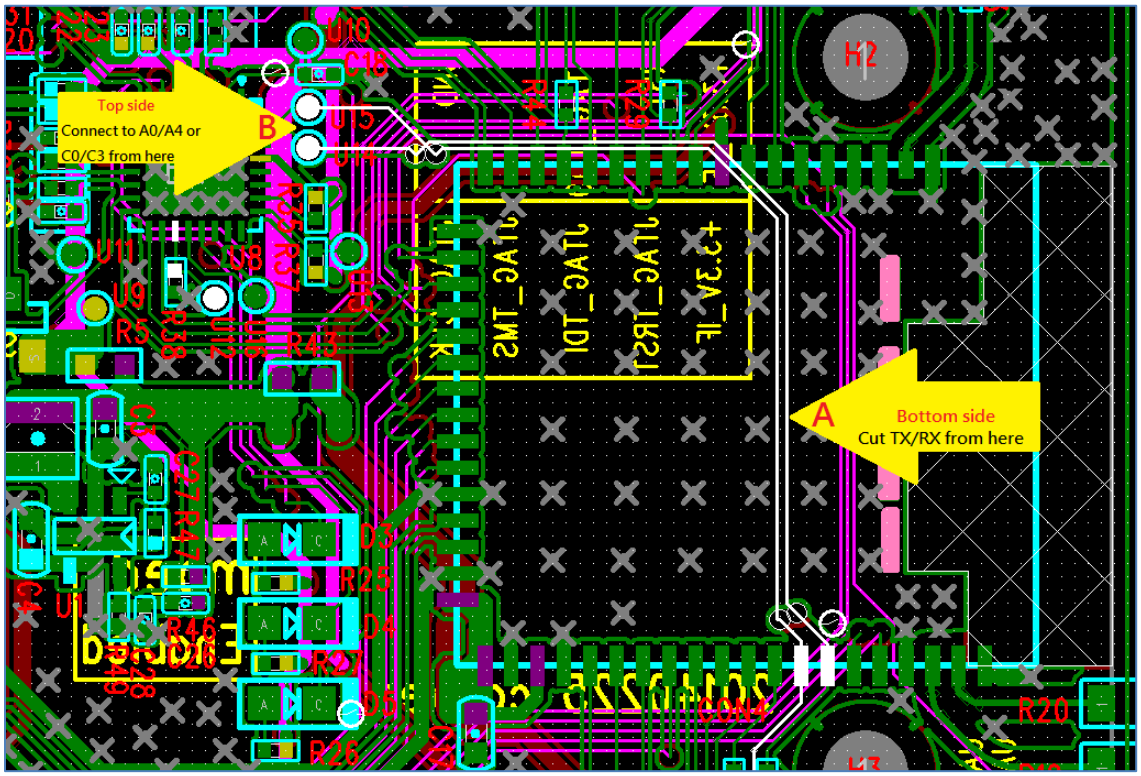
The tool downloads 40KB code from SWD interface to Ameba SRAM, and this 40KB code handles firmware downloading from high speed UART to Ameba flash.

2 Procedure

2.1 Hardware Setting

Take RTK Ameba EVB (RTL8711AF) as an example.

- Cut off DAP UART_TX, UART_RX from **A point** shown below.
- Connect **B point** DAP UART_TX to GPIOC_3 and DAP UART_RX to GPIOC_0, if UART0 is used for X-Modem transfer.
- Connect **B point** DAP UART_TX to GPIOA_4 and DAP UART_RX to GPIOA_0, if UART2 is used for X-Modem transfer.
- Image firmware downloader default uses GPIOC_3 (TX), GPIOC_0 (RX) for X-Modem transfer.



Ameba RTL8711AF PCB diagram

2.2 Support UART ports

The following is RTL8711AF pin function group table:

PIN name	JTAG	SDIO	UART Group	I2C Group	SPI Group	I2S Group	PCM Group	WL_LED	PWM	ETE	WKDT	GPIO INT	Default State	SCHMT
GPIOA_0		SD_D2	UART2_IN		SPI1_MISO							GPIO_INT	PH	O
GPIOA_1		SD_D3	UART2_CTS		SPI1_MOSI							GPIO_INT	HI	
GPIOA_2		SD_CMD	UART2_RTS		SPI1_CLK								PH	O
GPIOA_3		SD_CLK											PH	O
GPIOA_4		SD_D0	UART2_OUT		SPI1_CS								PH	
GPIOA_5		SD_D1									D_SBY0		PH	
GPIOB_0			UART_LOG_OUT							ETE0			HI	
GPIOB_1			UART_LOG_IN					WL_LED0		ETE1	D_SLP0		PH	
GPIOB_2				I2C3_SCL						ETE2			HI	O
GPIOB_3				I2C3_SDA						ETE3		GPIO_INT	PH	
GPIOC_0			UART0_IN		SPI0_CS0	I2S1_WS	PCM1_SYNC		PWM0	ETE0			HI	
GPIOC_1			UART0_CTS		SPI0_CLK	I2S1_CLK	PCM1_CLK		PWM1	ETE1		GPIO_INT	HI	O
GPIOC_2			UART0_RTS		SPI0_MOSI	I2S1_SD_TX	PCM1_OUT		PWM2	ETE2			HI	
GPIOC_3			UART0_OUT		SPI0_MISO	I2S1_MCK	PCM1_IN		PWM3	ETE3		GPIO_INT	HI	O
GPIOC_4				I2C1_SDA	SPI0_CS1	I2S1_SD_RX						GPIO_INT	HI	
GPIOC_5				I2C1_SCL	SPI0_CS2							GPIO_INT	HI	O
GPIOE_0	JTAG_TRST		UART0_OUT	I2C2_SCL	SPI0_CS0		PCM0_SYNC		PWM0				PH	O
GPIOE_1	JTAG_TDI		UART0_RTS	I2C2_SDA	SPI0_CLK		PCM0_CLK		PWM1			GPIO_INT	PH	O
GPIOE_2	JTAG_TDO		UART0_CTS	I2C3_SCL	SPI0_MOSI		PCM0_OUT		PWM2			GPIO_INT	PH	O
GPIOE_3	JTAG_TMS		UART0_IN	I2C3_SDA	SPI0_MISO		PCM0_IN		PWM3		D_SBY3	GPIO_INT	PH	O
GPIOE_4	JTAG_CLK				SPI0_CS1								PH	O

UART2 (GPIOA_0/GPIOA_4) and UART0 (GPIOC_0/GPIOC_3) can be used.

