

BLE Application Notes

Version 1.1
August 2015

About this Document

This document describes the BLE commands sequence to communicate with the iOS device.

Table Of Contents

1	Sample Flow of APIs for WiFi+BT LE Co-Ex mode	4
2	BLE Demo Steps in UART Binary Mode.....	5
3	Configuring Module as a Master in AT mode	10
4	Configuring Module as a Slave in AT mode	11

1 Sample Flow of APIs for WiFi+BT LE Co-Ex mode

In order to run the Wi-Fi client and BT-LE coexistence mode, user has to issue the operating mode as first command with coex parameters. After operating mode command module will operate in both WiFi STA mode and BT LE mode. So user can issue WiFi commands as well as BT LE commands in parallel on host interface.

Common Command: Set the operating mode command with below parameters to run in Wi-Fi + BT-LE Coex Mode.

Oper_mode = ((wifi_oper_mode) | (coex_mode << 16))

Wifi_oper_mode = 0 (to operate wifi in STA mode)

Coex_mode=13(to operate in WiFi+BT LE coex mode)

Feature_bit_map = 1(to operate WiFi in open security mode) Tcp_ip_feature_bit_map = 1 (TCP/IP Bypass mode)

Custom_feature_bit_map=0

Wi-Fi Command Sequence to Associate with Access Point:

Band :- This command sets the operating mode of the module

Init :- This command initializes the module

Scan :- This command scans for Aps and reports the Aps found

Join :- This command associates the module to the AP

Please refer RS9113-Wiseconnect-Software-PRM-vx.x.x.pdf for Wi-Fi commands description.

BT LE Command Sequence:

Scan :- This command scans for BT LE devices and reports the devices found

Connect :- This command associates the module to the remote device

After successful WiFi and BT LE connection user can send WiFi raw data packets into air and also can issue GATT commands

2 BLE Demo Steps in UART Binary Mode

1. Make sure that in the rsi_config.h in following path host/binary/apis/wlan/ref_apps/include is configured for the UART interface. Also the RSI_UART_DEVICE is configured based on the tty port detected e.g. “/dev/ttyUSB0”
2. Now go to the following path:
/host/binary/coex_ref_projects/LINUX_WINDOWS/src

Note: Make sure you have the latest project in which we provide defines characteristics and services.
And issue

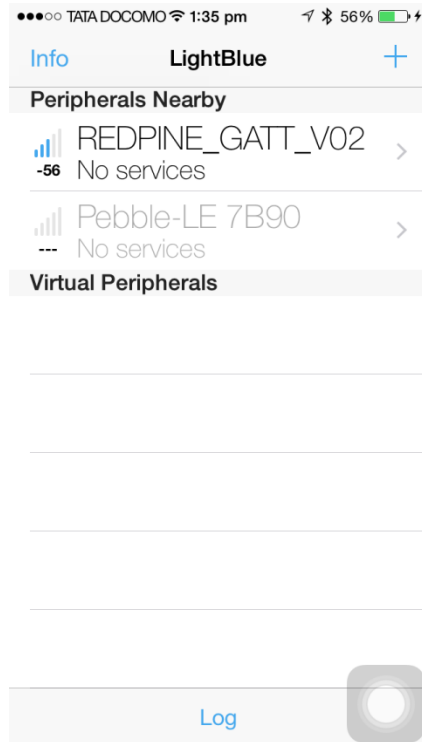
>>**make clean;make wlan_ble**

It will build the **rsi_wsc_uart_app** binary

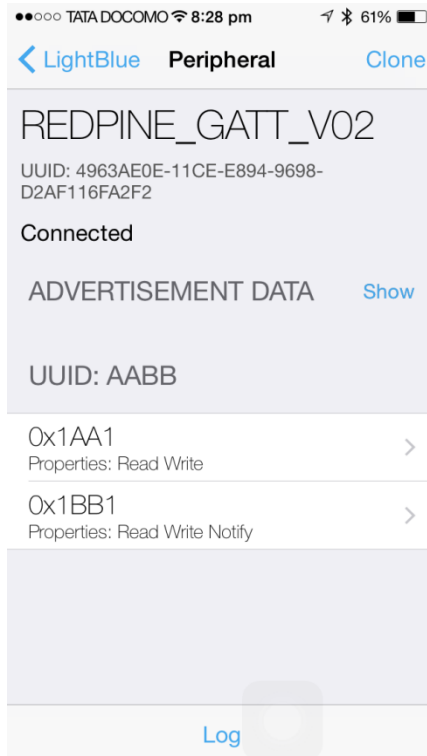
3. Once it runs it will scan or join to AP (based on the settings done in the rsi_config.h file
4. Now go to “LightBlue” iPhone app



