

Objective

This code example demonstrates the implementation of a simple BLE Immediate Alert Service (IAS)-based Find Me Profile (FMP) using PSoC® 6 MCU with Bluetooth Low Energy (BLE) Connectivity, using ModusToolbox™ integrated development environment (IDE).

Requirements

Tool: [ModusToolbox](#) IDE 1.0

Programming Language: C

Associated Parts: All [PSoC 6 MCU](#) dual-CPU parts with BLE connectivity

Related Hardware: [PSoC 6 BLE Pioneer Kit](#)

Overview

This design implements a BLE [FMP](#) that consists of an [IAS](#). FMP and IAS are BLE standard Profile and Service respectively, as defined by the [Bluetooth SIG](#). The design uses the RGB LED on the [CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit](#). The blue LED displays the alert level (OFF, flashing, or ON for no alert, mild alert, or high alert respectively). Green and red LEDs indicate whether the Peripheral device (the Pioneer kit) is advertising or disconnected.

The USB-BLE dongle provided with the CY8CKIT-062-BLE Pioneer kit or an iOS/Android mobile device can act as the BLE Central device, which locates the Peripheral device.

Hardware Setup

This example uses the kit's default configuration. Refer to the kit guide to ensure the kit is configured correctly.

Note: The PSoC 6 BLE Pioneer kit ships with KitProg2. ModusToolbox only works with KitProg3. Before using this code example, make sure that the kit is upgraded to KitProg3. See ModusToolbox Help > ModusToolbox IDE Documentation > User Guide; section PSoC 6 MCU KitProg Firmware Loader. If you do not upgrade, you will see an error like “unable to find CMSIS-DAP device” or “KitProg firmware is out of date”.

Software Setup

This code example consists of two parts: a locator and a target. For the locator, download and install either the [CySmart Host Emulation Tool](#) PC application or the CySmart app for [iOS](#) or [Android](#). You can test behavior with any of the two options, but the CySmart app is simpler.

Scan the following QR codes from your mobile phone to download the CySmart app.

iOS



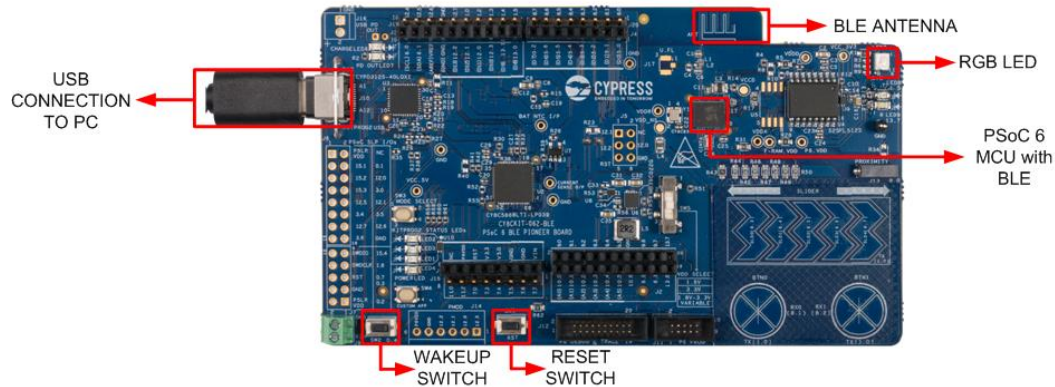
Android



Operation

1. Connect the kit to your PC using the provided USB cable.

Figure 1. CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit Baseboard



2. Import the code example into a new workspace. If you are not familiar with this process, see [KBA225201](#).
3. Program the PSoC 6 MCU device. In the project explorer, select the **mainapp** project. In the Quick Panel, scroll to the **Launches** section and click the **Program (KitProg3)** configuration.
4. Observe the green LED turn ON after the device starts advertisement.
5. To test using the CySmart mobile app:
 - a. Turn ON Bluetooth on your Android or iOS device.
 - b. Launch the CySmart app.
 - c. Press the reset switch on the Pioneer Kit to start BLE advertisements from your design. The advertisement LED (green LED) turns ON to indicate that BLE advertisement has started.
 - d. Pull down the CySmart app home screen to start scanning for BLE Peripherals; your device appears in the CySmart app home screen. Select your device to establish a BLE connection. Once the connection is established, the green LED turns OFF.
 - e. Select the 'Find Me' Profile from the carousel view.
 - f. Select an Alert Level value on the Find Me **Profile** screen. Observe the state of the blue LED on the device change based on the alert level.

Figure 2 and Figure 3 show the steps for using CySmart App on iOS and Android respectively.

