

Application Report

CC313x Production Line Guide



Table of Contents

1 Production Line Overview	2
2 Programming the CC313x in the Production Line	2
3 Creating the Gang Image	3
4 Programming Directly Through SPI	3
5 Programming Over UART	4
5.1 Using the Uniflash CLI.....	4
5.2 Using Embedded Programming.....	4
5.3 Configuration of the UART.....	4
5.4 UART Hardware Connection Using the FTDI Emulation Board.....	4
6 Over-the-Air Programming	6
7 Production Line RF Testing	6
7.1 Testing Software Options.....	6
8 Revision History	6

List of Figures

Figure 1-1. Production Line Overview.....	2
Figure 4-1. Programming Through SPI.....	3
Figure 5-1. CC31XXEMUBOOST Jumpers.....	5
Figure 5-2. Pin Map.....	5
Figure 5-3. Programming Through USB.....	5

Trademarks

Texas Instruments™, SimpleLink™, and Internet-on-a chip™ are trademarks of Texas Instruments. All other trademarks are the property of their respective owners.

1 Production Line Overview

Texas Instruments™ provides a number of resources that assist manufacturers using the CC313x device to produce products quickly and efficiently. To ensure products are designed with efficient production in mind, TI provides reference design collateral and application notes for schematic and PCB design. Software and hardware tools have been developed for programming and testing the CC313x device in the production line. In addition, Over-The-Air programming functionality allows for products to have their software updated periodically, even after they have been deployed.

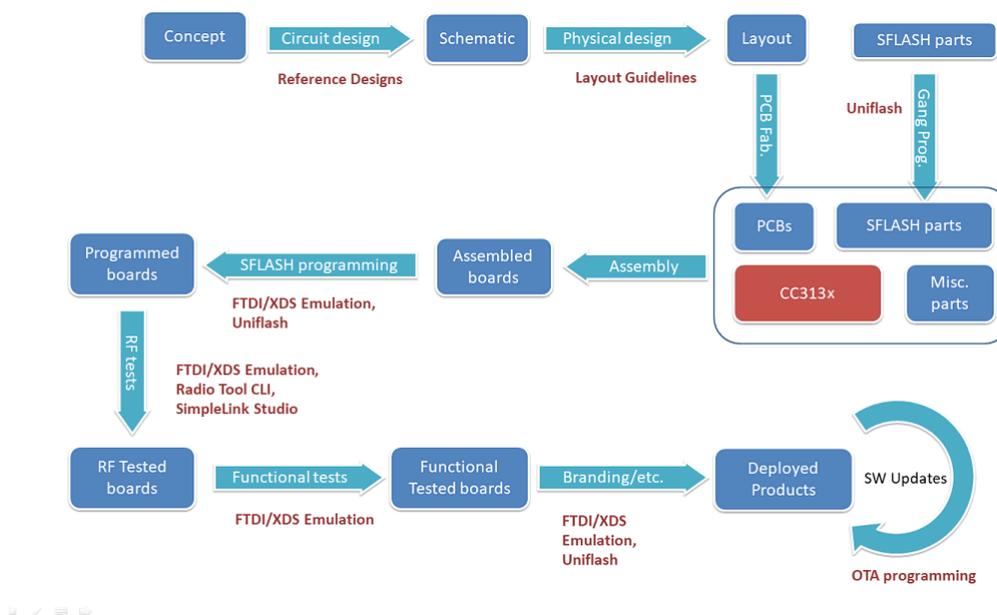


Figure 1-1. Production Line Overview

2 Programming the CC313x in the Production Line

Production with the CC313x device requires a file system to be created on the serial flash for proper operation. At the minimum, this includes the service pack that contains the necessary software updates and additional features. Configuration files may also be written, which provide an initial configuration for the device upon startup. Security certificates and other content such as web pages, images, scripts, and so forth, can also be included. Although most of this content is usually written during production, all content including the host program and the service pack can be continuously updated over the lifetime of the product.

All CC313x devices must initially be programmed with a gang image. The gang image usually contains all of the data content necessary for a functional product. The serial flash vendor may be able to pre-program the serial flash parts with the gang image before they are sent to the manufacturer. There are two methods of loading the gang image onto the CC313x serial flash:

- UART programming – A PC-based utility or embedded device can be used for indirectly programming the serial flash through UART.
- Direct SPI programming – Write directly to the serial flash through SPI, using an external off-the-shelf programming tool.

