WINC Firmware Update Guide

MPLAB Harmony Integrated Software Framework

WINC Firmware Update

Introduction

Description

The ATWINC1500 and ATWINC3400 WiFi devices require firmware to be loaded into flash memory to operate. These devices will come shipped with firmware preloaded, however it may become necessary to update this firmware to take advantage of fixes and new or expanded features.

MPLAB Harmony 2.07 and later include the WINC Firmware Update utilities required to update the flash firmware of the WINC devices. These utilities can be found in the `<install-dir>/utilities/wifi/winc` directory.

Serial Programming Modes

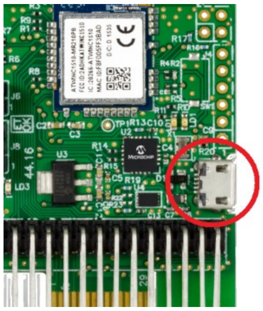
Firmware update of the WINC devices is performed over a serial interface. There are two methods of using this update protocol over serial to upgrade the WINC devices, direct to the device and via a serial bridge application running on the host MCU.

Direct to Device

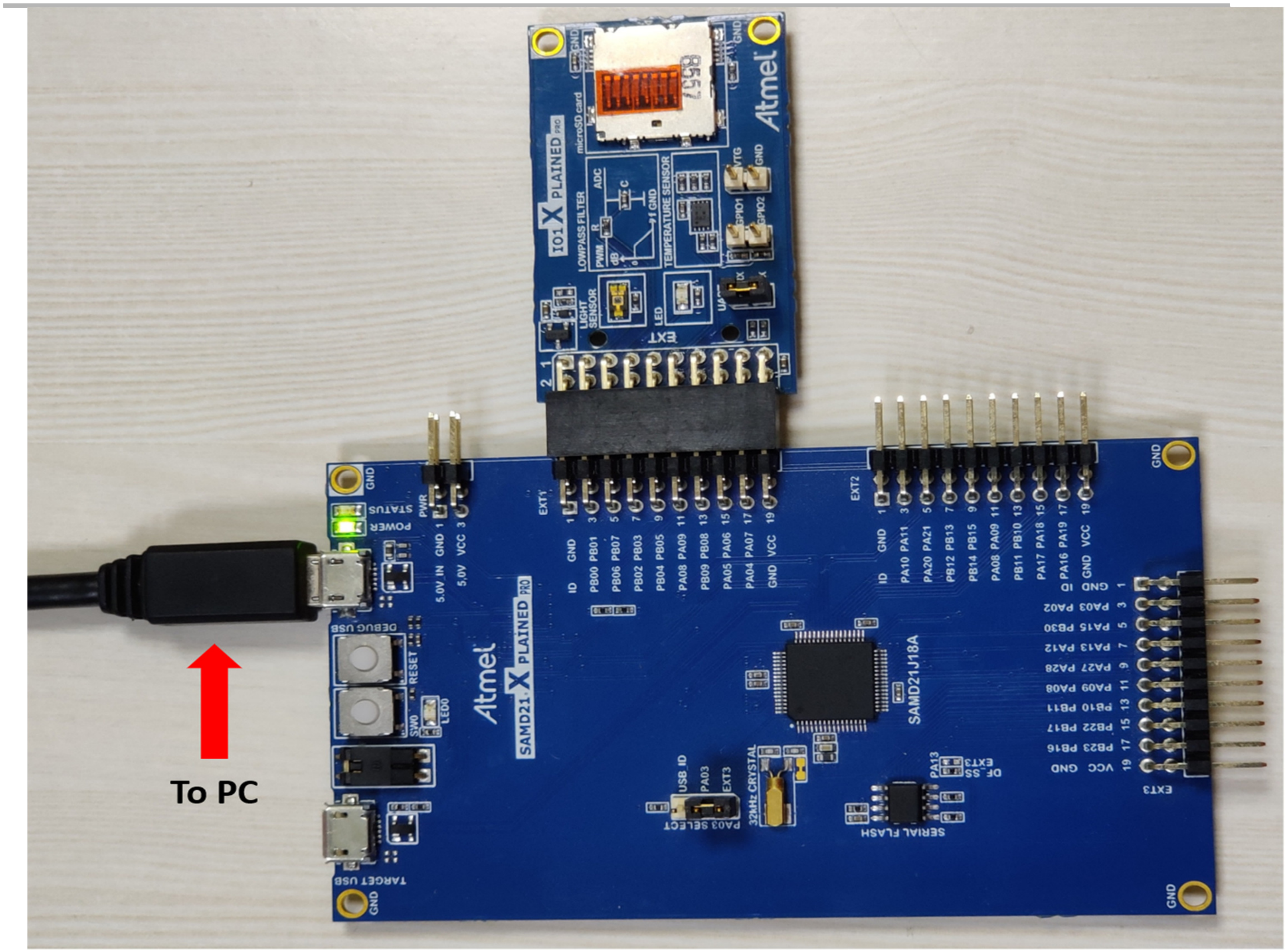
Direct to device programming is possible with the ATWINC1500 and ATWINC3400. In this mode the WINC must not be running. A serial connection is made directly to the serial port of the ATWINC1500 module (WINC1500 Wi-Fi PICtail/PICtail Plus Daughter Board).

