ANKHMAWAY-OFA (Open Firmware API)

Ultra-Slim Beacon

User Guide V1.0

Features

- - Ultra-slim Beacon
- - iBeacon
- - Eddystone
- - Custom Beacon
- - Support BLE 4.0 protocol.
- - Double Protocols
- - Broadcast by Shake
- - Press the button to broadcast
- - Intelligent power management
- - Acceleration Sensor
- - Support OAD
- - 90dB buzzer
- - Optimal size
- - Waterproof
- - Multi-Function

Support

- eBeacon
- iOS SDK
- Open source code for Android

Description

For Ultra Slim Beacon, developers can use its SDK build their own APP. It has several important features and can be used in every aspect of our life. Through this document, it will share to developers to build various apps.

The most important news is firmware interface ports are completely open, which means there is no limit for developing.

Ultra-slim Beacon can be customized, and if you need this service, please do contact us. We will try our best to help you realize it. If any suggestion, that will be appreciated. We will strive to build a Beacon World.

Contents

List of tables

Introduction

Ultra-slim Beacon, with thickness of 2.6mm, is convenient to put in or stick on daily supplies. We add a button and accelerometer in Ultra-slim Beacon, and make it different from other models (iB001N, iB004N...). We add three types of broadcast and will explain in details.

1. Broadcast when press the button

We add this new type of broadcast, which enables Ultra-slim Beacon can be used as a remote control. It provide the function that when you press the button, the smart phone will alert. Using the button to make smart phone sound, so Ultra- slim Beacon can be used to your smart phone. Besides, Ultra-slim Beacon can be used in many situations. For example, press the button to turn on/off lights; press the button to open the garage door when you drive home; if you want to turn on/off the TV, just press the button; if you want to turn on the air conditioner, just press the button. There is no need to find the remote control, and the air conditioner and so on. This button makes you control everything around you.

2. Broadcast when Ultra-slim Beacon is in motion

This type of broadcast means Ultra-slim Beacon will broadcast only when it is in motion. Once motion stops, Ultra-slim Beacon will stop broadcasting. So, this enables Ultra-slim Beacon to detect the movements of other objects. For example, stick Ultra-slim Beacon on the window, if the customer get in the store, the shopkeeper will get the message and will service the customer. Place Ultra-slim Beacon on every table, when customers finish the dish, they just shake Ultra-slim Beacon and the boss will come over to settle the bills. And we call this type of broadcast as Shake. Shake makes you know all info around you.

3. Keep broadcasting with multi-channels

Similar to our former beacons, Ultra-slim Beacon can be used as a tag. We offer two channels for Ultra-slim Beacon to broadcast two types of data. The first channel is only for iBeacon data and goes to www.apple.com to get more info. The second is a user-defined channel. Users can choose the second channel to broadcast content that they need, such as Eddystone data (Eddystone-UID, Eddystone-URL). It supports users to broadcast other types of data. In addition, we set a limit on the broadcast rate, so that Ultra-slim Beacon can be used for longer even years. Multi-channels design enables Ultra-slim Beacon to simulcast the data in these two channels, such as iBeacon data and Eddystone data.

Product Details

Ultra-slim Beacon works as an iBeacon and it accords iBeacon protocol. Users can reset all the attributes, including Proximity UUID,Major,Minor,Measured Power Value. And Broadcast Rate,Tx Power, Device Name is configurable.

Eddystone

Eddystone is an open source technology put forward by Google in 2015, and there are three data types in Eddystone, Eddystone-UID, Eddystone-URL, Eddystone-TLM. Ultra-slim Beacon supports Eddystone-UID and Eddystone-URL.

Eddystone-UID

The Eddystone-UID frame broadcasts an opaque, unique 16-byte Beacon ID composed of a 10-byte name space and a 6-byte instance. The Beacon ID may be useful in mapping a device to a record in external storage. The namespace portion of the ID may be used to group a particular set of beacons, while the instance ID identifies individual devices in the group. The division of the ID into namespace and instance components may also be used to optimize BLE scanning strategies, e.g. by filtering only on the namespace.

