Device Common Protocol V38

Catalogue

Document Status: [ ] Draft [ ] Officially released [√] Amendment [ ] Annulment	No.:	
	Type:	
	Version:	V38
	Editor ：	
	Released Date:	2023-11-22
	User:	
	Completion Date:	

Version	Amended Type	Amended Chapter	Amended Contents Overview（or Purpose）	Editor	Date
V10	A	Full text	Create a document	Feng Zhiliang	2021-12-21
V11	A	Part	Added some customization instructions (check yellow fill section for details)	Feng Zhiliang	2022-02-17
V12	A	6.10/6.11	Added terminal data request packets	Feng Zhiliang	2022-02-17
V13	A	6.1/6.4	6.1 Added 0x6E, A Terminal upgrade result Additional field 6.4 Added 0x73, terminal upgrade Instruction	Feng Zhiliang	2022-03-08
V14	A	Part	Modified individual description in the document	Feng Zhiliang	2022-05-21
V15	A	Part	Added "Normal Shutdown Reminder" event	Feng Zhiliang	2022-05-26

V16	A	Part	Added "IC Card Binding Read", deleted "Area IC card binding", added "business data binding Read", modified FTP upgrade, AES encryption, networking mode, WIFI parameters	Feng Zhiliang	2022-06-13
V17	A	Part	Added path parameters for FTP upgrade.	Feng Zhiliang	2022-06-27
V18	A	Part	Modified individual description in the document	Feng Zhiliang	2022-07-23
V19	A	Part	Modified individual description in the document	Feng Zhiliang	2022-08-08
V20	A	Part	Added dynamic and static password unlock events	Feng Zhiliang	2022-09-05
V21	A	Part	Added LORA Sub-lock swipe card event	Feng Zhiliang	2022-09-15
V22	A	Part	Added rules for Geo-fence	Feng Zhiliang	2022-12-16
V23	A	Part	Added events related to emergency physical key operations	Feng Zhiliang	2023-02-21
V24	A&M	Part	Modify 0x0200 LORA sub-lock extension data 0x63 Modify 0x0310 LORA sub-lock seal/unseal field 0x46 0x47 Add LORA sub-lock report events	Chen Dongchang	2023-03-17
V25	A&M	Part	1. Modify and add terminal parameter in Table 14 0x2F Module IMEI 0x30 Seal inspection 0x42 Modify WORD to DWORD 0x47 Set LORA sub-lock seal/unseal 2. Add 0x0200 extended data 0x80 Next report interval 3. Add 0x0200 event 0x2A Timed automatically unseal event	Chen Dongchang	2023-04-06
V26	M	Part	Collate AES encryption instructions	Chen Dongchang	2023-04-20
V27	A&M	Part	1. Modify and add terminal parameter in Table 14 0x30 seal inspection 0x5C Add 0x02 disable password 0x87 AO output setting 0x23 Add 0x0a to trigger the upload of 0200 message 0x94 Battery voltage value 2. Change LORA events name in 0200	Chen Dongchang	2023-05-05

V28	A&M	Part	1. Modify parameter Table 14 Split 0x47 into 0x47 & 0x49 0x29 ICCID parameter type changed to STRING 2. Add the data content format of 0200 message event.Adjust the length of business data written in 0218, and change DWORD to WORD 3.Update message example	Chen Dongchang	2023-05-16
V29	A&M	Part	1. Modify and increase terminal parameter Table 14 1) Add 0x94 platform encryption mode setting 2) Add 0x95 alarm status flag 3) Add 0x96 status flag 4) Add 0x97 GPS location information 5) Modify the content format of 0x56 MCU firmware version information 2. Add 0200 event 1) 0x2C handstand flip event 2) 0xA0 over-speed alarm event 3) 0xA1 parking overtime alarm event	Chen Dongchang	2023-05-31
V30	A&M	Part	Modify terminal parameter list Table 14 1) Change 0x94 platform encryption mode to 0x98 due to conflict 2)Add 0200 Geo-fence related events	Chen Dongchang	2023-06-01
V31	A&M	Part	Modify terminal parameter Table 14 1) 0x97 GPS location information, parameter length correction 2) Add 0x03 to 0x5c parameter to start random password and static password 3) 0x56 Modify the firmware version format 4) 0x2c increase GPS positioning quality 0x17 GPS PDOP default value changed to 0x003c	Chen Dongchang	2023-06-16
V32	A&M	Part	Correct the description of encryption matters	Chen Dongchang	2023-06-30

V33	A&M	Part	1. Add RSA encryption method (0610, 0611) 2. Modify terminal parameter Table 14 1) Parameter ID 0x5F blind spots saving, add 0x04 option to only save all blind spots in the Sealed state. 2) Add parameter 0xff to customize the device ID	Chen Dongchang	2023-08-21
V34	A&M	Part	Modify and add fence-related events in the **0200** message	Chen Dongchang	2023-08-25
V35	A&M	Part	Modify and add fence-related events in the **0200** message	Chen Dongchang	2023-09-04
V36	A&M	Part	Modify and add terminal parameter table 14 1) Add 0x1A APN automatic configuration mode 2) Modify 0x0F card mode switching	Chen Dongchang	2023-10-08
V37	A&M	Part	1. Modify the number of Geo-fences The rules limit 500 groups, 500 groups of circular fences, and 100 groups of polygonal fences. The protocols involved are 0x021C, 0x021D, 0x021E, 0x021F, 0x0220, 0x0221, 0x0222, 0x0223, 0x0228, 0x0229, 0x022A, 0x022B 2. When writing and reading protocols for Geo-fence rules, 0x0224, 0x0225, 0x0226, and 0x0227 will be replaced by 0x0228, 0x0229, 0x022A, and 0x022B.	Chen Dongchang	2023-10-09
V38	A	Part	Add data in Table 5 of **0200** Added Hall and Travel switch flags	Xiong Kang	2023-11-22

*Amended Type：A—ADDED，M—MODIFIED，D—DELETED

1. Overview

This document describes the interface protocol between HHD general terminal devices (such as G-Series E-lock, Bluetooth lock, Lora lock, Gateway, Sensor, GPS tracker, Intelligent box bag etc.) and the platform.

Note: In the following, the HHD terminal device is referred to as the terminal for short.

2. Communication Connection

2.1 Establish the connection

The terminal is divided into two communication methods: directly connected to the platform and indirectly connected to the platform.

Direct connection: refers to the terminal with GPRS/WIFI/RJ45 direct networking function, which can directly perform data connection interaction with the platform. The daily connection adopts the TCP method. After the terminal is powered on, it should establish a connection with the platform as soon as possible. After the connection is established, it will immediately send the terminal location and status reports to the platform.

Indirect connection: The terminal does not have the function of direct networking, so it can only acquire the ability of networking indirectly through LORA/ Bluetooth/other indirect networking methods, with the help of LORA gateway, Bluetooth gateway, mobile Bluetooth APP.

2. 2 Maintain the connection

After the connection is established, the terminal should periodically report the location status to the platform. After the platform receives it, it will send a platform general response message to the terminal. The sending period is specified by the terminal parameters. Note: When the terminal means that the terminal needs to maintain a long-term connection with the platform, the terminal needs to be able to receive sudden commands from the platform at any time.

2. 3 Disconnection

Both the platform and the terminal can actively disconnect according to the TCP protocol, and both sides should actively determine whether the TCP connection is disconnected.

The method for the platform to judge that the TCP connection is disconnected:

- ✧ Determine that the terminal actively disconnects based on the TCP protocol;
- ✧ The terminal with the same identity establishes a new connection, indicating that the original connection has been disconnected;
- ✧ No message from the terminal is received within a certain period of time (usually 10 minutes).

The method for the terminal to judge that the TCP connection is disconnected:

- ✧ The platform is judged to be actively disconnected based on the TCP protocol;
- ✧ The data communication link is disconnected;

- ✧ The data communication link is normal, but no response from the platform has been received after the number of re-transmissions has been reached.

3. Message Processing

3. 1 TCP Message Processing

3. 1.1 Messages sent proactively by the platform.

All messages sent proactively by the platform require the terminal response. After the sender times out waiting for a response, it should resend the message. The response timeout time and the number of re-transmissions are specified by the platform parameters. The response timeout time and the number of re-transmissions after each re-transmission are specified by the platform parameters. The calculation formula for the response timeout time after each re-transmission is shown in formula ①.

Formula ①：$T_{N+1}=T_N*(N+1)$

T_{N+1}——The response timeout time after each re-transmission;

T_N——The last response timeout time;

N——The number of re-transmissions.

3. 1.2 Messages sent proactively by the terminal.

3.1.2.1 Communication link connected.

When the data communication link is normal, all messages sent proactively by the terminal require the platform to respond, and the terminal should resend the message after waiting timeout for the response. The response timeout time and the number of re-transmissions are specified by the terminal parameters, and the response timeout time after each re-transmission is calculated according to formula ①. If no response is received after reaching the number of re-transmissions, it should be saved.

