CC3100 SimpleLink™ Wi-Fi® and IoT Solution Getting Started Guide

User's Guide

TEXAS INSTRUMENTS

Literature Number: SWRU375B
June 2014–Revised March 2015

Contents

List of Figures

User's Guide
SWRU375B – June 2014 – Revised March 2015

CC3100 SimpleLink™ Wi-Fi® and IoT Solution Getting Started Guide

ABSTRACT

This guide is intended to assist users in the initial setup and demonstration of the *Getting Started with WLAN Station* application. The guide explains how to install an Integrated Development Environment (IDE), and then compile, download and debug *Getting Started with WLAN Station*.

1 Introduction

1.1 Prerequisites

The user should have the following items:

- One CC3100BOOST
- One CC31XXEMUBOOST or MSP430F5529 Launchpad
- An 802.11b/g/n Wireless Access Point (AP)
- A computer running Microsoft® Windows® 7 or XP operating systems.

2 Getting Started

2.1 Download and Install Software

Download and install the following software:

- CC3100 SDK package http://www.ti.com/tool/cc3100sdk
  - This guide assumes the use of the default installation folder *C:\TI\CC3100SDK_1.1.0*\.

2.2 Update Service Pack

If the board being used is not already flashed with the service pack for SDK 1.1.0, the latest service pack for SDK 1.1.0 needs to be flashed on the CC3100. The latest service pack can be downloaded from http://www.ti.com/tool/cc3100sdk. Refer to the UNIFLASH Quick start guide for details on flashing the service pack to the CC3100 (http://processors.wiki.ti.com/index.php/CC31xx_%26_CC32xx_UniFlash#Service_Pack_Programming). Details on setting up the hardware for flashing can be found in Section 3.1.

SimpleLink is a trademark of Texas Instruments.
Wi-Fi is a registered trademark of Wi-Fi Alliance.

