Realtek Ameba Series xmodem UART Update firmware

This document illustrates how to use UART to update firmware with xmodem protocol

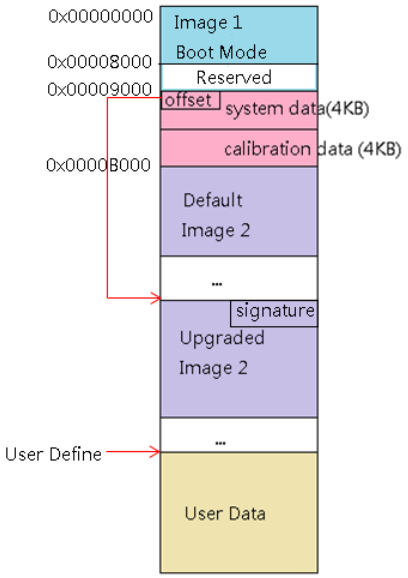
Table of Contents

1 UART update example

1.1 Introduction to UART OTA update

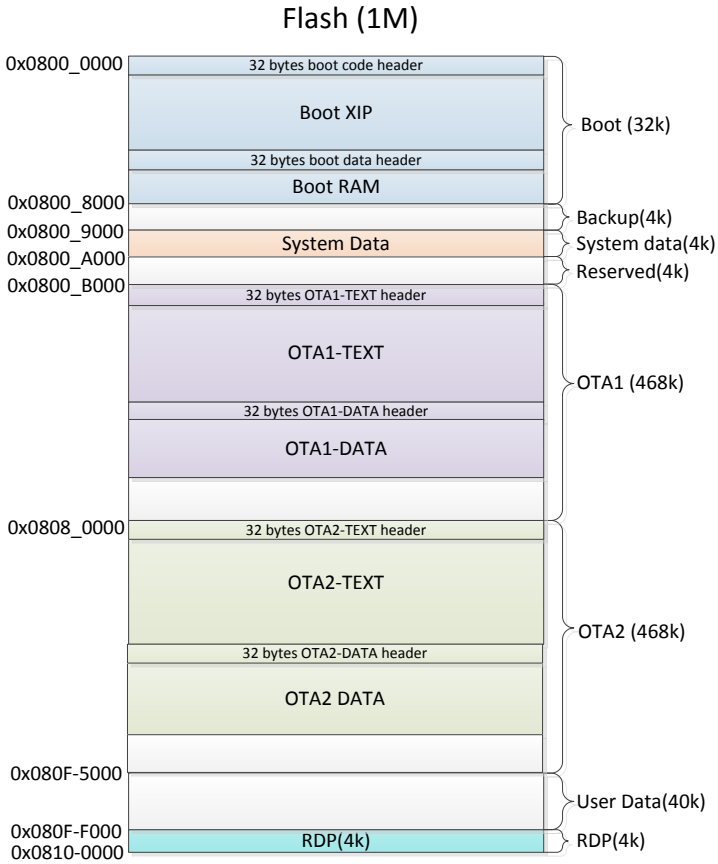
Here gives an example how to use UART update function in AMEBA-1(RTL8711AM/ RTL8195AM / RTL8711AF) and AMEBA-Z(RTL8710B). The UART update helps to flash the OTA image onto the AMEBA board using the UART protocol. Once the OTA image is flashed onto the AMEBA board the AMEBA will boot from the new updated OTA image.

For AMEBA-1, it is to be noted here that the original default image is not erased when the OTA image is being flashed onto the “Upgraded Image 2” section of the AMEBA-1’s memory layout as shown below and the system data offset is updated.



Since the OTA image flashed does not over-write the existing default image, we can switch the Ameba-1 to boot from the default image using AT Command “ATSC”, please refer to the AT Command manual for further details.

For AMEBA-Z, the example supports swap update between OTA1 and OTA2. The current image is not erased when the OTA image is being flashed onto the other section of the AMEBA-Z’s memory layout as shown below and the system data is updated.



Since the OTA image flashed does not over-write the current image, we can switch the AMEBA-Z to boot from OTA1 or OTA2 using AT Command “ATSC”, please refer to the AT Command manual for further details.

