MOKO SMART

# User Guide

## MOKO Button APP

*Version 1.2*

# Revision History

| Version | Data | Notes | Contributor(s) |
|---|---|---|---|
| V1.0 | Mar 30, 2022 | Initial version | Daniel |
| V1.1 | Jan 24, 2025 | 1. Added support for configuring standard ibeacon and Eddystone-UID broadcasts<br>2. Added support for getting button alarm events in the connected state.<br>3. Added support for configuring button trigger alert parameters in the connected state. | Leo |

# About this User Guide

This User Guide was designed to help users to know the overview of ***MOKO Button*** product and set up ***MOKO Button*** through ***MOKO Button APP***. Through this document, users will be initial to know the functional framework of product, setup process of Beacon advertisement, connection, configuration, DFU update and etc. This guide will not cover the sales administration and the ordering process. Some technical guides will be needed if further explanation is required.

- Applicable version description:

  ***Android APP Version: "MK Button" V2.0.0 or above***

  ***iOS APP Version: "MK Button" V2.0.0 or above***

  ***Firmware Version: BXP-B-D_V2.0.0 or above | BXP-B-CR V2.0.0 or above***

- Each figure (diagram, screenshot, or other image) and table are provided with a number and description:

  ***Figure 1: Example content of Beacon advertisement***

  ***Table 1: Content structure of Customized-Alarm mode advertising packet***

  The numbers and descriptions of the figure and table can be found in the "List of Figures" and "List of Tables".

# Table of Contents

# List of Figures

# List of Tables

# 1. Overview

This User Guide is mainly applicable for MOKO standard ***"MK Button"*** APP and MOKO Beacon with ***"MK Button"*** serial firmware, and mainly contained below parts:

- ***Content parsing of Beacon advertisement***
- ***MOKO Button APP user guidance***
- ***Common instructions of "nRF connect" APP***
- ***FAQ***

For more hardware technical info or assembly/disassembly instructions, please ask our sales team directly for official document – "***Product Specification***".

# 2. Content parsing of Beacon advertisement

Before introducing the APP user manual, we will guide you to fully understand the Beacon advertisement format and content parsing rules. In terms of the MOKO "***BXP-B-D***" series firmware, which is especially design for asset tracking, it contains several kinds of advertisement frame type which include standard format and customized format, that is:

- *Google Eddystone™ - UID*
- *Customized advertisement frame – "iBeacon"*
- *Customized advertisement frame - "Alarm info"*

For the Customized advertisement frame, the advertisement combination of ADV_IND (Advertising packet) and SCAN_RSP (Scan response packet). In the next we will describe the detailed information of these two packets.

- **ADV_IND (advertising packet)**----Mainly to express basic alarm information, Device ID and device name.
- **SCAN_RSP (scan response packet)**----Mainly to express some basic device information, for example 3-axis accelerometer parameters and sensor data. (It can be disabled in general settings)

To improve the readability of the screen formats, this document presents:

- The fixed Bluetooth fields highlighted in **blue**.
- The variable Bluetooth fields highlighted in **orange**.
- User-defined/configurable fields highlighted in **green**.
- Reserved fields highlighted in **red**.

The screenshots have been taken on the "nRF Connect" application, developed by Nordic Semiconductors

## 2.1 Google Eddystone – UID

In terms of UID advertisement frame type, this is exactly complied with Google Eddystone regulations, and you can refer to below link for more information: https://github.com/google/eddystone/tree/master/eddystone-uid

Below table described the detailed content structure and frame example of ***UID*** advertisement:

| *Eddystone - UID* | | | |
|---|---|---|---|
| | **Frame format** | | **Frame example** |
| **Frame bytes** | 00 | Length: 0x02 | |
| | 01 | Type: 0x01 | |
| | 02 | Data: 0x06 | |
| | 03 | Length: 0x03 | |
| | 04 | Type: 0x03 | |
| | 05 | Eddystone_UUID_LSB: 0xAA | |
| | 06 | Eddystone_UUID_MSB: 0xFE | |
| | 07 | Length: 0x17 | |
| | 08 | Type: 0x16 | |
| | 09 | Eddystone_UUID_LSB: 0xAA | |
| | 10 | Eddystone_UUID_MSB: 0xFE | |
| | 11 | Frame type UUID: 0x00 | |
| | 12 | RSSI@0m: 0x00 | |
| | 13 | Namespace ID[0] | |
| | 14 | Namespace ID[1] | |
| | 15 | Namespace ID[2] | |
| | 16 | Namespace ID[3] | |
| | 17 | Namespace ID[4] | |
| | 18 | Namespace ID[5] | |
| | 19 | Namespace ID[6] | |
| | 20 | Namespace ID[7] | |
| | 21 | Namespace ID[8] | |
| | 22 | Namespace ID[9] | |
| | 23 | Instance ID[0] | |
| | 24 | Instance ID[1] | |
| | 25 | Instance ID[2] | |
| | 26 | Instance ID[3] | |
| | 27 | Instance ID[4] | |
| | 28 | Instance ID[5] | |
| | 29 | Reserved | |
| | 30 | Reserved | |

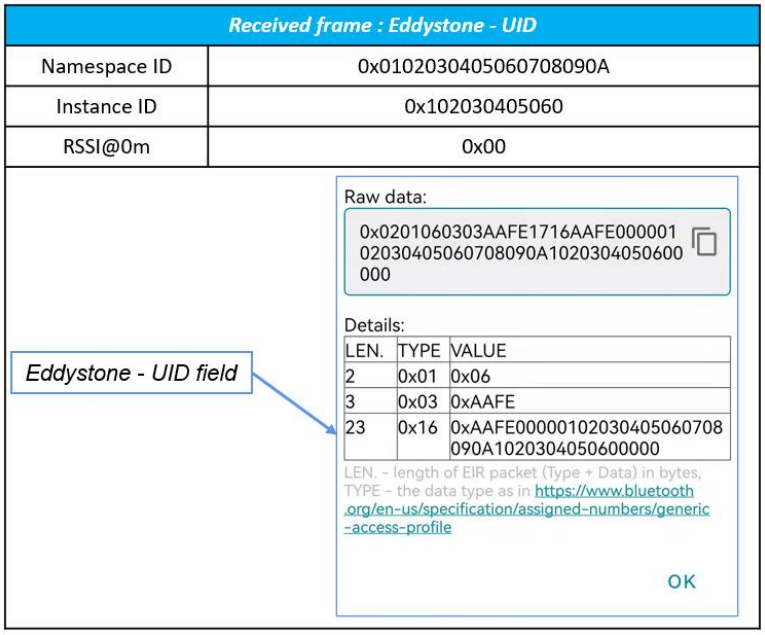



***Table 1: Content structure of UID advertisement***

## 2.2 Customized frame – "iBeacon"

This ***Customized iBeacon*** advertisement frame is combined with two parts, of which one is standard APPLE iBeacon format which comply with APPLE iBeacon regulations, another one is a customized scan response packet.

What the purpose of adding extra scan response packet is just displaying more information on iOS APP, as APP cannot get the detail information from APPLE iBeacon advertisement due to iOS system restrictions. For APPLE iBeacon regulations, you can refer to the link: https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf.

Below table described the detailed content structure and frame example of ***Customized-iBeacon*** advertisement:

| *MOKO Customized - iBeacon* | | | |
|---|---|---|---|
| **Frame bytes** | | **Standard iBeacon packet** | **Scan response packet** |
| | 00 | Length: 0x02 | Length: 0x02 |
| | 01 | Type: 0x01 | Type: 0x0A |
| | 02 | Data: 0x06 | Tx Power: 0x00 |
| | 03 | Length: 0x1A | Length: 0x1A |
| | 04 | Type: 0xFF | Type: 0x16 |
| | 05 | Apple CIN_LSB: 0x4C | Service data UUID_LSB: 0xAB (MOKO defined) |
| | 06 | Apple CIN_MSB: 0x00 | Service data UUID_MSB: 0xFE (MOKO defined) |
| | 07 | Beacon type: 0x02 | Frame type: 0x50 (MOKO defined) |
| | 08 | Data size: 0x15 | RSSI@1m: 0xC5 |
| | 09 | UUID[0] | ADV interval: 0x0A |
| | 10 | UUID[1] | UUID[0] |
| | 11 | UUID[2] | UUID[1] |
| | 12 | UUID[3] | UUID[2] |
| | 13 | UUID[4] | UUID[3] |
| | 14 | UUID[5] | UUID[4] |
| | 15 | UUID[6] | UUID[5] |
| | 16 | UUID[7] | UUID[6] |
| | 17 | UUID[8] | UUID[7] |
| | 18 | UUID[9] | UUID[8] |
| | 19 | UUID[10] | UUID[9] |
| | 20 | UUID[11] | UUID[10] |
| | 21 | UUID[12] | UUID[11] |
| | 22 | UUID[13] | UUID[12] |
| | 23 | UUID[14] | UUID[13] |
| | 24 | UUID[15] | UUID[14] |
| | 25 | Major[0] | UUID[15] |
| | 26 | Major[1] | Major[0] |
| | 27 | Minor[0] | Major[1] |
| | 28 | Minor[0] | Minor[0] |
| | 29 | RSSI@1m: 0xC5 | Minor[0] |
| | 30 | Not used | Not used |

***Table 2: Content structure of Customized – iBeacon advertisement***

| Received frame : MOKO Customized - iBeacon | |
|---|---|
| UUID | 0x00010203040506070809 10A0B0C0D0E0F |
| Major | 0x0001 |
| Minor | 0x0001 |
| RSSI@1m | 0xC5 = -59dBm |
| Tx Power | 0x00 = 0dBm |
| ADV interval (100ms/unit) | 0x0A = 1000ms |

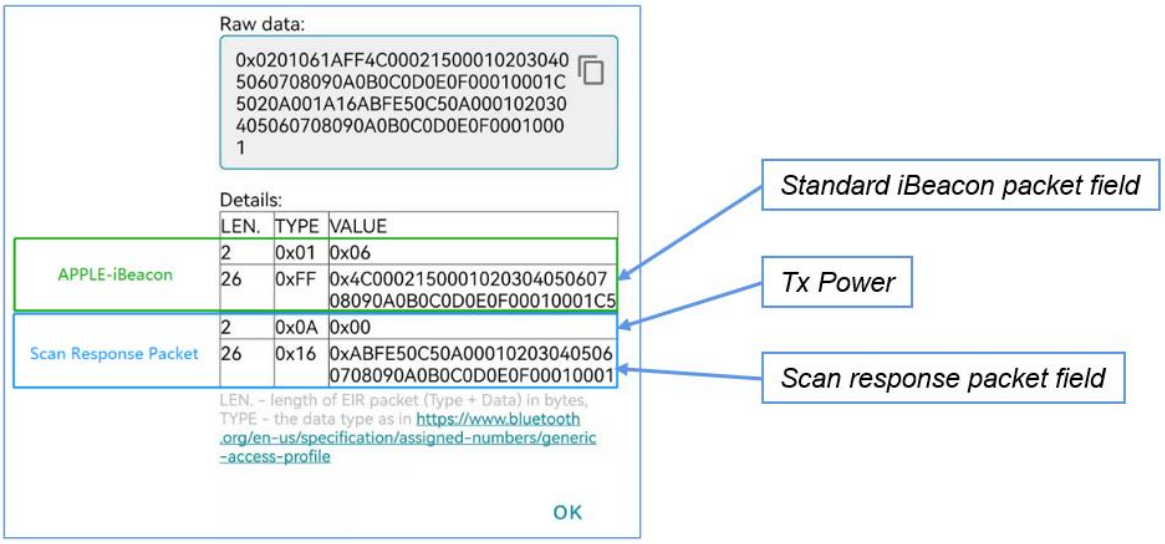



***Figure 1: Example content structure of Customized – iBeacon advertisement***

## 2.3 Customized frame – "Alarm info"

The ***Customized Alarm info*** advertisement frame actually includes four types of broadcasts: ***Single-press alarm info***, ***Double-press Alarm info***, ***Long-press Alarm info***, and ***Abnormal inactivity Alarm info***. These four types of broadcasts share the same broadcast format, with the primary distinction being made based on the **Byte7**.

This ***Customized Alarm info*** advertisement frame is combined with two parts, of which one is advertisement format which comply with device information, another one is a customized scan response packet.

Below table described the detailed content structure and frame example of ***Customized Alarm info*** advertisement:

| *MOKO Customized - Alarm info* | | | |
|---|---|---|---|
| **Frame bytes** | | **Standard iBeacon packet** | **Scan response packet** |
| | 00 | Length: 0x02 | Length: 0x18 |
| | 01 | Type: 0x01 | Type: 0x16 |
| | 02 | Data: 0x06 | Service data UUID_LSB: 0x00 (MOKO defined) |
| | 03 | Length: ≤ 0x0F | Service data UUID_MSB: 0xEA(MOKO defined) |
| | 04 | Type: 0x16 | Frame Type: 0x00 |
| | 05 | Service data UUID_LSB: 0xE0 (MOKO defined) | Accel.Full-scale |
| | 06 | Service data UUID_MSB: 0xFE (MOKO defined) | Accel.Motion threshold LSB |
| | 07 | Frame type: 0x20/0x21/0x22/0x23 | Accel.Motion threshold MSB |
| | 08 | Status flag: 0x01 | Accel.Dat_ X axis (mg) LSB |
| | 09 | TRIGGER_CNT_LSB | Accel.Dat_ X axis (mg) MSB |
| | 10 | TRIGGER_CNT_MSB | Accel.Dat_ Y axis (mg) LSB |
| | 11 | Device ID[0]: 0x00 | Accel.Dat_ Y axis (mg) MSB |
| | 12 | Device ID[1]: 0x00 | Accel.Dat_ Z axis (mg) LSB |
| | 13 | Device ID[2]: 0x01 | Accel.Dat_ Z axis (mg) MSB |
| | 14 | Device ID[3] | Beacon Temperature LSB |
| | 15 | Device ID[4] | Beacon Temperature MSB |
| | 16 | Device ID[5] | Ranging data: 0x00 |
| | 17 | Firmware type: 0x00/0x01 | Battery percentage(1%) / Battery voltage (1mV)_LSB |
| | 18 | RFU: 0x00 | Battery percentage(1%) / Battery voltage (1mV)_MSB |
| | 19 | Length: ≤ 0x0B | Mac address[0] |
| | 20 | Type:0x09 | Mac address[1] |
| | 21 | Device Name[0] | Mac address[2] |
| | 22 | Device Name[0] | Mac address[3] |
| | 23 | Device Name[0] | Mac address[4] |
| | 24 | Device Name[0] | Mac address[5] |
| | 25 | Device Name[0] | Length: 0x02 |
| | 26 | Device Name[0] | Type: 0x0A |
| | 27 | Device Name[0] | Tx Power |
| | 28 | Device Name[0] | Not used |
| | 29 | Device Name[0] | Not used |
| | 30 | Device Name[0] | Not used |