Eddystone-URL

The Eddystone-URL frame broadcasts a URL using a compressed encoding format in order to fit more within the limited advertisement packet.
Once decoded, the URL can be used by any client with access to the internet. For example, if an Eddystone-URL beacon were to broadcast the URL https://www.ankhmaway.com.cn/, then any client that received this packet could choose to visit that url.
You can download eBeacon from App Store to scan Eddystone data.

Use Ultra-slim Beacon as a remote control

We add a multi-function button in Ultra-slim Beacon to achieve remote control. Double-click the button, Ultra-slim Beacon will send data to other terminals, like a phone. The phone can handle this data for users. Ultra-slim Beacon will control your phone just like a TV remote controls the TV.

Only three steps are needed to configure Ultra-slim Beacon as a remote control

1. Configure the data that you want to send in iBeacon or user-defined channel.
2. Enable Ultra-slim Beacon to send data while button is being double-pressed.
3. Handle this message on your phone.

Shake-Shake Ultra-slim Beacon

In life we are constantly moving, never stop, shake function is to detect the state of the world. Shake Ultra-slim Beacon; it will broadcast the good data set by developer. When phone receives this command, phone will perform the appropriate action. Only three simple steps are needed to enable shake function.

1. Configure data in iBeacon and user-defined channel.
2. Enable shake state
3. Get the corresponding data on phone or other terminals.

Use Ultra-slim Beacon as a tag beacon

Ultra-slim Beacon has a tiny figure, which not only makes it more beautiful, but also makes it almost invisible in this world. You can stick it on any article or place it in any location; it will mark location for you. Ultra-slim Beacon, used as a tag, supports iBeacon and Eddystone. Ultra-slim Beacon marks each location for you.

Acceleration Sensor

Acceleration is a bright spot of Ultra-slim Beacon. It makes it can detect motion of objects. We also spent a lot of time on the acceleration sensor so it works with very low power consumption. This makes Ultra-slim Beacon work for a long time even with the acceleration sensor.

90dB buzzer

A 90dB buzzer is integrated in Ultra-slim Beacon; the maximum value can reach 100dB. This is a huge challenge for ultra-slim beacon in order to make it come true, we spent a lot of energy on this. High-decibel buzzer allows it play a huge potential as a tracker.

Waterproof

Ultra-slim Beacon can be operated in anywhere on normal use, even in wet places. Thus, customers are no longer worried about the Beacon being damped or soaked.
Great waterproof performance let Ultra-slim Beacon can be used to every aspect of life.

Advertisement Control

We use service 0xFF70/Characteristic 0x2A80 to control the advertisement of Ultra-slim Beacon. The following table is the relationship between values and the types of advertisement. Users only need to re-write this characteristic and change the mode of Ultra-slim Beacon.

Table1. Beacon State

0x01	Factory Mode, in a sleep state
0x02	Always broadcast iBeacondata
0x04	Broadcast iBeacon data when motion triggered.
0x08	Always broadcast data in user-defined channel
0x10	Broadcast data in user-defined channel when motion triggered.
0x80	Connectable when broadcast

All of these values can be combined, please refer to the following example for configuration.

1. If the user needs to have broadcast data in iBeacon channel data and broadcast data in user-defined channel when motion triggered at same time, the value should be rewritten to (0x02|0x10)=0x12. If beacons need to be connectable, then the value is (0x12| 0x80)=0x92.

2. If the user needs to broadcast data in iBeacon channel and broadcast data in user-defined channel when motion triggered at same time, the value should be rewritten as (0x04|0x10)=0x14, If beacons needs to be connectable, the value is (0x14| 0x80)=0x94
3. If the user wants to keep broadcasting data in iBeacon channel and user-defined channel, value should be (0x04| 0x08)=0x0C. If beacons need to be connectable, the value is 0x0C| 0x80 =0x8C.

