



Alexander Paul

ABSTRACT

This Bluetooth® low Energy product brief describes the Bluetooth software content that is supported by the CC23xx device platform, which is based on an ARM Cortex-M0+ core and offers a wide variety of peripheral options. This document provides a high level overview of the features and capabilities of the Bluetooth Low Energy Stack for the CC23XX platform without deep technical details. Further information and details can be found in the [TI BLE5-Stack User's Guide](#) which is provided within the [SimpleLink Low Power F3 SDK](#) documentation. Development and training can be started at [SimpleLink™ Academy for CC23xx](#).

Table of Contents

1 Introduction	2
2 Certification and Qualification	3
2.1 Bluetooth Qualification	3
2.2 Regional Compliance	3
3 Reference Examples	3
3.1 CC2340 Chipsets	4
3.2 Flash and RAM Allocation	5
4 Software Stack	5
4.1 BLE5-Stack Configurations	6
4.2 Software Offering	6
4.3 Supported PHYs	6
4.4 Supported Features	6
4.5 Multi-Connection	7
4.6 Coexistence (Planned)	7
5 Security	7
6 Performance and Test Data	8
6.1 Connection	8
6.2 Advertising	8
6.3 Stability Testing	9
6.4 Interoperability	9
7 Tools and Development Support	9
7.1 SmartRF Packet Sniffer 2	9
7.2 Smart RF Studio 8	10
7.3 Energy Trace	10
7.4 Code Composer Studio	10
7.5 SimpleLink Connect App	10
7.6 Uniflash	10
7.7 Antenna Reference Designs	10
7.8 Design Review Service	10
7.9 SysConfig	10
7.10 BTool	11
7.11 GitHub	11
7.12 SimpleLink Academy	11
8 Known Limitations	11
9 References	11
10 Revision History	12

Trademarks

SimpleLink™, SmartRF™, Code Composer Studio™ are trademarks of TI.

Bluetooth® is a registered trademark of Bluetooth SIG, Inc.

Windows® is a registered trademark of Microsoft.

Eclipse® is a registered trademark of Eclipse Foundation.

All trademarks are the property of their respective owners.

1 Introduction

Bluetooth Low Energy is a technology to provide a 2.4 GHz personal network that is designed by Bluetooth SIG. Compared to classic Bluetooth, Bluetooth LE was created to minimize the power consumption of devices that require a personal area network like the one implemented for Bluetooth Classic. This is done by transmitting very small packets of data at a time while consuming less power than BR (Basic Rate) / EDR (Enhanced Data Rate) devices. Devices designed for Bluetooth LE functionality try to minimize the peak current the device uses as well as reducing the active time required to implement the device's functionality.

The Bluetooth LE protocol is designed mainly for low power devices that run on a coin sized battery. The Bluetooth LE stack is created to run asymmetrically, which means devices with smaller energy sources are given fewer tasks to extend battery life. TI Bluetooth LE stack is a set of sample applications, tools, APIs and protocol stack libraries that enable engineers to develop Bluetooth 5 applications on SimpleLink devices as shown in [Table 1-1](#).

TI's royalty-free Bluetooth LE stack is Bluetooth 5.3 compatible with all necessary software features, example applications, and documentation to start the development of single mode Bluetooth Low Energy (Bluetooth LE) applications.

Table 1-1. Simplelink Devices

Component	Version
Bluetooth LE Stack Release	3.2.2 (with SDK 7.20)
Bluetooth LE Core Spec	5.3
SDK Release	SIMPLELINK-LOWPOWER-F3-SDK 7.20
IDE Support	CCS V12.3 for: <ul style="list-style-type: none"> • Windows® Windows 10 • Ubuntu 20.04 LTS 64-bit • macOS Ventura 13.4.1
Compiler Support	TI Clang Compiler tools: 2.1.2.LTS, IAR
RTOS Support	FreeRTOSv202104.00
Recommended Development Kits	LP-EM-CC2340R5, LP-XDS110ET and Simplelink Connect app
Supported Devices	CC2340R5
Certifications	Precertified full-featured Bluetooth 5.3 stack

2 Certification and Qualification

2.1 Bluetooth Qualification

Bluetooth Low Energy end products must undergo a Bluetooth SIG (special interest group) qualification and declaration process. In addition to the Bluetooth LE qualification and declaration process, the end device has to fulfill regional requirements like FCC (American) or CE/RED (European).