5. Once you open the APP you can see “REDPINE_GATT_V02”



- 6. Now connect to “REDPINE_GATT_V02” . You will see a BT CONN event on the module side.
- 7. Once connected you will see as below:

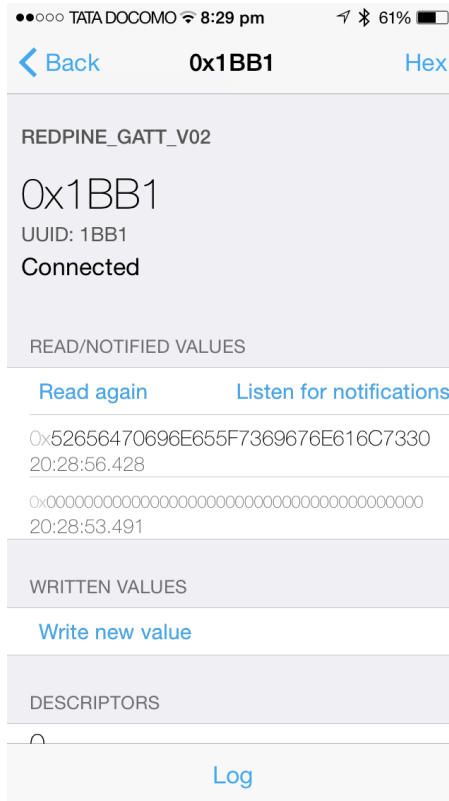


Now you need to select 1BB1 which is a Read Write Notify service.

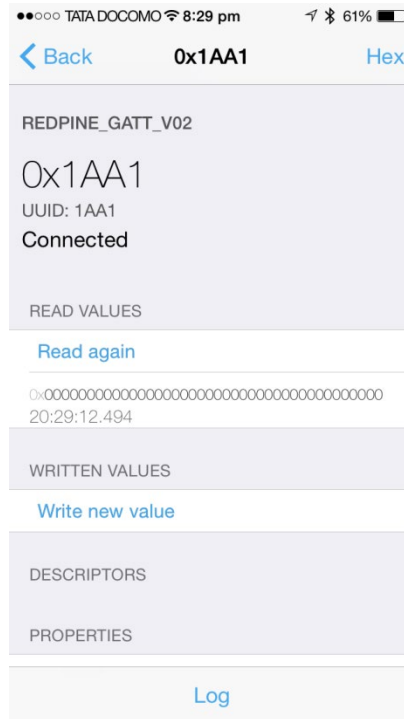
Enable the “Listen for notification”. This will show the hex stream e.g. 0x52656470696E655F7369676E616C7330

In ASCII → Redpine_signals0

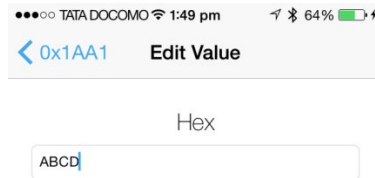
Last byte will be incremented and seen here every time we send something to module by writing in the 1AA1 service.



Now go to 1AA1 and select “write new value”



You can see below that ABCD is entered. Click done.
It will send ABCD on the other side



D	E	F
A	B	C
7	8	9
4	5	6
1	2	3
⌫	0	Done

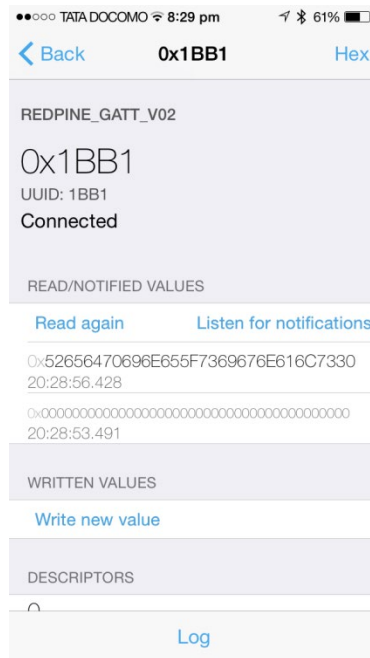
On the module side we will receive this

X Len of the packet: 76

```
0x3c 0x20 0x17 0x15 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x80 0x52 0xa5 0xd0 0x2f 0x69 0x0c 0x00 0x02 0xdc 0xba 0x00 0x00 0x00 0x00 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
```


0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
buf_ptr: 0x8c120e8
Received event type: , status:00
BDAddress:80-52-a5-d0-2f-69-
Handle:0x02c
Handle:0x02c
NumberOfValues:2
Long Value:
ab cd
CMD TYPE 342

