# CC3220 SimpleLink™ Wi-Fi® LaunchPad™ Development Kit Hardware

The CC3220 device is part of the SimpleLink™ microcontroller (MCU) platform which consists of Wi-Fi®, *Bluetooth*® low energy, Sub-1 GHz and host MCUs. All share a common, easy-to-use development environment with a single core software development kit (SDK) and rich tool set. A one-time integration of the SimpleLink platform lets you add any combination of devices from the portfolio into your design. The ultimate goal of the SimpleLink platform is to achieve 100 percent code reuse when your design requirements change. For more information, visit www.ti.com/simplelink.

The CC3220 SimpleLink LaunchPad™ Development Kit (CC3220-LAUNCHXL) is a low-cost evaluation platform for ARM® Cortex®-M4-based MCUs. The LaunchPad design highlights the CC3220 Internet-on-a chip™ solution and Wi-Fi capabilities. The CC3220 LaunchPad also features temperature and accelerometer sensors, programmable user buttons, three LEDs for custom applications, and onboard emulation for debugging. The stackable headers of the CC3220 LaunchPad XL interface demonstrate how easy it is to expand the functionality of the LaunchPad when interfacing with other peripherals on many existing BoosterPack™ Plug-in Module add-on boards, such as graphical displays, audio codecs, antenna selection, environmental sensing, and more.

## Contents



## List of Figures



## List of Tables

# 1 Introduction

## *1.1 CC3220 LaunchPad™*

Created for the Internet of Things (IoT), the SimpleLink Wi-Fi CC3220 device is a single-chip microcontroller (MCU) with built-in Wi-Fi connectivity for the LaunchPad ecosystem, which integrates a high-performance ARM Cortex-M4 MCU and lets customers develop an entire application with one device. With on-chip Wi-Fi, Internet, and robust security protocols, no prior Wi-Fi experience is required for fast development.

The CC3220 LaunchPad, referred to by its part number CC3220-LAUNCHXL, is a low-cost evaluation platform for ARM Cortex-M4-based MCUs. The LaunchPad design highlights the CC3220 Internet-on-a chip solution and Wi-Fi capabilities. The CC3220 LaunchPad also features temperature and accelerometer sensors, programmable user buttons, three LEDs for custom applications, and onboard emulation for debugging. The stackable headers of the CC3220 LaunchPad XL interface demonstrate how easy it is to expand the functionality of the LaunchPad when interfacing with other peripherals on many existing BoosterPack add-on boards, such as graphical displays, audio codecs, antenna selection, environmental sensing, and more. Figure 2 shows the CC3220 LaunchPad. There are two variants of the LaunchPad: the CC3220S-LAUNCHXL and the CC3220SF-LAUNCHXL. This user's guide applies to both variants, and any differences are pointed out in relevant sections.

Multiple development environment tools are also available, including TI's Eclipse-based Code Composer Studio™ (CCS) integrated development environment (IDE) and IAR Embedded Workbench®. More information about the LaunchPad, the supported BoosterPack modules, and the available resources can be found at TI's LaunchPad portal. Also visit the CC3220 Wiki page for design resources and example projects.

**NOTE:** The maximum RF power transmitted in each WLAN 2.4-GHz band is 17.5 dBM (EIRP power).

**NOTE:** The antennas used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons, and must not be colocated or operating in conjunction with any other antenna or transmitter.

**NOTE:** All figures and references in this document apply to RevA and RevB. Most of the document also applies to higher revisions, unless otherwise stated. For the exact list of changes made across board revisions, see the Revision History.

**Waste Electrical and Electronic Equipment (WEEE)**

This symbol means that according to local laws and regulations your product and/or its battery shall be disposed of separately from household waste. When this product reaches its end of life, take it to a collection point designated by local authorities. Proper recycling of your product will protect human health and the environment.

**Figure 1. WEEE Statement**

## 1.2 Key Features

- CC3220, SimpleLink, Wi-Fi, Internet-on-a chip solution with integrated MCU
- 40-pin LaunchPad standard that leverages the BoosterPack ecosystem
- XDS110-based JTAG emulation with serial port for flash programming
- Two buttons and three LEDs for user interaction
- Back-channel universal asynchronous receiver/transmitter (UART) through USB to PC
- Onboard chip antenna with U.FL for conducted testing
- Onboard accelerometer and temperature sensor for out-of-box demo
- Micro USB connector for power and debug connections

## 1.3 Kit Contents

- CC3220 LaunchPad development tool (CC3220SF-LAUNCHXL)
- Micro USB cable
- Quick start guide

## 1.4 Regulatory Compliance

The SimpleLink CC3220 Wi-Fi LaunchPad is tested for and found to be in compliance with FCC and ISED regulations regarding unlicensed intentional radiators.

Hereby, Texas Instruments Inc. declares that the radio equipment type CC3220S-LAUNCHXL and CC3220SF-LAUNCHXL are in compliance with Directive 2014/53/EU. The full text of the EU declaration of conformity is available at the following internet addresses:

- CC3220S-LAUNCHXL
- CC3220SF-LAUNCHXL

## 2 Hardware Description

Figure 2 shows the CC3220 LaunchPad EVM.

