**MR7901 Positing Base Station Communication Protocol**

V1.6

（2017.09.19）

**SHENZHEN MARKTRACE CO.,LTD**

Contents

[Contents 1](#_Toc493668373)

[1. Purpose 3](#_Toc493668374)

[2. Normative references 3](#_Toc493668375)

[3. Need to explain 3](#_Toc493668376)

[4. Communication process 4](#_Toc493668377)

[5. Communication packet format 6](#_Toc493668378)

[5.1 Packet format 6](#_Toc493668379)

[5.2 Message header 6](#_Toc493668380)

[5.3 Service Content 7](#_Toc493668381)

[6. Data interaction 8](#_Toc493668382)

[6.1 Registered （0x0008/ 0x8008） 8](#_Toc493668383)

[6.2 Logon（0x0001/ 0x8001） 10](#_Toc493668384)

[6.3 Heartbeat （0x0003/ 0x8003） 13](#_Toc493668385)

[6.4 Data reporting （0x0004/ 0x8004） 17](#_Toc493668386)

[6.5 Firmware update（0x000D/ 0x800D） 19](#_Toc493668387)

[6.6 Configuration parameters （0x000A/ 0x800A） 27](#_Toc493668388)

[7. TLV Index 44](#_Toc493668389)

[7.1 TLV Type list and format 44](#_Toc493668390)

[7.2 Tag format description 46](#_Toc493668391)

[8. Configuration Parameter Format Description 47](#_Toc493668392)

[9. Command List 49](#_Toc493668393)

[10. Check algorithm 50](#_Toc493668394)

[10.1 CRC16 Check algorithm 50](#_Toc493668395)

[10.2 Checksum algorithm 53](#_Toc493668396)

[11. appendix 55](#_Toc493668397)

[11.1 The relationship between the battery charge and the battery voltage 55](#_Toc493668398)



# Purpose

This document mainly introduces the communication format and precautions between MR7901 and platform server, and provides reference for engineers to complete MR7901 platform software development.

# Normative References

1. Common access standard for industrial IOT equipment.--SHENZHEN MARKTRACE CO.,LTD

# **Need to explain**

1. MR7901 and platform direct communication methods are: GPRS, LAN or other means；
2. Server: Platform software；
3. Terminal / Equipment: MR7901；
4. Load Balancing Server: The server that MR7901 sends the registration message；
5. Load server: The server that MR7901 sends messages such as login, heartbeat, data, configuration, etc.

# Communication process

The communication uses the "client" - "server" mode, Platform is server and MR7901 is client. The communication is initiated by the client to the server.

After MR7901 power on, you need to submit a registration application to the platform. After the registration is successful, you can log in to the platform. After the login is successful, the data can be exchanged. Data interaction is initiated by the device, the platform responds. The following is a business process diagram:



Examples of communication processes

# Communication packet format

## **Packet format**

The communication packet between the terminal and the platform is composed of four parts: the Start Flag, the Message Header, the Service Content and the Check. The start flag is fixed to 0x55, 0xAA.The check is the CRC16 check of the header and the service content,

The format is as follows

|  |  |  |  |
| --- | --- | --- | --- |
| Start Flag | Message Header | Service Content | Check |
| 2 Bytes | 28 Bytes | Variable length | 2 Bytes |

explanation ：

1. Start Flag: fixed to 0x55,0xAA
2. **The length of the packet header is fixed to 28Bytes;**
3. **The length of the** Service Content **is variable;**
4. Check **is in accordance with CRC16 CCITT standard -0x1021 (initial value is 0xFFFF),**check  **algorithm is described in the following sections**
5. 。Check  **is the calculation of the header and the** service content **part**

## **Message header**

The packet header is composed of message length, command code, protocol version, terminal serial number (or device ID).The header format is as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Message Header（28 Bytes）** | | | |
| **SN** | **Field** | **Length (byte)** | **Description** |
| 1 | Total message length | 2 | Contains from the beginning itself, until the end of the message (only the header and the service content) |
| 2 | Command code | 2 | Indicates the command to be executed or answered, such as login, data reporting, update, and so on. |
| 3 | Message serial number | 4 | 0x00000000 to 0xFFFFFFFF， The sender to maintain their own serial number, each successful communication, automatically add 1. |
| 4 | Message protocol version | 2 |  |
| 5 | Message security flag | 2 | Unencrypted messages default to 0x0000 |
| 6 | Device ID | 16 | 16-bit ASCII code |

explanation：

Command code definition format: The highest bit of the command code sent by the MR7901 to the platform is 0, and the command code that platform responds to the MR79601 at the highest position 1.For example, if the command code of device initiates a registration request to the platform is 0x0008, and the command code for the platform response is 0x8008.

egg：55 AA 00 22 00 08 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 01 78 56 34 12 A7 5C

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 22 | 00 | 08 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 00 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **desc\_code(H)** | **desc\_code (L)** | **reg\_code (MSB)** |  |  | **reg\_code (LSB)** |
| 39 | 36 | 00 | 01 | 01 | 78 | 56 | 32 | 12 |
| **crc16(H)** | **crc16(L)** |
| A7 | 5C |

among them, message header is： 00 22 00 08 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00

## Service Content

The service content, according to the different command code, contains the service content is different, the specific format look at the instructions behind.

# Data interaction

**Note：**

When device send message to server, the server must have a corresponding response, otherwise the device will repeatedly send the same message.

## R**egistered** （0x0008/ 0x8008）

### **Command frame definition**

Initiated by the device to the platform, the command code: 0x0008, platform confirmation code: 0x8008.

The device sends a registration message to the server (load balancing server)

After the server receives, respond to the registration status with the load server's IP and port.

**Note**：

If the device registration is unsuccessful, the registration message will continue to be sent.

The service content of the device registration includes a 2-byte device type description and a 4-byte registration code. The registration code is calculated by the device ID through a fixed algorithm. The specific algorithm is defined separately by the system.

**Command code**： 0x0008

**Service content**： As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| Device description | 2 | The high byte is the device type  **0x01** ——Data gateway  **0x02** —— RFID reader  **0x03** —— Computer  The low byte is the device model code  **0x01** —— MR7901 |
| Registration code | 4 | Reserved |

egg: 　55 AA 00 22 00 08 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 01 78 56 34 12 A7 5C

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 22 | 00 | 08 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 00 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **desc\_code(H)** | **desc\_code (L)** | **reg\_code (MSB)** |  |  | **reg\_code (LSB)** |
| 39 | 36 | 00 | 01 | 01 | 78 | 56 | 32 | 12 |
| **crc16(H)** | **crc16(L)** |
| A7 | 5C |

**Start flag**

**Start flag** sof ： 0x55AA

**Message header**

message length len ： 0x0022

command code cmd ： 0x0008

Message serial number seq ： 0x00000000

protocol version pro\_ver ： 0x0001 (V0.1)

security flag seq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Change to a string：“861694034205896”）

**Service content**

Device description desc\_code ：01 01 （device type：01 IOT gateway，model type：01）

Registration codereg\_code ：78 56 34 12

**Check**

crc16 : 0xA75C

### **Platform validation package definition**

Platform validation service content includes the registration result and the current real time, and the load server IP and port assigned to the device login.

**Validation code**： 0x8008

**Service content**：

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| Registration results | 1 | **0x00** ——Registered successfully (if the registration is successful, return to the load server IP and port)  **0xFE** ——Registration code error  **0xFF** ——Registration refused |
| Real time | 6 | Year Month Day Hours Minutes Seconds, year is 2000 |
| IP Load server IP | 32 | String type  egg： “218.17.157.214” |
| Load the server port | 2 | Unsigned integer  Low byte first, high byte in the post  eg:4501 |

egg： 55 AA 00 45 80 08 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 00 11 01 0E 11 16 32 32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 24 13 B6 41

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 45 | 80 | 08 | 00 | 00 | 00 |
| **seq**  **(LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev id(MSB)** |  |  |  |
| 00 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev id(LSB)** | **reg\_status** | **time（MSB）** |  |  |  |  |
| 39 | 36 | 00 | 00 | 11 | 01 | 0E | 11 | 16 |
| **time（LSB）** | **IP(MSB)** | **…** | **IP(LSB)** | **port(H)** | **port(L)** | **crc16(H)** | **crc16(L)** |
| 32 | 32 | … | 00 | 24 | 13 | B6 | 41 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0045

command code cmd ： 0x8008

Message serial number seq ： 0x00000000

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Registration results reg\_status ： 00 registration success

Time time ： 11 01 0E 11 16 32 The current time of the platform, corresponding to the year, month, day, hour, minute, second, the starting time is 2000 (January 14, 2017, 17:22:50)

Load server IP : 32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 (“218.17.157.214”)

Load server port : 24 13 is 0x1324 （Decimal for 4900）（Note: low byte first）

**Check**

crc16 : 0xB641

## Logon （0x0001/ 0x8001）

Initiated by the device to the platform，command code:0x0001，platform validation code：0x8001。

After the device sends a registration message to the load balancing server, the load balancing server issues the ip and port of the load server,After receiving the ip and port of the load service, the device sends a login message to the ip and port server (load server).

**Note：**

If the device fails to log in (without receiving the correct response), it will send 10 logon request messages (command code0x0001) repeatedly. After 10 times, the device will send a registration message again to the load balancing server.

### **Command package definition**

Login service content includes the software version number of the device and the CRC16 check of the device's configuration parameter list.

