SN8000 EVK User Guide

Version: 1.1

Release Date: October 11, 2014

Release Record

Version Number	Release Date	Comments
Version 0.1	9/23/2013	• Initial draft.
Version 0.2	10/03/2013	• Add appendix section about restore the preload FW
Version 1.0	12/20/2013	• Update board photos
Version 1.1	10/10/2014	• Updated "ignore leading path name segment" to 0

THE TABLE OF CONTENTS

TABLE of figures

TABLE OF TABLES

1. Introduction

SN8000 is a FCC/IC certified IEEE 802.11b/g/n Wi-Fi module. It integrates Wi-Fi BB/MAC/RF IC, RF front end, clock, and on-board antenna. The SN8000 Development Kit can serve as a software development platform to design IP-enabled WiFi systems using Broadcom's WICED architecture [3]. This document provides the necessary information for setting up the EVK.

1.1 Acronyms

Acronym	Meaning
API	Application Programming Interface
EVB	Evaluation Board
EVK	Evaluation Kit
FW	Firmware
GPIO	General Purpose Input/Output
PC	Personal Computer
SW	Software
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

1.2 References

[1] Murata, "SN8000 EVB schematics"
[2] Murata, "SN8000 SDIO adapter schematics"
[3] Broadcom WICED architecture (http://www.broadcom.com/support/wiced/)
[4] Wingbond, "W25Q80BL Serial Flash Memory With Dual and Quad SPI"
[5] ST Microelectronics, "UM0462: STM32™ and STM8™ Flash loader demonstrator"

2. Setting up the SN8000 EVK for Windows

The SN8000 Development Kit (SN8000 EVK) supports the evaluation and development of wireless IP networking products. It consists of an evaluation board (EVB) [1], a SN8000 SDIO adapter [2], a ribbon cable, a mini-USB cable, and an installation package and this user guide. The SDIO adapter contains a SN8000 module and a ribbon cable connector. The EVB contains a STM32F205RG microcontroller, a dual channel USB-UART/USB-JTAG interface IC, a ribbon cable connector and a SDIO socket. The SN8000 EVK is intended for the following purposes:

- Demonstrate the dual AP/STA mode of operation. The preloaded firmware supports a web-based interface for configuring the WiFi connection to a specified AP.
- Serve as a software development platform to design IP-enabled WiFi systems using the Broadcom WICED architecture [3].

Each of these steps is discussed in detail in the following subsections.

2.1 SN8000 SDIO adapter

The SDIO adapter board is a carrier board for the SN8000 module supporting 802.11 b/g/n WiFi. It can be connected to the EVB and provide the hardware platform for application development – insert the SDIO adapter to the SDIO socket and connect the ribbon cable for IO controls as shown in Figure 1.

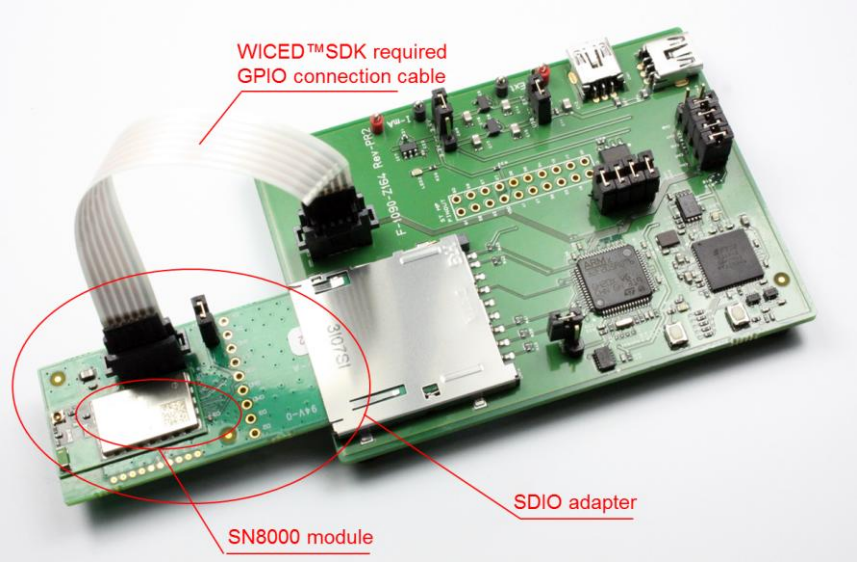


Figure 1 SN8000 SDIO adapter connected to the EVB

The ribbon cable connector CN2 provides the necessary GPIO signals used by WICED SDK. The signal assignment is as shown below:

- (1, SLOW_CLK) , (2, GPIO1), (3, GPIO0), (4, RST_N), (5, VDD_3V3_EN)

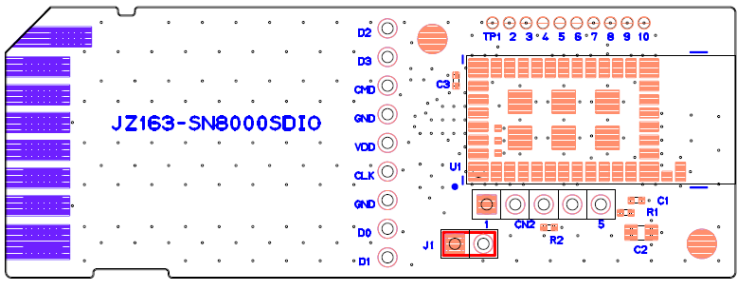


Figure 2 Ribbon cable connector CN2 signals

2.2 SN8000 EVB

The SN8000 EVB is paired with a SN8000 SDIO adapter board to provide the hardware platform for application development. Some major components of the EVB are shown in Figure 3.

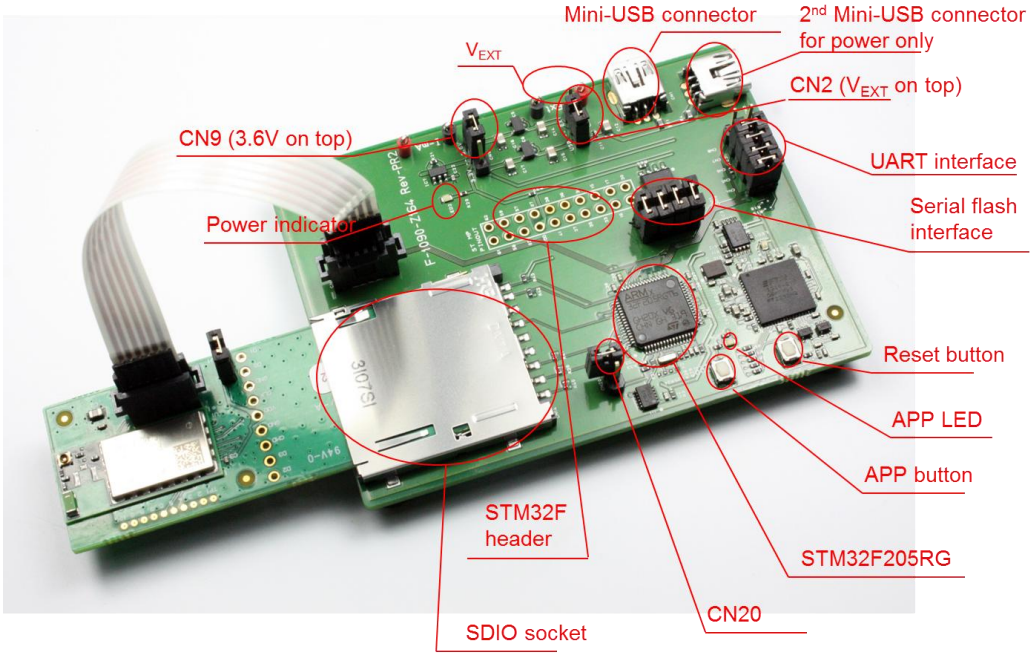


Figure 3 SN8000 EVB paired with a SN8000 SDIO adapter

- SDIO socket for the SN8000 adapter board
- STM32F205RG ARM Cortex-M3 processor with integrated memory
- Mini-USB connector supporting USB-JTAG and USB-serial interfaces
- Jumper CN2 to select power supply source between V_{USB} and V_{EXT}.
- Jumper CN9 to select power supply source for VDD_SD (3.3V or 3.6V)
- Jumper CN20 to select BOOT mode (default is off)
- Connectors for extern 5V power supply (V_{EXT})
- Power indicator LED for availability of power to the SDIO interface

- Reset button
- Application button
- Application LED
- Serial flash (8M bits) and interface connectors
- UART interface connectors
- Ribbon connector to select I/O pins of the SN8000 module
- Pads for header to access select I/O pins of the STM32F

2.2.1 Programming and debug interface

The SN8000 EVB provides a mechanism for programming and debugging applications on the STM32F through the USB-JTAG interface. The instructions for installing the driver for the USB-JTAG interface are described in Section 2.4.

