**Please note that Cypress is an Infineon Technologies Company.**

The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

**Continuity of document content**

The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

**Continuity of ordering part numbers**

Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

# CE217613 - Bluetooth Low Energy (BLE) Automation IO

## Objective

This BLE example project demonstrates how to use the BLE Component's Automation IO profile feature and related APIs.

## Overview

This example project configures the CY8CKIT-042-BLE PSoC 4 Pioneer Kit as an Automation Input Output Server (AOIS) with two instances of a Digital characteristic, two instances of an Analog characteristic, and an Aggregate characteristic:

- The value of the CapSense® Linear Slider position is used as the input parameter for Instance 0 of the Analog characteristic. Instance 1 of the Analog characteristic is used for setting the Current Digital-to-Analog Converter (IDAC) output current.
- The value of the **SW2** button is used as the input parameter for Instance 0 of the Digital characteristic and for indication of the Analog characteristic.
- Instance 1 of the Digital characteristic is used for the blue LED control on the CY8CKIT-042-BLE PSoC 4 Pioneer Kit

In this example project, security connection (Mode 1, Level 4 option) is enabled with the passkey-based authenticated man-in-the-middle (MITM) attack prevention and automatic fallback to the legacy authenticated MITM mode if security connection is not supported by the peer device or selected BLE device family.

This example supports all the GATT sub-procedures defined in the AIOS specification.

## Requirements

**Tool:** PSoC Creator 4.0 or later

**Programming Language:** C (GCC 4.9 or later)

**Associated Parts:** PSoC 4 BLE parts

**Related Hardware:** CY8CKIT-042-BLE PSoC 4 Pioneer Kit with the CY8CKIT-143A PSoC® 4 BLE 256-KB Module and CY5677 CySmart BLE 4.2 USB Dongle that supports Security Connection

## Design

This example project consists of the following components:

- Bluetooth Low Energy (BLE)
- Current Digital-to-Analog Converter (IDAC)
- Capacitive Sensing (CapSense)
- Universal Asynchronous Receiver-Transmitter (UART)
- LEDs
- SW2

The schematic is shown in Figure 1.

This project demonstrates the functionality of the BLE Component configured as the AIO Server. It is designed to work with the CySmart PC application.

After a startup, the device initializes the BLE Component. For proper operation, the Component requires several callback functions to receive events from the BLE Stack. The AppCallBack() function is used to receive general BLE events. Another callback (AiosCallBack()) is used to receive events specific to the service-attribute operations.

The CYBLE_EVT_STACK_ON event indicates the successful initialization of the BLE Stack. After this event is received, the Component starts fast advertising with the packet structure as configured in the BLE Component Customizer (Figure 7).

The **SW2** button on CY8CKIT-042 BLE is used to do the following:

- Accept the password displayed on a Windows terminal application such as HyperTerminal or PuTTy (This can also be done by pressing **y** on HyperTerminal. Optionally, the example project can use legacy Security Mode 1 Level 3 (Authenticated pairing with encryption.)
- Change the value of the Digital characteristic
- Present an indication of the Aggregate characteristic
- Exit the low-power mode

The IDAC is used for the output value of Analog characteristic Instance 1.

The CapSense Linear Slider is used as the input value of Analog characteristic Instance 0.

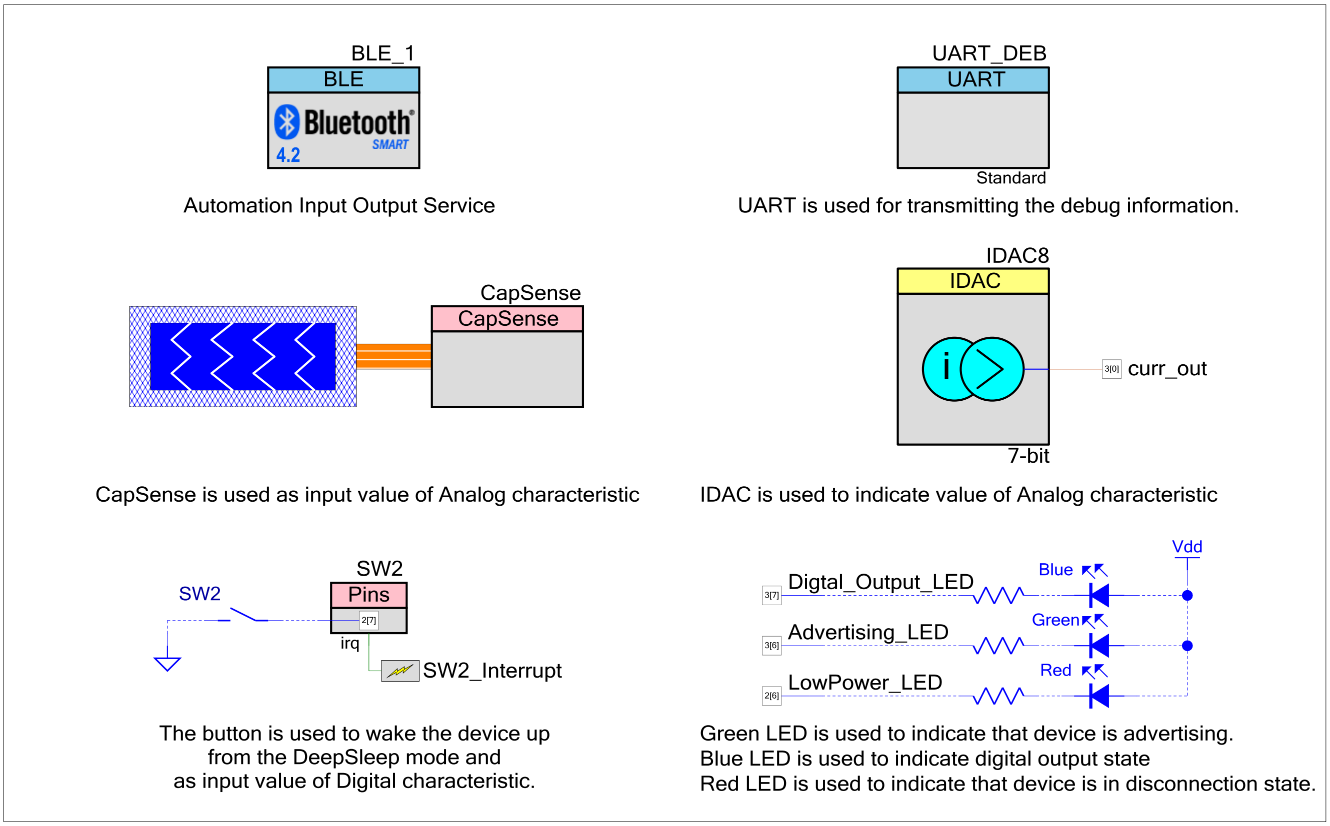
The green LED on CY8CKIT-042 BLE indicates that BLE is in advertisement mode.

