CE220272 - PSoC 6 MCU with BLE Connectivity: Direct Test Mode

Objective

This code example demonstrates Direct Test Mode (DTM) over the Host Controller Interface (HCI) using PSoC® 6 MCU with Bluetooth Low Energy (BLE) Connectivity.

Overview

This example implements the Direct Test Mode (DTM) as per BLE specification v5.0, Vol 6, Part F. The DTM allows testing the physical (PHY) layer of the BLE radio by requesting the device to transmit or receive certain test packets. It is used for RF qualification tests and production line tests without the need of going through the complete BLE protocol stack.

DTM is run over the standard Host Controller Interface (HCI). HCI is the protocol layer that bridges the BLE Host with the Controller.

This code example assumes that you are familiar with the PSoC 6 MCU and the PSoC Creator™ Integrated Design Environment (IDE). If you are new to PSoC 6 MCU, you can find an introduction in AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity.

Requirements

Tool: PSoC Creator 4.2

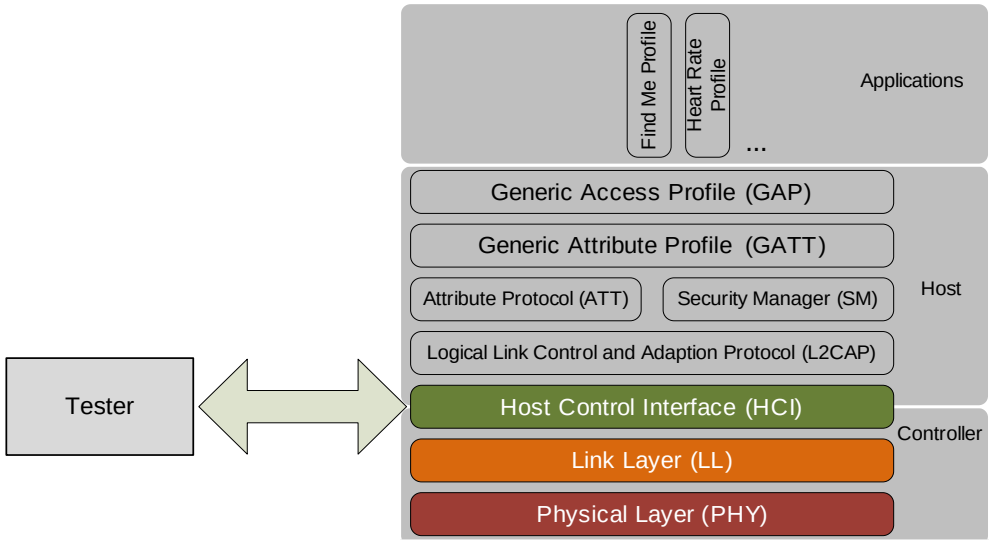
Programming Language: C (Arm® GCC 5.4.1)

Associated Parts: All PSoC 6 MCUs with BLE Connectivity (PSoC 6 BLE)

Related Hardware: CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit

Design

Figure 1. Direct Test Mode (DTM) over Host Controller Interface (HCI)



DTM setup has two parts:

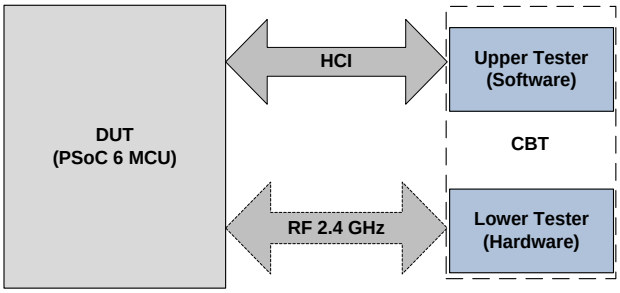
1. Device under Test (DUT): The device whose radio must be tested, such as PSoC 6 MCU on CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit.
2. Tester: The system that performs the test on DUT.