2.2.2 UART interface

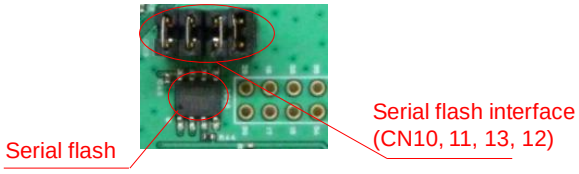
The EVB also provides a serial interface to the STM32F processor through the USB-UART interface. The instructions for installing the driver for that interface are described in Section 2.4. Connectors CN4-CN7 may be removed to disconnect the USB-UART interface. The ordering of the jumpers in the illustration below listed from top to bottom is CN4, 5, 6, 7 and 8. The left column of the pins is connected to the USB-UART port, and the right column is connected to STM32F. Jumper on CN8 should be removed.

Connector	STM32F pins	Signal	Remark
CN4	PA12	RTS/	Jumper on to use USB-UART interface
CN5	PA11	CTS/	Jumper on to use USB-UART interface
CN6	PA9	TXD	Jumper on to use USB-UART interface
CN7	PA10	RXD	Jumper on to use USB-UART interface

Table 1 UART interface jumpers

2.2.3 Serial flash

The EVB has an 8MBits Winbond W25Q80BL ([4]) serial flash that may be used for over-the-air FW upgrade by the WICED SDK (Section 2.4). The jumpers on connectors CN10-CN12 may be removed to disconnect the serial flash from the module so that those pins may be connected to the customer platform for other purposes. The ordering of the jumpers in the illustration below listed from left to right is CN10, 11, 13 and 12. The bottom row of the pins is connected to the serial flash, and the top row is connected to STM32F.



Connector	STM32F pins	Flash Signal	Remark
CN10	PA4	CS/	Jumper on to use serial flash

CN11	PA5	CLK	Jumper on to use serial flash
CN13	PA7	DI	Jumper on to use serial flash
CN12	PA6	DO	Jumper on to use serial flash

Table 2 Serial flash interface jumpers

2.2.4 EVB supply jumper setting

The EVB power can be provided either by the USB host or an external 5V supply. The supply source is configured by jumper CN2, as described in Table 3. By default, the USB is selected as the power source for the EVB.

Power source	CN2 position	Remark
USB	USB	Jumper away from VEXT
External 5V DC	EXT	Jumper closest to VEXT

Table 3 EVB power supply jumper setting

2.2.5 SN8000 supply jumper setting

The SN8000 power (VDD_SD supplying both VDD_IO and VDD_BAT) can be provided by the EVB and is selectable between 3.3V and 3.6V. The supply source is configured by jumper CN9, as described in Table 4. The power indicator LED shows the availability of power to the SN8000 SDIO adapter.

VDD source	CN3 position	Remark
3.3V	3.3V	Jumper closest to power indicator
3.6V	3.6V	Jumper away from power indicator

Table 4 VDD supply jumper setting

2.2.6 Application button and LED

The application button and LED are connected to STM32F for application usage, e.g., WICED bootloader button and LED. The button and LED are connected to PC4 and PC5, respectively, of STM32F.

2.2.7 Header pads

The SN8000 EVB provides the following header pads (0.1” pitch) for application-specific usage:

- Select STM32F205RG I/Os with the pin number as labeled on the silk screen

2.3 Using the preload firmware

By default the SN8000 EVB is already loaded with an application firmware. It provides basic features such as soft AP and simple web service. Power up the board through a mini USB cable connected to a PC. (Discard any “Found new USB device” message on PC for now. The demo only needs the USB interface to provide the +5V power. The USB driver installation will be addressed in the Section 2.4)

Once the board is power up, on a Laptop PC (or any web capable device), turn on WiFi and scan for the

soft AP's SSID "Murata-WS-xxxxxx", the "xxxxxx" is the last 3 octets of the EVB's MAC address.

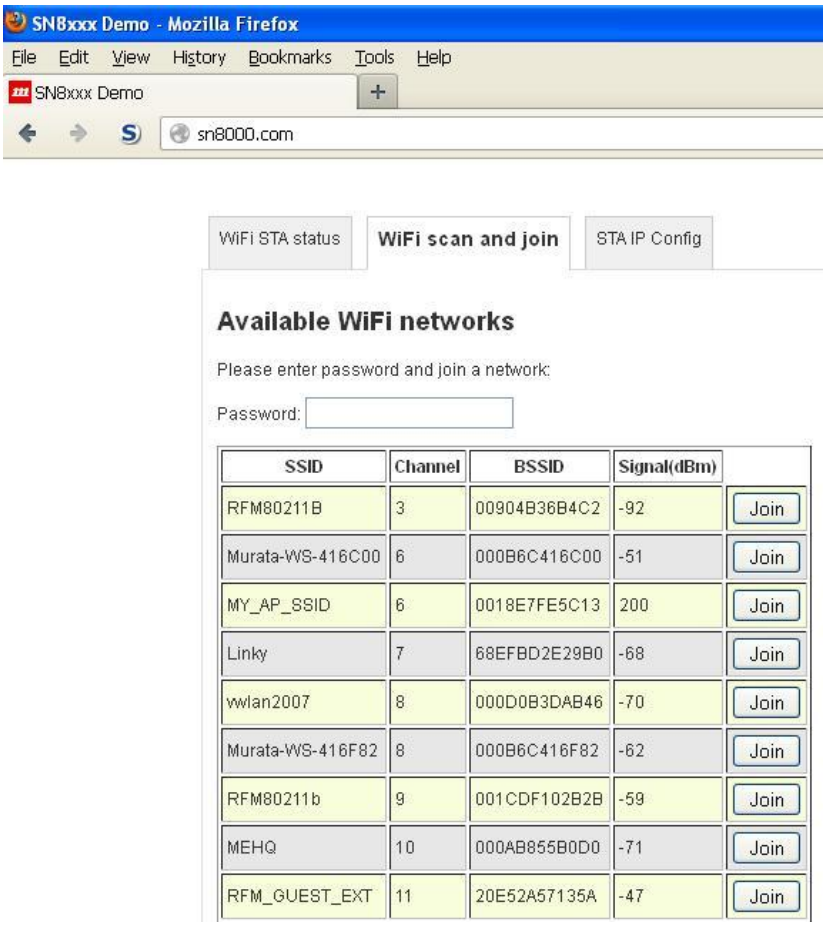
After joining the soft AP, the laptop should get IP address assignment by the soft AP, which is 172.31.0.xx, by default. Open a web browser such as Firefox and enter "sn8000.com". The following sections illustrate the functionalities of the sample web files.

