BLE Cycling Sensor

1.0

Features

- Cycling Power Profile in CP Sensor and Broadcaster role operation
- Cycling Speed and Cadence Service
- DeepSleep mode support
- LED status indication

General Description

This example demonstrates the Cycling Speed and Cadence Service (CSCS) and Cycling Power Service (CPS). Cycling Speed and Cadence simulates a cycling activity and reports the simulated cycling speed and cadence data to a BLE central device using CSCS. Cycling Power simulates cycling power data and reports the simulated data to a BLE central device using CPS.

Project Configuration

The example project consists of the following components: BLE, Global Signal, UART, digital output pin, digital input pin. The Global Signal Reference and WDT_Interrupt component is used for the ISR configuration from Watchdog Timer. This timer works over low power deep sleep mode, therefore used as a general timer for simulation purpose. The output pins are used to reflect the line signal output on the LED. The input pin is configured to the resistive pull up mode and is used to wake device from low power hibernate mode. The top design schematic is shown in **Figure 1**.

BLE Cycling Sensor Example project

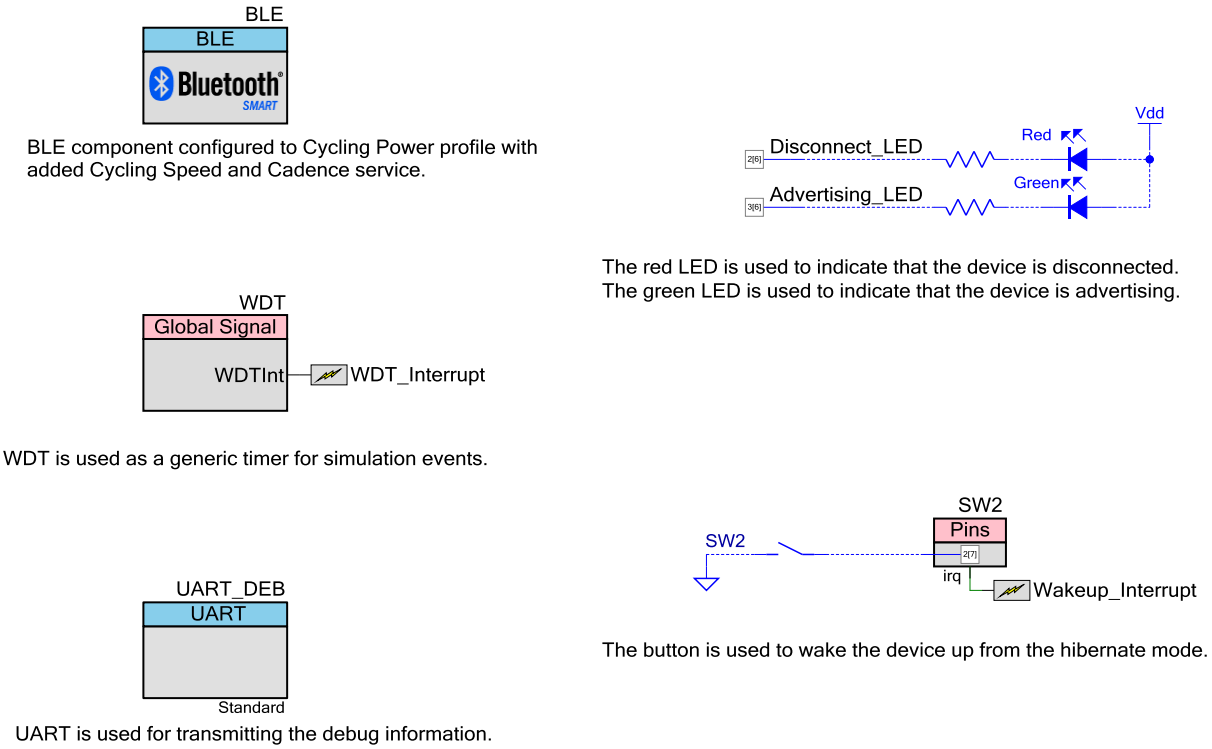

Figure 1. Top design schematic

The BLE component is configured as Cycling Power Profile in CP Sensor and Broadcaster role operation. Cycling Speed and Cadence service is added to CPP. Both services cover full sensor functionality required for cycling sensor.