3 Getting Started with SimpleLink Studio

3.1 Configure Boards

1. The jumpers on the CC3100BOOST should be connected as shown in Figure 1.

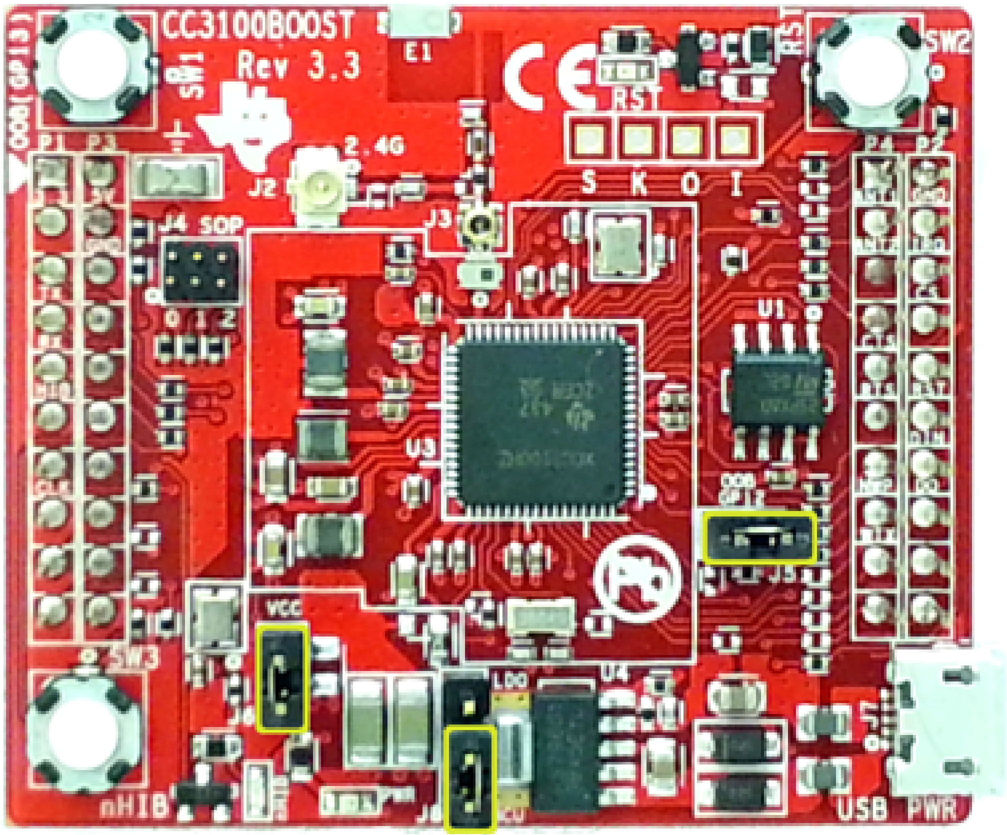


Figure 1. CC3100BOOST

2. The jumpers on the CC31XXEMUBOOST should be connected as shown in Figure 2.

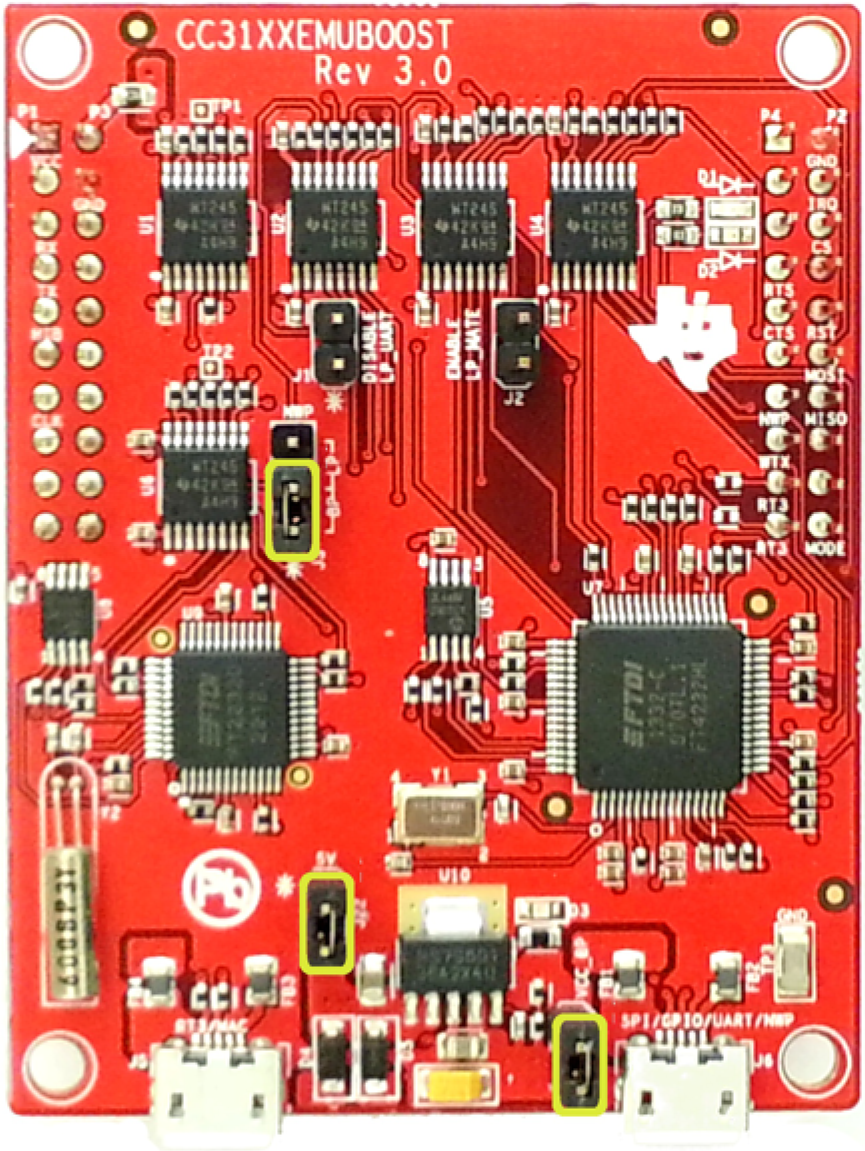


Figure 2. CC31XXEMUBOOST

3. Connect the CC3100BOOST to the CC31XXEMUBOOST as shown in Figure 3.

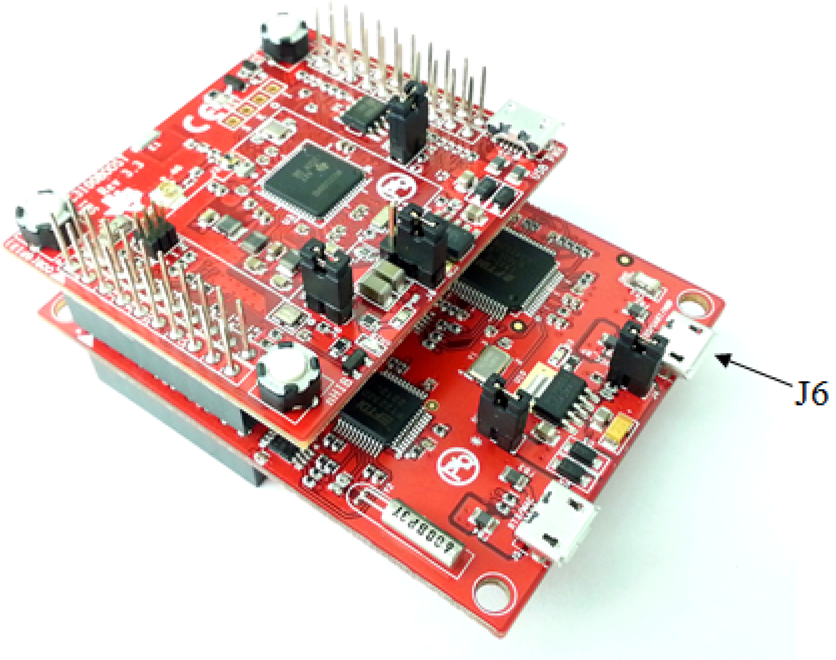
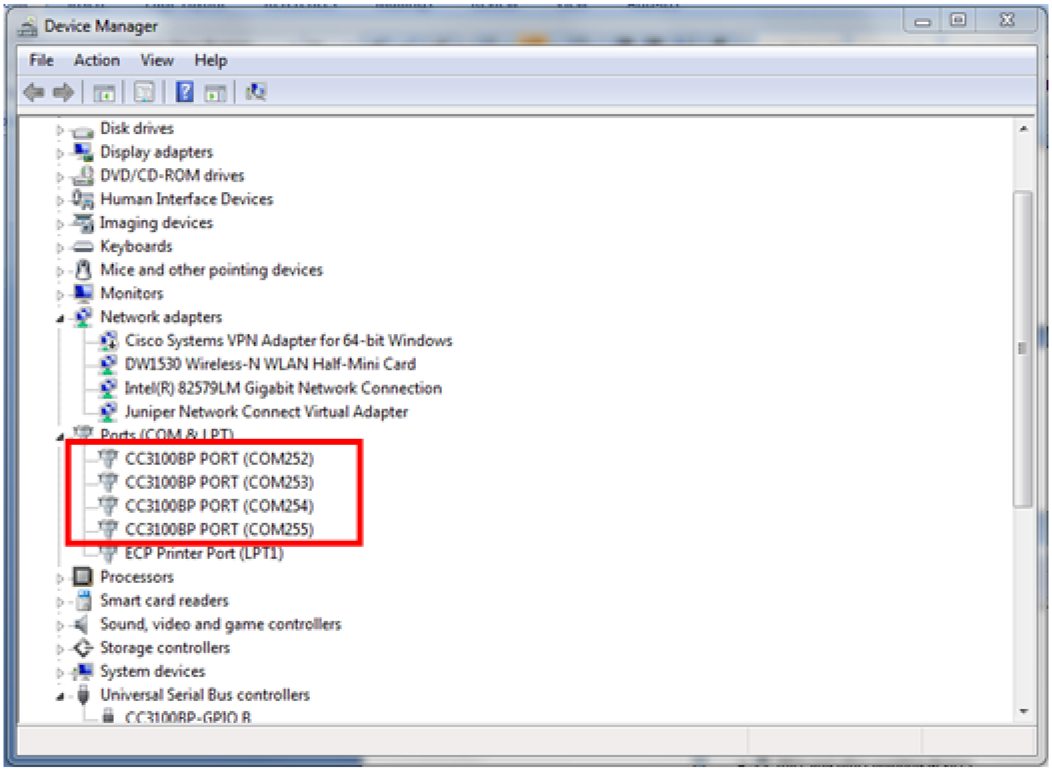


Figure 3. Connect the CC3100BOOST to the CC31XXEMUBOOST

4. Connect the J6 port CC31XXEMUBOOST to the PC using the provided micro-USB cable.
5. The CC3100BOOST will now be visible in the Device Manager as shown in Figure 4. The user may see two COM ports instead of four.



- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Human Interface Devices
- Imaging devices
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
  - Cisco Systems VPN Adapter for 64-bit Windows
  - DW1530 Wireless-N WLAN Half-Mini Card
  - Intel(R) 82579LM Gigabit Network Connection
  - Juniper Network Connect Virtual Adapter
- Ports (COM & LPT)
  - CC3100BP PORT (COM252)
  - CC3100BP PORT (COM253)
  - CC3100BP PORT (COM254)
  - CC3100BP PORT (COM255)
  - ECP Printer Port (LPT1)
- Processors
- Smart card readers
- Sound, video and game controllers
- Storage controllers
- System devices
- Universal Serial Bus controllers
  - CC3100BP-GPIO B

Figure 4. Device Manager

3.2 *Run the Software*

The *Getting Started with WLAN Station* example uses SimpleLink Studio. This example performs the following functions:

1. Program restores WLAN configuration to factory default
2. Prompts the user for the SSID of an AP to connect to.
3. Prompts the user for the security type.
4. Prompts the user for the password to the AP.
5. Attempts to acquire an IP address through DHCP.
6. Attempts to reach the internet.

Option 1. Microsoft Visual Studio:

1. Download and install Microsoft Visual Studio Express 2010 from http://www.visualstudio.com/en-us/downloads/download-visual-studio-vs#DownloadFamilies_4.
2. Open Microsoft Visual Studio Express, and select *File>Open>Project/Solution*.
3. Navigate to *C:\TI\CC3100SDK_1.1.0\cc3100-sdk\platform\simplelinkstudio\example_project_vs\getting_started_with_station*, and open *getting_started.sln*.
4. Select *Build>Build Solution* from the menu.
5. When building is complete, select *Debug>Start Debugging* from the menu.

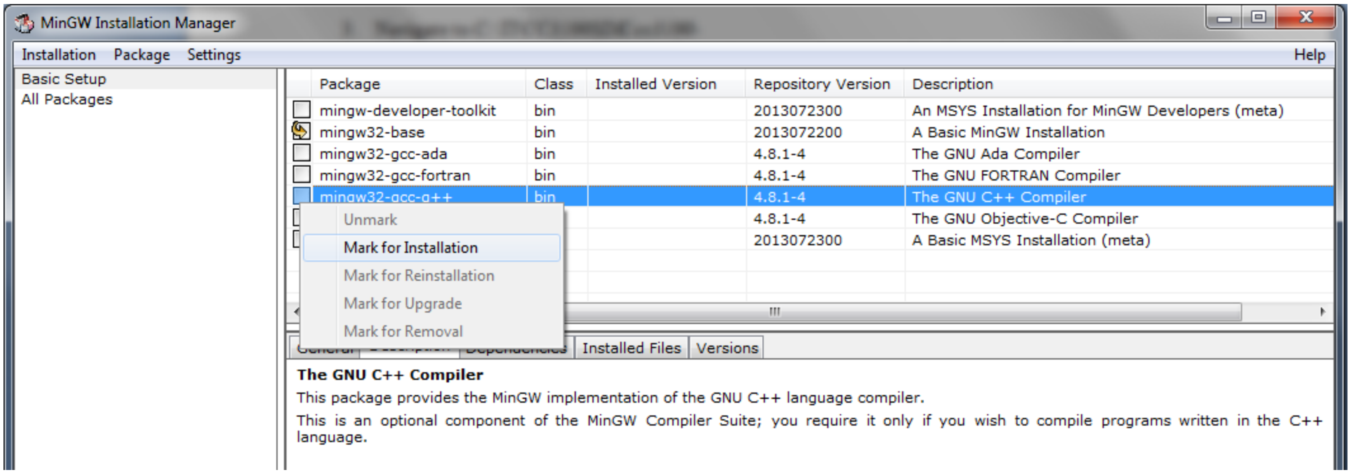
Option 2. Eclipse:

1. Download and install the latest version of Java: https://www.java.com/en/download/. Install the correct version for the system (64-bit or 32-bit).