***Table 3: Content structure of Customized – Alarm info advertisement***

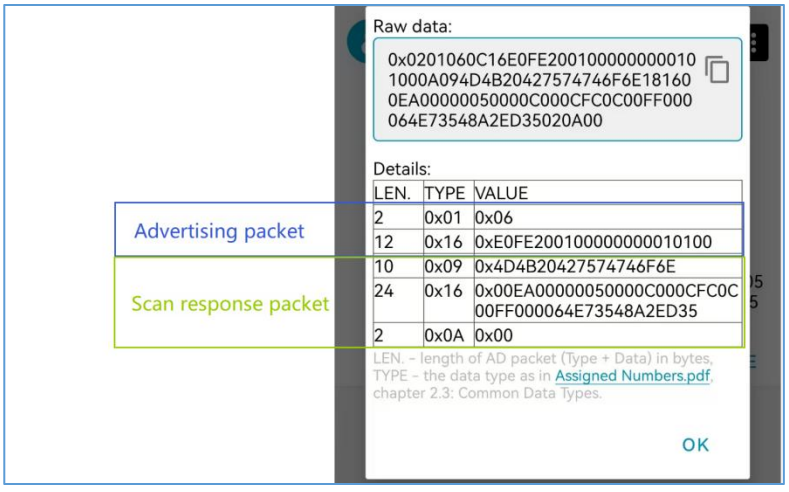
<table>
<tr><th colspan="4">Received frame : MOKO Customized - Alarm info (Advertising packet)</th></tr>
<tr><td colspan="2">Frame type</td><td colspan="2">MOKO-Defined advertisement frame type;<br>0x20: “Single press mode” advertisement<br>0x21: “Double press mode” advertisement<br>0x22: “Long press mode” advertisement<br>0x23: “Abnormal inactivity mode” advertisement<br>0x24 - 0x3F: RFU</td></tr>
<tr><td rowspan="3">Status flag</td><td>[Bit0]: Password verification status</td><td rowspan="3">0000 0001’b<br>(0x01)</td><td>[Bit0] = 0:Password verification disabled; 1: Password verification enabled</td></tr>
<tr><td>[Bit1]: Trigger status of alarm mode.</td><td>[Bit1] = 0: Alarm not be triggered. 1: Alarm be triggered</td></tr>
<tr><td>[Bit2-7]: Reserved</td><td>[Bit2-7]: Reserved</td></tr>
<tr><td colspan="2">TRIGGER_CNT</td><td colspan="2">2 bytes: 0x0000 = 0 (Times of magnetic triggered)</td></tr>
<tr><td colspan="2">Device ID</td><td colspan="2">1-6 bytes. Default value: 0x00 00 01 (3 bytes)</td></tr>
<tr><td colspan="2">Firmware type</td><td colspan="2">Below V2.0.x : 0x00<br>Above V2.0.x: 0x01</td></tr>
<tr><td colspan="2">DEV Name</td><td colspan="2">1-10 bytes. Default value: 0x4D 4B 20 42 75 74 74 6F 6E</td></tr>
<tr><th colspan="4">Received frame : MOKO Customized - Alarm info (Scan response packet)</th></tr>
<tr><td colspan="2">Frame type</td><td colspan="2">0x00: General device information advertisement</td></tr>
<tr><td colspan="2">Accel.Full-scale</td><td colspan="2">Full-scale of 3-axis accelerometer sensor. Default: ±2g<br>0x00: ±2g | 0x01: ±4g | 0x02: ±8g | 0x03: ±16g</td></tr>
<tr><td colspan="2">Accel.Motion threshold</td><td colspan="2">Motion threshold to judge movements. Default: 16mg</td></tr>
<tr><td colspan="2">Accel.Data_X axis</td><td colspan="2">0x003A = 58mg</td></tr>
<tr><td colspan="2">Accel.Data_Y axis</td><td colspan="2">0x0076 = 118mg</td></tr>
<tr><td colspan="2">Accel.Data_Z axis</td><td colspan="2">0x03A7 = 935mg</td></tr>
<tr><td colspan="2">Beacon Temperature</td><td colspan="2">Content Parse: 00 6C>>>108. 108×0.25=27℃. ( V1.0.4 )<br>Content Parse: 00 6C>>>108. 108×0.1=10.8℃.( Above V1.0.8 )</td></tr>
<tr><td colspan="2">Ranging Data</td><td colspan="2">Configuration range: -100 - 0dBm (Default: 0x00)</td></tr>
<tr><td colspan="2">Battery percentage(1%/unit) / Battery voltage (1mv/unit)</td><td colspan="2">If value greater than 100, it represents the battery voltage.<br>For example, 0x0CCC = 3276mV.<br>If value less than or equal to 100, it represents the battery percentage.<br>For example, 0x0064 = 100%.</td></tr>
<tr><td colspan="2">Mac address</td><td colspan="2">Beacon MAC address</td></tr>
<tr><td colspan="2">Tx Power</td><td colspan="2">1 byte signed integer, Tx Power, unit: dBm.</td></tr>
<tr><td colspan="4"> 
</td></tr>
</table>

***Table 4: Explanation of format parsing – Sensor info advertisement***

# 3. MOKO Button APP user guidance

Well, you have already familiar with the advertisement format and well to know how to parse the content in the advertisement, now we will guide you to know the *"MK Button"* APP and let's do some configuration on the Beacon.

## 3.1 Discover your Beacon device

The smartphone with MOKO *"MK Button"* APP (Android or iOS) installed can be as a *Bluetooth Center* to scan and discover the advertisement packets of Bluetooth Low Energy (BLE) Beacons, and it will provide multiple strategies to filter the Bluetooth packets and that can help you to capture the required BLE peripherals directly and quickly.

### 3.1.1 Select the device type of your tag

When enter into the app, the Beacon firmware options page will displayed, please make sure the firmware type of the device you are holding has the corresponding option, if you are not sure which one to choose, please contact our sales for confirmation.

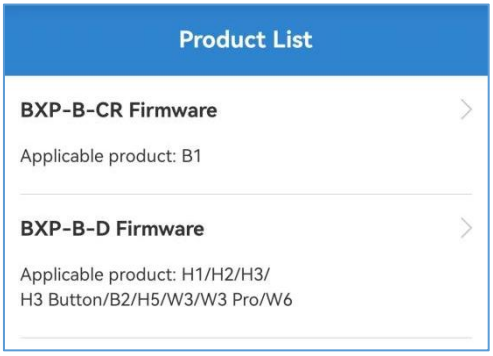



***Figure 2: Beacon firmware options interface preview***

The main difference between the ***BXP-B-CR*** firmware and ***BXP-B-D*** firmwares is as follows:

| Difference between BXP-B-D and BXP-B-CR | | | |
|---|---|---|---|
| | | BXP-B-D | BXP-B-CR |
| Adapted hardware | Chip model | nRF52810 | nRF52832 |
| | Buzzer | yes | yes |
| | Vibration motor | no | yes |
| | LED | yes | yes |
| | Rechargeable design | no | yes |
| Firmware function | Long Connection Alarm Function | yes | no |
| | Power off by button | no | yes |
| | Vibration Motor Related Functions | no | yes |
| | Sync beacon time | no | yes |
| | Broadcast battery percentage | yes | no |
| | Broadcast iBeacon and Eddystone-UID frame type | yes | no |
| | Button trigger event record storage and export | no | yes |

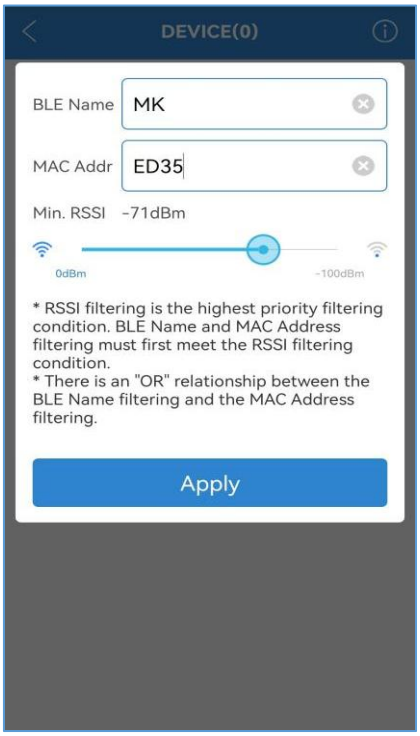
### 3.1.2 Scanning your Beacon device

After select the device type, the app page will enter into the "***SCANNING PREVIEW"*** interface, the *SCANNER* will keep scanning continuously and list all surrounding advertising Bluetooth LE peripherals which match the device type, unless stop scanning manually by pressing scanning button " ".

Below picture shows the default scanning preview of MOKO Button device. For more scanning previews of other advertisement type, please refer to ***chapter 3.3 - Getting your Beacon to broadcast***.

### 3.1.3 Filtering your Beacon device

To find your Beacon quickly, we have developed multiple strategies to filter among mass Beacon devices, users may narrow your device list by BLE name, MAC address and RSSI value through "***Edit Filter***" option.




| | |
|---|---|
| ***BLE Name*** | Allow to input 1-10 characters.<br>***Example:*** Only Beacon devices whose BLE name contain "MOKO" will be displayed in ***SCANNER*** interface.<br>***NOTE：*** *Space effective, and* ***case-insensitive*** *as well.* |
| ***Mac Addr*** | Allow to input 1-6 bytes (HEX).<br>***Example:*** Only Beacon devices whose Mac address contain "B3:E8" will be displayed in ***SCANNER*** interface.<br>***NOTE：*** *There is no need to input colon between MAC address in search filter column, and* ***case-insensitive*** *as well.* |
| ***Min.RSSI*** | Only the advertisement packets with RSSI not less than the Min.RSSI value will be displayed in ***SCANNER*** interface , and configuration range is -100dBm to 0dBm.<br>***Example:*** When setting the Min.RSSI value to -70, the ADV packets with RSSI of -70dBm or greater than -70dBm will be list in ***SCANNER*** interface. |

***NOTE:***

*1. RSSI is the highest priority filtering condition. It means that filtered device must meet the RSSI filtering condition.*

*2. There is an "OR" relationship between BLE Name and Mac Addr. It means that filtered device just need meet one of the filtering conditions.*

## 3.2 Connect with your Beacon device

For BLE devices, if the status is connectable, then it will shows "***CONNECT***" button in the "***SCANNING PREVIEW"*** interface. You can follow below steps to connect with your Beacon device:

**Step 1:** Press "***CONNECT***" button.

**Step 2:** Input connection password and then press "***OK***" button, *Green LED* will blink 2 times rapidly again.

***NOTE*** *– Default password in MOKO standard firmware is "Moko4321".*

## 3.3 Getting your Beacon to broadcast

Before setting up your Beacon advertisement, you may need to understand the basic advertising logical and working flowchart of the Beacon.

In Button products, SLOT advertisements are bound to "alarm mode", such as *"Single press mode"*, *"Double press mode"* and etc. In addition, the advertisement format of each SLOT (or alarm mode) are same exactly, you can refer to ***"chapter 2 – Content parsing of Beacon advertisement"*** to understand the content parsing in further.

Once device connected and the main screen of the "***MK Button***" APP appears, including ***ALARM interface***, ***SETTING interface*** and ***DEVICE interface***.

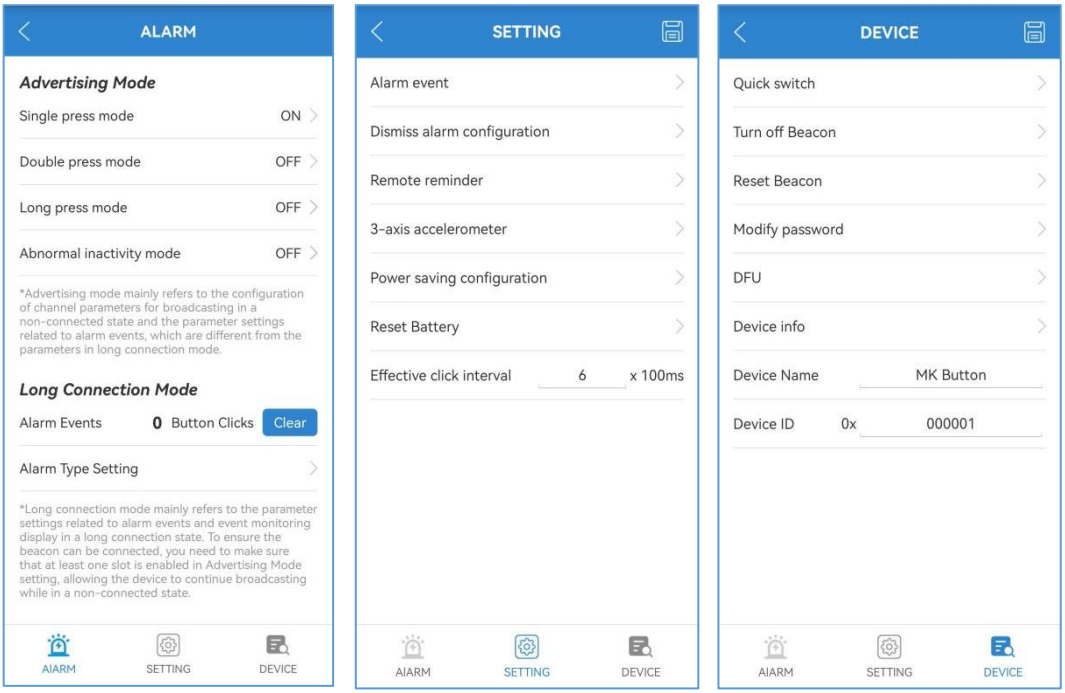



***Figure 3: Configuration interface overview of main screen***

***NOTE: BXP-B-CR firmware not support Long Connection Mode***

### 3.3.1 Setting up advertising mode

In the "***ALARM***" interface, there are two main types of modes, ***Advertising Mode*** and ***Long Connection Mode.*** Just as its name implies, ***Advertising Mode*** refers to the mode in which the device broadcasts data while in a **non-connected state**. The ***Long Connection Mode*** is the working mode when the device maintains a **continuous connection** with the scanning device(phone or gateway).

In the "***Advertising Mode***", there have four alarm modes which are corresponding to different advertisement slots in a sense and it can be configured separately or in parallel. You can define different advertising type and alarm type to each mode in practical. Meanwhile, it is also available for you to configure the advertisement parameters, alarm parameters and notifications in each mode separately.

- ***Single press mode***
- ***Double press mode***
- ***Long press mode***
- ***Abnormal inactivity mode****

***Remark:** Abnormal inactivity mode will be available only when device equipped with 3-axis accelerometer sensor.*

### 3.3.1.1 Set up "Single press mode"

This "*Single press mode*" defined the advertisement parameters and notification type when users single pressed the *"Alarm button"*. Only **single-click** the button will be taken effect to trigger the alarm in this mode.