The red LED indicates that the Automation IO Server is disconnected or is in the low-power mode.

The blue LED is used to indicate the state of Digital characteristic Instance 1.

The UART is used to print debug information and scan commands from a terminal.

Figure 1. BLE Automation Input Output Server Example Project Schematic



## Design Considerations

This code example is designed for the PSoC 4 BLE family and associated with the CY8CKIT-042-BLE PSoC 4 Pioneer Kit. The design is easily portable to other PSoC BLE devices and kits, typically by just changing the device and Components' pin assignments.

# Hardware Setup

1. Connect the BLE Pioneer Kit to the computer's USB port, as Figure 2 shows.

Figure 2. Connect USB Cable to J13

2. Connect the BLE Dongle to one of the USB ports on the computer.

Figure 3. Connect BLE Dongle to USB Port

# Software Setup

## Using UART for Debugging

A HyperTerminal program is required in a PC to receive the debug information. If you do not have a HyperTerminal program installed, download and install any serial port communication program. Freeware such as HyperTerminal, Bray's Terminal, or Putty are available on the web:

1. Connect the PC and kit with a USB cable.
2. Open the device manager program in your PC, find the COM port to which the kit is connected, and note the port number.
3. Open the HyperTerminal program and select the COM port to which the kit is connected.
4. Configure the baud rate, parity, stop bits, and flow control information in the HyperTerminal configuration window. The default settings are: baud rate – 115200, parity – none, stop bits – 1 and flow control – XON/XOFF. These settings have to match the configuration of the PSoC Creator UART Component in the project.
5. Start communicating with the device as explained in the project description.

# Components

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

Table 1. List of PSoC Creator Components

| Component | Hardware Resources |
|---|---|
| BLE | BLE Subsystem |
| UART | GPIO rx – P1[4], tx – P1[5] |
| IDAC | GPIO P3[0] |
| CapSense | GPIO P4[0] – Cmod<br>P2[1] - P2[5] – Linear Slider |
| Advertising_LED | GPIO P3[6] |
| Digtal_Output_LED | GPIO P3[7] |
| LowPower_LED | GPIO P2[6] |
| SW2 | GPIO P2[7] |

## Parameter Settings

### BLE Component

The BLE Component is configured as the Automation IO Profile in the Automation IO Server (GATT Server) Profile role with the settings shown in the figures below.