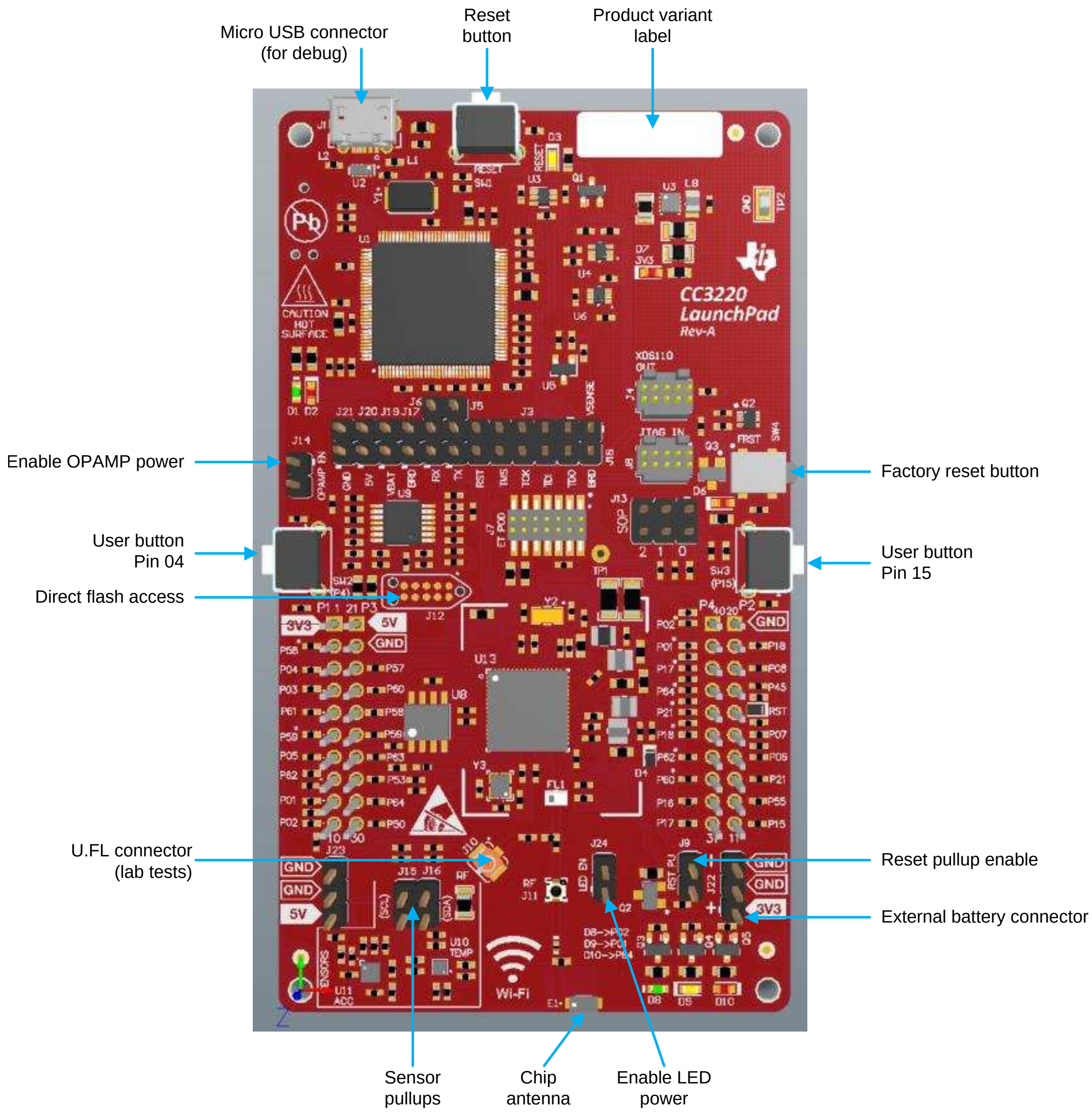


**Figure 2. CC3220 LaunchPad™ EVM Overview**

## 2.1 Block Diagram

Figure 3 shows the CC3220 block diagram.

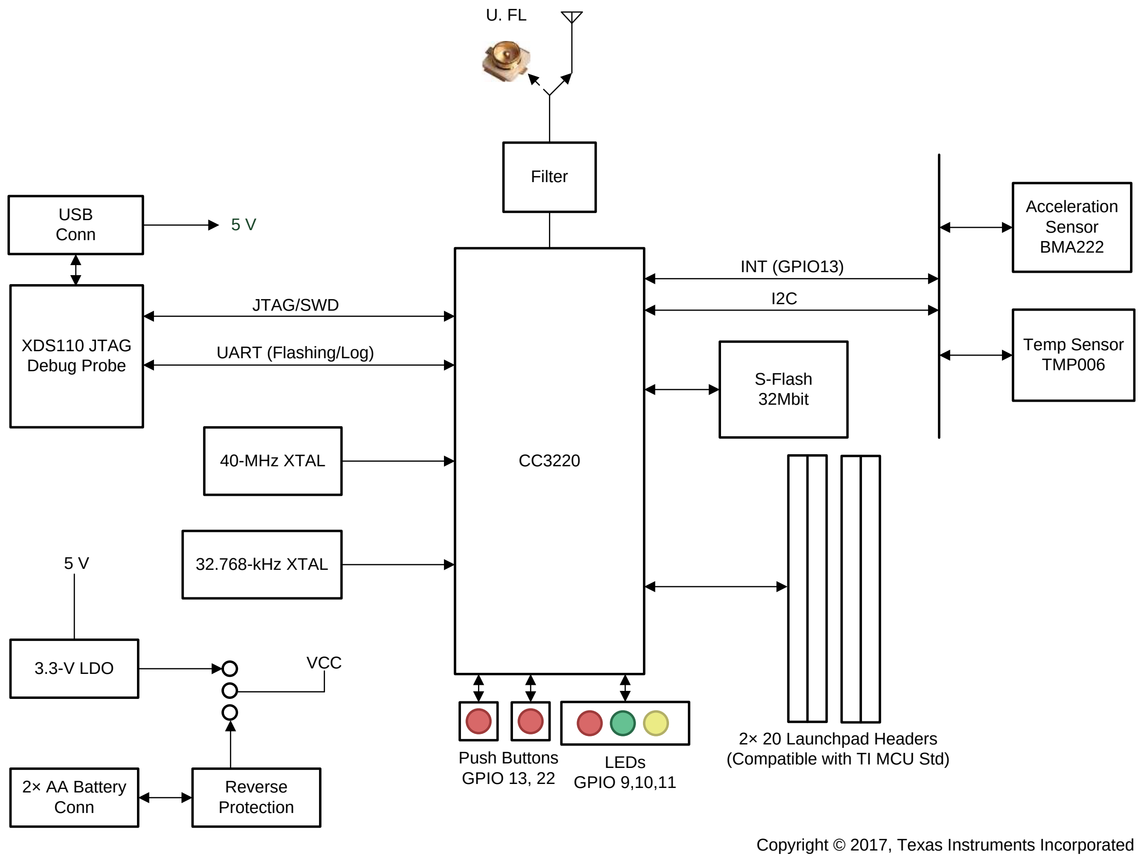


**Figure 3. CC3220 Block Diagram**

## 2.2 Hardware Features

- CC3220, SimpleLink, Wi-Fi, Internet-on-a chip solution with integrated MCU
- 40-pin LaunchPad standard that leverages the BoosterPack ecosystem
- TI Standard XDS110-based JTAG emulation with serial port for flash programming
- Supports both 4-wire JTAG and 2-wire SWD
- Two buttons and three LEDs for user interaction
- Virtual COM port UART through USB on PC
- Onboard chip antenna with U.FL for conducted testing selectable using 0-Ω resistors
- Onboard accelerometer and temperature sensor for out-of-box demo, with option to isolate them from the inter-integrated circuit ($I^2C$) bus
- Micro USB connector for power and debug connections
- Headers for current measurement and external JTAG connection with an option to use the onboard XDS110 to debug customer platforms
- Bus-powered device, with no external power required for Wi-Fi
- Long-range transmission with a highly optimized antenna (200-meter typical in open air with a 6-dBi antenna AP)
- Can be powered externally, working down to 2.3 V (typical)

### 2.3 Connecting a BoosterPack™

A compatible BoosterPack can be stacked on top of the LaunchPad using the 2-pin × 20-pin connectors. The connectors do not have a key to prevent the misalignment of the pins or reverse connection.

Ensure that the VCC and 5-V pins are aligned with the BoosterPack module header pins. On the CC3220 LaunchPad, a small white 3V3 tag symbol is provided near pin 1 (see Figure 4) to orient all BoosterPack modules. This same marking, provided on compatible BoosterPack modules, must be aligned before powering up the boards.

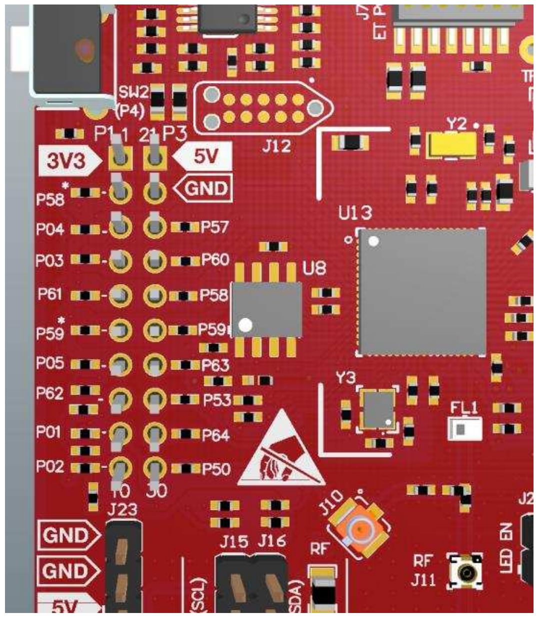


**Figure 4. Pin 1 Marking on CC3220LP (3V3 Tag)**

### 2.4 Wired Connections, Jumper Settings, Buttons, and LEDs

#### 2.4.1 JTAG Headers

The headers are provided on the board to isolate the CC3220 device from the onboard XDS110-based JTAG emulator. These jumpers are shorted by default when the board is shipped from TI. Figure 5 and Table 1 are for default configurations, and Figure 6 shows the external emulator connection.