To ensure the WINC device is not running it may be necessary for the host MCU to initialize the WINC device driver but not open an instance of the driver. An example application which performs this can be found in the `<install-dir>/apps/driver/wifi/winc/winc1500_prog` directory.

The serial connection is made between a PC and the micro USB connector J7 on the WINC1500 PICtail board.



In case of a SAMD21/SAME54, the Debug USB serial port connected to the PC helps update the FW on the WINC daughter card. The SAM D21 Xplained Pro Evaluation Kit allows the Embedded Debugger (EDBG) to be used for debugging. Connect the Type-A male to micro-B USB cable to the micro-B DEBUG USB port to power and debug the SAM D21 Xplained Pro Evaluation Kit.



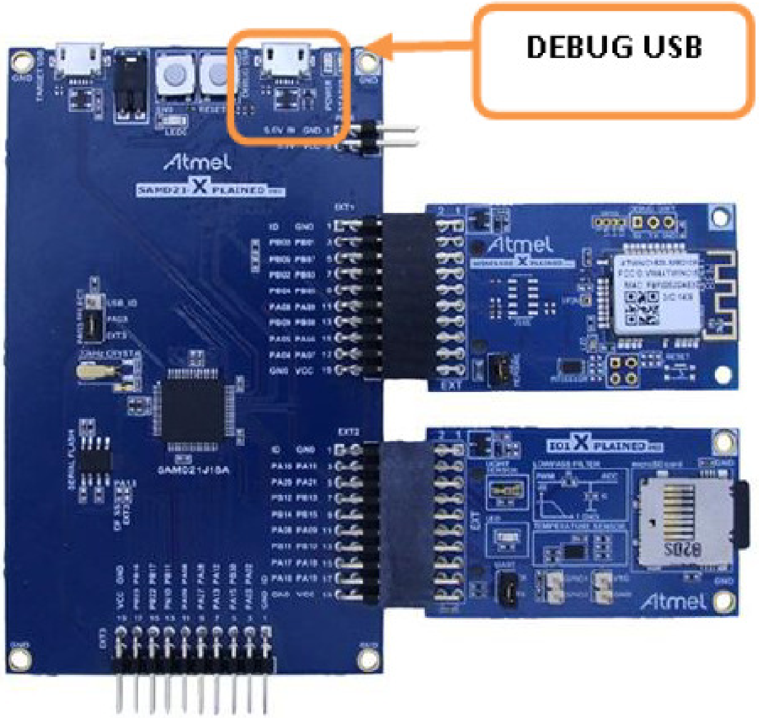
Serial Bridge

Normal communication with the WINC devices occurs via the SPI bus between the host MCU and the WINC device itself. To facilitate updating the WINC device the host MCU can run a service called Serial Bridge which links the update protocol available via the SPI bus to a suitable serial interface on the MCU. This then enables the WINC Firmware Update utilities to communicate with the WINC device through the host MCU serial interface.

