BLE Heart Rate Sensor

1.0

Features

- BLE Heart Rate Service support in the GATT Server role
- Simulating the Heart Rate data
- Reporting the workflow status through UART
- LED status indication

General Description

This example project demonstrates the BLE Heart Rate Sensor workflow. The project simulates Heart Rate data and performs communication with BLE enabled central/client device.

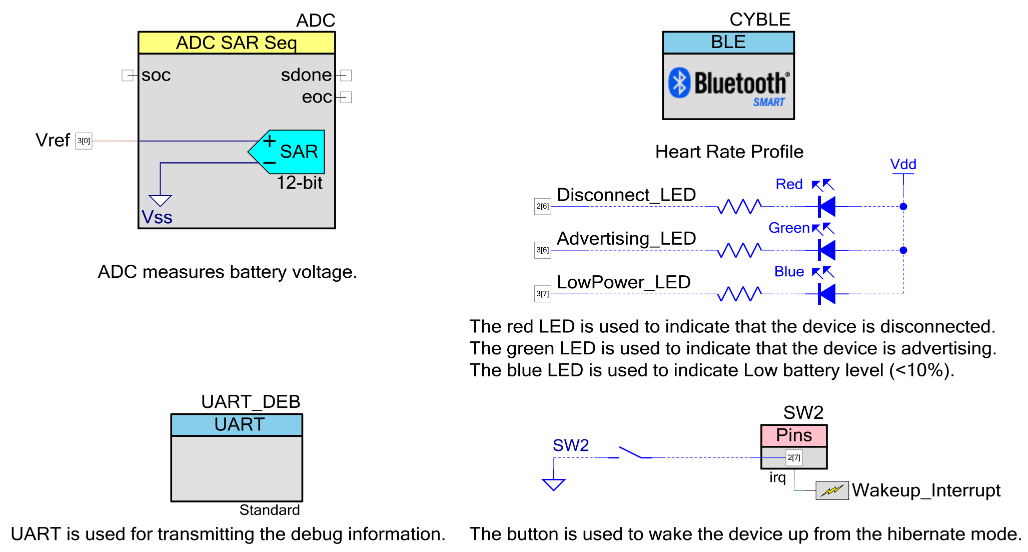
Development Kit Configuration

Default CY8CKIT-042 BLE Pioneer Kit configuration,

Connect J2 pin P3[0] to J3 pin VREF.

Project Configuration

BLE Heart Rate Sensor Example Project



ADC measures battery voltage.

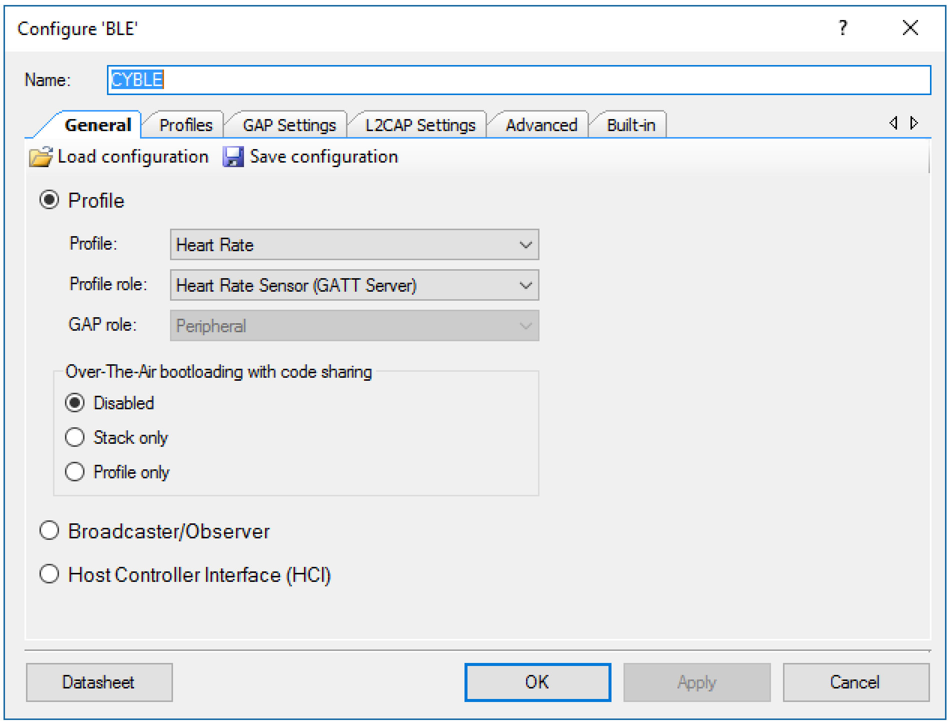
The red LED is used to indicate that the device is disconnected.
The green LED is used to indicate that the device is advertising.
The blue LED is used to indicate Low battery level (<10%).

UART is used for transmitting the debug information.

The button is used to wake the device up from the hibernate mode.

Figure 1. Top design schematic

The BLE component is configured as Heart Rate Sensor.