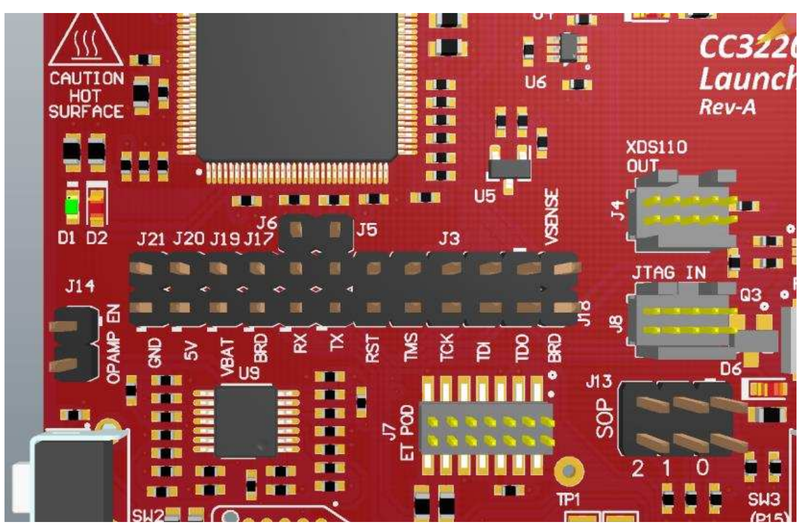


**Figure 5. Default Jumper Configuration for JTAG Lines**

**Table 1. JTAG Header Pin Definitions**

| Reference (Rev. A) | Reference (Rev. B)[1] | Use | Comments |
|---|---|---|---|
| J3 (TCK)[2] | J8 (TCK) | JTAG / SWD | Jumpers populated: onboard emulator connected<br>Jumpers not populated: onboard emulator disconnected |
| J3 (TMS)[2] | J8 (TMS) | JTAG / SWD | |
| J3 (TDI) | J8 (TDI) | JTAG | |
| J3(TDO) | J8 (TDO) | JTAG | |

(1) The only difference between Rev. A and Rev. B are the reference designators on the board.
(2) For SWD mode, only TCK and TMS must be shorted to the CC3220.

To connect an external emulator, remove these jumpers and place the external emulator on the JTAG IN connector.

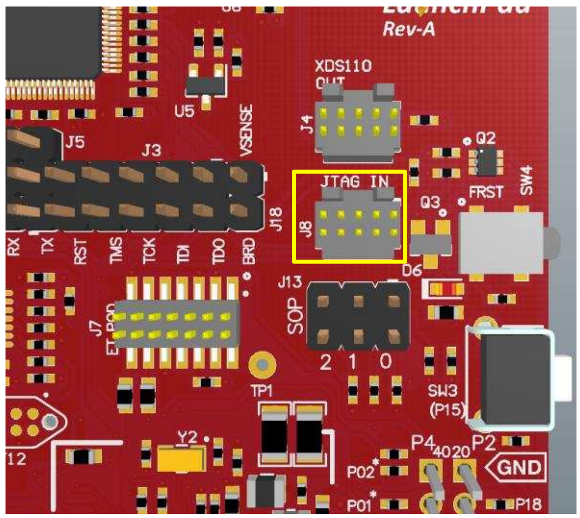


**Figure 6. JTAG IN Connector (J8)**

### 2.4.2 I$^2$C Connections

The board features an accelerometer and a temperature sensor for the out-of-box demo. These are connected to the I$^2$C bus, and can be isolated using the jumpers provided (shown as yellow jumpers J15 and J16 in Figure 7).

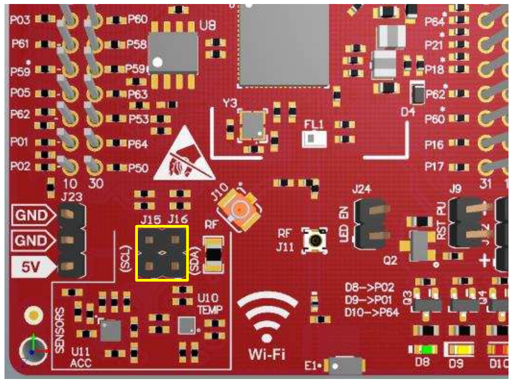


**Figure 7. I$^2$C Bus Connections**

By removing J15 and J16, the accelerometer and the temperature sensors are isolated from the I$^2$C bus. This measure also removes the I$^2$C pullup resistors from the sensor side of the circuit, and therefore any connection to the circuit requires the user to install external pullup resistors.

Table 2 lists the I$^2$C jumper definitions.

**Table 2. I$^2$C Jumper Definitions**

| Reference (Rev. A) | Reference (Rev. B) | Use | Comments |
|---|---|---|---|
| J16 | J2 | I2C SDA | Populated: CC3220 SDA connected to onboard sensors with pullup |
| | | | Open: CC3220 SDA disconnected from onboard sensors |
| J15 | J3 | I2C SCL | Populated: CC3220 SCL connected to onboard sensors with pullup |
| | | | Open: CC3220 SCL disconnected from onboard sensors |

#### *2.4.2.1 Default I$^2$C Addresses*

Table 3 lists the default I$^2$C addresses of the onboard sensors.

**Table 3. Default I$^2$C Addresses (of Onboard Sensors)**

| Sensor Type | Reference Designator on LP (Rev. A) | Reference Designator on LP (Rev. B) | Part Number (Manufacturer) | Default Slave Address (Hex) |
|---|---|---|---|---|
| Temperature (MEMS IR Thermopile) | U10 | U6 | TMP006 (TI) | 0x41 |
| Accelerometer (Triaxial) | U11 | U10 | BMA222E (Bosch) | 0x18 |

### 2.4.3 Power Connections

The board can be powered by using the onboard micro USB connector. An onboard DC-DC converter provides 3.3 V for the CC3220 and the rest of the board to operate. This supply can be isolated from the DC-DC using the jumpers on the board. See the yellow jumpers in Figure 8.

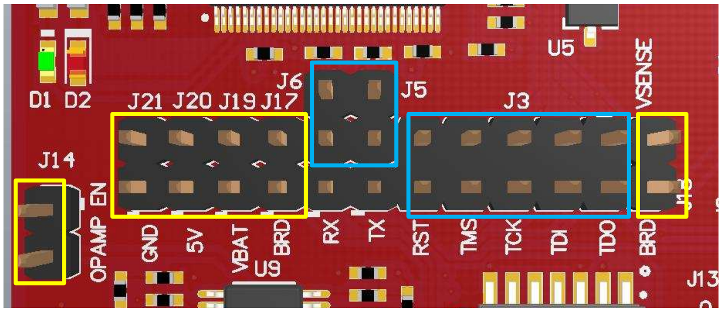


**Figure 8. Power Jumpers J14, J21, J20, J19, J17, and J18**

**NOTE:** The blue jumpers in Figure 8 are previously discussed (see Section 2.4.1) and are populated by default. Figure 8 does not show unpopulated jumpers (which would be populated normally).

