

# **SMP Application**

**User Guide**

**Version 0.2**

**May 2016**

## **Redpine Signals, Inc.**

2107 N. First Street, #540

San Jose, CA 95131.

Tel: (408) 748-3385

Fax: (408) 705-2019

Email: [info@redpinesignals.com](mailto:info@redpinesignals.com)

Website: [www.redpinesignals.com](http://www.redpinesignals.com)

---

### **About this Document**

This document describes the process of bringing up the RS9113 based module in BTLE central mode and connects with remote peripheral device with SMP.

### **Disclaimer:**

The information in this document pertains to information related to Redpine Signals, Inc. products. This information is provided as a service to our customers, and may be used for information purposes only. Redpine assumes no liabilities or responsibilities for errors or omissions in this document. This document may be changed at any time at Redpine's sole discretion without any prior notice to anyone. Redpine is not committed to updating this document in the future.

Copyright © 2015 Redpine Signals, Inc. All rights reserved.

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>4</b>
<b>1.1</b>	<b>Application Overview .....</b>	<b>4</b>
1.1.1	Overview .....	4
1.1.2	Sequence of Events .....	4
<b>1.2</b>	<b>Application Setup .....</b>	<b>4</b>
1.2.1	SPI based Setup Requirements .....	4
1.2.2	UART/USB-CDC based Setup Requirements .....	4
<b>2</b>	<b>Configuration and Execution of the Application .....</b>	<b>6</b>
<b>2.1</b>	<b>Initializing the Application .....</b>	<b>6</b>
2.1.1	SPI Interface .....	6
2.1.2	UART/USB-CDC Interface .....	6
<b>2.2</b>	<b>Configuring the Application .....</b>	<b>6</b>
<b>2.3</b>	<b>Executing the Application .....</b>	<b>7</b>

## Table of Figures

<b>Figure 1: Setup Diagram .....</b>	<b>5</b>
--------------------------------------	----------

## Table of Tables

No table of figures entries found.

## 1 Introduction

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

### 1.1 Application Overview

#### 1.1.1 Overview

This application demonstrates how to configure the WiSeConnect device in Central mode and connects with remote slave device and how to enable SMP pairing.

In this application, WiSeConnect module connects with peripheral device and initiates SMP pairing process. After successful SMP pairing, SMP encryption will be enabled in both Central and Peripheral device.

#### 1.1.2 Sequence of Events

This Application explains user how to:

- Configure device in Central mode
- Connect with remote BTLE peripheral device.
- Initiate SMP pairing with remote connected peripheral device.
- Initiate SMP pair response for the received SMP response event.
- Send SMP passkey for the received SMP passkey request.

### 1.2 Application Setup

The WiSeConnect in its many variants supports SPI and UART interfaces. Depending on the interface used, the required set up is as below:

#### 1.2.1 SPI based Setup Requirements

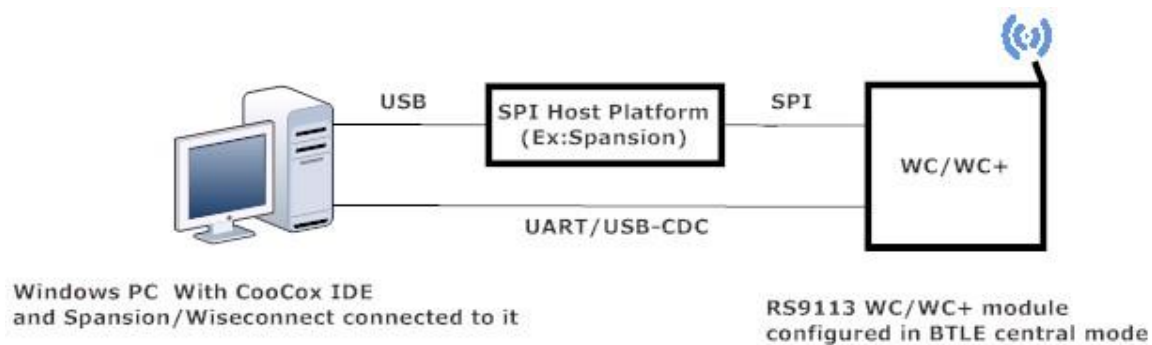
- Windows PC with Coocox IDE
- Spansion (MB9BF568NBGL) micro controller

**Note:** If user does not have Spansion (MB9BF568NBGL) host platform, please go through the SPI-Porting guide [\sapis\docs\RS9113-WiSeConnect-SAPI-Porting-Guide-vx.x.pdf](#) for SAPIs porting to that particular platform.