Broadcast data when the button is double-clicked

We use service 0xFF70 and Characteristic 0x2A84 to control the advertisement of Ultra-slim Beacon. The following table is the relationship between values and the type of advertisement. Users only need to re-write the value and change the mode.

Table2. Button-Adv State

0x00	Disable
0x01	Broadcast iBeacon channel data, and keep 5 seconds
0x02	Broadcast custom channel data, and keep 5seconds

Note: If the value of this characteristic is configured to 0x01, Ultra-slim Beacon will broadcast iBeacon channel data when double click the bulit-in button, and the radio will automatically stop after 5seconds. If you do not want to broadcast the data when you double-click the button, then the value of the characteristic should be changed to 0x00.

Service 0xFF70: Control Ultra-slim Beacon Adv State

We use service 0xFF70 to control the advertisement state of Ultra-slim Beacon

Characteristic 0x2A80

This characteristic is used to control the state of Ultra-slim Beacon, as well as the working mode of the configuration of motion sensing, the length of this characteristic value is one byte (8bits). We use each bit to control different functions. Please refer to table1

Table 3. Register Beacon State

Connectable_en	…	…	Custom_Channel_Trigger_en	Custom_Channel_Always_en	iBeacon_Trigger_en	iBeacon_Always_en	Disable

Beacon State Register Description

Table 4. Beacon State Register Description

Connectable_en	Control whether the Ultra-slim Beacon can be connected when advertisement Default value: 0 (0:Non-Connectable; 1: Connectable)

Custom_Channel_Trigger_en	Advertise custom channel data when motion triggered Default value: 0 (0: Disable; 1: Enable)
Custom_Channel_Always_en	Always broadcast data in user-defined channel Default value: 0 (0: Disable; 1: Enable)
iBeacon_Trigger_en	Advertise data in iBeacon channel when motion triggered Defaultvalue: 0 (0: Disable; 1: Enable)
iBeacon_Always_en	Always broadcast data in iBeacon channel Defaultvalue: 0 (0: Disable; 1: Enable)
Beacon_Disable	Factory state Default value:1 (0:Actived; 1: Factory)

Characteristic 0x2A81

This characteristic is used to control the broadcast rate when Ultra-slim Beacon is in always broadcast mode. The value can be dynamically adjusted, and this value is not effective in trigger mode, in other words, we use a fixed broadcast frequency to broadcast trigger data and user can not adjust it.

The unit of this value is 100ms, the user can set the value from 10 to 100, that is, the broadcast frequency is 1s to10s

Characteristic 0x2A83

This characteristic is used to adjust the intensity of the motion detection; the value of intensity is characteristic value multiplied by the unit.

The unit of this value is 63mg (note 1G equals 1000mg, where G is the acceleration of gravity). For example, the value is set to 0x01(=1), then the value is over 63mg, motion-detection works. The range of this value can be set to 1-127.

Characteristic 0x2A84

This characteristic is used to configure the Ultra-slim Beacon button state. Please refer to table5 to set this value

Table 5. Button-Adv State

0x00	Disable (does not broadcast when double-click the button)
0x01	Broadcast data in iBeacon channel when double-click the button,and keep 5seconds
0x02	Broadcast data in user-defined channel when double-click the button, and keep 5s

Service 0xFFF0: Configure iBeacon data

We use service 0xFF70 to configure iBeacon parameters. Please check the below for details.

Characteristic0xFFF1

This characteristic is used to verify the authority of the users. When connected to Ultra-slim Beacon, user must write the correct password to this characteristic within a minute, otherwise, it will disconnect with phone. The default password is “0x666666”.When you use eBeacon to connect Ultra-slim Beacon, just input“666666”.

Characteristic 0xFFF6

This characteristic is used to configure broadcast interval of Ultra-slim Beacon. It is samewithservice0xFF70 characteristic0x2A81. You can use one of them to configure it.