Table 4 lists the jumper settings for the LaunchPad power.

**Table 4. Jumper Settings for LaunchPad Power**

| Reference (Rev. A) | Reference (Rev. B) | Use | Comments |
|---|---|---|---|
| J14 | J5 | OPAMP EN | If the jumper is uninstalled, the power supply to the OPAMP is cut off. This can be used to enable low-current measurements. Ensure that this jumper is on to use the OPAMP to drive the input to the ADC. The reference voltage of the ADC is 1.47 V, so up to 3.48 V can be applied to the input of the OPAMP. For the configuration of the OPAMP, see the CC3220 LaunchPad Design Files. |
| J21 | J10 | GND | Ground reference |
| J20 | J29 | +5 VDC power jumper | Connects J19, +5 VDC to emulator section |
| J19 | J12 | Current measurement | Measures the current flowing into the CC3220 device. Also includes the serial flash and any stacked BoosterPack. |
| J17 | J13 | Board power | Short: Supply the board power from the onboard DC-DC converter. The board power includes the sensors, LED, and the OPAMP used to drive the ADC input. |
| J18 | J28 | VSENSE | Used to power the level shifters on the emulator side of the board. The level shifters can be powered by shorting this jumper. Removing this jumper enables low-current measurement. |

The board can be powered by an external supply when USB power is not available, by using either J22 or J23. J24 is also available to remove any current draw from LEDs being driven by the GPIOs, see Table 5.

**Table 5. External Supply Connections and LED Enable Jumper**

| Reference (Rev. A) | Reference (Rev. B) | Use | Comments |
|---|---|---|---|
| J19 | J12 | Alternative 3.3-V power input | Can be used to power the board from an external 3.3-V supply; this can be used to test the VBAT voltage range as the reverse voltage protection diode on J22 drops the input by approximately 150 mV. |
| J23 | J19 | 5-V power input | Used to power the board from an external 5-V supply. |
| J22 | J20 | 3.3-V power input | Used to power the board from an external 3.3-V supply. J22 has built-in reverse voltage protection to prevent the battery from being plugged in the reverse manner. |
| J24 | J9 | LED EN | If uninstalled, the LEDs connected to the GPIO are disabled; this can be used to enable low-power measurements. |

### 2.4.4 Reset Pullup Jumper

Table 6 lists the reset pullup jumper.

**Table 6. Reset Pullup Jumper**

| Reference (Rev. A) | Reference (Rev. B) | Use | Comments |
|---|---|---|---|
| J9 | J26 | RESET pullup | Install this jumper to enable the pullup resistor on the nRESET pin of the device, when the board is powered from an external supply. |

### 2.4.5 Sense on Power (SOP)

The CC3220 can be set to operate in four different modes, based on the state of the sense-on-power (SOP) lines. These SOP lines are pins 21, 34, and 35 on the CC3220 device. Table 7 describes the state of the device, and Figure 9 shows the SOP jumpers.

**Table 7. SOP[2:0] (J13 on LaunchPad)**

| Binary Value | Function |
|---|---|
| 000 | Functional mode and 4-wire JTAG |
| 001 | Functional mode and 2-wire JTAG |
| 010 | Functional mode and flash programming |
| 011 | Factory default |
| 100 | Flash programming |

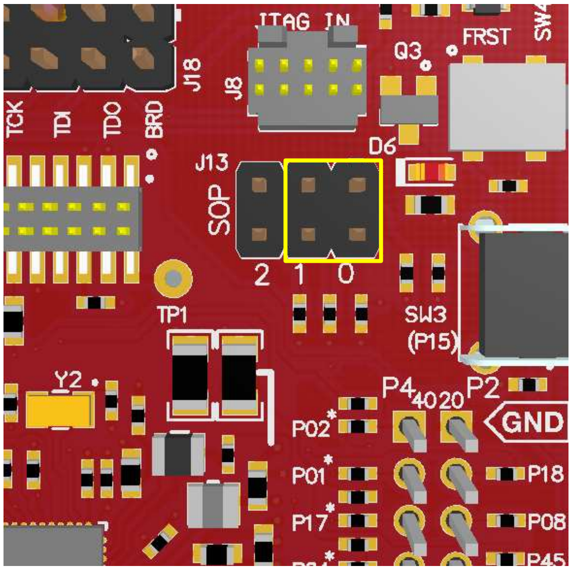


**Figure 9. SOP Jumpers (Default Setting Shown)**

**NOTE:** SOP[2:0] corresponds to J13 in the LaunchPad schematic design.

**NOTE:** No jumpers on the block ensure that the line is pulled low using 100-kΩ pulldown resistors. Placing the jumper pulls the pin high using a 270-Ω resistor.

### 2.4.6 UART Signals

The board supports a USB-based virtual COM port, using the Tiva™ ARM® MCU. The LaunchPad is shipped with the UART lines from the CC3220 connected to the UART on the Tiva MCU. The CC3220 UART can also be routed to the 20-pin connector for use as a GPIO or external UART. The selection is performed using jumpers on the board.

Figure 10 shows the UART routed to USB COM port and Figure 11 shows the UART routed to 20-pin header connector.