UART0 (GPIOE_0/GPIOE_3) share pin with JTAG, and the operation of JTAG may cause interference to UART operation, so this UART setting cannot be used.

2.3 Prepare tool

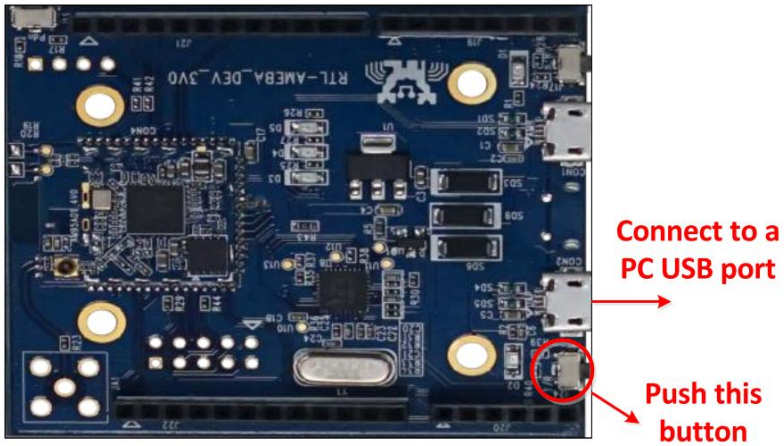
Get “ameba1-mp-fw-download-cpp-vx.zip” and unzip it.

2.4 DAP firmware upgrade

Upgrade CMSIS DAP firmware as attached in the “DAP_FW_Ameba_Vx_xx-boot-from-sram.bin”.

i. Disconnect the Ameba EVB from PC.

ii. Keep pushing the DAP FW Update button and then re-connect the Ameba EVB to the PC USB port. The Ameba EVB will be emulated as a USB disk with label “CRP DISABLD”.



iii. Delete the “firmware.bin” from the disk “CRP DISABLD”.

iv. Copy the “DAP_FW_Ameba_Vx_xx-boot-from-sram.bin” to the disk “CRP DISABLD”.

v. Power off and power on the Ameba EVB: Unplug the Ameba EVB and plug again.

2.5 Start firmware download

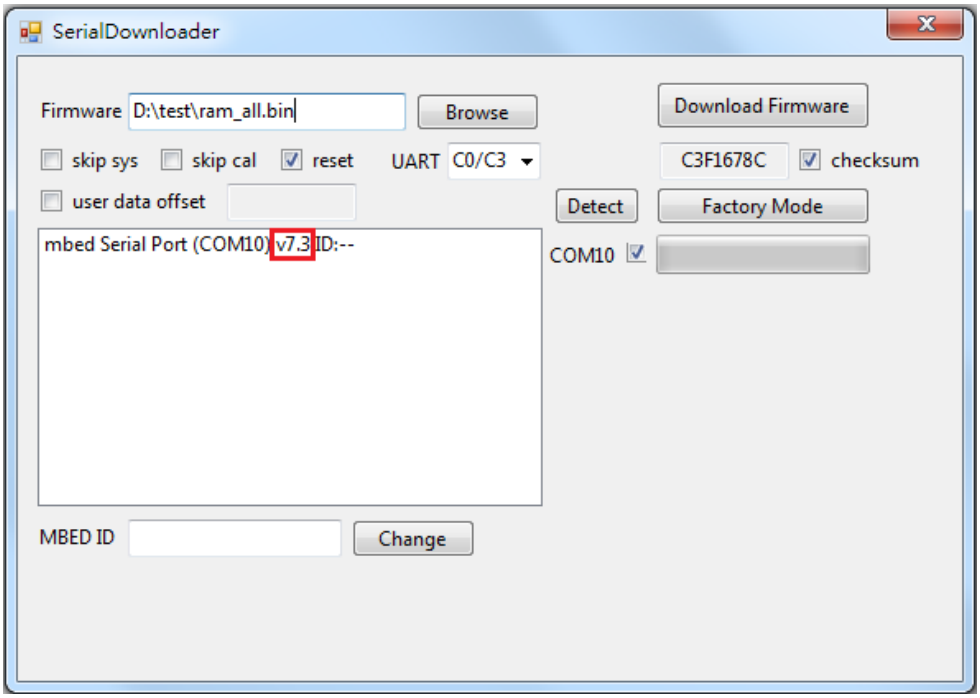
2.5.1 Windows form mode

➢ **Execution**

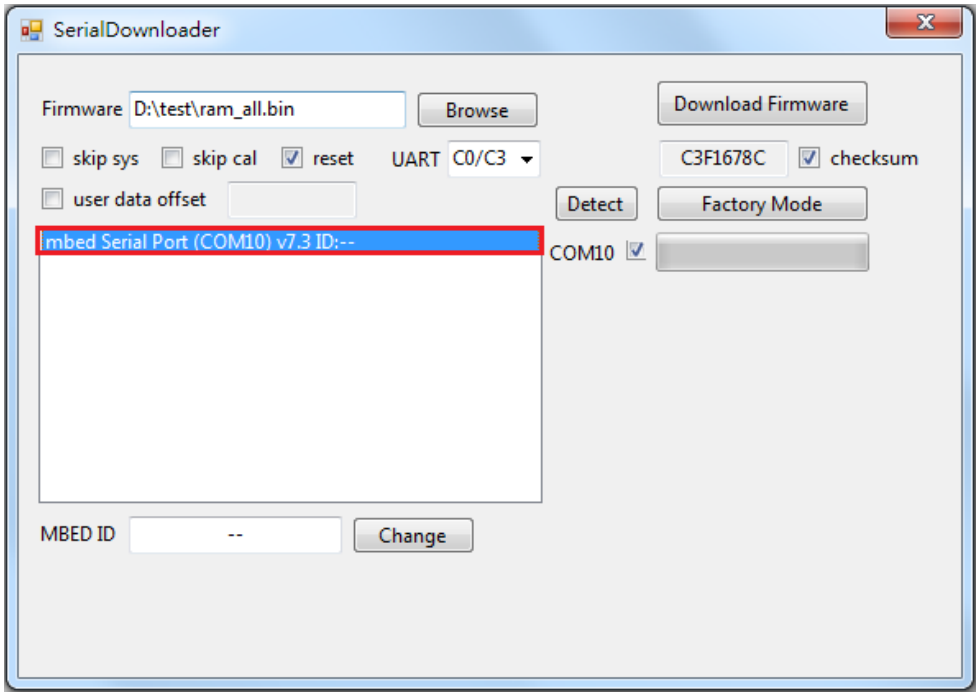
To use download tool in Windows form mode, double click the SerialDownloader_cpp.exe or execute SerialDownloader_cpp.exe on command prompt (cmd) without parameters.

```
$ D:\SerialDownloader_cpp.exe
```