Figure 2. Testing with the CySmart App on iOS

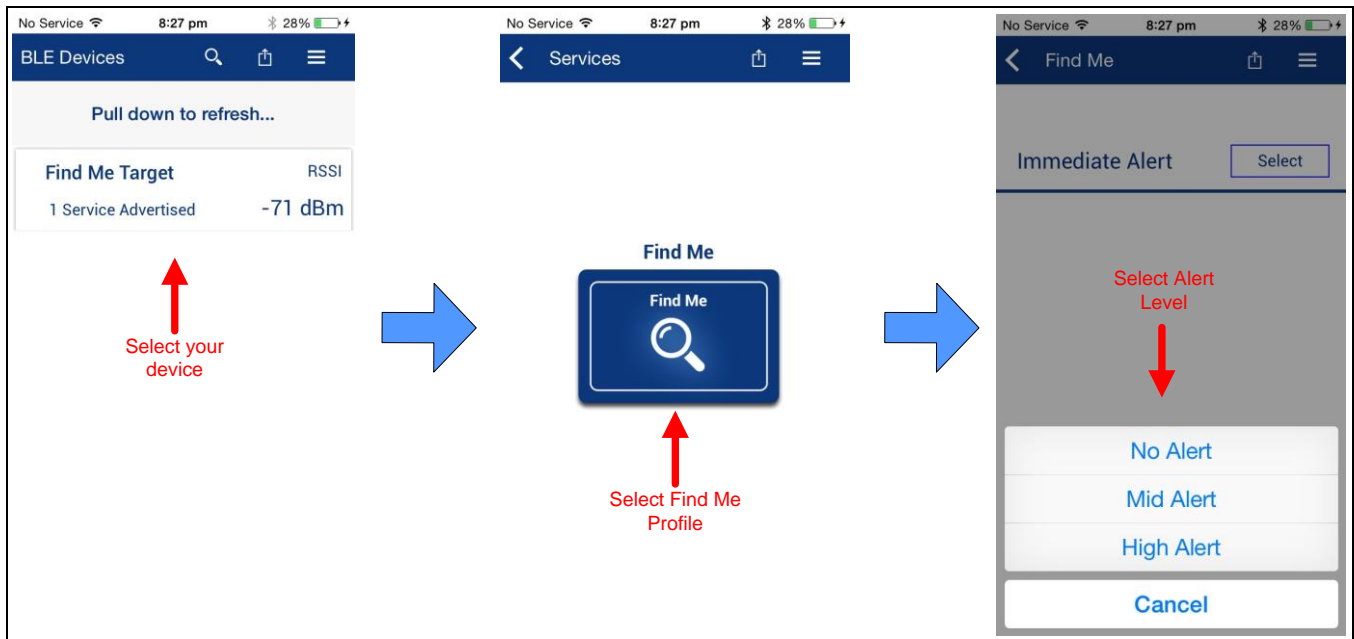
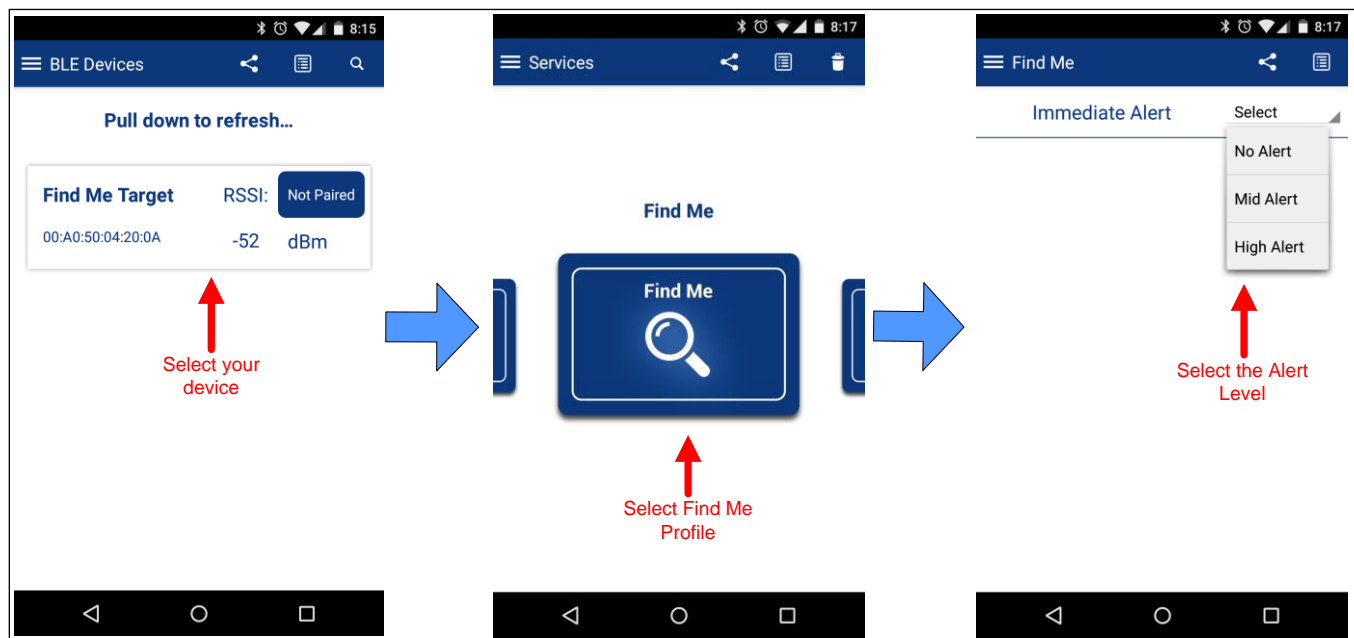


Figure 3. Testing with the CySmart App on Android



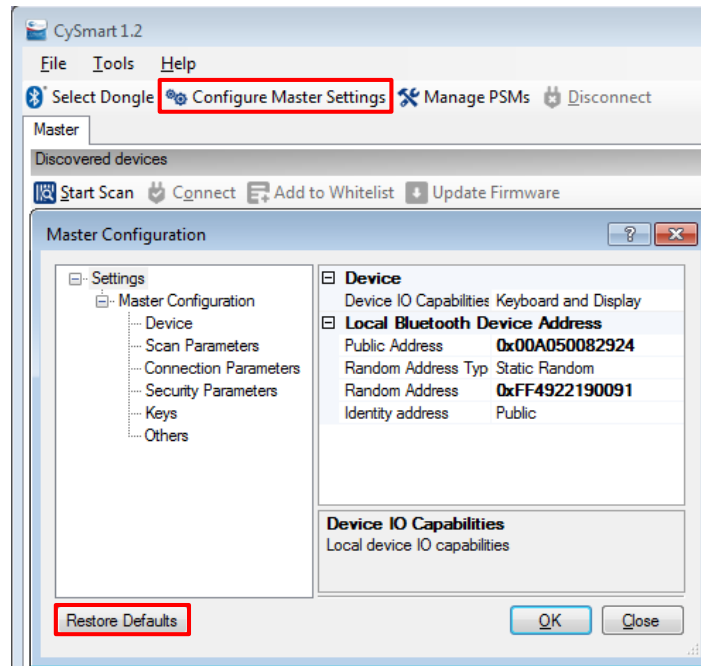
6. To test using the CySmart Host Emulation Tool:

- Connect the BLE Dongle to your Windows PC. Wait for the driver installation to complete.
- Launch the CySmart Host Emulation Tool.

Note: If the dongle firmware is outdated, you will be alerted. You must upgrade the firmware before you can complete this step. Follow the instructions in the window to update the dongle firmware.

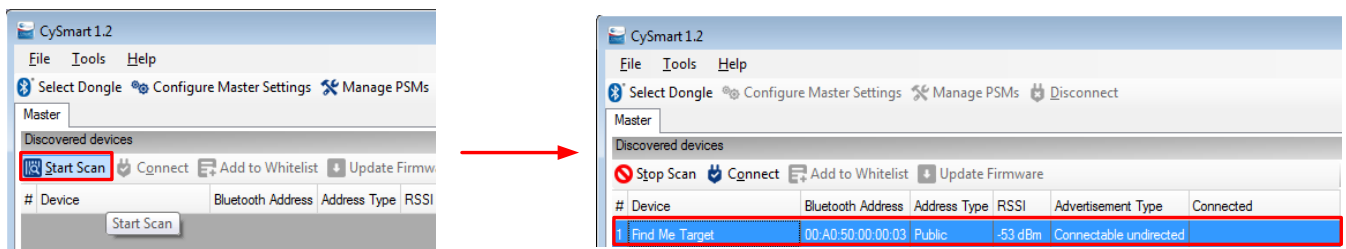
- Select **Configure Master Settings** and then, click **Restore Defaults**, as shown in Figure 4. Then, click **OK**.

Figure 4. CySmart Master Settings Configuration



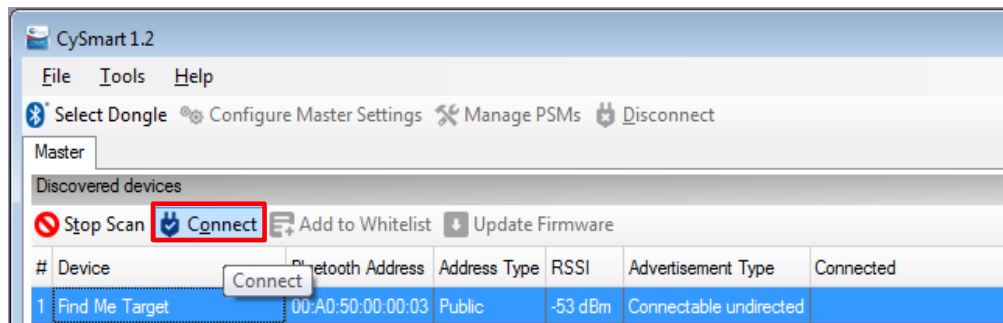
- Press the reset switch on the Pioneer Kit to start BLE advertisements from your design.
- On the CySmart Host Emulation Tool, click **Start Scan**. Your device name (configured as Find Me Target) should appear in the Discovered devices list, as shown in Figure 5.