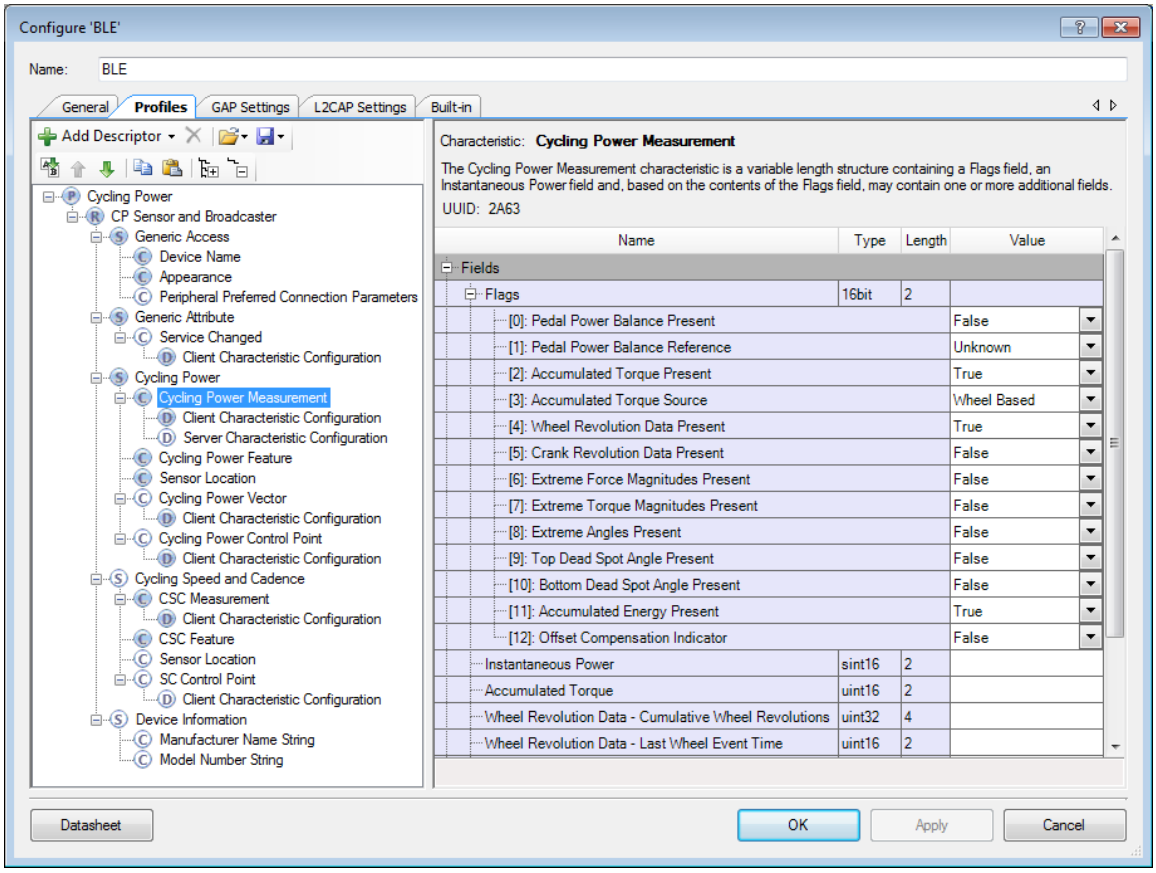

Figure 2. GATT settings

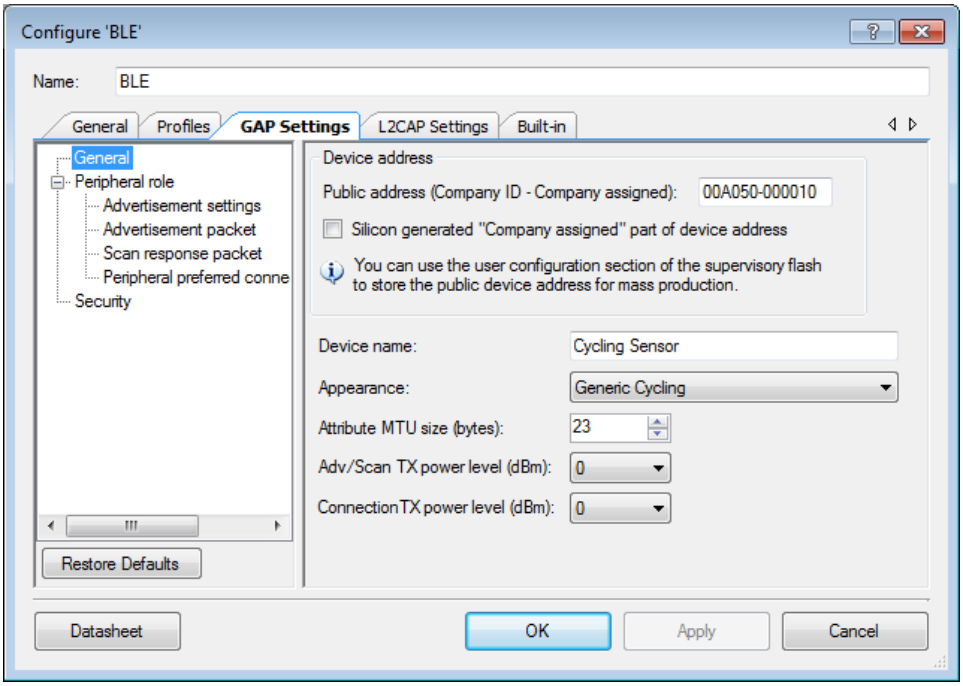

Figure 3. GAP settings

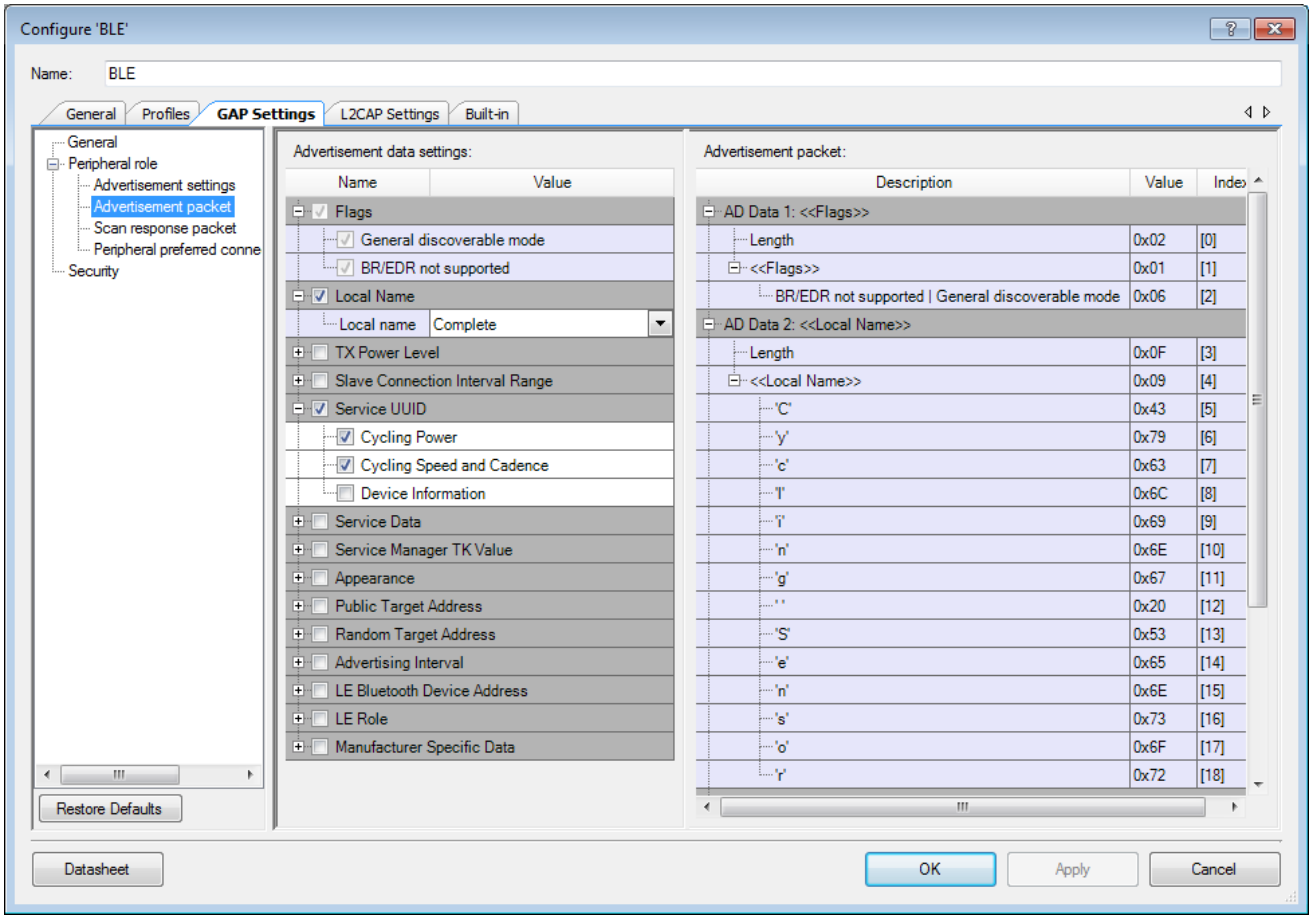

Figure 4. GAP settings -> Advertisement packet

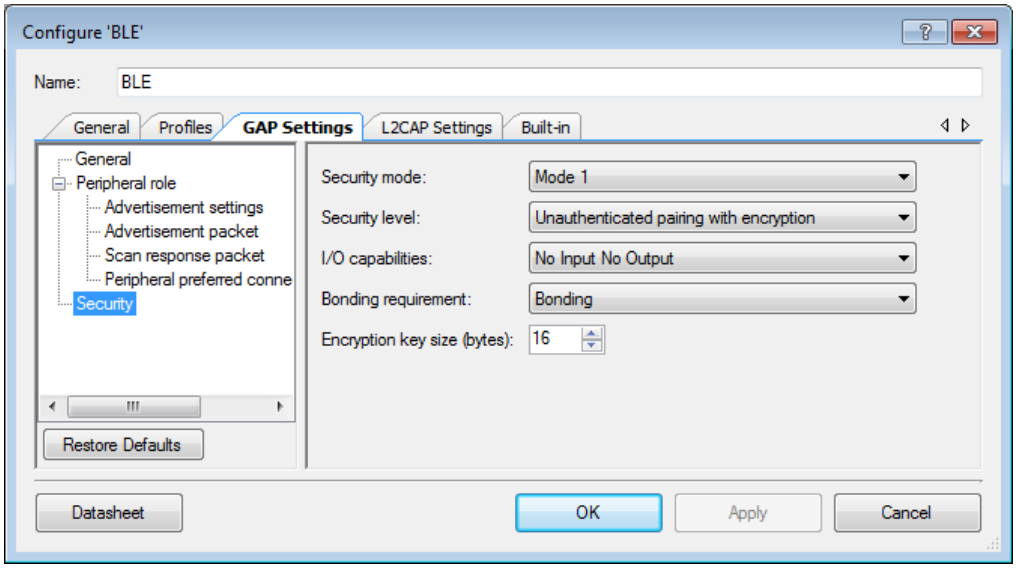

Figure 5. Security settings

Project Description

The project demonstrates BLE component functionality configured as a Cycling sensor.

One callback function (AppCallBack()) is required to receive generic events from BLE Stack. CyBle_GappStartAdvertisement() API is called after CYBLE_EVT_STACK_ON event to start advertising with the packet shown in **Figure 4**. CpsCallback() callback function receives events from the Cycling Power Service and CscsCallback() callback function receives events from the Cycling Speed and Cadence Service.