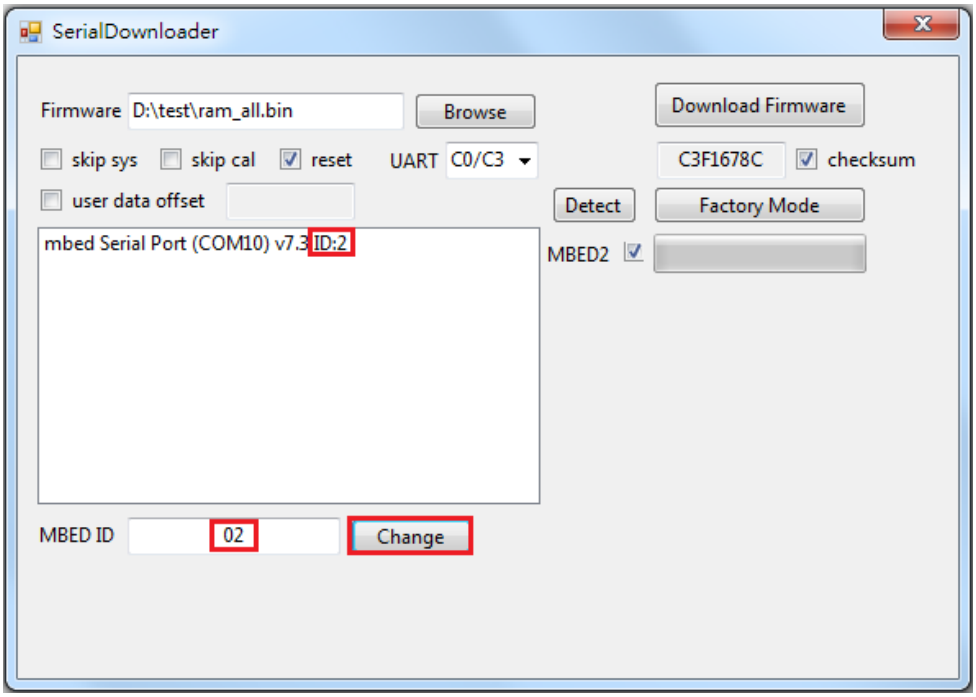
Check whether there is “mbed Serial Port (COMn) vx.xx ID:--” showing in the text box. vx.xx is DAP FW version.



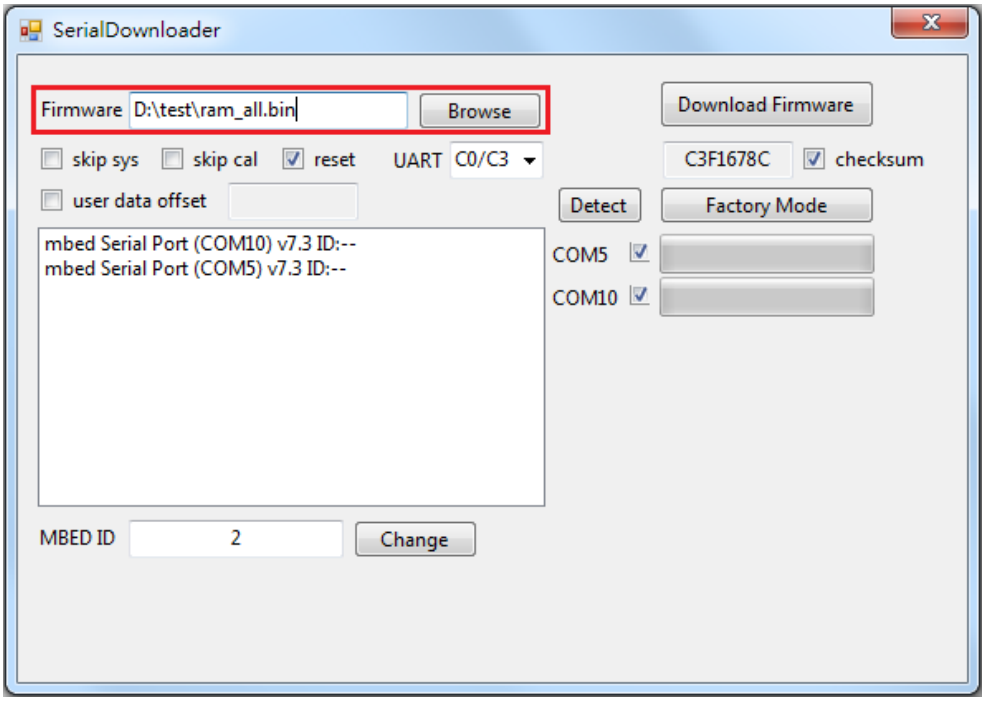
Select “mbed Serial Port (COMn) vx.xx ID:--”.



Click MBED ID to set ID number and click “Change” button. ID number will be set and store to Ameba’s flash.