2.3.1 Wifi status

The Wifi STA status tab indicates the current network status of the module. By default it is not connected to any network at the beginning.

2.3.2 Wifi scan and join

Click on "WiFi scan and join" tab, it will perform the Wifi scan request first and then list the scan result.



Select a SSID (In security cases, fill the password first), click on “Join”. Wait for the join result to be updated. If such join request succeeds, the previous status window will be refreshed, like this:

Click on “Get RSSI” will retrieve the RSSI reading, “Leave” will perform the Wifi leave request.

2.3.3 Wifi IP config

Once joined, switch to "STA IP Config" tab. Click on "Configure IP" first then click on "Get IP info". If succeed, the assigned IP value will be displayed at bottom, as illustrated in the figure above.

At this stage, users should be able to ping the SN8000 EVB from other nodes who stays within the same network domain.

2.4 Setting up the WICED SDK

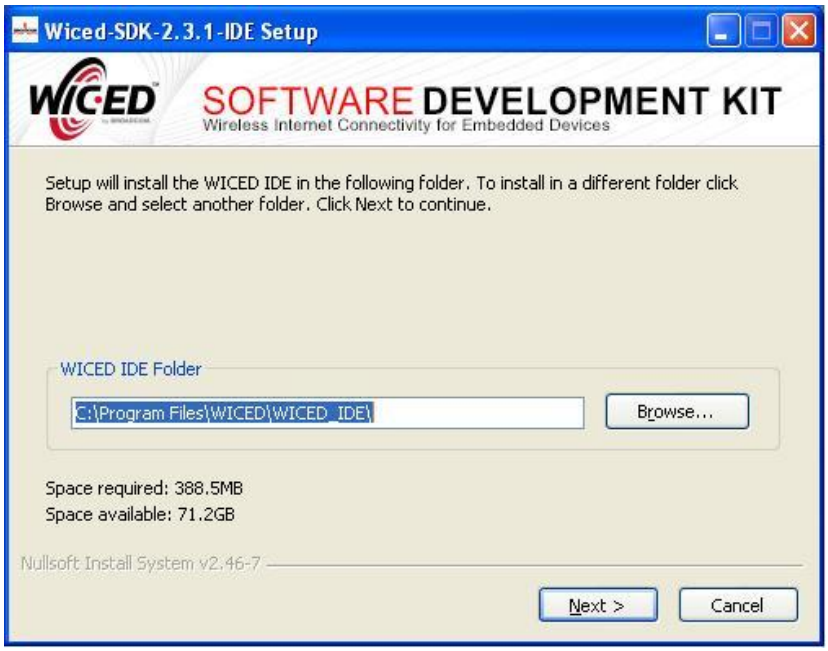
Users can develop their own applications on SN8000 EVB by using Broadcom® WICED SDK and IDE. To use the SDK, the following steps must be performed:

1. Download and install the WICED SDK and IDE
2. Apply the SN8000 platform patches
3. Create and compile an application
4. Load the application into the SN8000 EVB
5. Run the application
6. Debug the application

This section provides instructions for Windows XP/Windows 7 users. For other OS please refer to Broadcom WICED Quick Start Guide.

2.4.1 Install the WICED SDK and IDE

The WICED SDK and IDE package is available for download from the Broadcom® WICED website (http://go.broadcom.com/wiced). It is provided as a self-installing executable file. Double-click the Wiced-SDK-2.3.x-IDE-Installer.exe file to begin the installation. A setup window similar to the following appears.



Note: ensure that the SN8000 EVB is NOT connected to the PC prior to installing the WICED SDK.

Choose the installation folder for the WICED IDE and click **Next**, then choose the installation workspace folder for the WICED SDK and click **Install**. Once the installation completes, click **Finish** to immediately start the WICED IDE.

2.4.2 Connect the SN8000 Evaluation Board

The SN8000 EVB connects to the PC through USB. The USB interface provides +5V power as well as individual JTAG and UART interfaces to the STM32F205 onboard SN8000 EVB.

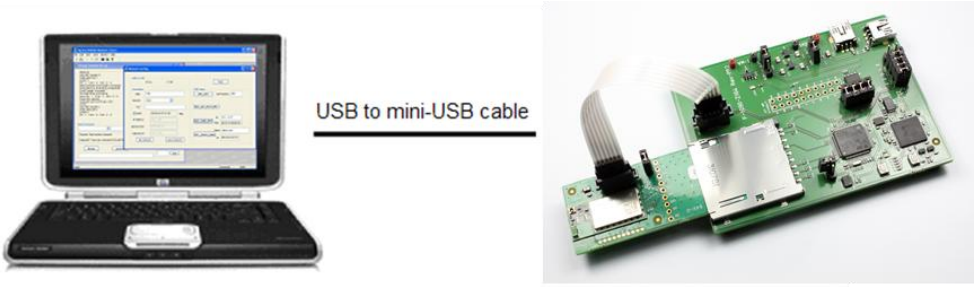


Figure 4 SN8000 EVK Configuration

Note: Do NOT plug the EVB into PC prior to installing the WICED SDK. Run Broadcom WICED SDK installer first so that USB driver for SN8000 EVB can be loaded properly.

The SN8000 EVB has two logical USB devices: a USB-JTAG device and a USB-UART device. USB drivers for the EVB were automatically installed during the previous SDK installation process.
Plug the SN8000 EVB into the development PC with a USB cable, the driver should automatically load.

2.4.3 Verifying USB Driver Installations

Verify that installation of the drivers was successful by checking the Device Manager window. Follow the steps below to open the Device Manager window:

1. For XP, select **Windows Start Button->Control Panel->System->Hardware->Device Manager**. For Win 7, select **Windows Start Button->Control Panel->Hardware and sound->Device Manager**.
2. The A device (WICED USB JTAG Port) should be under *<computer-name>\SN8200 USB-JTAG Device* as shown in the screen capture below.
3. The B device (WICED USB Serial Port) should be under *<computer-name>\Ports (COM & LPT)* as shown in the screen capture below.
4. Take note of the USB serial COM port number for later use. Your SN8000 EVB USB serial COM port will most likely be assigned to a different port number than shown in the screen capture.