2. Download and Extract Eclipse from http://www.eclipse.org/downloads/. Choose the Eclipse IDE for C/C++ Developers package. Install the correct version for the system (64-bit or 32-bit).
3. Download and install MinGW from http://sourceforge.net/projects/mingw/files/latest/download?source=files. During installation, make sure you have the following configurations selected:

   (a) Set the installation location as *C:\MinGW*.

   (b) In the MinGW Installation Screen, select packages for **mingw32-base** and **mingw32-gcc-g++** as shown in Figure 5.



Package	Class	Installed Version	Repository Version	Description
mingw-developer-toolkit	bin		2013072300	An MSYS Installation for MinGW Developers (meta)
mingw32-base	bin		2013072200	A Basic MinGW Installation
mingw32-gcc-ada	bin		4.8.1-4	The GNU Ada Compiler
mingw32-gcc-fortran	bin		4.8.1-4	The GNU FORTRAN Compiler
mingw32-gcc-g++	bin		4.8.1-4	The GNU C++ Compiler
			4.8.1-4	The GNU Objective-C Compiler
			2013072300	A Basic MSYS Installation (meta)

The GNU C++ Compiler

This package provides the MinGW implementation of the GNU C++ language compiler.

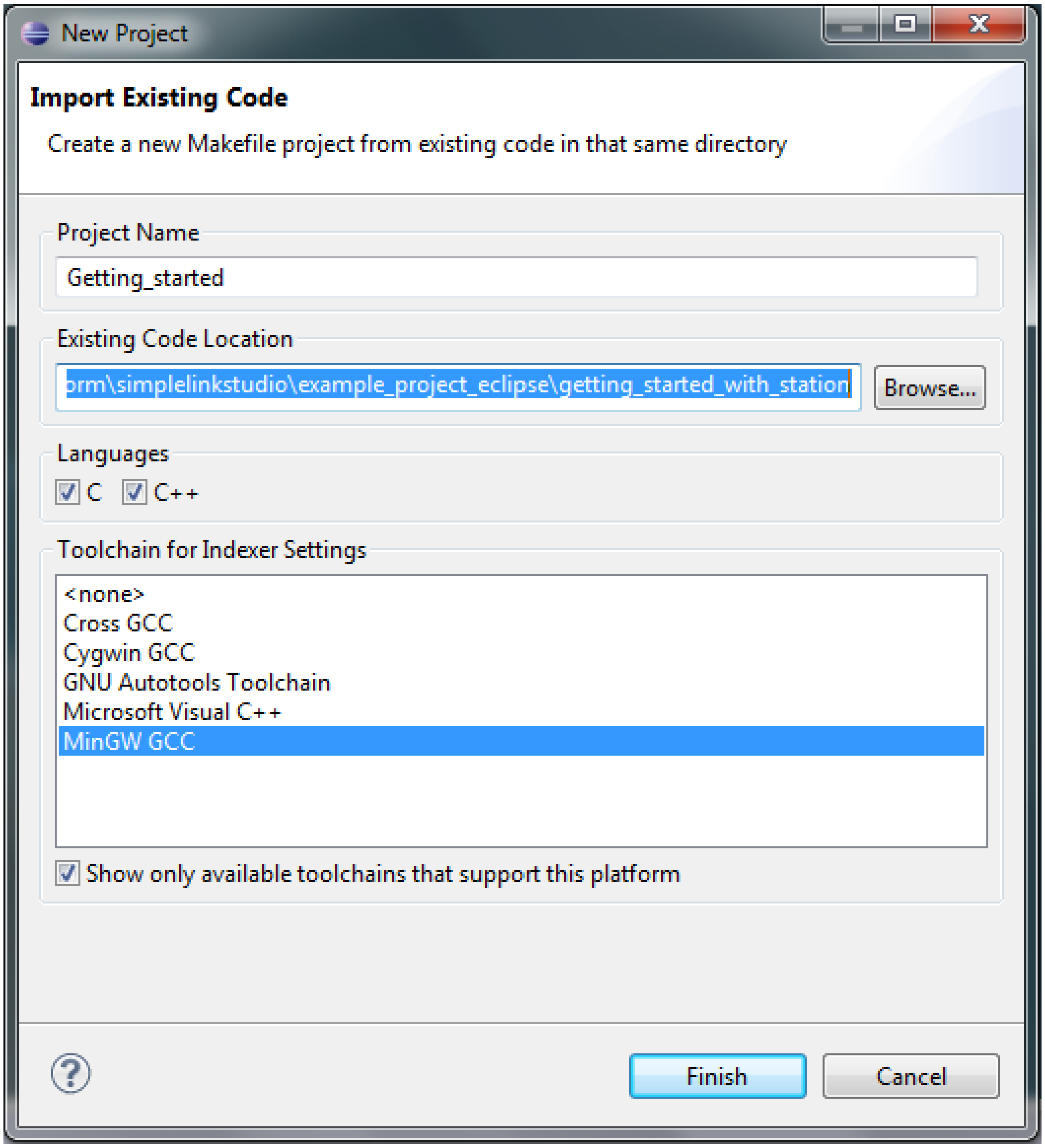
This is an optional component of the MinGW Compiler Suite; you require it only if you wish to compile programs written in the C++ language.

Figure 5. MinGW Installation Screen

   (c) After selecting the packages, choose *Installation>Apply Changes* from the menu, then press Apply.

   (d) Select *Installation>Quit* from the menu to quit the installer.

4. After a successful MinGW installation, add its path (*c:\MinGW\bin\*) to the Windows environment variable path by going into *Control Panel>System>Advanced System Settings>Environment Variables*. Under *System Variables*, select PATH and press Edit. Append “;C:\MinGW\bin\” to the end of the line and press Ok.
5. Open Eclipse by running *eclipse.exe* from the extracted Eclipse folder.
6. In the "Select a workspace," choose your desired workspace directory (for example: *C:\Users\myself\Desktop\eclipse_workspace*). This directory should be different from the user's other versions of Eclipse.
7. From the menu select *File>New>Makefile Project with Existing Code*.
8. Enter ‘Getting_started’ as the project name.
9. For Existing Code Location, enter *C:\TI\CC3100SDK_1.1.0\cc3100-sdk\platform\simplelinkstudio\example_project_eclipse\getting_started_with_station*.
10. For Toolchain for Indexer Settings, choose “MinGW GCC” as shown in Figure 6.