**command code**： 0x0001

**Service content**： As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| software version | 2 | main-version number and minor-version number |
| CRC16 check of the device's configuration parameter | 2 | The CRC16 check of the configuration parameter list. The server may decide whether or not to update the parameters accordingly. |

eg: 　 55 AA 00 20 00 01 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 02 07 02 69 C5 0C

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 20 | 00 | 01 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 00 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (MSL)** | **ver (H)** | **ver (L)** | **parm\_crc16(H)** | **parm\_crc16(L)** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 02 | 07 | 02 | 69 | C5 | 0C |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0020

command code cmd ： 0x0001

Message serial number seq ： 0x00000000

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

software version ver ：02 07 （device software version is V2.7）

CRC16 check of the device's configuration parameter：02 69

**Check**

crc16 : 0xC50C

### **Platform validation package definition**

The platform validation service content includes the login result (1 byte) and the current real time

**Validation code**： 0x8001

**Service content**： As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| Log in results | 1 | **0x00** ——Login successful, no operation request  **0x02** ——The login is successful and requires the firmware to be updated  **0x03** ——Login is successful, request to upload device hardware information  **0x10** ——The login is successful and requires updating the user configuration parameters  **0xFE** ——Login error  **0xFF** ——Login refused (the device received a deny login message, 3 minutes later to send a registration message) |
| Real time | 6 | Year Month Day Hours Minutes Seconds, year 2000 |

eg: 　 55 AA 00 23 80 01 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 00 11 01 0E 11 17 1E F6 C5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 23 | 80 | 01 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 00 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **login\_status** | **time（MSB）** |  |  |  |  |
| 39 | 36 | 00 | 00 | 11 | 01 | 0E | 11 | 17 |
| **time（LSB）** | **crc16 (H)** | **crc16 (L)** |
| 1E | F6 | C5 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0023

command code cmd ： 0x8001

Message serial number seq ： 0x00000000

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Login status login\_status ：00 （login successful）

Real time time ：11 01 0E 11 17 1E corresponding year, month, day, hour, minute, second, the starting time is 2000 (January 14, 2017, 17:23:30)

**Check**

crc16 : 0xF6C5

The validation package of requests device update

eg2: 55 AA 00 23 80 01 00 00 00 00 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 02 11 01 14 10 14 22 70 96

## Heartbeat（0x0003/ 0x8003）

Initiated by the device to the platform，command code:0x0003，plat validation code：0x8003。

**Note：**

If the device does not receive the correct response message after sending the heartbeat message, it will send 5 times the heartbeat message (command code0x0003). After 5 times, the device will send the login message (command code0x0001) again to the load service.

### Command package definition

The heartbeat service content includes the working status of the device (2 bytes) and the current state of the device.

**command code**： 0x0003

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| Device working status | 2 | **format specification**：  Low 4 byte, connection:：  The 0bit : GPRS connection，1 effective  The 1bit： LAN connection，1 effective  The 2bit： reserved  The 3bit： reserved  The 4bit： Tag transmission flag ，0：transfer tag records to platform；1： Do not transfer tag records to the platform  The 5bit： Device power failure flag，为0： device normal power supply，1： device external power supply is disconnected  The 6～7bit：reserved  The 8～11bit： battery voltage，0～10， respectively, there are 0 to 100% of the electricity. (This function will be achieved in MR7901 V3.2 or above version, the corresponding relationship between battery and battery voltage refer to Chapter 11)  The 12～15bit：reserved  **eg**：  00 01 ——GPRS connection, reported tag data  00 11 ——GPRS connection, do not report tag data  08 03 ——80%GPRS connection, wired connection, report tag data, battery power 80% |
| Device status | 2 | （E-bike V2.9,V3.0 and above）  0～7bit ： gprs signal strength, the normal range: 0 ~ 31, if value is 99, the access signal strength failure  8～11bit： Packet transmission mode (only one bit valid)  8bit ： Send packets through gprs  9bit ： Send packets over a wired network  10bit ： reserved  11bit ： reserved  12～15bit：reserved  eg:  11 01 ——gprs signal strength 17,send this heartbeat packet via gprs |
| Device version | 2 |  |
| Device time | 6 | Year, month, day, hour, minute, second |

eg: 　 55 AA 00 28 00 03 00 00 00 01 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 00 01 11 95 02 07 11 01 0E 11 17 1D DE 46（V2.6,V2.7，V2.8）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 28 | 00 | 03 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 01 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **work\_status(H)** | **work\_status(L)** | **parm\_crc16(H)** | **parm\_crc16(L)** | **ver(H)** | **ver(L)** |
| 39 | 36 | 00 | 00 | 01 | 11 | 95 | 02 | 07 |
| **time（MSB）** |  |  |  |  | **time（LSB）** | **crc16 (H)** | **crc16 (L)** |
| 11 | 01 | 0E | 11 | 17 | 1D | DE | 46 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0028

command code cmd ： 0x0003

Message serial number seq ： 0x00000001

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

device work status work\_status: 00 01 (gprs connection, reporting tag)

CRC16 check of the device's configuration parameter：11 95 （Temporarily unused）

software version ver ：02 07 （device software version V2.7）

device time time ：11 01 0E 11 17 1D Respectively, the year, month, day, hour, minute, second, the starting time is 2000 (January 14, 2017, 17:23:29)）

**Check**

crc16 : 0xDEC6

Following heartbeat package is V3.2 and above version’s

eg2: 　 55 AA 00 28 00 03 00 00 00 03 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 0A 01 10 01 03 02 11 06 01 0F 2C 28 BD CE

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0028

command code cmd ： 0x0003

Message serial number seq ： 0x00000003

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

device work status work\_status: 0A 01

0x0A01：gprs connection

Upload the tag data

External power supply is normal

Battery power is 10, means 100%

device status ：10 01

0x10 : gprs signal strength, is 16

0x01 : gprs transmission

software version ver ：03 02 （device software version V3.2）

Device time time ：11 06 01 0F 2C 28 Respectively, the corresponding year, month, day, hour, minute, second, the starting time is 2000 (June 1, 2017, 15:44:40)

**Check**

crc16 : 0xBDCE

### **Platform validation package definition**

Platform validation service content includes the operation indication (1 byte) and the current real time time (6 bytes, year, month, hour, minute, year 2000)

**validation code**： 0x8003

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| Operation instructions | 1 | **0x00** ——There is no operational indication  **0x02** ——Requires updating firmware  **0x03** ——Reset device (device received, do not respond, restart directly)  **0x04** ——Update the antenna firmware  **0x05** ——Get antenna information (version and gain)  **0x06** ——Set device time (after the device is received, set the time, do not respond)  **0x08** ——Clear tag data buffer (after the device is received, clear tag data buffer, do not respond)  **0x10** ——Request to update the user configuration parameters (V2.7 above supports)  **0x11** ——Get device status ( V2.8 and above version supports)  **0x12** ——Request to upload device hardware information (V3.1and above version supports) |
| Real time | 6 | year, month, hour, minute,second. year is 2000 |

eg: 　 55 AA 00 23 80 03 00 00 00 01 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 00 11 01 0E 11 17 1F 70 FC

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 23 | 80 | 03 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 01 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **ack** | **time（MSB）** |  |  |  |  |
| 39 | 36 | 00 | 00 | 11 | 01 | 0E | 11 | 17 |
| **time（LSB）** | **crc16 (H)** | **crc16 (L)** |
| 1F | 70 | FC |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0023

command code cmd ： 0x8003

Message serial number seq ： 0x00000001

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Operation instructions ask ：00 No operation

Platform real time time ：11 01 0E 11 17 1F Respectively, the year, month, day, hour, minute, second, the starting time is 2000 (January 14, 2017, 17:23:31)

**Check**

crc16 : 0x70FC

## **Data reporting**（0x0004/ 0x8004）

Initiated by the device to the platform，command code:0x0004，plat validation code：0x8004。

**Note：**

If the device sends a data message but does not receive the correct response message from the platform, it will send 5 times data messages (command code0x0004)

After 5 times,it will send login message to load server （command code0x0001）

### Command package definition

The data reporting service content includes several data TLVs

**command code**： 0x0004

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| TLV | 2+2+17 | Tag type (2Bytes), tag data length (2Btyes), tag data (length reference Chapter 7 TLV index)  Defined by the specific data content, the specific format see Chapter 7 (label format description) |
| TLV | 2+2+17 |  |
| …… |  |  |

TLV structure is as follows：

|  |  |  |
| --- | --- | --- |
| TLV type (2 bytes) | LENGTH(2 bytes) | VALUE(The length is defined by LENGTH） |

eg： 55 AA 00 46 00 04 00 00 00 8D 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 8B 01 00 11 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 8B 01 00 11 01 20 EB 14 4A 33 64 00 00 00 B8 11 01 0E 13 26 09 83 3F

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 46 | 00 | 04 | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 8D | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **TLV (MSB)** |  | **…** | **TLV (MSL)** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 8B | 01 | … | 09 | 83 | 3F |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0046

command code cmd ： 0x0004

Message serial number seq ： 0x0000008D

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

TLV data ：8B 01 00 11 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 8B 01 00 11 01 20 EB 14 4A 33 64 00 00 00 B8 11 01 0E 13 26 09 There are two tag records, the specific format, please see Chapter 7 label instructions)

The first TLV is: 8B 01 00 11 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 (as follows：

TLV type ： 0x8B01 RFID tag

TLV data length ： 0x0011

TLV data ： 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 The format is described in the tag data format)

**Check**

crc16 : 0x833F

### **Platform validation package definition**

**validation code**： 0x8004

**Service content**：same as heartbeat service content。

## **Firmware update**（0x000D/ 0x800D）

The device sends a firmware download request to the platform, command code: 0x000D, platform validation code: 0x800D.

When the platform needs to send new firmware (such as host firmware, antenna firmware) to the device, fill in the firmware update operation instruction code in the login package, heartbeat packet, and packet confirmation package. After the device receives the update operation instruction, the device requests the firmware update to the platform with command code0x000D.

### Command package definition

firmware update includes update host firmware and antenna firmware,service content definition as follows：

**command code**： 0x000D

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| file\_type | 1 | **0x01** ——host firmware  **0x02** ——antenna firmware  Other files are not yet defined |
| data\_type | 1 | mask flag：  **0x00** ——Application file information, including file version, file name, crc16 check, size;  **0x01** ——Application file content,when this bit is 1, the following two data segments (block size / block serial number) is valid;  **0x02** ——The upgrade is complete (followed by 3 bytes, where the first byte is the upgrade status, 0x01: the upgrade is successful, 0x00: the upgrade failed; the second and third bytes are the upgraded version information). |
| Block size | 2 | This section is valid when the application category is "0x01 - Request File Content"Each packet carries the file block size,Unit: Byte |
| Block index | 2 | This section is valid when the application category is "0x01 - Request File Content"  The file block number applied to the platform |

**1,Application file information**

When the device receives the upgrade instruction code, it sends the request message of the upgrade file information to the server, that is, the application file information.