1.2 Make changes to SDK

- Configuration for baud rate:
  In file: component\common\example\uart_firmware_update\example_uart_update.c”
  Chang the baud rate in Ameba SDK to **115200**:

```
#define XMODEM_UART_BAUDRATE 115200
```

  Note: If you are using Ameba EVB board, please set the baud rate not higher than 115200. However, if your board supports a higher baud rate, you could set it to 1000000 even 3000000 to get a higher transfer rate.

- Enable the UART firmware update feature:

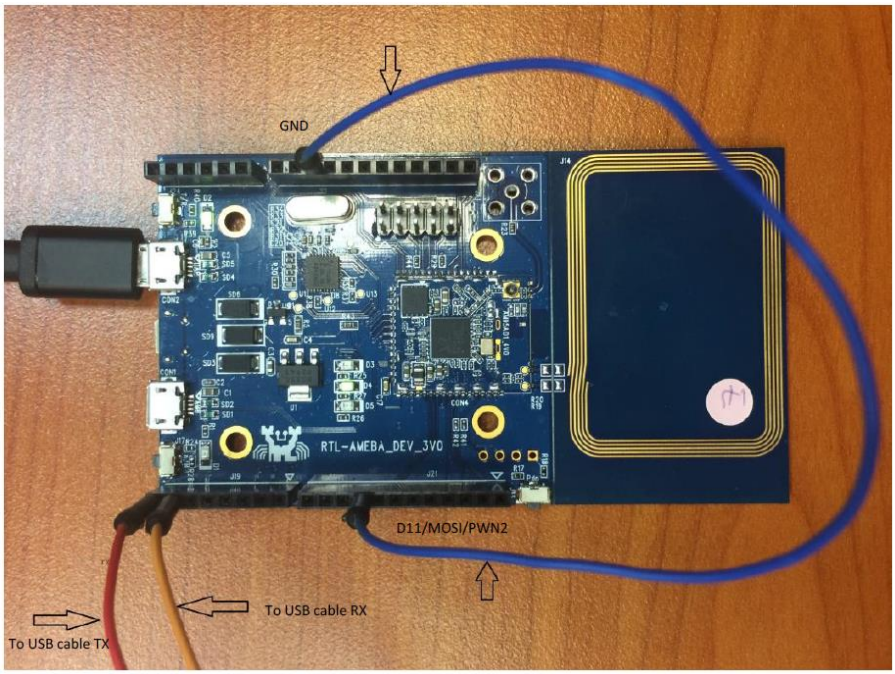
In file:" project\realtek_ameba1_va0_example\inc\platform_opts.h"

```
324  /*For uart update example*/
325  #define CONFIG_UART_UPDATE                1
326
```

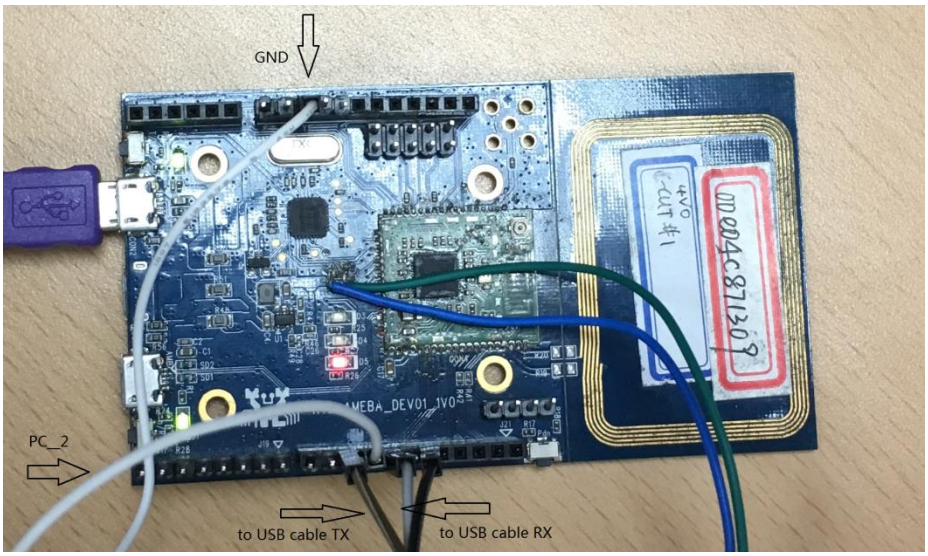
- Build the project after setting the above two parameters correctly and flash the AMEBA board.