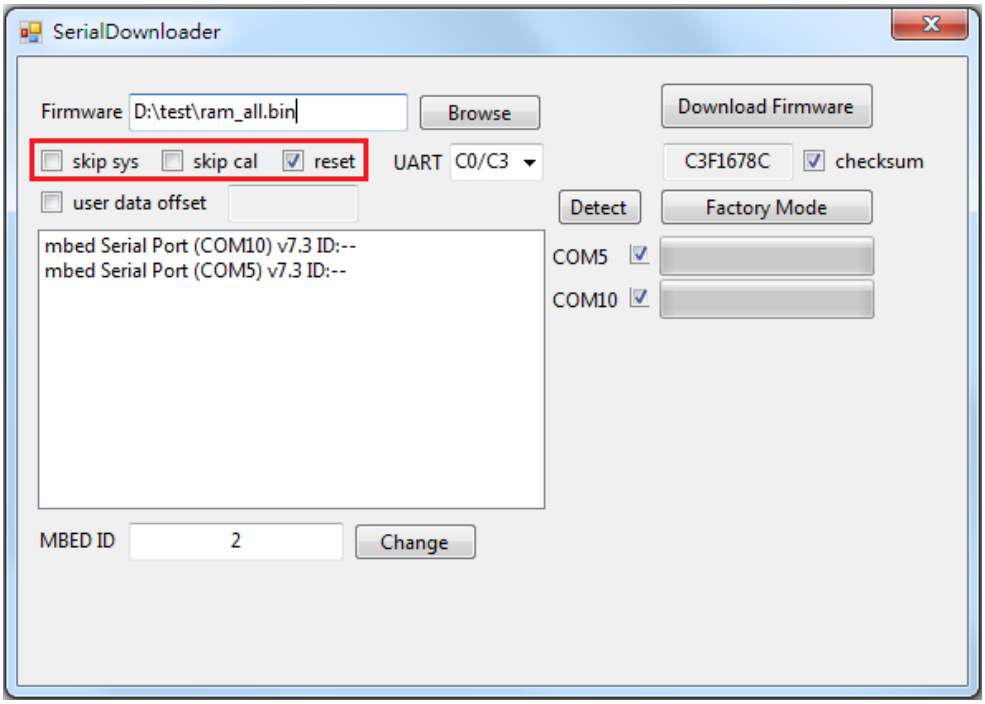


Click “Browse” button to select firmware which you would like to download.

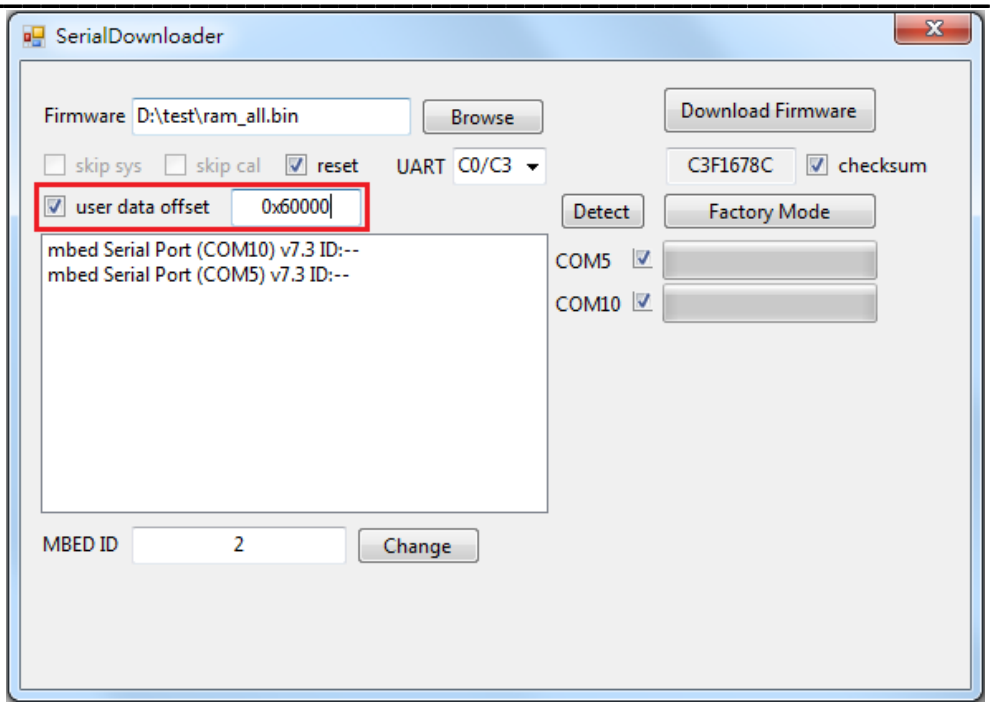


➢ **Configuration**

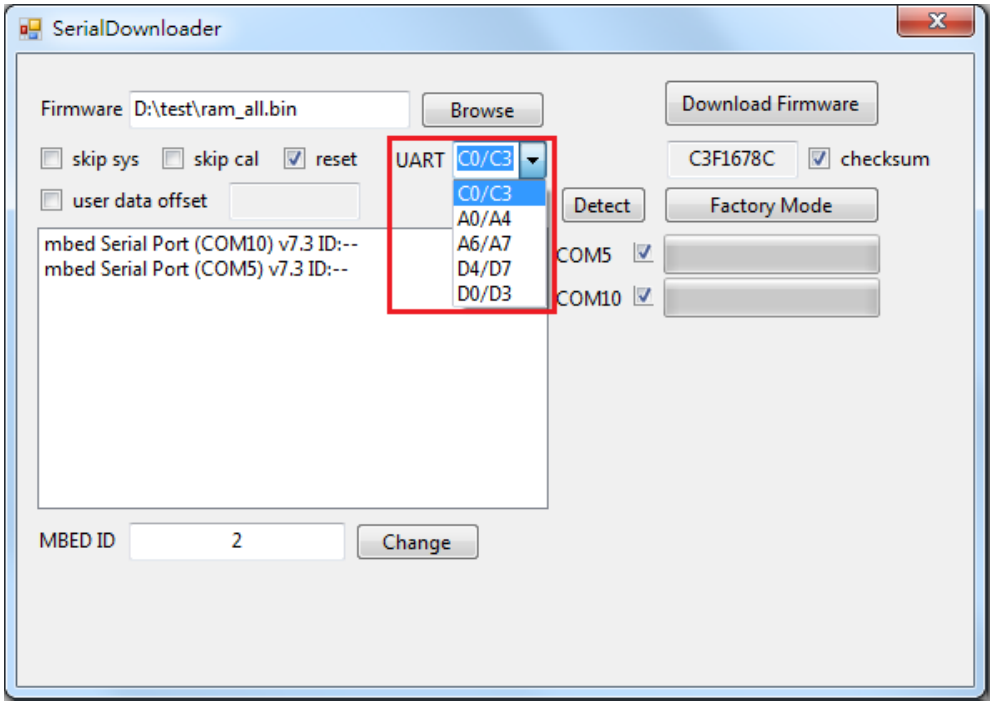
Select whether skipping system data area.
Select whether skipping calibration data area.
Select whether reset the DUT after download success.