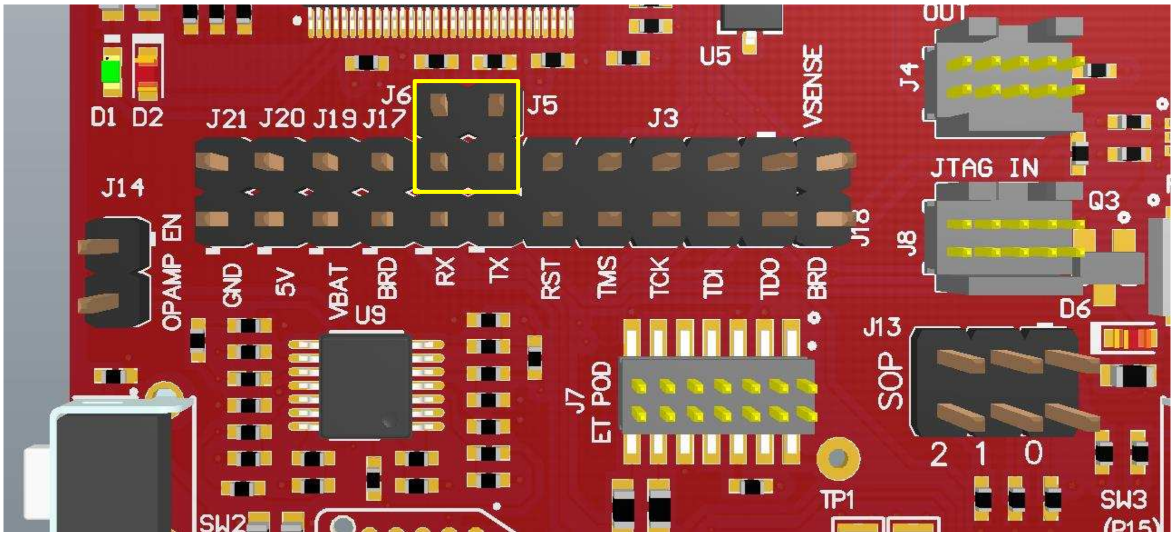


**Figure 10. UART Routed to USB COM Port**

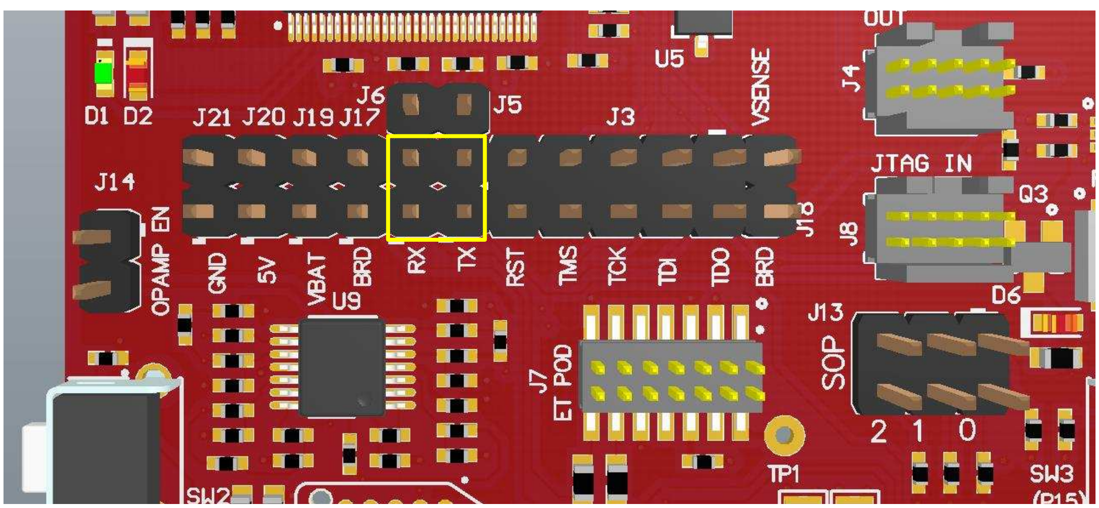


**Figure 11. UART Routed to 20-Pin Header Connector**

### 2.4.7 Push-Buttons and LED Indicators

Table 8 list the push-button definitions.

**Table 8. Push-Button Definitions**

| Reference (Rev. A) | Reference (Rev. B) | Use | Comments |
|---|---|---|---|
| SW1 | SW1 | RESET | This is used to reset the CC3220 device. This signal is also output on the 20-pin connector to reset any external BoosterPack which may be stacked. The reset can be isolated using the jumper block at the center of the board. |
| SW2 | SW3 | GPIO_13 | When pushed, GPIO_13 is pulled to VCC. |
| SW3 | SW2 | GPIO_22 | When pushed, GPIO_22 is pulled to VCC. |
| SW4 | SW4 | Factory default | Pressing this button and toggling RESET restores the factory default image on the serial flash. This can be used to recover a corrupted serial flash, provided the s-flash was programmed with a recovery image. |

Table 9 lists the LED indicators.

**Table 9. LED Indicators**

| Reference (Rev. A) | Reference (Rev. B) | Color | Use | Comments |
|---|---|---|---|---|
| D1, D2 | D2, D9 | Green and Red | Debug | Indicates the state of the JTAG emulator. For TI use only. |
| D3 | D1 | Yellow | nRESET | Indicates the state of the nRESET pin. If this LED is on, the device is functional. |
| D6 | D8 | Red | Factory Reset | Indicates that the push-button for the factory reset is pressed. |
| D7 | D4 | Red | Power | Indicates when the 3.3-V power is supplied to the board. |
| D8 | D5 | Green | GPIO_11 (1) | On when the GPIO is logic-1. |
| D9 | D6 | Yellow | GPIO_10 (1) | On when the GPIO is logic-1. |
| D10 | D7 | Red | GPIO_09 | On when the GPIO is logic-1. |

(1) GPIO_10 and GPIO_11 are also used as $I^2C$. Thus, when the pullup resistors are enabled, the LEDs are on by default, without configuring the GPIOs.

### 2.4.8 BoosterPack Header Pin Assignment

The TI BoosterPack header pinout specification is at Build Your Own BoosterPack. Also see the BoosterPack Pinout Standard.

The CC3220 LaunchPad follows this standard, with the exception of naming. (P1:P4 is used instead of J1:J4.) See Figure 12 for CC3220 pin-mapping assignments and functions.

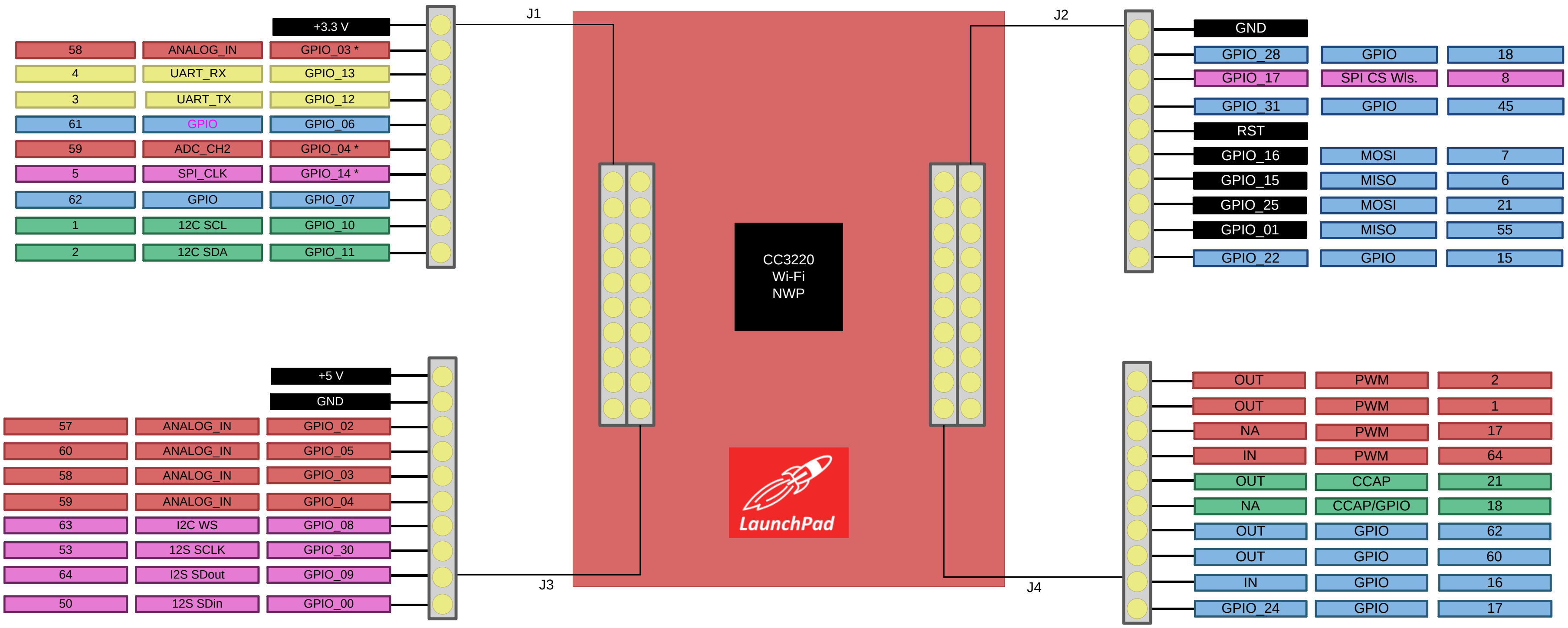




**Figure 12. CC3220 BoosterPack Header Pin Assignments**

---

**NOTE:** RESET output is an open-drain-type output and can only drive the pin low. The pullup ensures that the line is pulled back high when the button is released. No external BoosterPack can drive this pin low.

