WICED™ Application Development with SN82XX EVK

Version: 1.1

Release Date: October 10, 2014

Release Record

Version Number	Release Date	Comments
Version 0.1	10/03/2013	• Initial draft.
Version 1.0	02/04/2014	• Initial release
Version 1.1	10/10/2014	• Updated "ignore leading path name segment" to 0

THE TABLE OF CONTENTS

1. INTRODUCTION 5

1.1 ACRONYMS 5
1.2 REFERENCES 5

2. SETTING UP THE WICED™ SDK 6

2.1.1 Install the WICED™ SDK and IDE 6
2.1.2 Connect the SN82XX Evaluation Board 7
2.1.3 Verifying USB Driver Installations 7
2.1.4 Uninstalling the USB drivers 8

3. USING THE WICED™ IDE 9

3.1 START IDE AND APPLY THE PATCH 9
3.2 CREATE THE SN82XX MAKE TARGET 12
3.3 BUILD AND DOWNLOAD THE APPLICATION 13
3.4 RUN THE APPLICATION 15
3.5 DEBUG THE APPLICATION 16

TABLE of figures

Figure 1 SN82XX EVK Configuration 7
Figure 2 Device Manager showing installed USB serial and JTAG drivers.....8
Figure 3 Uninstalling USB drivers 8

1. Introduction

SN82XX is a family of modules which can be used to develop low power embedded wireless solutions to address the connectivity demand in home appliances and other applications. Each module integrates an ARM Cortex M3 micro-controller, WiFi BB/MAC/RF IC, RF front end, flash memory and high speed clocks into a small form factor module. The respective SN82XX 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 PC development platform.

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, "SN8200 SNIC EVK+ User Guide"
[2] Murata, "SN8205 SNIC EVK+ User Guide"
[3] Broadcom WICED™ architecture (http://www.broadcom.com/support/WICED/)

2. Setting up the WICED™ SDK

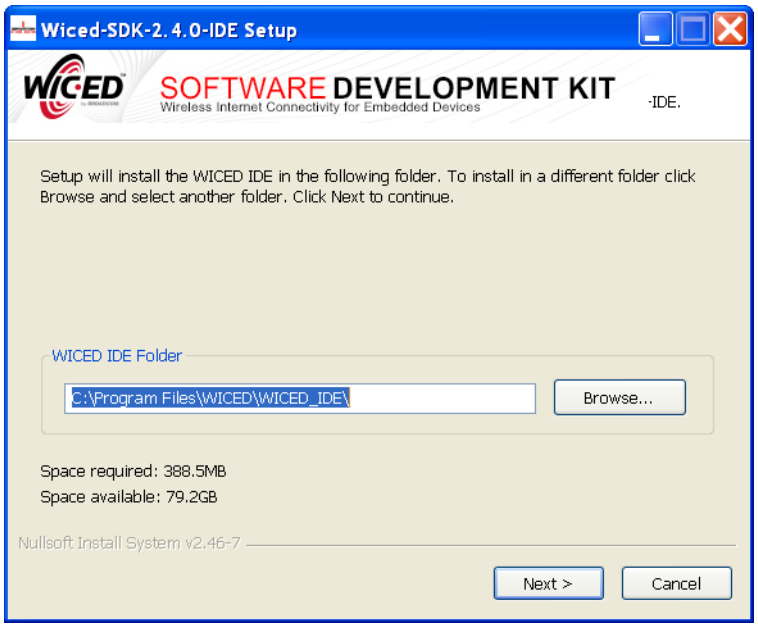
Users can develop their own applications on SN82XX EVB by using Broadcom® WICED™ SDK and IDE. To use the SDK, the following steps must be performed:

1. Download and install the WICED™ SDK and IDE
2. Apply the SN82XX platform patches
3. Create and compile an application
4. Load the application into the SN82XX 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.