On advertisement timeout, the system remains in the hibernate mode. Press the mechanical button on CY8CKIT-042 BLE (SW2) to wake up the system and start re-advertising. BLE subsystem and CPU enters into low power Deep-Sleep mode between connection and advertising intervals. BLE subsystem automatically wakes up to maintain connection and advertising data transfer.

The instantaneous power, accumulated torque, wheel revolution and accumulated energy are simulated and send when notification of the Cycling Power Measurement characteristic is enabled by Client. Sensor Location characteristic is configures to “Top of shoe” and it could be updated by Client through write to Cycling Power Control Point characteristic with “Update Sensor Location” Op Code.

The measurement interval value is set to 1 second.

To indicate that the device is advertising, the green LED is blinking. The red LED is turned on after disconnection to indicate that no Client is connected to the device. When a Client is connected successfully, both red and green LEDs are turned off.

The project simulates Cycling Power Measurements characteristic with instantaneous power, accumulated torque, cumulative wheel revolution and accumulated energy values. Following table contains example of simulated data and expected calculation results.

	Instantaneous Power [W]	Accumulated Torque	Expected Accumulated Torque	Cumulative Wheel Revolution	Last Wheel Event Time [1/2048s]	Expected Instantaneous Speed [km/h]	Accumulated Energy Value [kJ]	Expected Accumulated Energy [kJ]
1	200	64960	2030.0	1000	63000	N/A	65532	65532
2	201	65280	2040.0	1008	65048	60.48	65534	65534
3	202	64	2050.0	1016	1560	60.48	0	65536
4	203	384	2060.0	1024	3608	60.48	2	65538
5	204	704	2070.0	1032	5656	60.48	4	65540

Expected Instantaneous Speed calculation is based on a wheel circumference of 210 centimeters.

The Power Vector characteristic is simulated with cumulative crank revolutions and last crank event time values. Example is shown in the table below.

	Cumulative Crank Revolutions	Last Wheel Event Time [1/1024s]	Expected Instantaneous Cadence [rpm]
1	65470	9300	N/A
2	65530	10324	60
3	54	11348	60
4	114	12372	60
5	174	13396	60

Refer to Cycling Power Profile and Cycling Speed and Cadence Profile specifications for calculation details.

Expected Results

You can use CySmart app on a Windows PC, Android or iOS BLE-compatible device as Client for connection to Cycling Sensor.

To use CySmart Windows application as Cycling Power and Cycling Speed Client:

- Connect the CySmart BLE dongle to a USB port on the PC.
- Launch CySmart app and select connected dongle in dialog window.
- Reset development kit to start advertising by pressing SW1 button.
- Click **Start Scan** button to discover available devices.
- Select **Cycling Sensor** in the list of available devices and connect to it.
- Click **Pair**, then **Discover All Attributes**, then **Read All Characteristics**, and finally **Enable All Notifications** in CySmart app.
- **Enable notifications for Cycling Power and/or for Cycling Speed and Cadence:**