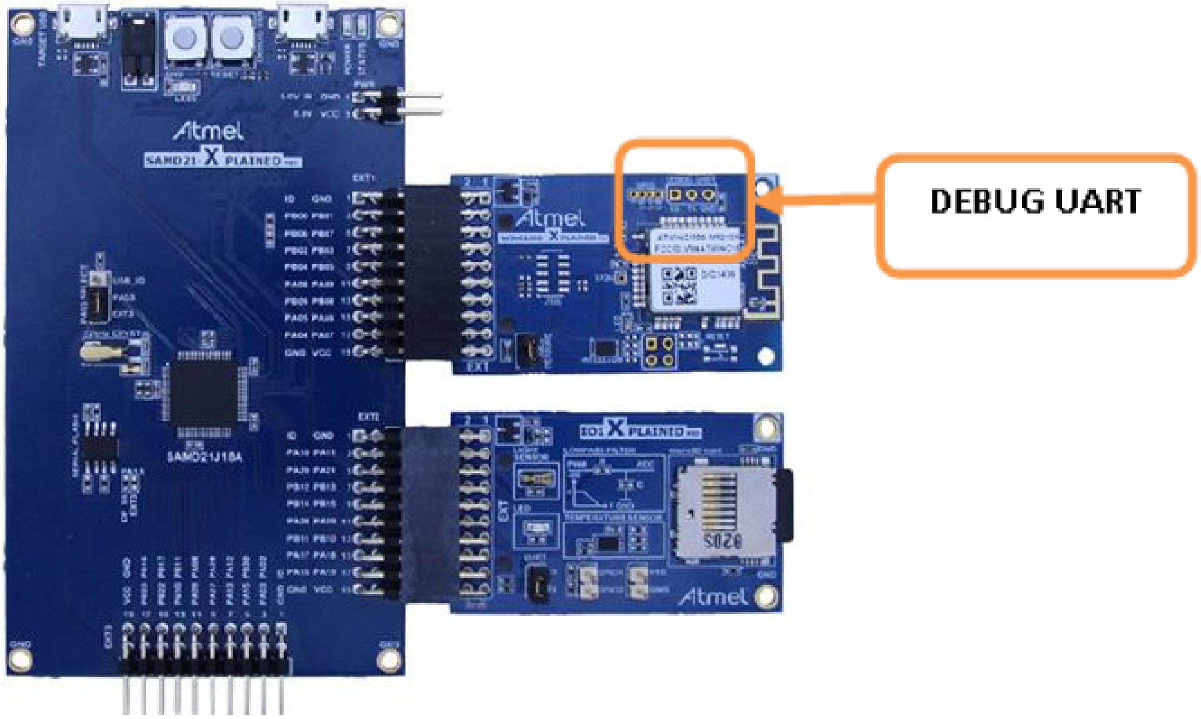
Located in the `<install-dir>/apps/driver/wifi/winc/serial_bridge` directory is an application which performs the serial bridge service. Currently there are four hardware configurations which are supported.

Option 1:

- SAMD21/SAME54 Xplained Pro development board
- USB to serial cables
- WINC1500/WINC3400 Daughter Board

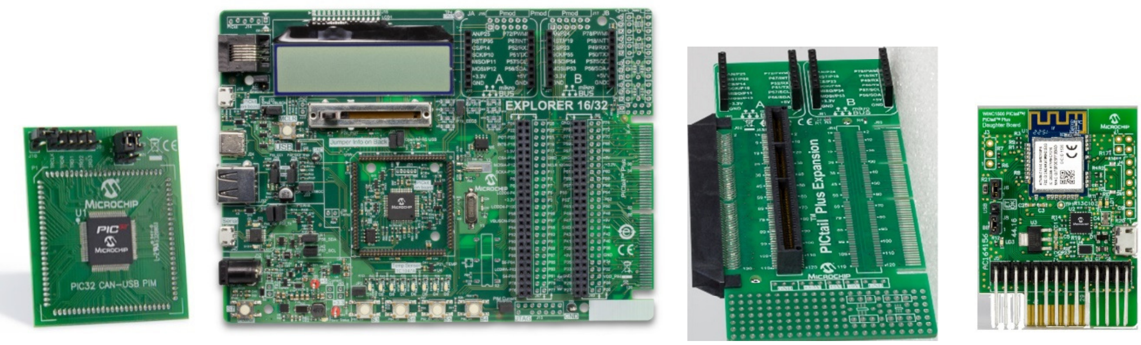


Debug UART connection can be used to monitor status of the FW upgrade in the WINC module.