Figure 5. CySmart Device Discovery



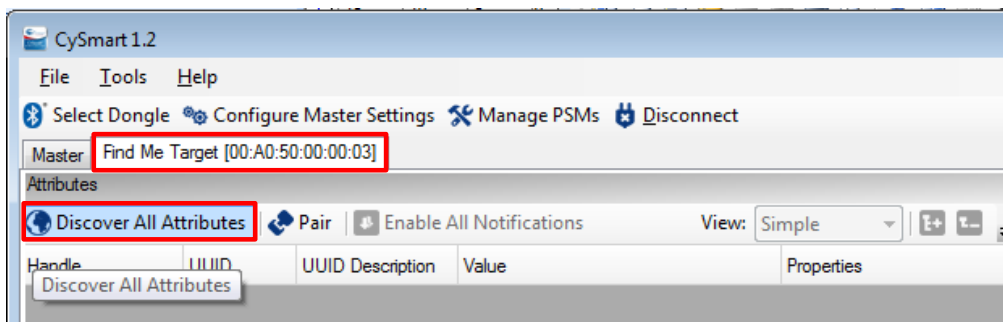
- Select your device and click **Connect** to establish a BLE connection between the CySmart Host Emulation Tool and your device, as shown in Figure 6.

Figure 6. CySmart Device Connection



- g. Once connected, switch to the **Find Me Target** device tab and discover all Attributes on your design from the CySmart Host Emulation Tool, as shown in Figure 7.

Figure 7. CySmart Attribute Discovery



- h. Scroll down the **Attributes** window and locate the **Immediate Alert** Service fields. Write a value of 0 – no alert, 1 – mild alert, or 2 – high alert to the **Alert Level** Characteristic under the **Immediate Alert Service**, as Figure 8 shows. Observe that the state of the LED on your device changes per your Alert Level Characteristic configuration.

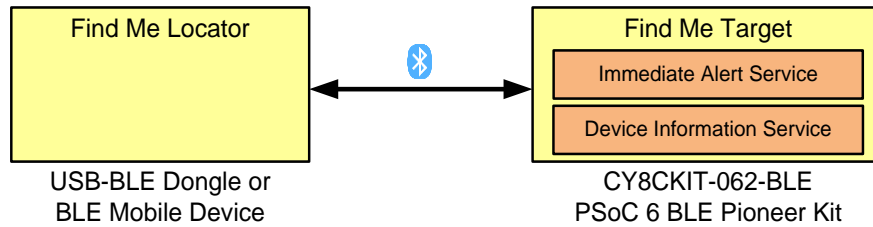
Debugging

You can debug the example to step through the code. Use a **Program+Debug** configuration. If you are unfamiliar with how to start a debug session with ModusToolbox IDE, see [KBA224621](#).

Design and Implementation

The 'Find Me Locator' (the BLE Central device) is a BLE GATT Client. The 'Find Me Target' (the Peripheral device) is a BLE GATT Server with the IAS and an additional Device Information Service implemented, as [Figure 10](#) shows.

Figure 10. Find Me Service Relationship



The BLE Find Me profile defines what happens when the locating Central device broadcasts a change in the alert level.

The Find Me locator performs service discovery using the 'GATT Discover All Primary Services' procedure. The BLE Service Characteristic discovery is done by the 'Discover All Characteristics of a Service' procedure. When the Find Me Locator wants to cause an alert on the Find Me Target, it writes an alert level in the Alert Level Characteristic of the IAS. When the Find Me Target receives an alert level, it indicates the level using the blue LED: OFF for no alert, blinking for mild alert, and ON for high alert.

The BLE interface is implemented on a PSoC 6 MCU with BLE Connectivity device using the BLE resource. The application runs primarily on the Arm® Cortex®-M4 core. The Cortex-M0+ core executes the BLE controller firmware and is responsible for maintaining the BLE connection.

See [AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy \(BLE\) Connectivity](#) to understand the design of firmware for this code example.

The device enters low-power Deep Sleep mode when BLE is idle. It wakes up automatically when there is activity on the BLE connection.

When BLE is disconnected, the device enters Hibernate mode. It wakes up when the reset switch or wakeup switch (SW2) is pressed and performs a complete reset sequence in firmware.

Resources and Settings

[Table 1](#) lists some of the ModusToolbox resources used in the example, and how they are used in the design. The *design.modus* file contains all configuration settings. For example, for pin usage and configuration, open the **Pins** tab of the design file.

Table 1. ModusToolbox Resources

Resource	Alias	Purpose	Non-default Settings
Bluetooth Low Energy (BLE)	BLE	Implement BLE communication	See Figure 11 through Figure 17
Serial Communication Block (SCB) 5	KIT_UART	Provide a serial interface for verbose messaging	See Figure 18
Multi-Counter Watchdog Timer (MCWDT) 0	MCWDT	Generate a tick every 250 msec	See Figure 19
Digital Output Pin	KIT_RGB_R	Provide visual feedback	See Figure 20
	KIT_RGB_G		
	KIT_RGB_B		

Resource	Alias	Purpose	Non-default Settings
	KIT_UART_TX	Used for UART transmit (Tx)	See Figure 21
Digital Input Pin	KIT_UART_RX	Used for UART receive (Rx)	See Figure 22
	KIT_BTN1	Wake up device from hibernate	See Figure 23

Note: [Figure 11](#) through [Figure 23](#) highlight the non-default settings for each resource in this example. For the clock resource settings, go to the **Platform** tab of the *design.modus* file.