1.3 Configure the GPIO of Ameba board

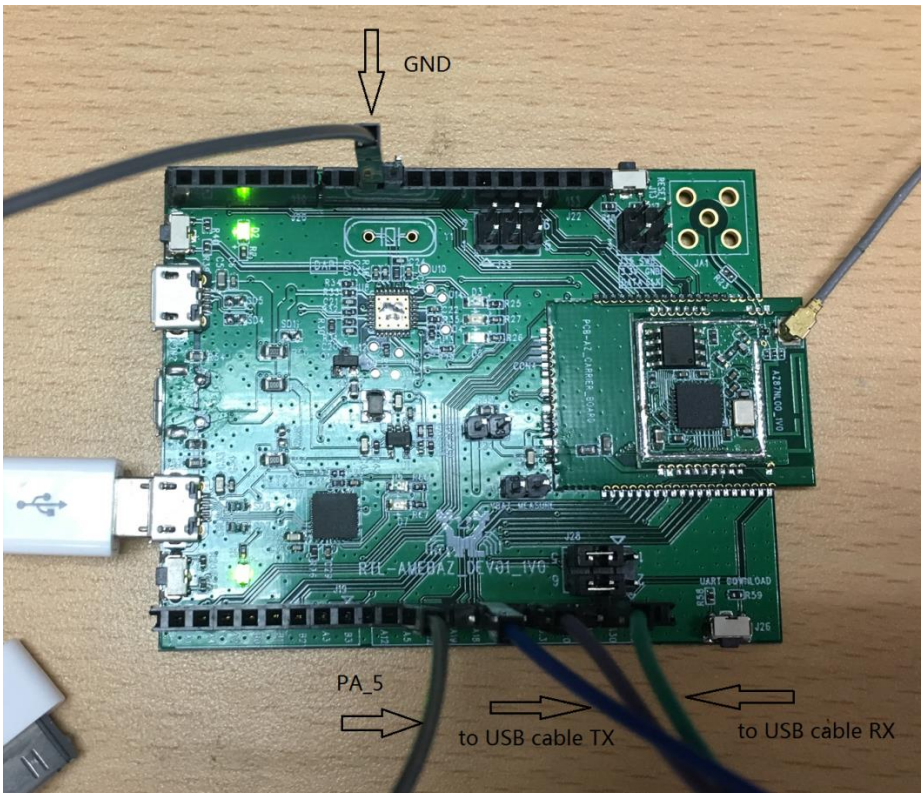
- Use PA_6 and PA_7 as UART TTL RX and TX pins in RTL8711AM/RTL8195AM. Use PC_0 and PC_3 as UART RX and TX in RTL8711AF, and change default XMODEM_UART_MUX in example_uart_update.c from 2 to 0. Use PA_18 and PA_23 as UART RX and TX in RTL8710B.
- Pull low PC_2 (D11/MOSI/PWN2) by connecting it to one of the GND pins in RTL8711AM/RTL8195AM/RTL8711AF, refer to "UM0048 Realtek Ameba1 DEV 1v0 User Manual_1v8_20160328" document for the AMEBA-1 pinout. Pull low PA_5 (PWN4) in RTL8710B; refer to "UM0113 Realtek Ameba-Z DEV 1v0 User Manual" document for the AMEBA-Z pinout.