Figure 4. GATT Settings

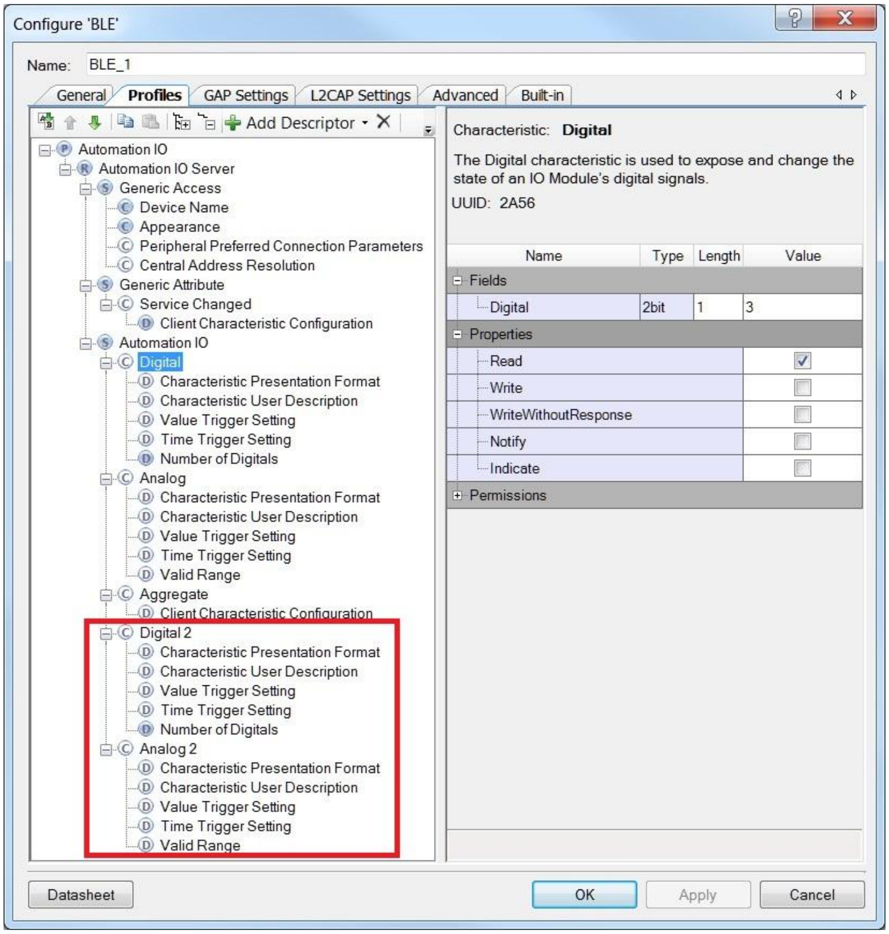



Figure 5. GAP Settings

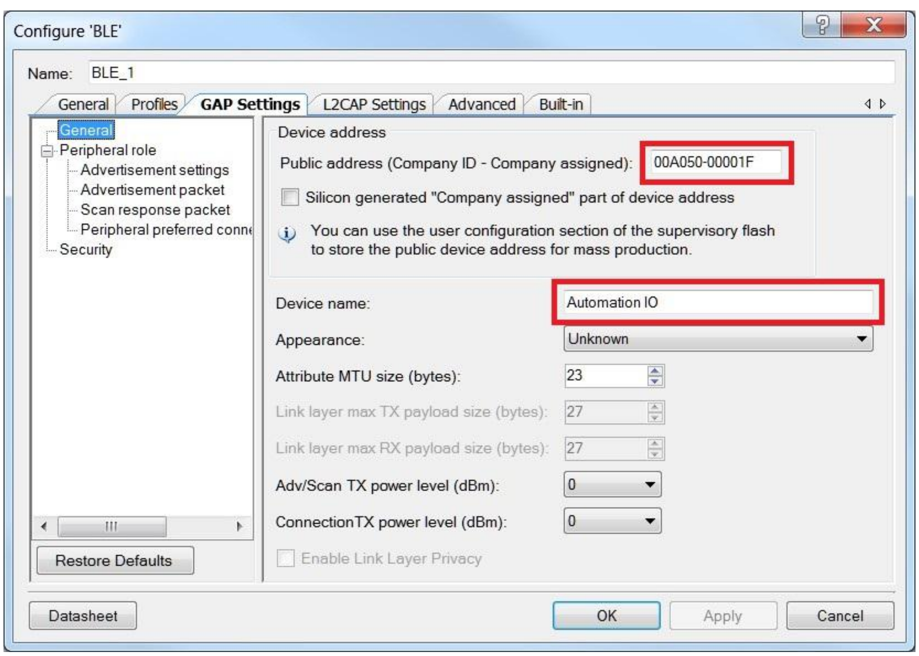


Figure 6. GAP Settings: Advertisement Settings

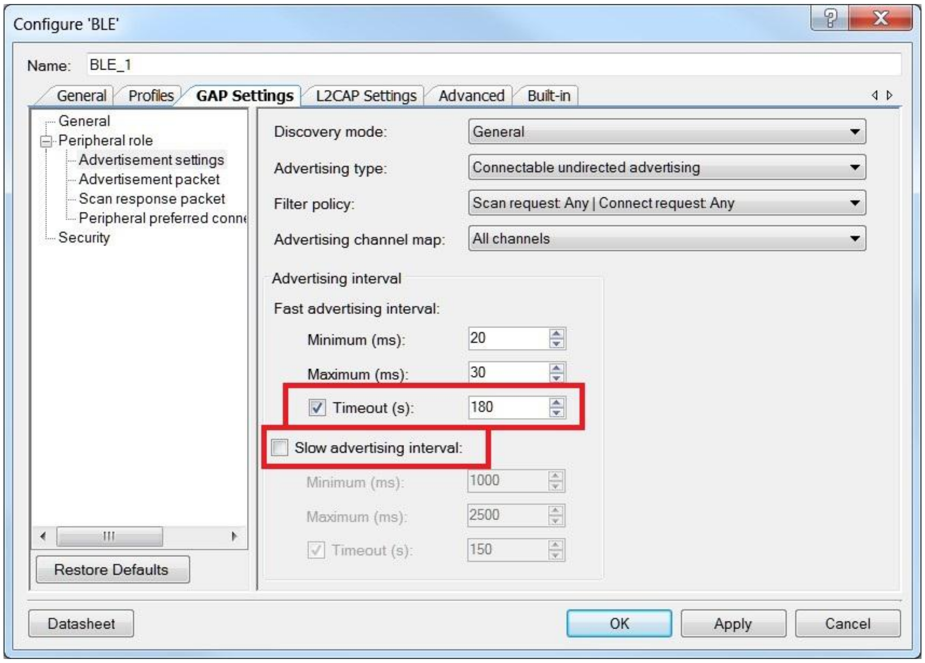



Figure 7. GAP Settings: Advertisement Packet

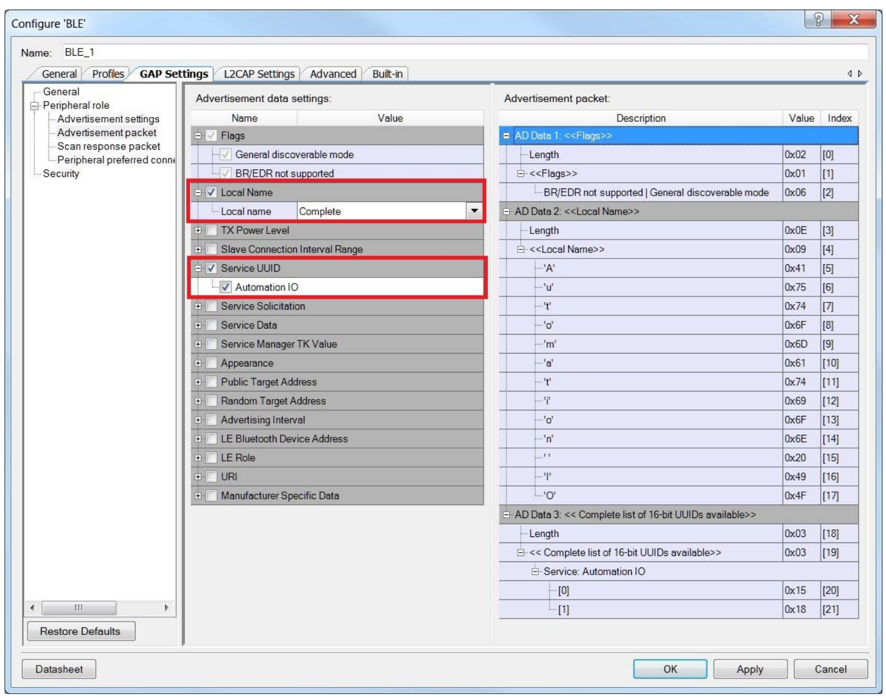
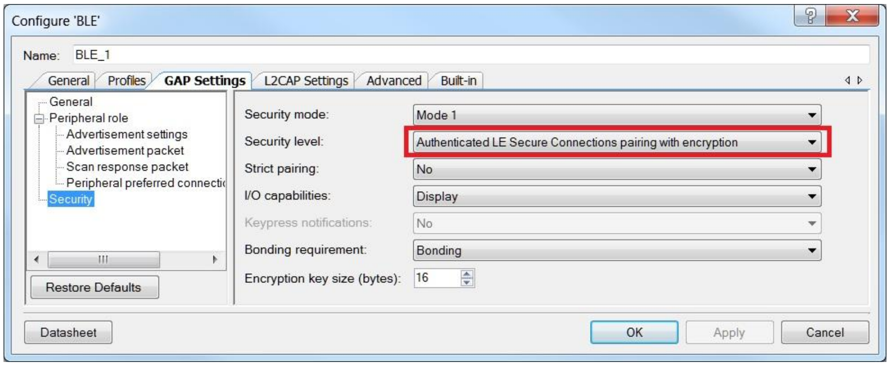


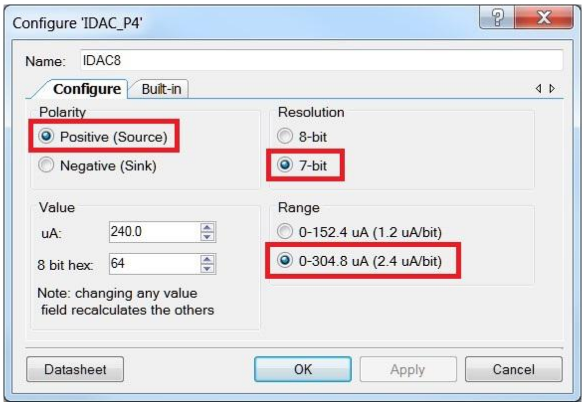
Figure 8. Security Settings




### IDAC

Figure 9 shows the settings for the IDAC Component. See the IDAC Component datasheet for additional information.