Figure 11. BLE: Protocol Configuration



Figure 12. BLE: Adding Find Me Profile

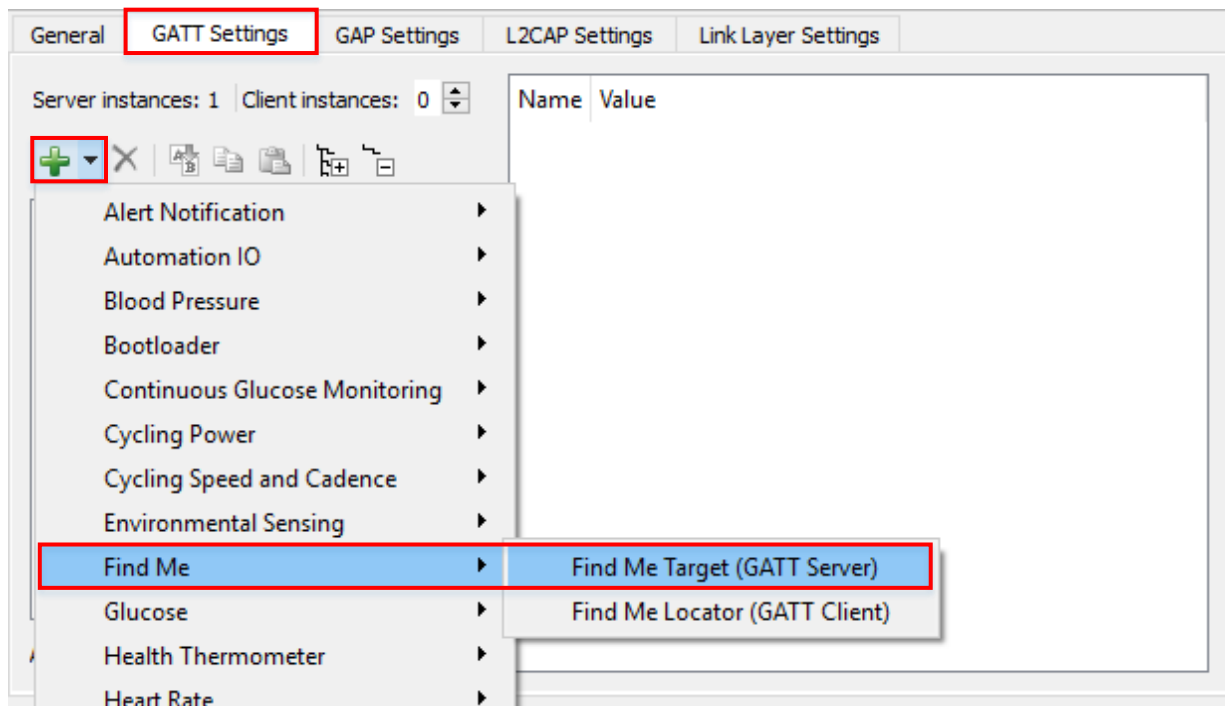


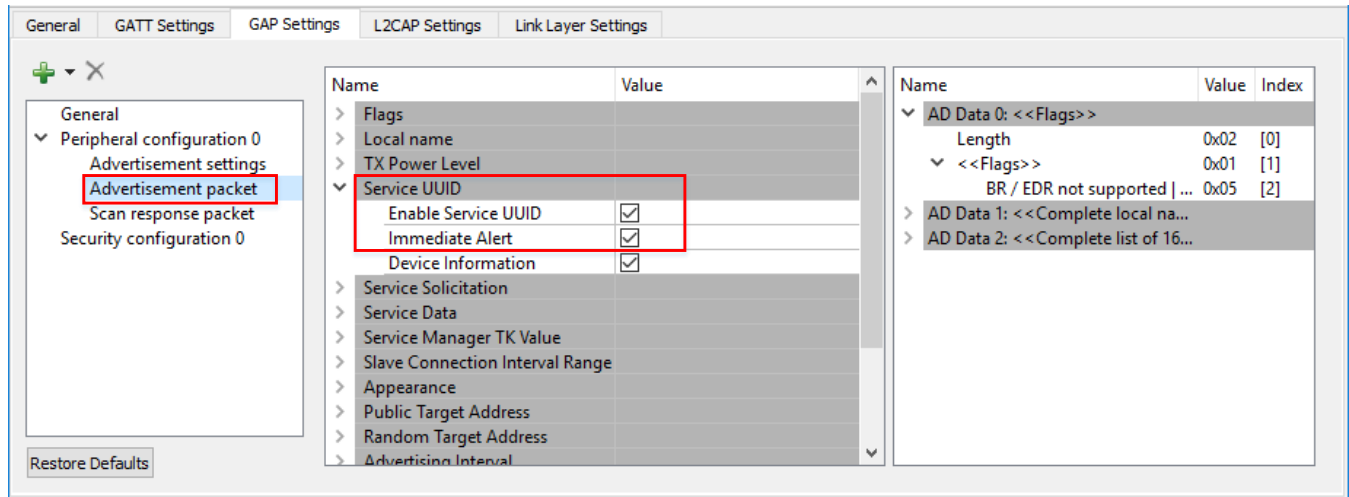
Figure 13. BLE: Device Configuration

General GATT Settings GAP Settings L2CAP Settings Link Layer Settings																									
<div> <div>+</div> <div>-</div> <div>×</div> </div> <div> <div>General</div> <div> <div>Peripheral configuration 0</div> <div>Advertisement settings</div> <div>Advertisement packet</div> <div>Scan response packet</div> <div>Security configuration 0</div> </div> </div> <div>Restore Defaults</div>	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Device Address</td> <td></td> </tr> <tr> <td>Device address</td> <td>00A050-000000</td> </tr> <tr> <td>Silicon generated 'Company assigned' part of address</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>General</td> <td></td> </tr> <tr> <td>Device name</td> <td>Find Me Target</td> </tr> <tr> <td>Appearance</td> <td>Generic Keyring</td> </tr> <tr> <td>TX Power Level</td> <td></td> </tr> <tr> <td>Connection TX power level (dBm)</td> <td>0</td> </tr> <tr> <td>Adv/Scan TX power level (dBm)</td> <td>0</td> </tr> <tr> <td>Other</td> <td></td> </tr> <tr> <td>Bond list size</td> <td>16</td> </tr> </tbody> </table>	Name	Value	Device Address		Device address	00A050-000000	Silicon generated 'Company assigned' part of address	<input checked="" type="checkbox"/>	General		Device name	Find Me Target	Appearance	Generic Keyring	TX Power Level		Connection TX power level (dBm)	0	Adv/Scan TX power level (dBm)	0	Other		Bond list size	16
Name	Value																								
Device Address																									
Device address	00A050-000000																								
Silicon generated 'Company assigned' part of address	<input checked="" type="checkbox"/>																								
General																									
Device name	Find Me Target																								
Appearance	Generic Keyring																								
TX Power Level																									
Connection TX power level (dBm)	0																								
Adv/Scan TX power level (dBm)	0																								
Other																									
Bond list size	16																								

Figure 14. BLE: Advertisement Settings

General GATT Settings GAP Settings L2CAP Settings Link Layer Settings																							
<div> <div>+</div> <div>-</div> <div>×</div> </div> <div> <div>General</div> <div> <div>Peripheral configuration 0</div> <div>Advertisement settings</div> <div>Advertisement packet</div> <div>Scan response packet</div> <div>Security configuration 0</div> </div> </div> <div>Restore Defaults</div>	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>AdvertisementProperties</td> <td></td> </tr> <tr> <td>Discovery mode</td> <td>Limited</td> </tr> <tr> <td>Advertising type</td> <td>Connectable undirected advertising</td> </tr> <tr> <td>Filter policy</td> <td>Scan request: Any Connect request: Any</td> </tr> <tr> <td>Advertising channel map</td> <td>All channels</td> </tr> <tr> <td>AdvertisementProperties</td> <td></td> </tr> <tr> <td>Fast advertising interval minimum</td> <td>20</td> </tr> <tr> <td>Fast advertising interval maximum</td> <td>30</td> </tr> <tr> <td>Fast advertising timeout</td> <td>30</td> </tr> <tr> <td>Enable slow advertising interval</td> <td><input type="checkbox"/></td> </tr> </tbody> </table>	Name	Value	AdvertisementProperties		Discovery mode	Limited	Advertising type	Connectable undirected advertising	Filter policy	Scan request: Any Connect request: Any	Advertising channel map	All channels	AdvertisementProperties		Fast advertising interval minimum	20	Fast advertising interval maximum	30	Fast advertising timeout	30	Enable slow advertising interval	<input type="checkbox"/>
Name	Value																						
AdvertisementProperties																							
Discovery mode	Limited																						
Advertising type	Connectable undirected advertising																						
Filter policy	Scan request: Any Connect request: Any																						
Advertising channel map	All channels																						
AdvertisementProperties																							
Fast advertising interval minimum	20																						
Fast advertising interval maximum	30																						
Fast advertising timeout	30																						
Enable slow advertising interval	<input type="checkbox"/>																						