Configure 'BLE'

Name: CYBLE

General | Profiles | GAP Settings | L2CAP Settings | Advanced | Built-in

Load configuration | Save configuration

(●) Profile

Profile: Heart Rate

Profile role: Heart Rate Sensor (GATT Server)

GAP role: Peripheral

Over-The-Air bootloading with code sharing

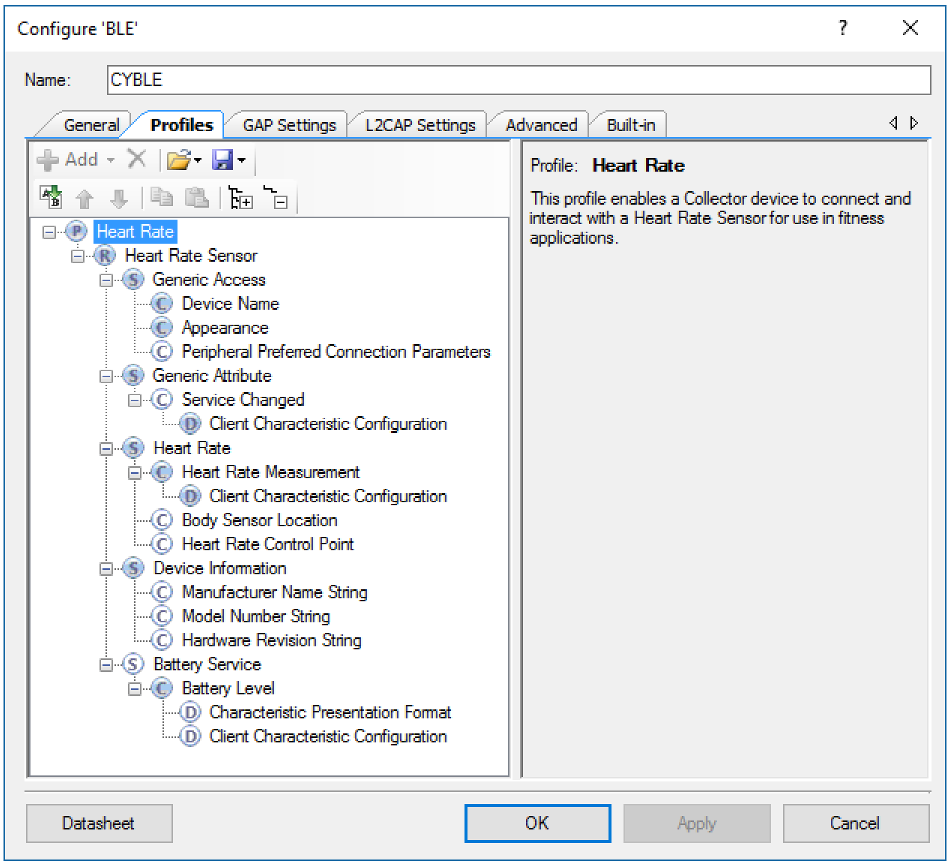
(●) Disabled
( ) Stack only
( ) Profile only

( ) Broadcaster/Observer

( ) Host Controller Interface (HCI)

Datasheet | OK | Apply | Cancel

Figure 2. BLE configuration



Configure 'BLE'

Name: CYBLE

General | Profiles | GAP Settings | L2CAP Settings | Advanced | Built-in

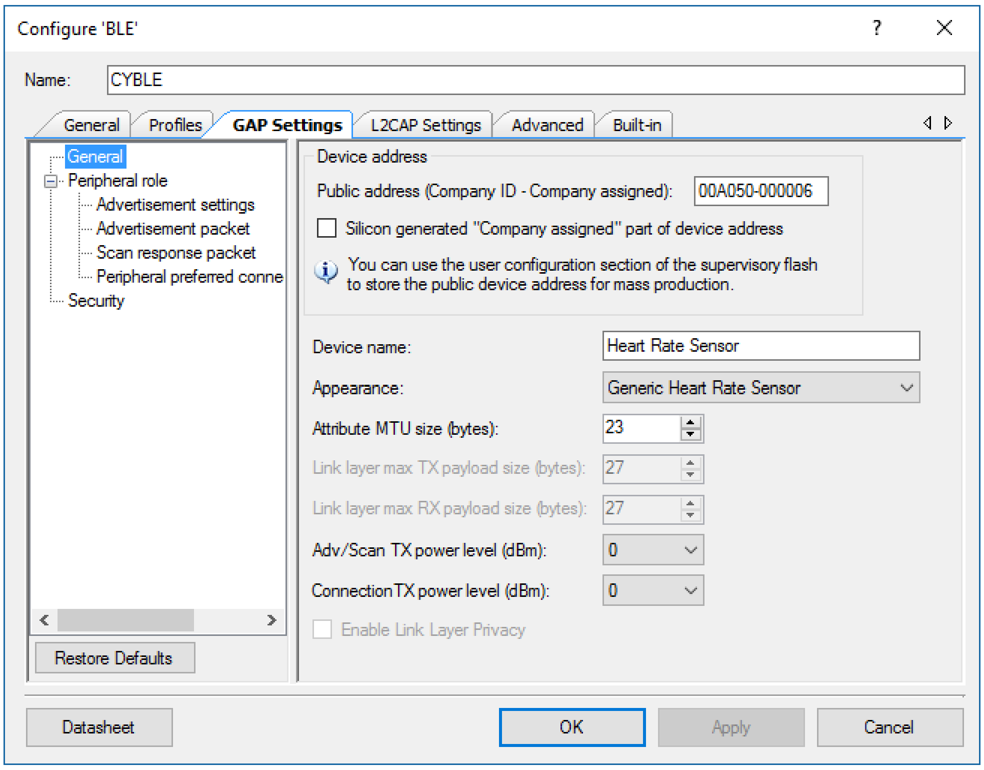
```
Heart Rate
  Heart Rate Sensor
    Generic Access
      Device Name
      Appearance
      Peripheral Preferred Connection Parameters
    Generic Attribute
      Service Changed
        Client Characteristic Configuration
    Heart Rate
      Heart Rate Measurement
        Client Characteristic Configuration
      Body Sensor Location
      Heart Rate Control Point
    Device Information
      Manufacturer Name String
      Model Number String
      Hardware Revision String
    Battery Service
      Battery Level
        Characteristic Presentation Format
        Client Characteristic Configuration
```

Profile: **Heart Rate**

This profile enables a Collector device to connect and interact with a Heart Rate Sensor for use in fitness applications.