New Project

Import Existing Code

Create a new Makefile project from existing code in that same directory

Project Name

Getting_started

Existing Code Location

prm\simplelinkstudio\example_project_eclipse\getting_started_with_station

Languages

☑ C ☑ C++

Toolchain for Indexer Settings

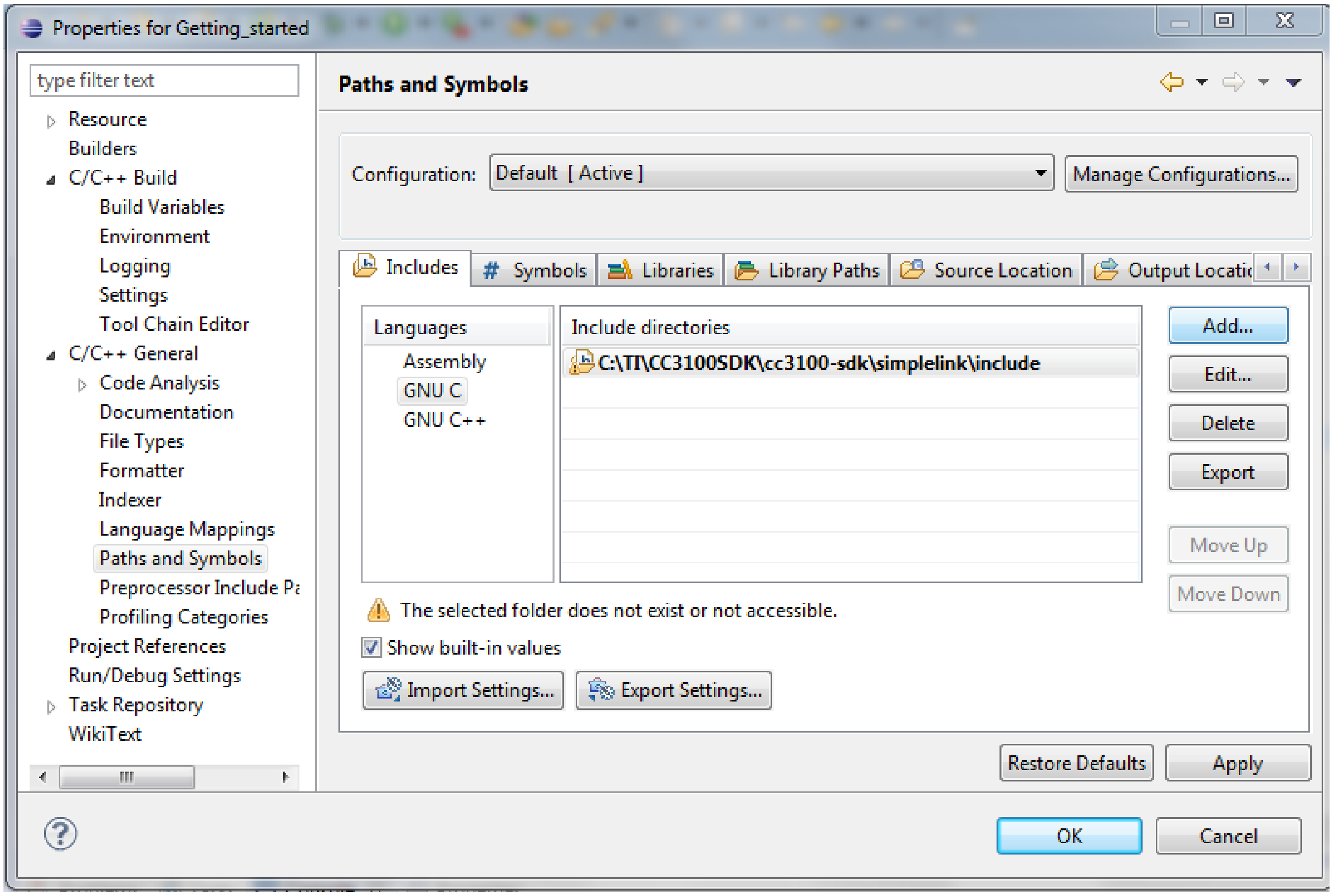
- <none>
- Cross GCC
- Cygwin GCC
- GNU Autotools Toolchain
- Microsoft Visual C++
- MinGW GCC

☑ Show only available toolchains that support this platform

Figure 6. Import Existing Code

11. Press Finish.
12. Select *Window>Show View>Project Explorer* from the menu.
13. Select the Getting_started project in the Project Explorer, and select *File>Properties* from the menu.
14. In the Project Explorer window, right click on <*Project_Folder_Name*>, then select Property.
15. Click the **C/C++ Build** menu and do the following:
    (a) Uncheck **Use the default build command**.
    (b) Type *mingw32-make -f Makefile* in **Build command**.
    (c) Uncheck **Generate Makefiles automatically**.
    (d) Set the Build Directory as: *${workspace_loc:/Getting_started}*
16. Expand the **C/C++ Build** menu, and select **Tool Chain Editor**. Under **Current Builder**, Select "Gnu Make Builder", then click Apply.
17. Select **Environment** in the **C/C++ Build** menu. Make sure the value of MSYS_HOME is empty, then click Apply.

18. Click the **C/C++ General** menu, and select **Paths and Symbols**. Under the **Includes** tab, in the Languages column, select **GNU C**. Press the Add button to add the directory: *C:\TI\CC3100SDK_1.1.0\cc3100-sdk\simplelink\include*.



Paths and Symbols

Configuration: Default [Active]

Languages	Include directories
Assembly	C:\TI\CC3100SDK\cc3100-sdk\simplelink\include
GNU C	
GNU C++	

The selected folder does not exist or not accessible.

☑ Show built-in values

Figure 7. Paths and Symbols

19. Press OK.
20. To fix a known bug in Eclipse console output, add the following line of code to the beginning of the main function:

```
setvbuf(stdout, NULL, _IONBF, 0);
```

21. Save the file, and select *Project>Clean* from the menu.
22. Select the Getting_started project and press OK.
23. Select the Getting_started project from Project Explorer, and from the menu select *Project>Build Project*.
24. Press Ctrl+F11 to start the program.