TI provides qualified Design listings with Qualified design identification for all SimpleLink MCUs running the TI Bluetooth LE stack. This removes the need for Bluetooth LE software stack testing for the end-product when using TI-provided Bluetooth LE stack.

Besides the software stack, for Bluetooth LE-Compliance, one has to perform physical performance tests to determine whether the RF-Interface fulfills minimum Bluetooth LE requirements. Unlike software stack testing, these tests have to be done on every end-device and cannot be reused. However, TI's EVMs typically are certified and listed on End-Product Listings (EPL) and Qualified Design IDs (QDIDs) can be provided. These TI RF-PHY and EPL QDIDs can be used as reference for end product certification if the end product is based on TI reference designs and following guidelines from the corresponding reference design. This can help to reduce the amount of qualification measurements that are required for your end product.

TI provides guides and reference designs, but the end customer is responsible for Bluetooth compliance. For more information on Bluetooth LE certification you can have a look into the [How to Certify Your Bluetooth Product Application Note](#) or the description of the Bluetooth qualification process by the Bluetooth SIG: [Qualify Your Product](#).

2.2 Regional Compliance

Bluetooth devices also have to comply to regional regulations. In the [How to Certify your Bluetooth Product Application Note](#) there is a detailed description on which requirements have to be fulfilled to operate in different areas.

The most common regulatory standards described in the document are: FCC, ARIB, CE, and IC.

3 Reference Examples

TI offers a range of different examples for the CC23xx Chipset as shown in [Table 3-1](#). These examples cover Bluetooth basic examples as well as advanced examples that cover OAD and security features.

More details on provided examples can be found on the [SimpleLink™ low power software development kits \(SDKs\)](#) page.

A list of examples provided with the 7.20 version of the SDK is in [Table 3-1](#).

There is a set of predefined services and profiles which are provided with the TI Bluetooth LE-stack. The provided Bluetooth LE profiles are shown in the table below:

Supported GATT profiles are:

- Simple Gatt
- CGM
- Glucose Meter

Table 3-1. CC23xx Chipset Examples

Example	Description
basic_ble	The basic_ble example prepares for custom development with TI-Bluetooth LE stack and gives a good overview on the environment. Afterwards, basic_ble shows how to interface tasks and take over control of the GATT table and the Bluetooth LE stack.
basic_ble_oad_offchip	The Bluetooth LE OAD offchip example demonstrates OAD functionality for designs with external flash attached. For more information on the OAD off-chip operation, please look into the BLE5-stack User's Guide .

Table 3-1. CC23xx Chipset Examples (continued)

Example	Description
basic_ble_oad_onchip	The Bluetooth LE oad onchip example demonstrates OAD functionality for designs using only internal flash. For more information on the OAD on-chip operation, please look into the BLE5-stack User's Guide .
basic_ble_profiles	An example that implements a Continuous Glucose Monitoring (CGM) profile based on the basic_ble example. This project is a good starter to develop on peripheral-role applications. Supported GATT profiles are: <ul style="list-style-type: none"> • Simple Gatt • CGM • Glucose Meter
basic_ble_ptm	Same as basic_ble above with Production Test Mode (PTM) functionality.
basic_ble_dual_image	OAD example with dual off-chip image.
basic_persistent	Permanently resident application that implements OAD profile.
data_stream	The data stream example helps to get a better understanding of how the TI-Bluetooth LE stack works, and helps to understand how data transfer and data streaming works.
host_test	A network processor project that is interfaced with HCI and Vendor Specific HCI commands. The host test sets the device to Direct Test Mode (DTM).
connection_monitor	The connection_monitor example is designed to 'sniff' a Bluetooth LE connection established between two devices. To monitor a specific connection, the connection monitor should receive the connection's access address, interval, hop increment value, channel map, and the central device sleep clock accuracy (mSCA). This example is typically used for connection-based localization. In such use cases it allows several devices running the connection_monitor example to monitor the signal strength (RSSI) of the packets exchanged in the same Bluetooth connection, offering the required spatial diversity needed for trilateration positioning techniques.

