PSoC4 BLE - Constant Broadcaster

Objective

This example demonstrates how to use PSoC4 BLE device as Constant Broadcaster.

Overview

This code example uses the BLE component to demonstrate the Broadcaster role (GAP role) of BLE device. In this Constant Broadcaster example, device advertises constant data (advertising type: Scannable Undirected advertising). Any nearby observer or central device can receive the advertising data during scanning. The device responds to scan request from devices which are present in the white list. In this project address 0x00A500-B2A230 is already added in white list.

Requirements

Tool: PSoC Creator 4.0 Update 1, CySmart 1.2

Programming Language: C (GCC 4.9 – included with PSoC Creator)
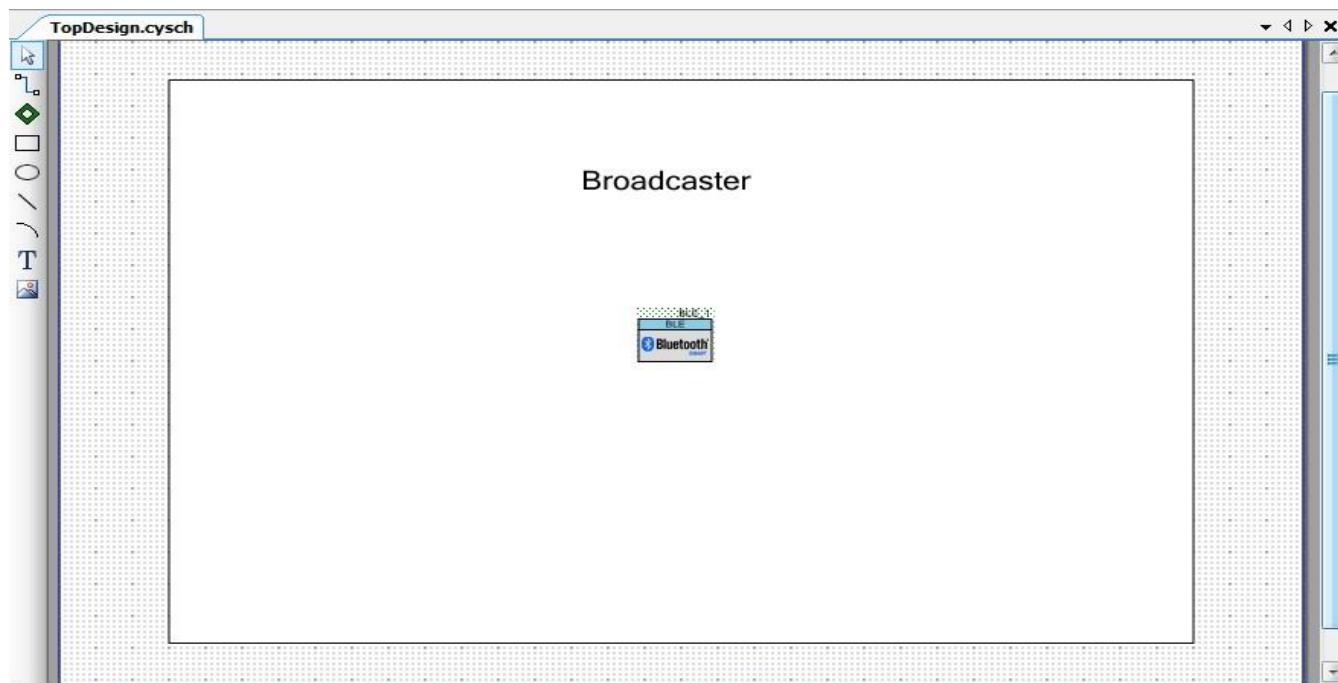
Associated Parts: All PSoC4 BLE parts

Related Hardware: CY8CKIT-042-BLE Bluetooth® Low Energy (BLE) Pioneer Kit

Hardware Setup:

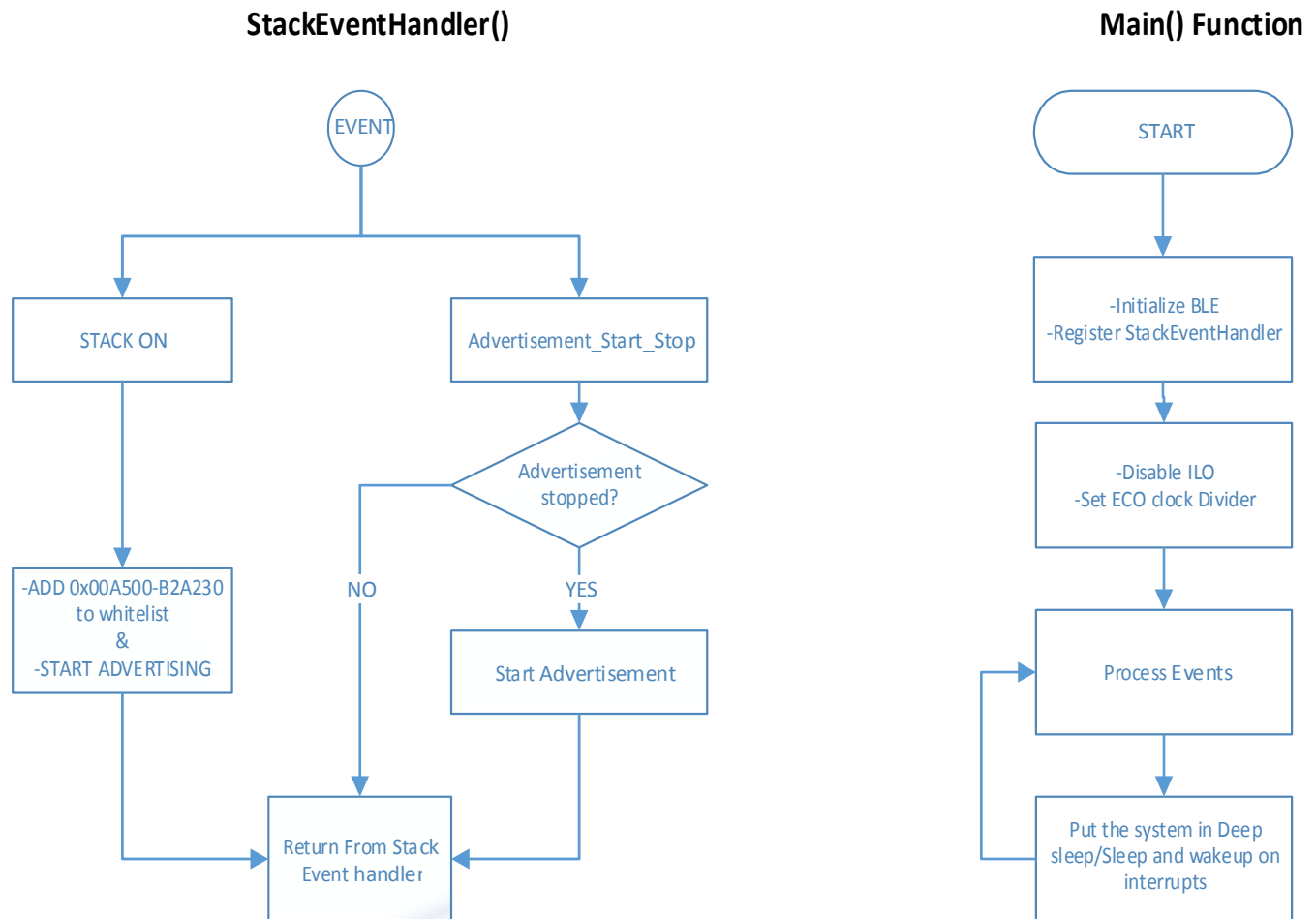BLE Pioneer Kit has the necessary hardware connections required for this lab. You can use your own hardware as well.