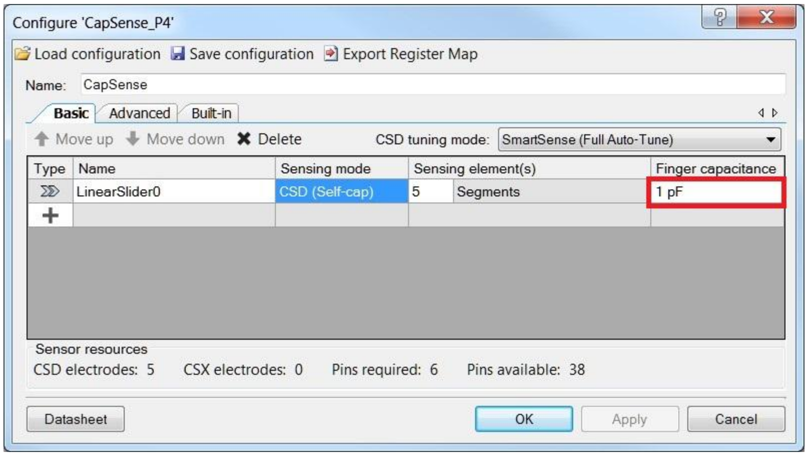
Figure 9. IDAC Component Parameters




### CapSense

Figure 10 shows the settings for the CapSense Component. See the CapSense Component datasheet for additional information.

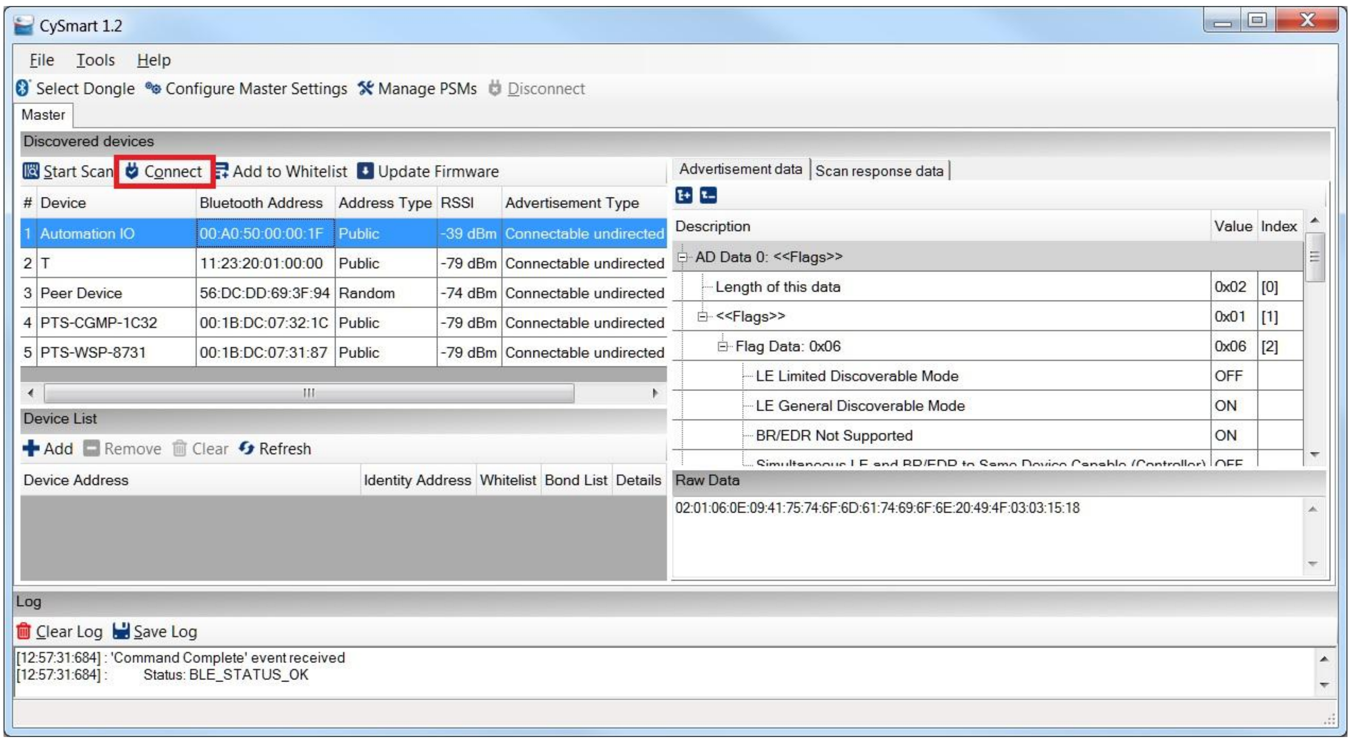
Figure 10. CapSense Component's Basic Tab



## Operation

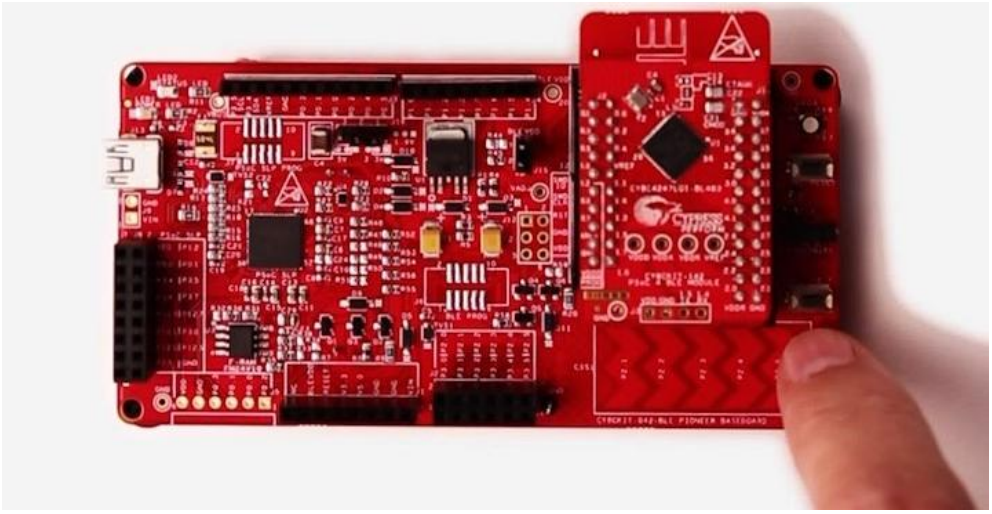
1. Build and program the BLE Automation IO Server project into the CY8CKIT-042 PSoC® 4 Pioneer Kit with a PSoC 4 BLE device.
2. Run a Windows terminal application such as HyperTerminal or PuTTy.
3. To use the CySmart Windows application as the BLE Automation IO Client, connect the CySmart BLE dongle to a USB port on the PC (Figure 3).
4. Launch the CySmart application and select the connected dongle in the dialog window.
5. Reset the development kit to start advertising by pressing the **SW1** button on the BLE Pioneer Kit.
6. Click the **Start Scan** button to discover available devices. Click **Stop Scan.**
7. Select **Automation IO** in the list of available devices and click **Connect** button:

Figure 11. CySmart Window



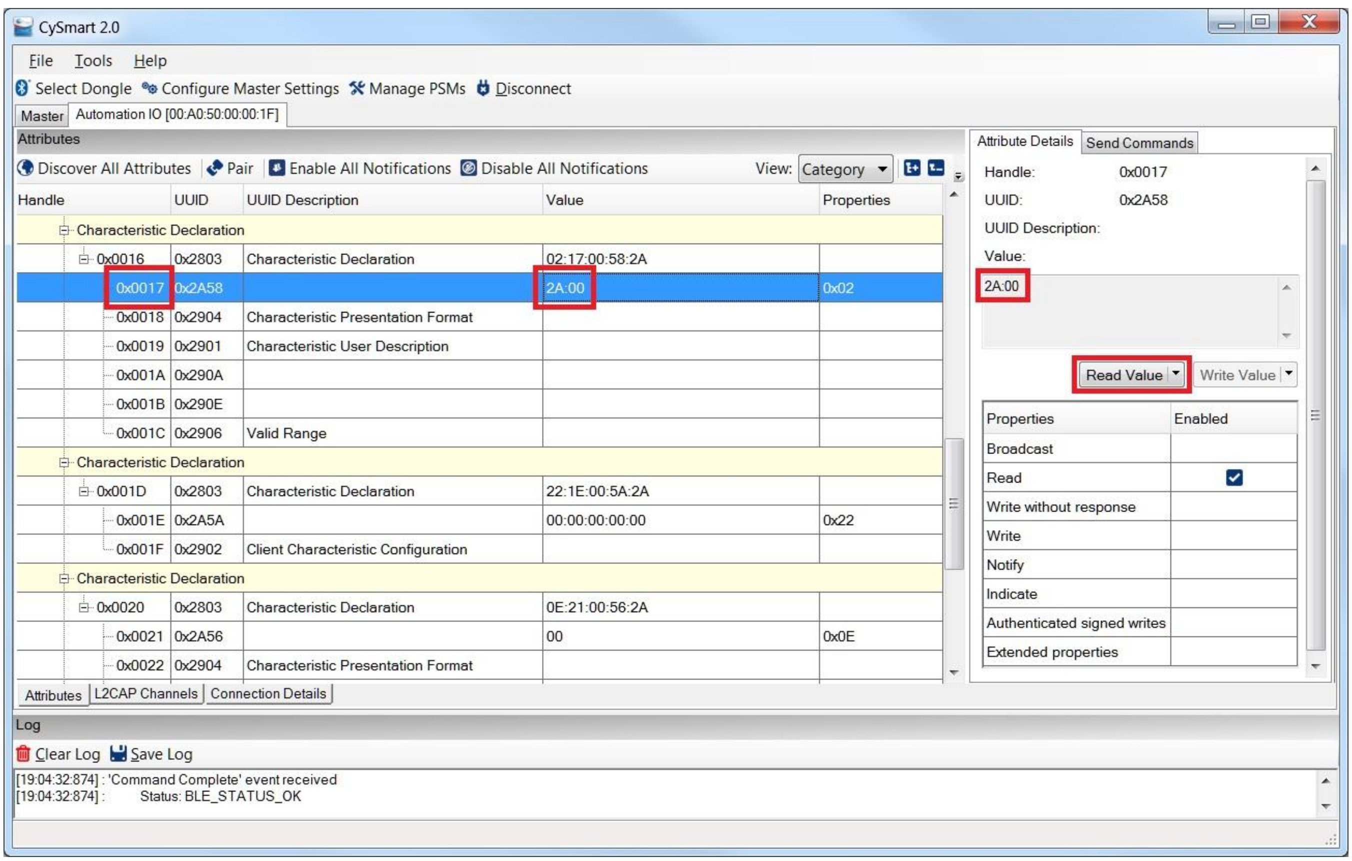

8. Click **Stop Scan** and **Start Scan** in CySmart. Select the IPS device.
9. Click **Pair**. Click **Yes** to the pairing request received from the peer device.
10. Compare the displayed passkeys on both devices. Click **Yes** on CySmart and **y** on the terminal application (or **SW2** button) to confirm the comparison pairing procedure.
11. Click **Discover All Attributes**, and then click **Read All Characteristics** in the CySmart application. Observe the received characteristic values.
12. To read a slider position, touch the linear slider on the CY8CKIT-042 PSoC 4 Pioneer Kit (Figure 12) and observe the slider position in the terminal program.

Figure 12. CapSense Linear Slider



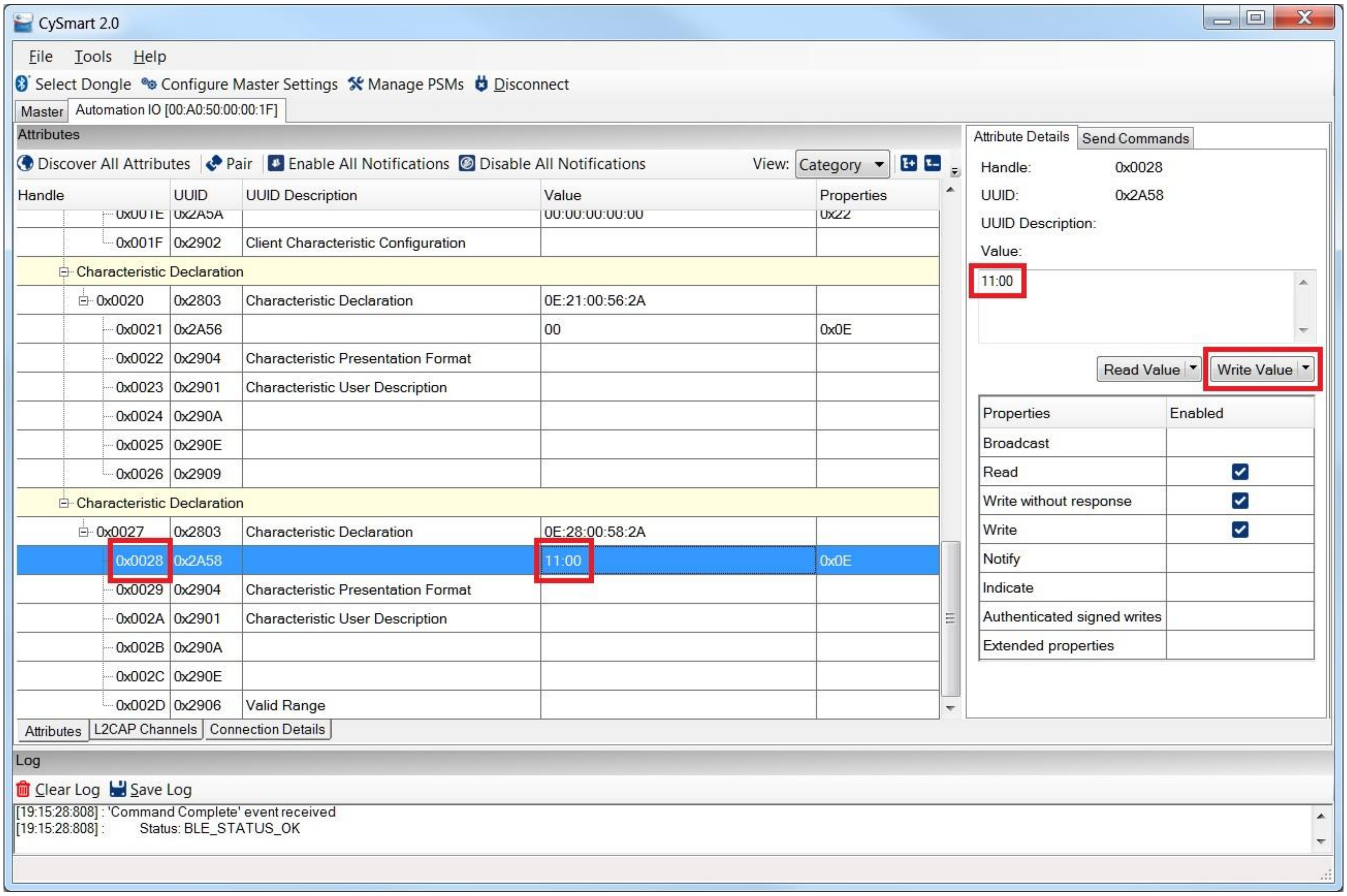
13. Select Analog characteristic Instance 0 (handle 0x0017) in CySmart and click **Read Value.** The received value should be the same as the slider position in the terminal program. See Figure 13. For detailed information about the CapSense Component, refer to the CapSense Component datasheet.