Note

Further examples and demos are published on [TI's public GitHub](#).

To give a good introduction on how to create your own product using the TI- Bluetooth Low Energy-stack, read the article [Bluetooth Low Energy Connections with basic_ble example](#) (part of SimpleLink Academy), which helps to understand the stack and shows how to develop your own applications.

SimpleLink Academy can be reached from the [TI-Developers Zone](#).

- SimpleLink basic_ble example: First overview on Bluetooth LE framework.
- SimpleLink data_stream example: First experiences with Bluetooth LE and how to customize applications.
- Scanning and Advertising Basics: Getting started with scanning and advertising.
- TI SimpleLink Connect App: First experiences with the TI SimpleLink Connect App.
- Bluetooth Low Energy Connections with basic_ble example: Deep dive into the connection concept of Bluetooth.

3.1 CC2340 Chipsets**Table 3-2. CC2340 Chipsets**

Chipset	Flash	RAM	CPU
CC2340R2	256 kB	28 kB	Arm-Cortex M0+ (48 MHz)
CC2340R5	512 kB	36-64 kB	Arm-Cortex M0+ (48 MHz)

3.2 Flash and RAM Allocation

The Flash that is required by TI Bluetooth LE5-stack devices depends on their role. Running the basic_ble example with different roles leads to the flash allocation in [Table 3-3](#):

Table 3-3. Flash Allocation Based on Role

Role	Flash
Single role	95 kB
Multi-role	110 kB

The basic_ble example overall requires about 183 kB of flash. The RAM that is required by the system depends on several aspects. The amount of RAM that is required for different applications can be estimated using [Table 3-4](#).

Table 3-4. RAM Needed Based on Feature

Feature	RAM Needed
RAM usage of a device running the basic_ble example as a peripheral	11 kB
RAM usage of a device running the basic_ble example as a central	13 kB
Default heap size allocated by FreeRTOS	20 kB
Heap allocated for each connection that has to be handled	0.75 kB
Standard system heap allocation on boot and Bluetooth LE stack initialization	≈8 kB

Note

Please note that these values have been determined using CC2340R5 with the 7.20 SDK. Results slightly vary using a different SDK or a different chip.

4 Software Stack

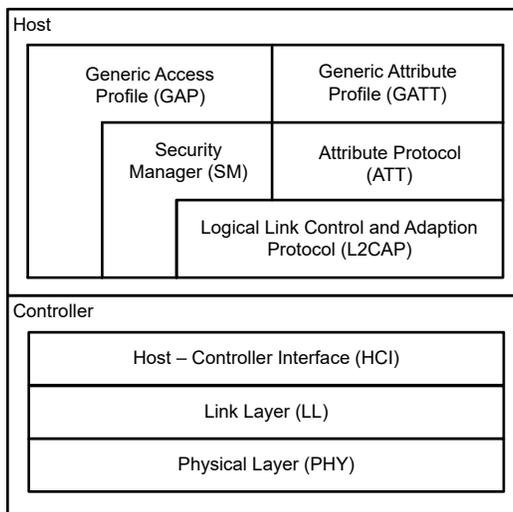


Figure 4-1. Bluetooth Low Energy Protocol Stack

The TI-Bluetooth LE protocol stack consists of a controller and host device as shown in [Figure 4-1](#). This comes from the classic Bluetooth specification.

The controller implements the lowest Layer of the Bluetooth LE stack, and contains the Link Layer (LL) which is responsible to implement a standard-interoperable wireless communication. The controller also implements the data exchange and the handling of the network connection.

The Host section is used to implement the Access and Attribute profiles and security features of the Bluetooth LE stack. This section is the highest layer of the LE stack and defines the role of the device by using different GAP and GATT profiles and different security features.