PSoC Creator Schematic

Figure 1. PSoC Creator Schematic

Operation:

Figure 2. Firmware Flow

StackEventHandler()

Main() Function

1. **main() function**: This is the central function which performs the initialization of the BLE stack, Disable ILO and set the clock divider for ECO, add dongle address to white list, executes the necessary routines to process the BLE events and starts advertising if the advertising timeout occurs. In the initial section of the *main()* function, the API *CyBle_Start(StackEventHandler)* is called to start the BLE Component and register a callback to the stack event handler. Note that the callback function can assume any name – in this project, we used StackEventHandler. Once the system is initialized, *main()* function continuously operates in a *while(1)* loop executing *CyBle_ProcessEvents()*. CyBle_ProcessEvents processes the events received by the BLE stack and enables application layer to use them and take the appropriate action. To reduce power consumption, in between the advertising interval the system will be in deep sleep/sleep when it is not advertising.

2. **StackEventHandler() function**: This function handles the common events generated for the BLE Stack. For example, the event *CYBLE_EVT_STACK_ON* is received when the Stack is initialized and turned ON. In this event address 0x00A500-B2A230 is added to white list. The event CYBLE_EVT_GAPP_ADVERTISEMENT _START_STOP is received when advertising is STARTED or STOPPED.

Build and Program the device:

This section shows how to build the project and program PSoC4 BLE device. If you are using a development kit with a built-in programmer (BLE Pioneer Kit, for example), connect the BLE Pioneer Baseboard to your computer using the USB Standard-A to Mini-B cable. For other kits, refer to the kit user guide.

If you are developing on your own hardware, you need a hardware debugger, for example, a Cypress CY8CKIT-002 MiniProg3.
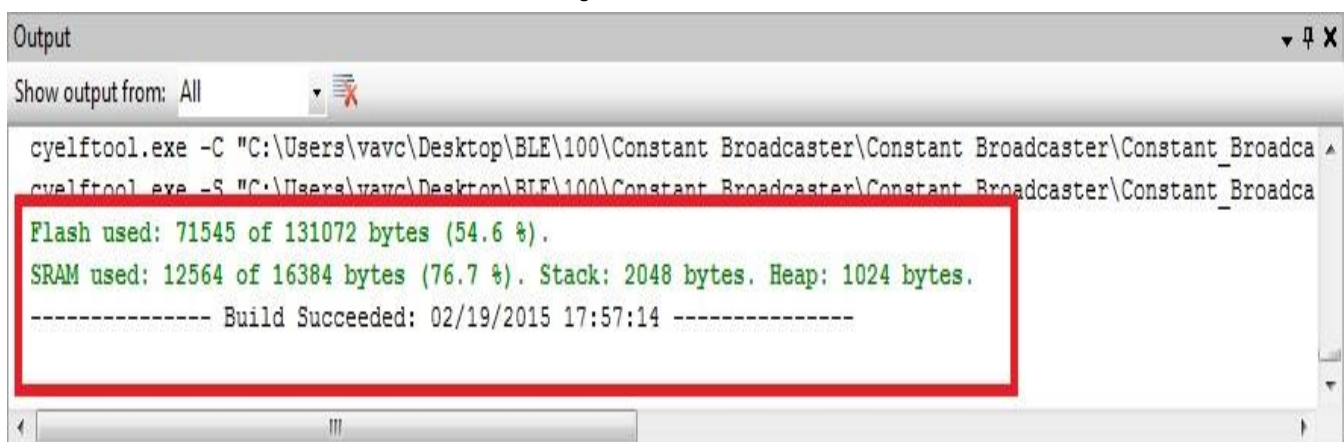
1. On PSoC Creator, select **Build** > **Clean and Build Constant_Broadcaster**, as shown in Figure 3.