Programming through UART has the benefit of using the same hardware interface and tools used for development and debugging in the production line. Programming the serial flash directly through SPI has the benefit of having a faster time in performing the write operations, since programming through the UART interface has a limitation of communicating at a speed of 921600 bps. A dedicated external programmer is required for programming over SPI. A single SPI programming tool may be capable of programming multiple serial flash devices simultaneously, whereas a separate programming device is required for programming each device through UART. Upon boot, the CC313x device detects the presence of a gang image, and converts it to the

target file system of the device. This formatting process is performed exclusively by the SimpleLink™ device, and does not require any inputs from external interfaces. It does, however, extend the duration of the first power-up. The formatted file system is created alongside the gang image. Thus, the serial flash needs to contain both the gang image and the file system. The gang image is retained to allow for factory default restore functionality. See [SimpleLink™ Wi-Fi® CC3x3x Serial Flash](#) for more details about serial flash usage in CC313x devices.

Generally, a single gang image is used per product release. There are some device-specific parameters that can be reconfigured through Uniflash CLI during the programming phase. Additional files can be added to the serial flash using the MCU program afterwards. If the MCU program can perform over-the-air programming, the serial flash contents can be updated by downloading content from the internet or from a local connection. Loading more content through serial interface is also possible if the MCU program allows for it.

Configure the gang image correctly for production, because some things cannot be updated through over the air updating:

- The gang image should be configured for production mode.
- The Trusted Root-Certificate Catalog used should be the production version.
- Do not use the dummy certificates and keys provided in the certificate-playground of the SDK for production.
- The formatted serial flash size cannot be changed.

3 Creating the Gang Image

The software tool used for creating gang images for the CC313x device is Uniflash (4.1 or greater). The country code for the device should be set for allowing RF compliance with regional governmental regulations (such as US, EU, or JP). This ensures maximum RF performance, while remaining compliant with regional law. PHY calibration mode should also be set during gang programming. Other configuration options, such as startup, WLAN role, and WLAN connection policy, can also be set during programming. The image can be secured with a private key. This is done when the code must be secured, and the serial flash is programmed in a location different than where the assembly takes place. Using an encrypted image necessitates the use of the UART interface for programming. See [CC3x20, CC3x35 SimpleLink™ Wi-Fi® Internet-on-a chip™ Solution Built-In Security Features](#) for more details about this feature.

4 Programming Directly Through SPI

The serial flash device is programmed directly, starting at memory offset 0, with one of the gang image files created by Uniflash: either Programming.bin or Programming.hex, depending on the SPI programmer. The serial flash may be programmed after board assembly, provided some schematic and layout considerations are taken:

- The serial flash SPI interface pins must be brought out for physical contact with the programmer (such as headers or test pads).
- The SPI lines must not be driven by any other source while programming.
- The CC313x device is held in reset during programming to prevent I/O contention.

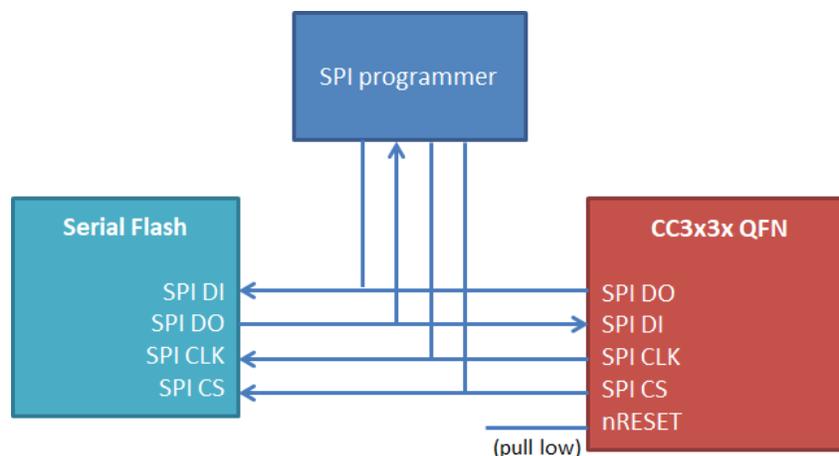


Figure 4-1. Programming Through SPI

5 Programming Over UART

The gang image can be programmed to the serial flash through the CC313x device through UART communication. The device is communicated with by using a bootloader protocol over UART. When using Uniflash, this can be accomplished using the CC31XXEMUBOOST.