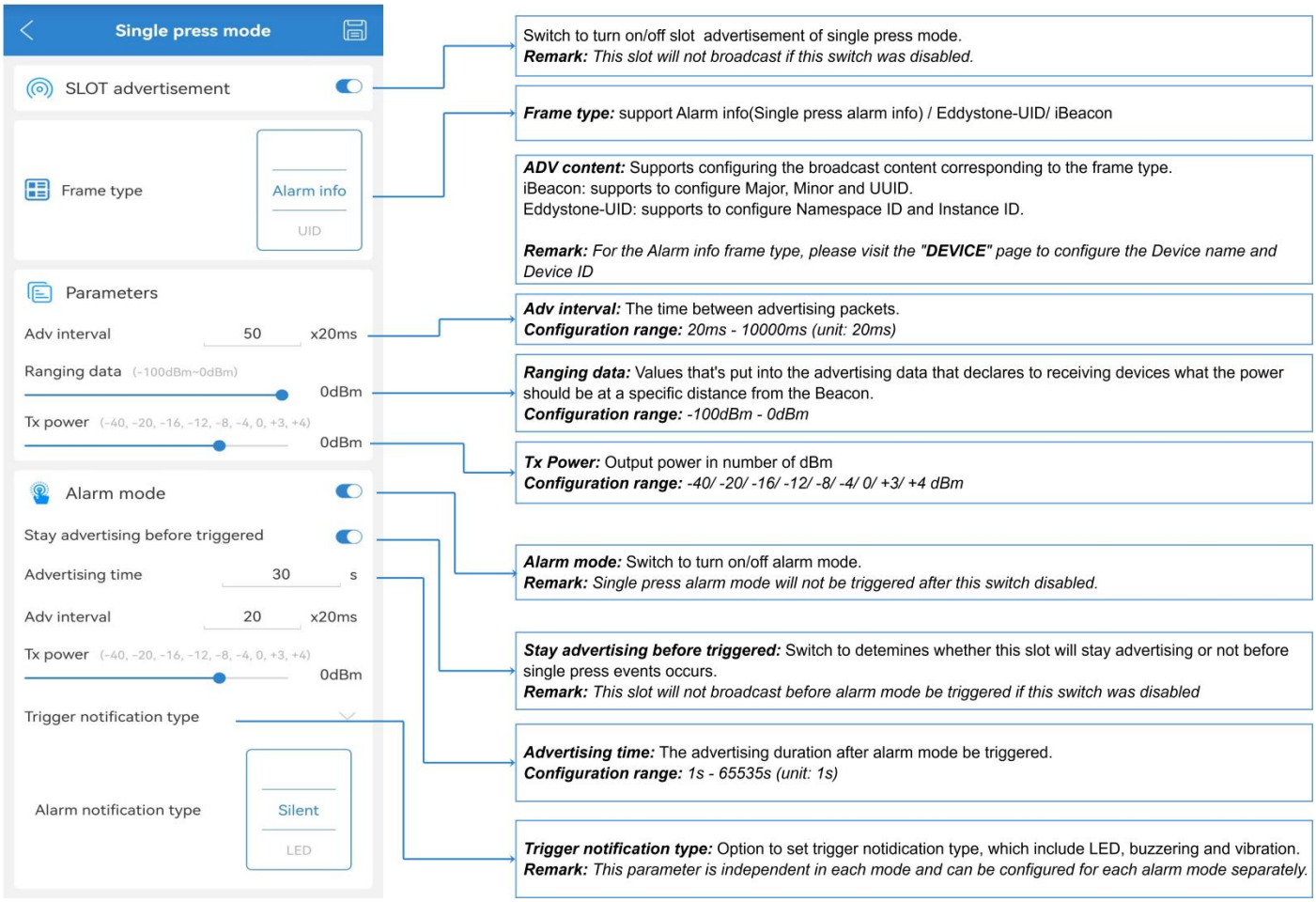



***Figure 4: How to configure "Single press mode" alarm***

The broadcast parameters are mainly divided into two parts: ***"Parameter"*** section and the ***"Alarm mode"*** section.

- If ***"Alarm mode"*** is disabled, the device will continuously broadcast according to the parameters under the ***"Parameter"*** section.
- If ***"Alarm mode"*** is enabled and the ***"Stay advertising before triggered"*** is disabled, the device will not broadcast until a button single press event occurs. After the event, it will broadcast according to the parameters under the ***"Alarm mode"*** section.
- If ***"Alarm mode"*** is enabled and the ***"Stay advertising before triggered"*** is enabled, the device will broadcast according to the parameters under the ***"Parameter"*** section before the button single press event occurs. After the button press event, the device will broadcast for a fixed duration(***Advertising time***) according to the parameters under the **Alarm mode** section, and then switch back to broadcasting according to the parameters under the ***"Parameter"*** section.

In here, there have an additional function - ***"Stay advertising before triggered"*** that need to be paid more attention. As we have mentioned in above picture demonstration, this is a switch to determines whether this slot will stay advertising or not before alarm mode be triggered. That is to say, if you enabled this option and finished the alarm mode settings, then device will keep advertising with the original parameters after disconnected. Once alarm be triggered, device will start advertising with the trigger parameters. For deeper understanding of this part, we have drawn a working flowchart in the below to express its different logical.

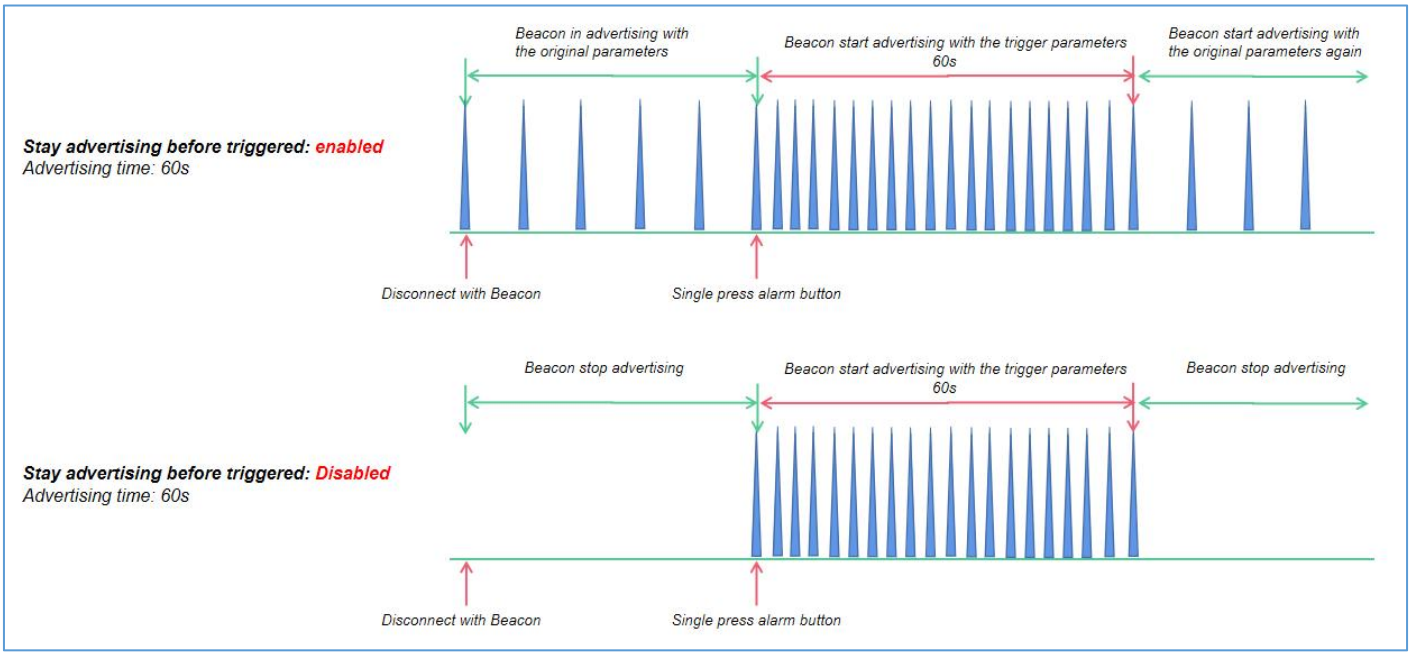



***Figure 5: What is "Stay advertising before triggered"***

***Remark:***

1. *The Alarm info in the Single press mode fixed broadcast Single press alarm info, the main difference from other modes is that the "Frame type" byte is 0x20. you can refer to the* ***chapter 2.3 Customized frame – "Alarm info"*** *to understand it.*

2. If you select the Alarm info frame type, After saving the configuration, please visit the ***"DEVICE"*** page to set the ***"Device name"*** and ***"Device ID"***.

### 3.3.1.2 Set up "Double press mode"

This "*Double press mode*" defined the advertisement parameters and notification type when users double pressed the *"Alarm button"*. Only **double-click** the button will be taken effect to trigger the alarm in this mode.

***Remark:***

1. *The Alarm info in the Double press mode fixed broadcast Double press alarm info, the main difference from other modes is that the "Frame type" byte is 0x21. you can refer to the* ***chapter 2.3 Customized frame – "Alarm info"*** *to understand it.*

2. If you select the Alarm info frame type, After saving the configuration, please visit the ***"DEVICE"*** page to set the ***"Device name"*** and ***"Device ID"***.

### 3.3.1.3 Set up "Long press mode"

This "*Long press mode*" defined the advertisement parameters and notification type when users long pressed the *"Alarm button"*. Only **long-press** the button will be taken effect to trigger the alarm in this mode.

***Remark:***

1. *The Alarm info in the Long press mode fixed broadcast Long press alarm info, the main difference from other modes is that the "Frame type" byte is 0x22. you can refer to the* ***chapter 2.3 Customized frame – "Alarm info"*** *to understand it.*

2. If you select the Alarm info frame type, After saving the configuration, please visit the ***"DEVICE"*** page to set the ***"Device name"*** and ***"Device ID"***.

### 3.3.1.4 Set up "Abnormal inactivity mode"

This "*Abnormal inactivity mode*" defined the advertisement parameters and notification type when device keep static for a long time.

Considering the practical construction or warehouse scenarios, when alone workers suffered an accidental injury and fails to trigger the button alarm in time, then the Beacon can also detect the status through 3-axis accelerometer sensor. If the duration of Beacon that stay in idle status exceed to the "abnormal inactivity time", then the Beacon will start advertising with the parameters you set.

***Remark:***

1. *The Alarm info in the Abnormal inactivity mode fixed broadcast Abnormal inactivity alarm info, the main difference from other modes is that the ''Frame type" byte is 0x23. you can refer to the* ***chapter 2.3 Customized frame –"Alarm info"*** *to understand it.*

2. If you select the Alarm info frame type, After saving the configuration, please visit the ***"DEVICE"*** page to set the ***"Device name"*** and ***"Device ID"***.

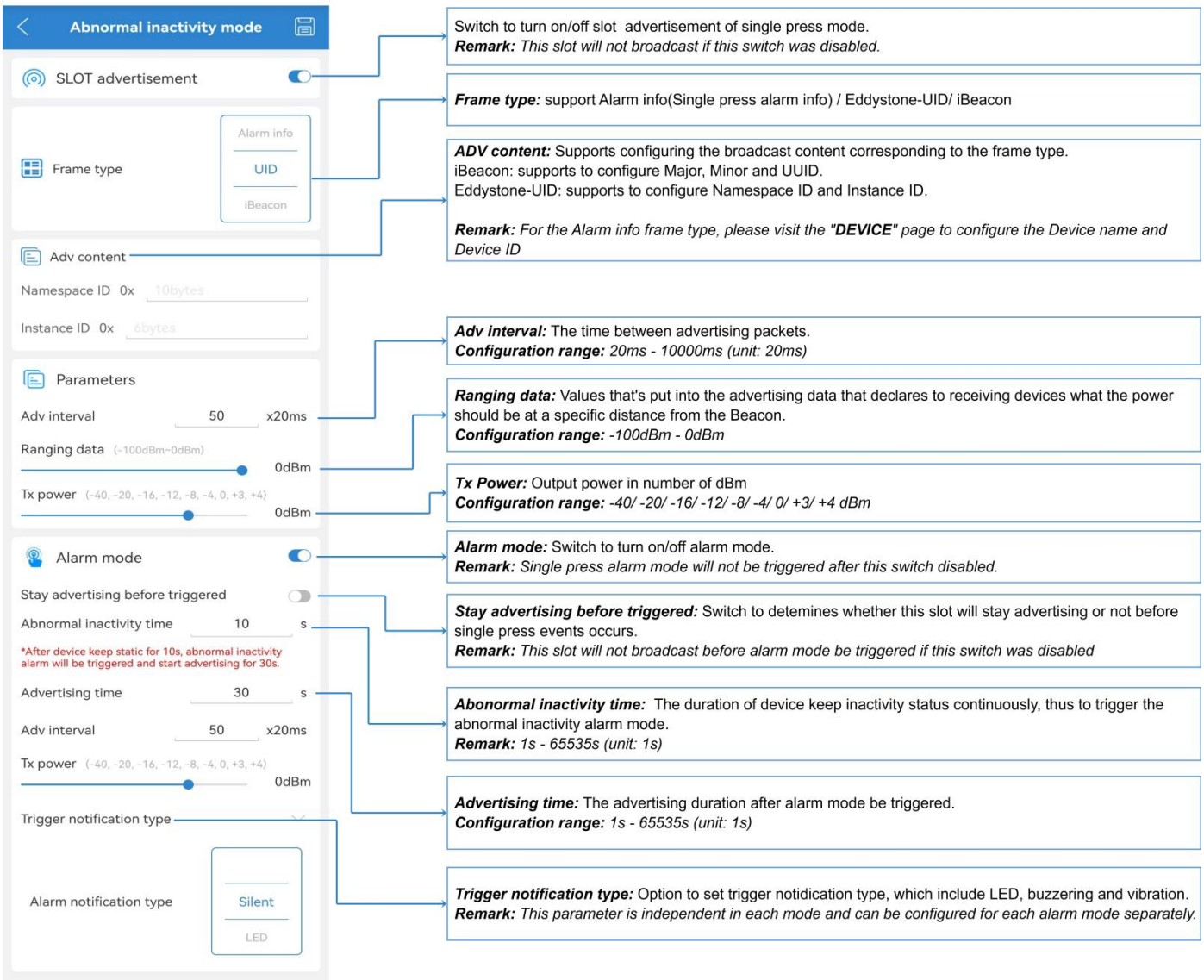



***Figure 6: How to configure "Abnormal inactivity mode" alarm?***

Now you have learned how to configure the parameters of "*Abnormal inactivity alarm*", and then we will also try to describe the flowchart after abnormal inactivity alarm be triggered, to give you a deeper understanding.