RTL8711AM/RTL8195AM

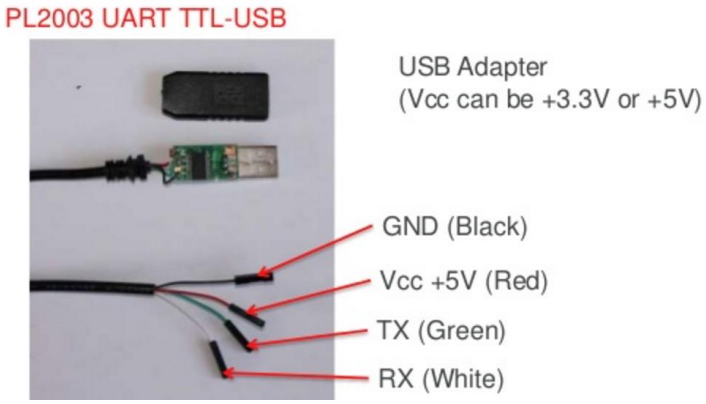


RTL8711AF

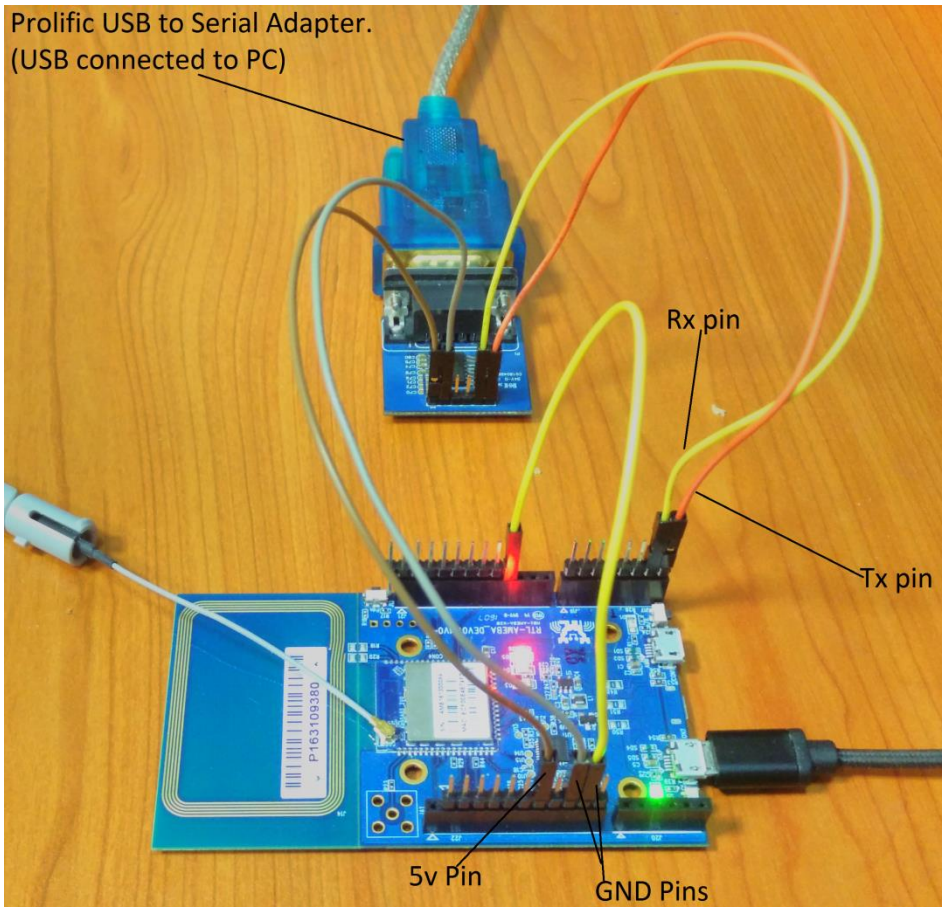


RTL8710B

- For connecting the tx and rx pins to the PC we can use a UART TTL-USB adapter as shown below. The Vcc and GND pins can be powered using the AMEBA board's pinout or can also be powered separately. The setup with the UART TTL-USB adapter has not been shown here as an alternative test setup was used in the lab setup which is shown in the next point after this.



- An alternative to connect the UART to pc is to use a combination of a UART TTL-RS232 Adapter and an RS323-USB serial adapter as shown below in the image.



As shown above the TTL-RS232 adapter can be powered by the AMEBA board directly or by using an external power supply as well.