4 Getting Started with the MSP430F5529

4.1 Configure Boards

1. The jumpers on the CC3100BOOST should be connected as shown in Figure 8.

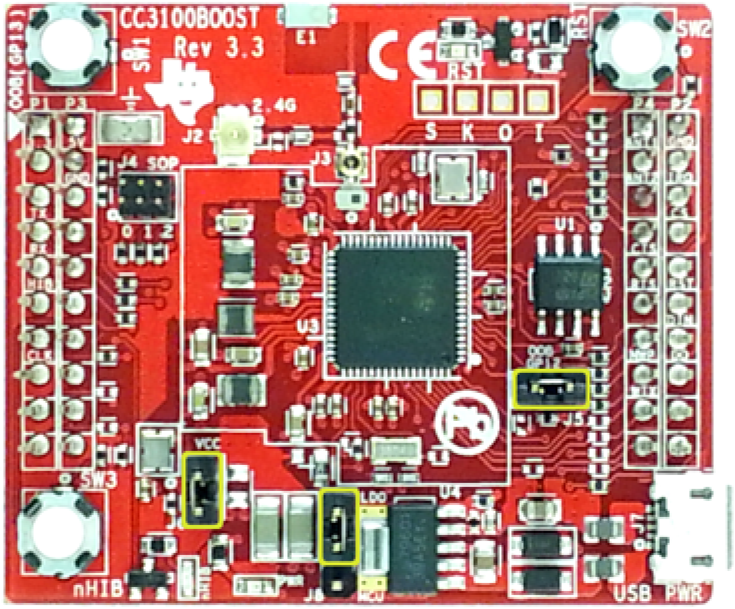


Figure 8. CC3100BOOST

2. The jumpers on the MSP430F5529 Launchpad should be connected as shown in Figure 9.

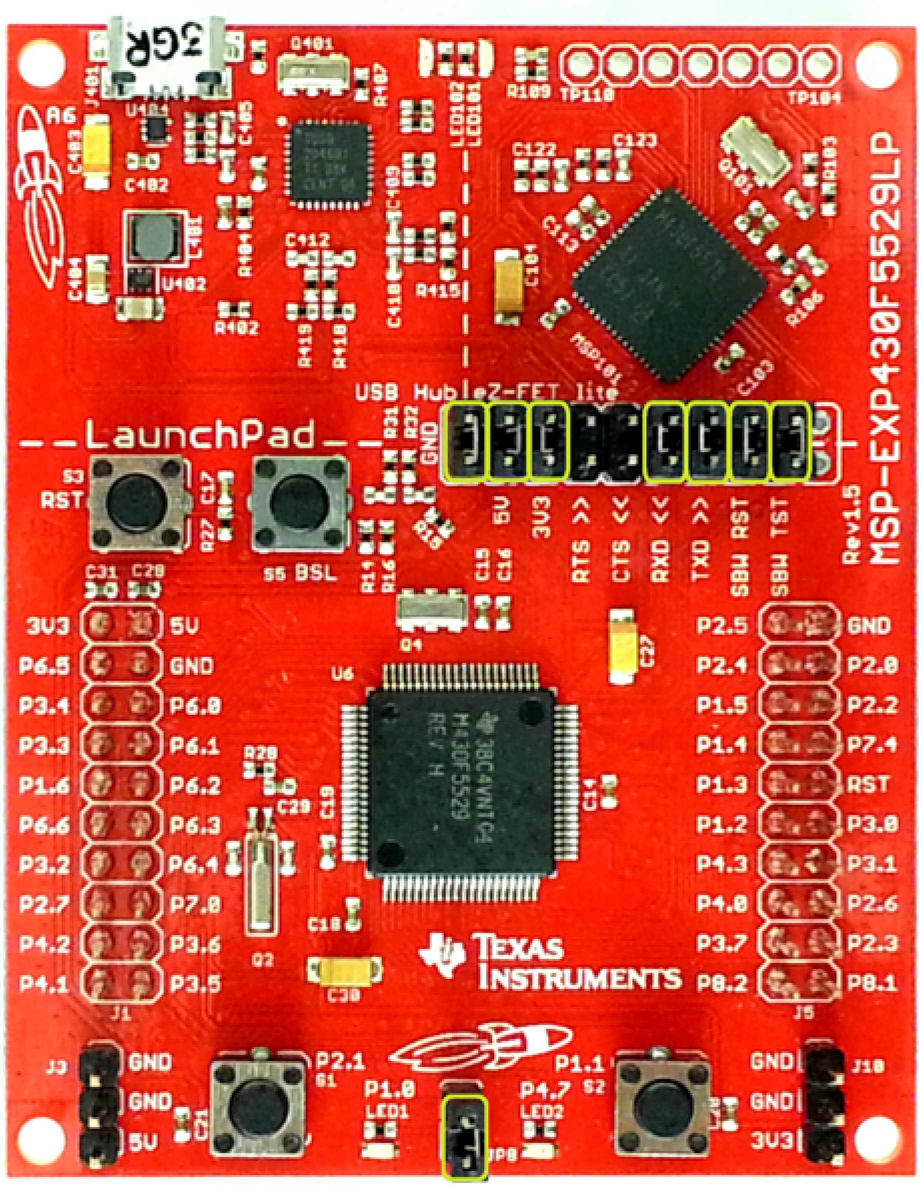


Figure 9. Jumpers on the MSP430F5529 Launchpad

3. Connect the CC3100BOOST to the MSP430F5529 Launchpad as shown in Figure 10.

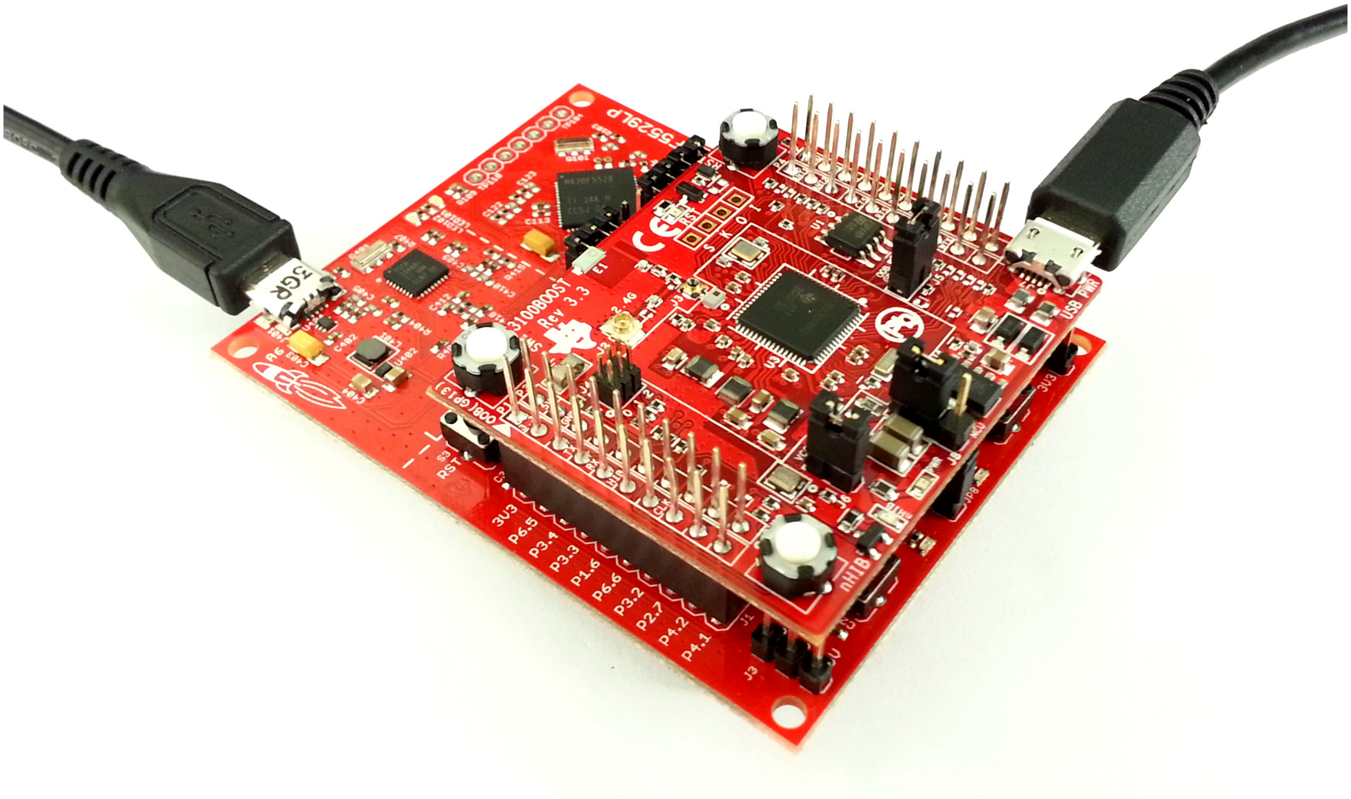


Figure 10. Connect the CC3100BOOST to the MSP430F5529 Launchpad

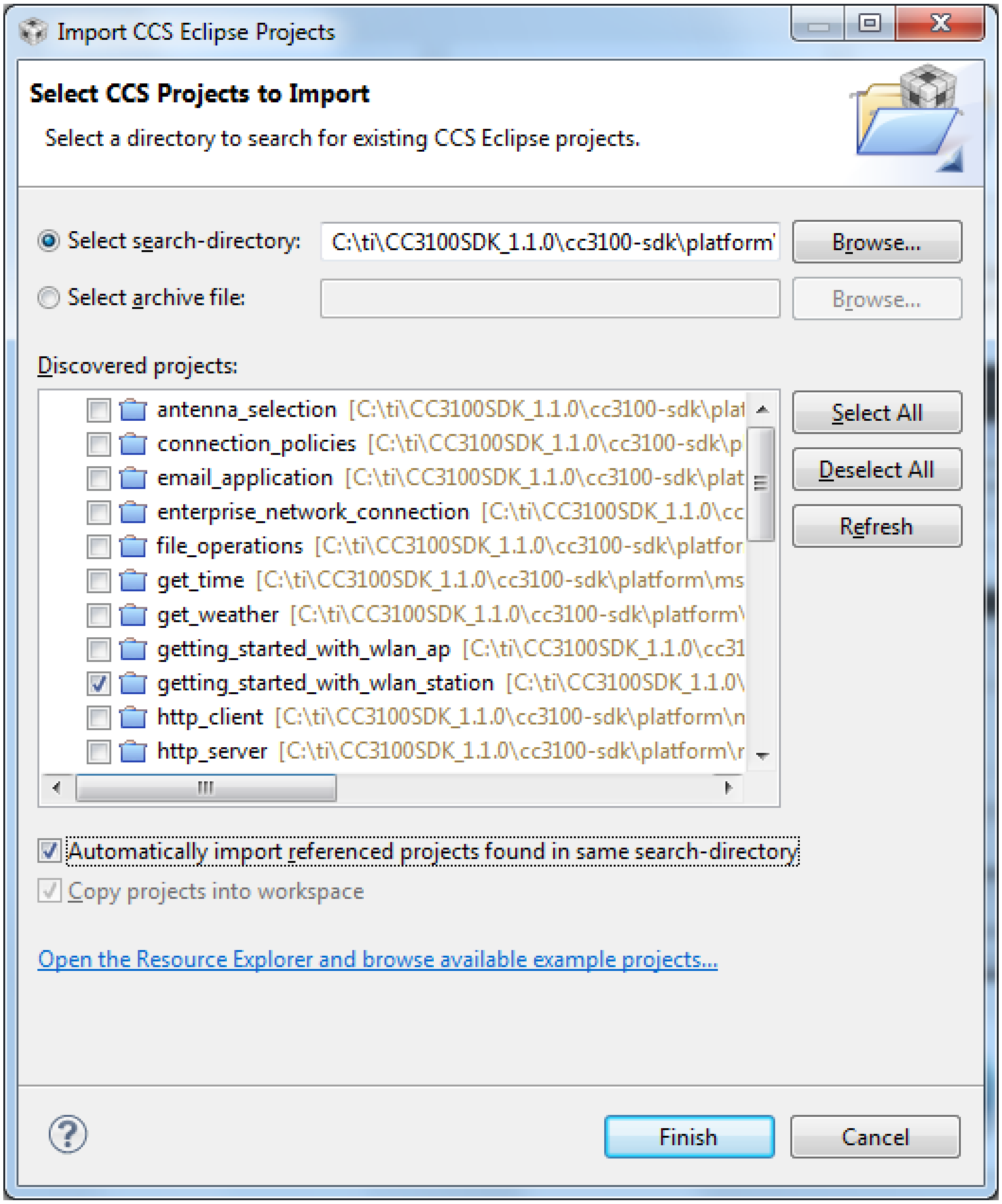
4. Connect a Micro-USB cable from J7 on the Boosterpack to the Windows PC, and connect a Micro-USB cable from J401 on the MSP430F5529 Launchpad to the Windows PC.