Datasheet | OK | Apply | Cancel

Figure 3. GATT settings


Configure 'BLE'

Name: CYBLE

General | Profiles | **GAP Settings** | L2CAP Settings | Advanced | Built-in

- General
- Peripheral role
  - Advertisement settings
  - Advertisement packet
  - Scan response packet
  - Peripheral preferred conne
- Security

Restore Defaults

Device address

Public address (Company ID - Company assigned): 00A050-000006

☐ Silicon generated "Company assigned" part of device address

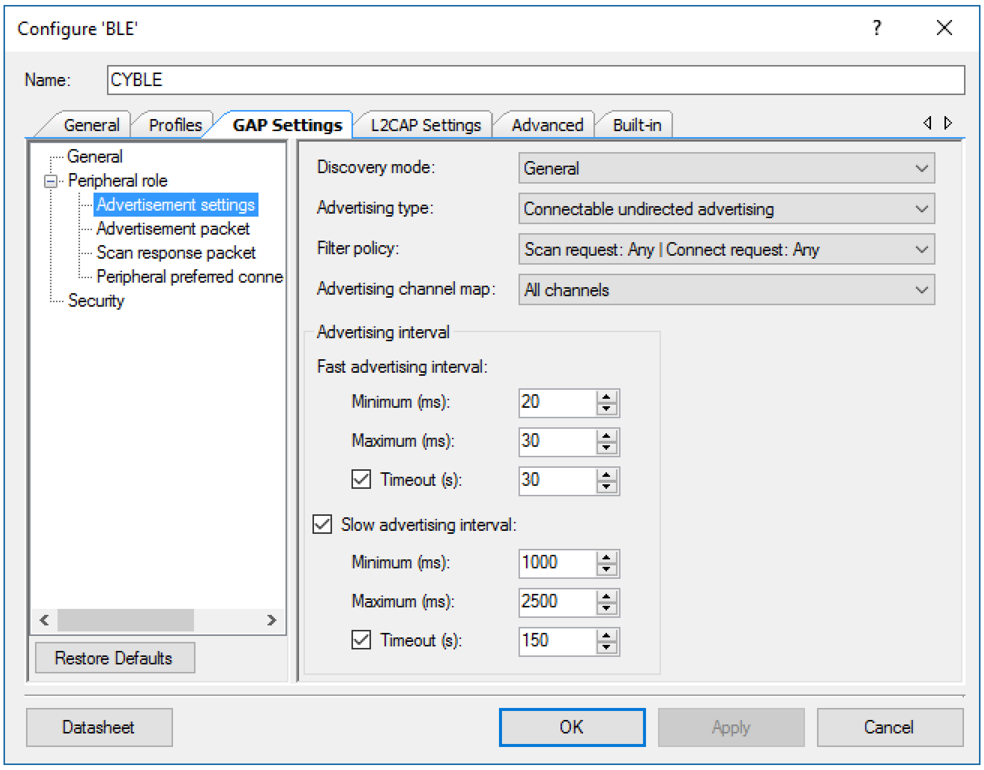
You can use the user configuration section of the supervisory flash to store the public device address for mass production.

Device name:	Heart Rate Sensor
Appearance:	Generic Heart Rate Sensor
Attribute MTU size (bytes):	23
Link layer max TX payload size (bytes):	27
Link layer max RX payload size (bytes):	27
Adv/Scan TX power level (dBm):	0
Connection TX power level (dBm):	0

☐ Enable Link Layer Privacy

Datasheet | OK | Apply | Cancel

Figure 4. GAP settings


Configure 'BLE'

Name: CYBLE

General | Profiles | **GAP Settings** | L2CAP Settings | Advanced | Built-in

- General
- Peripheral role
  - Advertisement settings
  - Advertisement packet
  - Scan response packet
  - Peripheral preferred conne
- Security

Restore Defaults

Discovery mode:	General
Advertising type:	Connectable undirected advertising
Filter policy:	Scan request: Any | Connect request: Any
Advertising channel map:	All channels

Advertising interval

Fast advertising interval:

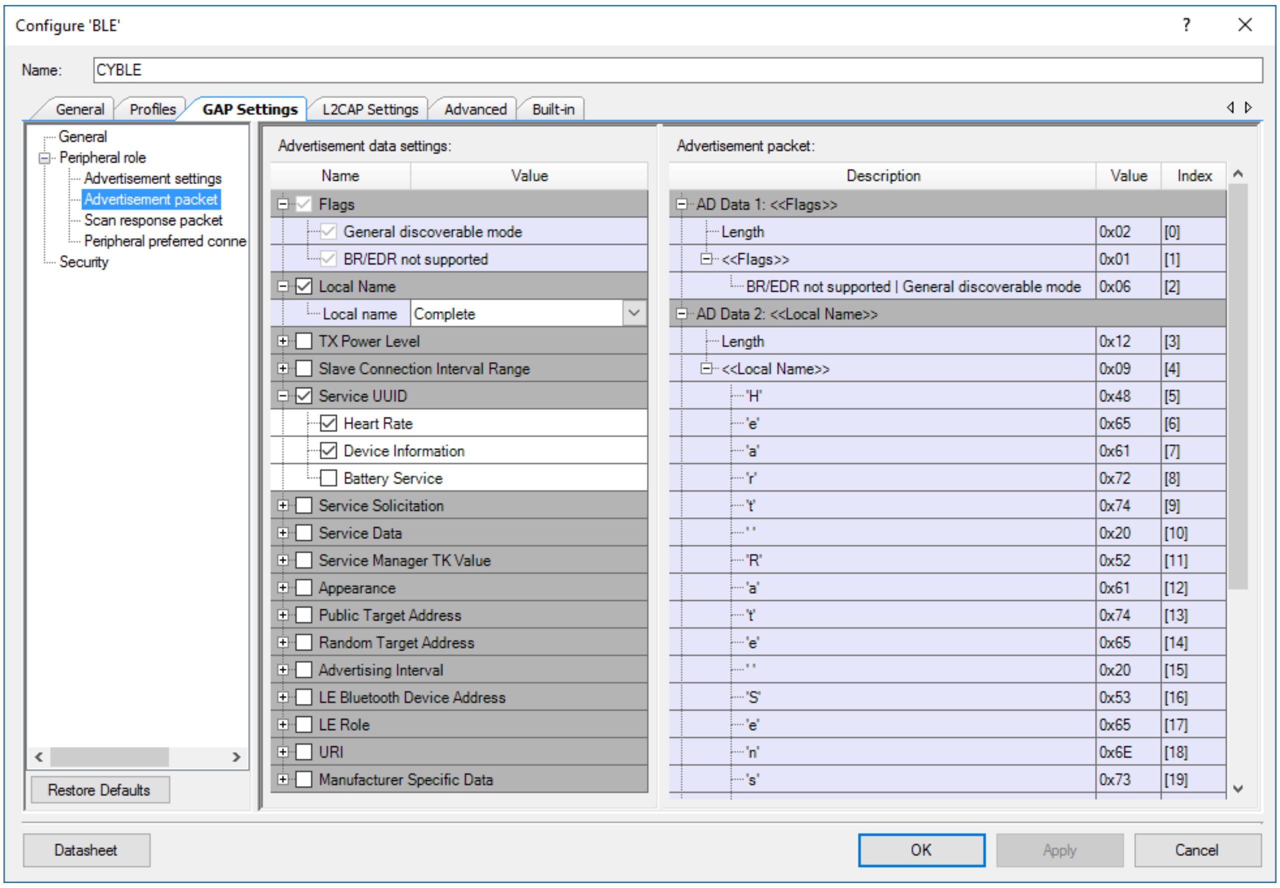
Minimum (ms):	20
Maximum (ms):	30
☑ Timeout (s):	30

☑ Slow advertising interval:

Minimum (ms):	1000
Maximum (ms):	2500
☑ Timeout (s):	150

Datasheet | OK | Apply | Cancel

Figure 5. GAP settings -> Advertisement settings



Name: CYBLE

- General
- Peripheral role
  - Advertisement settings
  - Advertisement packet
  - Scan response packet
  - Peripheral preferred conne
- Security

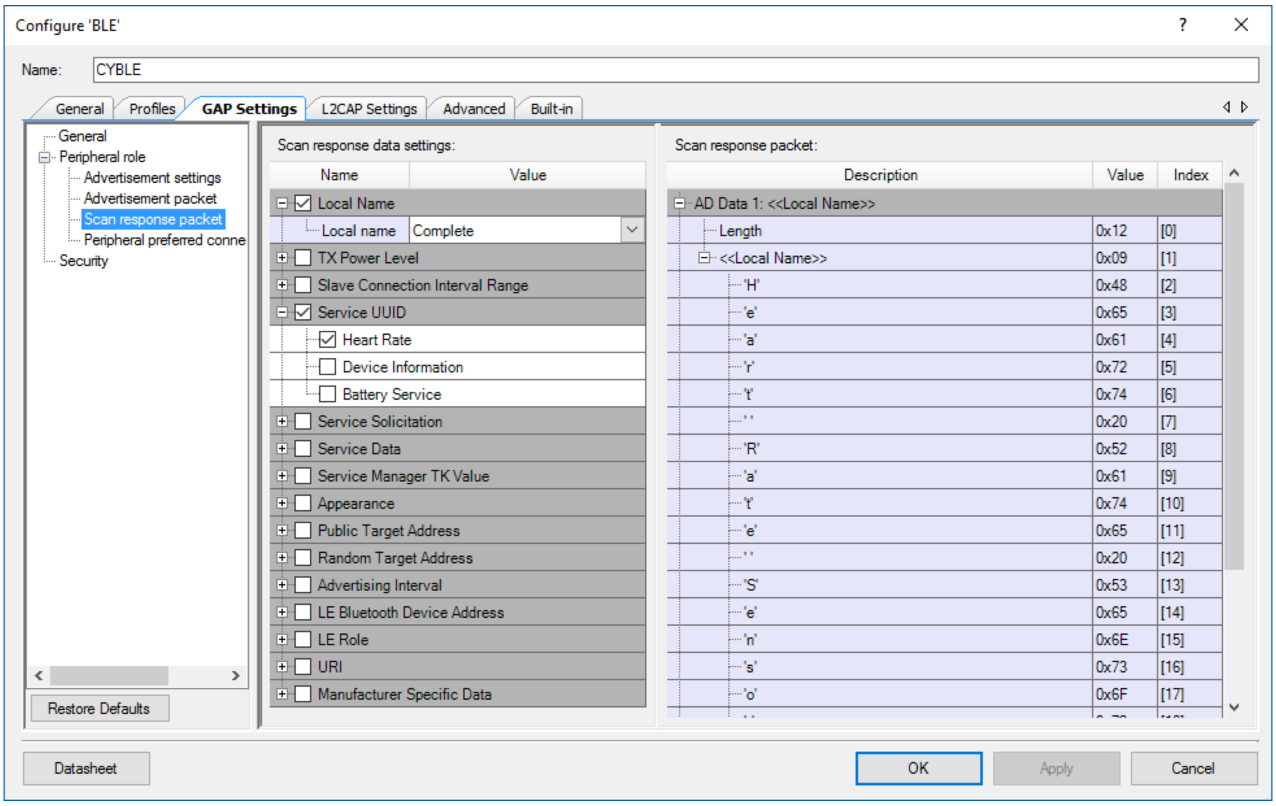
Advertisement data settings:

Name	Value
☑ Flags	
☑ General discoverable mode	
☑ BR/EDR not supported	
☑ Local Name	
Local name	Complete
☐ TX Power Level	
☐ Slave Connection Interval Range	
☑ Service UUID	
☑ Heart Rate	
☑ Device Information	
☐ Battery Service	
☐ Service Solicitation	
☐ Service Data	
☐ Service Manager TK Value	
☐ Appearance	
☐ Public Target Address	
☐ Random Target Address	
☐ Advertising Interval	
☐ LE Bluetooth Device Address	
☐ LE Role	
☐ URI	
☐ Manufacturer Specific Data	

Advertisement packet:

Description	Value	Index
AD Data 1: <<Flags>>		
Length	0x02	[0]
<<Flags>>	0x01	[1]
BR/EDR not supported | General discoverable mode	0x06	[2]
AD Data 2: <<Local Name>>		
Length	0x12	[3]
<<Local Name>>	0x09	[4]
'H'	0x48	[5]
'e'	0x65	[6]
'a'	0x61	[7]
'r'	0x72	[8]
't'	0x74	[9]
' '	0x20	[10]
'R'	0x52	[11]
'a'	0x61	[12]
't'	0x74	[13]
'e'	0x65	[14]
' '	0x20	[15]
'S'	0x53	[16]
'e'	0x65	[17]
'n'	0x6E	[18]
's'	0x73	[19]

Figure 6. GAP settings -> Advertisement packet



Name: CYBLE

- General
- Peripheral role
  - Advertisement settings
  - Advertisement packet
  - Scan response packet
  - Peripheral preferred conne
- Security

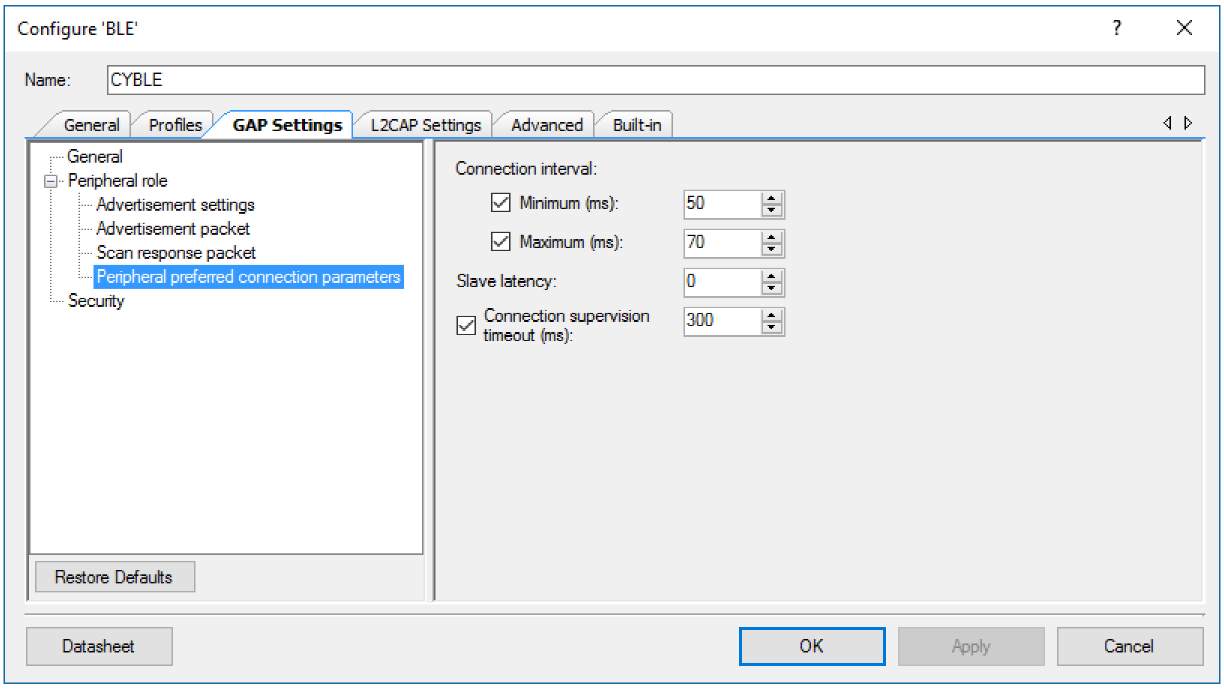
Scan response data settings:

Name	Value
☑ Local Name	
Local name	Complete
☐ TX Power Level	
☐ Slave Connection Interval Range	
☑ Service UUID	
☑ Heart Rate	
☐ Device Information	
☐ Battery Service	
☐ Service Solicitation	
☐ Service Data	
☐ Service Manager TK Value	
☐ Appearance	
☐ Public Target Address	
☐ Random Target Address	
☐ Advertising Interval	
☐ LE Bluetooth Device Address	
☐ LE Role	
☐ URI	
☐ Manufacturer Specific Data	

Scan response packet:

Description	Value	Index
AD Data 1: <<Local Name>>		
Length	0x12	[0]
<<Local Name>>	0x09	[1]
'H'	0x48	[2]
'e'	0x65	[3]
'a'	0x61	[4]
'r'	0x72	[5]
't'	0x74	[6]
' '	0x20	[7]
'R'	0x52	[8]
'a'	0x61	[9]
't'	0x74	[10]
'e'	0x65	[11]
' '	0x20	[12]
'S'	0x53	[13]
'e'	0x65	[14]
'n'	0x6E	[15]
's'	0x73	[16]
'o'	0x6F	[17]