Figure 3. Build Project
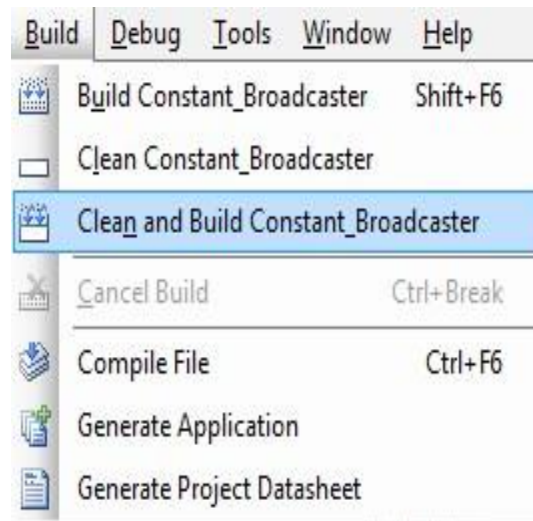
2. On successful build, total flash and SRAM usage is reported, as shown in
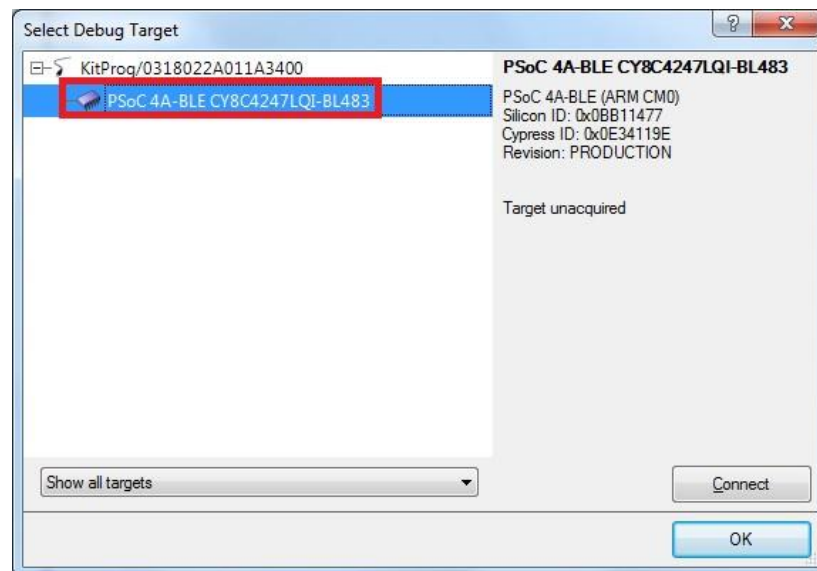3. Figure 4.

Figure 4. Build Succeeded

4.   Select **Debug** > **Select Debug Target**, as shown in Figure 5.

Figure 5. Selecting Debug Target
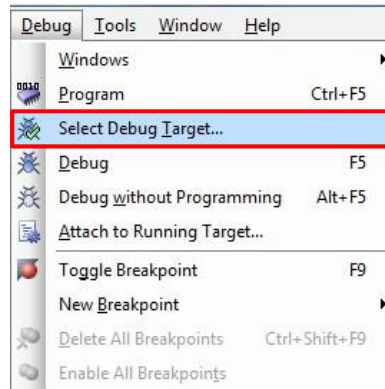
5.   In the Select Debug Target dialog box, click Port Acquire, and then click Connect, as shown in Figure 6. Click OK to close the dialog box.
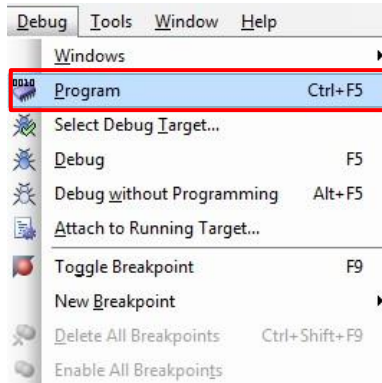
Figure 6. Connecting to a Device

If you are using your own hardware, make sure the Port Setting configuration under Select Debug Target window for your programming hardware is configured as per your setup.
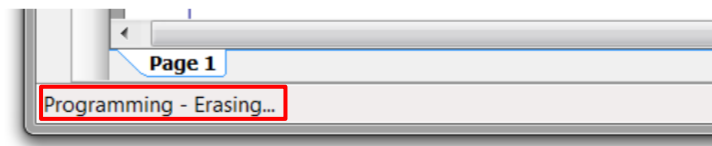
6. Select **Debug** > **Program** to program the device with the project, as shown in Figure 7.