Figure 15. BLE: Advertisement Packet Settings



General | GATT Settings | GAP Settings | L2CAP Settings | Link Layer Settings

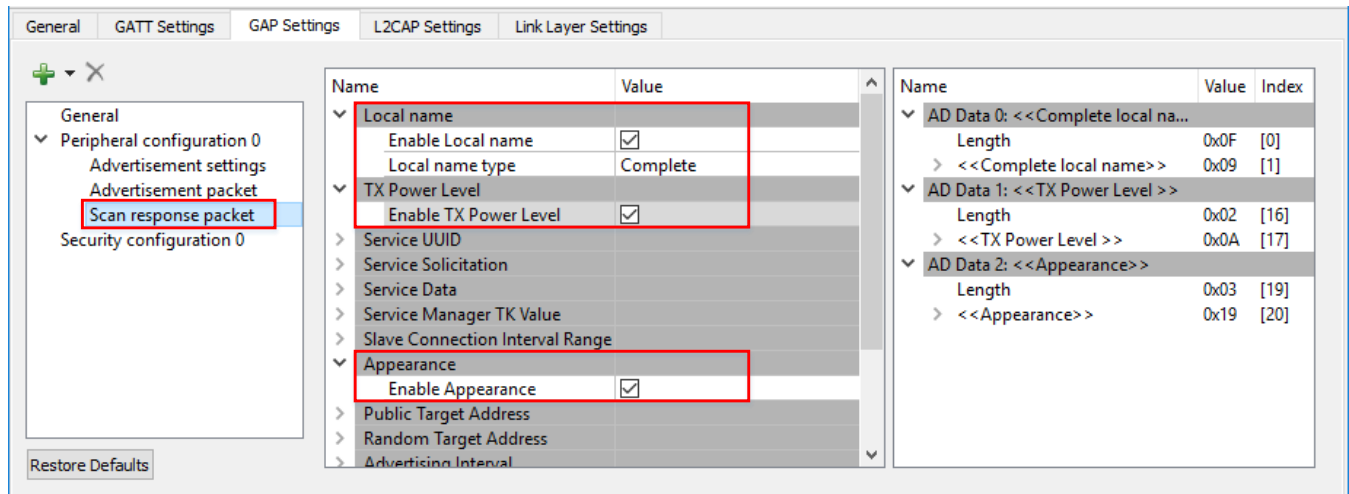
General
 Peripheral configuration 0
 Advertisement settings
Advertisement packet
 Scan response packet
 Security configuration 0

Restore Defaults

Name	Value
> Flags	
> Local name	
> TX Power Level	
> Service UUID	
Enable Service UUID	<input checked="" type="checkbox"/>
Immediate Alert	<input checked="" type="checkbox"/>
Device Information	<input checked="" type="checkbox"/>
> Service Solicitation	
> Service Data	
> Service Manager TK Value	
> Slave Connection Interval Range	
> Appearance	
> Public Target Address	
> Random Target Address	
> Advertising Interval	

Name	Value	Index
> AD Data 0: <<Flags>>		
Length	0x02	[0]
<<Flags>>	0x01	[1]
BR / EDR not supported ...	0x05	[2]
> AD Data 1: <<Complete local na...		
> AD Data 2: <<Complete list of 16...		

Figure 16. BLE: Response Packet Settings



General | GATT Settings | GAP Settings | L2CAP Settings | Link Layer Settings

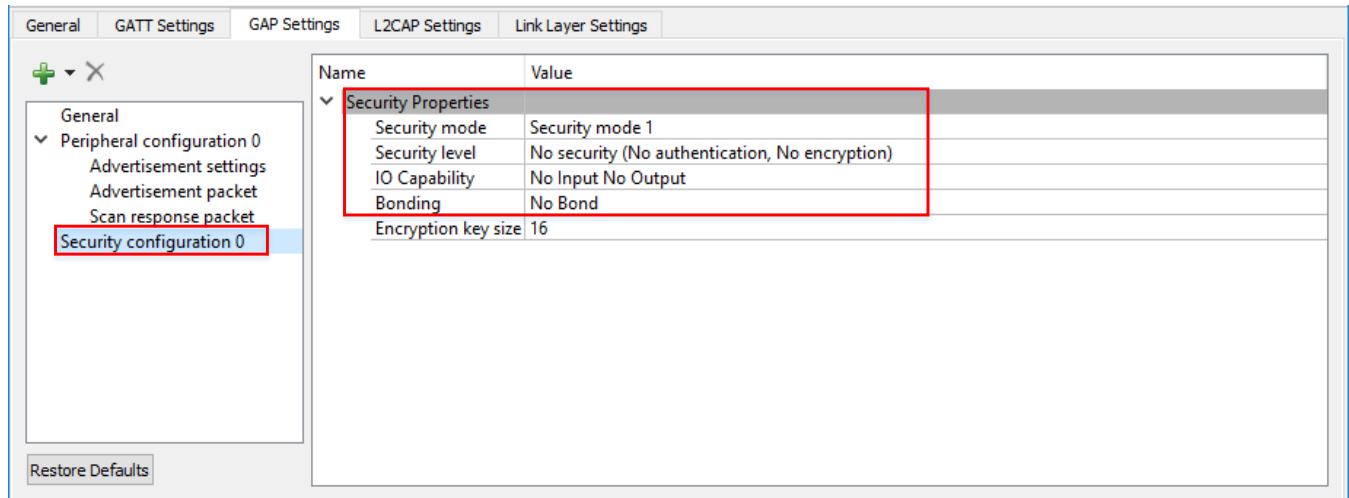
General
 Peripheral configuration 0
 Advertisement settings
 Advertisement packet
Scan response packet
 Security configuration 0

Restore Defaults

Name	Value
> Local name	
Enable Local name	<input checked="" type="checkbox"/>
Local name type	Complete
> TX Power Level	
Enable TX Power Level	<input checked="" type="checkbox"/>
> Service UUID	
> Service Solicitation	
> Service Data	
> Service Manager TK Value	
> Slave Connection Interval Range	
> Appearance	
Enable Appearance	<input checked="" type="checkbox"/>
> Public Target Address	
> Random Target Address	
> Advertising Interval	