Select “user data offset” if the image is downloaded as user data (e.g. license).
You also need to specify the target address where the user data locate. Note that the address should be 4k-aligned address and is in hexadecimal format (e.g. 0x60000).
Enable the user data download would make “skip sys” and “skip cal” unavailable.

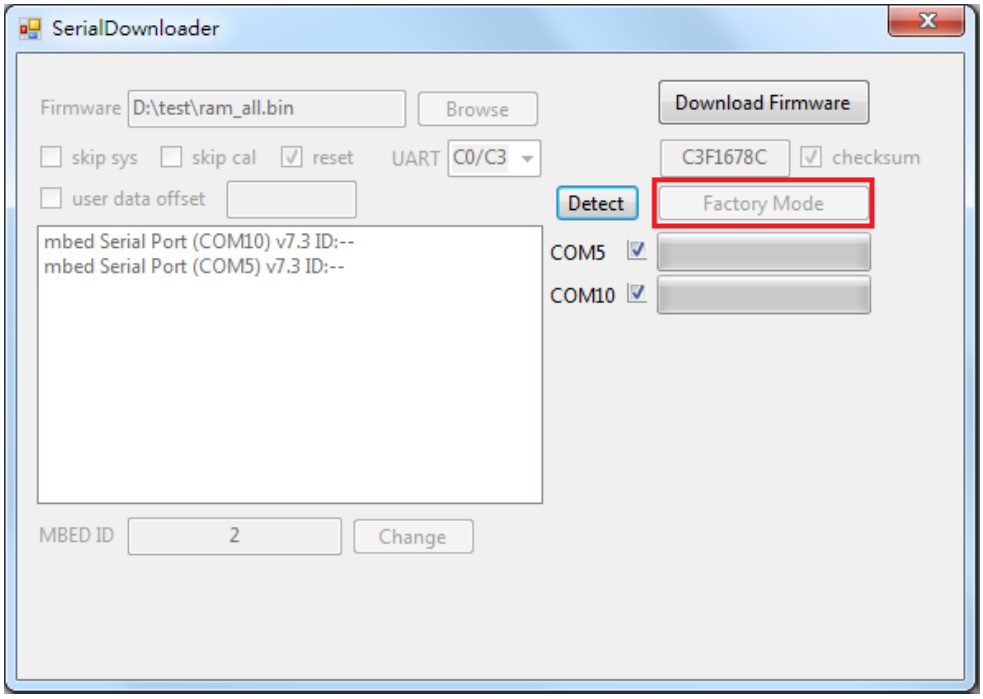


Select which UART pins (C0/C3 or A0/A4) are used for X-Modem transfer. UART pins (A6/A7 and D4/D7) are supported in Ameba RTL8711AM and RTL8195AM.

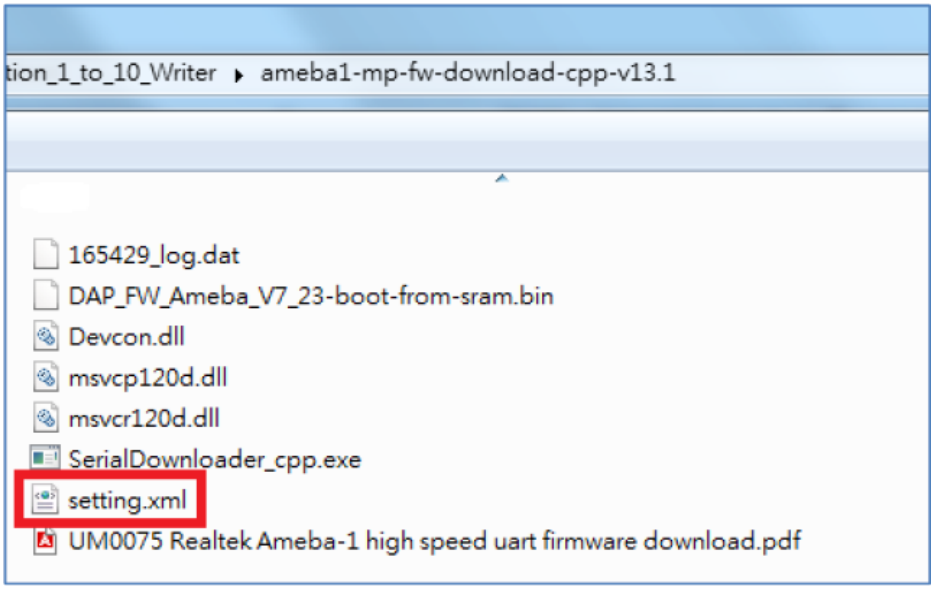


➢ **Factory mode**

Click button “Factory Mode” can disable some function after setting is done to avoid incorrect operations.