To interface the TI-Bluetooth LE-stack, the application developer can use a set of C-APIs to implement a Bluetooth LE application on top of the Bluetooth LE stack.

The CC2340R5 can implement all the GAP roles (i.e. broadcaster, peripheral, observer, central), including a combination of these (multirole). See Multi-Connection for details on the achievable performances.

4.1 BLE5-Stack Configurations

The TI-Bluetooth LE stack platform supports two different configurations and is supported to run in single device operation as well as in network processor mode. In single device operation the Host and controller as well as the application are running on the Simplelink wireless MCU.

This configuration is cost effective, but limited in memory and performance. CC2340R5 is recommended to implement this configuration. The second possibility is to use the device as companion option with an external processor.

In this configuration only the application running on the external processor which is communicating to a small radio MCU using the HCI runs on a serial interface. For this application the requirements on memory are fairly low and the CC2340R2 can be used.

4.2 Software Offering

With the SDK, TI provides a development environment. This includes evaluation modules and development boards and the protocol stack itself. Besides the Bluetooth LE stack, there are Drivers and a set of examples and trainings provided to help kickstart developing Bluetooth LE applications with TI.

- TI Drivers provides a device register abstraction layer and is used by software to control SimpleLink Devices on the lowest level.
- The Bluetooth Low Energy protocol stack is provided in library form with parts of the protocol stack stored in the SimpleLinks devices Flash.
- Sample applications and profiles make starting development of Bluetooth LE applications using both custom and adopted methods easier.

4.3 Supported PHYs

TI Bluetooth LE-stack offers the mandatory LE 1M PHY and several other PHYs listed in [Table 4-1](#) to adapt to different applications.

A more detailed description on the available PHYs can be found in the Physical Layer section in the [TI BLE5-Stacks User's Guide](#).

Table 4-1. PHYs

PHY	Transmission Rate	Approx. Max. Application Data Rate	Description
LE 1M	1 Mbit/s	750 kbps	Supports data transfer with mandatory data rate of 1 Mbps
LE 2M	2 Mbit/s	1400 kbps	Supports data transfer with up to 2 Mbps with a reduced range
LE Long Range Coded S2	500 kbps	400 kbps	Reduced data rate with up to 2 times higher range
LE Long Range Coded S8	125 kbps	100 kbps	Further reduced data rate with up to 4 times higher range

4.4 Supported Features

The BLE5-stack is a TI-provided set of applications, tools, APIs and libraries that are developed to bring Bluetooth 5 standalone or network processor LE applications to the SimpleLink™ family of wireless MCUs.

TI-Bluetooth-LE protocol stack supports all mandatory features and a wide range of optional features defined in [Bluetooth Core Specifications](#).

[Table 4-2](#) shows key features that are optional but contained in the BLE5-stack for CC2340 devices.

Table 4-2. Optional Key Features in BLE5-stack for CC2340 Devices

		CC2340R2 ⁽¹⁾	CC2340R5
V 4.1	LE Ping	Yes	Yes
V 4.2	LE Data Packet Length Extension	Yes	Yes
	Link Layer Extended Scanner Filter policies	Yes	Yes
V 5.0	LE Advertising Extensions	No	Planned
	High Duty Cycle Non-Connectable Advertising	No	Yes
	LE Periodic Advertising	No	Planned
	LE Channel Selection Algorithm #2	Yes	Yes
	High Output Power	Yes	Yes
V 5.1	Connection-less AoA/AoD	No	Planned
	Connection-oriented AoA/AoD	No	Planned
	Advertising Channel Index Changes	Yes	Yes

(1) Chipset not finalized and available yet.

4.5 Multi-Connection

The CC2340R5 is capable of running with up to 8 connections to peer devices but only 5 with bonding data. The remaining 3 connections can be performed without bonding. To support these devices, TI runs extended testing on the configurations in [Table 4-3](#). TI testing includes a 48 hour stability test for all connected nodes.