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| file\_type | 1 | **0x01** ——host firmware  **0x02** ——antenna firmware  Other files are not yet defined |
| data\_type | 1 | mask flag：  **0x00** ——Application file information, including file version, file name, crc16 check, size; |
| Reserved | 4 | 00 00 00 00 |

**eg1：**55 AA 00 22 00 0D 00 00 00 01 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 00 00 00 00 00 5E 5E

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 22 | 00 | 0D | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 01 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **file\_type** | **data\_type** | **Reserved (MSB)** |  |  | **Reserved (LSB)** |
| 39 | 36 | 00 | 01 | 00 | 00 | 00 | 00 | 00 |
| **crc16 (H)** | **crc16 (L)** |
| 5E | 5E |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0022

command code cmd ： 0x000D

Message serial number seq ： 0x00000001

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Request upgrade file type **file\_type** : 01 (update host firmware)

Request data type data\_type : 00 (Request upgrade file information)

reserved Reserved ：00 00 00 00 （Temporarily unused）

**Check**

crc16 : 0x5E5E

**1,Application file content**

After receiving the correct upgrade file information from the platform, the device sends the "application file content" message to the platform.

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| file\_type | 1 | **0x01** —— host firmware  **0x02** —— antenna firmware |
| data\_type | 1 | **0x01** —— Application file content |
| block\_size | 2 | Transfer block size of update file.The server according to this value, read the size of the block\_size data sent to the device, the last packet is less than block\_size, filled with 0.  The version before V2.9 is 20 00 (512Bytes)  V2.9 and above version is 01 DD (477Bytes) |
| block\_index | 2 | The block index of file |

**eg2：**55 AA 00 22 00 0D 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 01 02 00 00 00 26 CD

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 22 | 00 | 0D | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **file\_type** | **data\_type** | **block\_size(H)** | **block\_size(L)** | **block\_index(H)** | **block\_index(L)** |
| 39 | 36 | 00 | 01 | 01 | 02 | 00 | 00 | 00 |
| **crc16 (H)** | **crc16 (L)** |
| 26 | CD |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0022

command code cmd ： 0x000D

Message serial number seq ： 0x00000002

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

request update file type **file\_type** : 01 (update host firmware)

request update data type data\_type : 01 (Request to upgrade the contents of the file)

block size block\_size ：0x0200 （send block size of update file content is 0x200Bytes）

Block index block\_index ：0x0000 (update block 0 on file）

**Check**

crc16 : 0x26CD

**1，The upgrade is complete**

After host and antenna update file transfer are completed, will send update successful message to server.( V2.7 and previous version will send this message only antenna update success.

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| file\_type | 1 | **0x01** ——host firmware  **0x02** ——antenna firmware |
| data\_type | 1 | **0x02** ——update completed |
| update\_flag | 1 | Used to mark whether the upgrade file is received correctly by the device  00 : Upgrade failed  01 : update successed |
| update\_ver | 2 | The firmware version after the upgrade(V2.8 and above version supports this function)  eg:  02 09 ： updated firmware is V2.9 |
| reserved | 1 |  |

**eg3 :** 55 AA 00 22 00 0D 00 00 00 5E 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 02 01 02 08 00 05 C6

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 22 | 00 | 0D | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 5E | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **file\_type** | **data\_type** | **update\_flag** | **update\_ver (MSB)** | **update\_ver (LSB)** | **Reserved** |
| 39 | 36 | 00 | 01 | 02 | 01 | 02 | 08 | 00 |
| **crc16 (H)** | **crc16 (L)** |
| 05 | C6 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x0028

command code cmd ： 0x000D

Message serial number seq ： 0x0000005E

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

request update file type **file\_type** : 01 (update host firmware)

request update data type data\_type : 02 (update completed)

Upgrade the reception completion flag update\_flag : 01 （Receive the upgrade file successfully）

updated firmware version update\_ver : 02 08（The version of the upgrade file received is V2.8）

Reserve **Reserved** : 00 (reserve）

**Check**

crc16 : 0x05C6

### **Platform validation package definition**

Platform validation service content based on content application

**validation code**： 0x800D

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| data\_type | 1 | **0x01** ——Host file basic information  **0x02** ——Host file content data  **0x03** ——Antenna file basic information  **0x04** ——Antenna file content data  **0x05** ——The upgrade file ends |
| File basic information | 16 | When the data type is 0x01 / 0x03, this data segment exists  5 bytes file name, 3 byte extension, 2 byte file version, 2 byte file crc16 check (reserved), 4 byte file size |
| data length | 2 |  |
| data |  |  |

**1，Response to application file information**

When the platform receives the "Request File Message" sent from the device, the information of the upgrade file is sent to the device.

**Service content**：As shown in the table below

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| data\_type | 1 | **0x01** ——Host firmware file basic information  **0x03** ——Antenna firmware file basic information |
| file\_name | 5 |  |
| file\_ext | 3 |  |
| file\_version | 2 |  |
| file\_crc16 | 2 | The overall CRC16 checksum for the bit upgrade file  (See Chapter 9 for the calibration algorithm) |
| file\_size | 4 | High byte first |

**eg1：**55 AA 00 2D 80 0D 00 00 00 01 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 01 00 00 00 00 00 00 00 00 00 00 B5 87 00 01 0A EE 5F 29

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 2D | 80 | 0D | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 01 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **data\_type** | **file\_name** | **file\_ext** | **file\_version** | **file\_crc16** | **file\_size** |
| 39 | 36 | 00 | 01 | 00 00 00 00 00 | 00 00 00 | 00 00 | B5 87 | 00 01 0A EE |
| **crc16 (H)** | **crc16 (L)** |
| 5F | 29 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x002D

command code cmd ： 0x800D

Message serial number seq ： 0x00000001

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

request update file type **file\_type** : 01 (update host firmware)

update file name file\_name : 00 00 00 00 00 (5Byes, update file name)

Upgrade file extension file\_ext ：00 00 00 （3Byes, Upgrade the file extension）

Upgrade the file version file\_version ：00 00 （2Byes, Upgrade the file version）

Upgrade file crc16 check file\_crc16：B5 87 (2Byes, Upgrade file crc16 check is 0xB587）

Upgrade file size file\_size ：00 01 0A EE （4Byes, Upgrade file size is 0x00010AEE）

**Check**

crc16 : 0x5F29

**response application file content**

When the platform receives the "Request File Content" message from the device, the content of the upgrade file is issued according to the application file block size.

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| data\_type | 1 | **0x02** ——Host firmware file content data  **0x04** ——Antenna firmware file content data |
| block\_size | 2 | This package contains the block size of the upgrade file data  The version before V2.9 is 20 00 (512Bytes) V2.9 and above version is 01 DD (477Bytes) |
| data  update\_data | block\_size | update file data  The data length is determined by block\_size |

**eg2：**55 AA 02 1F 80 0D 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 02 02 00 F0 17 … 11 3C 49

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 02 | 1F | 80 | 0D | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **data\_type** | **block\_size(H)** | **block\_size(L)** | update\_data**（MSB）** | **…** | update\_data **(LSB)** |
| 39 | 36 | 00 | 01 | 02 | 00 | F0 | … | 11 |
| **crc16 (H)** | **crc16 (L)** |
| 3C | 49 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x021F

command code cmd ： 0x800D

Message serial number seq ： 0x00000002

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Respond to the requested data type **data\_type** : 02 (Host firmware file contents)

Block size block\_size : 02 00 (2Byes, The upgrade package size is 0x0200)

**Note：**

1. The version before V2.9 is 20 00 (512Bytes)

2. V2.9 and above version is 01 DD (477Bytes)

update file data update\_data ：F0 … 11（Upgrade the contents of the file）

**Check**

crc16 : 0x3C49

1. **Respond to upgrade complete**

After the device receives the upgrade file, it will send the upgrade complete message to the platform,The platform receives a confirmation message.

|  |  |  |
| --- | --- | --- |
| **Data segment** | **Bytes** | **Description** |
| data\_type | 1 | **0x05**——The upgrade file ends |

**eg3：**55 AA 00 1D 80 0D 00 00 00 18 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 05 47 04

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | | **cmd(L)** | | **seq (MSB)** |  |  |
| 55 | AA | 00 | 1D | 80 | | 0D | | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | | **dev\_id (MSB)** | |  |  |  |
| 18 | 00 | 01 | 00 | 00 | | 38 | | 36 | 31 | 36 |
|  |  |  |  |  | |  | |  |  |  |
| 39 | 34 | 30 | 33 | 34 | | 32 | | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **data\_type** | **crc16 (H)** | **crc16 (L)** | |
| 39 | 36 | 00 | 05 | 47 | 04 | |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x001D

command code cmd ： 0x800D

Message serial number seq ： 0x00000018

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

Response request data type data\_type : 05 (Upgrade complete confirmation)

**Check**

crc16 : 0x4704

## **Configuration parameters** （0x000A/ 0x800A）

Initiated by the device to the platform，command code:0x000A，platform validation code：0x800A。

When the platform needs to acquire or configure the device parameters, the operation instructions are filled in the login packet, heartbeat packet, and packet response. After receiving the operation instruction code, the equipment sends the corresponding parameter information to the platform.

### Command package definition

Service content includes parameter type and parameters

**command code**： 0x000A

**Service content**：As shown in the table below

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Data segment** | **Bytes** | **Description** |
| 1 | param\_type | 1 | **0x02** ——report antenna info（version，gain,rssi）  **0x10** ——report user parameters(V2.7 and above version support)  **0x11** ——report device statues(V2.8 and above version support)  **0x12** ——report device hardware info (V3.1 and above version support)  **0x80** —— report config confirmation (V2.8 and above support) |
| 2 | Parameters | x | **when the first byte is 0x02：**  12Bytes, Corresponds to the version of the four antennas、gain、rssi  **when the first byte is 0x03：**  1Byte, 1 Do not report tag data，0 report tag data  **when the first byte is 0x10：**  182Bytes, User configuration parameters, the specific format description see Chapter 8 configuration parameter format description  **when the first byte is 0x11：**  106Byte, The current status information for the device  **when the first byte is 0x12：**  127Byte, device hardware info  **when the first byte is0x80：**  1Byte, 1 The configuration parameters are successful，0 : Configuration failed |

1. **Report antenna info 0x02**

The antenna information contains the current version of the 4 antennas, gain, and filter the RSSI threshold.