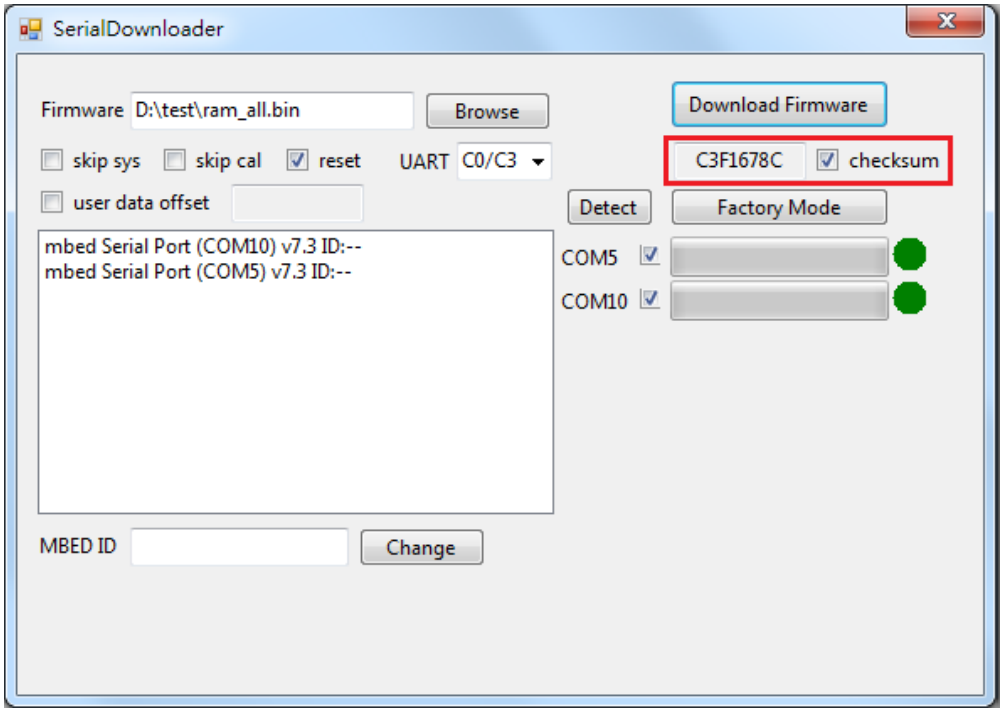


If you will disable this “Factory Mode” function, you can delete “setting.xml” in ameba1-mp-fw-download-cpp-vx folder and then re-execute SerialDownloader_cpp.exe again.



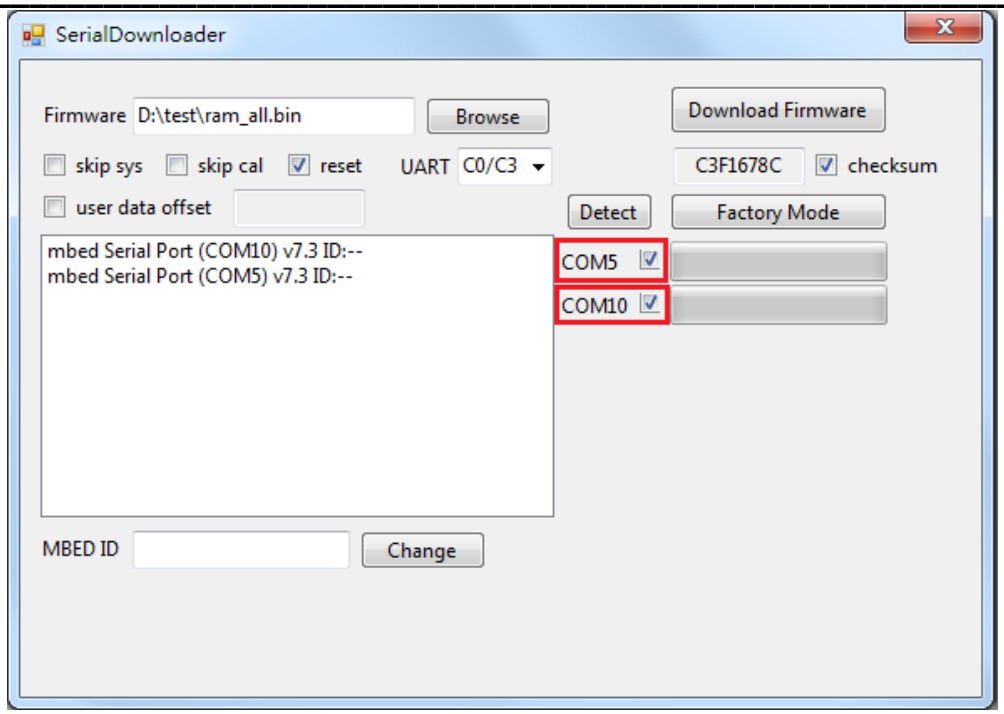
➢ **Checksum function**

If checksum box is selected, FW will calculate the checksum value of firmware while download is finished and then read back this checksum value to compare with SerialDownloader's. If checksum value is correct, **green** light is showed, otherwise **red** light is showed. If checksum box is not selected, we cannot guarantee whether written data is correct or not.



➢ **Download firmware through UART**

Click check box "MBED*n*" to select which MBED disk will be download.



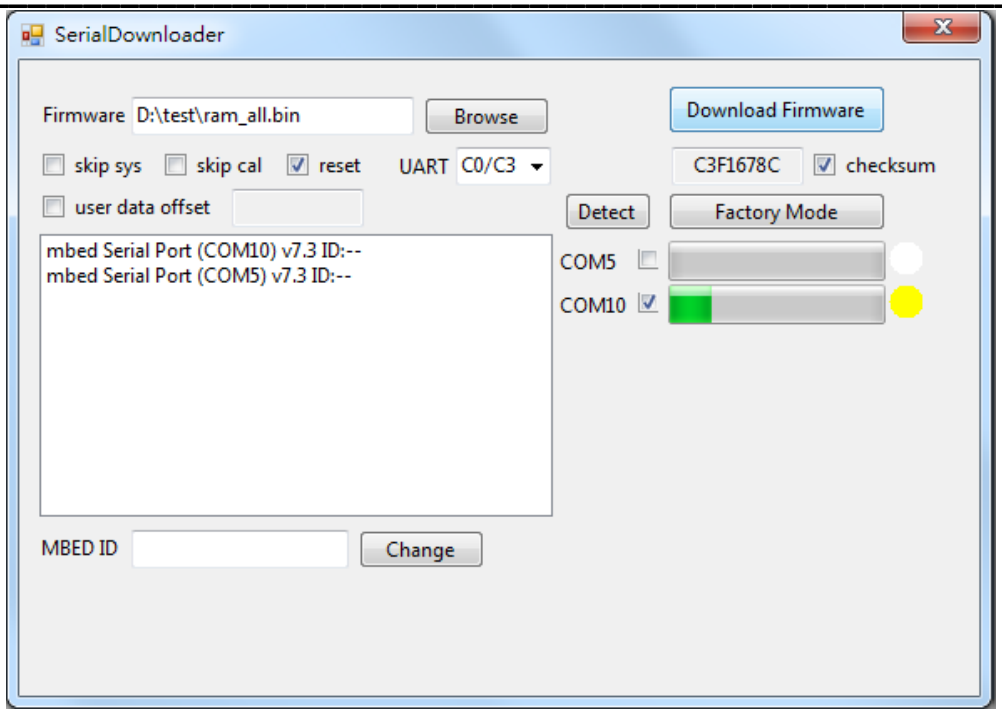
Check “checksum” box for checking correctness of firmware download.

