BT Power save Application

User Guide
Version 0.1

November 2015

Redpine Signals, Inc.

2107 N. First Street, #680

San Jose, CA 95131.

Tel: (408) 748-3385

This project is applicable to all the WiSeConnect variants like WiSeConnect Plus, WiSeMCU and WYZBEE. The term WiSeConnect refers to its appropriate variant.

Application Overview:

This application demonstrates SPP Chat between two BT devices in Power Save.

Setup required:

1. Windows PC with Coocox IDE
2. WiSeConnect module board
3. Android Smartphone with SPP Application(Ex: SPP Pro App)

Description:

WiSeConnect module acts as a BT Slave device with SPP profile running it. Smartphone acts as a BT Master device with SPP app running in it.

Set up WiSeConnect module as a Slave and accept basic connection from the Smartphone. Now establish SPP connection from the Smartphone and exchange data.

Details of the Application:

The application (running in WiSeConnect (module) includes following steps.

1. Configure WiSeConnect module to act as Slave
2. Accept the connection(SPP) from the Smart phone
3. Loop back the received message

Configuring the Application:

- Open ***sapis/include/rsi_wlan_config.h*** file and update/modify following macros,

```
#define CONCURRENT_MODE              RSI_DISABLE
#define RSI_FEATURE_BIT_MAP          FEAT_SECURITY_OPEN
#define RSI_TCP_IP_BYPASS            RSI_DISABLE
#define RSI_TCP_IP_FEATURE_BIT_MAP TCP_IP_FEAT_DHCPV4_CLIENT
#define  RSI_CUSTOM_FEATURE_BIT_MAP  0
#define  RSI_BAND                    RSI_BAND_2P4GHZ
```

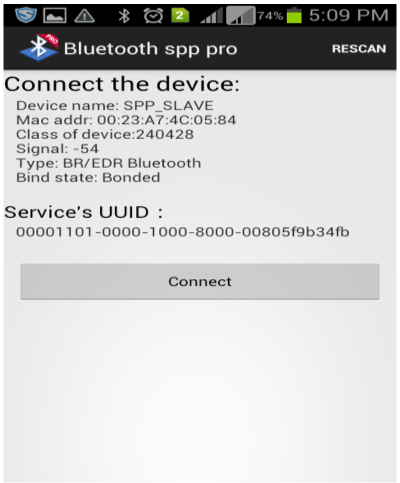
- Configure the below macros in the Application file.

1. RSI_BT_LOCAL_NAME **-** Name of the WiSeConnect BT device
2. PIN_CODE - Four byte string required for pairing process.

3. SNIFF_MAX_INTERVAL - Sniff Maximum interval value
4. SNIFF_MIN_INTERVAL – Sniff Minimum interval value
5. SNIFF_ATTEMPT - Sniff Attempt Value
6. SNIFF_TIME_OUT – Sniff Timeout Value
7. Following are the non-configurable macros in the Application file.
8. BT_GLOBAL_BUFF_LEN – Number of bytes required for the Application and the Driver.
9. RSI_APP_EVENT_CONNECTED - Event number to be set on connection establishment.
10. RSI_APP_EVENT_DISCONNECTED - Event number to be set on disconnection.
11. RSI_APP_EVENT_PINCODE_REQ - Event number to be set on Pincode request for pairing.
12. RSI_APP_EVENT_LINKKEY_SAVE - Event number to be set on link key save.
13. RSI_APP_EVENT_AUTH_COMPLT - Event number to be set on authentication complete.
14. RSI_APP_EVENT_LINKKEY_REQ - Event number to be set on link key request for connection.
15. RSI_APP_EVENT_SPP_CONN - Event number to be set on SPP connection.
16. RSI_APP_EVENT_SPP_DISCONN - Event number to be set on SPP disconnection.
17. RSI_APP_EVENT_SPP_RX - Event number to be set on SPP data received from Master.

Executing the Application:

1. Connect WiSeConnect module board to the Windows PC running Coocox IDE.
2. Configure the macros in the file located at **sapis/examples/bt/bt_power_save/rsi_bt_power_save_profile.c**
3. Compile and launch the application.
4. After the program gets executed, WiSeConnect module initializes the SPP profile and waits for the incoming connection.
5. Now initiate connection from the SPP App running in the Smartphone.

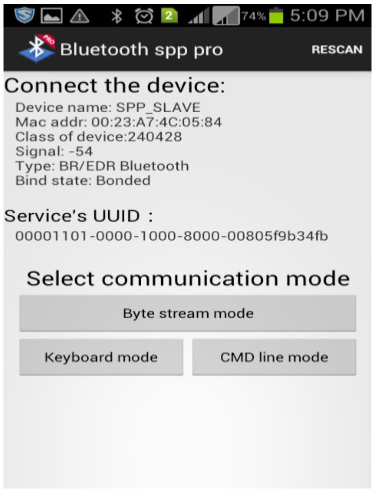


Connect the device:

Device name: SPP_SLAVE
Mac addr: 00:23:A7:4C:05:84
Class of device:240428
Signal: -54
Type: BR/EDR Bluetooth
Bind state: Bonded

Service's UUID :
00001101-0000-1000-8000-00805f9b34fb

1. After successful SPP connection, it will initiate Sniff command along with power save command.



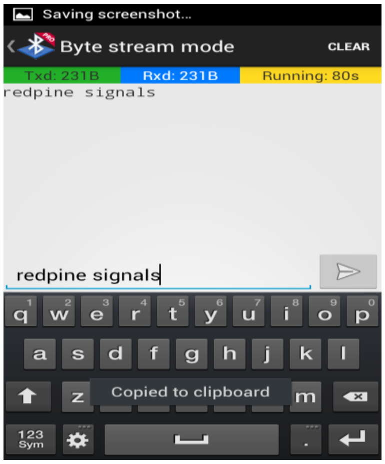
Connect the device:

Device name: SPP_SLAVE
Mac addr: 00:23:A7:4C:05:84
Class of device:240428
Signal: -54
Type: BR/EDR Bluetooth
Bind state: Bonded

Service's UUID :
00001101-0000-1000-8000-00805f9b34fb

Select communication mode

2. Send the data from the App and observe that the same data would be looped back to the App.



Txd: 231B Rxd: 231B Running: 80s

```
redpine signals
```

redpine signals