Characteristic 0xFFF8

This characteristic is used to change the connect password of Ultra-slim Beacon, the length of this characteristic is3 bytes. When you are trying to write this characteristic, you must write three bytes. Once you write success, the connect password is changed to the new one.

Characteristic 0xFFF2, 0xFFF3, 0xFFF4, 0xFFF5, 0xFFF7

These four characteristic are used to configure all iBeacon parameters.

Table 6. Characteristics of iBeacon channel configuration

Characteristic	Property	Value Length	Function	Write example
0xFFF2	Read/Write	16Bytes	Configure Proximity UUID	0xEBEFD08370A247C89837E7B5634DF524
0xFFF3	Read/Write	2Bytes	Configure Major Value	0x0001
0xFFF4	Read/Write	2Bytes	Configure Minor Value	0x0001
0xFFF5	Read/Write	1Byte	Configure Measured Power Value	0xCB
0xFFF7	Read/Write	2Bytes	Configure Mfgr Value	0x004C (iBeacon requirement)

Service0xFFD0: Configure user-defined data

We use service 0xFFD0 to configure data in user-defined channel, the length of this characteristic is 28bytes, please refer to BLE4.0 protocol to get the details.

Characteristic 0xFFD1
Configure the former 20 bytes of this channel data

Characteristic 0xFFD2
Configure the end 8 bytes of this channel data.

Characteristic 0xFFD3
This characteristic value is the valid length of this channel data. Users can only read this characteristic value to verify the configure state.

Example 1: Configure this service's data to Eddystone-UID.

Namespace ID: 0xEBEFD08370A247C89837.
Instance ID: 0xE7B5634DF524.
So all data is 0x0303AAFE1516AAFE00CBEBEFD08370A247C89837E7B5634DF524.
Firstly, write 0x0303AAFE1516AAFE00CBEBEFD08370A247C89837 to characteristic 0xFFD1.
Secondly, write 0xE7B5634DF524 to characteristic 0xFFD2.
Thirdly, check whether the value of characteristic 0xFFD3 is 26.
If so, the configuration is finished.

Example 2: Configure this service's data to Eddystone-URL.

URL: www.ankhmaway.com.cn
So all data is 0x0303AAFE1616AAFE10357777772E616E6B686D61.

Firstly write 0x0303AAFE1616AAFE10357777772E616E6B686D61 to characteristic 0xFFD1.
Secondly, write 0x776179072E636E00 to characteristic 0xFFD2.
Thirdly, check whether the value of characteristic 0xFFD3 is 27.
If so, the configuration is finished.

Service 0xAA10: Motion & Button Notify service

We use service 0xAA10 to notify the motion & button state of Ultra-slim Beacon to the phone.

Characteristic 0xAA12
This characteristic is used to notify the motion state. When motion entered, the phone will receive the data 0x01.

Characteristic 0xAA16
This characteristic is used to notify the button state. When button is pressed, the phone will receive the data of the button, please refer to below,

Table 7. Define of button press state

0x01	Button is being pressed
0x03	Button is being long-pressed

Service 0x1804: Configure Tx Power of Ultra-slim Beacon

We use service 0x1804 to configure the TX power of Ultra-slim Beacon, please take attention, the TX Power is not equal to Measured Power Value. Please refer to below.

Table 8. The value define of Tx Power setting

0x01	+4 dBm(Max)
0x02	0 dBm
0x03	-4 dBm
0x04	-8 dBm
0x05	-12 dBm
0x06	-16 dBm
0x07	-20 dBm
0x08	-30 dBm
0x09	-40 dBm(Min)

Service 0x1802: Call Ultra-slim Beacon

We use service 0x1802 to control the audio system. When user writes 0x01 to characteristic 0x2A06, the Ultra-slim Beacon will make a buzzer and keep 3 seconds. Once time out, the audio system will close automatically.