**Service content**：As shown in the table below

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Data segment** | **Bytes** | **Description** |
| 1 | param\_type | 1 | **0x02** ——report antenna info |
| 2 | ant\_fw\_ver | 8 | 1 to 4 antenna version information, each antenna version in 2 bytes  If it is FF FF, it means that the antenna version information of the channel is not read |
| 3 | ant\_gain | 4 | 1 ~ 4 antenna gain (range 0 ~ 31) If it is FF, it means that the antenna gain of the channel is not read |
| 4 | ant\_rssi | 4 | 1 ~ 4 antenna filter threshold (range -1 ~ -128)  If it is 01, it means that the RSSI of the channel is not read |

eg1： 55 AA 00 2D 00 0A 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 02 01 06 01 06 01 06 FF FF 1F 1F 1F FF A8 A8 A8 01 83 F3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 2D | | 00 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  | |  |  |  |  |  |
| 39 | 34 | 30 | 33 | | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | | **ant1\_fw\_ver(H)** | **ant1\_fw\_ver(L)** | **ant2\_fw\_ver(H)** | **ant2\_fw\_ver(L)** | **ant3\_fw\_ver(H)** |
| 39 | 36 | 00 | 02 | | 01 | 06 | 01 | 06 | 01 |
| **ant3\_fw\_ver(L)** | **ant4\_fw\_ver(H)** | **ant4\_fw\_ver(L)** | **ant1\_gain** | | **ant2\_gain** | **ant3\_gain** | **ant4\_gain** | **ant1\_rssi** | **ant2\_rssi** |
| 06 | FF | FF | 1F | | 1F | 1F | FF | *A8* | *A8* |
| **ant3\_rssi** | **ant4\_rssi** | **crc16 (H)** | **crc16 (L)** |
| *A8* | *01* | 83 | F3 |

**Start flag**

Start flagsof ： 0x55AA

**Message header**

message lengthlen ： 0x002D

command code cmd ： 0x000A

Message serial number seq ： 0x00000002

protocol versionpro\_ver ： 0x0001 (V0.1)

security flagseq\_flag： 0x0000

device ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （change to string ：“861694034205896”）

**Service content**

parameter type param\_type : 02 (antenna info)

antenna version ant\_version ：01 06 01 06 01 06 FF FF 1 to 4 antenna version information, each antenna version in 2 bytes If it is FF FF, it means that the antenna version information of the channel is not read）

explain：

antenna 1 firmware; V1.6

antenna 2 firmware; V1.6

antenna 3 firmware; V1.6

Antenna 4 did not get the firmware version information.

antenna gain ant\_gain ：1F 1F 1F FF （1 ~ 4 antenna gain (range 0 ~ 31) FF, invalid）

explain:

antenna 1 gain: 31

antenna 2 gain: 31

antenna 3 gain: 31

antenna 4 gain: not get

Antenna filter signal strength ant\_rssi：A8 A8 A8 01 1 ~ 4 antenna filter threshold (range -1 ~ -128) 0 is not filter.1 means that the RSSI of the channel is not read）

explain:

antenna 1 rssi: -88

antenna 2 gain: -88

antenna 3 gain: -88

antenna 4 gain: not get

**Check**

crc16 : 0x83F3























































































































1. **Report user configuration parameters 0x10**

**The body of the report**： As shown in the table below

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **The No. of bytes** | **Description** |
| 1 | Parameter type param\_type | 1 | **0x10** —— **Reporting system parameters** |
| 2 | Parameter | 182 | Take Chapter 8 for the reference. |

eg2:　 55 AA 00D300 0A 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 0010550102 090100B4 0038 36 31 36 39 34 30 33 34 32 30 35 38 39 36 0000 00 01 C0 A8 01 C7 FF FF FF 00C0 A8 01 0164 0032 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0024 1332 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00F8 1100 2E 12 3C 00 2500 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0001 06 01 06 01 06 FF FF104D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 00A8 A8 A8 A81F 1F 1F 1F00A100 00 00 0030 47

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 01 | 0D | 00 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **data**  **…** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 10 | … | 30 | 47 |

**Starting logo Starting logo**

Starting logosof ： 0x55AA

**Message header Message header**

len ： 0x00D3

cmd ： 0x000A

seq ： 0x00000002

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Convert to string：“861694034205896”）

**Service Content**

param\_type : 10 (User configuration parameters)

data ：（For a specific format, see chapter 8 for configuration parameters）

*55* : 1. The 32 bytes in the frame, the parameter eigenvalues, read the configuration parameters, fixed to 0x55

*01* : 2. The 33rd byte in the frame, the working mode is: GPRS transmission, and the label record is transmitted to the platform（Four definitions： 0x01: GPRS 0x02: LAN）

*02 09* : 3. The 34th byte of the frame starts, firmware version V2.9 (The major version number 2， ubversion9)

*01* : 4. The 36th byte of the frame, the buzzer logo, opens the buzzer（0x00: close， 0x01： open）

00 : 5. The 37th byte in the frame, reserved 1

*B4 00* : 6. At the beginning of the 38th byte in the frame, the label is to refilter time 0x00B4, which is 180 seconds（Low byte ahead，, high byte after），To determine the time to leave the base station above V3.0

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 : 7. The 40 bytes in the frame start, the device ID,”861694034205896”

00 00: 8. At the beginning of the 56th byte in the frame, the base station stops periodically to report the function shutdown（V3.3 version）

*01* : 9. The 58th byte of the frame, DHCP enabled （0x00: close， 0x01： open, for LAN）

*C0 A8 01 C7* : 10. At the 59th byte of the frame, LAN local IP，192.168.1.199

*FF FF FF 00* : 11. The 63rd byte of the frame begins, the LAN subnet mask 255.255.255.0

*C0 A8 01 01* : 12. The 67 bytes in the frame start, the LAN gateway 192.168.1.1

*64 00* : 13. The 71st byte of the frame begins, LAN local IP port 0x0064，is 100（Low byte ahead, high byte after）

*32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00* : 14. At 73 bytes in frame, the IP of GPRS server 1，“218.17.157.214”， Platform IP（Balance server address）

*F8 11* : 15. The 105th byte of the frame begins with the port of GPRS server 1, 0x11F8 is 4600 （Low byte ahead, high byte after），Platform port

*32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00*：16. The 107th byte in the frame starts, LAN server 1's IP, “218.17.157.214”， Platform IP（Balance server address

*F8 11* : 17. At the beginning of the 139 bytes in the frame, the port of LAN server 1, 0x11F8, is 4600（Low byte ahead, high byte after）

*00 2E 12 3C 00 25*: 18. The 141st byte of the frame begins, LAN local MAC address 00-2E-12-3C-00-25

*00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00* : 19. The 147 bytes in the frame start and remain 3

*01 06 01 06 01 06 FF FF* : 20. The 175th byte of the frame begins with the firmware version of the antenna, which corresponds to the firmware version of the antenna 1 ~ 4. Each antenna version occupies 2 bytes, if it is FF FF, Indicates failure to read the modified antenna version. the firmware version 1 ~ 4 is the firmware version：V1.6, V1.6, V1.6,（The no. 4 antenna reading failed, probably no. 4 channel with no antenna）

*10* : 21. In frame 183 bytes, GPRS signal strength 0x10(16)

*4D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 00*: 22. The 184th byte at the beginning of the frame, the device number “*MR7901-003C0025*”

*A8 A8 A8 A8* : 23. The 200 bytes of the frame start, the antenna 1,2,3,4 rssi filter threshold， Namely, respectively, -88dBm, -88dBm, -88dBm, -88dBm

*1F 1F 1F 1F* : 24. The 204 bytes of the frame start, the antenna 1,2,3,4 gain, that is, respectively 31dBm, 31dBm,31dBm,31dBm

*00* : 25. The 208th byte of the frame, the bluetooth output tag identifier, 0x00 not output the label, 0x01 output label

*A1* : 26. 209 bytes of the frame, the communication connection status, 0 xa1: device platform has been established through GPRS communication connection (if it is 0 xa2: equipment and the platform through the LAN communication connections, 0 xa3: equipment and platform through GPRS, LAN the two communication connection)

*00 00 00 00:* 27. The 210 bytes in the frame start, reserved 4

**Checksum**

crc16 : 0x5C66

1. **Report equipment status 0x11**

Used to obtain the current state of the device

**The style**： as shown in the table below

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **Byte** | **Description** |
| 1 | param\_type | 1 | **0x11** —— eport equipment status |
| 2 | fw\_version | 2 | eg：  03 00 —— Host firmware version V3.0 |
| 3 | gprs\_ip | 32 | **String type**  eg：  “218.17.157.214” |
| 4 | gprs\_port | 2 | Low byte ahead |
| 5 | gprs\_link\_status | 1 | **0x00** —— No connection to the platform  **0x01** —— The platform has connections |
| 6 | lan\_ip | 32 | **String type** |
| 7 | lan\_port | 2 | Low byte ahead |
| 8 | lan\_link\_status | 1 | **0x00** —— No connection to the platform  **0x01** —— The platform has connections |
| 9 | gprs\_bffer\_cnt | 2 | The number of tags sent to the platform via GPRS  Low byte ahead |
| 10 | lan\_bffer\_cnt | 2 | Number of tags sent to the platform via wire  Low byte ahead |
| 11 | tag\_filt\_cnt | 2 | The number of labels used to filter or pre-judge  Low byte ahead |
| 12 | ant\_fw\_version | 8 | The version information of 1 ~ 4 antenna, the version information of each antenna occupies 2 bytes  The version information that is not read to the antenna is not read for FF FF |
| 13 | ant\_gain | 4 | 1 ~ 4 antenna gain Settings  Value range: 0～31  In the case of FF, the antenna gain is not obtained |
| 14 | ant\_rssi | 4 | Rssi threshold setting for antenna 1 ~ 4 antenna  Range of values: -1 ~ 128 (band symbol single-byte, singed char)  At 1, the rssi is not obtained |
| 15 | gprs\_flash\_tag\_cnt | 4 | Stored in flash, the number of tags to be sent to the platform via GPRS  Low byte ahead |
| 16 | lan\_flash\_tag\_cnt | 4 | Stored in flash, ready to be sent to the platform by wire.  Low byte ahead |
| 17 | flash\_log\_cnt | 4 | Low byte ahead |

eg6: 55 AA 00 8700 0A 00 00 00 03 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 001103 0032 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0024 130132 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0025 130018 0018 0008 00FF FF FF FF 01 06 FF FFFF FF 01 FF01 01 D8 0100 00 00 0000 00 00 0074 00 00 00 D9 AA