Figure 13. Reading Slider Position



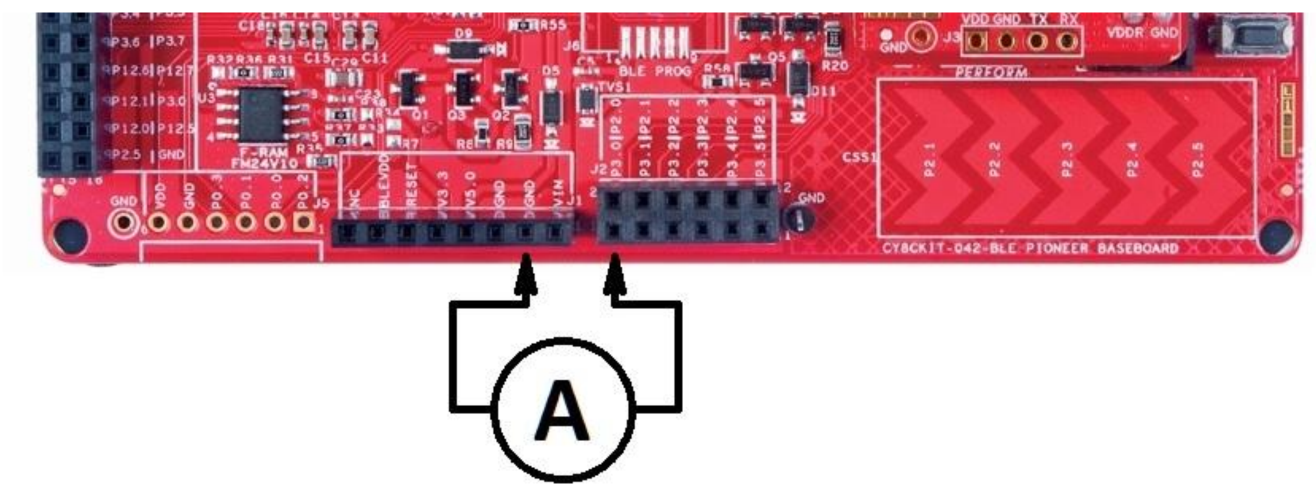
14. To set the IDAC current, select Analog characteristic Instance 1 (handle 0x0028) in CySmart, enter the value in the **Value** window, and click **Write Value** (Figure 14).

Figure 14. Setting IDAC Current




15. Observe the result in the terminal program. Also, you can measure the IDAC output current between P3.0 and GND pins on the CY8CKIT-042 PSoC 4 Pioneer Kit using an ammeter (Figure 15).

Figure 15. IDAC Current Measurement



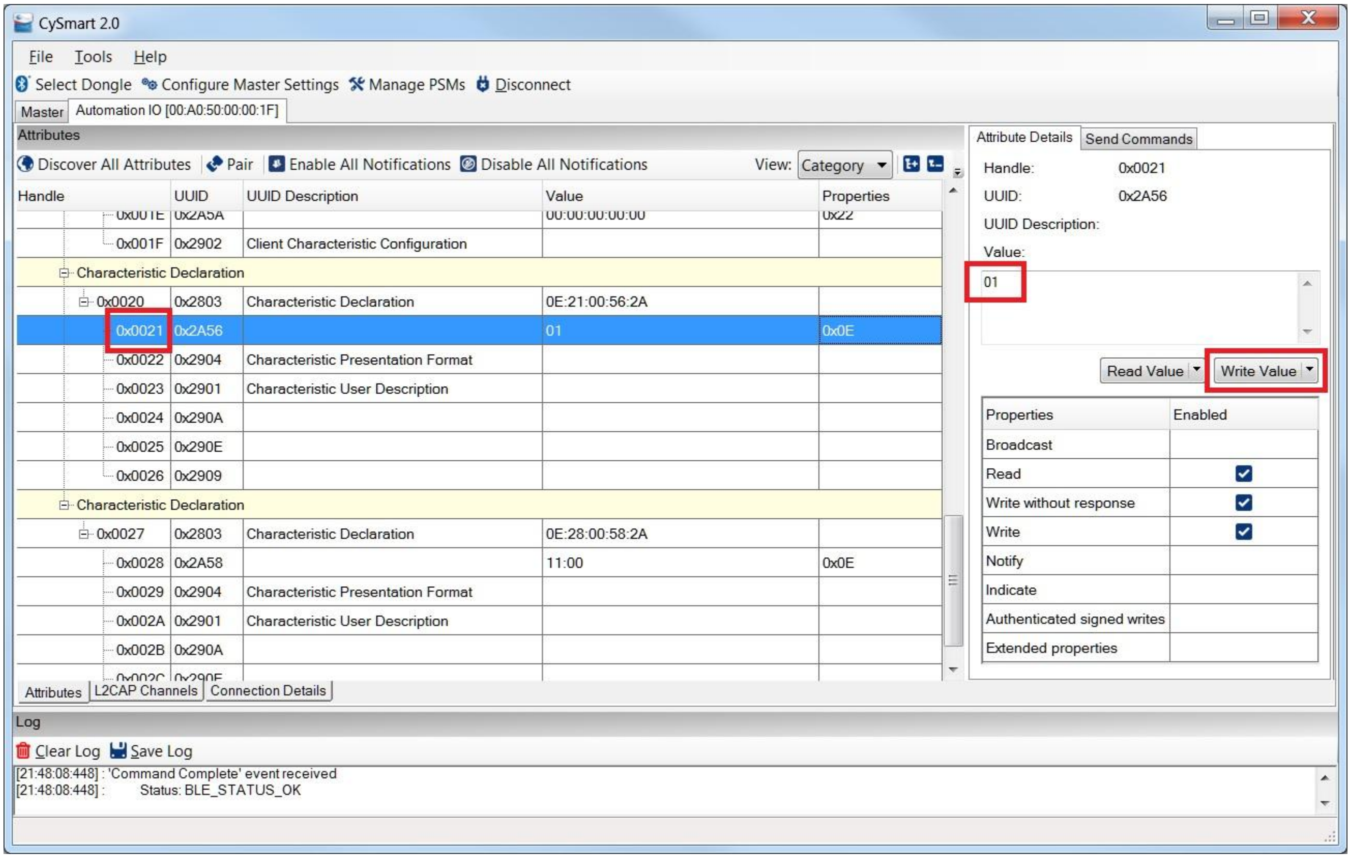
16. The max value of the IDAC current is 304.8 µA (2.4 µA/bit). The max value for Analog characteristic Instance 1 is 127.