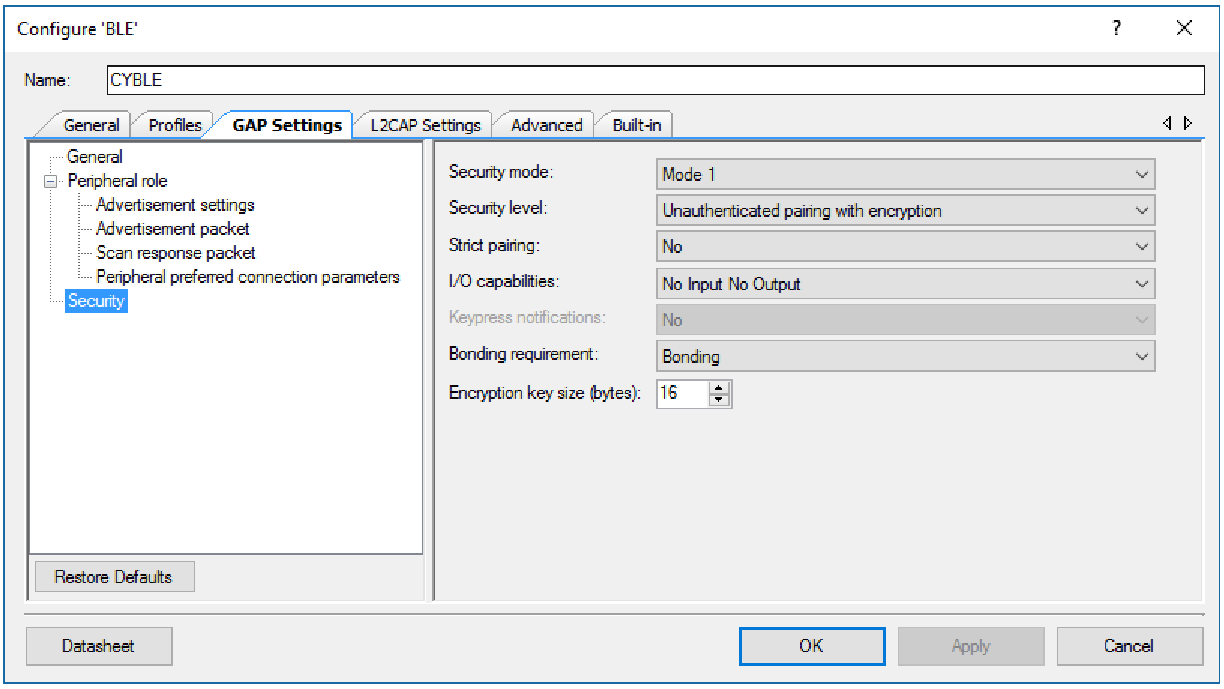
Figure 7. GAP settings -> Scan response packet



Connection interval:

☑ Minimum (ms):	50
☑ Maximum (ms):	70
Slave latency:	0
☑ Connection supervision timeout (ms):	300

Figure 8. GAP settings -> Peripheral preferred connection parameters



Security mode:	Mode 1
Security level:	Unauthenticated pairing with encryption
Strict pairing:	No
I/O capabilities:	No Input No Output
Keypress notifications:	No
Bonding requirement:	Bonding
Encryption key size (bytes):	16

Figure 9. Security settings

Project Description

The project demonstrates the BLE workflow procedures like advertising, connecting, notifying Heart Rate data, Battery Level, etc.

The project is designed so there is no need to initiate any of mentioned actions manually – it automatically starts the BLE Stack, then, when the Stack is on (STACK_ON event is received), the advertising GAP procedure is initiated. The green LED is blinking while the device is advertising. Once connection request is received, it performs the connection procedure and provides its GATT database (configured in the GATT tab) for discovery process performed by client. The supported services are: Generic Access (GAP) and Attribute (GATT) Services, Heart Rate Service (HRS), Battery Service (BAS) and Device Information Service (DIS). When the Heart Rate notification is enabled by Client, the project starts to simulate all the Heart Rate Service related data (Heart Rate itself, Energy expended, RR-intervals). When the Battery Level notification is enabled by Client, the project starts to measure the voltage on Vref pin and notify the battery level. The WDT is used to timing the simulations, measurements and LED blinking. The blue LED turns on when the battery level value is less than 10%. The red LED is turned on after disconnection to indicate that no Client is connected to the device. On disconnection event the device immediately starts to advertising. When the device connects successfully, both red and green LEDs are turned off.

After 180 seconds timeout, if no central device has been connected, the Heart Rate Sensor stops advertising, a red LED is turned on indicating the disconnection state and the system enters into the hibernate mode. Press the mechanical button on CY8CKIT-042 BLE (SW2) to wake up the system and start re-advertising.

Expected Results

The project sends the Heart Rate and Battery Level notifications to the Central Client device which can show them for user. LEDs are blinking as described in Project Description section.

The project is intended to work in pair with the BLE Heart Rate Collector Example Project.