---

All the signals are referred to by the pin number in the SDK; Figure 12 shows the default mappings. Some of the pins are repeated across the connector. For instance, pin 62 is available on P1 and P4, but only P1 is connected by default. The signal on P4 is marked with an asterisk (*) to signify that it is not connected by default. The signal can be routed to the pin by using a 0-Ω resistor in the path. For the exact resistor placement, see the CC3220 SimpleLink Wi-Fi Wireless MCU LaunchPad Board Design Files.

## 2.5 Power

### 2.5.1 USB Power

The LaunchPad is designed to work from the USB-provided power supply. The LaunchPad provides addresses as a bus-powered device on the computer. When the board is powered from the USB connector, the jumpers must be placed on the following headers, as shown in Figure 13.

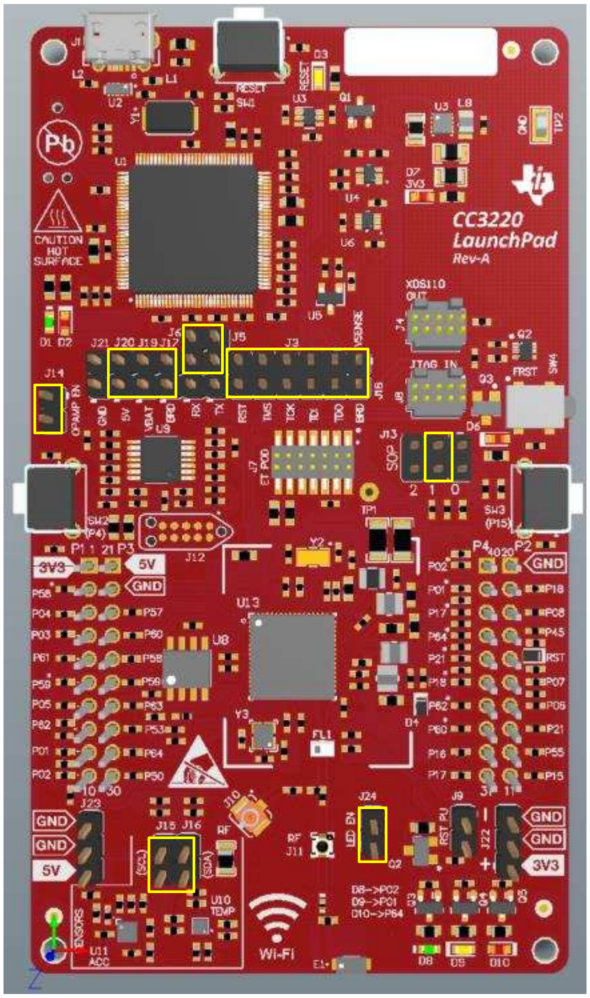

**Figure 13. Powering From USB Jumper Settings**

### 2.5.2 Battery Power

The LaunchPad can also be powered from an external battery pack by feeding the voltage on the J22 header. This input features reverse voltage protection to ensure that the board is not damaged due to an accidental reverse voltage. Perform the following steps before using the board with a battery.

1. Remove the USB cable.
2. Plug in the battery pack on J22 with the correct polarity (see Figure 14).
3. Connect the jumper across J17 and J19 as shown in Figure 14.

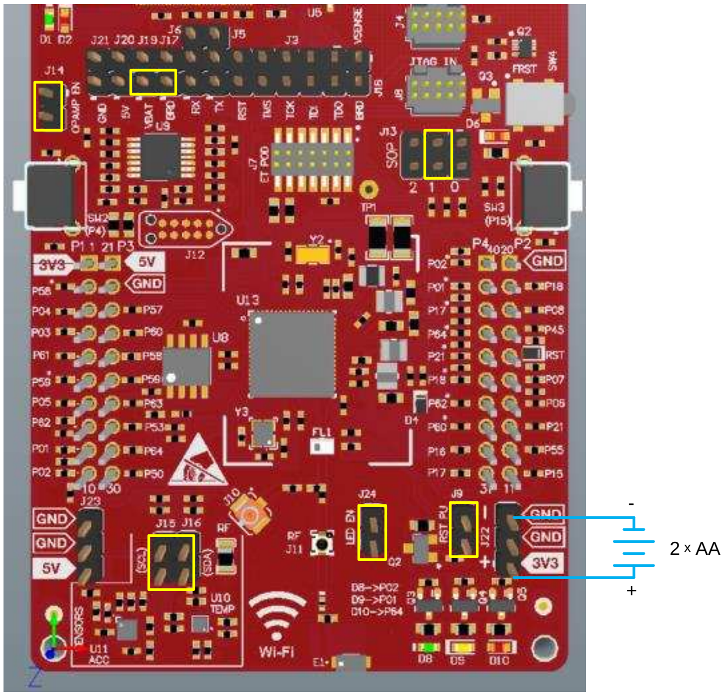


**Figure 14. Powering the CC3220LP From Battery**

### 2.5.3 Battery Powering Only the CC3220 and U8 (Onboard Serial Flash)

In some cases, there may be a requirement to power only the CC3220 and the serial flash from the battery. The usage may not require LEDs, OPAMP for the ADC, and the sensors. In this case, the other sections can be powered off by removing the appropriate jumpers. Ensure that a jumper is placed on RST_PU (J9) of the LaunchPad. The board would appear as shown in Figure 15.

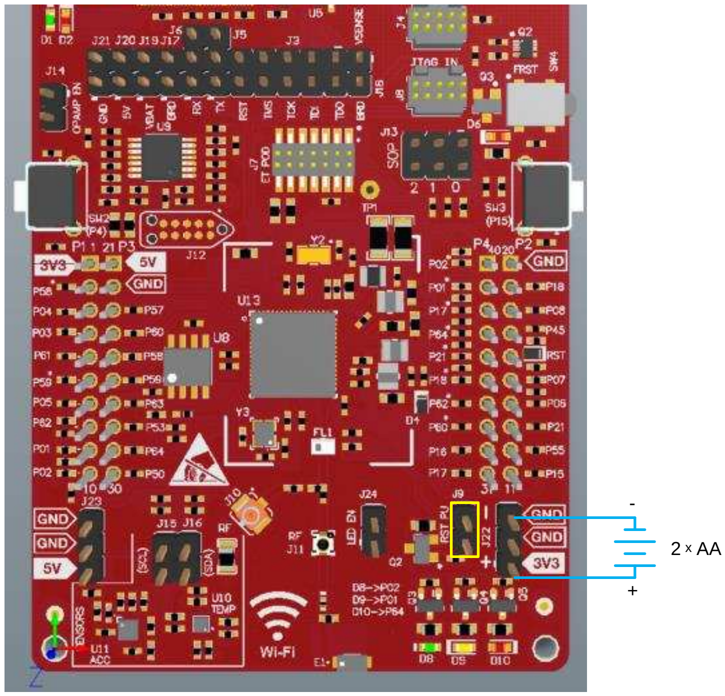


**Figure 15. Only CC3220 and Serial Flash Powered by Battery**

## *2.6 Isolated Current Measurement of the CC3220*

To measure the current draw of the CC3220 when powering with a USB cable, use the VBAT jumper on the jumper isolation block (J19). The current measured in this mode includes only the CC3220 current and the serial flash current, and no external blocks. However, if a GPIO of the CC3220 is driving a high-current load such as an LED, then that is also included in this measurement.