5.1 Using the Uniflash CLI

Uniflash communicates with the CC313x device using the bootloader protocol. The protocol includes a command line interface, which can be used in batch files and scripts for the purposes of programming the CC313x device in the production line through UART. Refer to [SimpleLink™ Wi-Fi® CC3x20, CC3x3x UniFlash ImageCreator](#) for information on how to use Uniflash ImageCreator.

5.2 Using Embedded Programming

In addition to using Uniflash, an embedded device can be used to program the CC313x device through UART. This can be advantageous if a larger number of programming setups are needed to operate simultaneously, and using multiple PCs with Uniflash would be cost prohibitive. Refer to [CC313x and CC323x Simplelink™ Wi-Fi® Embedded Programming User's Guide](#) for information on programming using an embedded device for UART programming.

5.3 Configuration of the UART

Programming the serial flash device through the CC313x UART interface requires the use of the following CC313x pins:

- 55 – UART TX
- 57 – UART RX
- 32 – nRESET or 2 – nHIB

The UART TX and RX pins are used for data transfer. RTS and CTS signals are not used. The nRESET pin is used to reset the device. The UART data transfer occurs at 921600 bps.

The UART configuration is as follows:

- Baud rate: 921600
- Data bits: 8 bits
- Flow control: None
- Parity: None
- Stop bits: 1
- Polarity: Positive

The CMOS logic level specifications for the UART can be found in the *Electrical Characteristics* of the [CC3135 SimpleLink™ Wi-Fi® Dual-Band Network Processor, solution for MCU applications Data Sheet](#) or the [CC3130 SimpleLink™ Wi-Fi® Wireless Network Processor with Coexistence for MCU Applications Data Sheet](#).

5.4 UART Hardware Connection Using the FTDI Emulation Board

The CC31XXEMUBOOST can provide the required USB to UART/GPIO interface for programming the serial flash through Uniflash. The PC drivers for this board are included in the installation of Uniflash. The CC31XXEMUBOOST is connected through USB from socket J6 to the PC. The jumpers on the CC31XXEMUBOOST should be connected as shown in [Figure 5-1](#).

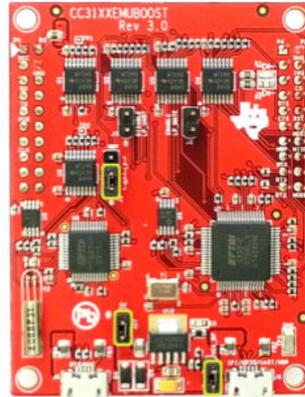


Figure 5-1. CC31XXEMUBOOST Jumpers

The CC31XXEMUBOOST uses a logic level of 3.3 V by default, but there are level shifters on the CC31XXEMUBOOST, and it can be powered in-dependently with a different I/O voltage by removing the jumper from J4, and applying the external power to pin 1 of jumper J4. On the product being programmed, the relevant CC313x pins must be brought out for physical contact with the programmer (such as male headers or test pads), and must be driven by no other source while programming. The HIB pin on the CC31XXEMUBOOST is temporarily pulled to GND by the Uniflash utility at the beginning of the flashing procedure, which is necessary to synchronize the device and the PC software. The CC31XXEMUBOOST is configured to receive commands from the Uniflash program for toggling its HIB pin at the appropriate times during the programming process.