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 1E | 00 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 03 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **data** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 11 | … | D9 | AA |

**Starting logoStarting logo**

Starting logoStarting logo sof ： 0x55AA

**Message headerMessage header**

len ： 0x001E

cmd ： 0x000A

seq ： 0x00000003

pro\_ver ： 0x0001 (V0.1)

seq\_flag ： 0x0000

ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （string：“861694034205896”）

**Service Content**

param\_type : 11 1. (**Report equipment status**)

firmware : 03 00 2. (Host firmware version V3.0)

IP : 32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3. (Current device GPRS connection ip，“218.17.157.214”)

IP port : 24 13 4. (IP ports currently connected to GPRS，0x1324, Is 4900, Low byte ahead)

gprs connection status : 01 5. (The current device is connected to the platform through GPRS)

Cable connectionIP : 32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6. (Current device wired IP，“218.17.157.214”)

IP port : 25 13 7. (The current wired IP port，0x1325, is 4901, Low byte ahead)

Connection status : 00 8. (The current device has no connection through cable and platform)

Gprs tag number : 18 00 9. (The number of tags sent to the platform via GPRS in the cache 0x0018, Low byte ahead)

Cable connection tag No. : 18 00 10. (The number of tags sent to the platform by wire in the cache 0x0018, Low byte ahead)

Tag filter : 08 00 11. (The number of tags sent to the platform by wire in the cache 0x0018, Low byte ahead)

Ant firmware : FF FF FF FF 01 06 FF FF 12. (1 ~ 4 antenna version information, among：

No. 1 antenna was not read

No. 2 antenna was not read

The version 3 antenna is V1.6

No. 4 antenna was not read version)

Ant gain setting : FF FF 01 FF 13. (1 ~ 4 antenna gain Settings，其中among：

Antenna no. 1 has no gain value

Antenna no. 2 has no gain value

The gain of antenna 3 is set to 1

Antenna no. 4 has no gain value)

rssi : 01 01 D8 01 14. (Rssi threshold setting for antenna 1 ~ 4 antenna, among：

No. 1 antenna has not been read to rssi values

Antenna no. 2 has not read the rssi value

The rssi of antenna 3 antenna is set to 0xD8, which is -40 symbol single-byte number，singed char

No. 4 antenna did not read the rssi value)

Flash gprs tag no. : 00 00 00 00 15. (In flash, the number of tags sent to the platform via GPRS 0x000000, Low byte ahead)

flash cable tag No. : 00 00 00 00 16. (In flash, the number of tags sent to the platform via wire 0x000000, Low byte ahead)

flash log No. : 74 00 00 00 17. (In flash, the number of tags sent to the platform via wire 0x000074, Low byte ahead)

**checksum**

crc16 : 0xD9AA

1. **Report equipment hardware information 0x12**

Equipment hardware information, current communication equipment ID, hardware ID, product number, GPRS module IMEI, SIM card CCID, battery power, external power identification, etc

Report: 128 bytes, the following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **Bytes** | **Descritption** |
| 1 | param\_type | 1 | **0x12** —— Report device hardware information |
| 2 | cur\_device\_id | 16 | String, the ID used in the current communication header, the last byte is 0 |
| 3 | mcu\_device\_id | 16 | The string, the last byte is 0 |
| 4 | product\_sn | 16 | The string, the last byte is 0 |
| 5 | gprs\_imei | 16 | The string, the last byte is 0 |
| 6 | sim\_ccid | 21 | The string, the last byte is 0 |
| 7 | battery\_level | 1 | Value: 0 ~ 10, 10 represents 100% electricity |
| 8 | power\_link\_status | 1 | 0: external power supply normal, 1: external power supply disconnect |
| 9 | reserved | 40 | Reserved |

eg7：55 AA 00 9C00 0A 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 001238 36 31 36 39 34 30 33 34 32 30 35 38 39 36 0034 33 35 35 31 30 35 30 30 33 43 30 30 32 35 004D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 0038 36 31 36 39 34 30 33 34 32 30 35 38 39 36 0038 39 38 36 30 32 62 31 31 39 31 36 39 30 30 30 32 33 32 39 00090000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0025 3E

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 01 | 9C | 00 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **data**  **…** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 12 | … | 25 | 3E |

**Starting logoStarting logo**

Starting logoStarting logo sof ： 0x55AA

**Message header**

len ： 0x009C

cmd ： 0x000A

seq ： 0x00000002

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

ID dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （ “861694034205896”）

**Service Content**

param\_type : 12 1.（**Report device hardware information**) the 31st byte of the frame (calculated from 0)

cur\_device\_id : 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 2. “861694034205896”（Start at 32 bytes in the frame）

mcu\_device\_id : 34 33 35 35 31 30 35 30 30 33 43 30 30 32 35 00 3. “4355105003C0025” （Start at 48 bytes in the frame）

product\_sn: 4D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 00 4. “MR7901-003C0025” （Start at 64 bytes in the frame）

gprs\_imei: 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 5. “861694034205896” （Start at 80 bytes in the frame）

sim\_ccid : 38 39 38 36 30 32 62 31 31 39 31 36 39 30 30 30 32 33 32 39 00 6. “898602b1191690002329”（Start at 96 bytes in the frame）

battery\_level : 09 7. Battery power 90%（Start at 117 bytes in the frame）

power\_link\_status : 00 8. External power supply（Start at 118 bytes in the frame）

reserved : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 9. Start at 119 bytes in the frame）

**Checksum**

crc16 : 0x253E

1. **Confirmation of information 0x80**

When the configuration message of the platform is received, such as configuring the user configuration parameter, configuring the antenna parameter, and so on, the device will respond to the following message to the platform, responding to the result of the configuration parameters of the platform.

Note: after the device responds to this message, it will be restarted immediately and run with the correct parameters configured.

Report: the following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | param\_type | 1 | **0x80** —— Report configuration confirmation message (V2.8 support) |
| 2 | return\_opt | 1 | **0x01** —— Configuration parameter success  **0x00** —— Configuration parameter failed |

eg8: 55 AA00 1E00 0A 00 00 00 06 00 01 00 00 34 33 35 35 31 30 33 30 30 33 45 30 30 33 39 0080 0147 84

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 1E | 00 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 06 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **return\_opt** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 80 | 01 | 47 | 84 |

**Starting logo Starting logo**

Starting logosof ： 0x55AA

**Message header**

len ： 0x001E

cmd ： 0x000A

seq ： 0x00000006

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （转为字符串为Convert to string：“861694034205896”）

**Service Content Service Content**

param\_type : 80 Platform confirmation message The configuration parameters have been received for the device platform)

return\_opt : 01 Configuration parameter success

**check sum**

crc16 : 0x4784

### platform validation package definition

After the platform receives the message of configuration parameters sent on the device (the command code is 0x000A), configure the configuration message, such as the configuration parameter system parameter, configuration antenna parameter, and so on, as needed.

Service Content: Contains parameter types and parameters

**Confirmation code**： 0x800A

**Service Content**： The following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | Type | 1 | **0x02** : gain,rssi Configure antenna gain,rssi  **0x10** : Configure user parameters (V2.7 support)  **0x80** : Platform confirmation message (V2.8 support) |
| 2 | Parameter | x | **he first byte is 0x02**  8Bytes antenna gain and RSSI  **The first byte is 0x10**  182Bytes, user configuration parameters  **The first byte is 0x80:**  1Byte, 1: the platform received the configuration parameter success; Zero: |

1. **Set the antenna parameter 0x02**

Gain gain for configuring the antenna to filter the RSSI threshold.

**Service Content**： The following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | param\_type | 1 | **0x02** —— Reported antenna information |
| 2 | ant\_gain | 4 | Gain of antenna 1 ~ 4 (range 0 ~ 31)  If it is FF, it indicates that the antenna gain is not read to the channel |
| 3 | ant\_rssi | 4 | 1 ~ 4 antenna filter threshold value (range -1 ~ 128)  If it is 01, it indicates that the antenna is not read to the channel RSSI |

eg3：55 AA 00 2580 0A 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00021F 1F 1F 1FA8 A8 A8 A8C4 88

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 25 | 80 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **ant1\_gain** | **ant2\_gain** | **ant3\_gain** | **ant4\_gain** | **ant1\_rssi** |
| 39 | 36 | 00 | 02 | 1F | 1F | 1F | 1F | *A8* |
| **ant2\_rssi** | **ant3\_rssi** | **ant4\_rssi** | **crc16 (H)** | **crc16 (L)** |
| *A8* | *A8* | *A8* | C4 | 88 |

**Starting logo**

Starting logosof ： 0x55AA

**Message header**

len ： 0x002D

cmd ： 0x800A

seq ： 0x00000002

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Convert to string：“861694034205896”）

**Service Content**

param\_type : 02 (Set antenna information)

ant\_gain ：1F 1F 1F 1F （The gain of 1 ~ 4 of the antenna is respectively corresponding to the range of 0 ~ 31）

analysis

Antenna 1 gain: 31

Antenna2 gain: 31

Antenna3 gain: 31

Antenna4 gain: 31

ant\_rssi：A8 A8 A8 A8 （The filter threshold of the signal strength of the antenna 1 ~ 4 is respectively corresponding to the range of value: 0 ~ -128）

Analysis

Antenna1 rssi: -88

Antenna2 gain: -88

Antenna3 gain: -88

Antenna4 gain: -88

**Checksum**

crc16 : 0xC488

1. **Set the transmission label data identification 0x03**

Used to configure whether the device reports label data to the platform.