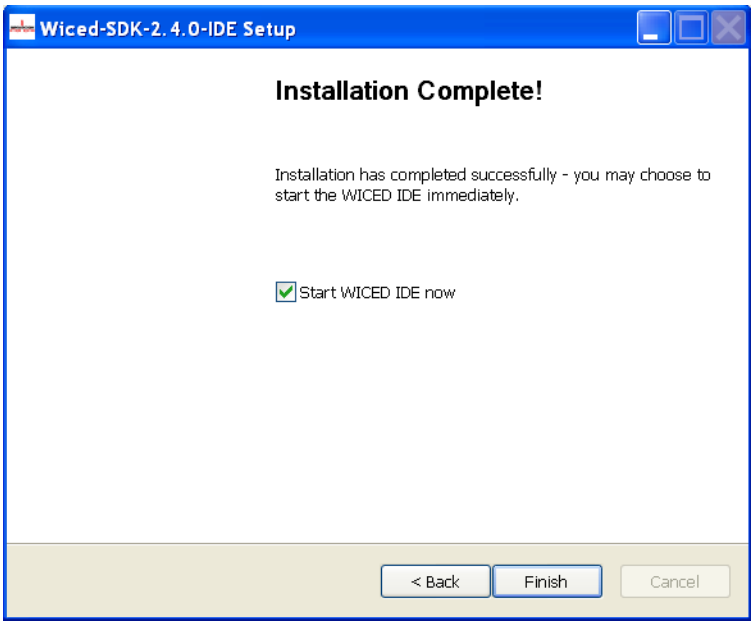
2.1.1 Install the WICED™ SDK and IDE

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-x.x.x-IDE-Installer.exe file to begin the installation. A setup window similar to the following appears.



Note: ensure that the SN82XX 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.



2.1.2 Connect the SN82XX Evaluation Board

The SN82XX EVK may be used to debug and test the WICED™ applications. The user is assumed to be familiar with the particular SN82XX EVK that is being used. Please see the respective EVK user guides for more details ([1][2]).

The SN82XX EVB connects to the PC through USB. The USB interface provides power as well as individual JTAG and UART interfaces to the STM32 onboard the SN82XX module.

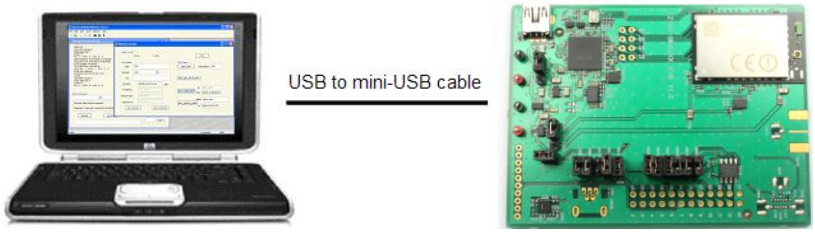


Figure 1 SN82XX EVK Configuration

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 SN82XX EVB can be loaded properly.

The SN82XX 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 SN82XX EVB into the development PC with a USB cable, the driver should automatically load.

2.1.3 Verifying USB Driver Installations

Verify that installation of the drivers was successful by checking the Device Manager. Follow the steps below to open the Device Manager window. If any driver is missing, try rebooting the computer.

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>\Broadcom 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 SN82XX EVB USB serial COM port will most likely be assigned to a different port number than shown in the screen capture.

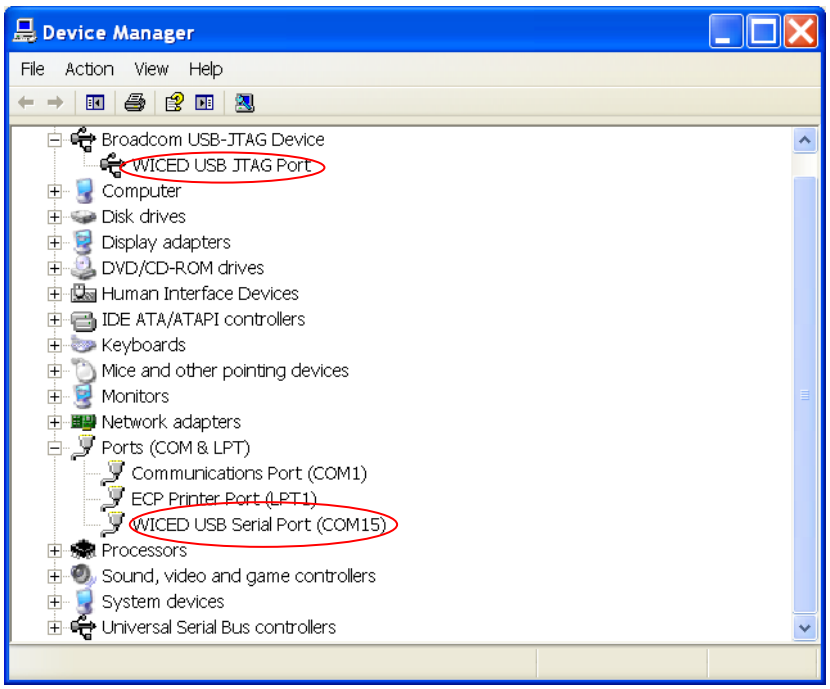


Figure 2 Device Manager showing installed USB serial and JTAG drivers

2.1.4 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.