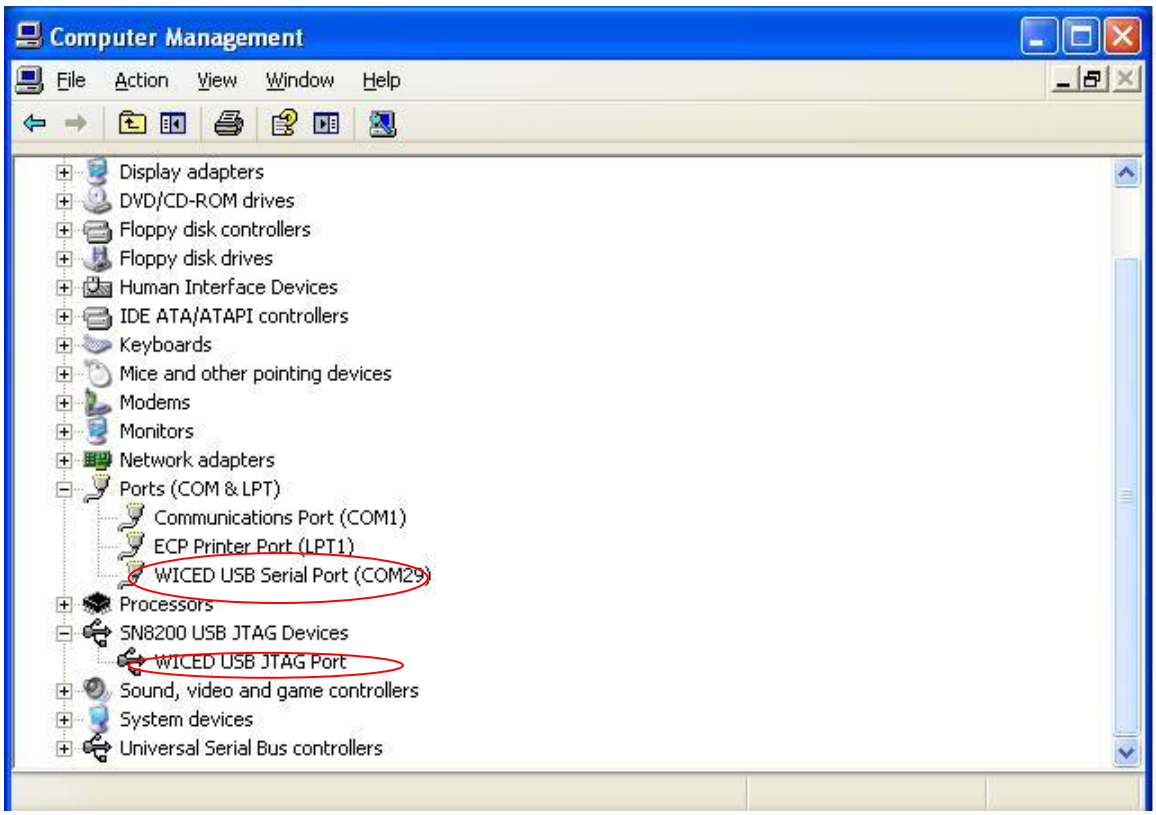


Figure 5 Device Manager screen showing installed SN8000/SN8200 serial and JTAG drivers

2.4.4 Uninstalling the USB drivers

If for any reason it is desired to uninstall the USB drivers, go to **Control Panel->Add Remove Programs** (for XP) or **Control Panel->Uninstall a program** (for Win 7) to remove the three drivers installed.

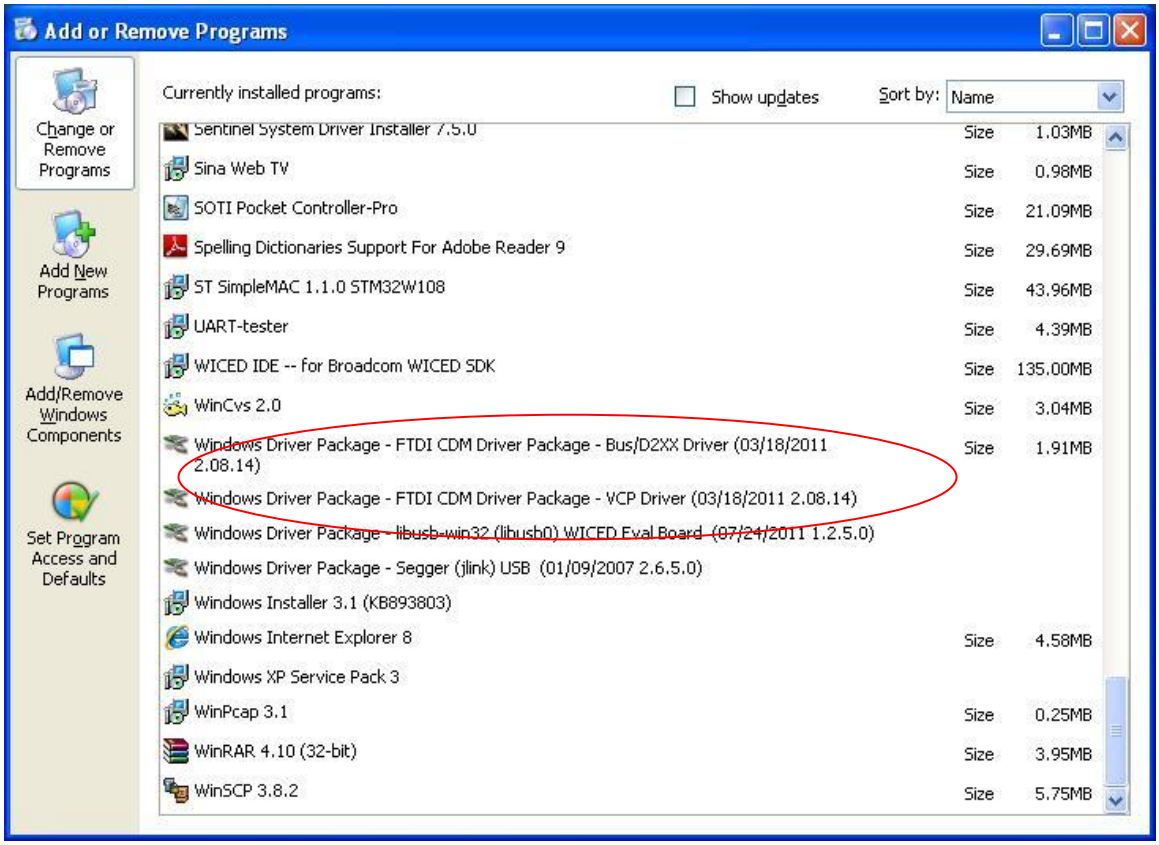


Figure 6 Uninstalling USB drivers

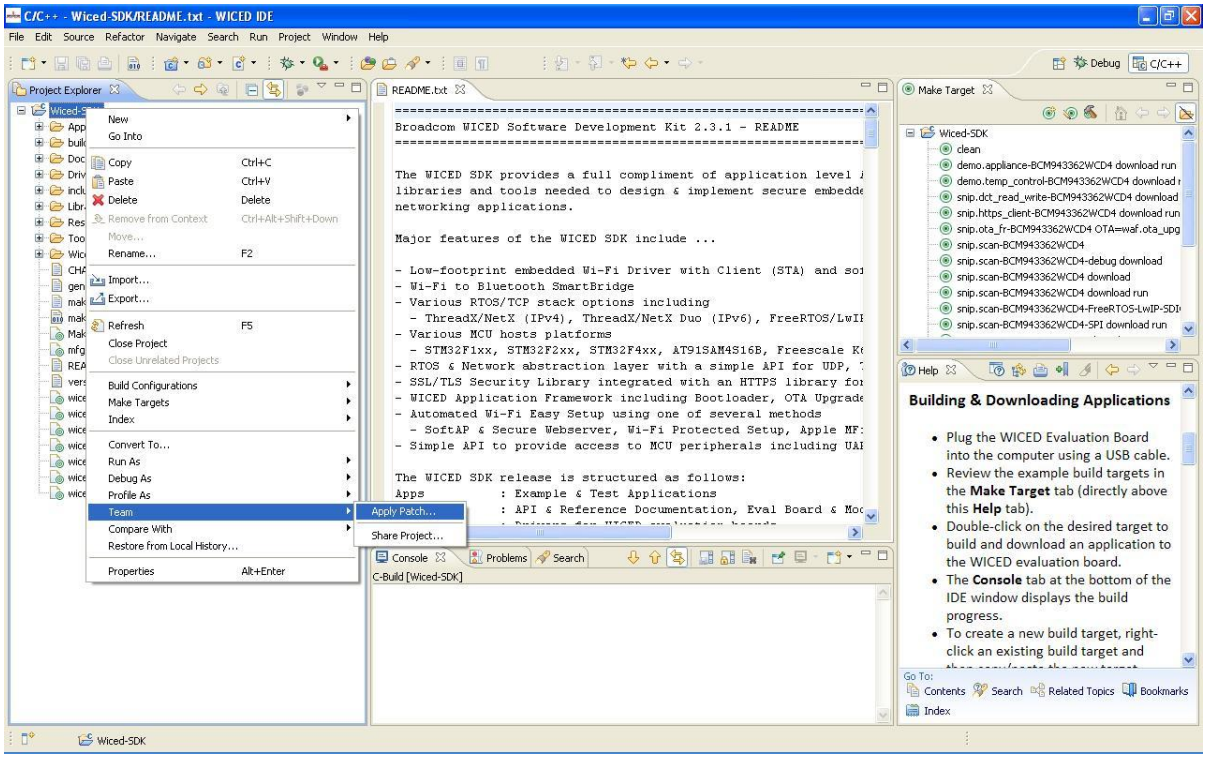
3. Using the WICED IDE

A patch file has been created to simplify the porting of WICED SDK to the SN8000 EVB. Please contact modules@murata.com to obtain the patch file. Ensure that the WICED SDK version being used matches that supported by the patch file.