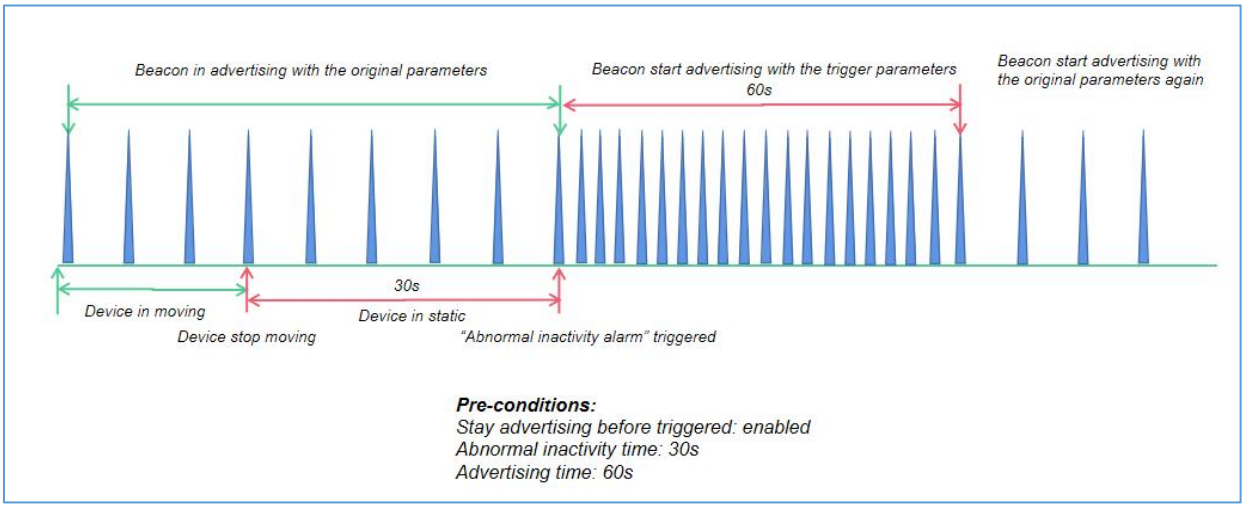



***Figure 7: Working principle of "Abnormal inactivity alarm"***

## 3.3.2 Setting up advertising mode alarm notifications

In last chapter you have learned how to set up the advertisement parameters and understood the flowchart of alarm trigger, in the next we will guide you to know the alarm notification configurations.

In most of practical scenarios, alarm notification is crucial and intuitive for users to perceive the alerts. In Button series, several kinds of alarm notifications, including LED, Beeping and vibration are available to be preset. Here we have listed the configurable type of notifications, and you can go to ***"SETTING >>> Alarm mode >>> Trigger notification type"*** to finish the configurations.

| *Alarm notification type* | *Description* |
|---|---|
| Silent | No alarm notifications after alarm triggered. |
| LED | LED will be blinking as following the parameters you set. |
| Vibration | Motor will be vibrating as following the parameters you set. |
| Buzzer | Buzzer will be beeping as following the parameters you set. |
| LED + Vibration | LED will be blinking and Motor will be vibrating as following the parameters you set. |
| LED + Buzzer | LED will be blinking and Buzzer will be beeping as following the parameters you set. |

***Table 5: Configurable type list of alarm notifications***

### 3.3.2.1 Set up "LED" alarm notification

In "LED" notification type, there have two parameters that need to be set.

- ***Blinking time.*** ----- LED blinking duration after alarm triggered
- ***Blinking interval.*** ----- LED blinking interval.

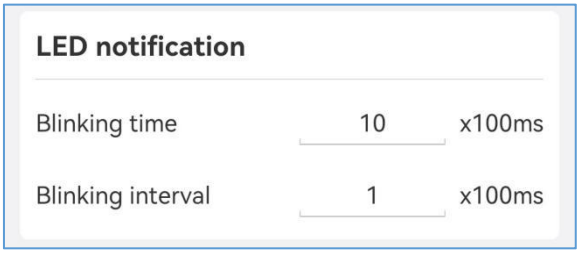



***Figure 8: How to set up "LED" alarm notification?***

### 3.3.2.2 Set up “Vibration” alarm notification

In “*Vibration*” notification type, there have two parameters that need to be set.

- ***Vibrating time.*** ----- Motor vibrating duration after alarm triggered
- ***Vibrating interval.*** ----- Motor vibrating interval.

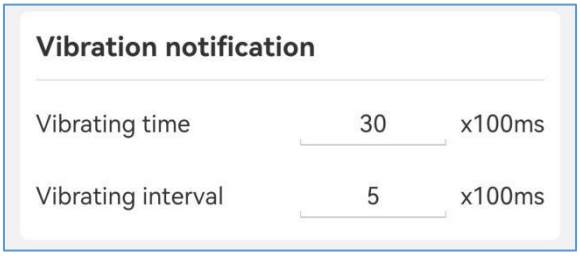



***Figure 9: How to set up “Vibration” alarm notification?***

***NOTE:*** *This configuration item is supported only by BXP-B-CR firmware and is not support by the BXP-B-D firmaware.*

### 3.3.2.3 Set up “Buzzer” alarm notification

In “*Buzzer*” notification type, there have two parameters that need to be set.

- ***Ringing time.*** ----- Buzzer ringing duration after alarm triggered. *Configuration range: 1 – 6000 (Unit: 100ms)*
- ***Ringing interval.*** ----- Buzzer ringing interval. *Configuration range: 1 – 100 (Unit: 100ms)*

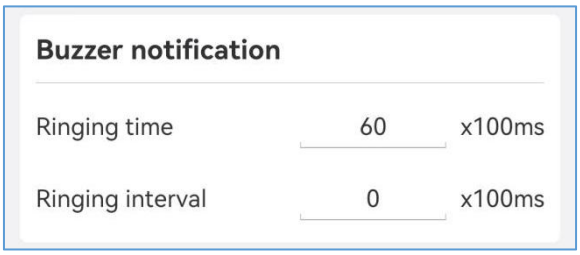



***Figure 10: How to set up “Buzzer” alarm notification?***

### 3.3.2.4 Set up “LED + Vibration” alarm notification

In order to compatible use for vision or hearing impairment, we have also provided some combination notifications in the firmware, for instance *“LED + Vibration”* and *“LED + Buzzer”*. It means that LED will be blinking and motor will be vibrating at the same time after alarm triggered.

In below “*LED + Vibration*” alarm notification type, you may need to configure LED notification parameters as well as vibration notification parameters, and the configuration range is totally same as we mentioned in last chapter.

***NOTE:*** *This configuration item is supported only by BXP-B-CR firmware and is not support by the BXP-B-D firmaware.*

### 3.3.2.5 Set up “LED + Buzzer” alarm notification

In below “*LED + Buzzer*” alarm notification type, you may need to configure LED notification parameters as well as buzzer notification parameters, and the configuration range is totally same as we mentioned in last chapter.

## 3.4 Long Connection Mode

***"Long Connection Mode"*** mainly refers to the parameter settings related to button trigger alarm events and button trigger event monitoring display in a long connection state

***NOTE:***

*1. To ensure the beacon can be connected, you need to make sure at lease one slot is enabled in "Advertising Mode" setting, allowing the device to continue broadcasting while in a non-connected state.*

*2. The Long Connection Mode is supported only by BXP-B-D firmware and is not support by the BXP-B-CR firmaware.*

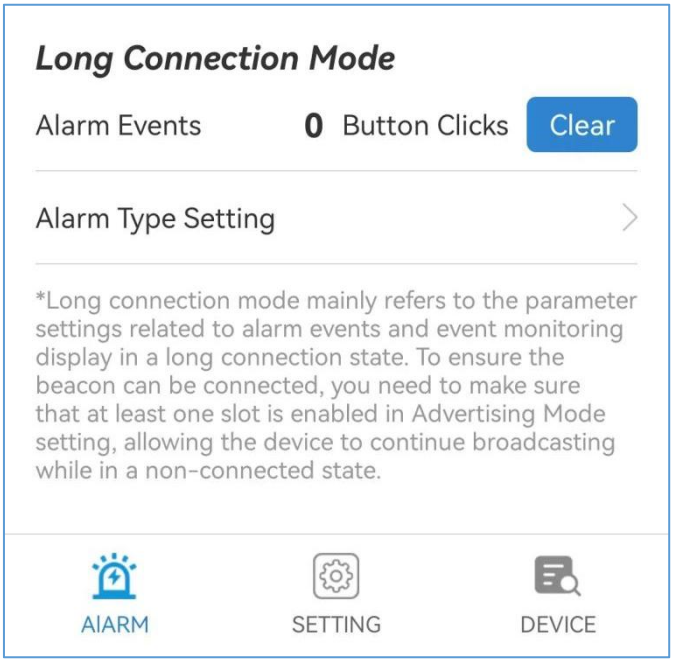



***Figure 11: Long Connection Mode parameters***

- ***Alarm Events:*** Displays the number of times the button has been pressed in one operation cycle in the connected state.
- ***Clear:*** clear the numbers of button clicks.
- ***Alarm Type Setting:*** In the connected state, after the button trigger event occurs. The device can also be configured to perform actions such as flashing LED and sounding buzzers. The definition of this parameter is basically the same as in ***"Advertising Mode"***, you can refer to ***chapter 3.3.2 Setting up advertising mode alarm notifications*** to understand the meaning of the relevant configuration. However, please note that this parameter belongs to the the Long connection mode, which is different from the trigger alarm in Advertising mode.

## 3.5 Dismiss alarm mode

In previous chapter, you have learned how to set up the alarm mode, and now you may need to know how to dismiss alarm mode. We have provided two ways to dismiss alarm, one is that dismiss alarm by triple press side button, and you can refer to **"*chapter 3.12.2 – Disable your Beacon's button function*"** to disable this function; Another one is that dismiss alarm through APP remote control.

In the meantime, when alarm mode be dismissed, we have also provided a choice to set the dismiss notifications, thus to remind users that the alarm mode has been dismissed successfully. For this, it can be widely used in task assignment scenarios.

***NOTE:*** *This feature is effective for both* ***"Advertising mode"*** *and* ***"Long connection mode"****.*

### 3.5.1 Remote dismiss alarm

You can access to ***"SETTING >>> Dismiss alarm configuration"*** option and then just tap the ***"Dismiss"*** button to dismiss the ongoing alarm.

***Remark:***

*1. Remote dismiss alarm will dismiss all ongoing alarm.*

*2. For "Long connection mode", this button function is the same as the 'Clear' button function under the Long Connection Mode section on the "Alarm" page.*


### 3.5.2 Configure dismiss alarm notifications

For the dismiss alarm notification type, it is totally same as alarm notifications, and you can refer to ***chapter 3.3.2 Setting up advertising mode alarm notifications*** for more configuration details.

You can access to ***"SETTING >>> Dismiss alarm configuration"*** option directly to finish the configurations.

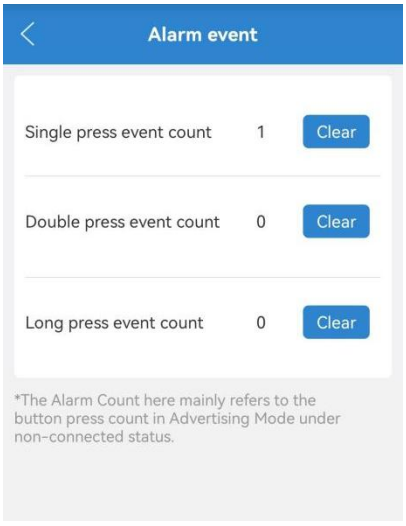
## 3.6 Obtain alarm event

In most practical scenarios, alarm event record is an essential requirement for further statistics and analysis. For instance, in some treatment compliance scenarios, when elder taken the medicine and press the button to report the event, and meantime it will be recorded with timestamp in the firmware for backup. These event records can be exported for comparison or analysis in some cases when dashboard missed some report events.

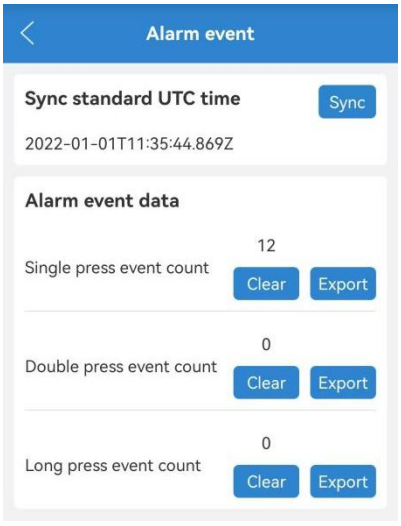
***NOTE:***

*1. This feature is supported only by the BXP-B-CR firmware. For the BXP-B-D firmware, it only supports retrieving the trigger count, but does not support storage and export functions.*

*2. This count mainly refers to the button trigger count in "Advertising mode" when the device is in a non-connected state. Additionally, you need to enable the "Alarm mode" feature in the SLOT configuration. Please refer to the **chapter 3.3.1 Setting up advertising mode** to review this point.*




*(BXP-B-D)*




*(BXP-B-CR)*

Regarding of alarm event record function, there have two parts you need to know for BXP-B-CR Firmware:

- ***Sync standard UTC time***
- ***Export alarm event data***

## 3.6.1 Sync Beacon time

Before using the alarm event record functions, it is better to sync the standard UTC time to Beacon. Go to ***"SETTIING >>> Alarm event"*** and just tap ***"Sync"*** button to sync the standard UTC time (standard time zone).

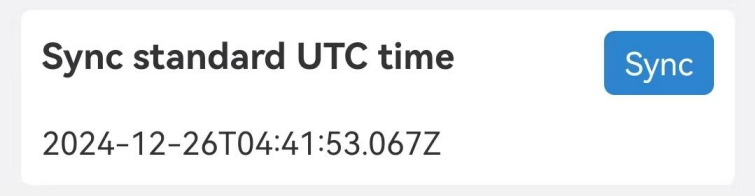



***Figure 12: How to sync standard UTC time to Beacon?***

***NOTE:*** *This configuration item is supported only by BXP-B-CR firmware and is not support by the BXP-B-D firmaware.*

## 3.6.2 Export alarm event data

For the BXP-B-CR firmware, the alarm event data currently only button alarm can be counted and exported, which include "*Single press event*", "*Double press event*" and "*Long press event*". Go to ***"SETTIING >>> Alarm event"*** interface and below we will describe some basic operations.