**Service Content**： The following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | param\_type | 1 | **0x03** ——Submit the transmission label data identification |
| 2 | Tag no. | 1 | 00 Report the label data to the platform  01 Do not report the label data to the platform |

eg4: 55 AA 00 1E80 0A 00 00 00 03 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 03016A CD

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 00 | 1E | 80 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 03 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **data** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 03 | 01 | 6A | CD |

**Starting logo**

Starting logosof ： 0x55AA

**Message header**

len ： 0x001E

cmd ： 0x800A

seq ： 0x00000003

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Convert to string：“861694034205896”）

**Service Content**

param\_type : 03 (**Report the label data identification**)

data ：

01 : Do not report the label data to the platform (0x00 reports the label data to the platform, 0x01 does not report the label data to the platform)

**Checksum**

crc16 : 0x6ACD

1. **Set user configuration parameters 0x10**

**Service Content**： The following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | param\_type | 1 | **0x10** —— Reporting system parameters |
| 2 | Parameter | 182 | For a specific format, see chapter 8, configuration parameters |

Note: the device will be restarted after receiving the correct message.

eg5:　 55 AA 00D380 0A 00 00 00 02 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 0010 55 *0102 0701013C 0038 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 00 0001 C0 A8 01 C7FF FF FF 00C0 A8 01 02DC FF32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0024 134D 54 49 49 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00236D 74 69 32 36 35 34 36 33 39 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0000 00 00 00 00 00 00 00 00144D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 00A8 A8 A8 A81F 1F 1F 1F00A100 00 0000*0E 47

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | **cmd(L)** | **seq (MSB)** |  |  |
| 55 | AA | 01 | 0D | 80 | 0A | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | **dev\_id (MSB)** |  |  |  |
| 02 | 00 | 01 | 00 | 00 | 38 | 36 | 31 | 36 |
|  |  |  |  |  |  |  |  |  |
| 39 | 34 | 30 | 33 | 34 | 32 | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **data**  **…** | **crc16 (H)** | **crc16 (L)** |
| 39 | 36 | 00 | 10 | … | 0E | 47 |

Starting logosof ： 0x55AA

**Message header**

len ： 0x00D3

cmd ： 0x800A

seq ： 0x00000002

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Convert to string：“861694034205896”）

**Service Content**

param\_type : 10 (User configuration parameters)

data ：（For a specific format, see chapter 8 for configuration parameters）

*55* : 1. The 32 bytes in the frame, the parameter eigenvalues, read the configuration parameters, fixed to 0x55

*01* : 2. The working mode is the 33rd byte in the frame, which is transmitted by GPRS, and the label record is transmitted to the platform (low four definitions: 0x01: GPRS 0x02: LAN).

*02 09* : 3. The 34th byte of the frame begins, firmware version V2.9 (main version 2, sub version 9)

*01* : 4. The 36th byte of the frame, the buzzer logo, opens the buzzer (0x00: close, 0x01: open)

*00* : 5. In frame 37 bytes, reseve 1

*B4 00* : 6. At the beginning of the 38th byte in the frame, the label is to refilter time 0x00B4, which is 180 seconds (Low byte at the front Low byte ahead, high byte), and above V3.0 to determine the time to leave the base station

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 : 7. Start of the 40th byte in frame, device ID, "861694034205896"

00 00: 8. The 56th byte of the frame begins, and the base station stops periodically to report the function shutdown (V3.3 version).

*01* : 9. The 58th byte of the frame, DHCP enabled (0x00: close, 0x01: open, applicable to LAN)

*C0 A8 01 C7* : 10. At the 59th byte of the frame, LAN local IP, 192.168.1.199

*FF FF FF 00* : 11. **At the beginning of the 63 bytes in the frame, the LAN subnet mask is 255.255.255.0**

*C0 A8 01 01* : 12. The 67th byte of the frame starts, and the LAN gateway is 192.168.1.1

*64 00* : 13. At the beginning of the 71st byte in the frame, LAN local IP port 0x0064, or 100（Low byte ahead， High byte behind）

*32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00* : 14. Start of 73 bytes in frame, IP of GPRS server 1, "218.17.157.214", platform IP (balanced server address)

*F8 11* : 15. The 105th byte of the frame begins, the port of GPRS server 1, 0x11F8 is 4600 （Low byte ahead， High byte behind）

*32 31 38 2E 31 37 2E 31 35 37 2E 32 31 34 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00*：16. The 107th byte of the frame begins, LAN server 1 IP, "218.17.157.214", platform IP (balanced server address)

*F8 11* : 17. Start at 139 bytes in frame, LAN server 1 port, 0x11F8, or 4600（Low byte ahead， High byte behind）

*00 2E 12 3C 00 25*: 18. At the start of the 141 bytes in the frame, LAN local MAC address 00-2e-12-3c -00-25

*00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00* : 19. The 147 bytes in the frame start and reseved 3

*01 06 01 06 01 06 FF FF* : 20. Firmware version 175 bytes of the frame, antenna, respectively corresponding to the firmware version of 1 ~ 4 antenna version of 2 bytes, each antenna for FF FF, said reading change antenna version failed. The firmware version 1 ~ 4 is: V1.6, V1.6, V1.6, no (no 4 antenna reading failed, may be no. 4 channel without antenna)

*10* : 21. In frame 183 bytes, GPRS signal strength 0x10(16)

*4D 52 37 39 30 31 2D 30 30 33 43 30 30 32 35 00*: 22. Starting with the 184 bytes in the frame, the device number "mr791-003c0025"

*A8 A8 A8 A8* : 23. At the beginning of the 200 bytes in the frame, the rssi filter threshold of 1,2,3 and 4, namely, -88dbm, -88dbm, -88dbm, -88dbm

*1F 1F 1F 1F* : 24. The 204 bytes of the frame start, the antenna 1,2,3,4 gain, that is 31dBm,31dBm, 31dBm,31dBm

*00* : 25. The 208th byte of the frame, the bluetooth output tag identifier, 0x00 not output the label, 0x01 output label

*A1* : 26. 209 bytes of the frame, the communication connection status, 0 xa1: device platform has been established through GPRS communication connection (if it is 0 xa2: equipment and the platform through the LAN communication connections, 0 xa3: equipment and platform through GPRS, LAN the two communication connection)

*00 00 00 00:* 27. The 210 bytes in the frame start and reseved 4

**checksum**

crc16 : 0x0E47

1. **Confirmation of the platform 0x80**

The platform receives configuration parameter information from the device, such as report user configuration parameter, antenna information, etc. Used to inform the device platform that the reported configuration message has been received.

**Service Content**： The following table

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Description** |
| 1 | param\_type | 1 | **0x80** —— Report configuration confirmation message (V2.8 support) |

eg6: 55 AA 00 1D80 0A 00 00 00 03 00 01 00 00 38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 80A6 E0

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **sof(H)** | **sof(L)** | **len(H)** | **len(L)** | **cmd(H)** | | **cmd(L)** | | **seq (MSB)** |  |  |
| 55 | AA | 00 | 1D | 80 | | 0A | | 00 | 00 | 00 |
| **seq (LSB)** | **pro\_ver (H)** | **pro\_ver (L)** | **sec\_flag (H)** | **sec\_flag (L)** | | **dev\_id (MSB)** | |  |  |  |
| 03 | 00 | 01 | 00 | 00 | | 38 | | 36 | 31 | 36 |
|  |  |  |  |  | |  | |  |  |  |
| 39 | 34 | 30 | 33 | 34 | | 32 | | 33 | 35 | 38 |
|  |  | **dev\_id (LSB)** | **param\_type** | **crc16 (H)** | **crc16 (L)** | |
| 39 | 36 | 00 | 80 | A6 | E0 | |

**Starting logo**

Starting logosof ： 0x55AA

**Message header**

len ： 0x001E

cmd ： 0x800A

seq ： 0x00000003

pro\_ver ： 0x0001 (V0.1)

seq\_flag： 0x0000

dev\_id ：

38 36 31 36 39 34 30 33 34 32 30 35 38 39 36 00 （Convert to string：“861694034205896”）

**Service Content**

param\_type : 80 Platform confirmation message (for response device, notification device platform has received configuration parameters)

**checksum**

crc16 : 0xA6E0

# TLV index

## TLV Type list and format

### format

|  |  |  |
| --- | --- | --- |
| TLV 2bytes | LENGTH 2bytes | VALUE |

### type index

|  |  |  |  |
| --- | --- | --- | --- |
| **TLV type** | **TAG** | **LENGTH** | **VALUE** |
| RFID item monitor | 0x8801 | 16 | 1 byte of signal strength +4 byte reader ID+1 byte reader state +4 byte tag ID+ 6 byte collection time  The reader state 0 is normal and 1 is removed |
| Current tag monitor | 0x8901 | 16 | 1 byte signal strength +4 byte tag ID +5 byte tag sensor information +6 byte acquisition time |
| Wristband tag data | 0x8A01 | 18 | 1 byte signal strength +4 byte bracelet ID+1 byte type +2 bytes bytes +6 byte acquisition time +6 byte reception time |
| tag | 0x8B01 | 17 | Antenna Channel(1byte)+ tag type (1byte)+ id(4bytes)+sum(1byte)+ incentive address (2bytes)+ voltage state (1Bytes) +rssi(1byte) + receive time (6bytes) |
| Attendance tag | 0x8B02 | 17 | (1byte)+ id(4bytes)+sum(1byte)+ incentive address (2bytes)+ voltage state (1Bytes) +rssi(1byte) + receiving time (6bytes) |

tag eg1： 8B 0100 1101 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09

|  |  |  |
| --- | --- | --- |
| **TLV type** | **TAG** | **LENGTH** |
| 8B 01 | 00 11 | 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 |