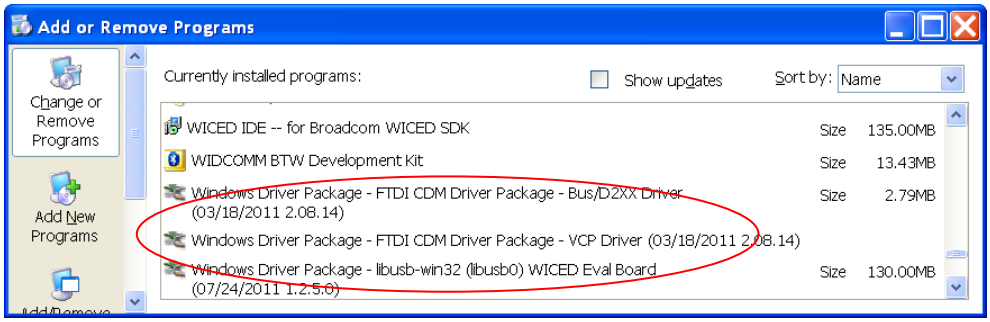


Figure 3 Uninstalling USB drivers

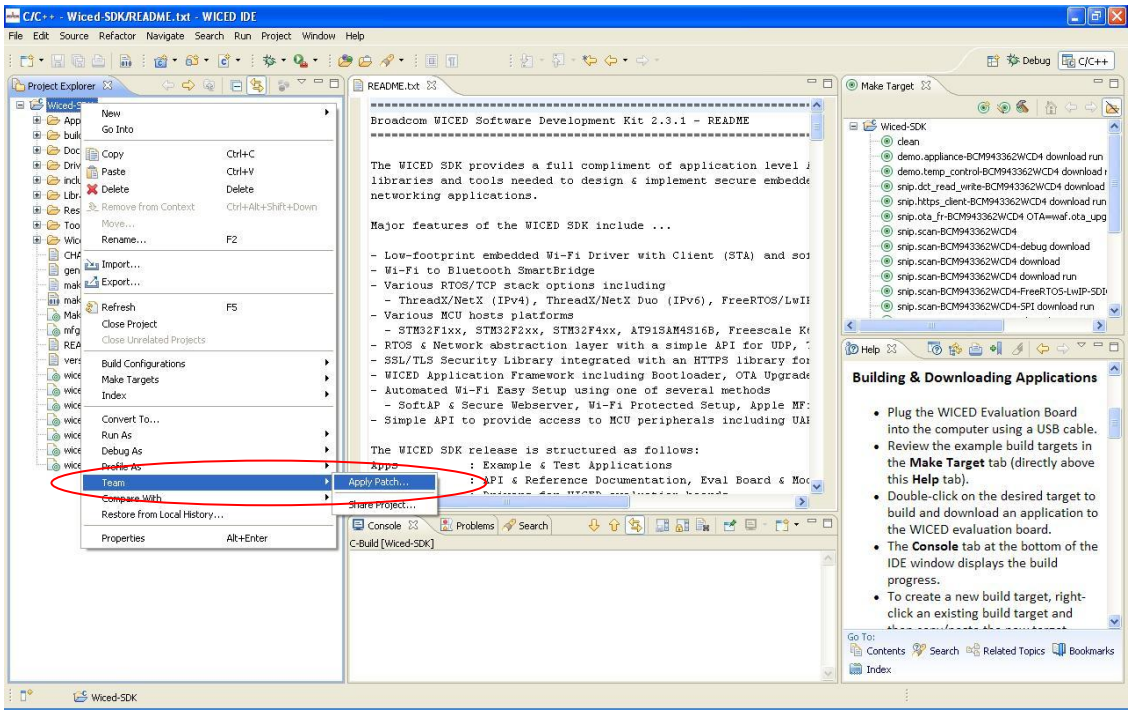
3. Using the WICED™ IDE

A patch file has been created to simplify the porting of WICED™ SDK to the SN82XX 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.