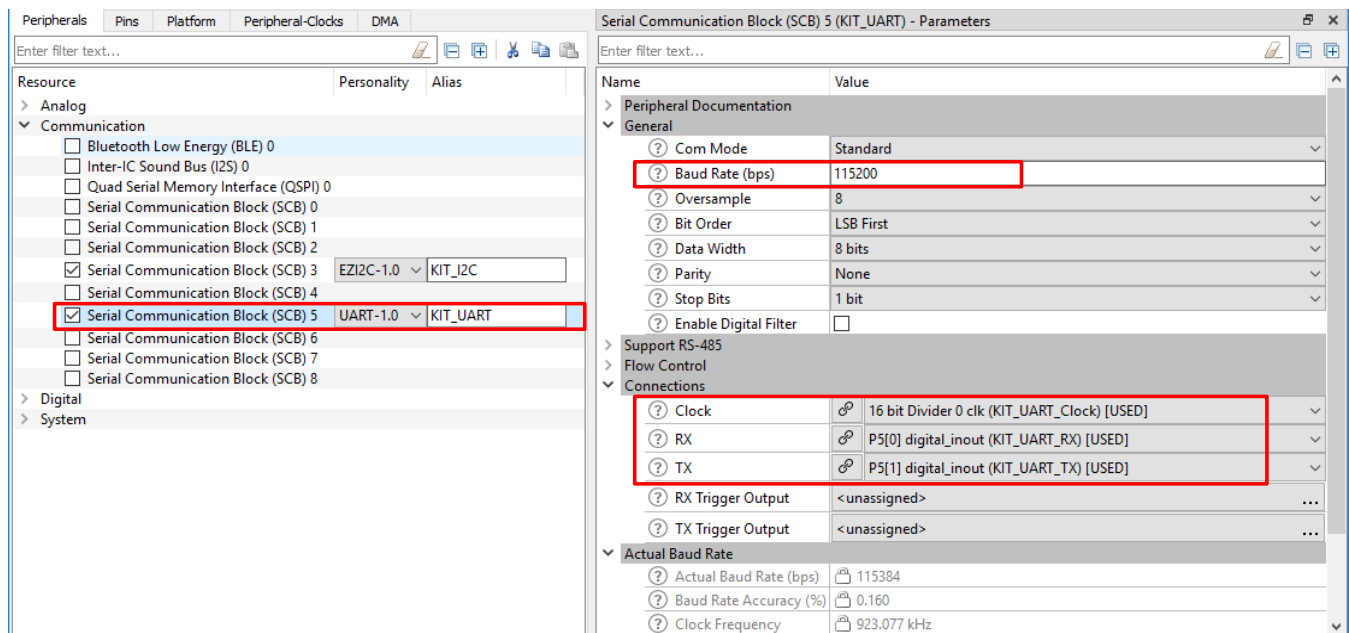
Name	Value	Index
> AD Data 0: <<Complete local na...		
Length	0x0F	[0]
<<Complete local name>>	0x09	[1]
> AD Data 1: <<TX Power Level >>		
Length	0x02	[16]
<<TX Power Level >>	0x0A	[17]
> AD Data 2: <<Appearance>>		
Length	0x03	[19]
<<Appearance>>	0x19	[20]

Figure 17. BLE: Security Configuration



Name	Value
Security Properties	
Security mode	Security mode 1
Security level	No security (No authentication, No encryption)
IO Capability	No Input No Output
Bonding	No Bond
Encryption key size	16

Figure 18. UART Configuration



Name	Value
Peripheral Documentation	
General	
Com Mode	Standard
Baud Rate (bps)	115200
Oversample	8
Bit Order	LSB First
Data Width	8 bits
Parity	None
Stop Bits	1 bit
Enable Digital Filter	<input type="checkbox"/>
Support RS-485	
Flow Control	
Connections	
Clock	16 bit Divider 0 clk (KIT_UART_Clock) [USED]
RX	P5[0] digital_inout (KIT_UART_RX) [USED]
TX	P5[1] digital_inout (KIT_UART_TX) [USED]
RX Trigger Output	<unassigned>
TX Trigger Output	<unassigned>
Actual Baud Rate	
Actual Baud Rate (bps)	115384
Baud Rate Accuracy (%)	0.160
Clock Frequency	923.077 kHz