3.1.2.2 Communication link disconnected

When the data communication link is abnormal, the terminal should save the location information report message that needs to be sent. After the data communication link returns to normal, while ensuring real-time message transmission, the spare time is used to immediately resend the saved messages.

4 . Protocol Basis

4.1 Protocol Basis Description

Information such as operating commands and parameters is transmitted between the terminal and the platform through messages, and information exchange messages are divided into two types: downlink (platform to terminal) and uplink (terminal to platform). The commands carried in the message are distinguished by unique command codes.

4. 2 Data Type

Table 1: All Data Types

Data Type	Description & Requirements
BYTE	Unsigned single-byte integer (byte, 8 bits)
WORD	Unsigned double-byte integer (word, 16 bits)
DWORD	Unsigned four-byte integer (double word, 32 bits)
BYTE[n]	n Byte
BCD[n]	8421 code, n Byte
STRING	GBK encoding, if there is no data, leave it blank

4. 3 Transmission Rule

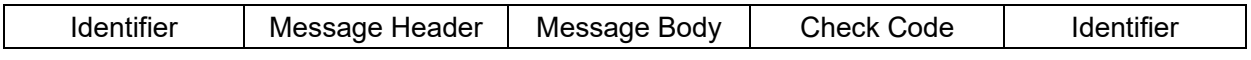
The protocol uses big-endian network byte order to transfer words and double words.
The transmission agreements are as follows:

- ✧ **BYTE :** Transmitted in the form of Byte stream;
- ✧ **WORD:** Transfer the upper eight bits first, then the lower eight bits;
- ✧ **DWORD:** First transfer the high 24 bits, then transfer the high 16 bits, and then transfer the high eight bits, the lower eight bits are passed last.

5. Message Composition

5. 1 Message Structure

Figure 1: Message Structure



5. 2 Identifier

0x7e is the identifier. If **0x7e** appears in the check code, message header and message body, it must be escaped.
The escaping rules are defined as follows:

0x7e<——>0x7d **0x7e** escape to: **0x7d 0x02** (**0x7d** followed by **0x02**);
0x7d<——>0x7d **0x7d** escape to: **0x7d 0x01** (**0x7d** followed by **0x01**)

The escaping process is as follows:

When sending a message: Message encapsulation -> Calculate and fill the check code -> Escape;

When receiving a message: escape and restore -> verify the check code -> parse the message.

Example:

Sending a data packet with the content of **0x30 0x7e 0x08 0x7d 0x55** is encapsulated as follows: **0x7e 0x30 7d 0x02 0x08 0x7d 0x01 0x55 0x7e**

5. 3 Message Header

Table 2: Message Header Contents

Start Byte	Field	Data Type	Description & Requirements
0	Message ID	WORD	
2	Message Body Attribute	WORD	The message body attribute format structure diagram is shown in **Figure 2**
4	Terminal Serial Number	BCD[6]	Generally use the last 12 digits of the IMEI code or the MAC address of the chip.
10	Message Serial Number	WORD	Cyclically accumulate from 0 in order of sending
12	Message Packet Encapsulation Item	Null/WORD[2]	1. If the relevant flag bit in the message body attribute determines that the message is processed in multiple packets, then this item has content, otherwise there is no such item. 2. WORD[0] = Total number of packages; 3. WORD[1] = Current packet serial number, starting from 1. 4. If the total number of packets = the current packet number, it means that the packet transmission is complete.
If the message is encrypted, insert the security code item in front of the message body as part of the message body.			
0	Security Code	BCD[6]	When the uploaded message is encrypted, the terminal will include the latest "security code" every time the message is uploaded, and any command sent by the platform must carry the latest received "security code". Only when the "security code" is consistent can the command sent by the platform be considered valid, preventing downlink messages from being copied and reused. (Generally, the 0 time zone time code is used)

Figure 2: Message Body Length

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Encryption Type	Connection Attribute	Multi-packet	Encryption Method: 00:AES128 01:RSA1024 02:SM2 03:SM3		Message Length (The actual data content length after decryption, the message body contains the "security code")										

Multi-packet identification：

When **Bit13** is 1, it means that the message body is a long message, which is sent in packets. The specific packet information is determined by the message packet encapsulation item; If **Bit13** is 0, there is no message packet encapsulation item field in the message header.

Connection Attribute：

When **Bit14** is 0, it means keep; When **Bit14** is 1, it means short connection; (data interaction ends, the device automatically disconnects the original communication link to further save power)

If Bit15 is 0, it means no encryption;

If Bit15 is 1, it means that encryption is enabled; (check **Bit11-Bit12** for encryption method)

For example: Bit11=0, Bit12=0, which means the message body adopts AES symmetric encryption, ECB mode, data block is 128 bits, password is 16 Byte, and Padding Mode.None is used for padding;

Things to note after encryption:

1) In order to ensure the efficiency of network interaction between the terminal and the platform, and at the same time ensure that the uplink and downlink cipher-texts of the platform and the terminal are not deciphered and copied for secondary use, the platform and the terminal will use the "security code" in the cipher-text for mutual authentication, thus ensure the "one-time" and "uniqueness" of uplink and downlink messages.

2) When the message is encrypted, an "security code" will be inserted in front of the message body as part of the message body.

3) Encryption scope: the entire message body (including "security code").

4) Each time the terminal side sends a message, it will take the latest time of the internal RTC as the "security code", and the format is: YY-MM-DD-hh-mm-ss.

5) Every time the platform side sends a message, it will take the latest time of the server as the "security code", and the format is: YY-MM-DD-hh-mm-ss.

6) When the terminal or platform receives the time "security code" of the other side, it first compares the time error. If the error exceeds 10 minutes, the message is considered invalid. When the error range is qualified and the currently received "security code" value is greater than the last stored "security code" value, the message can be considered valid, and the currently received "security code" is updated and stored for the next message comparison .

7) Before encryption, the content length of the message body is made into a multiple of 16 Byte. If the multiple of 16 is not satisfied, **0x00** will be added at the end to make it a multiple of 16 Byte; cipher-text length = ((plain-text length /16)+1)*16

8) The length of the message body refers to the length of the actual data content after decryption, and the message body contains the "security code". Please note that this does not refer to the cipher-text length.

5. 4 Message Body

Check relevant instructions in Part 5 for details.

5. 5 Check Code

The Check Code start from data message header, and XOR (Exclusive OR) of the next byte, until first one byte of the Verification Code, occupies one byte.:
Online BCC Validation tool link：http://www.ip33.com/bcc.html

6. Common Commands

6. 1 Real-time Position & Status Report（0X0200）

The terminal periodically sends position status information report messages according to parameter settings. According to parameter control, the terminal can immediately send position status information report messages when it determines that the vehicle is turning or the status changes.
The position information report message body consists of basic location information and a list of additional location information items.

Figure 3: Position report message structure

Basic position info (Table 3)	List of additional position info (Table 6)

Message ID：0x0200

Table 3: Basic Position Information data format

Start Byte	Field	Data Type	Description & Requirements
0	Alarm Identifier	DWORD	Check **Table 4** for the definition of alarm identifier
4	Status	DWORD	Check **Table 5** for the definition of status
8	Latitude	DWORD	Unit: degree, multiplied by 1,000,000. Accuracy: 0.000001 degree
12	Longitude	DWORD	Unit: degree, multiplied by 1,000,000. Accuracy: 0.0000001 degree
16	Elevation	WORD	Altitude, in meters (m)
18	Speed	WORD	0.1km/h
20	Direction	WORD	0-359, true north is 0, clockwise
21	Time	BCD[6]	(GMT+8 time, all times involved after this standard adopt this time zone)

Table 4:Definition of Alarm Identifier

Bit	Definition	Process Description
0	1：Shackle been pulled out (Under the sealing state)	

1	1：Shell tampered/damaged alarm	
2	1：Low battery alarm	
3	1：MCU communicate abnormal alarm	
4	1: Shackle / String cable been cut/ disconnected alarm	
5	1：GPS antenna open circuit alarm	
6	1：GPS antenna short circuit alarm	
7	1: Motor is stuck during sealing	Only the version with photoelectric switch can support this function
8	1：Motor is stuck during unsealing	Only the version with photoelectric switch can support this function
9	1：Over-speed alarm	Kept until the alarm condition is cleared
10	1：Timeout parking alarm	Kept until the alarm condition is cleared
11	1：GNSS module failure	Kept until the alarm condition is cleared
12	1：Main power failure	Kept until the alarm condition is released
13-31	Reserved (default 0)	Reserved (default 0)

Table 5 Status Bits Definition

Bit	Status
0	0: ACC OFF ;1:ACC ON（Padlock products don't need this ）
1	0: GPS invalid; 1: GPS valid
2	0: North Latitude; 1: South Latitude
3	0: East Longitude; 1: West Longitude
4	0: Long connection;1: Short connection
5	0: GPS module OFF; 1: GPS module ON
6	Motion sensor(G-sensor) status ------0: motion state; 1: static state
7	~~0: Travel switch closed 1: Travel switch opened~~
8	~~0: Hall sensor connected 1: Hall sensor disconnected~~
9-13	Reserved field
14	0：Unseal; 1: Seal;
15	0：Shackle opened; 1:Shackle closed;
16-17	00: No charging 01: Charging 10: Fully charged
18-19	00: default; 01:2G; 10:3G; 11:4G
20-21	0: default; 01: SIM 1; 10: SIM 2
22-23	00: Non; 01:WIFI; 10:RJ45;
24-31	Reserved field

Considering has the requirement of low power and super long endurance, the terminal working connection mode is divided into two ways: long connection and short connection.