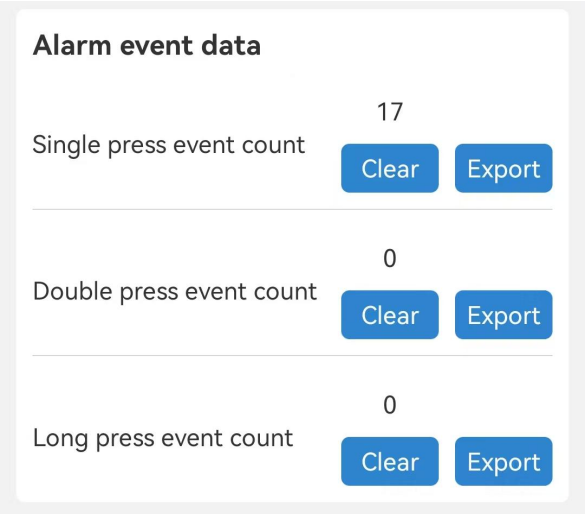



***Figure 13: How to check the counted record of alarm event?***

- ***Clear*** ----- *To clear the record of alarm event. (It will erase the event record in the Beacon)*
- ***Export*** ----- *To export the record of alarm event.*

Please pay attention that these counted alarm event are independent and can be exported separately. Choose any alarm event and then tap ***"Export"*** button to enter into the event record interface and we will describe some relevant operations in the below.

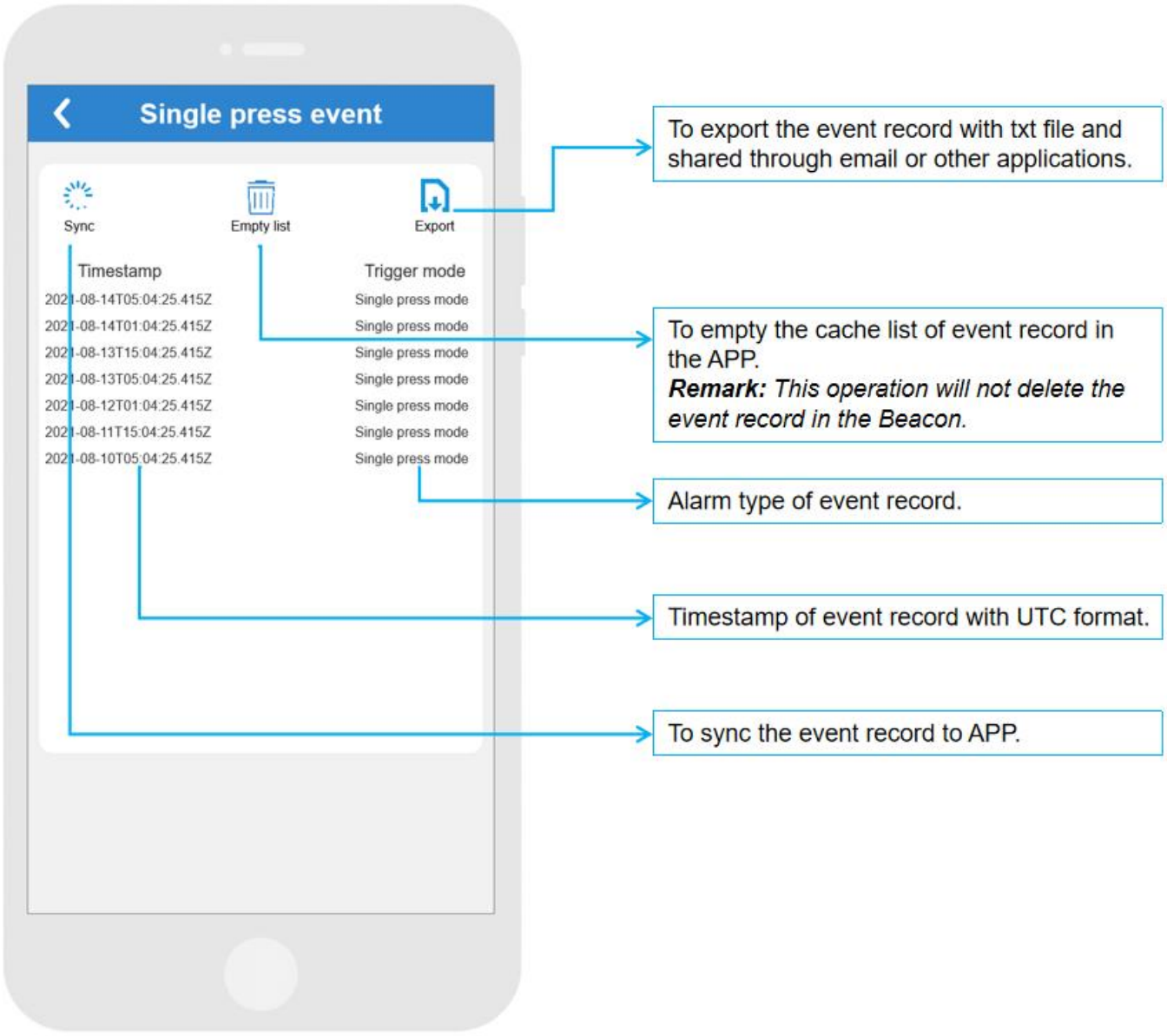



***Figure 14: How to export alarm event record?***

***NOTE:*** *This configuration item is supported only by BXP-B-CR firmware and is not support by the BXP-B-D firmaware.*

## 3.7 Remote trigger your Beacon

In some cases when you need to find your Beacon, then you can use this "*Remote reminder*" functions. You can go to ***"SETTING >>> Remote reminder"*** and then set the remote remind notification type, then just tap ***"Remind"*** button, Beacon will start blinking/Vibrating/Beeping according to the preset parameters.

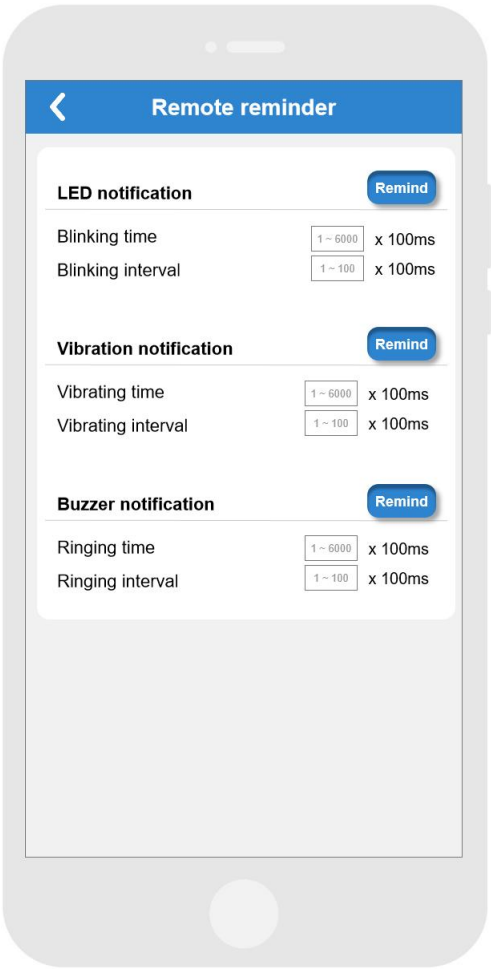



***Figure 15: How to remote trigger your Beacon?***

There have three kinds of reminder notifications available for configuration, as listed in the below. However, these three notification type are independent in this remote reminder function, and it can only be issued and executed separately due to its specific scenarios.

- ***LED notification***
- ***Vibration notification*** ***(only BXP-B-CR firmware supoort)***
- ***Buzzer notification***

***Remark:*** *Configuration range of notification parameters in here are totally same as in alarm notifications.*

***NOTE*** *– Please make sure that the hardware of Beacon support such kind of notification type before configurations.*

## 3.8 Access to 3-axis accelerometer sensor

Regarding of MOKO Button series product, currently we have only adopted ***3-axis accelerometer sensor*** for "*power saving mode*" and "*abnormal inactivity alarm" functions*. For power saving mode, you can refer to ***"chapter 3.9 – Power saving mode"*** for more information; For abnormal inactivity alarm, you can refer to ***"chapter 3.3.1.4 – Set up "Abnormal inactivity mode"*** for more information.

You can go to "***SETTING >>> 3-axis accelerometer"*** interface for further configurations. For the detailed parameters, please refer to below descriptions.

***Remark:*** *Configuration interface will be shown only when device equipped with 3-axis accelerometer sensor.*

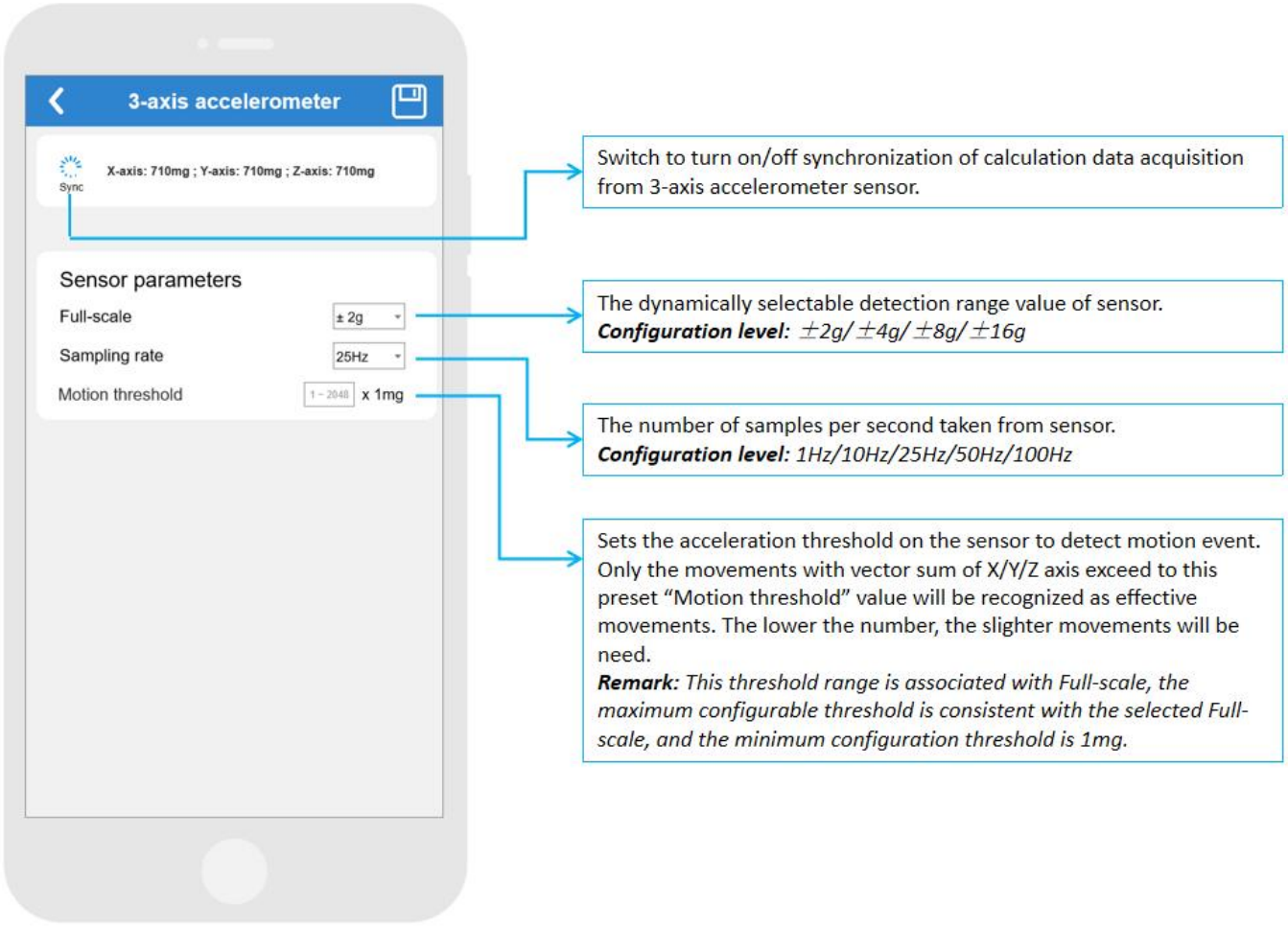



***Figure 16: Configurations of 3-axis accelerometer sensor***

## 3.9 Power saving mode

To save the power consumption of Beacon and extend to longer lifetime, we have provided a power saving mode to configure. The working logical is : When the duration that Beacon stay in idle status exceed to the specific value (*Static trigger time*) you preset, then Beacon will go to power saving mode and stop advertising, unless detect the movements again. You can also refer to below picture demonstration.

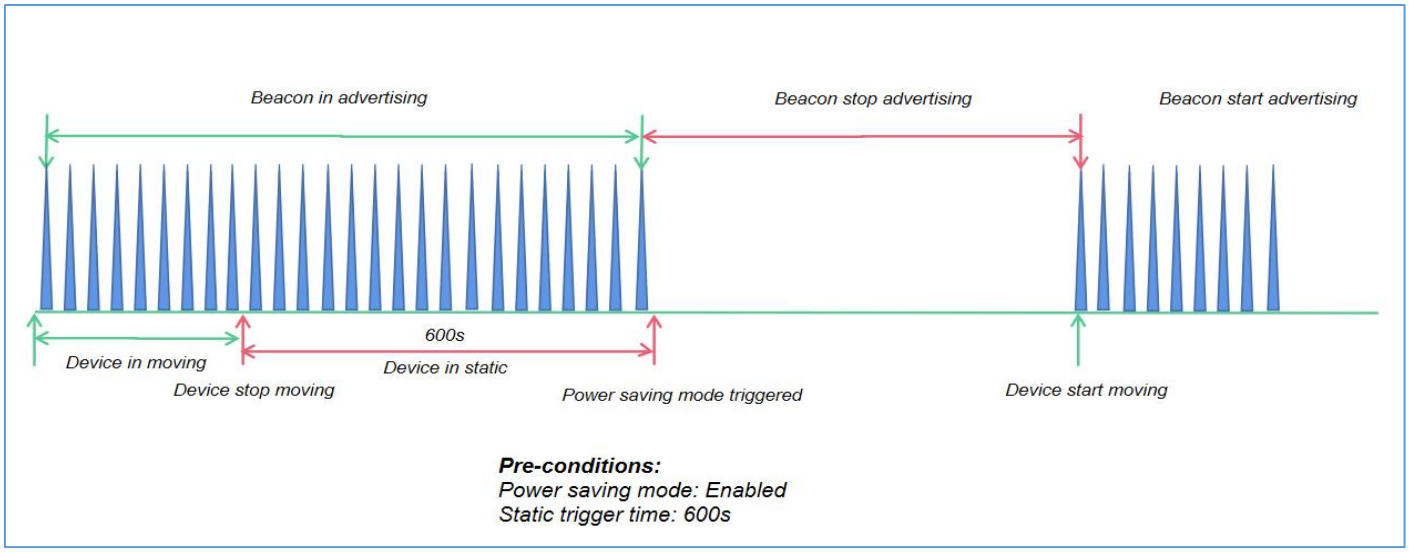



***Figure 17: Working flowchart of power saving mode***

After familiar with the working principle, then you can go to ***"SETTING >>> Power saving configuration"*** option to enable ***"Power saving mode"*** function, and set the static trigger time as you want.