    For detailed information about the IDAC, refer to the IDAC Component datasheet.

17. Do the following to control the LEDs from CySmart :
    - Turn the blue LED ON: Write 1 to the Digital characteristic Instance 1 (handle 0x0021).
    - Turn the blue LED OFF: Write 0 to the Digital characteristic Instance 1 (handle 0x0021).

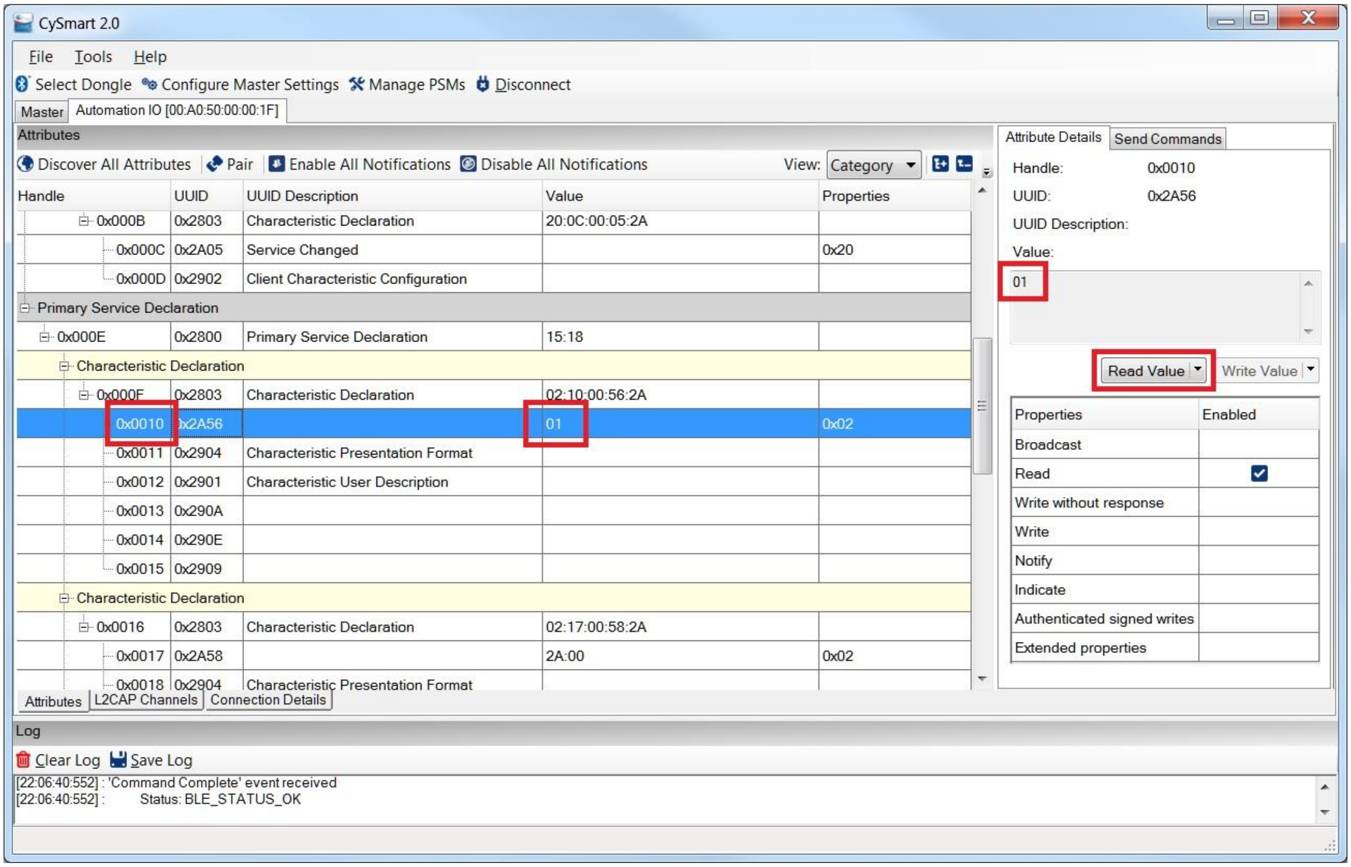
18. To execute the previous operation, do the following:
    - Select the Digital characteristic Instance 1 (handle 0x0021) in CySmart.
    - Enter the required value in the **Value** window.
    - Click **Write Value** (Figure 16).