Long connection:

The terminal needs to keep the connection state with the platform for a long time. The terminal needs to receive the burst instruction at any time. When the terminal position status reporting cycle parameter is more than 300 seconds, the heartbeat package must be sent, if the status report is not sent after 300 seconds to maintain the terminal GPRS link not being regarded as an empty link by the GSM operator cut off.

Short connection:

The terminal only needs the timing or the terminal self-state change to send the position state information to the platform actively. The terminal does not have to keep the GPRS and the platform link, and the terminal doesn't need receive the burst instruction at any time. Short connections can save battery power to a maximum to meet some special applications that need a few months to use without charging or using a disposable battery. **(When the terminal is in the short connection mode, after 0X0200 command is sent, if the platform receives the general response 0X8001 and there is no blind spot data, the terminal will immediately turn off the GSM to enter the sleep state. If the platform has a reservation command to send, the command to be sent should be finished before the 0X8001 response.)**

Table 6: Position additional message details

Field	Data Type	Description
Additional Message ID	BYTE	Allow to extend more information
Length of Additional Message	BYTE	
Additional Message		For additional information definition refer to Table 7

Table 7: Additional Message Definition

Additional Message ID	Length of Additional Message	Description
Event Extension		
0x60	Length of event information	Event ID[WORD] + Event type [Byte] + Event content [N]--Detail refers to table 7-1 Event type： 0x00----Normal events（such: lock shackle open/close） 0x01----Alarm event（such: shackle cut and device tampered）
Status data expansion		
0x01	4	Mileage, DWORD,0.1km, corresponding to the odometer reading on the vehicle

0x02	2	Fuel tank capacity, WORD 0.1L, corresponding to fuel gauge reading (or AD value)
0x03	2	Tank oil volume, WORD 1L, corresponding to on-board fuel gauge reading (or AD value)
0x04	Entire message length	Load sensor ID1[3]+AD value 1， Load sensor ID2[3]+AD value 2. Maximum of 4 groups of load sensors
0x50	WIFI_MAC length=N*7. （N<=8）	When the satellite positioning is invalid, the terminal actively uploads the current WIFI_MAC to the platform for auxiliary positioning. The WIFI_MAC information does not exceed 10 groups The format of each group is as follows: WIFI_MAC [6]+ Signal value [byte]
0x51	BLE_MAC length =N*7. （N<=8）	When the satellite positioning is invalid, the terminal actively uploads the current WIFI_MAC to the platform for auxiliary positioning. The BLE_MAC information does not exceed 10 groups The format of each group is as follows: BLE_MAC [6] + Signal value [byte] (The general version does not have a BLE positioning function, this function belongs to a customized application)
0x63	Entire message length （LORA Slave lock status） Length=11*N	（Slave ID [6]+ slave battery [2]+ Seal/Unseal status [1]+Physical state [1]+ RSSI signal value [1]）*N N indicates maximum 12 sub-locks. Such as: sub-lock ID ------ 0X31 0X39 0X33 0X31 0X38 0X38 Slave battery -------- 0x1022(4.130V) Seal/Unseal status ----- 0x30: Unseal 0x31: Seal **Physical state ----- 1 byte** Bit0: Shackle status 0 Opened 1 Closed Bit1: Shackle cut 0 Normal 1 Cut Bit2: Shell tampered 0 Normal 1 Tampered Bit3: Severe vibration 0 Normal 1 Vibration Bit4: Low battery 0 Normal 1 Low battery Bit5: Reserved（Non） 0 Bit6: Communication timeout 0 Normal 1 Timeout Bit7: Initial State 0 Initial 1 Normal Initial state: The sub-lock is bound, but no data interaction has been performed. If the master lock has just been powered on, the state of the sub-lock at this time is the initial state.

		If Bit6 is 1 or Bit7 is 0, nothing else will be displayed except the status of this bit. If the communication is overtime, it will display the communication timeout; for the initial state, it will display the initial state. Wireless RSSI signal value----RSSI is usually a negative value, and a positive value is used in transmission. For example, if RSSI=-80, the upload value is 0x50(The communication times out, and the value is 0).
0x64	Entire message length （LORA Temperature and humidity data） Length =24*N	(Temperature and humidity sensor ID [6] + battery voltage [2] + real-time temperature value [2] + upper temperature alarm value [2] + low temperature alarm value [2] + continuous over temperature alarm time threshold value [2] + real-time over temperature alarm state [1] + real-time humidity value [1] + upper humidity alarm value [1] + lower humidity alarm value [1] + continuous over humidity alarm time threshold value [2] + real-time over humidity alarm state [1] + wireless RSSI signal value [1] + x [2] + y [2] + Z [2] * n; Wireless RSSI signal value---RSSI is usually a negative value, and a positive value is used in transmission. For example, if RSSI=-80, the upload value is 0x50(The communication times out, and the value is 0). N indicates the number of bound T/H devices/Sensors N does not exceed 8
0x65	Coordinates after LBS analysis (some versions support this feature) The length value is 0x08	Latitude [DWORD]+Longitude [DWORD] The value in degrees is multiplied by 10 to the 6th power, accurate to one millionth of a degree
0x66	Entire message length （LBS base station）	When the positioning is invalid, the terminal actively uploads the current base station information to the platform for auxiliary positioning. The base station information does not exceed 3 groups at most, and the first field is the mcc mobile country code. In order to save the length of the agreement, each group of base stations will no longer carry the mcc mobile country code repeatedly. **Mcc**: mobile country code [WORD] The information of each group of base stations is as follows (up to 3 groups): **rxl**: received field strength [BYTE] **mnc**: mobile network code [WORD] **cellid**: Cell ID [DWORD] **lac**: location area number [WORD]
0×67	8	Keypads unlock dynamic password (with P model support, such as G300P, the device will automatically generate

		random passwords after unlocking each time)
0x68	4	The percentage of battery power is 4Byte (reserve two decimal places in the percentage, and it will be enlarged by 100 times.) For example: 76.51% = 7651 Value range: 0 - 10000
0x69	2	Battery voltage: 2Byte (unit :10mV)
0x6a	1	Network CSQ signal value: 1Byte (range: 0-31)
0x6b	1	Number of satellites used: 1Byte
0x6C	SIM_IMSI value [N]	IMSI code of the SIM ---- The upload is triggered when the network is reconnected
0x6D	BYTE[N]	Client ID (0X0310 can be configured. After the configuration, each 0200 packet will be uploaded with the client ID.)
0x6E	STRING	Terminal upgrade result: Upgrade result of the current version, for example, 0, G4-360_20220301 Upgrade result: 0 - The upgrade succeeds. 1 - The upgrade fails
0x80	BYTE + DWORD	Format: current report interval mode + next 0x0200 report interval Current return interval mode: 0x01 Low battery 0x02 Long time static 0x03 Seal 0x04 Useal 0x05 Shutdown 0x06 Low battery shutdown Notice: The report interval in shutdown and low power shutdown mode is 0xFFFFFFFF. Priority: Low battery report interval> Schedule timetable report interval (IMZ customized) > Static report interval > Seal/unseal report interval (normal timing) Next 0x0200 timing report interval unit: second 0xFFFFFFFF means that there is no more regular reporting
Fields used by devices to request AGPS		
0x6F	STRING	Hardware version information
0x70	STRING	Software version information

0x71	STRING	Terminal ICCID code
0x72	STRING	TERM_ID
0x73	STRING	Bluetooth name
0x74	BYTE[6]	Bluetooth MAC address of the chip itself: MAC[6]
0x75	BYTE[6]	WIFI MAC address of the chip itself: MAC[6]
0x76	1	GNSS Module Manufactuer 0x01: zkw------中科微 0x02: ublox------U-BLOX