The analysis is as follows：

The TLV type ： 0x8B01

TLV data length ： 0x0011

The TLV data ： 01 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 （take the electronic tag data format specification for example)

Attendance tag eg2： 8B 0200 1181 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09

|  |  |  |
| --- | --- | --- |
| **TLV type** | **TAG** | **LENGTH** |
| 8B 02 | 00 11 | 81 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 |

The analysis is as follows：

The TLV type ： 0x8B02

TLV data length： 0x0011

The TLV data ： 81 20 78 2B 6A A4 2F 00 00 00 A9 11 01 0E 13 26 09 （take the electronic tag data format specification for example)

### description of electronic tag format

Electronic tag (type 0x8B01) (17 bytes)

eg : 01**20 E3 AF 22 32 FA 00 00 00 B2**11 01 0E 13 26 09

|  |  |  |
| --- | --- | --- |
| **channel (1byte)** | **Tag data**  **（10Bytes）** | **time**  **(6bytes)** |
| 01 | **20 E3 AF 22 32 FA 00 00 00 B2** | 11 01 0E 13 26 09 |

Description:

**01 ：** Access to base station status/antenna Channel number (Channel), this Byte

The state of the base station;

low digit 4bit (0 ~ 3bit) : antenna number, 1,2,3,4 corresponding to the east, south, west, and north 4 antennas.

|  |  |  |  |
| --- | --- | --- | --- |
| **Exit base station status/antenna Channel** | | | |
| 7 bit | 6 bit | 5,4 bit | 3～0 bit |
| Base station status:  1 ： Read range from base station  0 ： Read the range out the base station | Station stop sign  1 ： Base station for  0 ： It's not a base station stop | reserve | Read the tag's antenna channel number |

0x**01** , in， Read from antenna no. 1

*Note:*

*1. To judge the need for the base station, 7bit and 6bit should be considered simultaneously.*

*2. When the base station is marked as 1, the base station status is invalid.*

**20 E3 AF 22 32 FA 00 00 00 B2 ：** Please refer to chapter 7.2 for the label data

11 01 0E 13 26 09 : The label receives (reads) time, namely year, month, day, hour, minute, second, year is based on 2000, January 14, 2017 19:38:09

### The attendance tag format description

The attendance tag (type 0x8B02) format specification (17 bytes).

eg : 81**20 E3 AF 22 32 FA 00 00 00 B2**11 01 0E 13 26 09

|  |  |  |
| --- | --- | --- |
| **Attendance/antenna channel no (1byte)** | **tag data**  **（10Bytes）** | **attendance time**  **(6bytes)** |
| 81 | **20 E3 AF 22 32 FA 00 00 00 B2** | 11 01 0E 13 26 09 |

**81 ：** Attendance/antenna Channel; In this 1Byte, the highest (8bit) indicates attendance or the coming out, of which 1 is in and 0 is out

|  |  |  |  |
| --- | --- | --- | --- |
| **Attendance/antenna Channel** | | | |
| 7 bit | 6 bit | 5,4 bit | 3～0 bit |
| Sign in and out of attendance：  1 ： in  0 ： out | Identification of unilateral attendance  1 ： unilateral attendance  0 ： not unilateral attendance | reserved | Read the tag's antenna channel number |

0x**81** ， Enter the door, read from the antenna no. 1

***Note****：*

*1. It is necessary to consider the 7bit and 6bit for the entry and exit*

*2. When the unilateral attendance mark is 1, the attendance check mark is invalid*

**20 E3 AF 22 32 FA 00 00 00 B2 ：** Please refer to chapter 7.2 for the label data

11 01 0E 13 26 09 : The label receives (reads) time, namely year, month, day, hour, minute, second, year is based on 2000, January 14, 2017 19:38:09

## tag format description

The tag data is fixed to 10 bytes. The specific format is defined as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Lenth(Byte)**  **tag type** | **type**  **(1Byte)** | **tag ID**  **(4Bytes）** | **CC**  **(1Byte)** | **RSV**  **(2Bytes)** | **Status**  **(1Byte)** | **version**  **(1Byte)** |
| student card | 0x20 | Use IC card number | Check sum of front 5Btyes (frame head, card number) |  | Bit0: low pressure indication  (0: normal,  1: low voltage  )  Bit4: key state  (0: press the button,  1: press without a button  ) | Version number  Bit7:  Special mark |
| electric bike tag | 0x30 | Production batch number + serial number, production batch number is four decimal number, the actual use hex storage and transport, such as 1708, the actual 0 x17 0 x08 serial number to four decimal number, such as 1012, the actual data of 0 x10 and 0 x12. |  |
| Electric bike key card | 0x31 | Production batch number + flow number, rule same |  |

eg : **20E3 AF 22 32FA00 0000B2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| tag type（1byte） | ID  （4byte） | Checksum  （1byte） | Incentives to address  （2byte） | Low pressure alarm  （1byte） | RSSI  （1byte） |
| 20 | E3 AF 22 32 | FA | 00 00 | 00 | B2 |

**20 ：** tag type, student card

**E3 AF 22 32 ：** Tag ID，

**FA** ： 20 E3 AF 22 32 and checksum (check algorithm to see chapter 8 calibration algorithm)

**00 00** **：** Motivational address, not motivational

**00** **：** No voltage alarm (low voltage is 0x01, normal is 00

**B2** **：** Signal strength, -78dbm (symbols for single byte)

# Configuration parameter format description

The following is the configuration parameter format of the device (182Bytes).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item** | **Data segment** | **bytes** | **Read and write attribute** | **type** | **Description** |
| 1 | Parameter eigenvalue | 1 | R/W | value | When writing, keep 0x55 constant, but normal configuration parameters  When written, it is not 0x55, that is, the factory default parameter is restored  When read, it is fixed at 0x55 |
| 2 | Communication mode | 1 | R/W | value | Low 4 bits, read only, transmission mode:  1bit: read only, GPRS connection, 1 valid  No. 2bit: read-only, LAN connection, 1 valid  3bit: retention  No. 4bit: reservation  No. 5bit: read and write, tag transfer mark, 0: transmit tag records to the platform; 1: no label record is transmitted to the platform  6-8bit: reservations  eg： 0x11, which represents the GPRS connection, does not transmit tag records to the platform  eg: 0x01, identifies the GPRS connection and transmits the label record to the platform |
| 3 | Firmware version | 2 | R | value | Firmware version, the main version number is in front  eg: 02 07， V2.7 |
| 4 | The buzzer logo | 1 | R/W | value | 1. Start the buzzer, 0: turn off the buzzer |
| 5 | reserved 1 | 1 | - | - |  |
| 6 | To heavy window  (leave base station to judge time) | 2 | R/W | value | 20～65535 TAB to refilter window, unit: second; 0x0000 does not filter, Low byte at the front Low byte ahead, high byte in the back  Range: 20 ~ 65535 |
| 7 | Device ID | 16 | R/W | string | The 15bit is product serial number Item is ASCII, followed by 1 byte at 0x00  eg: “861694034205896” |
| 8 | The base station stays over time | 2 | R/W | value | The tag stays at the base station and is reported to the platform at the interval time (the V3.3 version only has this) low byte before and after high byte  Unit: second  Value range: 0, 60 ~ 65535  0: do not use this feature  60 ~ 65535: timeout period |
| 9 | DHCP able | 1 | R/W | value | 1： open, 0: closed, for LAN |
| 10 | LAN local IP | 4 | R/W | value | Used for LAN network parameter configuration |
| 11 | LAN Mask | 4 | R/W | value | Used for LAN network parameter configuration |
| 12 | LAN gateway | 4 | R/W | value | Used for LAN network parameter configuration |
| 13 | LAN local port | 2 | R/W | value | Local IP port, applicable to LAN, low byte before, high byte after (value range 0 ~ 65536)  eg: *24 13 , is 0x1324(HEX) = 4900 (DEC)* |
| 14 | GPRS server 1 IP | 32 | R/W | string | IP or domain name, string |
| 15 | G GPRS server 1 port | 2 | R/W | value | low byte before, high byte after (value range 0 ~ 65536) |
| 16 | LAN server 1 IP | 32 | R/W | string | IP or domain name, string |
| 17 | LAN server 1 port | 2 | R/W | value | low byte before, high byte after (value range 0 ~ 65536) |
| 18 | LAN local mac | 6 | R/W | value | Used for LAN network parameter configuration |
| 19 | Reserved 3 | 28 | - | - | - |
| 20 | Antenna version | 8 | R | value | he firmware version corresponding to four antennas  The version of each antenna occupies two bytes  eg： 0106 is V1.6, if FF FF indicates that the read antenna version failed (probably not connected to the antenna) |
| 21 | GPRS rssi | 1 | R | value | 00 or 99 means GPRS no signal  99 indicates failure to read the GPRS signal  The normal range is 0 ~ 31 |
| 22 | Device ID | 16 | R | string | eg: “MR7901-003C0025” |
| 23 | Antenna RSSI threshold | 4 | R/W | Signed value | Value range: -1 ~ -128  The signal strength RSSI filter threshold of 1,2,3 and 4 corresponding to the antenna is respectively.  eg: -88 |
| 24 | Antenna gain | 4 | R/W | value | The value range is 0~ 31, corresponding to the signal gain value of the antenna 1,2,3 and 4 |
| 25 | Bluetooth output label identification | 1 | R/W | value | 0: the label cannot be exported via bluetooth  1: the label can be exported via bluetooth, and the tag that the antenna reads will only be exported via bluetooth. GPRS and LAN are forbidden to export labels |
| 26 | Communicate status | 1 | R | value | Direct connection to the platform  Low 4 bits, connection mode  1bit: read only, GPRS connection, 1 valid  No. 2bit: read-only, LAN connection, 1 valid  3bit: retention  No. 4bit: reservation  It's 4 digits high, fixed as A  0 xa0 connectionless  0xA1 has GPRS connections  0xA2 has a LAN connection  0xA3 has GPRS, LAN connections |
| 27 | Reserved 4 | 4 | - | - |  |