Figure 16. LED Control



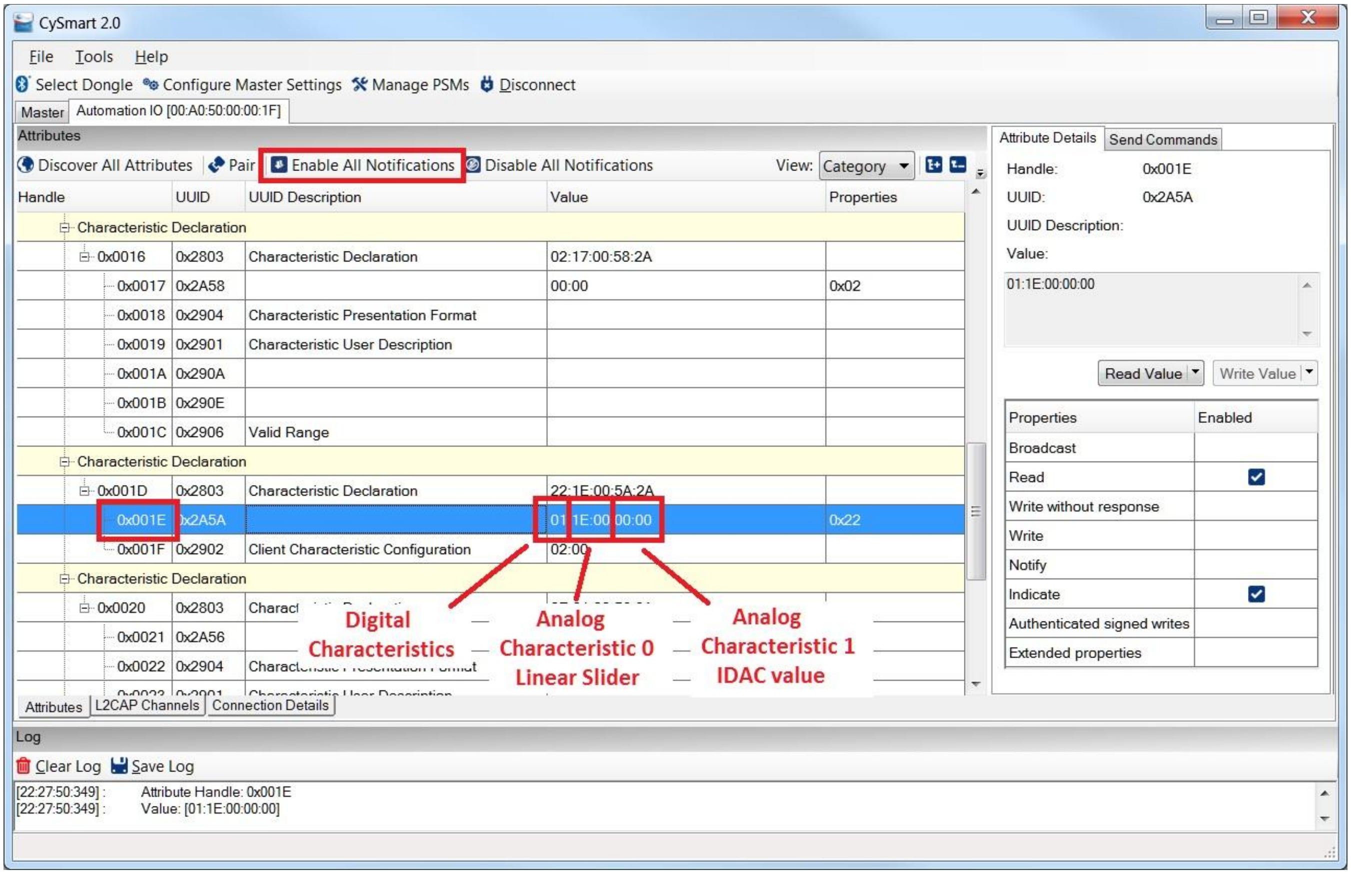
19. To read the **SW2** button, select the Digital characteristic Instance 0 (handle 0x0010) in CySmart and click **Read Value** (Figure 17).

Figure 17. LED Control



20. To aggregate the characteristic indication, click **Enable ALL Notifications** in CySmart and press the **SW2** button on the CY8CKIT-042 PSoC 4 Pioneer Kit. Observe the indicated value in the Aggregate characteristic (handle 0x001E) value window (Figure 18).

Figure 18. Aggregate Characteristic Indication




For detailed information about the CySmart Central Emulation Tool, refer to the CySmart User Guide.

## Related Documents

Table 2 lists all relevant application notes, code examples, knowledge base articles, device datasheets, and Component datasheets.

Table 2. Related Documents

| | | |
|---|---|---|
| **Application Notes** | | |
| AN91267 | Getting Started with PSoC® 4 BLE | Introduces PSoC® 4 BLE, an ARM® Cortex™-M0 based Programmable System-on-Chip (PSoC) with a Bluetooth Low Energy. |
| AN94020 | Getting Started with PRoC™ BLE | Introduces PRoC™ BLE, an ARM® Cortex®-M0 based programmable radio-on-chip with Bluetooth Low Energy. |
| AN91184 | PSoC 4 BLE - Designing BLE Applications | Shows how to design the BluetoothLow Energy (BLE) application based on PSoC 4 BLE, using standard profiles defined by the Bluetooth SIG included in the BLE Component in PSoC  Creator.  Demonstrates how to build an application with the BLE Health Thermometer Profile on the CY8CKIT-042-BLE kit. |
| **Videos** | | |
| PSoC 4 BLE 101: Intro to Bluetooth Low Energy | | This is the first installment of a series of getting-started videos on Cypress Bluetooth Low Energy solutions. |
| PSoC 4 BLE 101: 2 Configuring a Find Me Profile with BLE | | Using Cypress Pioneer kit with a PSoC 4 Radio module. Alan Hawse walks you through a simple example for a find-me tag application. |
| PSoC 4 BLE 101: 3 Finishing the Find Me Application with Firmware | | In this lesson, we take the Find Me profile you configured in the previous video and add the firmware required to make it work on the PSoC 4 BLE device. |
| PSoC 4 BLE 101: 4 Adding Battery Level Service and Testing with CySmart | | This lesson takes the Find Me profile built in the first two lessons and adds a Battery Level service. |
| PSoC 4 BLE 101: 5 Using CapSense with Bluetooth Low Energy | | In this BLE lesson, we show how to use PSoC Creator's Custom Service to quickly and easily add a CapSense® slider to a BLE (Bluetooth Low Energy) design. |
| PSoC 4 BLE 101: 6 Extending Battery Life with PSoC Low Energy Modes | | Adds power savings into your BLE designs easily using PSoC and PSoC Creator. In the last lesson, we created Find Me peripheral with the Battery Level service. |
| **Software and Drivers** | | |
| CySmart – Bluetooth® LE Test and Debug Tool | | CySmart is a Bluetooth® LE host emulation tool for Windows PCs. The tool provides an easy-to-use Graphical User Interface (GUI) to enable customers to test their Bluetooth LE peripheral applications. |
| **PSoC Creator Component Datasheets** | | |
| Bluetooth Low Energy (BLE) Component | | The Bluetooth Low Energy (BLE) Component provides a comprehensive GUI-based configuration window to facilitate designing applications requiring BLE connectivity. |
| PSoC 4 Serial Communication Block (SCB) Component | | Supports a PSoC 4 multifunction hardware block that implements $I^2C$, SPI, UART, and EZI2C communications |
| **Device Documentation** | | |
| PSoC® 4: PSoC 4XX7_BLE Family Datasheet Programmable System-on-Chip (PSoC®) | | |
| PSoC® 4: PSoC 4XX8_BLE Family Datasheet - Programmable System-on-Chip (PSoC®) | | |
| PSoC® 4: PSoC 4XX8  BLE 4.2 Family Datasheet Programmable System-on-Chip (PSoC®) | | |
| **Development Kit (DVK) Documentation** | | |
| Bluetooth® Low Energy Pioneer Kit (CY8CKIT-042-BLE) | | |

# Document History

Document Title: CE217613 - Bluetooth Low Energy (BLE) Automation IO

Document Number: 002-17613

| Revision | ECN | Origin of Change | Submission Date | Description of Change |
|---|---|---|---|---|
| ** | 5550104 | AZOV | 12/12/2016 | New code example. |

## 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

## 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