- ***Static trigger time*** *---- The duration of Beacon stay in idle status to judge whether Beacon will go to power saving mode or not. Configuration range: 1 -65535s.*

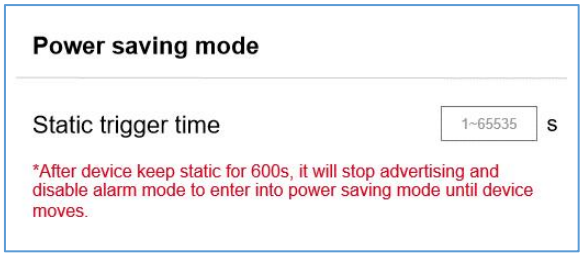



***Figure 18: How to configure power saving mode?***

## 3.10 Reset Battery

The device supports broadcasting and displaying the remaining battery level as a percentage. For the battery percentage, If you have replaced the battery with a new one, please connect the device and enter ***"SETTING>>>Reset Battery"*** configuration item to reset the battery, ensuring that the device can calculate the battery level starting from 100%. Additionally, please make sure that the battery you replace is a new one with a full charge.

***NOTE:*** *Only BXP-B-D firmware(Above V2.0.0) support broadcast battery battery percentage in the "Alarm info" frame type. The firmware by default broadcasts the battery percentage. If you wish to broadcast the battery voltage in the "Alarm info" advertising packet , please refer to* ***chapter4.6 How to configure broadcast battery voltage (Alarm info)*** *for switching.*

## 3.11 Configure “Effective click interval”

“Effective click interval” is defined as a time interval between two continuous clicks, and two clicks within this value will be recognized as continuous press. In some practical scenarios, for example in hospital or home care, elderly person comes with the majority of product audience. Considering the limitations and success rate of elderly people being able to press buttons in multiple times continuously within a short period of time , we have provided a solution for end user to lengthen this press interval to make it more adaptive for elderly person.

You can go to ***“SETTING >>> Effective click interval”*** option to preset this value and it will take effect immediately. To make sure the user experience, we have restricted it in reasonable range : *500ms - 1500ms (unit: 100ms)*.

***Remark: Default effective click interval is 600ms.***

For more, please see below picture demonstrations.

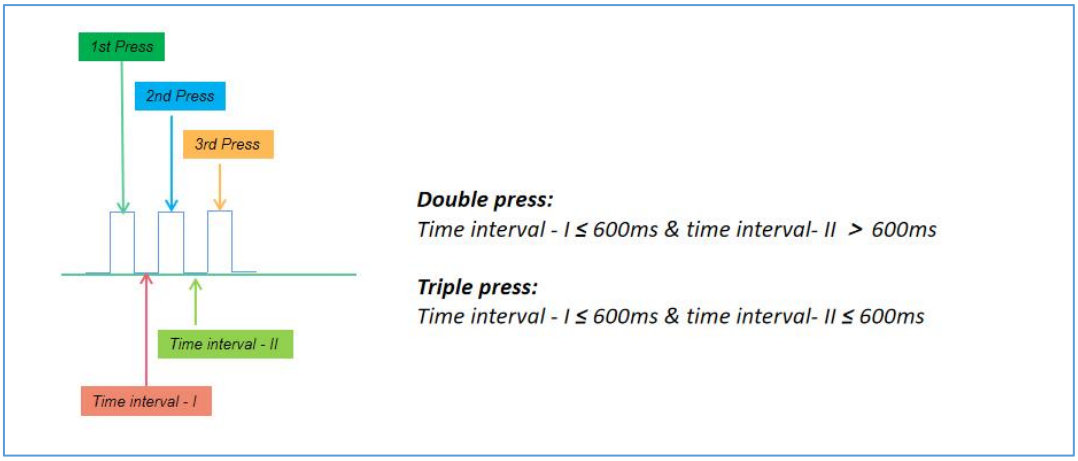
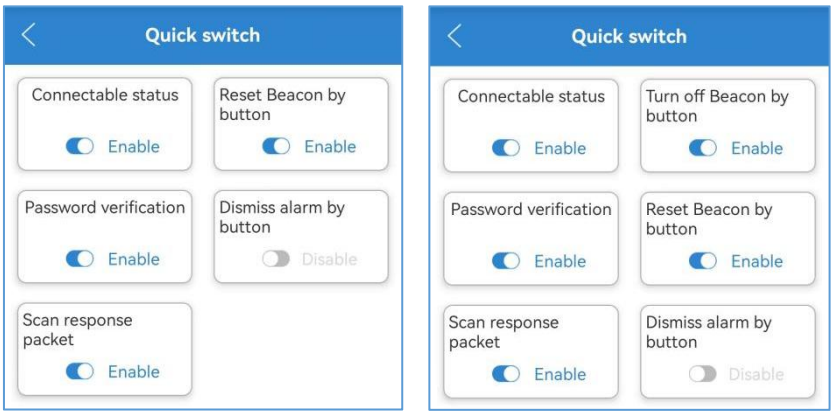



***Figure 19: How to configure the reasonable “effective click interval”?***

## 3.12 Quick switch for basic features

In current firmware, there have some basic functions such as *“Turn off Beacon by button”*, *“Reset Beacon by button”* and etc. As well, we have designed switches for these basic functions, you can disable/enable these features through the switch, thus adapting to different use cases.

You can enable/disable these function switch in ***“Device >>> Quick switch”*** section, and below shows the “*Quick switch”* interface as well as relevant feature descriptions.




*(BXP-B-D)* *(BXP-B-CR)*

***Figure 20: Configurations of Quick switch interface***

### 3.12.1 Switch your Beacon to unconnectable

In some cases, you will need to disable the Beacon connectable property, to let it working just as a location tag. So you can disable the connectable status in ***DEVICE >>> Quick switch >>> Connectable status.*** After disabled, device will not be connected by master BLE device any more unless reset the Beacon.


***Figure 21: Switch your Beacon to unconnectable status***

### 3.12.2 Disable your Beacon's button function

Before disabling the button function of Beacon, you need to know that the device support of our ***BXP-B-CR*** firmware has two separate buttons to realize the basic functions, and we will describe the difference between them below:

| *BXP-B-CR Series Product* | | |
|---|---|---|
| ***Button type*** | ***Descriptions*** | ***Function*** |
| Side button | The button in the side of the Beacon | Power on/off (Long press)<br>Battery indicator (Single press)<br>Reset (Long press + Single press) |
| *"Alarm button"* | The button in the positive center of the Beacon | Single press alarm<br>Double press alarm<br>Long press alarm<br>Dismiss alarm (Triple press) |

***Table 6: Difference between "Side button" and "Alarm button" (BXP-B-CR)***

For the ***BXP-B-D*** firmware product, the device only support one button.

| *BXP-B-D Series Product* | | |
|---|---|---|
| ***Button type*** | ***Descriptions*** | ***Function*** |
| *"Alarm button"* | The button in the positive center of the Beacon | Power on<br>Reset (Long press + Single press)<br>Single press alarm<br>Double press alarm<br>Long press alarm<br>Dismiss alarm (Triple press) |

***Table 7: Button function (BXP-B-D)***

However in some specific cases, there will need to disable the button functions in order to avoid intended operations, so in logically there also should have the switch to enable/disable these functions. You can go to "***DEVICE >>> Quick switch"*** to disable these button control functions.

- ***Turn off Beacon by button*** ----- *You will not be able to power off Beacon through button once disabled.*
  *(Remark: Only BXP-B-CR firmware support power off the beacon by button. BXP-B-D firmware only support power off beacon by APP)*
- ***Reset Beacon by button*** -------- *You will not be able to reset Beacon through button once disabled.*
- ***Dismiss alarm by button*** ------- *You will not be able to dismiss alarm through button once disabled.*

### 3.12.3 Disable password verification

You can go to ***DEVICE >> Quick switch >> Password verification*** to disable password verification. Once password verification disabled, there will no need to input password when you connect with Beacon.

***Remark:*** *If password verification feature was disabled, you will not be able to change password or reset Beacon due to security process restrictions.*

### 3.12.4 Disable scan response packet

As we have described previously, advertisement format is consistent of ADV_IND and SCAN_RSP. In order to save the power consumption, we also provided the switch to turn on/off the scan response packet.

You can access to "***DEVICE >> Quick switch >> Scan response packet"*** to disable it and then relevant information which included in scan response packet will not be shown in the "*SCANNING PREVIEW"* interface.

## 3.13 Remote turn off your Beacon

In some cases that mechanical button function should be disabled to avoid intentional shut down situations, then we will need to remote shut down function for backup.

Under this circumstance, you can remote turn off your Beacon by just tapping "***DEVICE >>> Turn off Beacon"*** option, then it will give prompt to you to confirm the shut down operations.

***NOTE*** *– Please make sure that device have mechanical button, or other ways to turn on the Beacon before you do remote shut down operations.*

## 3.14 Remote reset your Beacon

In some cases that mechanical button function should be disabled or without mechanical button, to avoid intentional reset situations, then we will need remote reset function for backup. You can remote reset your Beacon by just tapping "***DEVICE >>> Reset Beacon"*** option, then it will give prompt to you to confirm the reset operations.

***Remark:*** *Remote reset operations will not change the password to factory reset status. If you want to totally reset the Beacon with password restore as well, then you may need to do hardware reset operations.*

## 3.15 Modify "Connection password"

In MOKO firmware, password verification is ON by default and *"Moko4321"* was set as default connection password. You can modify your Beacon's connection password by just tapping ***"DEVICE >>> modify password"*** option, and then input the password as you want.

***Remark:*** *Connection password should not exceed 16 characters in length.*

## 3.16 Modify “Device Name”

As we have described in the content parsing of Beacon advertisement, “*Device name*” was set as a global variable and contained in the ***“Alarm info”*** advertising packet, so you can access to ***“DEVICE >>> Device Name”*** to full-fill with 1-10 ASCII characters.

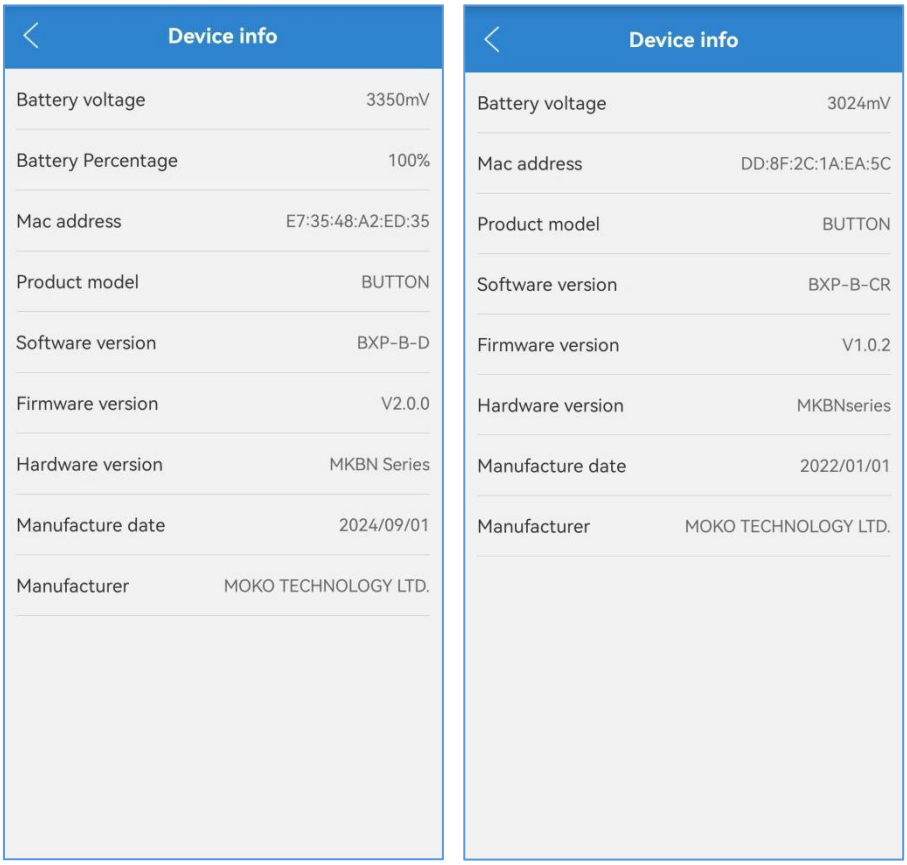
## 3.17 Modify “Device ID”

As we have described in the content parsing of Beacon advertisement, “*Device ID*” was set as a global variable and contained in the ***“Alarm info”*** advertisement packet. It was designed to adapt some special rules of serial ID or combination of Major/Minor and etc., so you can access to ***“DEVICE >>> Device ID”*** to full-fill with 1-6 bytes hexadecimal values.

## 3.18 Check your device information

In some cases, you will need to know the firmware version of Beacon, or product model and etc. You can access to “***Device >>> Device info”*** interface to check and confirm these essential information.

In customization firmware, these information can also be edit as customer required.




*(BXP-B-D)* *(BXP-B-CR)*

***Figure 22: Device information display interface***

# 3.19 Upgrade your Beacon's firmware

The Beacon can realize OTA firmware update through DFU operations. Regarding of different import method of android APP and iOS app, here we will describe the detailed steps separately.

## 3.19.1 Android – MOKO APP DFU instructions

**Step 1:** Copy the upgrade package to a folder in the root directory of the smartphone.

**Step 2:** ***Connect*** with device and go to "***SETTING***" interface, press "***DFU***" option and then file manager route of smartphone will be prompted.

**Step 3:** Choose the required upgrade firmware package and the upgrade process will start automatically. Wait for the upgrade to complete, APP will disconnect with Beacon after upgrade completed successfully.