Figure 19. MCWDT Settings

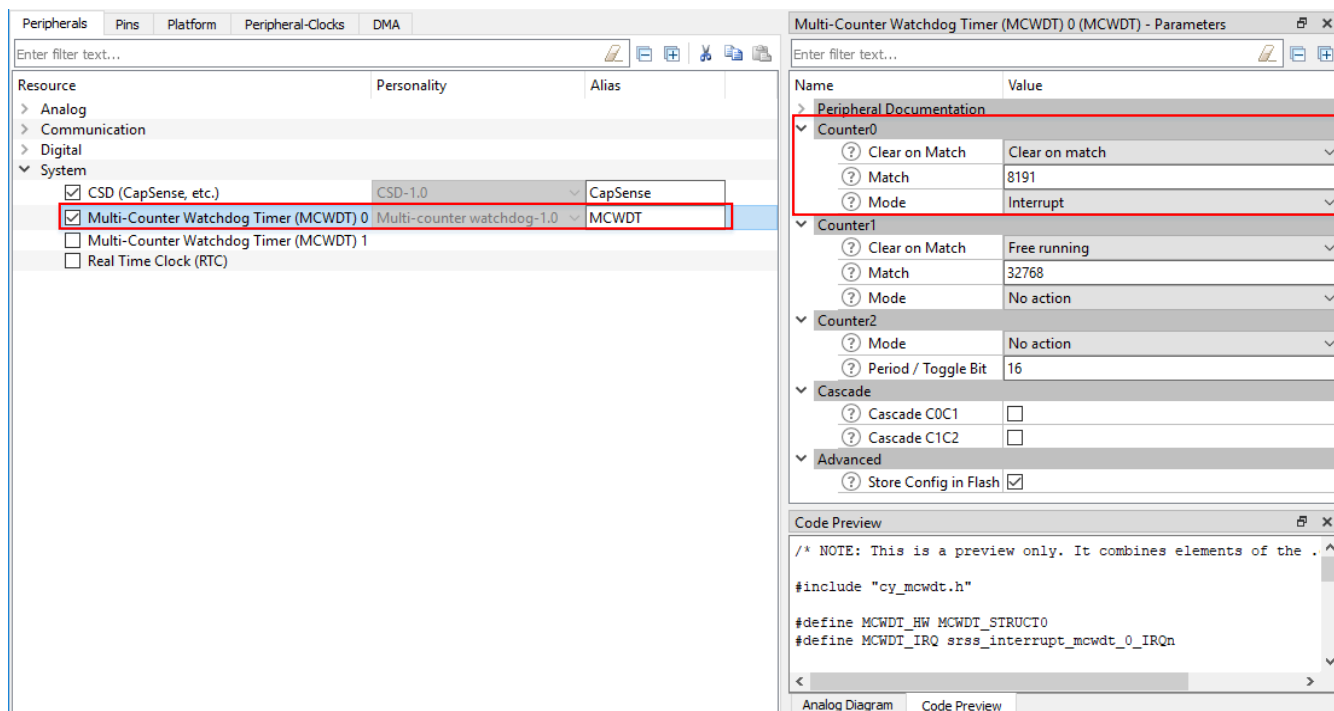


Figure 20. GPIO Pin Configuration for RGB LED

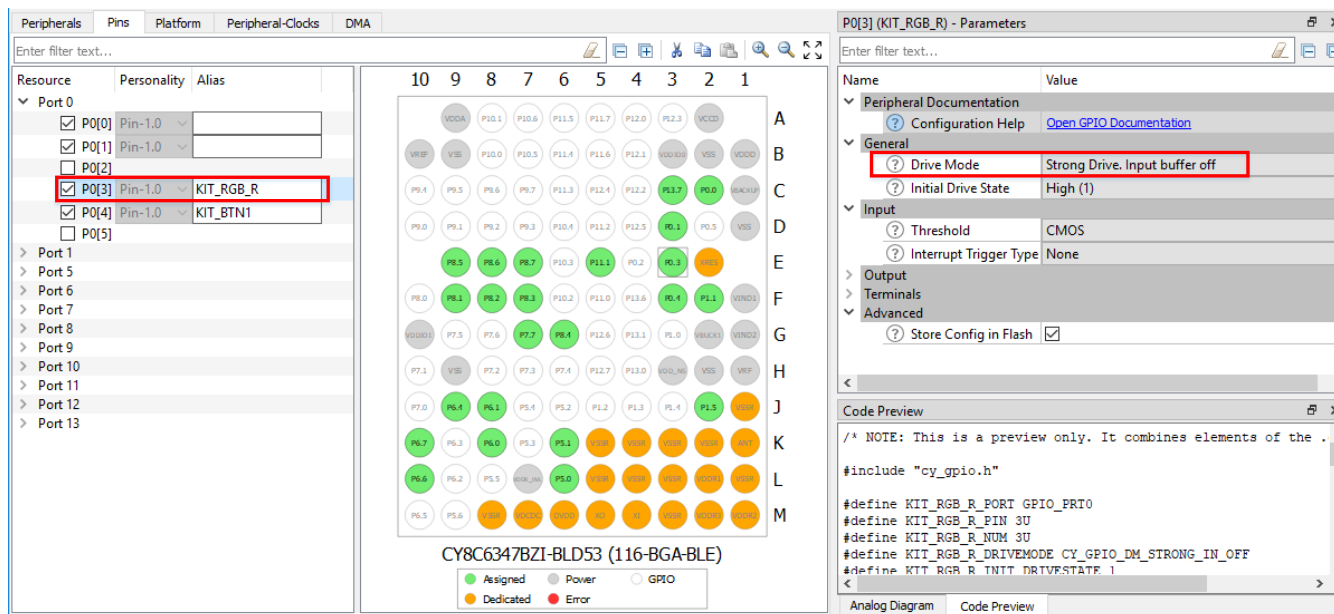
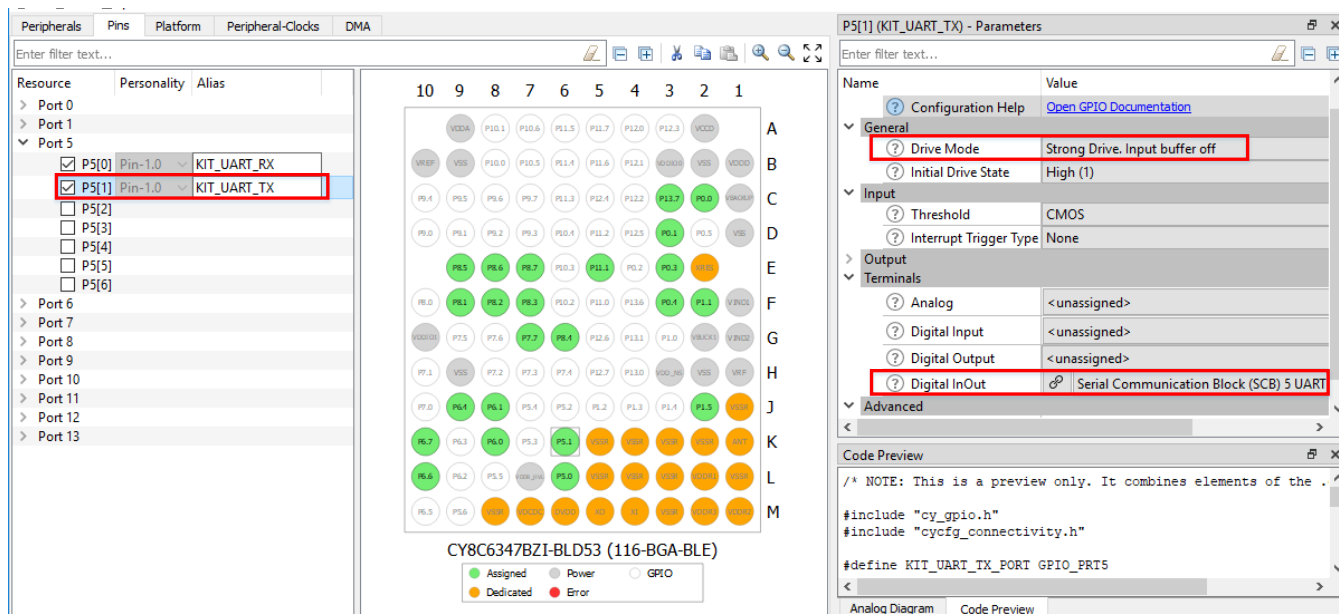


Figure 21. GPIO Pin Configuration for UART Tx



Resource

- Port 0
- Port 1
- Port 5
 - ☒ P5[0] Pin-1.0 KIT_UART_RX
 - ☒ P5[1] Pin-1.0 KIT_UART_TX
 - ☐ P5[2]
 - ☐ P5[3]
 - ☐ P5[4]
 - ☐ P5[5]
 - ☐ P5[6]
- Port 6
- Port 7
- Port 8
- Port 9
- Port 10
- Port 11
- Port 12
- Port 13

Pin Configuration Diagram

CY8C6347BZI-BLD53 (116-BGA-BLE)

Legend: ● Assigned ● Power ● Dedicated ● Error

P5[1] (KIT_UART_TX) - Parameters

Name	Value
Configuration Help	Open GPIO Documentation
General	
Drive Mode	Strong Drive, Input buffer off
Initial Drive State	High (1)
Input	
Threshold	CMOS
Interrupt Trigger Type	None
Output	
Terminals	
Analog	<unassigned>
Digital Input	<unassigned>
Digital Output	<unassigned>
Digital InOut	Serial Communication Block (SCB) 5 UART
Advanced	