Table 4-3. CC2340R5 Configurations

Role	Connections	Peers
Peripheral	8	8 Centrals
Central	8	8 Peripherals
Multi-Role	8	4 Centrals +4 Peripherals

4.6 Coexistence (Planned)

CC2340 series offer a coexistence feature to avoid collisions between Bluetooth Low Energy and WIFI devices. A PTA (Packet Traffic Arbitration) interface is used to decide which device is allowed to use the frequency band. The Coexistence feature requires the WIFI device to run in a Master role. In this case the Bluetooth LE device has to request access to use the antenna. The WIFI chip provides this by Granting access. The 2 Mbps transfer mode implemented in CC2340 offers lower radio communication time and thus is designed for Coexistence use. A detailed description of the implementation of the Coexistence scheme can be found in our [SimpleLink WLAN Access Point Application Note](#).

Note

Please note that the Coexistence feature is not published within our recent SDK. This feature is included with one of our future releases.

5 Security

CC2340 parts implement a range of security features that are provided by hardware. These include: Device Identity, Debug Security, AES- 128-bit hardware acceleration, SHA256 and a Random Number Generator. More information on the hardware provided security features can be found in the [CC2340R5 SimpleLink™ Bluetooth 5.3 Low Energy Wireless MCU](#). The TI-Bluetooth LE-stack, implements further security features to enable security for the Bluetooth communication including:

Network Security

As part of the Bluetooth LE specification there are several security features implemented as specified in the [Bluetooth 5.3 specification](#). These are defined by the Security Manager Layer and define methods to pair and distribute keys and to perform secure connection and data exchange. The security features that are discovery and connection related are performed by the GAP (Generic Access Profile) layer.

The supported security features are:

- Authentication and Authorization
- LE Secure Connection features
 - Pairing (generating and exchanging keys)
 - Encryption (data is encrypted)
 - Association
 - Authentication (Pairing process uses association to support MITM (Man in the Middle protection))
 - Bonding (Bonding pairing process with storing keys for encryption sequence)
 - OOB (Out-of-band pairing)
 - MITM (Provides authentication to prevent attackers)
 - Just Works (no security)
- Privacy (allows devices to generate new address to use over the air)
 - Public Device Address
 - Random Static Address
 - Random Resolvable Private Address
 - Random non-Resolvable Private Address

OAD Security

The TI Bluetooth LE-stack provides methods to authenticate the image that has been transferred on OAD process. Secure OAD uses ECDSA (Elliptic Curve Digital Signature Algorithm) to sign and verify secure OAD images.

For verification of new images, TI offers an implementation of the MCUboot 32-bit secure bootloader.

6 Performance and Test Data

TI stack is pre-certified thus all qualification measurements that are required by the Bluetooth SIG are performed and passed. The set of Test Cases is shown on the [Test Case Reference Lists \(TCRLS\)](#) of the Bluetooth SIG.

TI performs additional measurements which include Power consumptions, Stability and Interoperability.

6.1 Connection

The power consumption measured using different PHYs with a connection interval of 1 second.

Table 6-1. Connection - PHYs

PHY	txPower	Connection Interval	Current Consumption
1M	0 dbm	1 sec	7 μ A
2M	0 dbm	1 sec	6.6 μ A
Coded S2	0 dbm	1 sec	11.5 μ A
Coded S8	0 dbm	1 sec	13.1 μ A

6.2 Advertising

The power consumption measured while advertising with changing advertising intervals and no active connection is shown in the table below.

Table 6-2. Advertising - PHYs

PHY	txPower	Advertise Interval	Payload	Current Consumption
1M	0 dbm	0.1 sec	0 Byte	125 μ A
1M	0 dbm	1 sec	0 Byte	11.6 μ A

Table 6-2. Advertising - PHYs (continued)

PHY	txPower	Advertise Interval	Payload	Current Consumption
1M	-9 dbm	5 sec	17 Byte	2.8 μ A

6.3 Stability Testing

TI Bluetooth LE stack is tested for stability in several configurations as part of our extensive stability testing on device and connected system level. In general, TI Bluetooth LE stack is tested in several test scopes which are listed below.