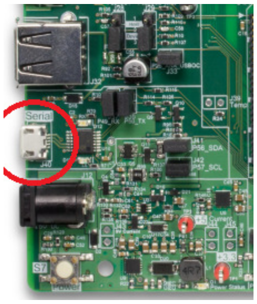


Option 2:

- Explorer 16/32 Development Board
- PIC32MX795F512L CAN-USB Plug-in Module (PIM)
- PICtail Plus Expansion Board
- WINC1500 Wi-Fi PICtail/PICtail Plus Daughter Board

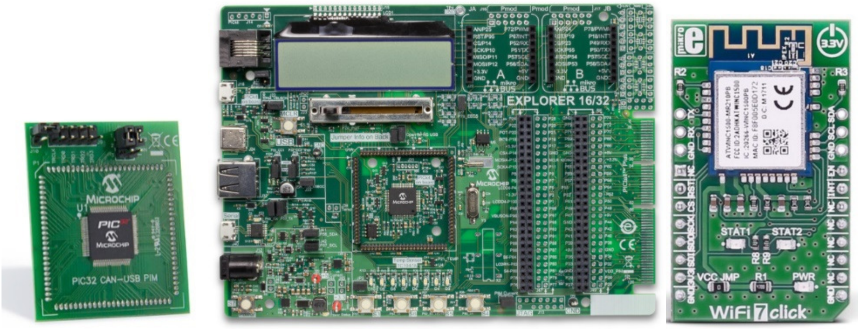


The serial connection is made between a PC and the micro USB connector J40 (marked Serial) on the Explorer 16/32 development board.

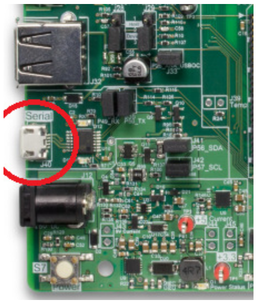


Option 3:

- Explorer 16/32 Development Board
- PIC32MX795F512L CAN-USB Plug-in Module (PIM)
- MikroElektronika WiFi 7 Click Board (insert in click port A)

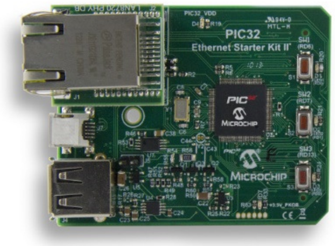


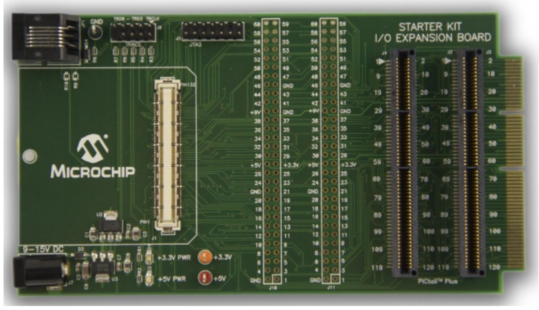
The serial connection is made between a PC and the micro USB connector J40 (marked Serial) on the Explorer 16/32 development board.



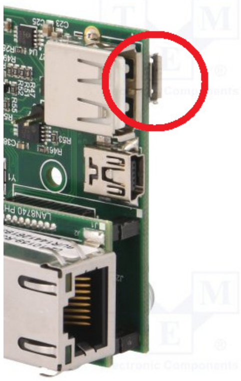
Option 4:

- PIC32 Ethernet Starter Kit II
- Starter Kit I/O Expansion Board
- WINC1500 Wi-Fi PICtail/PICtail Plus Daughter Board

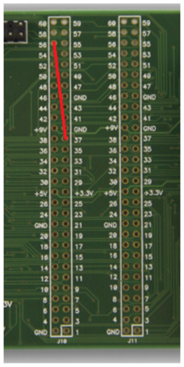




The serial connection is made between a PC and the micro USB connector J3 on the underside of the PIC32 Ethernet Starter Kit II development board.