Table 7-1 Event Content

Event ID	Event	Event Type	Event Content
Events that occur on the device itself			
0x0000	Shackle closed	0x00（Normal）	0x00（Default）
0x0001	Shackle opened	0x00（Normal）	0x00（Default）
0x0002	Close shackle auto sealed	0x00（Normal）	0x00（Default）
0x0003	Swipe IC card to seal successful	0x00（Normal）	8 digits card ID
0x0004	Swipe card to unseal successful	0x00（Normal）	8 digits card ID
0x0005	BLE seal successful	0x00（Normal）	APP_BLE user name or ID;
0x0006	BLE unseal successful	0x00（Normal）	APP_BLE user name or ID;
0x0007	Platform seal successful	0x00（Normal）	Platform user name or ID
0x0008	Platform unseal successful	0x00（Normal）	Platform user name or ID
0x0009	SMS seal successful	0x00（Normal）	SMS mobile phone number
0x000A	SMS unseal successful	0x00（Normal）	SMS mobile phone number
0x000B	Unregistered IC card seal fails	0x00（Normal）	8 digits card ID
0x000C	Unregistered IC card unseal fails	0x00（Normal）	8 digits card ID
0x000D	IC card seal fails due to outside of the area	0x00（Alarm）	8 digits card ID
0x000E	IC card unseal fails due to outside of the area	0x00（Alarm）	8 digits card ID
0x000F	Shackle been opened under sealing states	0x01（Alarm）	0x00:（Default）
0x0010	Device Shell was tampered/removed	0x01（Alarm）	0x00:（Default）
0x0011	Shackle /Wire rope been cut	0x01（Alarm）	0x00:（Default）
0x0012	Battery voltage low get offline	0x01（Alarm）	0x00:（Default）

0x0013	Too many incorrect passwords are entered	0x01(Alarm)	0x00:（Default）
0x0014	Auto seal due to after unsealing for long time shackle did not been pull out	0x00（Normal）	0x00:（Default）
0x0015	Short connection handstand activation online	0x00（Normal）	0x00:（Default）
0x0016	Short connection touch activates online	0x00（Normal）	0x00:（Default）
0x0017	User set low battery threshold alarm reminder	0x01(Alarm)	0x00:（Default）
0x0018	Motor got stuck during sealing	0x01(Alarm)	0x00:（Default）
0x0019	Motor got stuck during unsealing	0x01(Alarm)	0x00:（Default）
0x001A	GPS antenna open circuit	0x01(Alarm)	0x00:（Default）
0x001B	GPS antenna short circuit	0x01(Alarm)	0x00:（Default）
0x001C	MCU communication abnormal	0x01(Alarm)	0x00:（Default）
0x001D	Full charged	0x00（Normal）	0x00（Default）
0x001E	Disconnected charger	0x00（Normal）	0x00（Default）
0x001F	Connected charger	0x00（Normal）	0x00（Default）
0x0020	Device/Terminal shut down reminder	0x00（Normal）	0x00（Default）
0x0021	Motion wake up under short connection	0x00（Normal）	0x00（Default）
0x0022	Static passwords unlock events	0x00（Normal）	Static unlock password value; (usually 8 digits)
0x0023	**APP reports BLE sealing**	**0x00（Normal）**	**APP_BLE username or ID;(Used when APP is connected to pure Bluetooth lock)**
0x0024	**APP reports BLE unsealing**	**0x00（Normal）**	**APP_BLE username or ID;(Used when APP is connected to pure Bluetooth lock)**
0x0025	Business data changes	0x00（Normal）	0x00（Default）
0x0026	Dynamic passwords unlock events	0x00（Normal）	Dynamic unlock password value; (usually 8 digits)
0x0027	Emergency key unseal event	0x00（Normal）	0x00（Default）
0x0028	Emergency key seal event	0x00（Normal）	0x00（Default）
0x0029	Emergency physical key is not closed, sealing failure event	0x00（Normal）	0x00（Default）

0x002A	Scheduled unseal events	0x00（Normal）	0x00（Default）
0x002C	Handstand flip (180 degree rotate)	0x01（Alarm）	0x00（Default） This event is only applicable to the use of fixed equipment, such as: smart box, fixed lock, etc.
GEO-FENCE Related Events			
RFID/NFC/IC Cards			
0x0030	Geo-Fence in invalid date swipe card, unseal failure	0x00（Normal）	8 digits card ID: Format: string
0x0031	Geo-Fence in invalid date swipe card, seal failure	0x00（Normal）	8 digits card ID: Format: string
0x0032	Geo-Fence in invalid time swipe card, unseal failure	0x00（Normal）	8 digits card ID: Format: string
0x0033	Geo-Fence in invalid time swipe card, seal failure	0x00（Normal）	8 digits card ID: Format: string
0x0048	Swipe unregister seal card fail in Geo-Fence	0x00（Normal）	8 digits card ID: Format: string
0x0049	Swipe unregister unseal card fail in Geo-Fence	0x00（Normal）	8 digits card ID: Format: string
0x004A	Swipe unseal card successful in Geo-Fence	0x00（Normal）	8 digits card ID: Format: string
0x004B	Swipe seal card successful in Geo-Fence	0x00（Normal）	8 digits card ID: Format: string
0x004C	Outside Geo-Fence swipe unseal card failure	0x00（Normal）	8 digits card ID: Format: string
0x004D	Outside Geo-Fence swipe seal card failure	0x00（Normal）	8 digits card ID: Format: string
Password			
0x0034	Geo-Fence invalid password, unseal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0035	Geo-Fence invalid password, seal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0036	Geo-Fence in invalid date password unseal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String

0x0037	Geo-Fence in invalid date password seal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0038	Geo-Fence in invalid time password unseal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0039	Geo-Fence in invalid time password seal failure	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x004E	Password unseal successful in Geo-Fence	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x004F	Password unseal failure in Geo-Fence	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0114	Password seal successful in Geo-Fence	0x00（Normal）	Unlock password value usually 8 digits. Format: String
0x0115	Password seal failure in Geo-Fence	0x00（Normal）	Unlock password value usually 8 digits. Format: String
SMS			
0x003A	Geo-Fence invalid SMS, unseal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x003B	Geo-Fence invalid SMS, seal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x003C	Geo-Fence in invalid date SMS unseal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x003D	Geo-Fence in invalid date SMS seal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x003E	Geo-Fence in invalid time SMS unseal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x003F	Geo-Fence in invalid time SMS seal failure	0x00（Normal）	SMS mobile phone number. Format: String
0x0100	SMS seal successful in Geo-Fence	0x00（Normal）	SMS mobile phone number. Format: String
0x0101	SMS unseal successful in Geo-Fence	0x00（Normal）	SMS mobile phone number. Format: String
0x0102	SMS seal failure outside Geo-Fence	0x00（Normal）	SMS mobile phone number. Format: String

0x0103	SMS unseal failure outside Geo-Fence	0x00（Normal）	SMS mobile phone number. Format: String
BLE			
0x0040	Geo-Fence invalid BLE, unseal failure	0x00（Normal）	Platform user name or ID Format: String
0x0041	Geo-Fence invalid BLE, seal failure	0x00（Normal）	Platform user name or ID Format: String
0x0042	Geo-Fence in invalid date BLE seal failure	0x00（Normal）	Platform user name or ID Format: String
0x0043	Geo-Fence in invalid time BLE unseal failure	0x00（Normal）	Platform user name or ID Format: String
0x0044	Geo-Fence in invalid time BLE seal failure	0x00（Normal）	Platform user name or ID Format: String
0x0045	Geo-Fence in invalid date BLE seal failure	0x00（Normal）	Platform user name or ID Format: String
0x0104	BLE seal successful in Geo-Fence	0x00（Normal）	Platform user name or ID Format: String
0x0105	BLE unseal successful in Geo-Fence	0x00（Normal）	Platform user name or ID Format: String
0x0106	BLE unseal failutre outside Geo-Fence	0x00（Normal）	Platform user name or ID Format: String
0x0107	BLE seal failutre outside Geo-Fence	0x00（Normal）	Platform user name or ID Format: String
Touch/Press Button			
0x0046	Geo-Fence in invalid date touch seal failure	0x00（Normal）	0x00（Default）
0x0047	Geo-Fence in invalid time touch seal failure	0x00（Normal）	0x00（Default）
0x0112	Touch seal successful in Geo-Fence	0x00（Normal）	0x00（Default）
0x0113	Touch seal failure in Geo-Fence	0x00（Normal）	0x00（Default）
LORA sub-lock Related Events			

0x0051	Web remote access gateway to unseal sub-lock via LORA successfully	0x00（Normal）	Slave ID [6] + [Remote unseal operator information] Format: Slave ID [6]: Byte displayed in HEX hexadecimal format Remote operator info: String
0x0052	APP unsealed sub-lock via BLE	0x00（Normal）	Slave ID [6] + [APP operator information] Slave ID [6]: Byte displayed in HEX hexadecimal format APP operator info: String
0x0053	APP access gateway through BLE to unseal sub-lock via LORA successfully	0x00（Normal）	Slave ID [6] + [APP operator information] Slave ID [6]: Byte displayed in HEX hexadecimal format APP operator info: String
0x0054	SMS remote access gateway to unseal sub-lock via LORA successfully	0x00（Normal）	Slave ID [6] + [mobile phone number that sent SMS] Slave ID [6]: Byte displayed in HEX hexadecimal format Mobile phone number for sending SMS: String
0x0055	Sub-lock shackle closed, auto sealed	0x00（Normal）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format
0x0056	Sub-lock shackle opened after unsealing	0x00（Normal）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format
0x0057	Sub-lock shell was tampered	0x01（Alarm）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format
0x0058	Sub-lock shackle was opened/cut in sealing state	0x01（Alarm）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format
0x0059	Sub-lock communicate timeout	0x01（Alarm）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format