Service 0xFF60: Configure audio state

We use service 0xFF60 to configure the audio state of Ultra-slim Beacon, please check the below to get the value's define of characteristic 0x2A70.

Table 9. Characteristics of service 0xFF60

Characteristic	Property	Value Length	Function	Write example
0x2A70	Read/Write	1Byte	Control Beacon Audio State	0x01

Notes: When the value is 0x01, means allowing the Beacon buzzing when wake-up or reset.

0x01: Buzzing when restart and wake-up(default state).

0x02: Buzzing when restart, not when wake-up.

0x03: Buzzing when wake-up, not when restart.

0x04: No buzzing when restart or wake-up.

Service0xFF80: Configure the name of Ultra-slim Beacon

We use service 0xFF80 to configure the device name of Ultra-slim Beacon, you can change the name to a new one which length if less than 15 bytes. Please change the string to ASCII code. For example, “jaalee” is equal to {0x6a, 0x61, 0x61, 0x6c,0x65, 0x65}

Service 0x180F: Battery level

Using this service, users can get the battery level left of Ultra-slim Beacon. Just read the characteristic 0x2A19 to get the percent of battery level. If the value if 0x64, it means the battery level is 100%.

Service 0x180A: The device information

Using this service, you can read the info of Ultra-slim Beacon, please check the below.

Table 10. Characteristics of device information

Characteristic	Property	Function
0x2A29	Read	Manufacturer info
0x2A26	Read	Firmware version
0x2A24	Read	Serial number
0x2A23	Read	The unique ID of Ultra-slim Beacon

Default Setting

-iBeacon Proximity UUID: EBEFD083-70A2-47C8-9837-E7B5634DF524
-iBeacon Major: 0x0001
-iBeacon Minor: 0x0001
-iBeacon Measured Power Value: 0xCB
-Ultra-slim Beacon Broadcast Interval: 1Hz
-User-defined channel data: Eddystone-UID (namespace ID: EBEFD08370A247C89837, instance ID：E7B5634DF524)

eBeacon Tool

eBeacon tool is an application provide by Ankhmaway. It can be used to configure all the Beacons. Developers can use this tool to reconfigure and test Beacons. Please refer to below to start using this application.

Step 1: Download the latest version eBeacon application (version2.5.8) from apple store

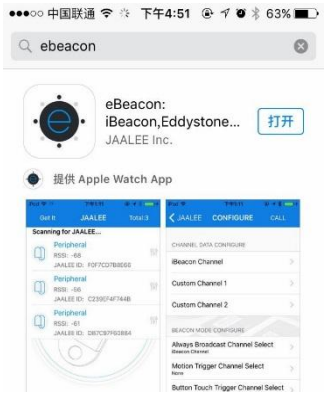
Step 2: Open the Bluetooth and eBeacon application.

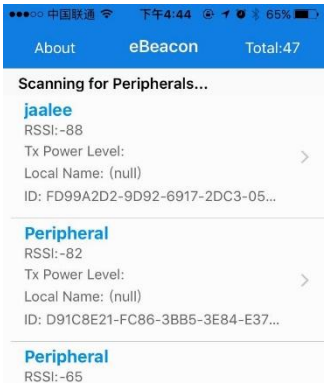
Step 3: Long-press build-in button change Ultra-slim Beacon to configuration mode.

Step 4: Click on the Beacon in the eBeacon application that need to configure, and enter the password to

connect.

Step 5: Check the user guide of the service define to reconfigure Ultra-slim Beacon





Working Mode Detail

There are three types working mode of Ultra-slim Beacon. They are Sleep Mode, Configuration Mode, and Normal Working Mode. When user long press the button of Ultra-slim Beacon until 5 second, Ultra-slim Beacon will enter the Configuration Mode, which can last for 15 seconds. After timeout, Ultra-slim Beacon will enter the Normal Working Mode or Sleep Mode according to the different configuration. When the connection is disconnected, Ultra-slim Beacon will enter Configuration Mode and this will last for 15 seconds. Then Ultra-slim Beacon can switch to Sleep Mode or Configuration Mode.