4.2 Run the Software

The example shown provides output via UART. To see the output from the program, download Tera Term (or similar software) from this link: http://en.sourceforge.jp/projects/ttssh2/releases/

Option 1. Code Composer Studio (CCS):

1. Download and run the Code Composer Studio 6.0.1 (CCS) installation wizard (*ccs_setup_win32.exe*) from the TI website or from the CCS Wiki page. Must be **Version 6.0.1.00040** or later. When prompted to select processor support, select the '*MSP Ultra Low Power MCUs*' processor support option. The remaining options for the installer should be left as the default. Installation may take up to an hour.
2. Open CCS, and choose *File>Import* from the menu. Under C/C++, choose **CCS Projects**.
3. Under *Select Search Directory*, enter the path: *C:\TI\CC3100SDK_1.1.0\cc3100-sdk\platform\msp430f5529lp\example_project_ccs*.
4. Check the project *getting_started_with_wlan_station* and press Finish.



Import CCS Eclipse Projects

Select CCS Projects to Import

Select a directory to search for existing CCS Eclipse projects.

● Select search-directory: C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platform' [Browse...]

○ Select archive file: [Browse...]

Discovered projects:

- □ antenna_selection [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\plat
- □ connection_policies [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\p
- □ email_application [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\plat
- □ enterprise_network_connection [C:\ti\CC3100SDK_1.1.0\cc
- □ file_operations [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platfor
- □ get_time [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platform\ms
- □ get_weather [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platform\
- □ getting_started_with_wlan_ap [C:\ti\CC3100SDK_1.1.0\cc31
- ☑ getting_started_with_wlan_station [C:\ti\CC3100SDK_1.1.0\
- □ http_client [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platform\n
- □ http_server [C:\ti\CC3100SDK_1.1.0\cc3100-sdk\platform\r

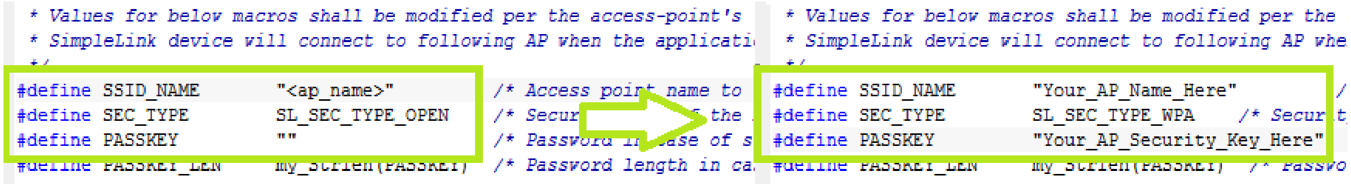
☑ Automatically import referenced projects found in same search-directory

☑ Copy projects into workspace

Open the Resource Explorer and browse available example projects...

Figure 11. Select CCS Projects to Import

5. Open the *sl_common.h* file located at the path C:\TI\CC3100SDK_1.1.0\cc3100-sdk\examples\common\.
6. Edit *sl_common.h* to use the SSID, security type and security key of the Access Point being used. Edit the macros SSID_NAME, SEC_TYPE and PASSKEY to contain the Access Point's information as shown in Figure 12. The security types supported for this demo are WPA/WPA2, WEP (5/13 ASCII), and Open. WEP supports key index 1 only. For Open security, define SEC_TYPE as SL_SEC_TYPE_OPEN. For WPA and WPA2 security, define it as SL_SEC_TYPE_WPA. For WEP security, define it as SL_SEC_TYPE_WEP.

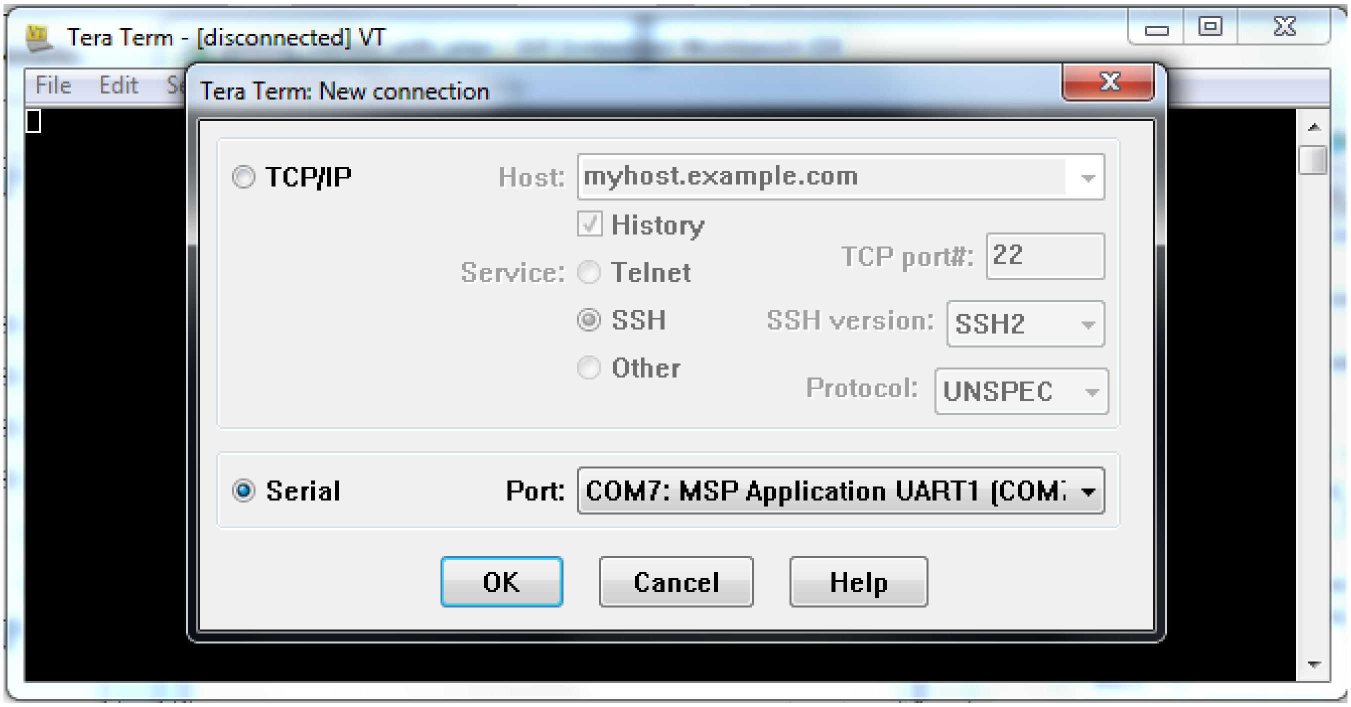


```
 * Values for below macros shall be modified per the access-point's
 * SimpleLink device will connect to following AP when the applicati
 */
#define SSID_NAME        "<ap_name>"          /* Access point name to
#define SEC_TYPE         SL_SEC_TYPE_OPEN     /* Secur              the
#define PASSKEY          ""                   /* Password in case of s
#define PASSKEY_LEN      my_strlen(PASSKEY)   /* Password length in ca
```

```
 * Values for below macros shall be modified per the
 * SimpleLink device will connect to following AP whe
 */
#define SSID_NAME        "Your_AP_Name_Here"          /
#define SEC_TYPE         SL_SEC_TYPE_WPA     /* Secur t
#define PASSKEY          "Your_AP_Security_Key_Here"
#define PASSKEY_LEN      my_strlen(PASSKEY)   /* Passwo
```

Figure 12. Define SSID_Name

7. Select the *getting_started_with_wlan_station* project in Project Explorer and select *Project>Build Project* from the menu.
8. Launch Tera Term, and create a new serial connection to the MSP430F5529 Launchpad COM port as shown in Figure 13. The baud rate should remain at 9600.
9. Press F11on the CCS window to start debugging.