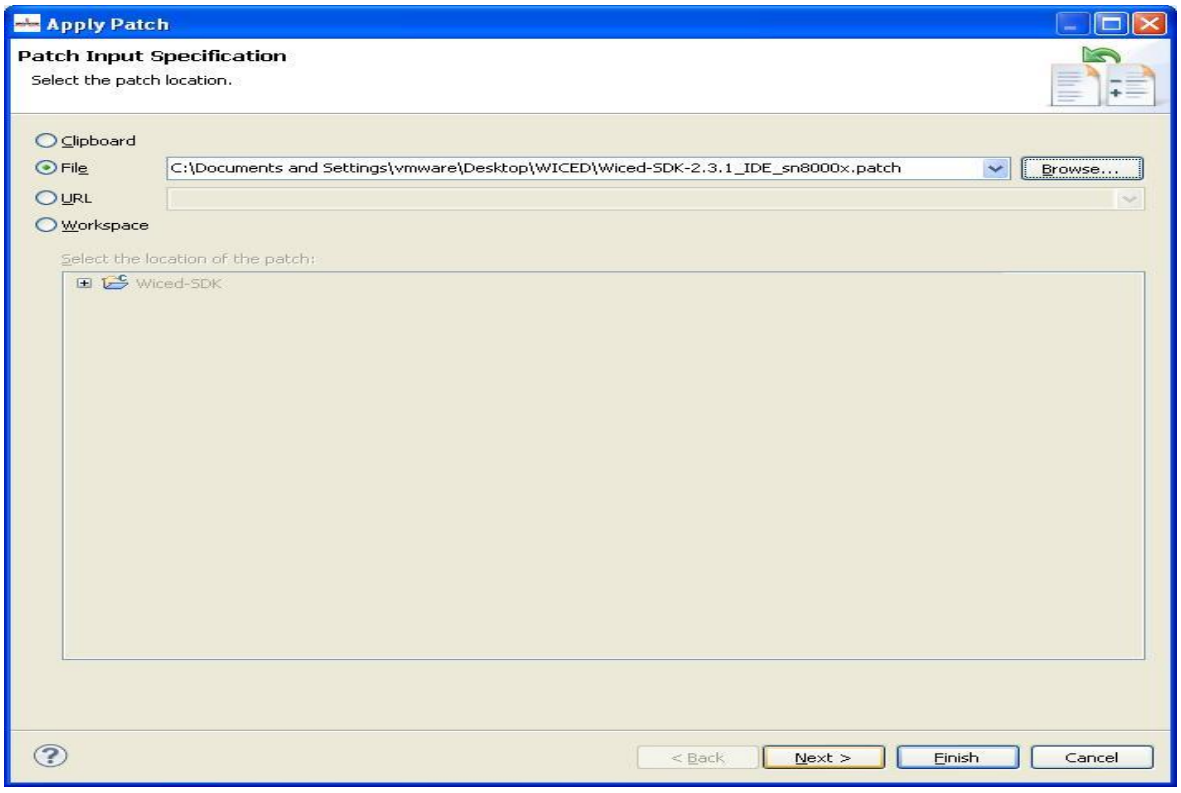
3.1 Start IDE and apply the patch

The WICED IDE should have been started during previous installation process, or in any cases user has closed it, re-launch the WICED IDE by selecting **START** > **All Programs** > **Broadcom** > **WICED IDE**.

After startup, use **Project Explorer** tab on the left of the IDE, right-click on Wiced-SDK project, and then in the menu select Team->Apply Patch, as shown below.

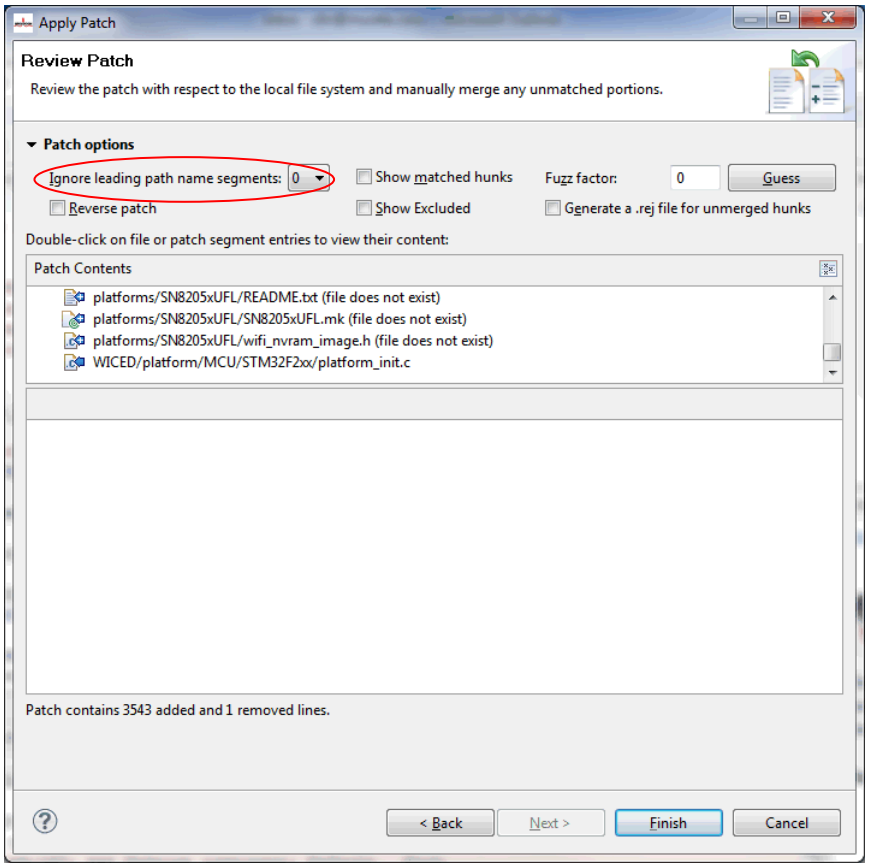


On the next window of Patch Input Specification, browse for the SN8000 patch file. Once the location is selected, click “Next” to proceed.