However, it can work with any other BLE-compatible device (e.g. phone, tablet) with appropriate software (with e.g. Android, iOS with installed application which supports Heart Rate Profile). For instance, you can use CySmart mobile app (Android / iOS) as Heart Rate Service client:

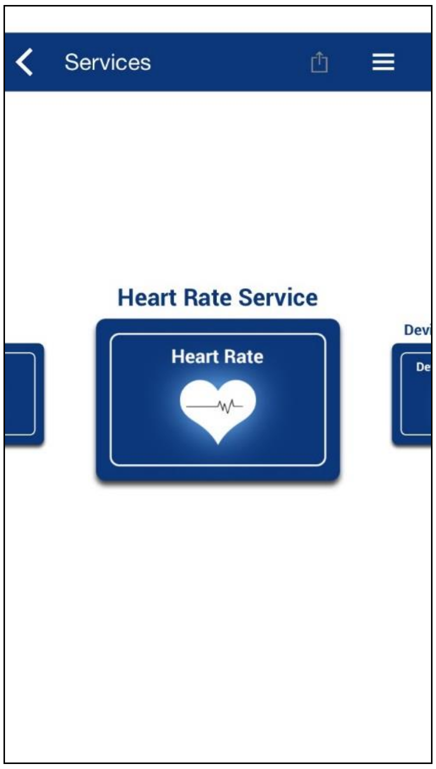


Figure 10. CySmart iOS app

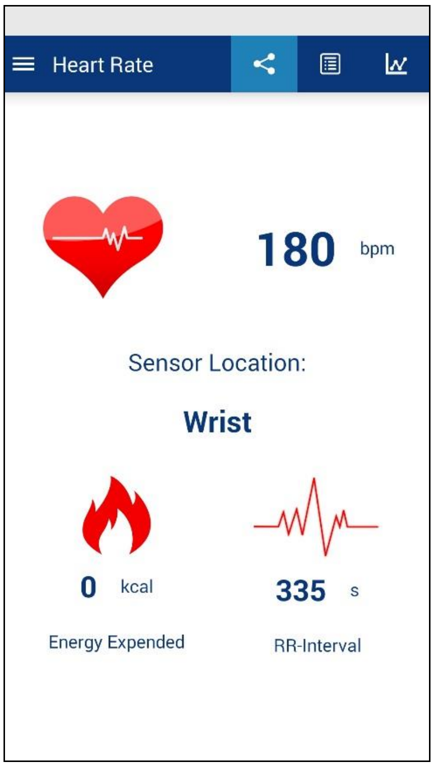


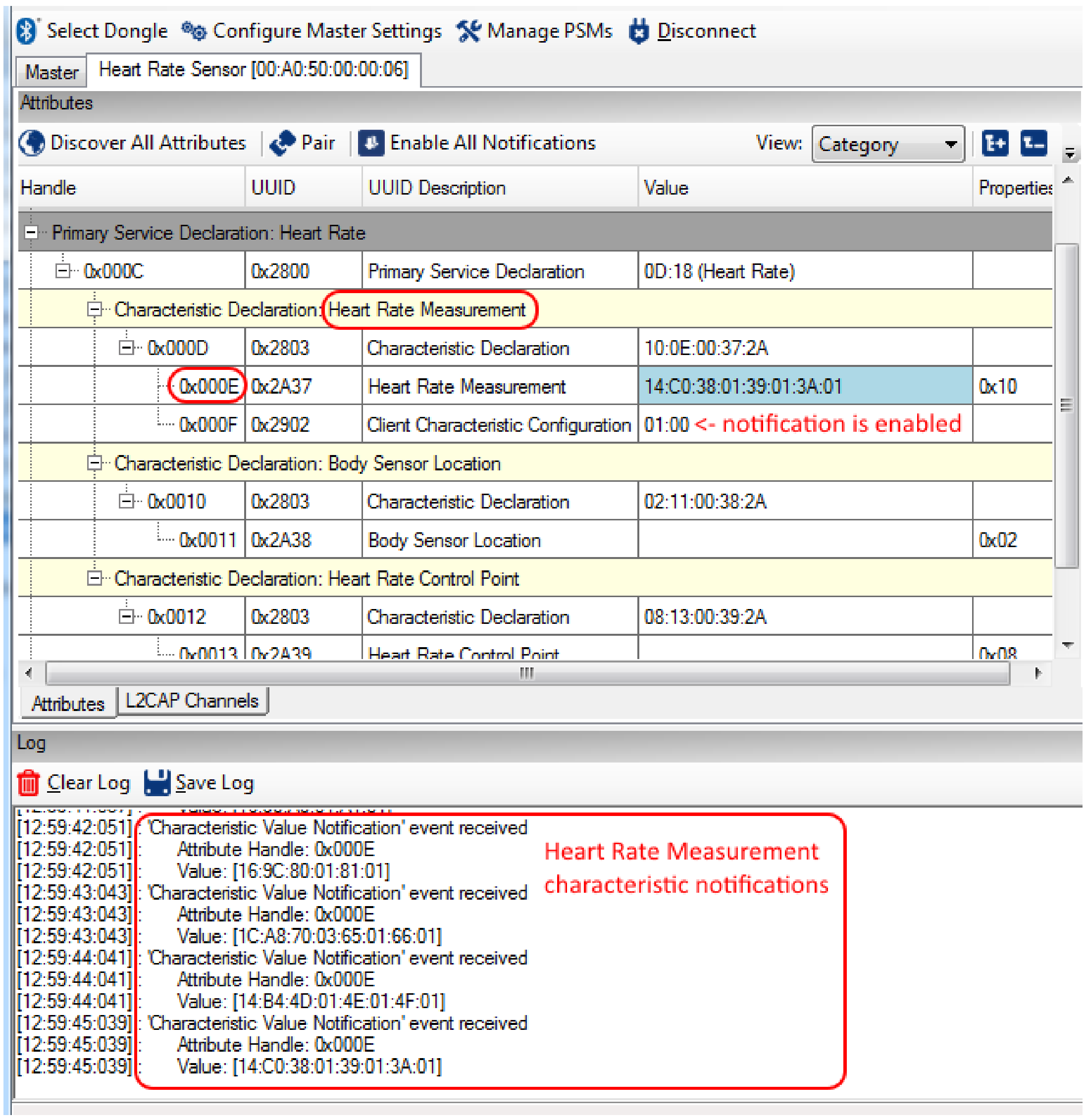
Figure 11. CySmart Android app

Also, the Heart Rate Sensor can be used together with CySmart app for Windows. It is required to match the security settings between Heart Rate Sensor and CySmart Client and perform pairing (bonding) before any writing (enabling notifications etc.) into Server's GATT database. For further instructions on how to use CySmart application, see CySmart User Guide.