Sleep Mode

When Ultra-slim Beacon stays asleep, this mode will wait for being wake-up.

Configuration Mode

On Configuration Mode, Ultra-slim Beacon can be quickly connected and reconfigured, which needs long press the button of Ultra-slim Beacon, and it will last for 15 seconds. It means that Ultra-slim Beacon can be connected in 15 seconds. And it will restore to Sleep Mode or Normal Working Mode if it is not connected in 15 seconds.

Normal Working Mode

After set the Beacon State on Broadcast Mode or Trigger Mode, beacon is on the Normal Working Mode. On this mode, beacon can broadcast the corresponding data according to the Beacon State. If the broadcasting data in the Non-Connectable Mode, Ultra-slim Beacon need long press the button to change the state to Configuration Mode. If the configuration of Ultra-slim Beacon is always in Connectable Mode and Ultra-slim Beacon keeps broadcasting, you can connect Ultra-slim Beacon without long-press the button.

Services Integrate

Service 0xFFF0

Table 11. Characteristics of service 0xFFF0

Characteristic	Property	Value Length	Function	Write example
0xFFF1	Read/Write	2Bytes	Write Connect Password	0x666666
0xFFF2	Read/Write	16Bytes	Configure Proximity UUID of iBeacon channel	0xEBEFD08370A247C89837E7B
0xFFF3	Read/Write	2Bytes	Configure Major Value of iBeacon channel	0x0001
0xFFF4	Read/Write	2Bytes	Configure Minor Value of iBeacon channel	0x0001
0xFFF5	Read/Write	1Byte	Configure Measured Power Value of iBeacon channel	0xCB
0xFFF6	Read/Write	3Bytes	Configure broadcast Interval Of Ultra-slim Beacon	0x0A
0xFFF7	Read/Write	2Bytes	Configure Mfgr of iBeacon channel	0x0059
0xFFF8	Read/Write	3Bytes	Change Connect Password	0x123456

Note: The default configuration password is 0x666666, broadcast interval in units is 100 milliseconds, 0x0A is equivalent to 10*100(ms)=1(s). When connected to Ultra-slim Beacon, the user must input the password to Characteristic 0xFFF1 within 1 minute, otherwise the Beacon will require disconnected automatically.

Service 0xFFD0

Table 12. Characteristics of service 0xFFD0

Characteristic	Property	Value Length	Function	Write example
0xFFD1	Read/Write	20Bytes	the former 20bytes of user-define channel data	0x0303d8fe0d16d8fe0021006a61616c65
0xFFD2	Read/Write	8Bytes	The latter 8 bytes of user-define channel data	0x0000000000000000
0xFFD3	Read	1Bytes	the length of user-define channel data	0x12

Note: the data of user-define channel is 28bytes at the most. If the data is less than 20bytes, user only needs to write the data in Characteristic 0xFFD1. If not, user needs to write the former 20bytes of the data in Characteristic 0xFFD1 and write the latter bytes in Characteristic 0xFFD2.

Service 0xAA10

Table 13. Characteristics of service 0xAA10

Characteristic	Property	Value Length	Function	Write/Read example
0xAA12	Read/Notify	1Byte	Detect motion	0x01
0xAA16	Read/Write	1Byte	Button press state	0x01

Service 0xFF80

Table 14. Characteristics of service 0xFF80

Characteristic	Property	Value Length	Function	Write example
0x2A90	Read/Write	18Bytes	Configure Device Name	0x6A61616C6565

Note: The value needs to be input the port should be converted to hexadecimal ASCII characters. For example, the corresponding hexadecimal ASCII for jaalee is {0x6A, 0x61, 0x61, 0x6C, 0x65, 0x65}, then the value should be input is 0x6A61616C6565.