# Summary of the commands

|  |  |  |
| --- | --- | --- |
| **Item** | **cmd** | **Description** |
| 1 | 0x0008 | Terminal registration request |
| 2 | 0x8008 | Platform confirmation terminal registration |
| 3 | 0x0001 | Terminal login request |
| 4 | 0x8001 | The platform confirms the terminal login |
| 5 | 0x0003 | Terminal heartbeat |
| 6 | 0x8003 | Platform confirm heartbeat |
| 7 | 0x0004 | The terminal sends the tag data |
| 8 | 0x8004 | The platform acknowledges receipt of the tag |
| 9 | 0x000D | To upgrade the firmware |
| 10 | 0x800D | Platform to confirm upgrade firmware |
|  |  |  |
|  |  |  |

# Checksum algorithm

## CRC16 algorithm

The data packet of communication with the platform, using CRC16 algorithm. The following two kinds of checking algorithms are introduced, which are recommended using method 2. Method 2 is used to check the table, which is 8 times faster than method 1

### C#.net method 1

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name : crc16\_ccitt

\*\* Descriptions : Cyclic redundancy check -16 （CCITT standard-0x1021）

\*\* input parameters : buf The data to be checked

\*\* len Check the length of the data

\*\* output parameters : None

\*\* Returned value : value

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

uint16\_t crc16\_ccitt(uint8\_t \*buf, uint16\_t len)

{

uint16\_t i, j;

uint16\_t crc\_reg = 0xFFFF;

uint16\_t crc\_val;

for (i = 0; i < len; i++)

{

crc\_val = buf[i] << 8;

for (j = 0; j < 8; j++)

{

if (((int16\_t)(crc\_reg ^ crc\_val)) < 0)

crc\_reg = (crc\_reg << 1) ^ 0x1021;

else

crc\_reg <<= 1;

crc\_val <<= 1;

}

}

return crc\_reg;

}

### C#.net method 1 Look-up table method

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name : CRC16

\*\* Descriptions : yclic redundancy check -16 （CCITT standard-0x1021）

\*\* input parameters : buf The data to be checked

\*\* len Check the length of the data

\*\* output parameters : None

\*\* Returned value : value

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

const uint16\_t crc16\_table[]= //\* CRC16 CCITT标准-0x1021

{

0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50a5, 0x60c6, 0x70e7,

0x8108, 0x9129, 0xa14a, 0xb16b, 0xc18c, 0xd1ad, 0xe1ce, 0xf1ef,

0x1231, 0x0210, 0x3273, 0x2252, 0x52b5, 0x4294, 0x72f7, 0x62d6,

0x9339, 0x8318, 0xb37b, 0xa35a, 0xd3bd, 0xc39c, 0xf3ff, 0xe3de,

0x2462, 0x3443, 0x0420, 0x1401, 0x64e6, 0x74c7, 0x44a4, 0x5485,

0xa56a, 0xb54b, 0x8528, 0x9509, 0xe5ee, 0xf5cf, 0xc5ac, 0xd58d,

0x3653, 0x2672, 0x1611, 0x0630, 0x76d7, 0x66f6, 0x5695, 0x46b4,

0xb75b, 0xa77a, 0x9719, 0x8738, 0xf7df, 0xe7fe, 0xd79d, 0xc7bc,

0x48c4, 0x58e5, 0x6886, 0x78a7, 0x0840, 0x1861, 0x2802, 0x3823,

0xc9cc, 0xd9ed, 0xe98e, 0xf9af, 0x8948, 0x9969, 0xa90a, 0xb92b,

0x5af5, 0x4ad4, 0x7ab7, 0x6a96, 0x1a71, 0x0a50, 0x3a33, 0x2a12,

0xdbfd, 0xcbdc, 0xfbbf, 0xeb9e, 0x9b79, 0x8b58, 0xbb3b, 0xab1a,

0x6ca6, 0x7c87, 0x4ce4, 0x5cc5, 0x2c22, 0x3c03, 0x0c60, 0x1c41,

0xedae, 0xfd8f, 0xcdec, 0xddcd, 0xad2a, 0xbd0b, 0x8d68, 0x9d49,

0x7e97, 0x6eb6, 0x5ed5, 0x4ef4, 0x3e13, 0x2e32, 0x1e51, 0x0e70,

0xff9f, 0xefbe, 0xdfdd, 0xcffc, 0xbf1b, 0xaf3a, 0x9f59, 0x8f78,

0x9188, 0x81a9, 0xb1ca, 0xa1eb, 0xd10c, 0xc12d, 0xf14e, 0xe16f,

0x1080, 0x00a1, 0x30c2, 0x20e3, 0x5004, 0x4025, 0x7046, 0x6067,

0x83b9, 0x9398, 0xa3fb, 0xb3da, 0xc33d, 0xd31c, 0xe37f, 0xf35e,

0x02b1, 0x1290, 0x22f3, 0x32d2, 0x4235, 0x5214, 0x6277, 0x7256,

0xb5ea, 0xa5cb, 0x95a8, 0x8589, 0xf56e, 0xe54f, 0xd52c, 0xc50d,

0x34e2, 0x24c3, 0x14a0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405,

0xa7db, 0xb7fa, 0x8799, 0x97b8, 0xe75f, 0xf77e, 0xc71d, 0xd73c,

0x26d3, 0x36f2, 0x0691, 0x16b0, 0x6657, 0x7676, 0x4615, 0x5634,

0xd94c, 0xc96d, 0xf90e, 0xe92f, 0x99c8, 0x89e9, 0xb98a, 0xa9ab,

0x5844, 0x4865, 0x7806, 0x6827, 0x18c0, 0x08e1, 0x3882, 0x28a3,

0xcb7d, 0xdb5c, 0xeb3f, 0xfb1e, 0x8bf9, 0x9bd8, 0xabbb, 0xbb9a,

0x4a75, 0x5a54, 0x6a37, 0x7a16, 0x0af1, 0x1ad0, 0x2ab3, 0x3a92,

0xfd2e, 0xed0f, 0xdd6c, 0xcd4d, 0xbdaa, 0xad8b, 0x9de8, 0x8dc9,

0x7c26, 0x6c07, 0x5c64, 0x4c45, 0x3ca2, 0x2c83, 0x1ce0, 0x0cc1,

0xef1f, 0xff3e, 0xcf5d, 0xdf7c, 0xaf9b, 0xbfba, 0x8fd9, 0x9ff8,

0x6e17, 0x7e36, 0x4e55, 0x5e74, 0x2e93, 0x3eb2, 0x0ed1, 0x1ef0

};

uint16\_t CRC16(uint8\_t \* Data, uint16\_t Length)

{

uint16\_t crc;

uint8\_t da;

crc = 0xFFFF;

while(Length--!=0)

{

da=(uint8\_t) (crc/256);

crc <<= 8;

crc ^= crc16\_table[da^\*Data];

Data++;

}

return crc;

}

### JAVA Method

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* CRC-CCITT standard JAVA

\*

\*

\* data Input: bytes of bytes required to add validation

\*

\* Output: returns a hexadecimal check code of 2 bytes

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

private static String getCrc(byte[] data) {

int crc = 0xFFFF; //crc

for (int i = 0; i < data.length; i++) {

crc = (data[i] << 8) ^ crc;

for (int j = 0; j < 8; ++j) {

if ((crc & 0x8000) != 0)

crc = (crc << 1) ^ 0x1021;

else

crc <<= 1;

}

}

return Integer.toHexString(crc & 0xFFFF).toUpperCase();

}

## Checksum algorithm

The tag data is used and verified.

### C#.NET Calculation function

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\* Function name : CheckSum

\*\* Descriptions : checksum

\*\* input parameters : uBuff he data to be checked

\*\* uBuffLen Check the length of the data

\*\* output parameters : None

\*\* Returned value : value

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

uint8 CheckSum(uint8 \*uBuff, uint16 uBuffLen)

{

uint16 i;

uint8 uSum=0;

for(i=0;i<uBuffLen;i++)

{

uSum = uSum + uBuff[i];

}

uSum = (~uSum) + 1;

return uSum;

}

### JAVA method

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Check and JAVA algorithms in tag data

\*

\* @param sendbyte You need to compute the checksum interval: 1 byte tag TYPE + 4 byte tag ID

\*

\* @return Calculate the checksum

\* 20 E3 AF 22 32 checksum：FA(10 dex -6）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

protected static byte sendRcvByteNum(byte[] sendbyte) {

byte sum = 0;

for (int i = 0; i < sendbyte.length; i++) {

sum += sendbyte[i];

}

byte rebyte = (byte) (~sum + 1);

System.out.println("校验位：" + rebyte);

return rebyte;

}

// Check sum of the calculated and 20 E3 AF 22 32: FA(10 is -6)

public static void main(String[] args) {

byte[] b = new byte[5];

b[0] = 0x20;

b[1] = (byte) 0xE3;

b[2] = (byte) 0xAF;

b[3] = 0x22;

b[4] = 0x32;

sendRcvByteNum(b);

}

# The appendix

## Relationship between battery power and battery voltage

The battery voltage on the MR7901 is shown as follows in relation to the battery charge in the heartbeat message (or in the upload device hardware information)

|  |  |
| --- | --- |
| **Battery power** | **battery voltage** |
| 10 | ≥ 8.00 |
| 9 | ≥ 7.75, ＜8.00 |
| 8 | ≥ 7.50, ＜7.75 |
| 7 | ≥ 7.25, ＜7.50 |
| 6 | ≥ 7.00, ＜7.25 |
| 5 | ≥ 6.75, ＜7.00 |
| 4 | ≥ 6.50, ＜7.75 |
| 3 | ≥ 6.25, ＜6.50 |
| 2 | ≥ 6.00, ＜6.25 |
| 1 | ≥ 5.75, ＜6.00 |
| 0 | ＜5.75 |

Note

1. The battery voltage is less than 5.75 V, that is, when the battery is less than 1, the equipment cannot work properly；

2. When the battery is less than 9, it can be judged that the external power supply has been switched off (or lost power).