The simple example how to use CySmart Windows application as Heart Rate Service client is the next:

- Connect the CySmart BLE dongle to a USB port on the PC.
- Launch CySmart app and select connected dongle in the dialog window.
- Reset the development kit to start advertising by pressing SW1 button.
- Click **Start Scan** button to discover available devices.
- Select **Heart Rate Sensor** in the list of available devices and connect to it.
- Click **Pair**, then **Discover All Attributes**, and **Enable All Notifications** in CySmart app.

Observe the Heart Rate Measurement characteristic notifications with simulated data:

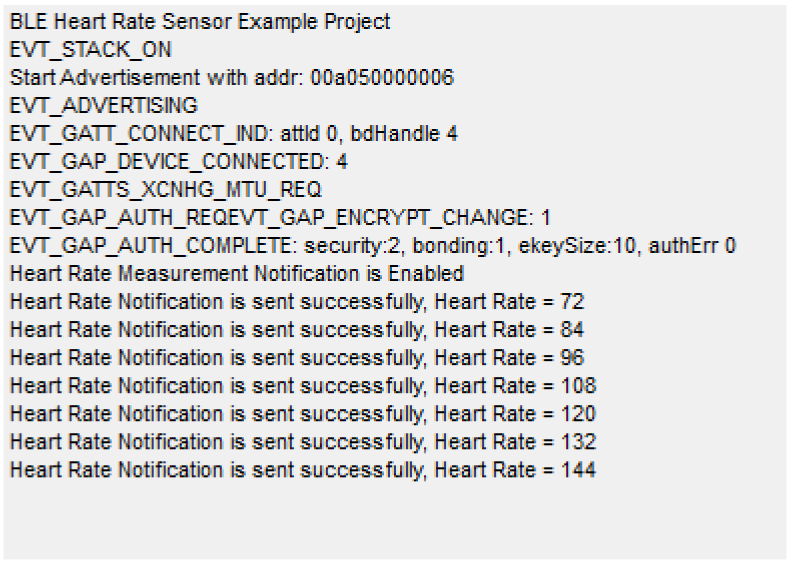


Handle	UUID	UUID Description	Value	Properties
Primary Service Declaration: Heart Rate				
0x000C	0x2800	Primary Service Declaration	0D:18 (Heart Rate)	
Characteristic Declaration: Heart Rate Measurement				
0x000D	0x2803	Characteristic Declaration	10:0E:00:37:2A	
0x000E	0x2A37	Heart Rate Measurement	14:C0:38:01:39:01:3A:01	0x10
0x000F	0x2902	Client Characteristic Configuration	01:00 <- notification is enabled	
Characteristic Declaration: Body Sensor Location				
0x0010	0x2803	Characteristic Declaration	02:11:00:38:2A	
0x0011	0x2A38	Body Sensor Location		0x02
Characteristic Declaration: Heart Rate Control Point				
0x0012	0x2803	Characteristic Declaration	08:13:00:39:2A	
0x0013	0x2A39	Heart Rate Control Point		0x08

```
[12:59:42:051]: 'Characteristic Value Notification' event received
[12:59:42:051]:     Attribute Handle: 0x000E
[12:59:42:051]:     Value: [16:9C:80:01:81:01]
[12:59:43:043]: 'Characteristic Value Notification' event received
[12:59:43:043]:     Attribute Handle: 0x000E
[12:59:43:043]:     Value: [1C:A8:70:03:65:01:66:01]
[12:59:44:041]: 'Characteristic Value Notification' event received
[12:59:44:041]:     Attribute Handle: 0x000E
[12:59:44:041]:     Value: [14:B4:4D:01:4E:01:4F:01]
[12:59:45:039]: 'Characteristic Value Notification' event received
[12:59:45:039]:     Attribute Handle: 0x000E
[12:59:45:039]:     Value: [14:C0:38:01:39:01:3A:01]
```

The details about the Heart Rate Service characteristic data structures are in the HRS Specification.

Optionally project can send log messages through UART. The example log is shown below:



```
BLE Heart Rate Sensor Example Project
EVT_STACK_ON
Start Advertisement with addr: 00a050000006
EVT_ADVERTISING
EVT_GATT_CONNECT_IND: attId 0, bdHandle 4
EVT_GAP_DEVICE_CONNECTED: 4
EVT_GATTS_XCNHG_MTU_REQ
EVT_GAP_AUTH_REQEVT_GAP_ENCRYPT_CHANGE: 1
EVT_GAP_AUTH_COMPLETE: security:2, bonding:1, ekeySize:10, authErr 0
Heart Rate Measurement Notification is Enabled
Heart Rate Notification is sent successfully, Heart Rate = 72
Heart Rate Notification is sent successfully, Heart Rate = 84
Heart Rate Notification is sent successfully, Heart Rate = 96
Heart Rate Notification is sent successfully, Heart Rate = 108
Heart Rate Notification is sent successfully, Heart Rate = 120
Heart Rate Notification is sent successfully, Heart Rate = 132
Heart Rate Notification is sent successfully, Heart Rate = 144
```