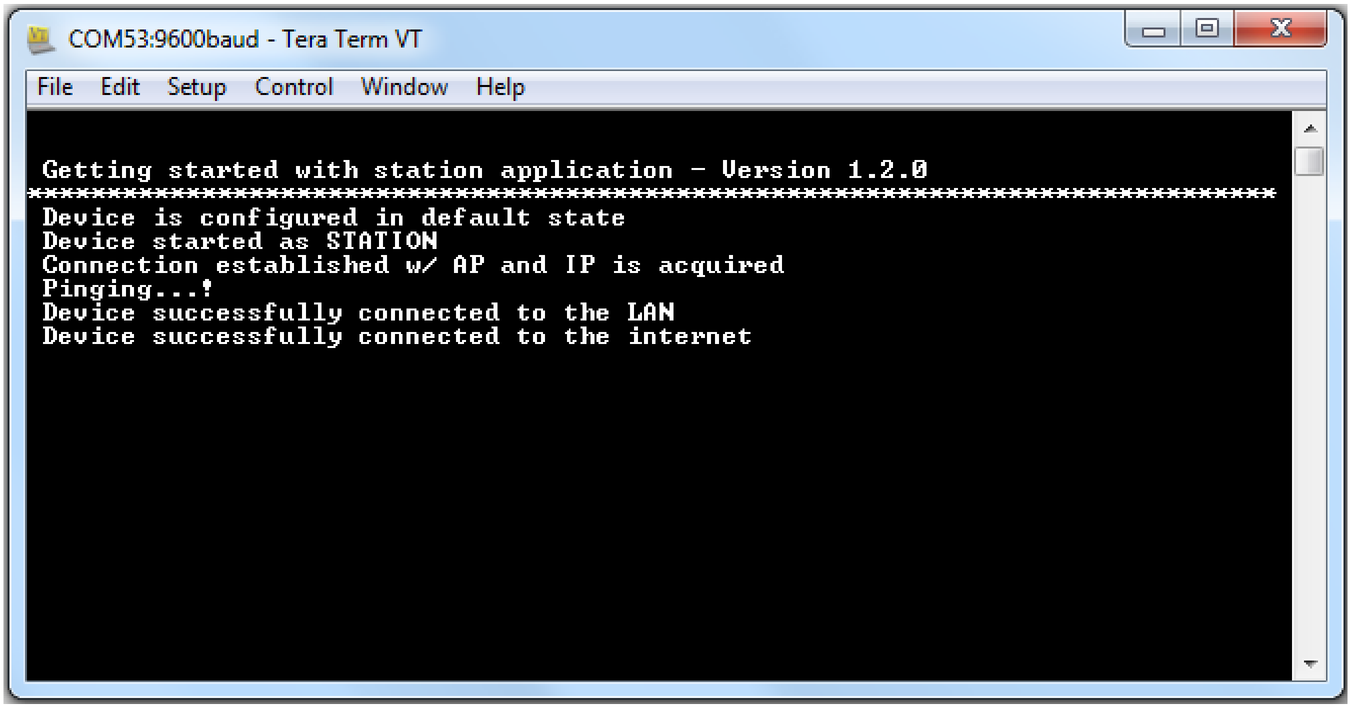


○ TCP/IP Host: myhost.example.com
☑ History
Service: ○ Telnet TCP port#: 22
● SSH SSH version: SSH2
○ Other Protocol: UNSPEC

● Serial Port: COM7: MSP Application UART1 (COM

Figure 13. Launch Tera Term

10. If the CC3100 successfully completes all steps, the serial output appears as shown in Figure 14.

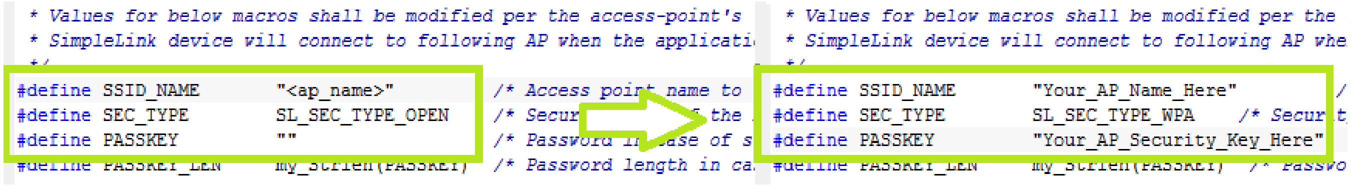

COM53:9600baud - Tera Term VT
File Edit Setup Control Window Help

```
Getting started with station application - Version 1.2.0
*******************************************************************************
Device is configured in default state
Device started as STATION
Connection established w/ AP and IP is acquired
Pinging...!
Device successfully connected to the LAN
Device successfully connected to the internet
```

Figure 14. Tera Term VT

Option 2. IAR Workbench

1. Install IAR Workbench for MSP430 version 6.20 or later: http://www.iar.com/en/Products/IAR-Embedded-Workbench/TI-MSP430/.
2. Open IAR Workbench and select *File>Open>Workspace* from the menu.
3. Select the project: *C:\TI\CC3100SDK_1.1.0\cc3100-sdk\platform\msp430f5529lp\example_project_iar\getting_started_with_wlan_station\getting_started_with_wlan.eww*.
4. Open the *sl_common.h* file located at the path C:\TI\CC3100SDK_1.1.0\cc3100-sdk\examples\common\.
5. Edit *sl_common.h* to use the SSID, security type and security key of the Access Point being used. Edit the macros SSID_NAME, SEC_TYPE and PASSKEY to contain the Access Point's information as shown in Figure Figure 15. The security types supported for this demo are WPA/WPA2 and Open. For Open security, define SEC_TYPE as SL_SEC_TYPE_OPEN. For WPA and WPA2 security, define it as SL_SEC_TYPE_WPA.