The Tester in DTM can be divided into two parts, as shown in Figure 2:

1. Upper Tester: The part of the tester that communicates commands with the DUT over the HCI interface.
2. Lower Tester: The part of the tester that communicates commands with the DUT over RF.

Most commercially available testers contain both Upper tester and Lower tester, along with software tools to configure tests and read reports.

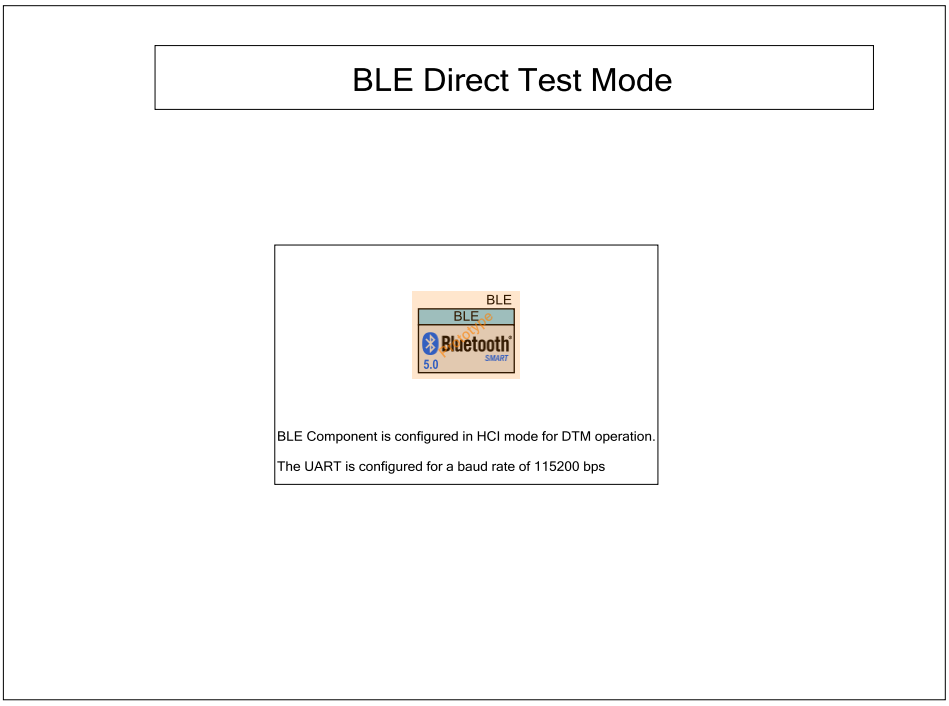
Figure 2. Block Diagram for DTM over HCI



The underlying physical communication for the HCI protocol is UART. Through UART, commands and responses are communicated between PSoC 6 MCU and the tester. When HCI mode is selected in the BLE Component, a UART block in PSoC 6 MCU is reserved for this use. The corresponding UART signals are exposed in the project's design wide resources (*.cydwr*). These signals need to be assigned to appropriate GPIOs on PSoC 6 MCU and connected to UART signals of the external tester.

When the BLE Component is configured for HCI mode, there is no further application code to be written to implement the DTM mode, except enabling global interrupts, starting BLE Component and processing the BLE events. The appropriate response to commands from the tester is handled by the BLE stack as part of the BLE Component.