Figure 7. Programming the Device

You can view the programming status on the PSoC Creator status bar (lower-left corner of the window), as shown in Figure 8.
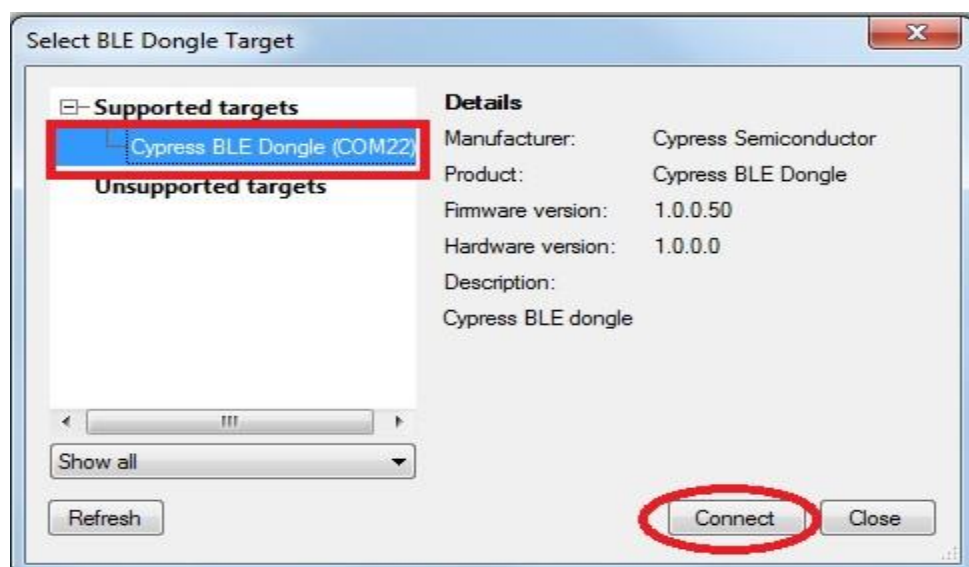
Figure 8. Programming Status

Testing and Expected Results:

1. Plug the BLE Pioneer Kit to your computer's USB port and programming the device.
2. Connect dongle to PC and then open Cysmart. Select dongle and then "**CONNECT**" as shown in Figure 9.

Figure9. Select dongle in CySmart

3. Go to "**Configure Master Settings**" to change the address of central (dongle) device

Edit the address of the dongle to **0x00A500B2A230** as shown in Figure 10(i), 10(ii)