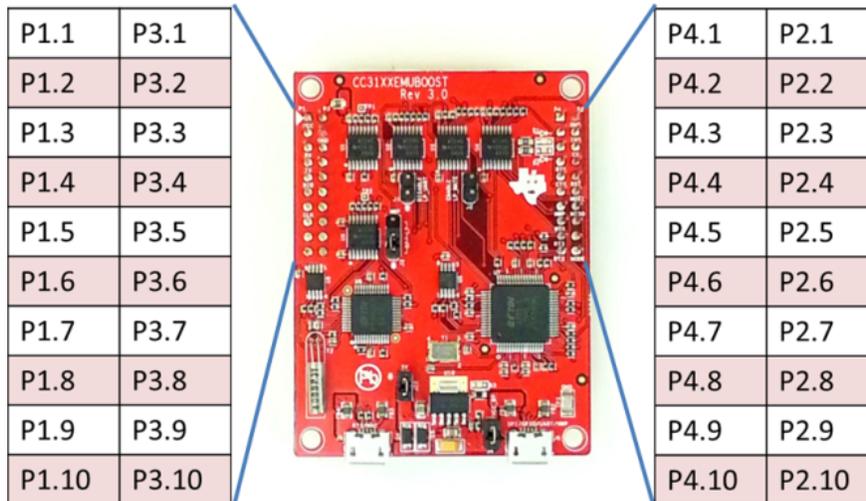


Figure 5-2. Pin Map

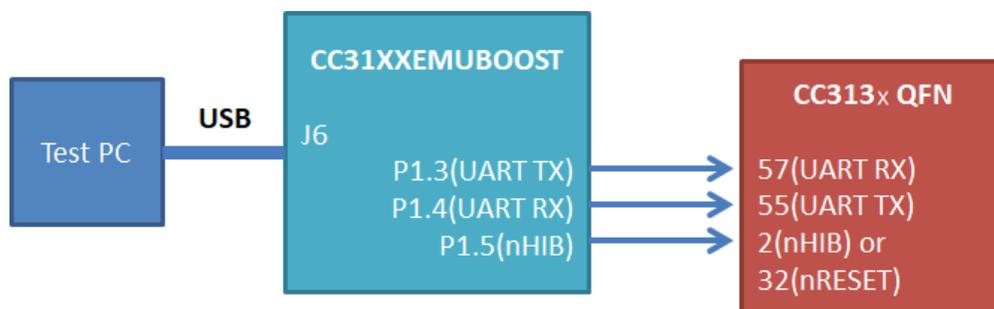


Figure 5-3. Programming Through USB

6 Over-the-Air Programming

The CC313x device has the capability for Over-the-Air programming, which allows files to be written and updated over a network connection. An OTA programming library is available in the CC313x SDK. Using OTA in the production line can enable faster data transfer over other methods in some instances. To use OTA in the production line, a PC on the local network can run the OTA server. The service pack should be updated before doing OTA or using any wireless functionality. For the fastest transfer of data using OTA, TI recommends minimizing RF congestion in the production environment. See the [SimpleLink™ Wi-Fi® CC3x20, CC3x3x Over-the-Air Update](#) for more information.

7 Production Line RF Testing

Testing of hardware and software functionality is highly specific to each product, but there are some tools Texas Instruments has made available to assist with testing RF performance. The CC313x device can be instructed to perform RF testing operations in a number of ways:

- The host program may have a built-in subroutine dedicated to RF testing. This could be run once upon first power-up, or could be triggered using a special external command.
- A script using the Radio Tool CLI could control the CC313x device from a PC, as detailed in [SimpleLink™ Wi-Fi® CC3x20, CC3x3x Radio Tool](#).
- The CC313x device can be controlled by interfacing with a dedicated RF tester.

7.1 Testing Software Options

7.1.1 MCU-Controlled RF Testing

SimpleLink™ API functions are available that can put the CC313x device into modes used for RF testing. This allows for:

- Transmission of packets at specified channels, modulations, and so forth.
- Receipt of packets while gathering statistics for RSSI, modulation, and so forth.
- Carrier wave transmission

For comprehensive information about the SimpleLink™ API for transceiver mode, see Chapter 13 in the [SimpleLink™ Wi-Fi® CC3x20, CC3x3x Network Processor](#).

7.1.2 Testing With an Access Point

A straightforward method of checking for acceptable RF performance is to put the device being tested through a trial run in an RF environment with worst-case conditions. The trial run begins with the device under test connecting to an access point, then communicating with either a PC on the local network or with a remote cloud server. The communication between the device under test and its peer can be monitored for reliability and speed. To get consistent and relevant results for all devices being tested, some actions may be taken with respect to the controlling RF environment for this type of testing:

- Minimize unintentional RF congestion in the test area. This can be accomplished by turning off other nearby 2.4-GHz band devices, or performing the testing in an RF-shielded enclosure.
- Introduce controlled RF congestion. This can involve something such as having another device connected to the same access point, which transmits a steady stream of packets to the access point.
- Introduce attenuation in the antenna path for the access point, or place at a distance from the device being tested.
- Set the access point to communicate only on a specific channel, modulation, and so forth.

8 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision * (January 2019) to Revision A (September 2020)	Page
• Changed the document title and throughout document to include CC3130 devices.....	2

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2022, Texas Instruments Incorporated