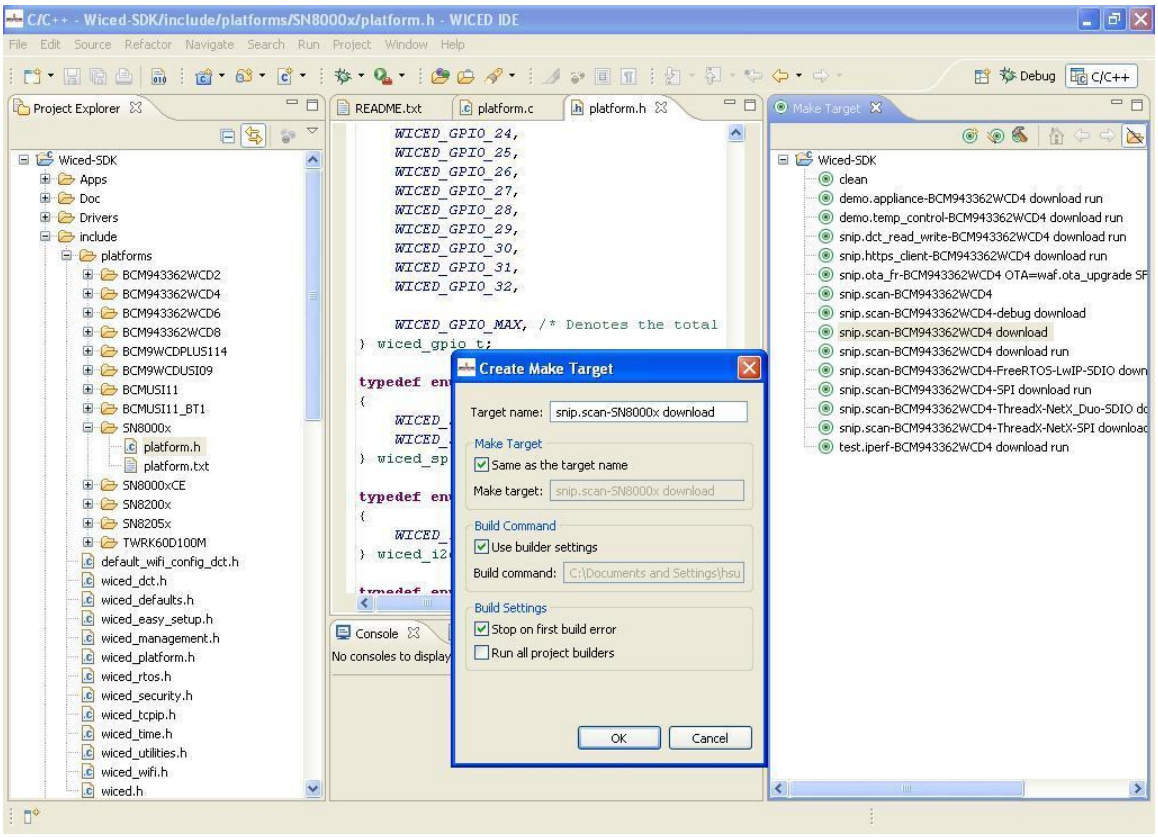
On the next window of Target Resource, keep the default setting and click on “Next”.
On the next window of Review Patch, under “Patch options” set Ignore leading path name segments to be ‘0’, then click “Finish”.

Upon this step, the WICED SDK project will be updated with SN8000 platform profiles.



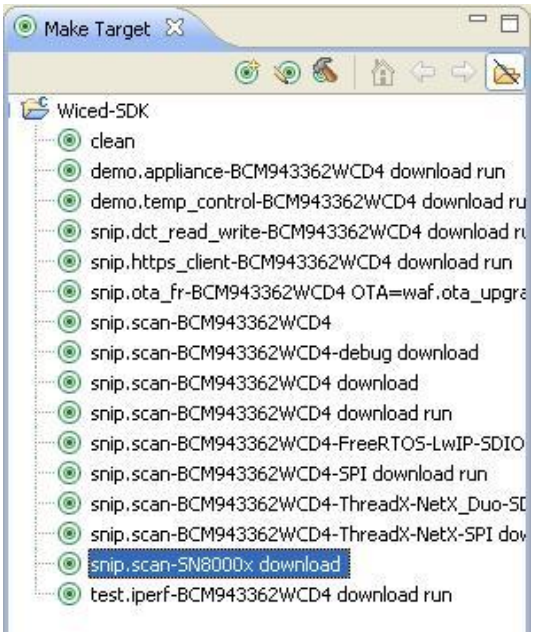
3.2 Create the SN8000 make target

After patch applied, the SN8000x headers and platform files should occur in the updated Wiced-SDK project. Now use **Make Target** tab on the right top of the IDE. Pick up an existing target, i.e., snip.scan-BCM943362WCD4 download, right click on it and copy/paste the target. The “Create Make Target” window will pop up. Rename the target as snip.scan-SN8000x download.



3.3 Build and download the application

The new created target "snip.scan-SN8000x download" should appear in the updated Wiced-SDK target list. Double click on it to build the scan application based on SN8000 platform, the IDE console window displays the build progress.



The build target is appended with the 'download' and 'run' options, these options tell the toolchain to download the firmware and run the application after the build completes.
The build output looks similar to the following:

```
**** Build of configuration Default for
project Wiced-SDK ****

C:\My Documents\WICED\Wiced-SDK-2.3.1\Wiced-
SDK\make.exe snip.scan-SN8000x download
Building Bootloader

Building App

Making .gdbinit
Creating security credentials
Converting resources
```

```
Making DCT image
Compiling App_Scan
Compiling Platform_SN8000x
Compiling Wiced
Compiling STM32F2xx
Compiling common_GCC
Compiling WWD_ThreadX_Interface
Compiling WWD_NetX_Duo_Interface
Compiling Wiced_ThreadX_Interface
Compiling Wiced_NetX_Interface
Compiling Supplicant_besl
Compiling Lib_http_server
Compiling Lib_dns_redirect_daemon
Compiling Lib_dns
Compiling WWD_for_SDIO_ThreadX
Compiling STM32F2xx_Drv
Compiling SPI_Flash_Library_SN8000x
Compiling Lib_dhcp_server
Compiling Wiced_Wifi_image
Making snip_scan-SN8000x.elf
Making snip_scan-SN8000x.bin
Build complete

snip_scan-SN8000x
----------------------------------|---------|---------|
                                  |         |  Static |
             Module               |  Flash  |   RAM   |
----------------------------------+---------+---------|
App                               |     905 |       4 |
Bootloader                        |     133 |       0 |
Host MCU-family library           |   11852 |    2626 |
Interrupt Vectors                 |     660 |       0 |
libc                              |     180 |       4 |
Networking                        |     385 |   23584 |
NetX                              |    1348 |      92 |
Other                             |   23980 |    2260 |
Platform                          |     668 |       0 |
RAM Initialisation                |    2252 |       0 |
Startup Stack & Link Script fill  |     434 |     844 |
Supplicant - BESL                 |      48 |       8 |
ThreadX                           |    7088 |     392 |
Wi-Fi Firmware                    |  191684 |       0 |
Wiced                             |    2249 |     773 |
WWD                               |   10834 |    1037 |
----------------------------------+---------+---------|
TOTAL (bytes)                     |  254700 |   31624 |
----------------------------------|---------|---------|

Downloading Bootloader ...
Download complete

Downloading Application ...
Download complete

Downloading DCT ...
```

```
Download complete
```

Note: If the EVB is not recognized by the programming tools, it may be necessary to disconnect and then reconnect the board to the computer before trying again. The following message indicates there was an error with the download process:

```
"**** OpenOCD failed - ensure you have installed the driver from the drivers
directory, and that the debugger is not running **** In Linux this may be due
to USB access permissions. In a virtual machine it may be due to USB
passthrough settings ****"
Resetting target
make: *** [run] Error 1
```

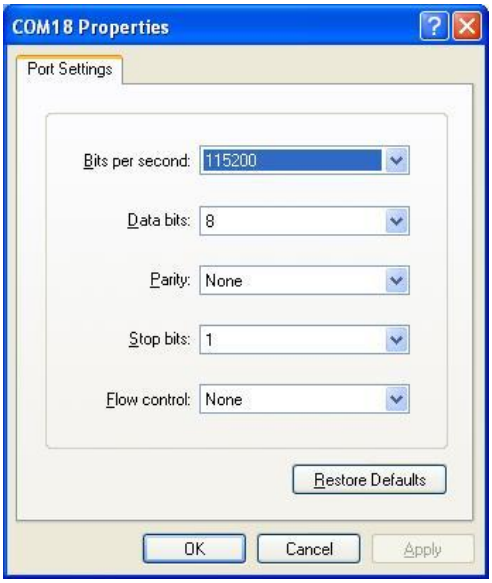
3.4 Run the application

This section assumes you have successfully completed previous section, and the scan application is running on the SN8000 EVB.