1.4 Reboot Ameba board

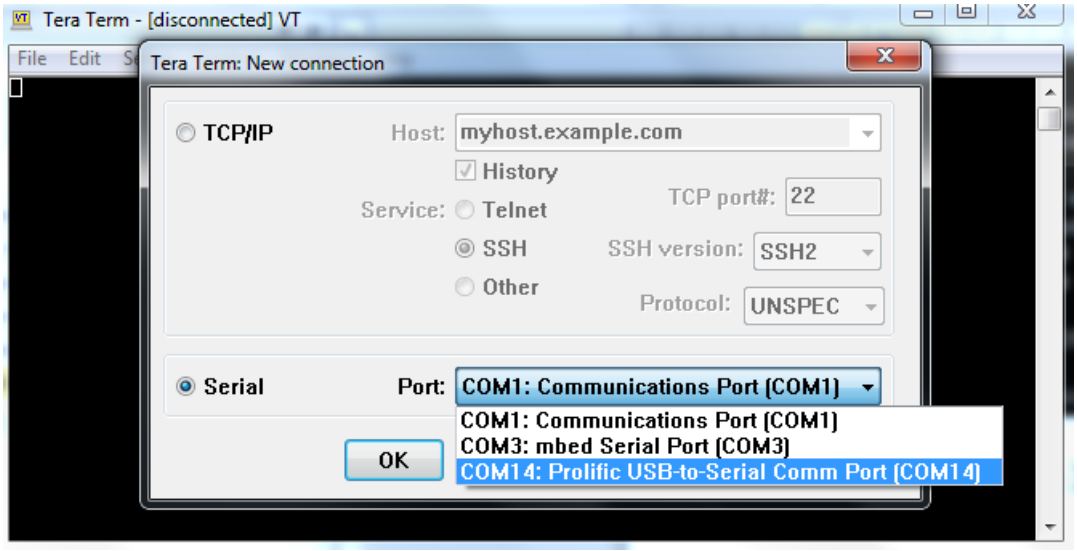
After building the software as mentioned in section 1.2 on the AMEBA board and making the necessary hardware and pins setup in section 1.3, we should now reboot the AMEBA board. Then you will see the following information, which indicates the xmodem uart update feature is enabled:

```
interface 0 is initialized
interface 1 is initialized

Initializing WIFI ...
Start LOG SERVICE MODE

# waitting update enable
update image enabled with xmodem protocol!
FW Update Over UART0, PinMux=2, Baud=115200
FW Update Image2 @ 0x80000

WIFI initialized

init_thread(53), Available heap 0xcca8
```

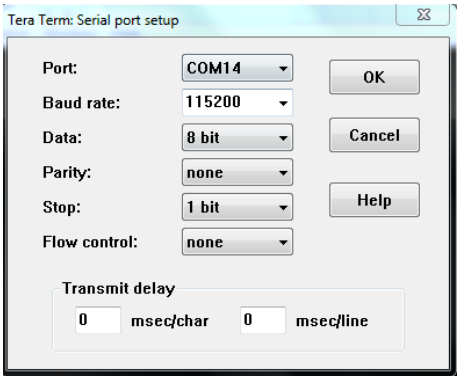
Once the following log message is printed the AMEBA is ready to receive an XMODEM OTA update from the PC.

1.5 Trigger an XMODEM OTA Software Update

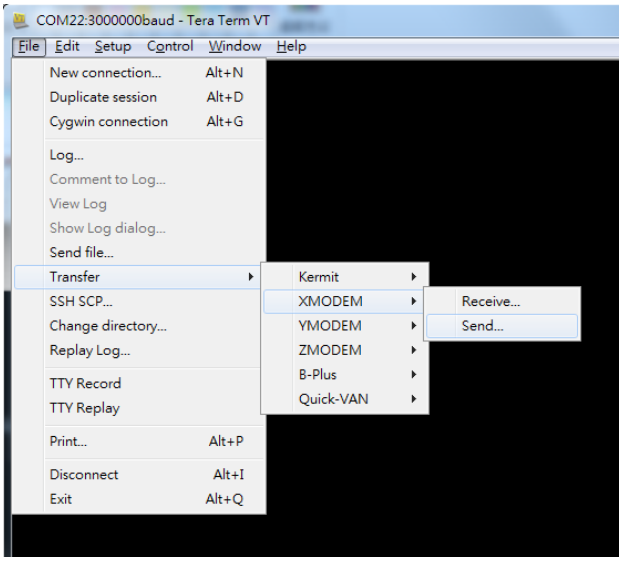
- In order to trigger an XMODEM OTA software update using UART we use the TeraTerm software on the PC. This is free software that can be downloaded from the internet.
- Open the TeraTerm software and select your COM port.