Service 0x1804

Table 15. Characteristics of service 0x1804

Characteristic	Property	Value Length	Function	Write example
0x2A07	Read/Write	1Byte	Configure TX Power	0x01

Note: The relationship between the value input and its corresponding TX Power is as the following table.

Table 16. TX Power Settings

Write Value	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09
TX Power (dBm)	4	0	-4	-8	-12	-16	-20	-30	-40

Service 0xFF70

Table 17. Characteristics of service 0xFF70

Characteristic	Property	Value Length	Function	Write example
0x2A80	Read/Write	1Byte	Configure Beacon State	0x01:Disable
0x2A81	Read/Write	1Byte	Configure the broadcast rate	0x0A
0x2A83	Read/Write	1Byte	Configuring the intensity value of the motion detection	0x09
0x2A84	Read/Write	1Byte	button-adv state	0x01

Note: The broadcast rate is same with the values under 0xFFF0->0xFFF6.

Service0xFF60

Table 18. Characteristics of service 0xFF60

Characteristic	Property	Value Length	Function	Write example
0x2A70	Read/Write	1Byte	Control Beacon Audio State	0x01

Notes: When the value is 0x01, means allowing the Beacon buzzing when wake-up or reset.

0x01: Buzzing when restart and wake-up(default state).

0x02: Buzzing when restart, not when wake-up.

0x03: Buzzing when wake-up, not when restart.

0x04: No buzzing when restart or wake-up.

Service0x1802

Table 19. Characteristics of service 0x1802

Characteristic	Property	Value Length	Function	Write example
0x2A06	Read/Write	1Byte	Call Ultra-slim Beacon	0x01

Note: When put in value 0x01 to this port, it buzzing.

Service 0x180F

Table 20. Characteristics of service 0x180F

Characteristic	Property	Value Length	Function	Read Value
0x2A19	Read/Notify	1Byte	Battery level	0x64

Note: Read battery level of Ultra-slim Beacon

Key Words

Measured Power Value: Apple device use the measured Power value of a Proximity Beacon to help with ranging accuracy.

TX Power: This value represents the Beacon's firmware Transmit Power.

RSSI: The value is the signal strength of the scanned device which can be used to measure the distance.

Electronic Parameters

Table 21. Electronic Parameters

Item	Test Data	Remarks
Chip model	nRF51822	Nordic Semiconductor 256k
Battery model	CR2477	Coinbattery,3.0Vdc,1pc
Operation Voltage	1.8-3.6V	DC
Operation Frequency	2400-2483.5MHz	Programmable
Frequency Error	+/-20KHz	Null
Modulation	Q-QPSK	Null
Sleep current	About3.6uA	Null
Output Power	4dBm--40dBm	Programmable
Receiving Sensitivity	-93dBm	High gain mode
Transmission distance	70meters	BER<0.1%,Openspace
Antenna	50ohm	Onboard
Size	42.3mmx42.3mmx13.4mm	Null

Operation State and Power Consumption

Table 22. Operation State and Power Consumption

State	Operation Current(μA)
Sleep Mode	About3.6
Configurable Mode	About142
Trigger Mode for iBeacon Data	About10
Broadcast Mode for iBeacon Data (Default Setting)	Abuout22.5
Trigger Mode for UriBeacon Data	About10
Broadcast Mode for UriBeacon Data(Default Setting)	About14.6
Trigger Mode for AccBeacon Data	About10
Broadcast Mode for AccBeacon Data(Default Setting)	About446

Contact Us

Shenzhen AnkhMaway Electronics Technology Co., Ltd.

www.ankhmaway.com.cn

kylie@ankhmaway.com
vivien@ankhmaway.com

Tel: +86 755 2834 1116
Mob: +86 189 4241 8755

Address:
3rd Floor, Building 4, Liangtang Industrial Area, Nanwan Street, Longgang, Shenzhen, China 518000

Revisionhistory

Date	Version	Description
November 2015	1.0	First release