Figure 3. PSoC Creator Schematic



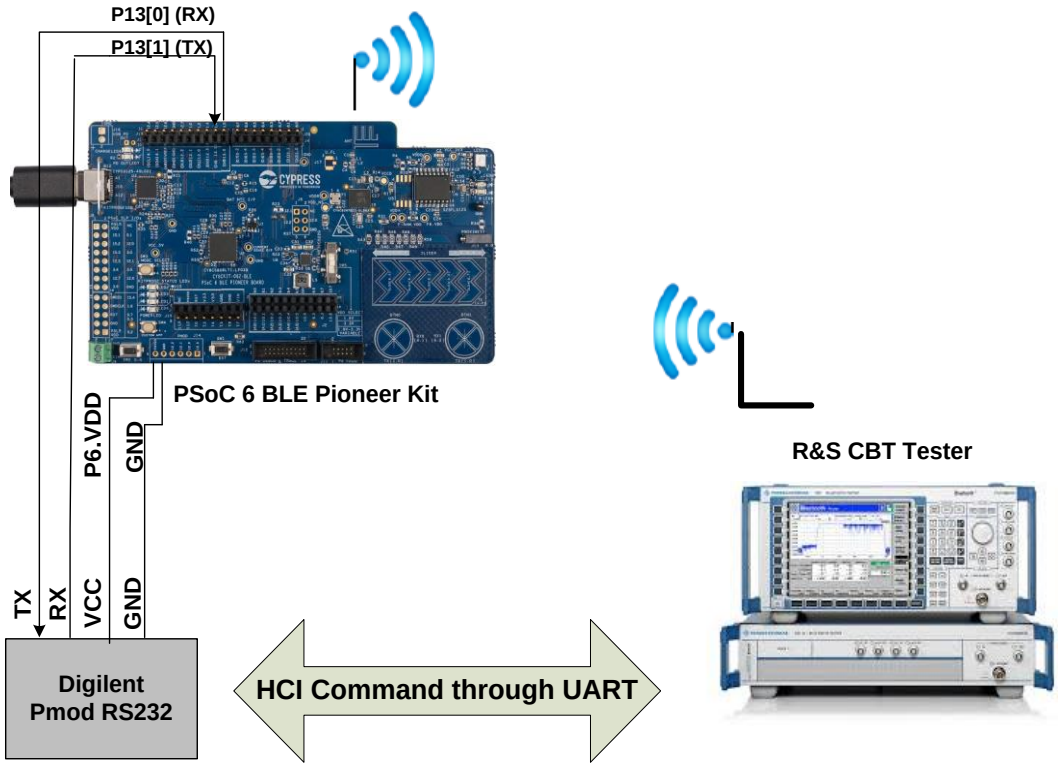
The project consists of the following file:

main_cm0p.c enables global interrupts, starts the BLE Component, and then processes BLE events in the main loop. All other processing is handled by the BLE Component.

Hardware Setup

The Upper Tester communicates with the DUT over RS232. To use the PSoC 6 BLE Pioneer Kit with these testers, a RS232 voltage translator is required. One such module is Digilents's Pmod RS232. The UART signals from the PSoC 6 MCU on PSoC 6 BLE Pioneer Kit are connected to the pins on this module and the UART port of the module is used to connect to the Upper Tester. The setup is shown in Figure 4.

Figure 4. Hardware Setup



Software Setup

No additional software setup is required for this code example. Follow the DTM Tester software instructions to run the test.

Operation

1. Connect the Pioneer Baseboard to your PC using the provided USB cable through the USB connector (**J10**).
2. Program the Pioneer Baseboard with the *CE220272_BLE_Direct_Test_Mode* project. See the CY8CKIT-062-BLE kit guide for details on how to program firmware into the device.
3. Connect the pins corresponding to the PSoC 6 MCU's UART signals to an RS232 translator as shown in Figure 4.
4. Connect the CBT tester to the PSoC 6 BLE Pioneer Kit via the RS232 translator using an RS232 cable.
5. If you are using the PSoC 6 MCU with the U.FL connector as the antenna, then connect this to the U.FL connector of the CBT tester using a cable.

If you are using the PSoC 6 MCU with a PCB antenna (present on the PSoC 6 BLE Pioneer Kit), then connect an antenna to the SMA connector of the CBT tester.

6. In the CBT tester, send the communication commands to start and run the test.
7. At the end of the test, the RF performance test results are displayed on the CBT tester.

Components

Table 1 lists the PSoC Creator Components used in this example, as well as the placement / hardware resources used by each.