Advertise Stability (1M Phy) using simple_peripheral and Multirole examples:

- Maintain legacy set for 48 hours.
- Maintain Advertising set with RPA/public address for 48 hours.
- Run Connect and Disconnect operations for 48 hours.
- Run Pairing & Bonding operation for 48 hours.

Scan Stability (1M PHY) using host_test:

- Run Scanning for 48 hours.
- Run Scan operations for different windows size.
- Run Continuous Scan operation for 48 hours.
- Run scan operations while in connection for 48 hours.

Connection Stability (1M, 2M and Coded PHYs) with Peripheral, Central and Multirole devices:

- Maintain 1 Secure Connection (2 for multirole) for 48 hours.
- Run GATT Read and Write connection for 48 hours.
- Run GATT Read and Write operations for 48 hours.
- Run Connection parameter update operations for 48 hours.

BLE Profiles (1M, 2M and Coded PHYs) running the data_stream example:

- Maintain secure connection for 24 hours.
- GATT operations for 24 hours.
- MTU exchange operations for 24 hours.

6.4 Interoperability

All essential TI BLE5-stacks functions are tested for interoperability as Peripheral, Central and Multirole. Interoperability tests have been performed with several devices and OS (list available upon request.)

7 Tools and Development Support

TI offers a range of tools which are provided for free and can be very helpful when working on your wireless design. These tools are developed to help TI engineers as well as customers, to ease development.

7.1 SmartRF Packet Sniffer 2

SmartRF Packet Sniffer 2 includes software and firmware to capture and display over-the-air packets with Wireshark. SmartRF Packet Sniffer 2 supports CC13xx and CC26xx as capture devices.

This document describes the installation and setup of hardware and software. The SmartRF Packet Sniffer 2 package includes the following components:

- SmartRF Sniffer Agent (PC tool) for configuration and communication with the capture device.
- Firmware that enables CC13xx or CC26xx LaunchPads to be used as capture device.
- Firmware source code.
- Dissectors for Wireshark.

7.2 Smart RF Studio 8

SmartRF™ Studio is a Windows application that helps designers of RF systems to easily evaluate the radio at an early stage in the design process for all TI CC1xxx and CC2xxx low-power RF devices. SmartRF™ Studio simplifies generation of the configuration register values and commands, as well as practical testing and debugging of the RF system to generate a PHY. The application also helps to test your wireless system settings in your environment and under different conditions.

7.3 Energy Trace

Energy Trace is a analyzing tool that displays the power profile of your application. The hardware is provided with our [XDS110ET LaunchPad™ development kit debugger with EnergyTrace™ software](#) and is integrated with our free Compiler Code Composer Studio or as a standalone option.

7.4 Code Composer Studio

Code Composer Studio™ IDE is an integrated development environment (IDE) for TI's microcontrollers and processors. This IDE is also available in the cloud by visiting the [TI Developer Zone](#).

Code Composer Studio IDE includes an optimizing C/C++ compiler, source code editor, project build environment, debugger, profiler and many other features. The IDE assists you by offering advanced debug features like ROV (Runtime Object View) or EnergyTrace.

Code Composer Studio combines the advantages of the Eclipse® and Theia frameworks with advanced capabilities from TI resulting in a compelling feature-rich environment. The cloud-based Code Composer Studio enables development in the cloud without the need to download and install large amounts of software.

Download V12.4.0 (the latest release from September 2023) of the [Code Composer Studio™ integrated development environment \(IDE\)](#). The TI-Bluetooth LE stacks release notes point to the currently recommended CCS version.

7.5 SimpleLink Connect App

The SimpleLink connect app is an TI provided example for a Bluetooth LE application that is running on a mobile end device. TI offers the open source app for Android as well as IOS provide easy start up and debug mechanism for applications. The app provides a baseline for users to build their own application, The open-source code is provided on GitHub to fit into development chains. For more information about the app, visit the [SimpleLink Connect mobile app](#) page.

7.6 Uniflash

Uniflash is a TI provided software tool, which can be used to program the on-chip flash memory off TI processors and wireless devices.