- WiSeConnect device
- BTLE peripheral device which supports SMP pairing(This Application uses TI sensor tag for remote device)

#### 1.2.2 UART/USB-CDC based Setup Requirements

- Windows PC with Dev-C++ IDE
- WiSeConnect device
- BTLE peripheral device which supports SMP pairing(This Application uses TI sensor tag for remote device)



**Figure 1: Setup Diagram**

## 2 Configuration and Execution of the Application

The example application is available in the Release at {Release \$}/host/sapis/examples. These examples will have to be initialized, configured and executed to test the application. The initialization varies based on the interface but configuration and execution are the common.

### 2.1 Initializing the Application

#### 2.1.1 SPI Interface

If User using SPI interface, Please refer the document *sapis/platforms/spansion\_MB9BF568NBGL/RS9113-WiSeConnect\_SAPIS\_Spansion\_Project\_User\_guide.pdf* for opening the *simple\_smp* example in CoCoX IDE.

#### 2.1.2 UART/USB-CDC Interface

If User using UART interface, Please refer the document *sapis/platforms/windows\_uart/RS9113-WiSeConnect\_SAPIS\_Windows\_Project\_UserGuide.pdf* for opening the *simple\_smp* example in Dev-C++ IDE

### 2.2 Configuring the Application

1. Open *sapis/examples/ble/simple\_smp/rsi\_ble\_smp.c* file and update/modify following macros,

**RSI\_BLE\_REMOTE\_ADDR** refers address of the remote device to connect.

```
#define RSI_BLE_REMOTE_ADDR "00:1A:7D:DA:71:13"
```

**RSI\_BLE\_DEVICE\_NAME** refers the name of the WiSeConnect device to appear during scanning by remote devices.

```
#define RSI_BLE_DEVICE_NAME "WLAN_BLE_SIMPLE"
```

**RSI\_BLE\_SMP\_IO\_CAPABILITY** refers IO capability.

```
#define RSI_BLE_SMP_IO_CAPABILITY 0x00
```

**RSI\_BLE\_SMP\_PASSKEY** refers address type of the remote device to connect.

```
#define RSI_BLE_SMP_PASSKEY 0
```

Following are the non-configurable macros in the application.

```
#define RSI_BLE_CONN_EVENT 0x01
#define RSI_BLE_DISCONN_EVENT 0x02
#define RSI_BLE_SMP_REQ_EVENT 0x03
#define RSI_BLE_SMP_RESP_EVENT 0x04
#define RSI_BLE_SMP_PASSKEY_EVENT 0x05
#define RSI_BLE_SMP_FAILED_EVENT 0x06
#define RSI_BLE_ENCRYPT_STARTED_EVENT 0x07
```

**BT\_GLOBAL\_BUFF\_LEN** refers Number of bytes required by the application and the driver

```
#define BT_GLOBAL_BUFF_LEN 10000
```

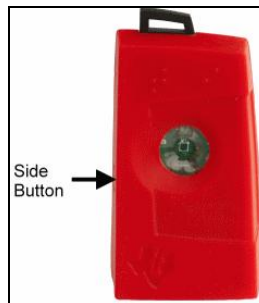
2. Open *sapis/include/rsi\_wlan\_config.h* file and update/modify following macros,

```
#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
```

## 2.3 Executing the Application

1. Configure the remote TI sensor tag in peripheral mode and put it in advertising mode.

Press the side button to configure TI sensor tag in advertising mode



### 2. SPI Interface

If User using SPI interface, Please refer the document *sapis/platforms/spansion\_MB9BF568NBGL/RS9113-WiSeConnect\_SAPIS\_Spansion\_Project\_User\_guide.pdf* for executing the *simple\_smp* example in CooCox IDE.

### 3. UART/USB-CDC Interface

If User using UART interface, Please refer the document *sapis/platforms/windows\_uart/RS9113-WiSeConnect\_SAPIS\_Windows\_Project\_UserGuide.pdf* for executing the *simple\_smp* example in Dev-C++ IDE

4. After the program gets executed, WiSeConnect device tries to connect with the remote device specified in **RSI\_BLE\_REMOTE\_ADDR** macro.
5. Observe that the connection is established between the desired device and WiSeConnect device.
6. After successful connection, application will initiate SMP paring and wait for SMP response event and SMP passkey request event. After receiving SMP response and SMP passkey events,

---

application sends SMP response and SMP Passkey responses. If SMP success, Device sends SMP encrypt started event to host. If not success, Device sends SMP failure event to host.