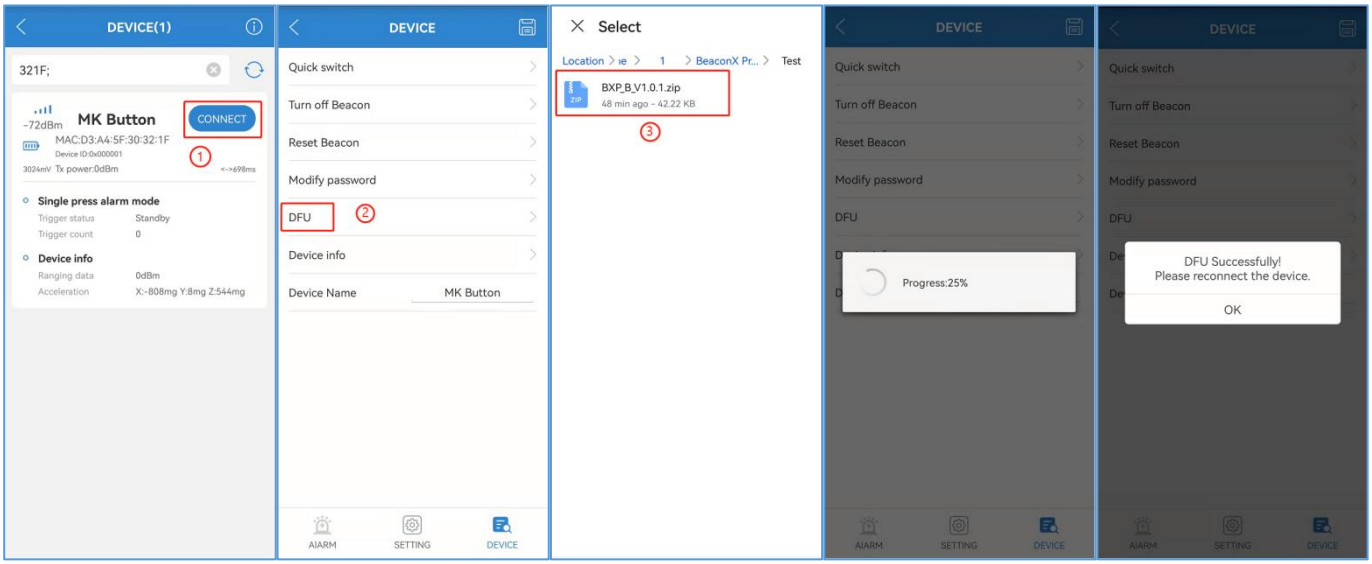



***Figure 23: Firmware upgrade workflow of MOKO Button application (Android)***

## 3.19.2 iOS – MOKO APP DFU instructions

**Step 1:** Install iTunes in PC and then import firmware upgrade package into ***"MK Button"*** APP, you can refer to below picture demonstrations – ***Figure 35: Load upgrade package to nRF connect APP via iTunes.***

**Step 2:** Connect with device and go to "***SETTING***" interface, press "***DFU***" option and then firmware upgrade package will be shown automatically in DFU path.

**Step 3:** Choose the required upgrade firmware package and the upgrade process will start automatically. Wait for the upgrade to complete, APP will disconnect with Beacon after upgrade completed successfully.

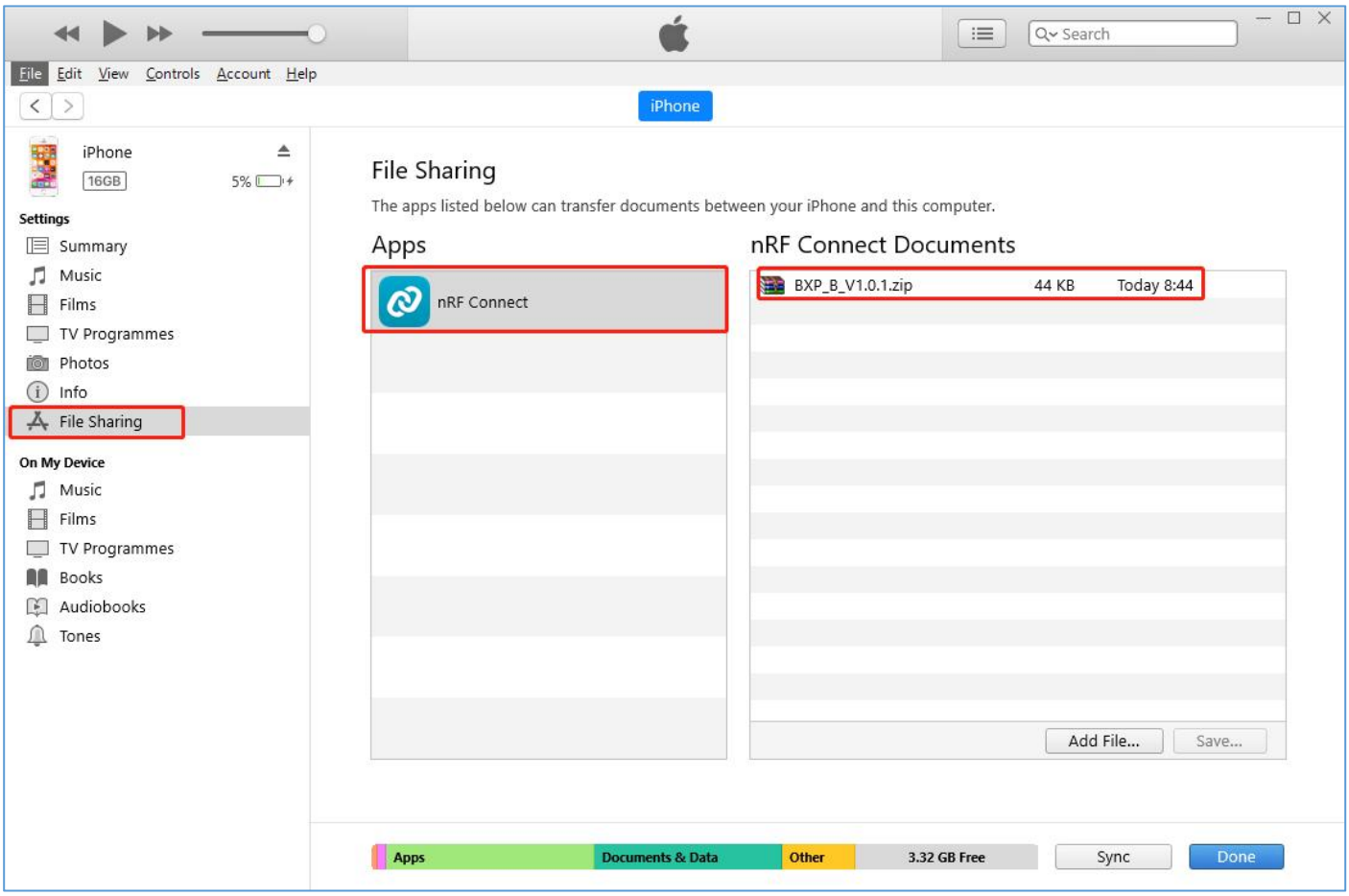



***Figure 24: Load upgrade package to nRF connect APP via iTunes***

## 3.20 Obtain log file from Android APP

Sometimes when testing with our APP, there may occurred some issues that cannot be reproduced in MOKO side, then we will need you to help obtain the log file from android APP and then provide for us to further analysis.

Currently only Android APP support log record, and the ***LOG*** option is in "***About***" interface, you can export the log file and then send to us by following below steps if rare issue happened.

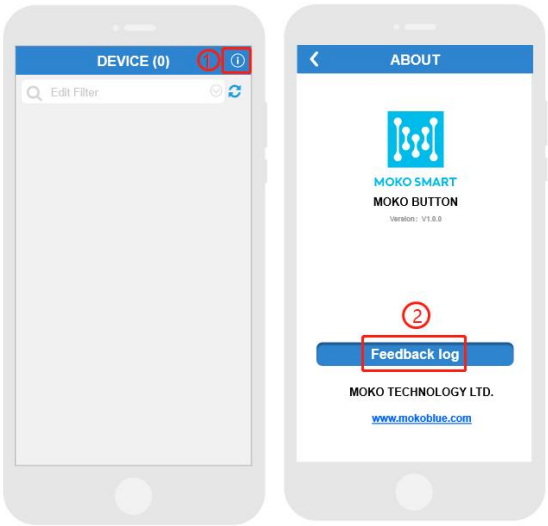



***Figure 25: Obtain log file from Android APP***

# 4. Common instructions of “nRF connect” APP

In case of some customers who using the “nRF connect” APP to compatible with different vendors’ Beacon, here we have listed some common instructions of ***android “nRF connect” APP*** as a template.

## 4.1 How to unlock device (password verification)?

Due to most operations based on device unlocked status, so here described the steps to unlock device based on ***Android nRF Connect*** APP in order to continue other operations further.

For ***iOS nRF Connect APP***, you can also follow this similar steps with same *SERVICES* and *CHARACTERISTICS*. This chapter will not give example of password verification for ***iOS nRF Connect APP*** again.

**Step 1:** ***Connect*** device - ①.

**Step 2:** Click “***configuration Service***”- ***UUID: 0000AA00-0000-1000-8000-00805f9b34fb*** and then turn on the *“notify”* property in ***unlock*** characteristic - ***UUID: 0000AA07-0000-1000-8000-00805f9b34fb*** - ② first.

**Step 3:** Write ***unlock*** property - ③ and there will have a prompt. In this step , you need to convert password (ASCII) into HEX first and then input in the option box. In this step , we have adopted customized password verification protocol, and the details as below:

- Password verification protocol : EA01550846D6F6B6F34333231
  - “EA0155” ---- Fixed protocol header.
  - “08” ---- Password length (need to be coordinate with password).
  - “4D6F6B6F34333231” ---- Password ASCII “Moko4321” in HEX format.

For instance： If you are using default connection password in the firmware – “Moko4321” (ASCII), the corresponding HEX is *“4d6f6b6f34333231”*, then you will need to input ***“EA0155084D6F6B6F34333231”*** accordingly.

**Step 4:** After unlock value input and then it will shows ***“EB-01-55-01-AA”*** to indicate it is unlocked.

***NOTE:*** *If you disabled password verification in the Beacon, it will be no need to do “unlock device” operations.*

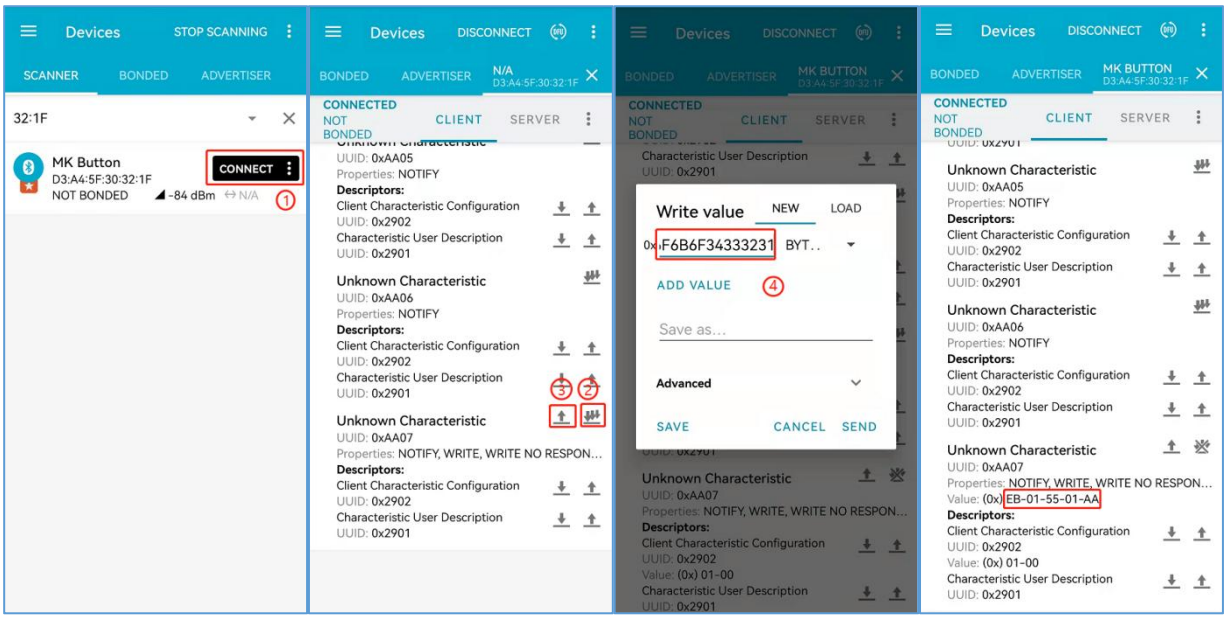



***Figure 26: Workflow of unlocking device in android nRF Connect APP***

## 4.2 How to modify device name?

**Step 1:** ***Connect*** device.

**Step 2:** ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device***.

**Step 3:** Modify device name

① Assume that you will need to modify device name into “Moko4321”, then convert into HEX – “4d6f6b6f34333231”

② Find “***ADV Slot Data***” characters in “***Eddystone configuration Service***”, then input “404d6f6b6f34333231” and press OK. Device name will be shown as “Moko4321”.

## 4.3 How to upgrade firmware through “nRF connect” APP?

### 4.3.1 Android – “nRF connect” APP DFU instructions

**Step 1:** Copy the upgrade package to a folder in the root directory of the smartphone.

**Step 2:** Turn on the Bluetooth of the smartphone and use the ***nRF Connect*** APP to scan the Beacon with the default name of ***“MK Button”***. (You can use the filtering function of the APP to filter the Beacon name)

**Step 3:** ***Connect*** device.

**Step 4:** ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device*.**

**Step 5:** Click the ***DFU*** button on the top of the screen and select the ***Distribution packet (ZIP)*** from the file type list, and then select the upgrade package from the smartphone folder.

**Step 6:** The upgrade process will start automatically and wait for the upgrade to complete, APP will disconnect with Beacon after upgrade completed successfully.