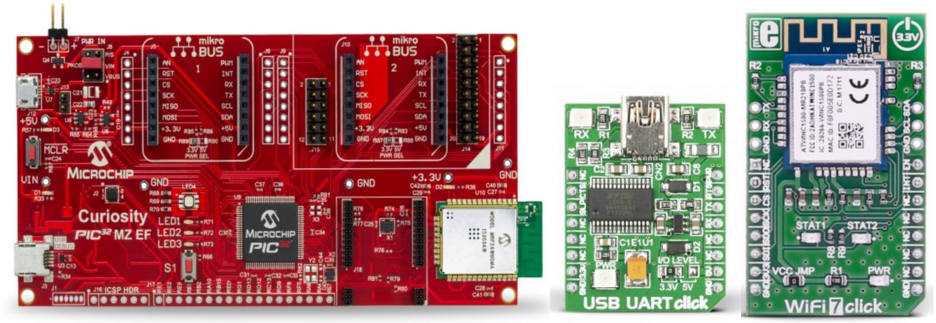
Note: When using the second option a hardware modification needs to be made. The IRQn signal from the WINC device connects through to INT1 on the Starter Kit I/O Expansion Board, this signal is routed to pin 85 of J1 which connects to the PIC32 Ethernet Starter Kit II board however it is not then connected to the PIC32. It is thus necessary to connect INT1 to an alternative connection which does connect through to the PIC32. The modification involves connecting INT1 (pin 37) to PMPA4 (pin 56) on the test point header J10.



Connecting INT1 to PMPA4

Option 5:

- Curiosity PIC32MZ EF Development Board (P/N: DM320104)
- USB UART click (P/N: MIKROE-1203) (insert in to click port 1)
- WiFi 7 click - ATWINC1510-MR210PB (P/N: MIKROE-2046) (insert in to click port 2)



The serial connection is made between a PC and the micro USB connector J3.

Firmware Update Utilities

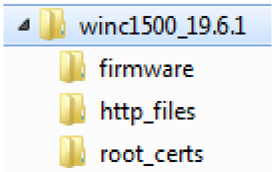
Updating a WINC device requires two utilities, an Image Tool which builds a file containing the changes required to the WINCs flash memory and a WINC Programming Tool which applies the changes to the WINC device.

Image Tool

The Image Tool is responsible for constructing an image of the WINCs flash memory from components such as firmware files, certificates and other data and creating a change description file which can be passed to the WINC Programmer to be applied to a WINC device.

The Image Tool is located in the `<install-dir>/utilities/wifi/winc` directory.

Each WINC firmware release will consists of a configuration file and a number of firmware and other files. The configuration file is presented to the Image Tool and is processed, in whole or in part to produce and output file. An example directory layout of a WINC firmware release is:



```
winc1500_19.6.1
    firmware
    http_files
    root_certs
```

The configuration file, `config.txt`, is in the `winc1500_19.6.1` directory. There are three subdirectories in this release:

- Firmware – contains firmware which runs on the WINC device.
- HTTP Files – contains files which are loaded into the HTTP web server made available by the WINC device when it enters a provisioning Soft-AP mode.
- Root Certs – contains root certificates used to validate a TLS connection.

Generating Device Programming Files

The following command line will generate a device programming file `winc1500_19.6.1.prog`.

```
>image_tool.exe -c winc1500_19.6.1\config.txt -o winc1500_19.6.1.prog -of prog
```

The file produced can be used to upgrade a WINC1500 to version 19.6.1 including the firmware, HTTP files and root certificates.

WINC Programmer

The WINC Programmer is responsible for reading, erasing and writing the flash memory of the WINC device. A device programming file can be provided to the tool to update the flash memory contents.

The WINC Programmer is located in the `<install-dir>/utilities/wifi/winc` directory.