0x005A	Sub-lock low battery alarm	0x01（Alarm）	Slave ID [6] Format: Byte displayed in HEX hexadecimal format
0x005B	Auto disabled LORA unsealing in moving sate	0x00（Normal）	0x00（device detects motion status and enables automatic sealing, so use 0x00）
0x005C	BLE parameter setting unsealed via LORA	0x00（Normal）	APP_BLE User Name or ID Format: String
0x005D	Web parameter setting sealed via LORA	0x00（Normal）	Platform User Name or ID Format: String
0x005E	Web parameter setting unsealed via LORA	0x00（Normal）	Platform User Name or ID Format: String
0x005F	SMS setting sealed via LORA	0x00（Normal）	[sent SMM Mobile phone number]; Format: String
0x0060	SMS setting unsealed via LORA	0x00（Normal）	[sent SMM Mobile phone number];
0x0061	Unsealed sub-lock by own registered card	0x00（Normal）	Slave ID [6] + card ID [8] Format: Slave ID [6]: Byte The display uses HEX hexadecimal Card ID [8]: String
0x0062	Sealed sub-lock by own registered card	0x00（Normal）	Slave ID [6] + card ID [8] Slave ID [6]: Byte The display uses HEX hexadecimal Card ID [8]: String
0x0063	Sub-lock swiped unregistered unseal card	0x00（Normal）	Slave ID [6] + card ID [8] Slave ID [6]: Byte The display uses HEX hexadecimal Card ID [8]: String
0x0064	Sub-lock swiped unregistered seal card	0x00（Normal）	Slave ID [6] + card ID [8] Slave ID [6]: Byte The display uses HEX hexadecimal Card ID [8]: String
0x0065	Card unsealed sub-lock through gateway	0x00（Normal）	Slave ID [6] + card ID [8] Slave ID [6]: Byte The display uses HEX hexadecimal

			Card ID [8]: String
0x0066	Card sealed sub-lock through gateway	0x00（Normal）	Slave ID [6] + card ID [8] Slave ID [6]: Byte The display uses HEX hexadecimal Card ID [8]: String
0x0067	Web remote unsealed sub-lock through gateway via LORA	0x00（Normal）	Slave ID [6] + [Remote unseal operator information] Format: Slave ID [6]: Byte The display uses HEX hexadecimal Remote operator info : String
0x0068	APP BLE sealed sub-lock	0x00（Normal）	Slave ID [6] + [APP operator information] Format: Slave ID [6]: Byte The display uses HEX hexadecimal APP operator info : String
0x0069	APP BLE sealed sub-lock via gateway LORA	0x00（Normal）	Slave ID [6] + [APP operator information] Format: Slave ID [6]: Byte The display uses HEX hexadecimal APP operator info : String
0x006A	SMS remote sealed sub-lock via gateway LORA	0x00（Normal）	Slave ID [6] + [mobile phone number that sent SMS] Format: Slave ID [6]: Byte The display uses HEX hexadecimal Mobile phone number for sending SMS: String
0x006B	BLE Parameter setting unsealed via LORA	0x00（Normal）	APP_BLE username or ID; Format: String
0x006C	Sub-lock shackle not open for a long time after unsealing, auto sealed	0x00（Normal）	Slave ID [6] Format: Byte The display uses HEX hexadecimal

LORA Temperature and Humidity Sensor Related Events			
Event ID	**Event**	**Event Type**	**Event Content**
0x0070	Temperature exceeded upper limit	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
0x0071	Temperature exceeded the lower limit alarm	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
0x0072	Humidity exceeded upper limit alarm	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
0x0073	Humidity exceeded lower limit alarm	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
0x0074	Timeout alarm event	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
0x0075	3-axis impact alarm event	0x01（Alarm）	Temperature and humidity device ID [6] Format: Byte The display uses HEX hexadecimal
Version number-related events			
Event ID	**Event name**	**Event Type**	**Event content**
0x0090	Firmware version reported event	0x00	Version number (Upload only after the device is reset and reboot) Format: String
Driving events			
0x00A0	Overspeed alarm	0x01（Alarm）	0x00(Default) If the speed exceeds the set threshold, report an over-speed alarm event. If the speed exceeds continuously or the speed returns to normal, report an over-speed alarm event within 5 minutes.

0X00A1	Overparking alarm	0x01（Alarm）	0x00(Default) 1) If the parking time exceeds the set threshold, only one event will be reported if the parking time is in a stationary state continuously. 2)Only after the movement is changed and the parking timeout threshold is reached again, the event reporting will be triggered again.

0X0200 Message Example:

7e 02 00 00 3B 01 00 36 52 64 47 51 47 00 00 00 00 00 00 00 01 01 5a 31 92 06 c8 6b a0 00 00 00 00 00 00 19 03 19 12 16 08 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 FE 7e

7e：Header Identifier

02 00：Location & status report command message ID

00 3B： Message Property (The length of the data message body, excluding the checksum code and t Frame end identifier and all the contents of the data message header)

01 00 36 52 64 47：Terminal ID

51 47：Serials Number

00 00 00 00：Alarm Flag Bit (Details please refer to table 4)

00 00 00 01：Status Bits Definition (Details please refer to 5)

01 5a 31 92：Latitude (Hexadecimal convert to decimal and divide 10^6)

06 c8 6b a0：Longitude (Hexadecimal convert to decimal and divide 10^6)

00 00：Altitude (unit: Meter)

00 00：Speed（0.1km/h）

00 00：Direction (0-359, True North is 0, clockwise)

19 03 19 12 16 08：12:16:08 on March 19, 2019（GMT+0）

66：Additional Message ID, Base Station ID, do not upload when GPS positioning is valid (Details please refer to table 7)1D：Base station message length (Max. 3 groups)

0194： MCC

31： RXL1

000B： MNC1

0000CF47： CELLID1

0069： LAC1

31： RXL2

000B： MNC2

000058A4： CELLID2

0069： LAC2

30： RXL3

000B： MNC3

00008824： CELLID3

0069： LAC3

FE：Check Code (Details please refer to Part 5.5)

7e：Frame end identifier

6. 2 Platform Universal Response(0X8001)

Message ID: 0X8001

Platform universal response data format description please refer to table 10

Table 10: Platform universal response data format

Start Byte	Field	Data Type	Description
0	Answer serial number	WORD	Serial number of the corresponding terminal message
2	Answer ID	WORD	ID of the corresponding terminal message
4	Result	BYTE	0: Succeed /Confirmed; 1: Failure; 2: Error; 3: Do not Support

Note: The terminal must obtain the platform universal response when sending a location status report data message, and the terminal data message serial number of the platform response should be consistent with the serial number in the location status report data message header, and the corresponding terminal message ID must be the same. At this point, the terminal can think that the data upload is successful.

Below is a platform universal response example, corresponding above example report data message:

7e 80 01 00 05 01 00 36 52 64 47 24 5f 51 47 02 00 00 ad 7e

7e：Frame Header Identifier

80 01 00 05 01 00 36 52 64 47 24 5f：Data Message Header (Details please refer to table 2)

51 47：Must correspond to the serial number in the Message head sent from the device.

02 00：Corresponding to the terminal message ID

00：Succeed /Confirmed

ad：Check Code (Details please see Part 5.5)

7e：Frame end identifier

6. 3 Location & Status Historical Report（0X0210）

Terminal fails to communicate with the GPRS network, and the correct response from the platform is still not received before the new data message is generated, the current Location & Status data (0X0200) will be saved locally in the terminal.

Noted：

1） Blind spot /historical data save only for 0x0200 location & status data report messages.

2） 0x0210 command can package one or more 0X0200 blind spot data in the memory upload to the platform at a time.

3） In order to ensure that the entire data package of the batch blind spot data is not easily unpacked transmitted to the platform by the GPRS network, the number of data packet uploaded by the terminal should be <600Byte.

4） The format shown in Table 11 can be used for single or batch sending.

Table 11: Batch Resend historical Data Format

Start Byte	Field	Data Type	Description
6 M	single packet …… multiple N packet	BYTE	Location&status data packet (with base station information, if any) Batch package group principle： M <= 512 Byte. Number of packets <=16 packets;

0x0210 Example 1:

7E 02 10 01 A2 01 00 36 52 64 47 00 0A 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 21 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 22 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 23 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 24 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 1C 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 25 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 2666 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 3B 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 27 66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69 1C 00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 28 F5 7E

Example description:

In this example, there are total 8 packets of location&status data to be transmitted together, of which 6 packets of data generally contain base station information because the GPS positioning is invalid, and the other 2 packets of GPS positioning are valid and generally do not contain base station information.