Now module will set the local attribute value which can be seen in 1BB1



3 Configuring Module as a Master in AT mode

```
[TX] - at+rsi_opermode=851968,0,4,0<CR><LF>
```

```
[TX] - at+rsibt_getlocalname?<CR><LF>
```

```
[TX] - at+rsibt_getlocalbdaddr?<CR><LF>
```

```
[TX] - at+rsibt_addservice=2,180A,3,30<CR><LF>
```

```
[TX] - at+rsibt_addattribute=1558C,B,2,2803,2,6,8,0,0C,00,00,2A<CR><LF>
```

```
[TX] - at+rsibt_scan=1,0,0<CR><LF>
```

```
[TX] - at+rsibt_connect=0,00-23-A7-4C-24-95<CR><LF>
```

```
[TX] - at+rsibt_getdevstate?<CR><LF>
```

```
[TX] - at+rsibt_getallprofiles=00-23-A7-4C-24-95,1,10<CR><LF>
```

```
[TX] - at+rsibt_getcharservices=00-23-A7-4C-24-95,1,10<CR><LF>
```

```
[TX] - at+rsi_getlocalattvalue=A<CR><LF>
```

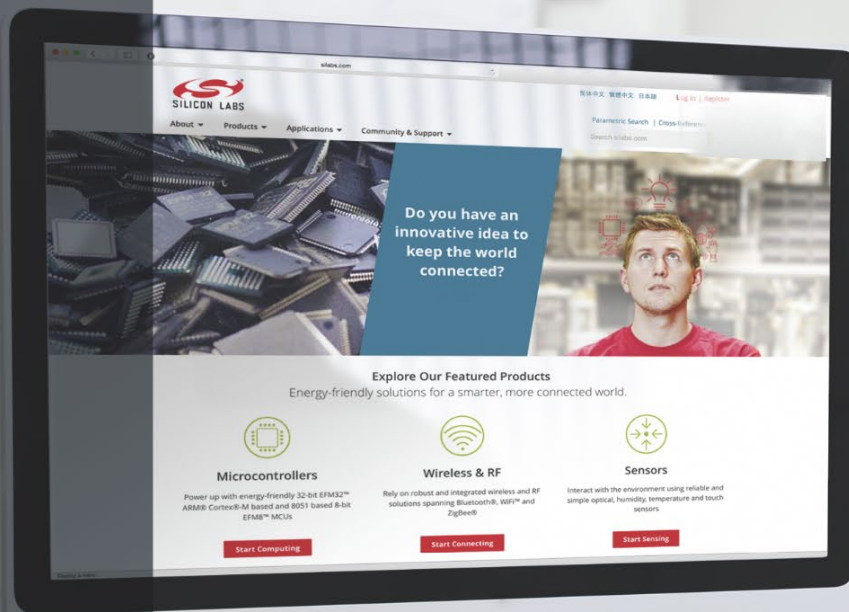
```
[TX] - at+rsibt_writevalue=00-23-A7-4C-24-95,C,A,r,e,d,p,i,n,e,l,e<CR><LF>
```

4 Configuring Module as a Slave in AT mode

```
[TX] - at+rsi_opermode=851968,0,4,0<CR><LF>
[TX] - at+rsibt_setlocalname=8,redpines<CR><LF>
[TX] - at+rsibt_getlocalbdaddr?<CR><LF>
[TX] - at+rsibt_addservice=2,180A,3,30<CR><LF>
[TX] - at+rsibt_addattribute=1558C,B,2,2803,2,6,8,0,0C,00,00,2A<CR><LF>
[TX] - at+rsibt_addattribute=1558C,C,2,2A00,8,A,gurucharan<CR><LF>
[TX] - at+rsibt_advertise=1,0,0,0,0<CR><LF>
[TX] - at+rsibt_getdevstate?<CR><LF>
[TX] - at+rsibt_getallprofiles=00-23-A7-80-70-5F,1,10<CR><LF>
[TX] - at+rsibt_getcharservices=00-23-A7-80-70-5F,1,10<CR><LF>
[TX] - at+rsi_getlocalattvalue=A<CR><LF>
[TX] - at+rsibt_writevalue=00-23-A7-80-70-5F,C,4,41,42,43,44<CR><LF>
```

Revision History

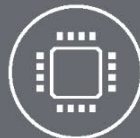
Revision No.	Version No.	Date	Author	Changes
1	1.0	Aug'15	Tarun	Initial Version



Smart.
Connected.
Energy-Friendly



Products
www.silabs.com/products



Quality
www.silabs.com/quality



Support and Community
community.silabs.com

Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

Trademark Information

Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOModem®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

<http://www.silabs.com>