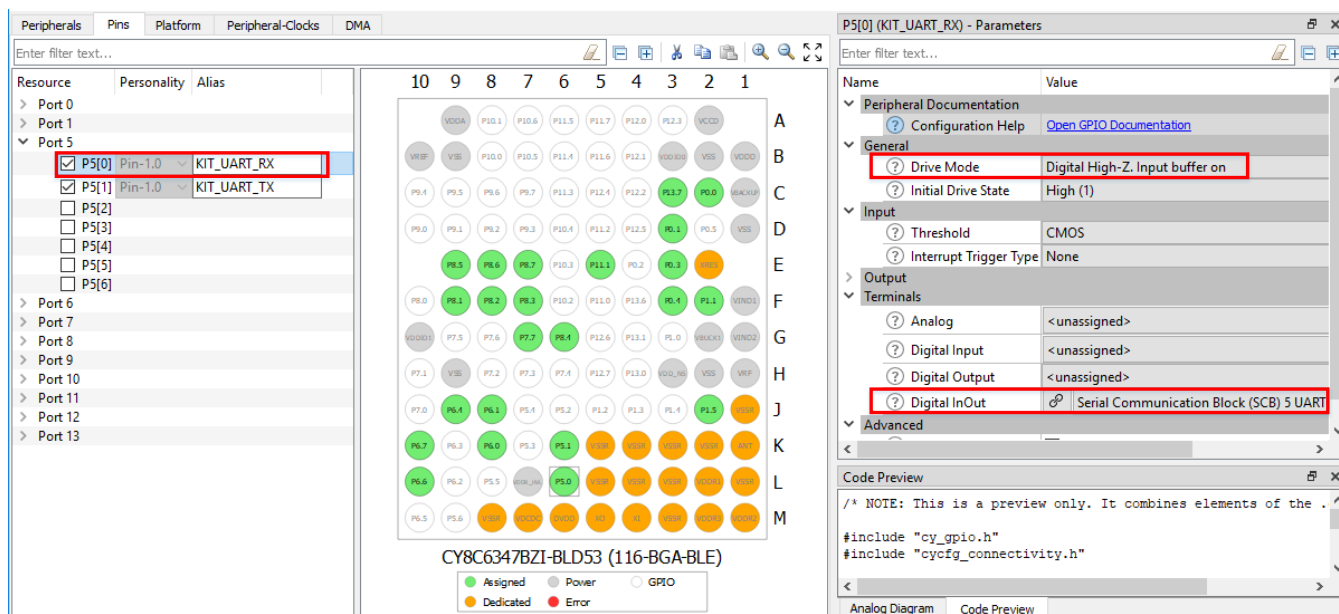
Code Preview

```

/* NOTE: This is a preview only. It combines elements of the .
#include "cy_gpio.h"
#include "cycfg_connectivity.h"

#define KIT_UART_TX_PORT GPIO_PRI5
  
```

Figure 22. GPIO Pin Configuration for UART Rx



Resource

- Port 0
- Port 1
- Port 5
 - ☒ P5[0] Pin-1.0 KIT_UART_RX
 - ☒ P5[1] Pin-1.0 KIT_UART_TX
 - ☐ P5[2]
 - ☐ P5[3]
 - ☐ P5[4]
 - ☐ P5[5]
 - ☐ P5[6]
- Port 6
- Port 7
- Port 8
- Port 9
- Port 10
- Port 11
- Port 12
- Port 13

Pin Configuration Diagram

CY8C6347BZI-BLD53 (116-BGA-BLE)

Legend: ● Assigned ● Power ● Dedicated ● Error

P5[0] (KIT_UART_RX) - Parameters

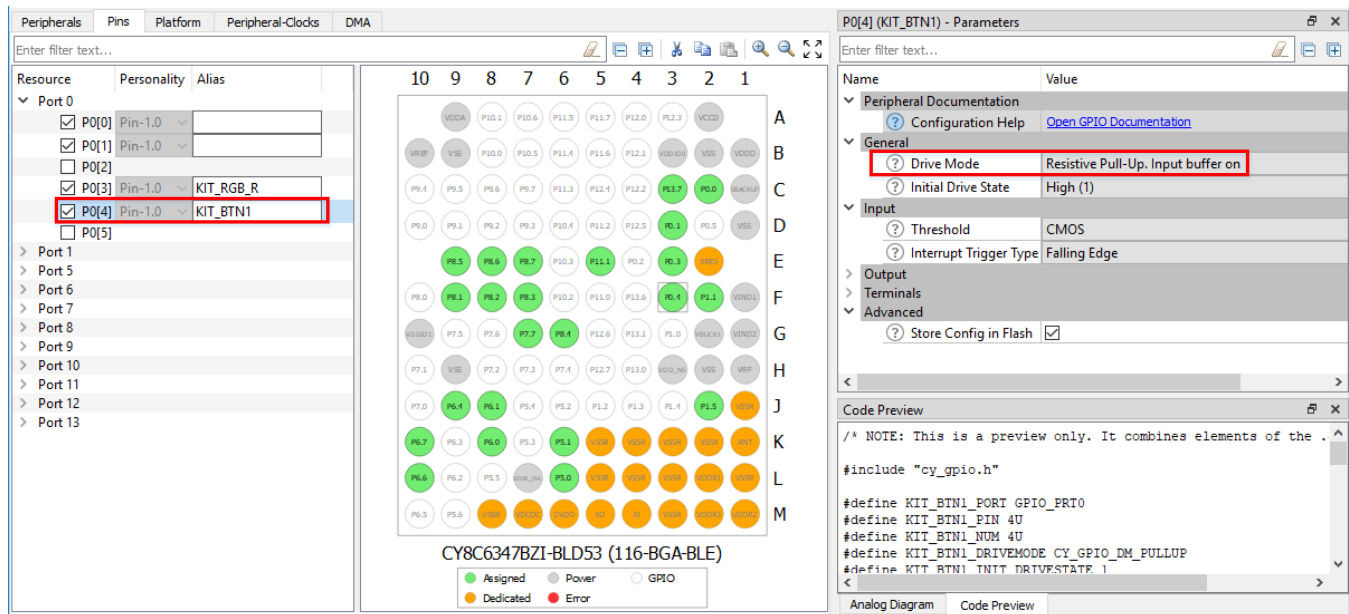
Name	Value
Peripheral Documentation	Open GPIO Documentation
General	
Drive Mode	Digital High-Z, Input buffer on
Initial Drive State	High (1)
Input	
Threshold	CMOS
Interrupt Trigger Type	None
Output	
Terminals	
Analog	<unassigned>
Digital Input	<unassigned>
Digital Output	<unassigned>
Digital InOut	Serial Communication Block (SCB) 5 UART
Advanced	

Code Preview

```

/* NOTE: This is a preview only. It combines elements of the .
#include "cy_gpio.h"
#include "cycfg_connectivity.h"
  
```

Figure 23. GPIO Pin Configuration for User Button



Related Documents

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes PSoC 6 MCU with BLE Connectivity devices and how to build your first PSoC Creator project
AN215656 – PSoC 6 MCU: Dual-CPU System Design	Describes the dual-CPU architecture in PSoC 6 MCU, and shows how to build a simple dual-CPU design
Code Examples	
Visit the Cypress GitHub site for a comprehensive collection of code examples using ModusToolbox IDE	
Device Documentation	
PSoC 6 MCU: PSoC 63 with BLE Datasheet	PSoC 6 MCU: PSoC 63 with BLE Architecture Technical Reference Manual
Development Kits	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	
CY8CKIT-062-WiFi-BT PSoC 6 WiFi-BT Pioneer Kit	
CY8CPROTO-062-4343W PSoC 6 Wi-Fi BT Prototyping Kit	
Tool Documentation	
ModusToolbox IDE	The Cypress IDE for IoT designers

Cypress Resources

Cypress provides a wealth of data at www.cypress.com to help you to select the right device, and quickly and effectively integrate the device into your design.

For the PSoC 6 MCU devices, see [KBA223067](#) in the Cypress community for a comprehensive list of PSoC 6 MCU resources.

Document History

Document Title: CE212736 – PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity - Find Me

Document Number: 002-25467

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	6366433	SNVN	11/13/2018	New code example
*A	6390655	SNVN	11/21/2018	Added a note about KitProg3 in Hardware Setup

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/pmhc
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

[Community](#) | [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, 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.