Example field description:

7E: Header Identifier

02 10 01 A2 01 00 36 52 64 47 00 0A: Data Message Header (Details please refer to table 2)

3B Length：one of the locations & status report data in batch, including the length value of base station information

00 00 00 00 00 00 00 01 01 58 F5 22 06 C9 16 B0 00 00 00 00 00 00 16 11 18 11 01 21: Basic location information (Detail refer to Table 3)

66 1D 01 94 31 00 0B 00 00 CF 47 00 69 31 00 0B 00 00 58 A4 00 69 30 00 0B 00 00 88 24 00 69：Base station information (Detail refer to Table 7)

Where 66 is the base station information ID, 1D is the base station data body length : 01 94 is MCC。

【31 Rxl1+00 0B Mnc1+00 00 CF 47 Cellid1+00 69 Lac1】

【31 Rxl2+00 0B Mnc2+ 00 00 58 A4 Cellid2+00 69 Lac2】

【30 Rxl3+00 0B Mnc3+ 00 00 88 24 Cellid3+00 69 Lac3】Total 3 groups of base station information.

…

1C Length: where one of the locations & status report data packets does not contain the length value of the base station information for the batch supplementary transmission.

…

F5: Verification Code (Details please see Part 5.5)

7E: Frame end identifier

This command requires a general response from the platform, see Table 10 for details; note that the serial number of the corresponding terminal message and the ID of the corresponding terminal message must be consistent.

Corresponding to the example data message of the batch location & status information supplementary report uploaded by the above terminal, the platform corresponds to the platform general response given by the platform

0x8001 Example:

7E 80 01 00 05 01 00 36 52 64 47 4A 8D 00 0A 02 10 00 1D 7E

7E：Frame Header Identifier

80 01 00 05 01 00 36 52 64 47 4A 8D：Data Message Header (Details please refer to table 2)

00 0A：Must correspond to the serial number in the message head sent from the terminal.

02 10：Corresponding to the terminal message ID_

00：Succeed /Confirmed

1D：Verification Code (Details please see Part 5.5)

7E：Frame end identifier

6.4 Terminal Parameter Setting（0X0310）

Table 12: Terminal Parameter Setting Data Format

Start Byte	Field	Data Type	Description
0	Number of Parameters	BYTE	Total number of parameters

1	Parameter List		See Table 13 for terminal parameter entry data format definitions

Table 13: Terminal Parameters Definitions

Field	Data Type	Description
Parameter ID	BYTE	Parameter ID definition and description refer to Table 14
Parameter Length	BYTE	The length value of each parameter
Parameter Value		Allows users to set multiple parameters at the same time. If multiple values are used, the data message uses multiple ID parameters

Appendix 1: List of terminal parameter items

Table 14: Terminal Parameter List

（The specific parameters of each model are subject to the HHD product model specification）

Scope	ID	Read-Write	Data Type	Description
GSM/LTE Network Parameters				
Universal	0x03	read-write	STRING	Master server address, IP or Domain Name For example: device.hhdlink.top
Universal	0x04	read-write	STRING	Back up Server, IP or Domain Name
Universal	0x05	read-write	STRING	Server TCP port example: 3809
Universal	0x07	read-write	STRING	Backup server TCP port 3808 For example:3808
Universal	0x09	read-write	STRING	APN Name (SIM 1)
Universal	0x0A	read-write	STRING	APN User Name (SIM 1)
Universal	0x0B	read-write	STRING	APN Password (SIM 1)
Universal	0x0C	read-write	STRING	APN Name (SIM 2)
Universal	0x0D	read-write	STRING	APN User Name (SIM 2)
Universal	0x0E	read-write	STRING	APN Password (SIM 2)
Univeral	0x1A	read-write	BYTE	APN automatic configuration mode 00: Disabled 01: Enabled

Universal	0x0F	read-write	BYTE	SIM mode switching: 00: Automatic switching 01: Use only SIM 1 02: Use only SIM 2
Universal	0x10	read-write	BYTE	4G/3G/2G mode switching: 00：2/3/4G； 01：4/3/2G； 02：only 2G 03：only 3G 04：only 4G
Universal	0x13	read-write	STRING	SMS access password (8 Byte) Example:66556688
Universal	0x29	read	BYTE [10]	SIM ICCID code
Universal	0x2F	read	STRING	Module IMEI code
GNSS Parameters				
Disable	0x07	read-write	BYTE	Positioning mode, defined as follows: bit0, 0：disable GPS positioning, 1：Enable GPS positioning. bit1, 0：0：disable BD positioning, 1：Enable CD positioning;
Universal	0x17	read-write	WORD	Positioning accuracy selection: Default value: 0x003C PDOP setting range: 1.0 to 99.9; Replace decimals with 10x magnification. The smaller the value, the higher the selected positioning accuracy. For example: 0x0041 = 65 = 6.5 (it means that the device considers positioning effective only when PDOP < = 6.5) If it is less than 1.0, it will be treated as 1.0; if it is greater than or equal to 99.9, it means that the PDOP value is not used to judge the positioning effectiveness, and only the 2D positioning conditions are met, that is, the positioning is considered effective.
Universal	0x18	read-write	BYTE	00: disable satellite positioning function; 01: enable satellite positioning function;
Disable	0x3a	read-write	WORD [3]	Parking timeout XXXX minutes (speed < XXXX Km and continuous > XXXX times continuously, it is considered that the parking timeout has occurred) (0x000000000000 indicates that it is not enabled)
Disable	0x3b	read-write	WORD	Maximum transport timeout (XXXX minutes) (0x0000 indicates disabled)
Disable	0x3c	read-write	BYTE [10]	Place of origin longitude 4BYTE, latitude 4BYTE, radius 2BYTE (m). If it is within the coordinate range of the starting point and the driving speed is higher than 6KM/H for 6 consecutive seconds, it is considered that the

				transportation time begins.
Disable	0x3d	read-write	BYTE [10]	Destination longitude 4BYTE, latitude 4BYTE, radius 2BYTE (meters).
Gateway	0x80	read-write	WORD	Set the parking timeout alarm time (in minutes) (0: this function is not enabled)
Gateway	0x81	read-write	WORD	Set over-speed alarm, unit: km (0: indicates this function is not enabled)
BLE Parameters				
Universal	0X26	read-write	BYTE [8]	(Bluetooth APP password), default: "12341234"
Universal	0x28	read-write	BYTE [6]	Query Bluetooth chip-set MAC
Universal	0x35	read-write	STRING	Modify Bluetooth name（ASCII）
Universal	0x39	read-write	BYTE [16]	Modify Bluetooth AES password （16 Byte）Default: 123456789012346
Only for E-locks	0x5D	read-write	BYTE	Bluetooth broadcast strategy under short connection: 00: disable broadcasting in short connection hibernation (broadcast once in 0.2 seconds after activation until hibernation turns off broadcasting) 01: enable broadcasting in short connection, broadcasting once every 1 second (broadcasting once every 0.2 seconds after activation, and broadcasting once every 1 second until hibernation) -- default 02: in the case of short connection, start broadcasting once in 0.5s (broadcasting once in 0.2S after activation, and broadcasting once again in 0.5s after hibernation)
Time interval parameter				
Disable	0x02	read-write	WORD	Default 0000, long connection mode. Timing startup duration, unit: minutes. (If the parameter is not 0, it means the short connection mode. In the absence of external trigger conditions, the terminal will start up and go online regularly according to this parameter.)

Universal	0x06	read-write	DWORD	Regular reporting interval: Unit（S）--- for example:60 seconds [hex:3C] 1. Detect terminal status change/time reach, start GPRS to upload new data package 2. If the platform sends the setting command to the server in advance, when the device goes online to send the location data packet, the server should first send the setting command and get the terminal's response to the command, and finally give the terminal a 0x8001 general response.) 3. If the report interval is >=5 minutes, the short connection mode is used to communicate with the platform by default.
Universal	0x08	read-write	WORD	Heartbeat sending interval (unit: s). The default value is 300 seconds.
Universal	0x15	read-write	BCD [7]	Real-time clock time (the device is automatically calibrated through the network and can be read) Example hex：[20][18][12][31][23][59][59]
Only for E-locks	0x3B	read-write	BYTE	In short connection mode, the maximum allowable working time after activation. Unit: minutes 0x00: all blind spots/historical data must be transmitted before hibernation. 0x01-0xff: 1-255 minutes; (the default is 10 minutes)
Disable	0x3f-0x40	read-write	WORD	Reserve (original sleep worksheet)
Only for E-locks	0x40	read-write	WORD [2]	Long time static automatic modification return interval: Static duration [min] + short connection return interval [min] For example: 0x0000 + 0x0000 --- if one of the values is less than 10 minutes, it means that the power saving strategy is not enabled. For example: 0x000a + 0x0168 ------ indicates that after the static state exceeds 10 minutes, the return interval will be temporarily set to 360 minutes, which is equivalent to 6 hours report 1 time. If motion occurs during sleep, it will automatically return to the previously set return interval.
Only for E-locks	0x41	read-write	BCD[7]*N (N<=5)	Auto unseal at setting time, maximum support 5 groups of dates and times. For example, 20 19 04 18 17 05 00 (17:05:00 on April 18, 2019) means that only one group is set. If N is 0, it means that this function is disabled.