Programming Device

The following command line will program a WINC1500 device via the serial port COM29.

```
>winc_programmer -p COM29 -d winc1500 -i winc1500_19.6.1.prog -if prog -w -r
                              -pfw winc1500_19.6.1\firmware\programmer_firmware_3A0.bin
```

```
WINC Programming Tool 1.0.0 [r291] (Jun 27 2018)
Copyright (C) Microchip Technology Inc. 2018

WINC serial bridge found
chip ID is 0x001503a0
programming fw file: winc1500_19.6.1\firmware\programmer_firmware_3A0.bin
starting device
reinitialise serial bridge to 500000
waiting for firmware to run
flash ID 0x001440ef
flash size is 8 Mb
begin write operation

 0x000000:[....w..w] 0x008000:[w.wwwwww] 0x010000:[wwwwwwww] 0x018000:[wwwwwwww]
 0x020000:[wwwwwwww] 0x028000:[wwwwwwww] 0x030000:[wwwwwwww] 0x038000:[wwwwwwww]
 0x040000:[wwwww...] 0x048000:[........] 0x050000:[........] 0x058000:[........]
 0x060000:[........] 0x068000:[........] 0x070000:[........] 0x078000:[........]
 0x080000:[........] 0x088000:[........] 0x090000:[........] 0x098000:[........]
 0x0a0000:[........] 0x0a8000:[........] 0x0b0000:[........] 0x0b8000:[........]
 0x0c0000:[........] 0x0c8000:[........] 0x0d0000:[........] 0x0d8000:[........]
 0x0e0000:[........] 0x0e8000:[........] 0x0f0000:[........] 0x0f8000:[........]

begin read operation

 0x000000:[rrrrrrrr] 0x008000:[rrrrrrrr] 0x010000:[rrrrrrrr] 0x018000:[rrrrrrrr]
 0x020000:[rrrrrrrr] 0x028000:[rrrrrrrr] 0x030000:[rrrrrrrr] 0x038000:[rrrrrrrr]
 0x040000:[rrrrrrrr] 0x048000:[rrrrrrrr] 0x050000:[rrrrrrrr] 0x058000:[rrrrrrrr]
 0x060000:[rrrrrrrr] 0x068000:[rrrrrrrr] 0x070000:[rrrrrrrr] 0x078000:[rrrrrrrr]
 0x080000:[rrrrrrrr] 0x088000:[rrrrrrrr] 0x090000:[rrrrrrrr] 0x098000:[rrrrrrrr]
 0x0a0000:[rrrrrrrr] 0x0a8000:[rrrrrrrr] 0x0b0000:[rrrrrrrr] 0x0b8000:[rrrrrrrr]
 0x0c0000:[rrrrrrrr] 0x0c8000:[rrrrrrrr] 0x0d0000:[rrrrrrrr] 0x0d8000:[rrrrrrrr]
 0x0e0000:[rrrrrrrr] 0x0e8000:[rrrrrrrr] 0x0f0000:[rrrrrrrr] 0x0f8000:[rrrrrrrr]
```

```
verify range 0x000000 to 0x100000

begin verify operation


 0x000000:[....p..v] 0x008000:[p.vvvvvv] 0x010000:[vvvvvvvv] 0x018000:[vvvvvvvv]

 0x020000:[vvvvvvvv] 0x028000:[vvvvvvvv] 0x030000:[vvvvvvvv] 0x038000:[vvvvvvvv]

 0x040000:[vvvvp...] 0x048000:[........] 0x050000:[........] 0x058000:[........]

 0x060000:[........] 0x068000:[........] 0x070000:[........] 0x078000:[........]

 0x080000:[........] 0x088000:[........] 0x090000:[........] 0x098000:[........]

 0x0a0000:[........] 0x0a8000:[........] 0x0b0000:[........] 0x0b8000:[........]

 0x0c0000:[........] 0x0c8000:[........] 0x0d0000:[........] 0x0d8000:[........]

 0x0e0000:[........] 0x0e8000:[........] 0x0f0000:[........] 0x0f8000:[........]


verify passed
```

Conclusion

Once verified, the Firmware has been updated. The WINC module now contains the updated Firmware and it can be used with any other application of choice.