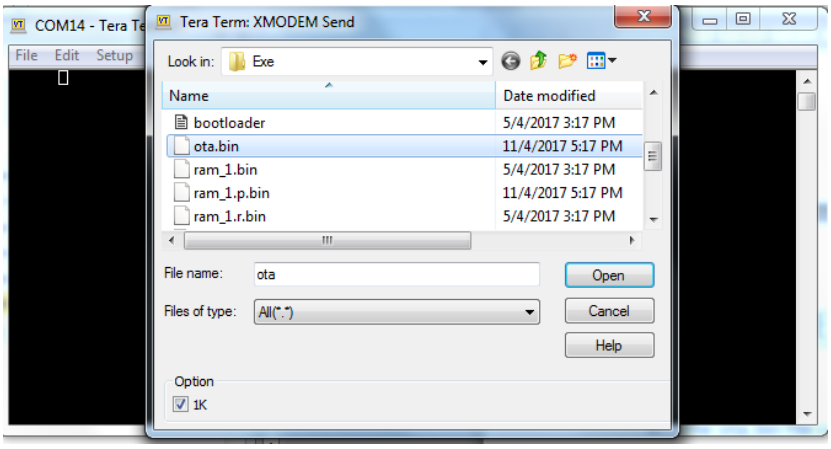
- Once the correct COM port is opened select the setup→Serial Port… option in Tera Term.
- Set your serial baud rate to **115200 (the same as in SDK)**:



- Choose xmodem to send new firmware(**ota.bin for AMEBA-1 or ota_all.bin for AMEBA-Z**) to Ameba board:

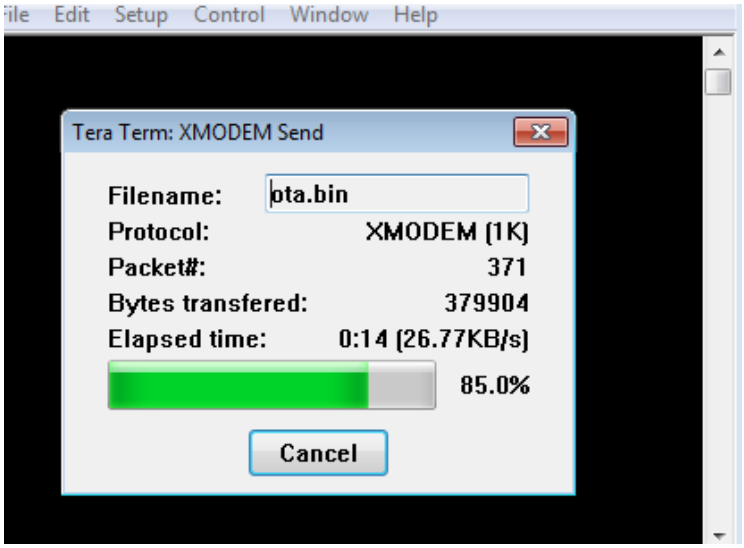


- The ota.bin file can be found in the Debug\Exe folder of the IAR project. The ota_all.bin is generated by image tool; refer to “AN0112 Realtek Ameba-Z Image Tool user manual” document.



Once you select the ota.bin or ota_all.bin, make sure to check the "1k" checkbox and click open.

- Then you will see the transfer progress:



- After file transfer is done, reboot the Ameba board, and you will see the new firmware is running.

```
===== Enter Image 2 ====
This is the new firmware to test xmodem update
interface 1 is initialized
interface 0 is initialized

Initializing WIFI ...
Start LOG SERVICE MODE

#
WIFI initialized

init_thread(53), Available heap 0x57220
```

In order to check whether the new firmware is flashed successfully, after flashing the UART update example mentioned in section 1.2, make a small change in the SDK, add a

printf statement in the main and rebuild the project. Once the project is rebuilt, a new ota.bin is generated. Use the new ota.bin to test the example.

2 Trouble shooting

If the UART xmodem update does not work well, please check the following places first.

- Make sure UART T/Rx pin definition is correct.
- Make sure baud rate is the same between send-side and receive-side.
- Make sure Physical connection of UART is correct.
- Make sure CONFIG_UART_UPDATE in platform_opts.h is enabled.
- Make sure external interrupt pin is connected correctly.