Only for E-locks	0x42	read-write	DWORD	Special regular reporting interval in unsealing state, unit: seconds (s). If this value is 0, it means that the data is uploaded according to the return interval in the original 0x06. When this value is not 0, the original 0x06 will be used for the timing return interval in the sealed state, and this value is used for the timing return interval in the unsealed state.
Terminal with LED display	0x37	read-write	1byte	Terminal display time zone setting (total 34, 0x00----0x21) 0x00:---(UTC-12) 0x01:---(UTC-11) 0x02:---(UTC-10) 0x03:---(UTC-9) 0x04:---(UTC-8) 0x05:---(UTC-7) 0x06:---(UTC-6) 0x07:---(UTC-5) 0x08:---(UTC-4:30) 0x09:---(UTC-4) 0x0A:---(UTC-3:30) 0x0B:---(UTC-3) 0x0C:---(UTC-2) 0x0D:---(UTC-1) 0x0E:---(UTC-0) 0x0F:---(UTC+1) 0x10:---(UTC+2) 0x11:---(UTC+3) 0x12:---(UTC+3:30) 0x13:---(UTC+4) 0x14:---(UTC+4:30) 0x15:---(UTC+5) 0x16:---(UTC+5:30) 0x17:---(UTC+5:45) 0x18:---(UTC+6) 0x19:---(UTC+6:30) 0x1A:---(UTC+7) 0x1B:---(UTC+8)0x1C:---(UTC+9)0x1D:---(UTC+9:30) 0x1E: ---(UTC+10) 0x1F:---(UTC+11) 0x20:---(UTC+12) 0x21: ---(UTC+13)
Power/Battery Parameters				
Universal	0x11	read-write	WORD	Terminal alarm voltage (unit: 10MV)
Universal	0x94	read	WORD	Battery voltage value (unit: 10MV)
Clients customized products	0x90	read-write	WORD [2]	Low voltage value [1] + short connection return interval [1]; 1) When the device voltage is lower than the set threshold, it will automatically switch to the short connection mode. 2) Voltage unit: MV; Short connection return interval unit: minutes (10-65535 minutes); If the voltage value is set to 0, this power saving mode is not enabled.

Clients customized products	0x91	read-write	BYTE [7+7+4+2+2] *N groups (N<=12)	(Start date BCD [7] + end date BCD[7] + return interval [4] seconds + static timing threshold [2] seconds + static return interval [2] minutes) *N Mode 1 (stationary timing threshold=0, GPS not off): 5 seconds <= return interval <= 30 seconds; static timing threshold = 0; short connection interval = 0 minutes. Mode 2 (GPS works intermittently according to the return interval): 31 seconds <= return interval = < 600 seconds; static timing threshold < 600 seconds; short connection interval = 0 minutes. Mode 3 (Long time static GSM off, and motion is activated to upload positioning): 5 seconds <= return interval <= 600 seconds; static timing threshold >= 600 seconds; short connection interval >= 10 minutes. Mode 4 (short connection mode) 600 seconds < return interval; static timing threshold < 600 seconds; Notice: 1) When N=0, it means clearing the schedule timetable. N<=12; 2) If it is not in the schedule timetable, it will work according to the current default return interval. 3) When the power is lower than the set threshold, it will work in the low power short connection mode. 4) If the battery is too low, it will automatically sleep and wake up only when there is an event.
NFC Parameters				
Universal	0x12	read-write	STRING	IC card access password (12 Byte)
Universal	0x2a	read-write	DWORD	0: disable automatic card binding mode >=1: Time allowed for automatic card binding (unit: minutes) When reading: return the remaining valid time of automatic card binding
Universal	0x2b	read-write	BYTE	Card trigger mode: 0: automatic detection card when approaching 1: Timing detection card

Universal	0x2c	read-write	BYTE	GNSS positioning effective quality setting: 00:2D Positioning mode (poor) 01:3 D Positioning mode +1 time to reach the standard (middle) 02: 3D positioning +PDOP meet +1 time standard (good) 03:3 D positioning +PDOP meet +5 times reach the standard (excellent) -- default
G-sensor Parameters				
Universal	0x14	read-write	WORD	Vibration or 3-axis displacement detection: unit: s. Disable: 0 Second Default: 180 seconds [hex: 00b4] (when the device fails to detect the vibration or displacement of the device within 180 seconds, the device will consider it as static and turn off GPS, and the uploaded position message will adopt the last GPS coordinate value when it is stationary.)
Shackle open/close strategy parameters				
Universal	0x16	read-write	BYTE	00: Shackle closed, don't physical lock and don't seal 01: Shackle close automatically physical lock but don't seal 02：Shackle closed, automatic lock and seal
Clients customized products	0x19	read-write	BYTE	00: after unsealing, the shackle not pulling out timeout, automatically seal. 1-255: after unsealing in 1-255 minutes, the shackle not been pulled out timeout, automatically seal.
Clients customized products	0x20		BYTE	00: the 4G upload can only be triggered by abnormal events and close shackle auto seal. 01: any event will immediately trigger 4G upload.
Only for E-lock	0X27	read-write	WORD	Lock Shackle/Wire rope disconnection delay shutdown /sleep (unit: minute) **9999** means that the terminal will not shut down or sleep after the shackle/wire rope is disconnected/pulled out. **0000** means that after the shackle/wire rope is disconnected, it will shut down once gets the correct response from the platform. (But if the shackle/wore rope is disconnected continuously for more than 30 minutes, it will be forced to shut down and sleep even if it does not get the correct response from the platform)
Disable	0x3e	read-write	WORD	Shackle is not closed and sealed for a long time, including when the device is shutdown, it will automatically start up and alarm when the preset time is reached, unit: min. If it is 0x0000, this function is not enabled.

Keypad Password Configuration Parameters				
With keypad products	0x43	read-write	WORD	The number of consecutive errors allowed in one minute: 5 times by default.
With keypad products	0x5A	read-write	BYTE [8]	Random password reading or modification. (If it is set to 0000 0000, it means disabled)
With keypad products	0x5B	read-write	BYTE [8]	Static password reading or modification. (If it is set to 0000 0000, it means disabled)
With keypad products	0x5C	read-write	BYTE	Password generation mechanism: 0x00: random password is generated by the device. 0x01: Random passwords are forbidden to be generated by the device. 0x02: Disable random and static passwords. 0x03: Enable random and static passwords.
Only for E-lock	0x5E	read-write	BYTE	Touch wakes up: 00: Disable touch wake-up under short connection. 01: Enable touch wake-up under short connection. ---- by default,
WIFI Configuration Parameters				
For WIFI	0x4C	read-write	STRING	configure WiFi hotpot information, maximum 10 groups. Format: WiFi name, WiFi password, WiFi encryption method (0: WEP 1: WPA 2: WPA2 3: WPA3) .... no more than 10 groups are allowed For example: HHD, hlink2011,2, HHD_ 5G,12345678,2
For WIFI	0x4D	read-write	BYTE	WiFi enable and disable strategy: 00: Disable 01: WiFi Positioning only 02: WiFi only. 03: WiFi Positioning + WiFi networking
For WIFI	0x4E	read-write	BYTE	WiFi power saving strategy: 00: Normal mode (after startup, WiFi scans once every 10 seconds to extract Mac. It is used for location reporting and uploading. During this period, if there is bound WiFi hotpot information, take the initiative to establish a connection to the Internet.) 01: Power saving mode (WiFi scanning is started 5 seconds before the satellite positioning is invalid and the position report timing arrives. During this period, if there is bound WiFi hotpot information, it is actively connected to the Internet.)
LORA Sub-lock Configuration Parameters				

Disable	0x17 To 0X22	read-write	STRING	The ID number of the Sub-lock inside the master lock (the master lock is allowed to bind 12 Sub-lock IDS at most). If delete 1 Sub-lock, it is written with 000000,
Disable	0X25	read-write	WORD	A total of 12 Sub-locks are allowed from 01 to 12. If the corresponding BIT of each Sub-lock is 0, it means that the Sub-lock is allowed to be unsealed, and if it is 1, it means that the Sub-lock is prohibited from being unsealed. For example, 0x0123: changes to binary bits, only the last 12 bits need to be seen, i.e., 0001 0010 0011 (Sub-locks 1, 2, 6, and 9 are prohibited from unseal, and the remaining sub-locks 3, 4, 5, 7, 8, 10, 11, and 12 are allowed to unseal.)
~~For LORA~~	~~0x44~~	read-write	~~BYTE~~	~~01: Enabled --- when the vehicle is detected to be running, allow LORA to unlock.~~ ~~00: Disabled --- when the vehicle is detected to be running, prohibit LORA to unlock.~~
For LORA	0x45	read-write	Byte [6] *N (N maximum 12)	Bind Lora Sub-lock to the master lock (only the bound Sub-lock, Sub-lock ID and status can be uploaded to the platform through the master lock.) Format: Sub-lock ID1 [6], Sub-lock ID2 [6] Maximum: 12 Sub-locks Noted: If the ID is 6 0x00 (i.e.,000000), it indicates delete. Each time binding is performed, the device will clear all Sub-lock IDs of the original binding.
For LORA	0X46	write	Byte [1] +STRING	Seal/Unseal flag (Byte [1]) + username (STRING) 0x00: all bound Sub-locks are set to allow Lora unlocking. 0x01: all bound Sub-locks are set to prohibit Lora from unlocking;
For LORA	0X47	write	N + N*(Byte [6] + Byte) + STRING	Set LORA slave-lock to seal/unseal. Format: N number of slave-locks (Byte) + N* (slave-lock ID1[6] + seal/unseal flag Byte) + username (STRING) **Seal/Unseal flag:** 0x00: slave-lock setting LORA unseal. 0x01: slave-lock setting LORA seal. Notice: The Sub-lock for the operation must be bound
For LORA	0x48	read-write	BYTE [16]	LoRa AES password (16 Byte) Default: 123456789012346
LoRa Temperature and Humidity Sensor Parameters				