Click “Download Firmware” button.

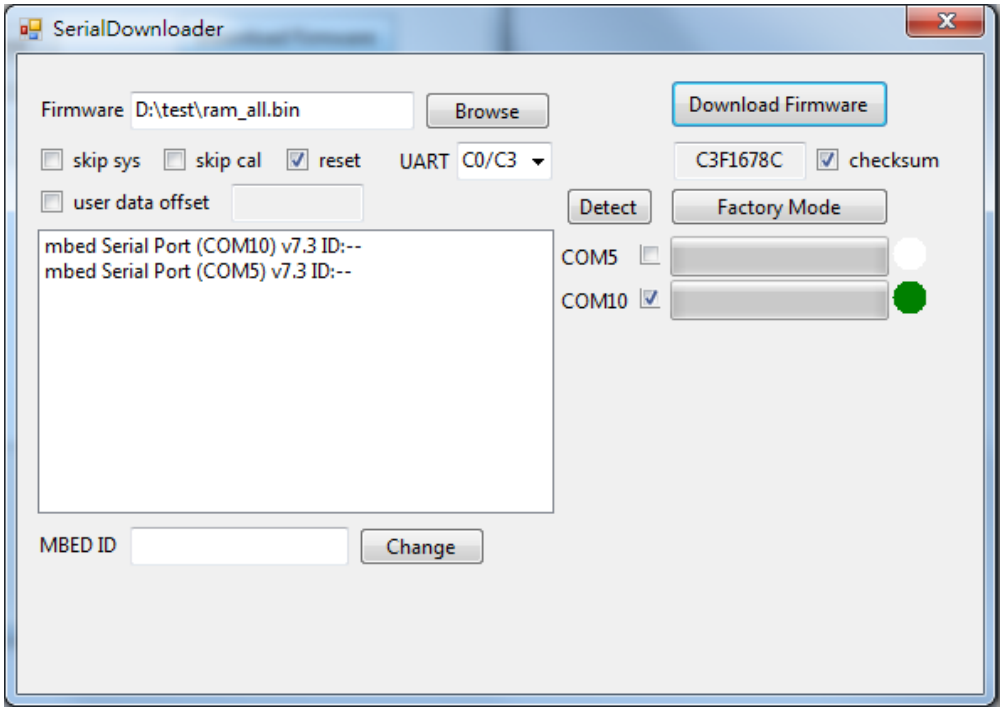
It blinks **yellow** light during firmware downloading.

If checksum is not selected, it turns **blue** once download completion, otherwise, it turns **red**.

If checksum box is selected, it turns **green** if checksum pass, otherwise, it turns **red**.



Download Firmware in progress



Download firmware complete and checksum verification okay

2.5.2 Command line mode

➢ **Execution**

To use download tool in command line mode, execute SerialDownloader_cpp.exe on command prompt (cmd) with the defined parameters. For example, you can type **-help** to show the usage of the download tool.

$ D:\SerialDownloader_cpp.exe -help

```
C:\>D:\SerialDownloader_cpp.exe -help
usage: -download [-port=x,y..] [-id=a,b..]
               E.g. -download
                 or -download -port=5,10 -id=3
usage: -check [devices|setting|skipsys|skipcal|reset|uartset|checksum|firmware|checks
umvalue|userdataoffset]
usage: -set [skipsys|skipcal|reset|checksum] [True|False]
usage: -set [uartset] [0|1|2|3]
               0: C0/C3, 1: A0/A4, 2: A6/A7, 3: D4/D7
usage: -set [firmware] <path>
               E.g. -set firmware "D:\test\ram_all.bin"
usage: -set [userdataoffset] [<hex_addr>|NULL]
               E.g. -set userdataoffset 0x60000, note that the address should be 4k-
aligned
                 or -set userdataoffset NULL, it will disable user data download
```

To check the devices, you can use **-check devices** as parameters.

$ D:\SerialDownloader_cpp.exe -check devices

```
C:\>D:SerialDownloader_cpp.exe -check devices
COM5 v7.3 ID:--
COM10 v7.3 ID:--
```

➢ **Configuration**

To check the tool configuration, you can use **-check setting** as parameters.

$ D:\SerialDownloader_cpp.exe -check setting

```
C:\>D:SerialDownloader_cpp.exe -check setting
SKIPSYS=False
SKIPCAL=False
RESET=False
UARTSET=0
CHECKSUM=False
FIRMWARE=
USERDATAOFFSET=
```

To set whether skipping system data area, use **-set skipsys [True|False]**.

$ D:\SerialDownloader_cpp.exe -set skipsys [True|False]

```
C:\>D:SerialDownloader_cpp.exe -set skipsys True
SKIPSYS=True
```

To set whether skipping calibration data area, use **-set skipcal [True|False]**.

$ D:\SerialDownloader_cpp.exe -set skipcal [True|False]

```
C:\>D:SerialDownloader_cpp.exe -set skipcal True
SKIPCAL=True
```

To set whether auto reset the device after successfully download, use **-set reset [True|False]**.

$ D:\SerialDownloader_cpp.exe -set reset [True|False]

```
C:\>D:SerialDownloader_cpp.exe -set reset True
RESET=True
```

To select which UART pins are used for X-Modem transfer, please use **-set uartset [0|1|2|3]**.
Uartset *0* represents C0/C3, *1* represents A0/A4, *2* represents A6/A7, and *3* represents D4/D7.

$ D:\SerialDownloader_cpp.exe -set uartset [0|1|2|3]

```
C:\Users>D:\SerialDownloader_cpp.exe -set uartset 1
UARTSET=1
```

To set the firmware you want to download, please use **-set firmware <path>**.

$ D:\SerialDownloader_cpp.exe -set firmware "D:\test\ram_all.bin"

```
C:\Users>D:\SerialDownloader_cpp.exe -set firmware "D:\test\ram_all.bin"
FIRMWARE=D:\test\ram_all.bin
```

To set the image downloaded as user data, please use **-set userdataoffset <hex_addr>**.
On the other hand, if you don't want the image downloaded as user data, please use **-set userdataoffset NULL**.