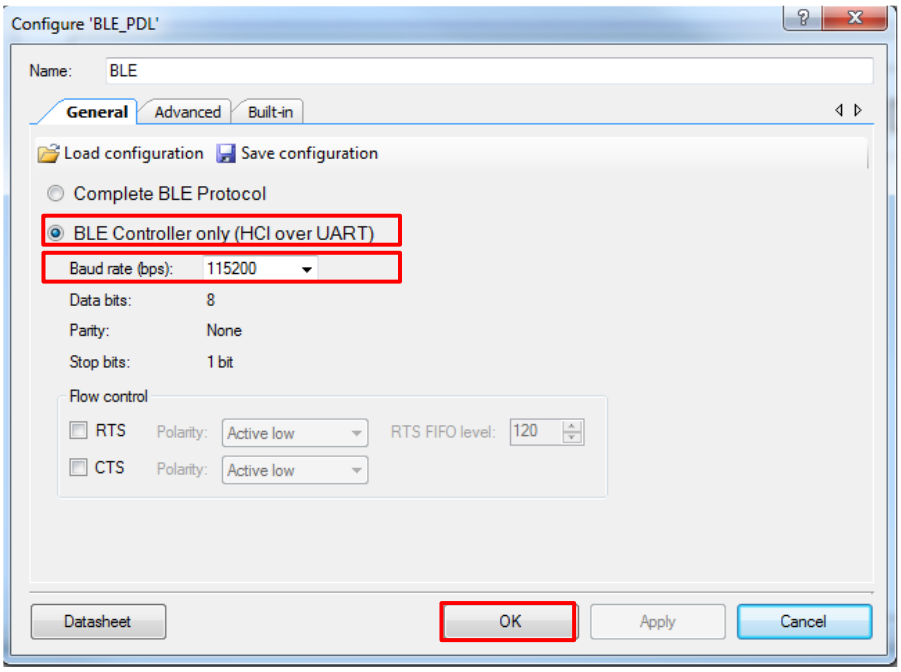
Table 1. List of PSoC Creator Components

Component	Version	Placement / Hardware Resources
Bluetooth Low Energy (BLE)	V2.0	Bluetooth Low Energy Subsystem

Parameter Settings

PSoC 6 MCU with BLE Connectivity supports DTM using the HCI interface. HCI is enabled from the **General** tab in the BLE Component. Note that as soon as HCI mode is selected, the profiles related tabs become hidden because GAP and GATT layer are not used in HCI mode.

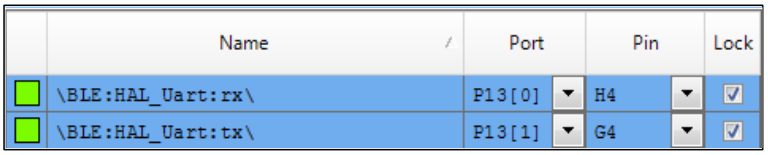
Figure 5. HCI Mode in BLE Component



Design-Wide Resources

Under **Pins** tab of **Design Wide Resources** (*.cydwr*) assign the pins as shown in Figure 6.

Figure 6. PSoC 6 MCU UART Pin Assignment



Related Documents

Application Notes	
AN210781 Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes the PSoC 6 MCU with BLE Connectivity and how to build your first PSoC Creator project.
AN91445 Antenna Design guide	Provides guidelines on how to design an antenna for BLE applications.
PSoC Creator Component Datasheets	
Bluetooth Low Energy	Facilitates designing applications requiring BLE connectivity
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual
Development Kit (DVK) Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	

Document History

Document Title: CE220272 - PSoC 6 MCU with BLE Connectivity: Direct Test Mode

Document Number: 002-20272

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5815771	VKVK	08/23/2017	New Code Example
*A	6001013	VKVK	12/21/2017	Updated template and minor text changes. Updated project to PSoC Creator 4.2 Beta
*B	6074768	VKVK	02/19/2018	Updated the project with PSoC Creator 4.2

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2017-2018. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.