7.7 Antenna Reference Designs

TI offers two CC-Antenna Development Kits:

- [CC-ANTENNA-DK2](#)
- [CC-ANTENNA-DK-RD](#)

which can be used for testing and finding the appropriate antenna for any application. The tested antennas with test reports are available in the [CC-Antenna-DK2 and Antenna Measurements Summary Application Report](#). Design files and selection guides are also provided on [CC-ANTENNA-DK-RD](#).

7.8 Design Review Service

SimpleLink hardware design review process gives you access to world class experts that help to review your design and give valuable feedback. View the [Hardware design reviews for SimpleLink™ CC2xxx devices](#) page for information on how to start your Hardware design and request a design review.

7.9 SysConfig

SysConfig is a graphical user interface to configure our SimpleLink devices, and is usable to configure pins, peripherals, radios and subsystems. SysConfig helps to create c-header and code files that can be used as

startup configuration of your custom design. SysConfig also allows you to adapt your system during your development process.

7.10 BTool

BTool is a PC Application capable of sending HCI commands over UART to a USB-connected device. The device is programmed to interpret and execute HCI command. For example, by running the `host_test` example or a PTM-enabled example. BTool can be used for evaluation and Bluetooth certification. More details on the usage of BTool can be found in the [Using BTool Guide](#).

More information on the network processor configuration and usage can be found in the [TI BLE5-Stack User's Guide](#). The Host Test sample application is explained in the [Host Test Guide](#).

More information on the HCI interface, as well as details on the HCI vendor specific commands is available in the [TI API documentation](#).

7.11 GitHub

CC2340 related examples and content the Linux Serial Bootloader is also published on the [TI GitHub](#).

7.12 SimpleLink Academy

[SimpleLink Academy](#) provides an interactive learning experience for the SimpleLink portfolio of devices covering all phases of development from getting started to advanced debug, optimization, and stack development. For each wireless technology and topic, SimpleLink Academy also provides a suggested curriculum of trainings as a guide through the development process. SimpleLink Academy is a quick and easy way to start developing your next connected application.

8 Known Limitations

For the latest information, including Known Issues and Limitations, see the Release Notes for the SDK version you are using.

9 References

- Texas Instruments, [TI BLE5-Stack User's Guide](#).
- Texas Instruments, [SimpleLink Low Power F3 SDK](#).
- Texas Instruments, [SimpleLink Academy for CC23xx](#).
- Texas Instruments, [How to Certify Your Bluetooth Product Application Note](#).
- Bluetooth, [Qualify Your Product](#).
- Texas Instruments, [SimpleLink™ low power software development kits \(SDKs\)](#).
- Texas Instruments, [Bluetooth Low Energy Connections with basic_ble example](#).
- Texas Instruments, [TI Developer Zone](#).
- Bluetooth, [Bluetooth Core Specifications](#).
- Texas Instruments, [SimpleLink WLAN Access Point Application Note](#).
- Texas Instruments, [CC2340R5 SimpleLink™ Bluetooth® 5.3 Low Energy Wireless MCU](#).
- Bluetooth, [Test Case Reference Lists \(TCRLS\)](#).
- Texas Instruments, [XDS110ET LaunchPad™ development kit debugger with EnergyTrace™ software](#).
- Texas Instruments, [Code Composer Studio™ integrated development environment \(IDE\)](#).
- Texas Instruments, [SimpleLink Connect mobile app](#).
- Texas Instruments, [CC-ANTENNA-DK2](#).
- Texas Instruments, [CC-ANTENNA-DK-RD](#).
- Texas Instruments, [CC-Antenna-DK2 and Antenna Measurements Summary Application Report](#).
- Texas Instruments, [Hardware design reviews for SimpleLink™ CC2xxx devices](#).
- Texas Instruments, [Host Test Guide](#).

10 Revision History

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

Changes from October 31, 2023 to November 6, 2023 (from Revision * (October 2023) to Revision A (November 2023))

	Page
• Updated SDK version to 7.20.....	2
• Updated SDK version to 7.20.....	3
• Removed release notes link for SDK 7.20.00.29.....	11
• Removed released notes link for SDK 7.20.00.29.....	11

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](https://www.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 © 2023, Texas Instruments Incorporated