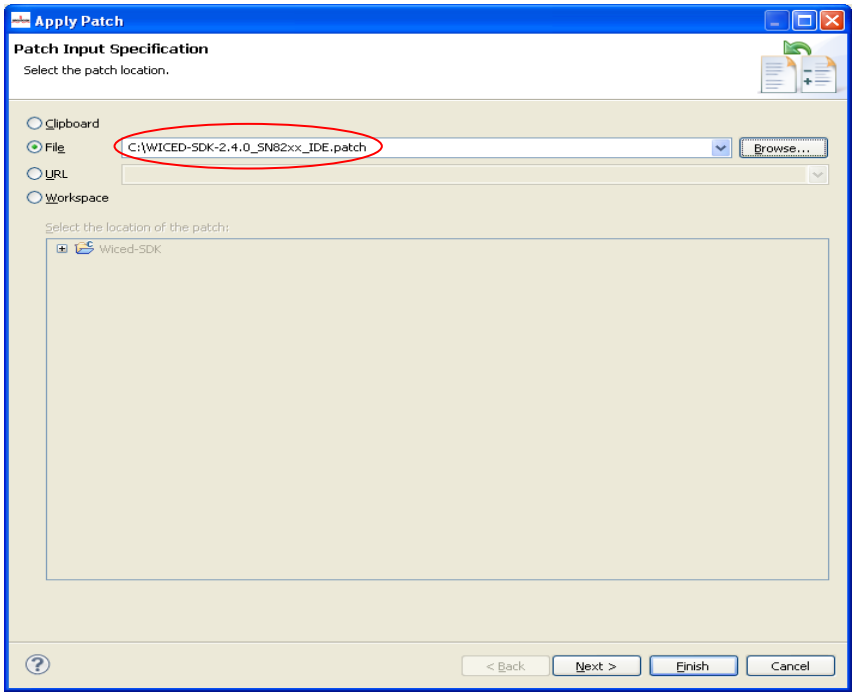
3.1 Start IDE and apply the patch

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

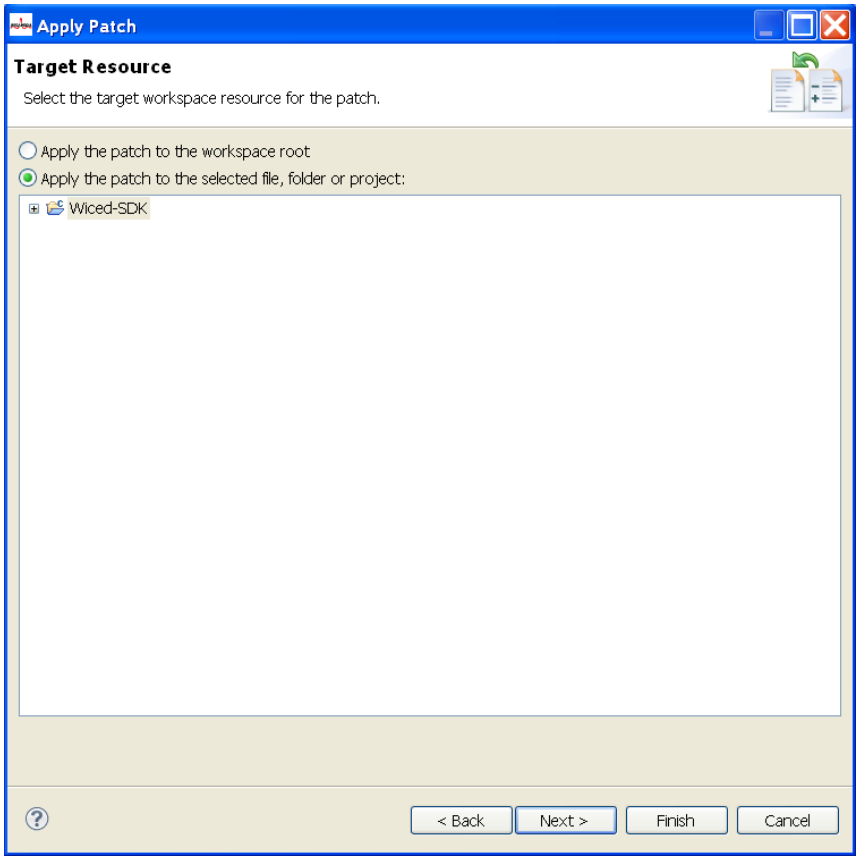
After startup, use **Project Explorer** tab on the left of the IDE, right-click on WICED-SDK project, and then in the menu select **Team > Apply Patch**, as shown below.