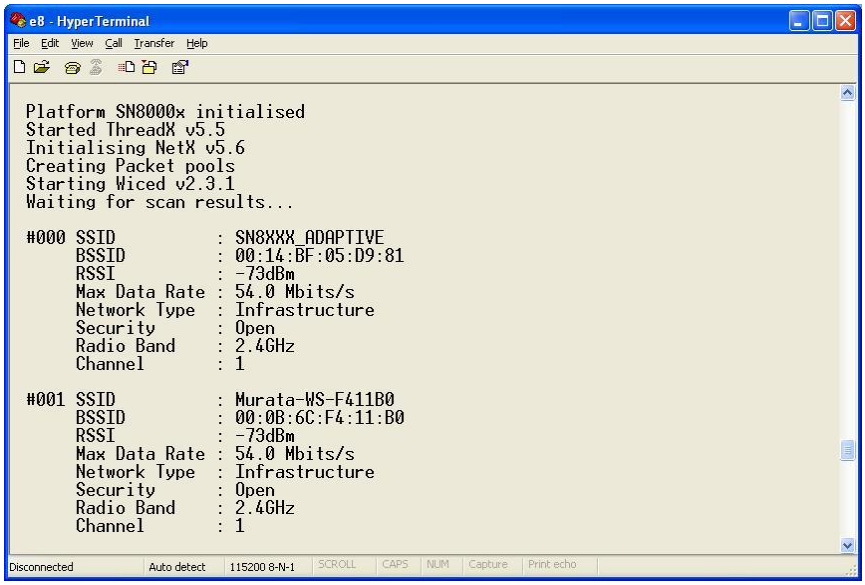
After power-on-reset, the scan application prints status messages to the USB UART of the SN8000 EVB.

To verify printing, launch a terminal application such as Microsoft® Hyper Terminal. (For Windows XP, it's located at **START** > **All Programs** > **Accessories** > **Communications** > **Hyper Terminal**)

Use following settings on COM port connected (in this example it's COM 18):



Start the terminal application then press the reset button on the board. The texts are expected to appear as shown below:



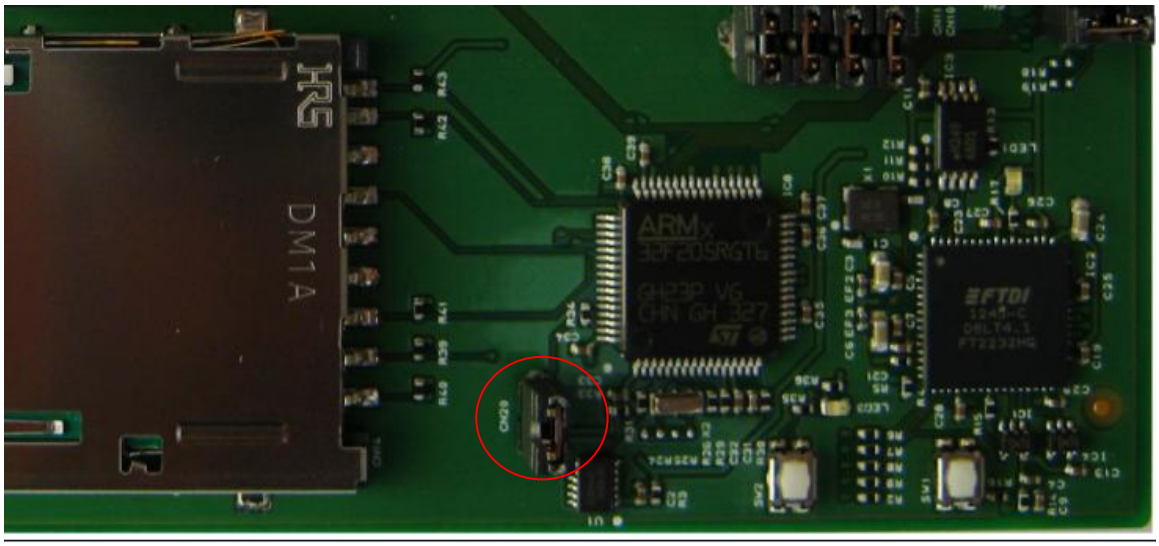
3.5 Debug the application

For detailed debug procedure, please refer to Broadcom WICED Quick Start Guide Sec. 4.3.

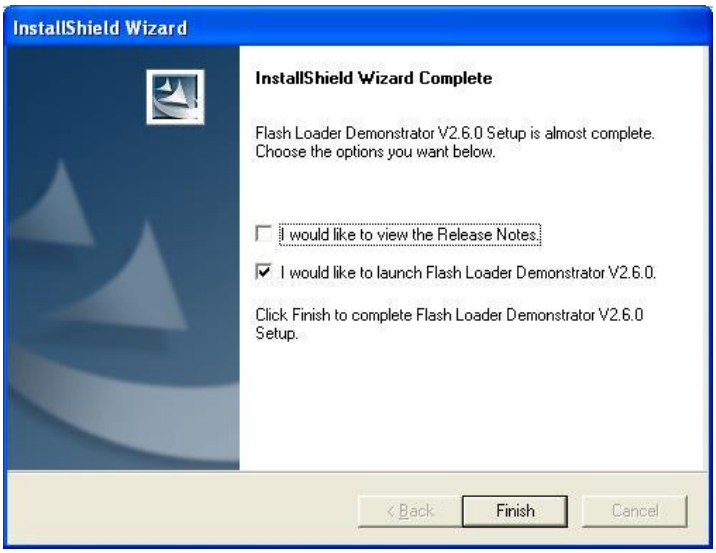
Appendix. Restore the pre-load firmware

Once the user has used the WICED SDK to build and download an image to the SN8000 EVK, the preloaded image in the STM32 is erased. The following procedure may be used to reload the original firmware through the serial port.

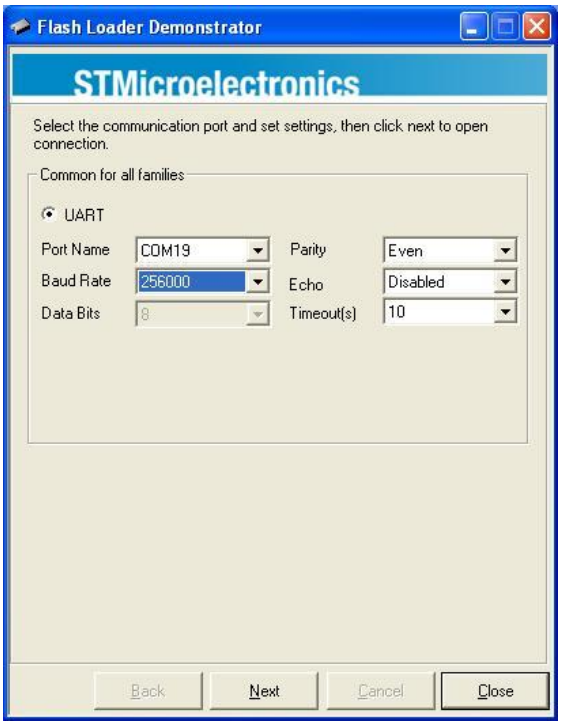
1. Place the STM32 onboard the SN8000 EVB into boot mode to prepare to load image through USART1.
    a. Unplug the mini-USB cable from the SN8000 EVB. Configure the EVB to be powered by the USB.
    b. Pull up STM BOOT0 by placing the jumper on CN20.
    c. Reconnect the mini-USB cable to the SN8000 EVK. Press and release reset button.