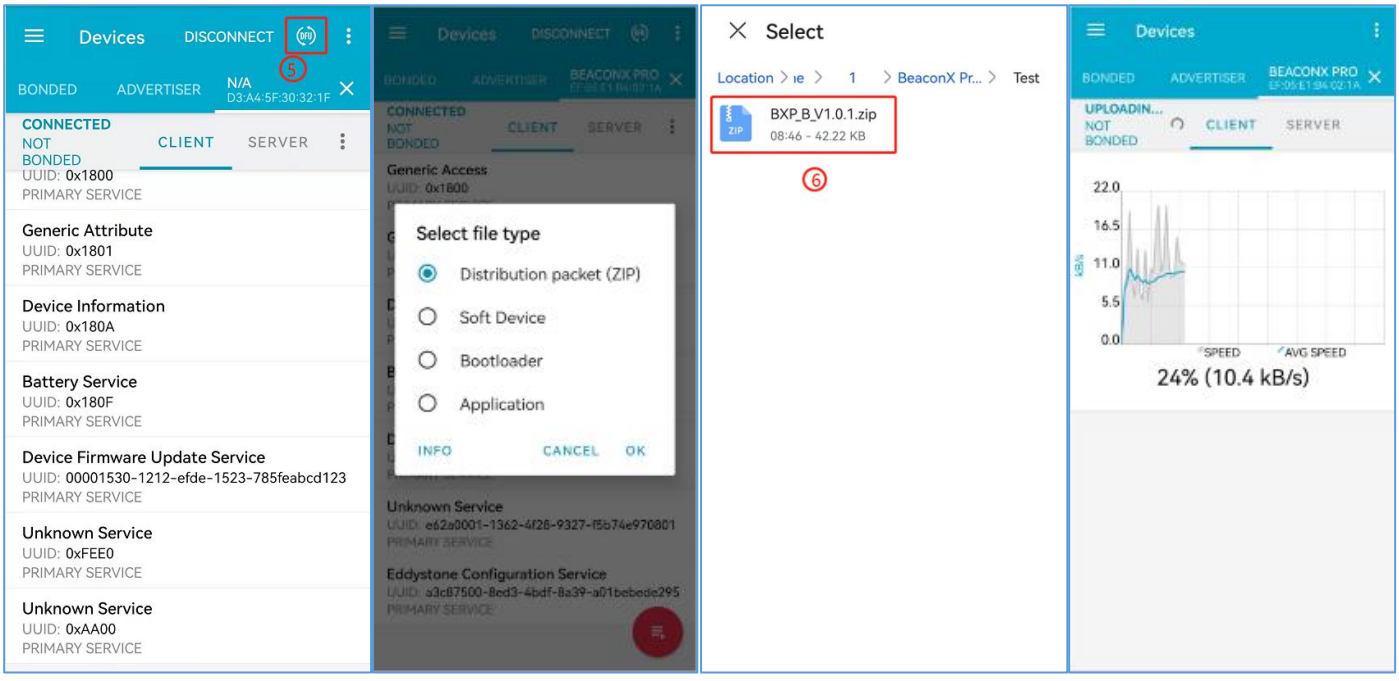



***Figure 27: Firmware upgrade workflow of Android nRF Connect APP***

## 4.3.2 iOS – "nRF connect" APP DFU instructions

**Step 1:** Use the desktop application ***iTunes*** to load the upgrade package to the *nRF Connect* App. You need connect your iPhone to your computer using the USB cable. Click your device in iTunes and then click ***File Sharing*** in the left sidebar of iTunes. You can find the *nRF Connect* on the Apps list and then drag and drop the upgrade package from a folder or window onto the ***Documents*** list to copy it to the *nRF Connect* App on your smartphone. Please refer to ***Figure 35: Load upgrade package to nRF connect APP via iTunes.***

**Step 2:** Turn on the Bluetooth of the smartphone and use the ***nRF Connect*** APP to scan the Beacon with the default name of ***"MK Button"***. (You can use the filtering function of the APP to filter the Beacon name)

**Step 3:** ***Connect*** device and then ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device.***

**Step 4:** Click the ***DFU*** button on the top of the screen and select the upgrade package.

**Step 5:** Click the ***Start*** button on the bottom of the screen and the upgrade process will start automatically. Wait for the upgrade to complete, APP will disconnect with Beacon after upgrade completed successfully.

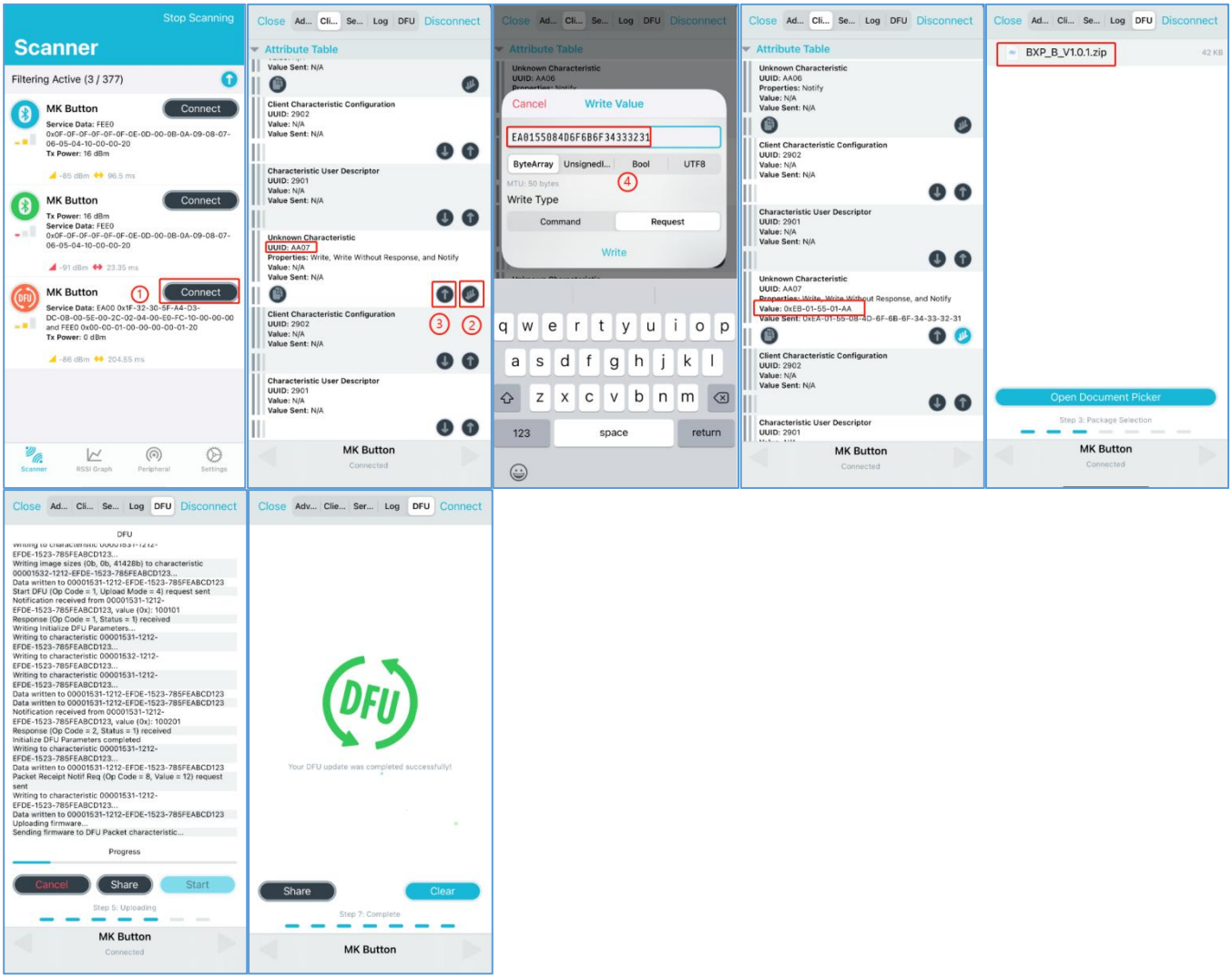



***Figure 28: Firmware upgrade workflow of iOS nRF Connect APP***

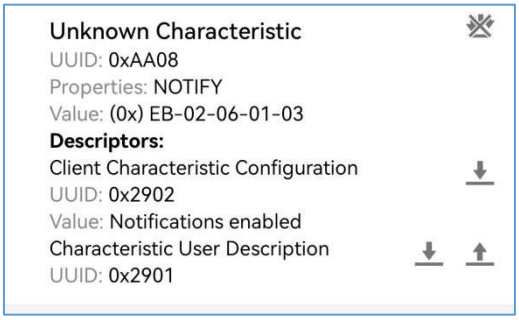
## 4.4 How to listening button event in Long connection mode

**Step 1:** ***Connect*** device.

**Step 2:** ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device***.

**Step 3:** Find "***AA08***" characters, then enable the notify property.

*For exmaple: EB 02 06 01 03 (The last byte indicates the number of button click times)*




## 4.5 How to clear button click times in Long connection mode

**Step 1:** ***Connect*** device.

**Step 2:** ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device***.

**Step 3:** Find "***AA08***" characters, then enable the notify property.

*RX: EB 02 06 01 03 (The last byte indicates the number of button click times)*

**Step 4:** Find "***AA01***" characters,

① Enable the notify property of ***AA01***.

② Write the remote alarm cancellation protocol to the feature value ***AA01***

*TX: EA 01 41 00*

*RX: EB 01 41 01 AA (The last byte return: AA -- Success ; 00 -- Fail)*

**Step 5:** "***AA08***" characters return: *EB 02 06 01* ***00***

## 4.6 How to configure broadcast battery voltage (Alarm info)

**Step 1:** ***Connect*** device.

**Step 2:** ***Unlock*** device. Please refer to ***chapter 4.1 – How to unlock device***.

**Step 3:** Find "***AA01***" characters,

① Enable the notify property of ***AA01***.

② Write the remote alarm cancellation protocol to the feature value ***AA01***

*TX: EA 01 61 01 01 (00: battery percentage; 01 - battery voltage)*

*RX: EB 01 61 01 AA (The last byte return: AA -- Success ; 00 -- Fail)*

# 4.7 How to parse button trigger event in Advertising mode?

If you want to parse the button trigger events in the advertising mode, it is recommended to enable the channel's "***Alarm mode***" and "***Stay advertising before trigger***" function. This allows the device to broadcast both before and after a button event occurs. To distinguish whether the button event has occurred, **you can determine it through the relevant bytes in the broadcast format**. The methods for determining different broadcast types are as follows:

- ***Alarm info:*** Bit1 of the Byte8 in the Alarm info frame in the broadcast format.

| Byte offset | Field | Example Value | Description |
|---|---|---|---|
| 0 | Data length | 0x02 | ADV length of "*Flags*" content |
| 1 | Data type | 0x01 | ADV type: *Flags* |
| 2 | Advertising type | 0x06 | BR/EDR not supported / LE general discoverable mode |
| 3 | Data length | 0x0C | ADV length of "*Service Data*" content |
| 4 | Data type | 0x16 | ADV type: *Service Data* |
| 05-06 | 16bit Service UUID | 0xE0 FE | MOKO-Defined UUIDs (little endian) |
| 7 | Frame type | 0x21 | MOKO-Defined advertisement frame type;<br>***0x20:*** "*Single press mode*" advertisement<br>***0x21:*** "Double *press mode*" advertisement<br>***0x22:*** "Long *press mode*" advertisement<br>***0x23:*** "*Abnormal inactivity mode*" advertisement<br>***0x24 – 0x3F:*** *RFU* |
| 8 | Status flag | 0x01 | ***Bit 0:*** Password verification status. *0:*Password verification disabled; *1:* Password verification enabled<br>***Bit 1:*** Trigger status of alarm mode. *0:* Alarm not be triggered. *1:* Alarm be triggered.<br>***Bit 2 - Bit 7:*** Reserved for future use. |

- ***iBeacon:*** The broadcast parameters after the trigger will automatically increment the first byte of the iBeacon UUID parameter you configured by 1.

| ***iBeacon parameters*** | | ***Before the button trigger event occurs*** | ***After the button trigger event occurs*** |
|---|---|---|---|
| ***UUID*** | ***01****BBCCDD-AABB-CCDD-AABB-CCDDAABBCCDD* | ***01****BBCCDD-AABB-CCDD-AABB-CCDDAABBCCDD* | ***02****BBCCDD-AABB-CCDD-AABB-CCDDAABBCCDD* |
| ***Major*** | ***1*** | ***1*** | ***1*** |
| ***Minor*** | ***1*** | ***1*** | ***1*** |

- ***Eddystone-UID:*** After the button trigger event, the last two bytes of the Eddystone-UID broadcast frame will be 0x00 01. Before the button trigger event, the last two bytes of the Eddystone-UID broadcast frame is 0x00 00.

| Byte offset | Field | Example Value | Description |
|---|---|---|---|
| 0 | Data length | 0x02 | AD length of Flags content |
| 1 | Data type | 0x01 | AD type: Flags |
| 2 | Advertising type | 0x06 | BR/EDR not supported / LE general discoverable mode |
| 3 | Data length | 0x03 | AD length of Complete List content |
| 4 | Data type | 0x03 | AD type: Complete List of 16-bit Service Class UUIDs |
| 05-06 | Service UUID | 0xAA FE | Google Eddystone UUIDs |
| 7 | Data length | 0x17 | AD length of Service Data content |
| 8 | Data Type | 0x16 | AD type: Service Data |
| 09-10 | Service UUID | 0xAA FE | Google Eddystone UUIDs |
| 11 | Frame type | 0x00 | Google Eddystone frame type, 0x00: UID |
| 12 | RSSI@0m | 0x00 | Calibrated Tx power at 0 m, defined by user |
| 13-22 | Namespace ID | 0x01 02 03 04 05 06 07 08 09 0A | 10-bytes Namespace ID, defined by user |
| 23-28 | Instance ID | 0x10 20 30 40 50 60 | 6-bytes Instance ID, defined by user |
| 29-30 | RFU | 0x00 00 | 2-bytes reserved for future use, must be 0x00 00 |

# 5. FAQ

***General Questions***

- *What is RSSI?*

The RSSI (Received Signal Strength Indication) is a measurement of the power present in a received radio signal. It is usually displayed in dBm (Decibel-milliwatts). Low power devices, like Bluetooth Low Energy devices, are using very week power to transfer data, therefore the usual values of RSSI are between -100dBm (Very week signal) to around -20 dBm (strong signal).

- *Why the RSSI values are different on two Android devices.*

By definition the RSSI value is a measurement of the power present in received radio signal. The signal strength depends on the antennas, device orientation, location, distance from the phone to target device, humidity etc.

- *What's the difference between Advertising data and Scan Response data?*

In Bluetooth 4+ device, if not connected, continuously advertises with the Advertising data. When a scanner receives it, it may send the Scan request packet to obtain more advertising information. Then, the advertising device sends the Scan response data. Both packets may have up to 31 bytes.

***APP Questions***

- Why the device name shows N/A?

Device name is contained in scan response packet in "*Device info*" advertisement frame type, if "*Device info*" slot is not set, then APP will not achieve the device name and will shows N/A accordingly.

# Appendix

## A.1 Frame type of MOKO defined advertisement

| Service UUID | Frame type | HEX value |
|---|---|---|
| 0xE0 FE | Single press mode | 0x20 |
| | Double press mode | 0x21 |
| | Long press mode | 0x22 |
| | Abnormal inactivity mode | 0x23 |
| 0x00 EA | General device information | 0x00 |

***Table 8: Frame type of MOKO defined advertisement***

## A.2 Sampling rate comparison table of 3-axis accelerometer sensor

| Sensor model | Sampling rate | HEX value |
|---|---|---|
| LIS3DH | 1Hz | 0x00 |
| | 10Hz | 0x01 |
| | 25Hz | 0x02 |
| | 50Hz | 0x03 |
| | 100Hz | 0x04 |

***Table 9: Sampling rate comparison table of 3-axis accelerometer sensor***

## A.3 Full-scale comparison table of 3-axis accelerometer sensor

| Sensor model | Full-scale | HEX value |
|---|---|---|
| LIS3DH | ±2g | 0x00 |
| | ±4g | 0x01 |
| | ±8g | 0x02 |
| | ±16g | 0x03 |

***Table 10: Full-scale comparison table of 3-axis accelerometer sensor***

# Contact

**MOKO TECHNOLOGY LTD.** An original manufacturer for IoT smart devices

***Address:*** *4F, Building 2, Guanghui Technology Park, MinQing Rd, Longhua, Shenzhen, Guangdong, China*

***E-mail:*** *Support_BLE@mokotechnology.com*

***Website:*** *www.mokosmart.com*

*www.mokoblue.com*