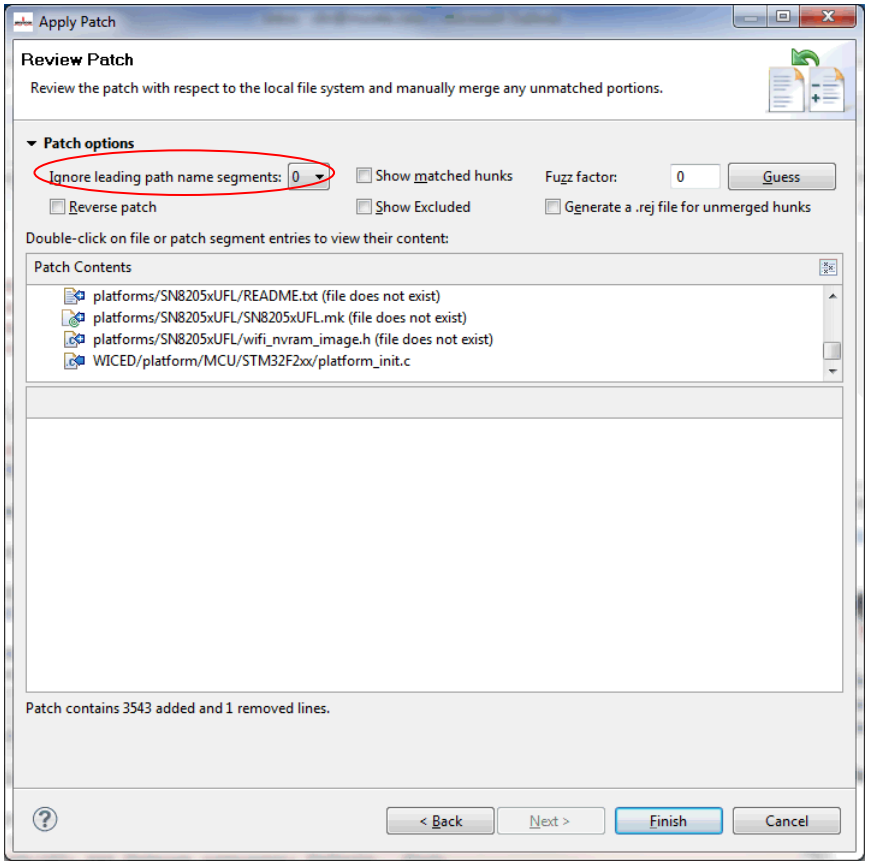
On the next window of Patch Input Specification, browse for the SN82XX patch file (Note: the name of the file may be different than shown in the screenshots). Once the location is selected, click "Next" to proceed.



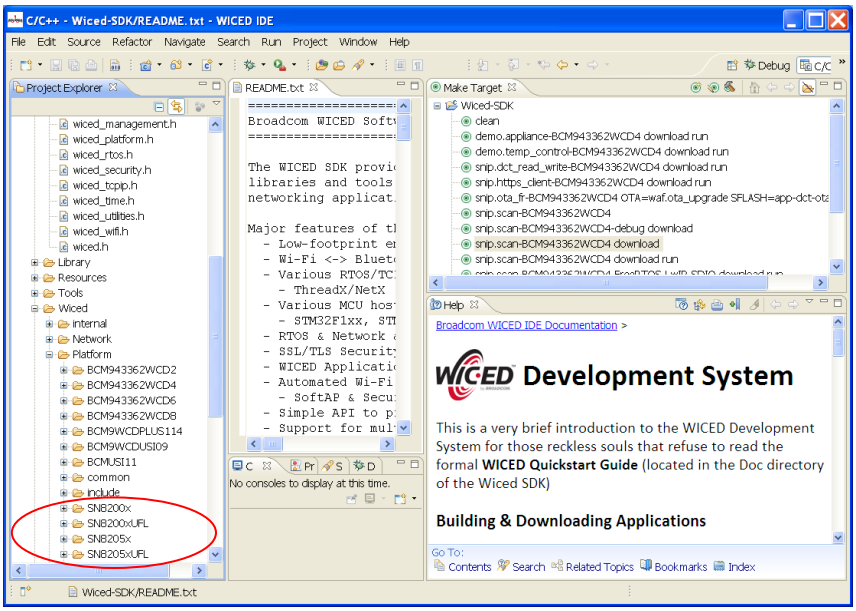
On the next window of Target Resource, keep the default setting and click on “Next”.



On the next window of Review Patch, under “Patch options” set “Ignore leading path name segments” to be ‘0’, then click “Finish”.