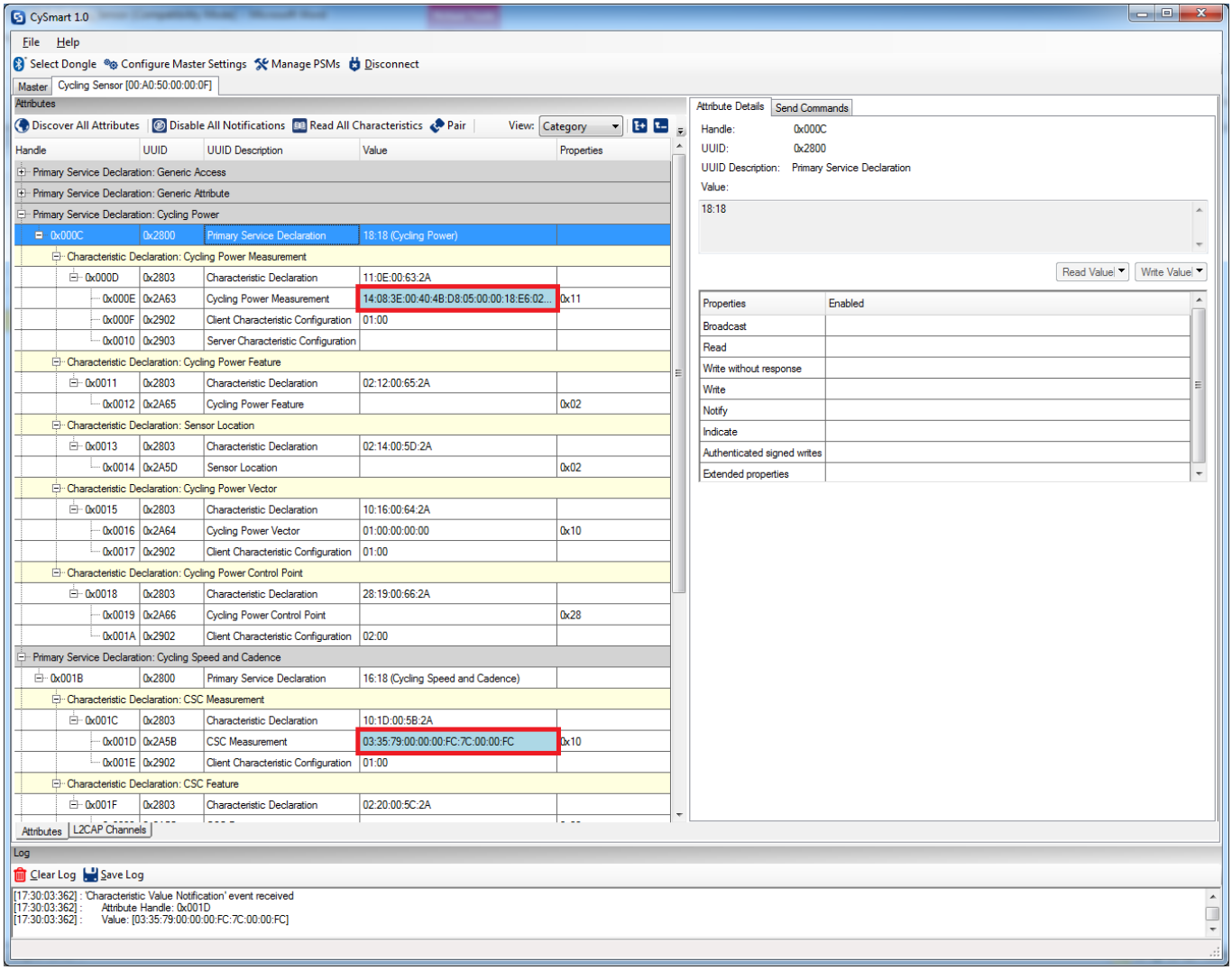

Figure 6. CySmart Windows app

If you have some problems with usage of CySmart app, please refer to CySmart User Guide.

To use CySmart mobile app as Cycling client:

- Launch CySmart mobile app (Android/iOS), and swipe down to refresh the list of found BLE devices.
- Connect to “Cycling Sensor” device and open “Cycling Speed and Cadence Service” or “Cycling Power Service” service.

Notice the simulated values, for example:

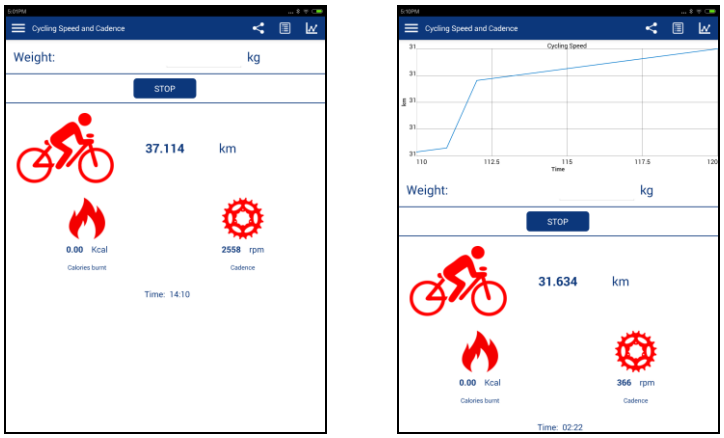

Figure 7. CySmart Android app