### 2.6.1 Low-Current Measurement With USB Power (< 1 mA)

Follow these steps to measure ultra-low power operation of the CC3220:

1. Remove the VBAT jumper (J19); attach an ammeter across this jumper, as shown in Figure 16. Ensure that jumpers are placed on J5, J6, and J17. The CC3220 device should not drive any high-current loads directly (such as an LED) because this can draw a large current.

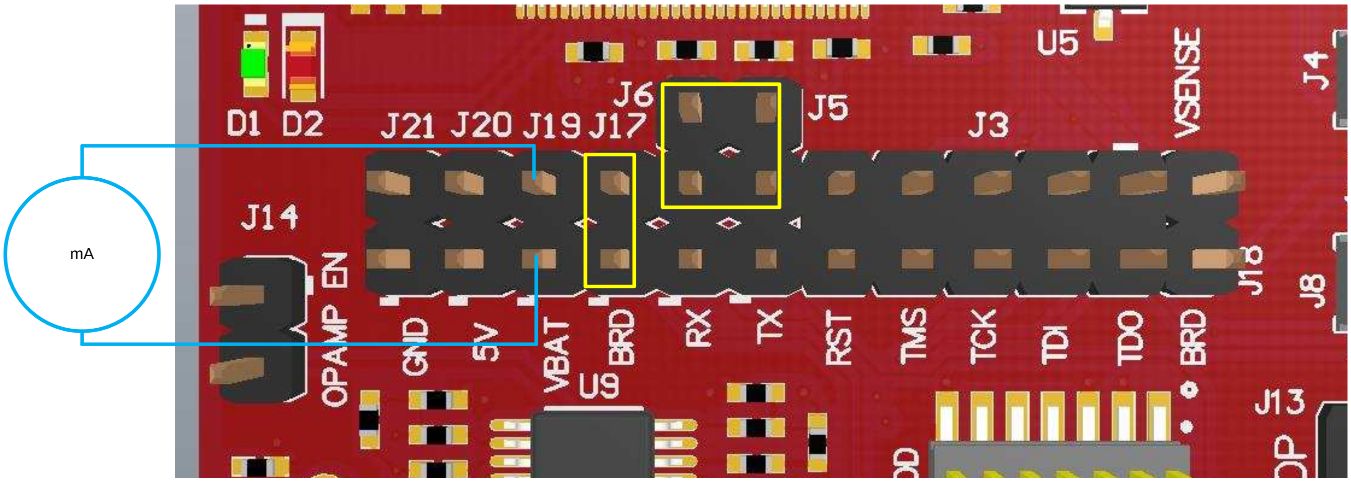


**Figure 16. Low-Current Measurement (<1 mA)**

2. Begin target execution and set the device to low-power modes (LPDS or hibernate).
3. Measure the current. If the current levels are fluctuating, it may be difficult to get a stable measurement. It is easier to measure quiescent states.

---

**NOTE:** To measure the low-power numbers, remove the LEDs (D8, D9, and D10 on the board) by removing the LED EN jumper (J24). Similarly, the shutdown mode leaks approximately 33 µA into the pullup resistor (R136) on the nRESET pin. This pullup resistor must also be removed to measure the total current below 1 µA in shutdown mode.

---

### 2.6.2 Active Power Measurements (>1 mA)

Follow these steps to measure active operation of the CC3220:

1. Remove the VBAT jumper (J19).
2. Solder a 0.1-Ω resistor on a wire, which can be connected to an oscilloscope, as shown in Figure 17. Or, attach a jumper wire between J19 so that it can be used with a current probe.

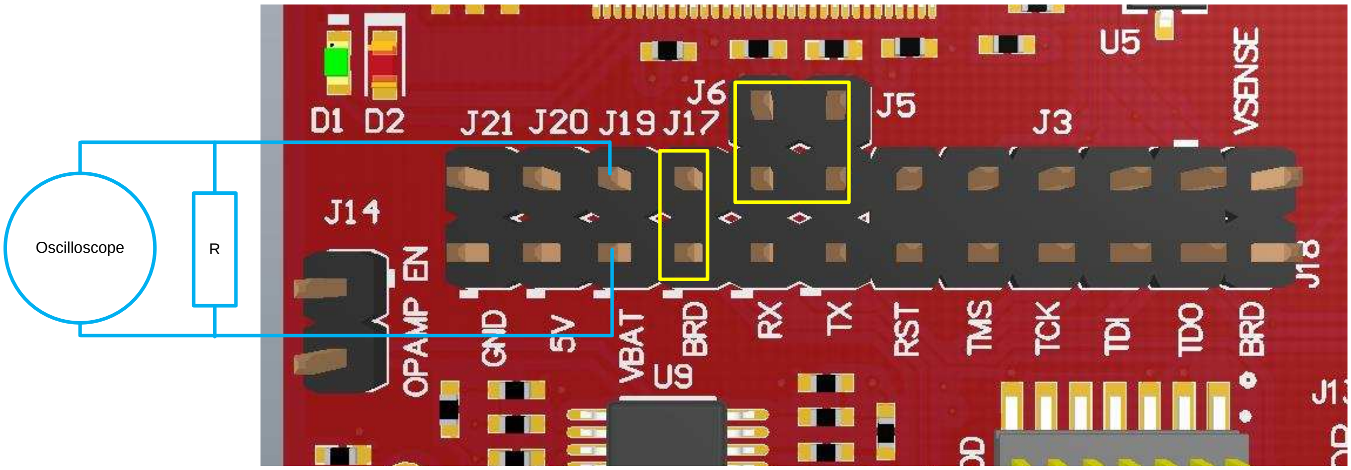


**Figure 17. Active Power Measurements (>1 mA)**

3. Measure the voltage across the resistor using an oscilloscope with a differential probe. (For the current probe, coil the wire around the sensor multiple times for good sensitivity.) An ammeter can also be used for this measurement, but the results may be erroneous due to the switching nature of the current.

## 2.7 RF Connections

### 2.7.1 AP Connection Testing

By default, the board ships with the RF signals routed to the onboard chip antenna, as shown in Figure 18.

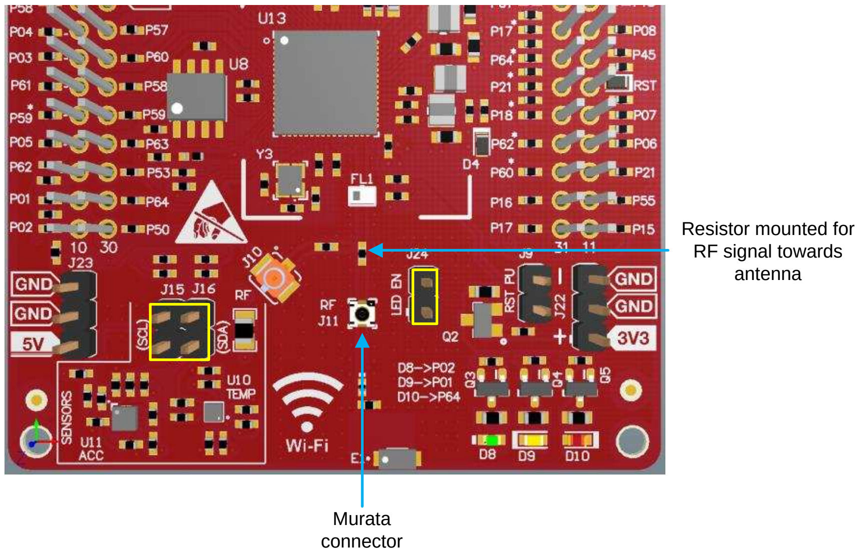


**Figure 18. Using Onboard Antenna (Default Condition)**

A miniature UMC connector (Murata MM8030-2610) provides a way to test in the lab using a compatible cable. Alternately, for testing the conducted measurement a U.FL connector is provided on the board. A rework must be performed before this connector can be used; this involves swapping the position of a resistor. The modified board would appear as in Figure 19.