$ D:\SerialDownloader_cpp.exe -set userdataoffset [<hex_addr> | NULL]

```
C:\>D:SerialDownloader_cpp.exe -set userdataoffset 0x60000
USERDATAOFFSET=0x60000
```

- **Checksum function**

If you enable the checksum function, FW will calculate the checksum value of firmware when download is finished and then compare it to the checksum value calculated by download tool. If the checksum value is incorrect, the tool will show that the download is failed.
To enable/disable the checksum function, please use **-set checksum [True|False]**.

$ D:\SerialDownloader_cpp.exe -set checksum [True|False]

```
C:\Users>D:\SerialDownloader_cpp.exe -set checksum True
CHECKSUM=True
```

To check the checksum value calculated by download tool, use **-check checksumvalue**. Note that if the firmware is not set, the checksum value would be empty.

$ D:\SerialDownloader_cpp.exe -check checksumvalue

```
C:\Users>D:\SerialDownloader_cpp.exe -check checksumvalue
C3F16F8C
```

- **Download firmware through UART**

To download the firmware, make sure you have set the configurations correctly (**-check setting**) and then use **-download** to start the UART download.

$ D:\SerialDownloader_cpp.exe -download [-port=x,y..] [-id=a,b..]

```
C:\Users>D:\SerialDownloader_cpp.exe -download
COM4 v7.23 ID:2 success
COM5 v7.24 ID:3 success
```

You can append optional **–port** and **–id** to specify the target devices you want to download. The tool would filter out the un-selected devices from image download. Note that the devices should be split with "**,**" if there are multiple COM port/ID devices you want to download.

```
C:\>D:\SerialDownloader_cpp.exe -download -port=5,10
COM5 v7.3 ID:-- success
COM10 v7.3 ID:-- success
```

And if the output shows that the download procedure failed, you can check the log file which is located at *./log/* to get the fail reason code.

```
C:\Users>D:\SerialDownloader_cpp.exe -download
COM4 v7.24 ID:-- fail
COM5 v7.24 ID:3 success
```

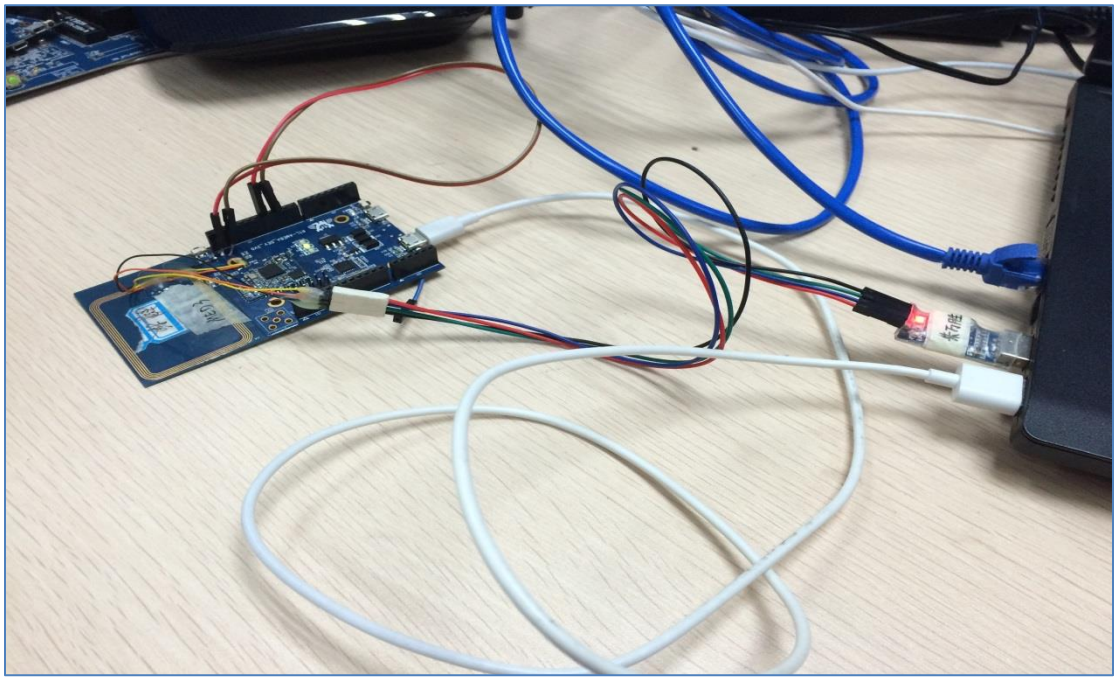
```
03:38:18 : Scan Device Start
03:38:18 : COM5scan: MBED serial found
03:38:18 : COM4scan: MBED serial found
03:38:18 : Scan Device End
03:38:18 : User click fiwmware download
03:38:18 : COM4 download start
03:38:18 : COM5 download start
03:38:18 : COM5 CPU detected
03:38:18 : COM4 CPU detected
03:38:19 : COM5 Boot OK
03:38:19 : COM5 Firmware read done
03:38:19 : COM5 Vendoer command ok
03:38:19 : COM4 Boot OK
03:38:19 : COM4 Firmware read done
03:38:19 : COM4 Vendoer command ok
03:38:19 : COM4 Xmodem return
03:38:27 : COM5 Xmodem return
03:38:27 : COM4 download fail: 6
03:38:27 : COM5 download success
```

Message from log file

Error code	Description
1	Checksum fail
2	UART IO exception
3	Cannot find CPUID
4	Copy sram_all.bin fail
5	Xmodem cancel
6	No sync, retry fail
7	No firmware file
8	Xmodem xmit error
9	Xmodem remote not ack

3 Check debug message

If user wants to use uart log of Ameba to watch the message from serial port, user can connect tx/rx of UART of Ameba to another serial port module.



4 Trouble Shooting

- If user uses high speed UART firmware downloader to burn image file to customized module, he/she needs to check whether SWD interface has connection. (On Ameba EVB, SWD wires have be connected between DAP chip and Ameba wifi module.)
- Some anti-virus software will cause high speed UART firmware downloader cannot work normally since anti-virus disallows UART port access. Please try to remove or disable it.