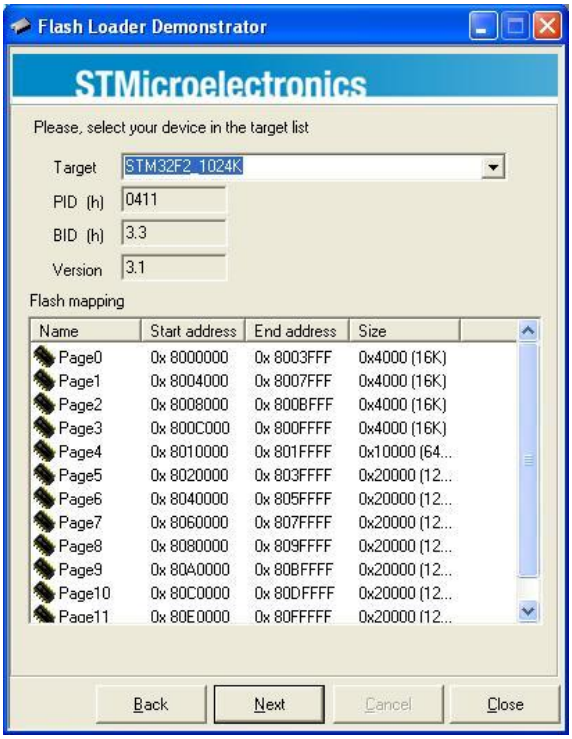
2. Download and install the tool "STM32 and STM8 Flash loader demonstrator" from ST website (www.st.com) to the PC. The keyword to search is "flash loader". The snapshots shown below are based on STMSW-MCU005 (version 2.6.0). More details can be found in [5].
3. Double click the installer in the package. Once the installation process is completed, check the option to launch the program then click on finish.



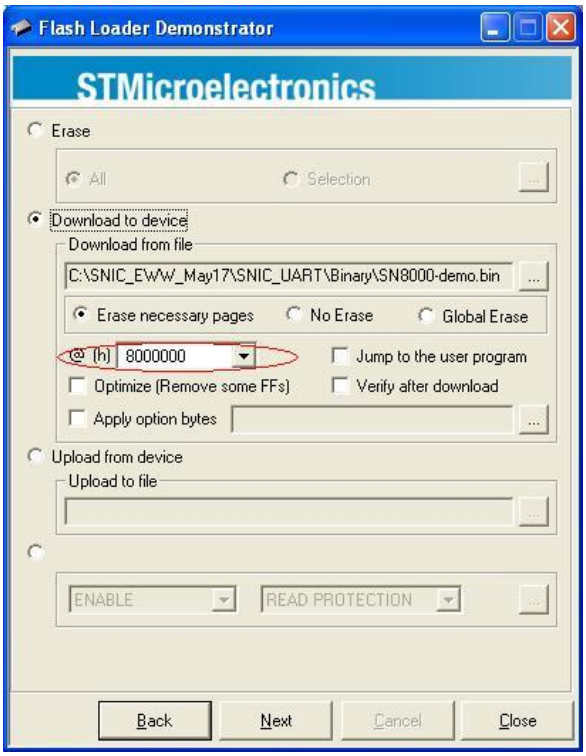
4. Some configurations need to be specified to the tool, as shown below:
    a. For UART setting, the COM port should be same as shown in Section 2.4.3. Timeout is set as 10 sec, but user can adjust to a bigger interval if needed, e.g., if the erase phase exceeds 10 sec to finish, use a larger value to avoid the erase failure. The baud rate may also be changed to a larger value. Click Next when done.

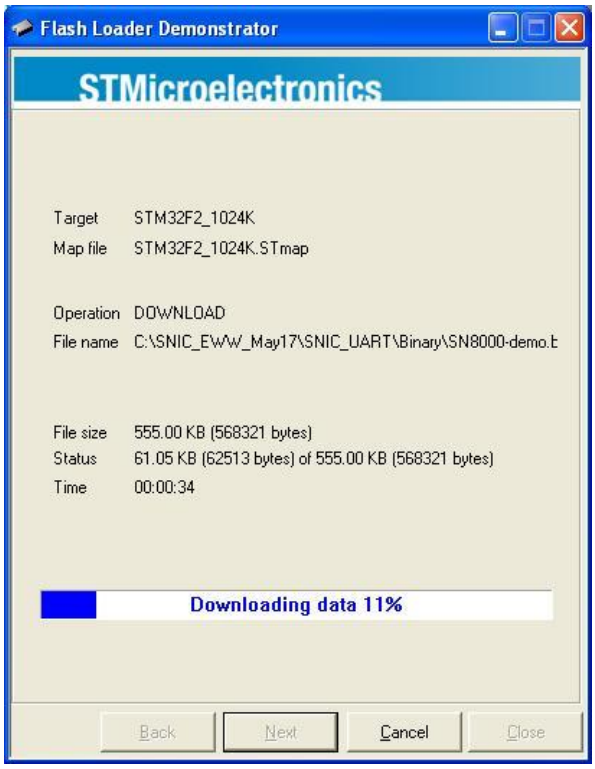


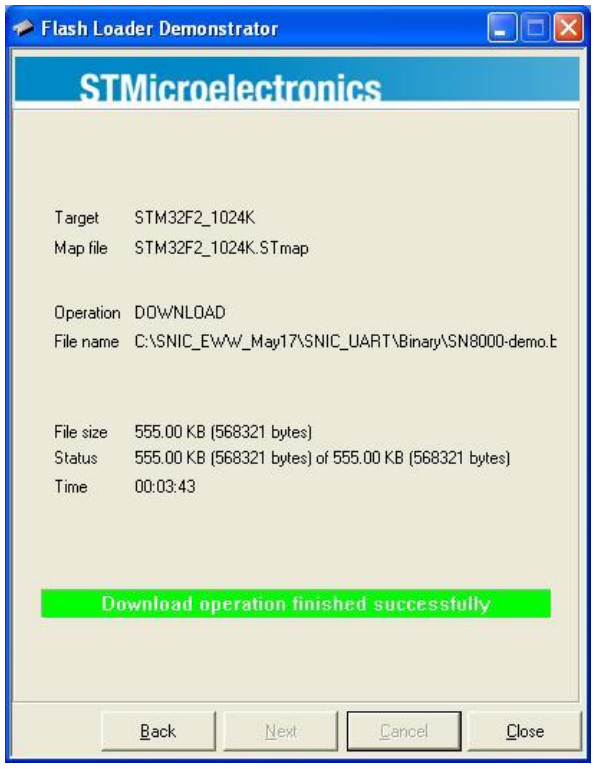
b. Select the correct STM target, i.e., STM32F2_1024K for STM32F205RG.



c. When selecting the target bin file, note that the starting address MUST be 0x8000000. Click Next to start FW download.




Flash Loader Demonstrator
STMicroelectronics
Target STM32F2_1024K
Map file STM32F2_1024K.STmap
Operation DOWNLOAD
File name C:\SNIC_EWW_May17\SNIC_UART\Binary\SN8000-demo.b
File size 555.00 KB (568321 bytes)
Status 61.05 KB (62513 bytes) of 555.00 KB (568321 bytes)
Time 00:00:34
Downloading data 11%
Back
Next
Cancel
Close


Flash Loader Demonstrator
STMicroelectronics
Target STM32F2_1024K
Map file STM32F2_1024K.STmap
Operation DOWNLOAD
File name C:\SNIC_EWW_May17\SNIC_UART\Binary\SN8000-demo.b
File size 555.00 KB (568321 bytes)
Status 555.00 KB (568321 bytes) of 555.00 KB (568321 bytes)
Time 00:03:43
Download operation finished successfully
Back
Next
Cancel
Close

5. Once the download is successful, close the tool and disconnect the mini-USB cable. Drive STM32F BOOT0 pin back to low by removing the jumper from CN20 to enable the firmware to run upon subsequent power up.

(END)