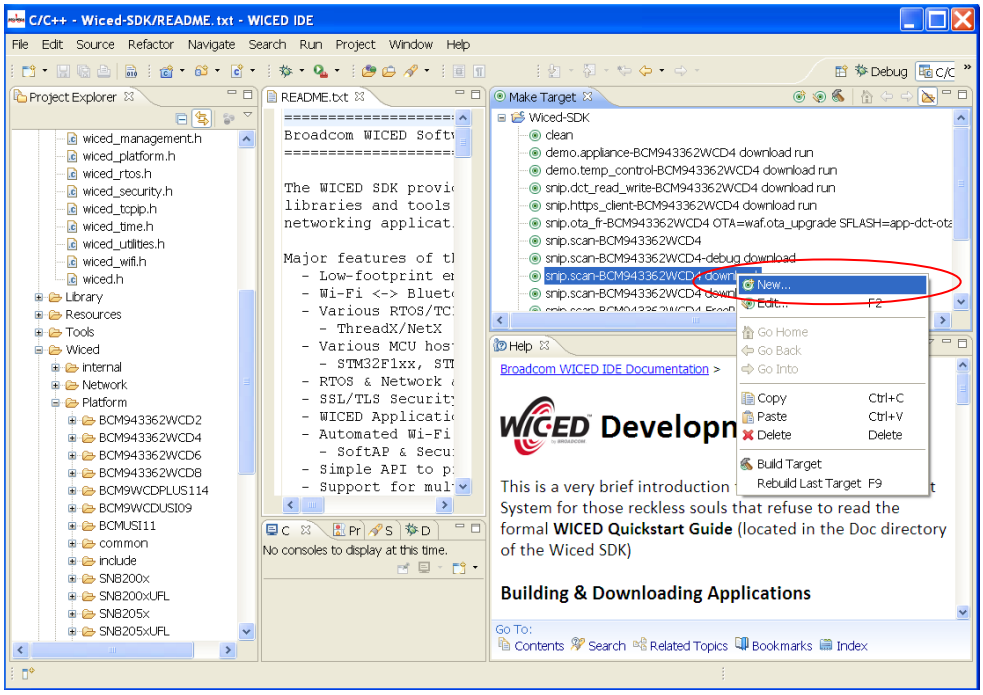


After this step, the WICED SDK project will be updated with SN82XX platform profiles. They are listed under “Project Explorer.Wiced-SDK.Wiced”.

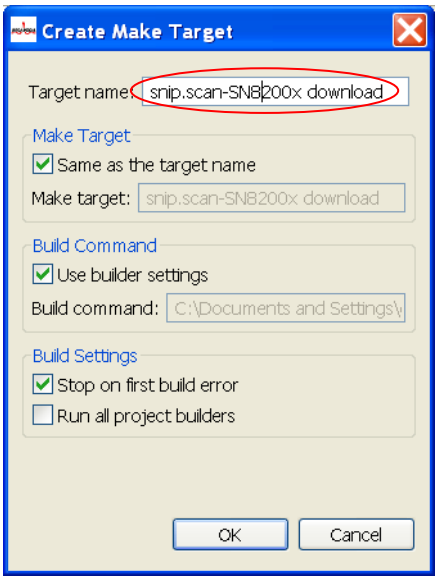


3.2 Create the SN82XX make target

After the patch is applied, the SN82XX headers and platform files should be present in the updated WICED-SDK project. Now use **Make Target** tab on the upper right corner of the IDE to create your own make rule. Select an existing target (e.g., "snip.scan-BCM943362WCD4 download"), right click on it and choose the "**New**" option to bring up the "**Create Make Target**" panel.

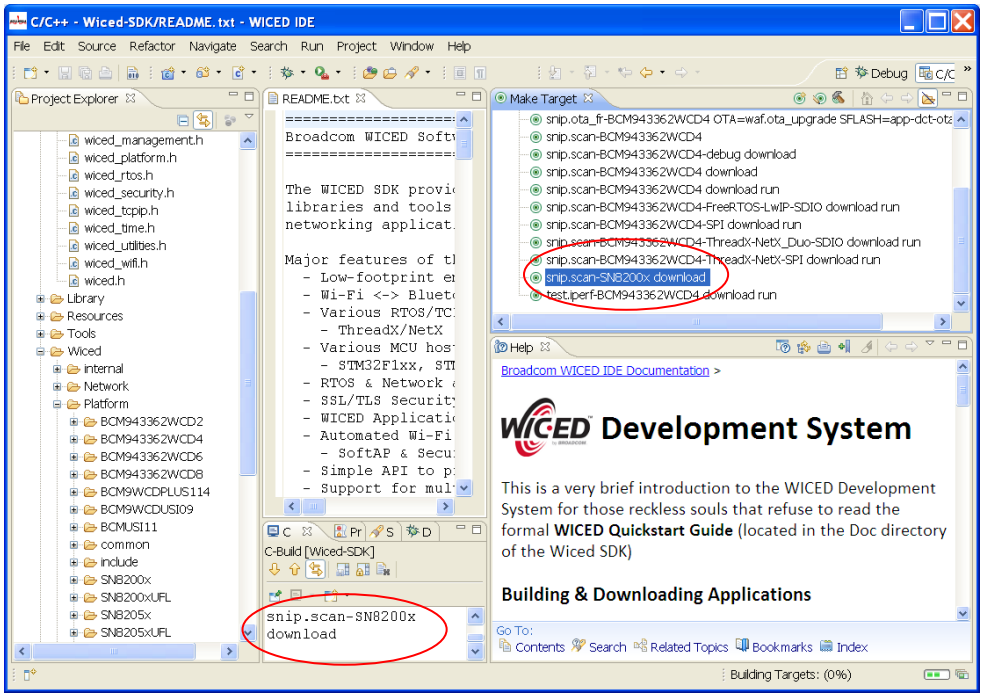


Rename the target to be of form “snip.scan-<PLATFORM> download”, e.g., “snip.scan-SN8200x download” for the SN8200x platform. Ensure that the name of the target is exactly the same as that specified in the “Project Explorer” panel and matches the Murata module type.