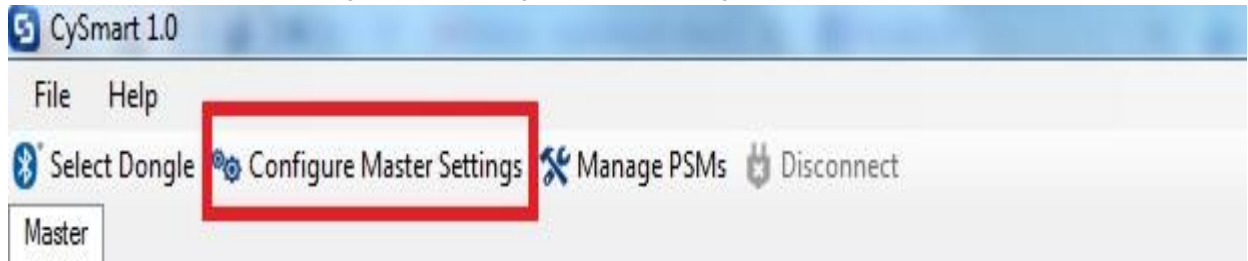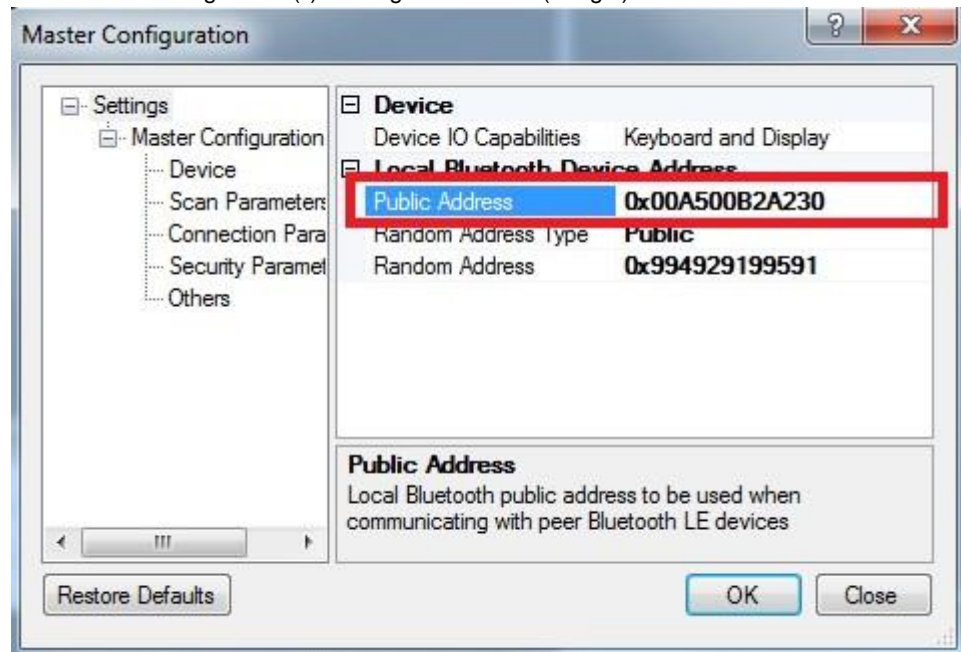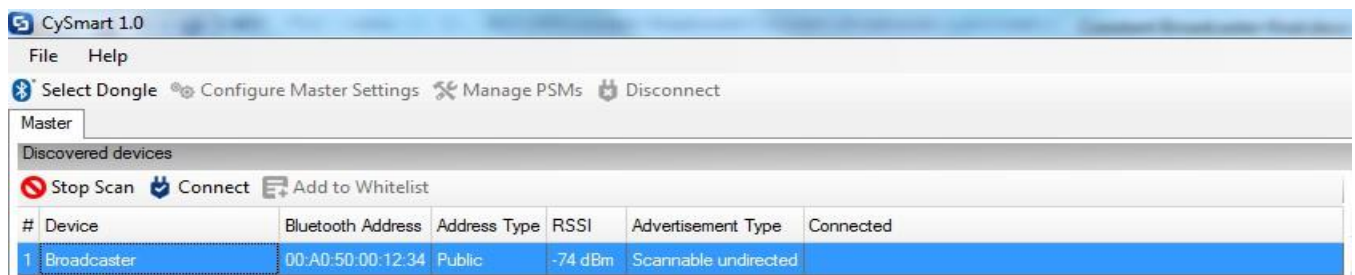Figure 10(i). Change the central (dongle) device address

Figure 10 (ii). Change the central (dongle) device address

4. Now select **START SCAN** in the Cysmart. We can see the device **Broadcaster** in the list as shown in Figure11

Figure11. Dongle detects "Broadcaster" device

5. To see the advertising data received from "Broadcaster" device select **Advertisement data** in the Cysmart as shown in Figure12.

Figure 12 Advertisement data

6. To see the scan response data received from "Broadcaster" device select **Scan Response Data** in the Cysmart as shown in Figure13.

Figure 13. Scan Response Data

Related Documents

Table 1 lists all relevant application notes, code examples, knowledge base articles, device datasheets, and Component / user module datasheets.

Table 1. Related Documents

Document	Title	Comment
AN91267	Getting Started with PSoC4 BLE	Provides an introduction to PSoC4 BLE device that integrates a Bluetooth Low Energy radio system along with programmable analog and digital resources.
AN91445	Antenna Design Guide	Provides guidelines on how to design an antenna for BLE applications.