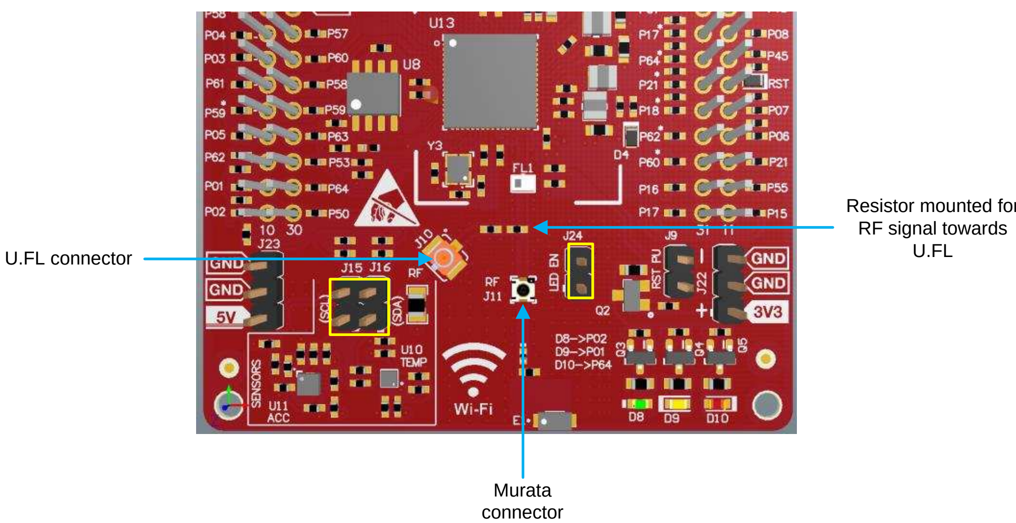


**Figure 19. Board Modified for External Antenna Connections**

## 2.8 Assembly Drawing

Figure 20 shows the top layer assembly drawing of the CC3220x LaunchPad (Rev. A).

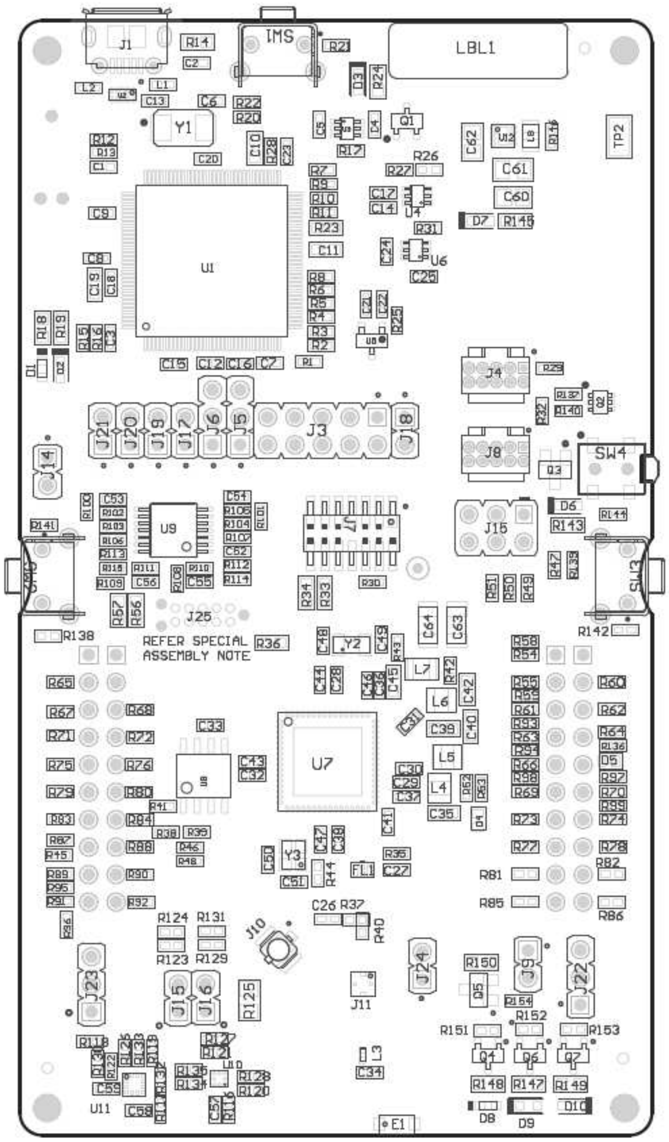


**Figure 20. CC3220x LaunchPad Top-Layer Assembly Drawing**

## 2.9 Design Files

### 2.9.1 Hardware Design Files

All design files, including schematics, layout, Bill of Materials (BOM), Gerber files, and documentation are available for download from CC3220-LAUNCHXL-RD.

## 2.10 Software

All design files, including firmware patches, software example projects, and documentation are available from the SimpleLink Wi-Fi Platform page.

The software development kit (SDK) for the CC3220 LaunchPad can be obtained from CC3220SDK.

# 3 Development Environment Requirements

The following software examples with the LaunchPad require an integrated development environment (IDE) that supports the CC3220 device.

The *CC3220, CC3220S, CC3220SF SimpleLink™ Wi-Fi® and Internet of Things Solution, A Single-Chip Wireless MCU* programmer's guide has detailed information about software environment setup with examples. See this document for further details on the software sample examples.

## 3.1 CCS

CCS 6.0 or higher is required. When CCS is launched, and a workspace directory is chosen, use *Project → Import Existing CCS Eclipse Project*. Direct it to the desired demo project directory containing main.c.

## 3.2 IAR

IAR 6.70 or higher is required. To open the demo in IAR, choose *File → Open → Workspace…*, and direct it to the *.eww workspace file inside the \IAR subdirectory of the desired demo. All workspace information is within this file.

The subdirectory also has an *.ewp project file; this file can be opened into an existing workspace, using *Project → Add-Existing-Project…*.

# 4 Additional Resources

## 4.1 CC3220 Product Page

For more information on the CC3220 device, visit the CC3220 product page, which includes the *CC3220x SimpleLink™ Wi-Fi® Wireless and Internet-of-Things Solution, a Single-Chip Wireless MCU Data Sheet* and key documents such as the *CC3220, CC3220S, CC3220SF SimpleLink™ Wi-Fi® and Internet-of-Things Technical Reference Manual* and the http://www.ti.com/SimpleLinkWiFi-Wiki, which contains information on getting started, hardware details, software details including porting information, testing and certification, support, and the CC3220 community.

## 4.2 LaunchPad™ Wiki

Most updated information is available on the CC3220 Wiki page.

# Revision History

**Changes from February 7, 2017 to January 24, 2018** **Page**