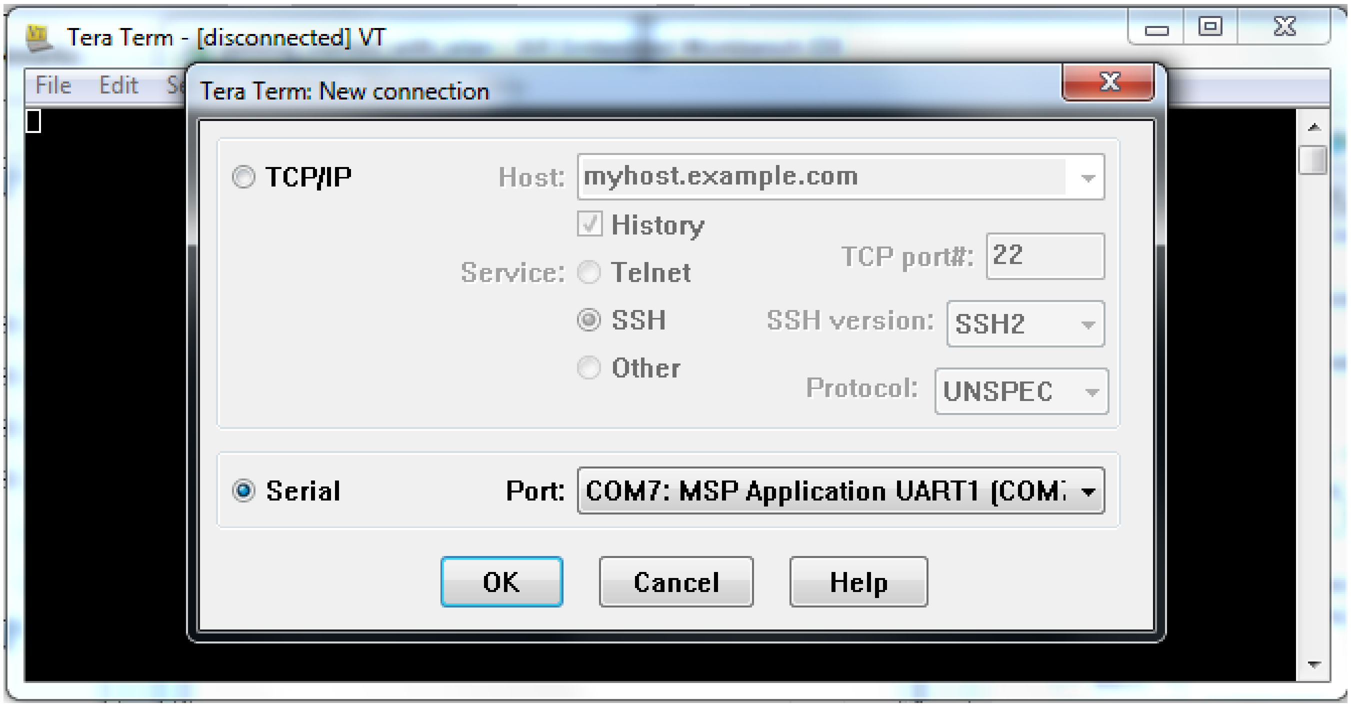


```
 * Values for below macros shall be modified per the access-point's
 * SimpleLink device will connect to following AP when the applicati
 */
#define SSID_NAME        "<ap_name>"         /* Access point name to
#define SEC_TYPE         SL_SEC_TYPE_OPEN    /* Secur          the
#define PASSKEY          ""                  /* Password in case of s
#define PASSKEY_LEN      my_strlen(PASSKEY)  /* Password length in ca
```

```
 * Values for below macros shall be modified per the
 * SimpleLink device will connect to following AP whe
 */
#define SSID_NAME        "Your_AP_Name_Here"             /
#define SEC_TYPE         SL_SEC_TYPE_WPA     /* Secur  t
#define PASSKEY          "Your_AP_Security_Key_Here"
#define PASSKEY_LEN      my_strlen(PASSKEY)    /* Passwo
```

Figure 15. Macro Definition of SSID_Name

6. Select *Project>Rebuild All* from the menu.
7. Launch Tera Term, and create a new serial connection to the MSP430F5529 Launchpad COM port as shown in Figure 16. The baud rate should remain at 9600.
8. After building is finished, select *Project>Download and Debug* from the menu to start debugging.


Tera Term - [disconnected] VT

Tera Term: New connection

○ TCP/IP Host: myhost.example.com

☑ History

Service: ○ Telnet TCP port#: 22

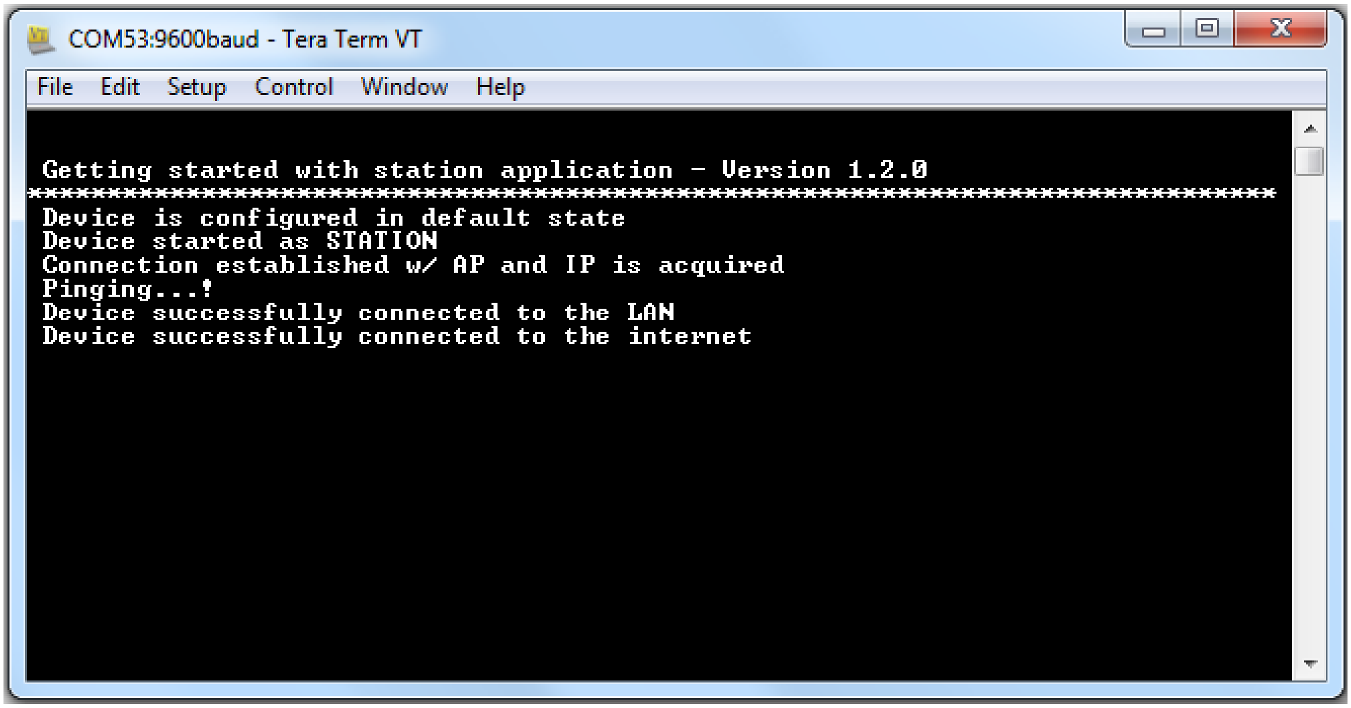
● SSH SSH version: SSH2

○ Other Protocol: UNSPEC

● Serial Port: COM7: MSP Application UART1 (COM7

Figure 16. Launch Tera Term

9. If the CC3100 successfully completes all steps, the serial output appears as shown in Figure 17.


COM53:9600baud - Tera Term VT

```
 Getting started with station application - Version 1.2.0
********************************************************************************
 Device is configured in default state
 Device started as STATION
 Connection established w/ AP and IP is acquired
 Pinging...!
 Device successfully connected to the LAN
 Device successfully connected to the internet
```

Figure 17. Tera Term VT

5 Summary

After the development environment has been set up, see the following resources for further assistance in development:

- CC3100 Programmer's Guide – This guide contains information on how to use the SimpleLink API for writing WLAN-enabled applications.
- Uniflash – The Uniflash tool is used for manually storing files on the external serial flash. This includes the SimpleLink firmware patch file and any configuration files, security certificates, web pages, and so forth.
- CC3100 Wiki – All information and tools for the CC3100, including the above, can be found on the CC3100 Wiki page.

6 Acronyms Used

STA – Wi-Fi Station

AP – Wi-Fi Access Point

WLAN – Wireless LAN

CCS – Code Composer Studio

GCC – GNU Compiler Collection

Revision History

Changes from A Revision (July 2014) to B Revision	Page
• Updated for SDK 1.1.0.	4
• Updated image.	15
• Updated image.	17
• Updated image.	18

Revision History

Changes from Original (June 2014) to A Revision	Page
• Replaced image.	3
• Replaced image.	3
• Changed path to C:\TI\CC3100SDK_1.0.0\	4
• Added Update Service Pack section.	4
• Moved Steps 1 and 8 to Section 3.1, deleted Section 3.2	7
• Updated link.	8
• Changed path.	8
• Changed path.	9
• Changed path.	11
• Added Tera Term download link.	14
• Updated version number.	14
• Changed path.	14
• Updated image.	14
• Updated image.	15
• Added new download link for sl_common.h file.	15
• Replaced main.c with sl_common.h	15
• Replaced image.	16
• Added Steps 8 and 9.	16
• Changed path.	17
• Replaced main.c with sl_common.h.	17
• Changed path.	17
• Replaced image.	17
• Added Steps 7 and 8.	18

IMPORTANT NOTICE

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265