3.3 Build and download the application

The newly created target “snip.scan-SN8200x download” should appear in the updated WICED-SDK target list. Double click on it to build the scan application based on SN8200x platform; the IDE console window displays the build progress.



The build target may be appended with the ‘download’ and ‘run’ options. Those 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:\Documents and Settings\Dev\My Documents\WICED\Wiced-SDK-2.4.0\Wiced-SDK\make.exe
snip.scan-SN8200x download
Making config file for first time
+-------------------------------------------------------------------------------------------
-------------+
| IMPORTANT NOTES
|
+-------------------------------------------------------------------------------------------
-------------+
| Wi-Fi MAC Address
|
|    The target Wi-Fi MAC address is defined in <WICED-SDK>/generated_mac_address.txt
|
|    Ensure each target device has a unique address.
|
+-------------------------------------------------------------------------------------------
-------------+
| MCU & Wi-Fi Power Save
|
|    It is *critical* that applications using WICED Powersave API functions connect an
accurate 32kHz |
|    reference clock to the sleep clock input pin of the WLAN chip. Please read the WICED
Powersave   |
|    Application Note located in the documentation directory if you plan to use powersave
features.   |
+-------------------------------------------------------------------------------------------
-------------+
Building Bootloader

Building App

Making .gdbinit
Converting resources
Creating security credentials
Making DCT image
Compiling App_Scan
Compiling Platform_SN8200x_ThreadX
Compiling Wiced
Compiling STM32F1xx_lib
Compiling WWD_ThreadX_Interface
Compiling Wiced_ThreadX_Interface
Compiling WWD_NetX_Duo_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 common_GCC
Compiling STM32F1xx_Drv
Compiling SPI_Flash_Library_SN8200x
Compiling Lib_dhcp_server
Compiling Wiced_Wifi_image
Making snip_scan-SN8200x.elf

Making snip_scan-SN8200x.bin
snip_scan-SN8200x
---------------------------------|---------|---------|
                                 |         |  Static |
              Module             |  Flash  |   RAM   |
```

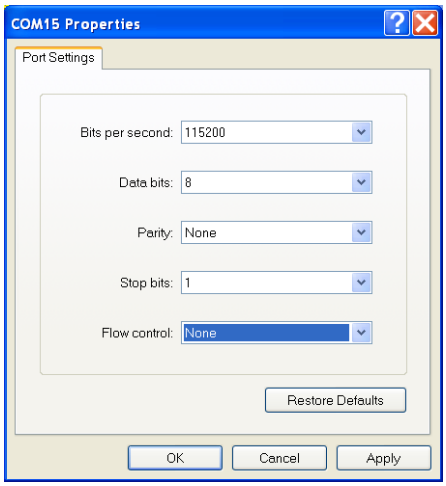
```
-----------------------------------+---------+---------|
App                                |     416 |       8 |
Bootloader                         |     133 |       0 |
Host MCU-family library            |    9256 |    2629 |
Interrupt Vectors                  |     512 |       0 |
libc                               |     180 |       4 |
Networking                         |     649 |   21232 |
NetX                               |    3168 |      92 |
Other                              |   23316 |    2260 |
Platform                           |     692 |       0 |
RAM Initialisation                 |    2252 |       0 |
Startup Stack & Link Script fill   |     433 |     839 |
Supplicant - BESL                  |      56 |      12 |
ThreadX                            |    7864 |     392 |
Wi-Fi Firmware                     |  203268 |       0 |
Wiced                              |    4949 |     822 |
WWD                                |   14048 |    1042 |
-----------------------------------+---------+---------|
TOTAL (bytes)                      |  271192 |   29332 |
-----------------------------------|---------|---------|

Build complete
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:

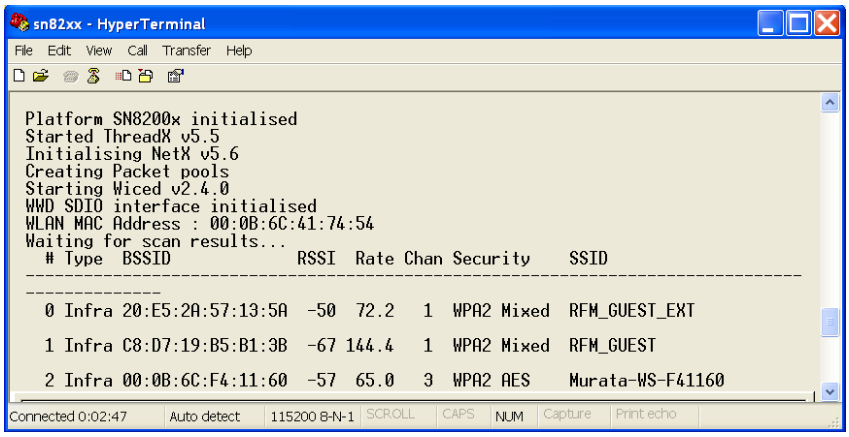
```
"**** OpenOCD failed - ensure you have installed the driver from the drivers directory,
and that the debugger is not running **** In Linux this may be due to USB access
permissions. In a virtual machine it may be due to USB passthrough settings ****"
Resetting target
make: *** [run] Error 1
```

3.4 Run the application

This section assumes you have successfully completed the previous procedure and the scan application has been downloaded to the EVB. To verify the application, launch a terminal application such as Microsoft® Hyper Terminal. For Windows XP, it's located at **START** > **All Programs** > **Accessories** > **Communications** > **Hyper Terminal**. Use the following settings (115.2kbps 8N1) for the COM port connected (in this example it's COM 15):

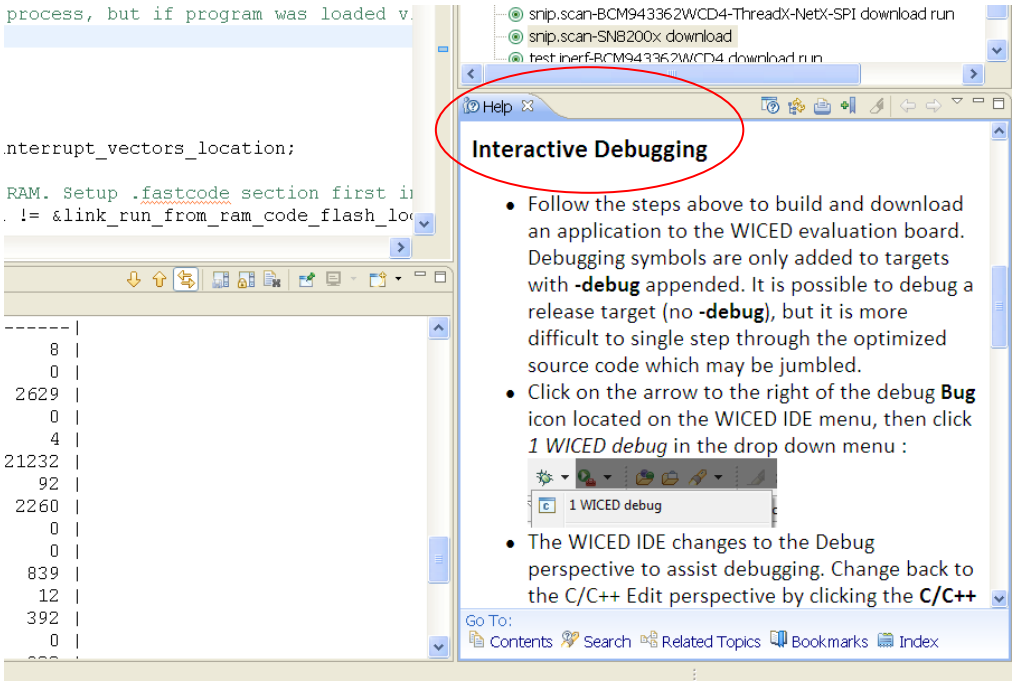


Start the terminal application then press the reset button on the board. Text similar to the following should appear on the screen:



3.5 Debug the application

For detailed debug procedure, please refer to Broadcom WICED™ Quick Start Guide or consult the **Help** screen in the lower right corner of the WICED™ IDE.


process, but if program was loaded v
nterrupt_vectors_location;
RAM. Setup .fastcode section first i
!= &link_run_from_ram_code_flash_lo
snip.scan-BCM943362WCD4-ThreadX-NetX-SPI download run
snip.scan-SN8200x download
Help
Interactive Debugging
Follow the steps above to build and download an application to the WICED evaluation board. Debugging symbols are only added to targets with -debug appended. It is possible to debug a release target (no -debug), but it is more difficult to single step through the optimized source code which may be jumbled.
Click on the arrow to the right of the debug Bug icon located on the WICED IDE menu, then click 1 WICED debug in the drop down menu :
1 WICED debug
The WICED IDE changes to the Debug perspective to assist debugging. Change back to the C/C++ Edit perspective by clicking the C/C++
Go To:
Contents
Search
Related Topics
Bookmarks
Index

(END)