For LORA （Temperature and humidity）	0x3F	read-write	Byte [16] *N(N maximum 8)	Lora temperature and humidity sensor binding format: (LORA temperature and humidity sensor ID [6] + upper temperature alarm value [2] + lower temperature alarm value [2] + continuous over temperature alarm time threshold value [2] + upper humidity alarm value [1] + lower humidity alarm value [1] + continuous over humidity alarm time threshold value [2] * N Noted: If the ID is 6 0x00, it indicates deletion. Each time binding is performed, the device will clear all the sensor IDs that have been bound.
Storage / sharing file parameters				
Universal	0x5F	read-write	BYTE	Blind spot storage rules: 0x01 blind spot only saves events. 0x02 blind spot only saves the track after sealing. 0x03 blind spot saves all data---- by default 0x04 blind spot saves all data onlly in the sealed state
For video lock	0x64	read-write	STRING	The path address of the platform server for automatic uploading of shared files within the terminal Shared file explanation: 1) Shared file type: video, picture, document and other types and formats. 2) Shared file transmission channel: 4G channel and WiFi hotpot channel can be downloaded, uploaded and deleted. 3) Source of shared files: uploaded by users and automatically generated by device (such as close shackle, seal/unseal operation, shackle cut, shell been tampered etc) 4) Storage location of shared files: terminal internal storage, TF card and background server.

For video lock	0x65	read-write	BYTE	Shared file upload strategy configuration: 00: do not upload to the server. 01: automatically upload all shared files to the server. 02: only upload all document. 03: only upload all photos. 04: only upload all short videos type. 05: only upload all alarm files. 06: only upload alarm videos. 07: only upload alarm photos;
For video lock	0x66	read-write	WORD	Shared file automatic deletion strategy: For example: 0000 indicates that all shared files will be deleted automatically after the unsealing. For example, 0001 indicates that the file will be deleted automatically after the upload is successful. For example, 0005 means that only save the latest 5 files
For video lock	0x68	read-write	BYTE	Shared storage location: 00: terminal internal (default); 01: TF card;
For video lock	0x69	read	DWORD [3]	Query the number of files in the current terminal shared record. DWORD [0] = "number of files not uploaded" DWORD [1] = number of uploaded files DWORD [2] = total number of documents
Upgrade Version Parameters				
Universal	0X56	read	STRING	MCU firmware version information Format: MCU1 version information: MCU2 version information Information format of each MCU version: device type -project name -MCU number -version number -time Example: W-800ZK-1-V1_0_0-230616: W-800ZK-2-V1_0_0-230616 Each version information is separated by an English colon: W-800ZK-1-V1_0_0-230616 represents MCU1 version information. W-800ZK-2-V1_0_0-230616 represents MCU2 version information

~~Disable~~	~~0x57~~	~~read-write~~	~~STRING~~	~~Firmware version of the MCU to be upgraded ,such as:G-310N-20201230~~
~~Disable~~	~~0x58~~	~~read-write~~	~~STRING~~	~~Upgrade firmware HTTP address, for example, http://gpslock.vip/ota/G-310N-20201230.bin~~
Universal	0x59	read-write	STRING	MD5 check code
For video lock	0x61	read-write	STRING	Terminal_APP_APK Version
For video lock	0x62	read-write	STRING	Terminal _APP_APK Version to be upgraded
For video lock	0x63	read-write	STRING	Terminal _APP_APK Download address
For video lock	0x67	read-write	STRING	Path address for the terminal to download the shared file from the server
For FTP upgrde	0x73	read-write	STRING	Terminal upgrade instruction: MCU serial number, IP: port, path (including file name), username, password, file name. Example: 1,61.145.69.198:21, ota/file, hhd,123456, G4-360-20220301.bin File Description: the last two Byte of the bin file are unsigned CRC-16. After downloading the file, the device performs a CRC check and comparison. If it passes, the device will be upgraded. MCU serial number: the terminal has multiple MCUs, generally the main MCU is 1, the sub MCU is 2, and the extension MCU is 3.
Disable	0x01	read-write	BCD [6]	Terminal serial number can be customized by the user. Generally, there are three methods: 1.Use mobile phone number 2.Use IMEI in the terminal 3.Use terminal label number
Video Terminal Configuration Parameters				
For video lock	0x6a	read-write	BYTE	Video enable/disable strategy: 00: Disable. 01: Automatic recording after the terminal is turned on (double cameras enabled). 02: Automatic recording after the terminal is turned on (only the main camera is turned on). 03: Automatic recording after the terminal is turned on (only the auxiliary camera is turned on). 04: Recording when personnel approach (double cameras enabled). 05: Recording when personnel approach (turn on the main camera);

				06: Recording when personnel approach (start the auxiliary camera);
For video lock	0x6b	read-write	BYTE	Number of photos triggered by the event (one for each camera): 00: close 01: take 1 photo * 2 02: take 2 photos * 2 03: take 3 photos * 2
For video lock	0x6c	read-write	BYTE [2]	Event triggered recording video (unit: in Second); How many seconds does the front camera record and how many seconds does the top camera record
Device Operation Instruction				
Universal	0X23	write	BYTE	01: Clear all bound IC card 02: Terminal restore factory value (Terminal will automatically restart) 03: Terminal forcibly restarts after 10 seconds 04: Clear terminal blind spot data/historical data 05: Triggers upload latest business data 0x06：Clear all circle fences data at terminal. 0x07：Clear all polygonal shape fences data at terminal. 0x08：Clear all seal/unseal rules of fences data at terminal. 0x09：Clear all circle+ polygon + electronic fence in the device to apply seal/unseal rules. 0x0A: Triggers to upload the 0200 packet
For E-lock	**0X24**	write	**BYTE+ STRING**	Seal/Unseal + User ID（Such as platform user name /APP user name） 00：Unseal 01：Seal Return： 00 Succeed 01 Failure 02 Terminal shackle/wire rope disconnects and seal failure
For E-lock	**0X30**	read	**BYTE**	**Seal Inspection** **0x30：Unseal 0x31：Seal**
For video lock	0x6d	write	BYTE	01: clear all shared file image storage of the terminal 02: notify the terminal to obtain and download the shared file picture from the server immediately 03: platform applies for video call; (the device automatically answers) 04: the platform applies for audio calls; (the device

				automatically answers)
Device Diagnostic Instruction				
Maintain diagnostic instructions	0x70	read-write	Byte [11]	Network and blind spot storage diagnosis: Networking status [1] + base station CSQ value [1] + last online duration [4] + external memory status [1] + number of historical data saved [2] + number of historical data sent [2] Example description: Networking status [1] ------ 0x00: normal; (unable to connect to the Internet, the reason can only be obtained through Bluetooth APP) 0x01: communication module startup failed; (possible hardware failure or low battery level) 0x02: the module fails to detect the SIM. 0x03: unable to register with the network (possible SIM business problems, such as arrears, downtime, APN error). 0x04: unable to connect to the platform (possibly IP or port error or platform failure). 0x05: the platform has no location response command (generally caused by problems such as devices not registered in the platform). Last online duration [4] - 0x0000000A: 10 minutes External memory status [1] - 0x00: normal; 0x01: abnormal. Number of historical data saved: [2] - 0x000a: 10 Number of historical data sent: [2] - 0x0005: 5
Maintain diagnostic instructions	0x71	read-write	Byte [26]	Positioning status diagnosis: Static and motion state [1] + positioning state [1] + positioning module state [1] + positioning PDOP [2] + real-time longitude coordinate [4] + real-time latitude coordinate [4] + total number of satellites Participating in the solution [1] + number of BD satellites Participating in the solution [1] + number of GPS satellites Participating in the solution [1] + CN value of 10 satellites with the strongest signal [10] Example description:

				Static and motion state [1] - 0x00: motion state; 0x01: static state; 0x02: motion sensor fault; Positioning status [1] - 0x00 positioning is invalid; 0x01: 2D positioning; 0x02: 3D positioning. Positioning antenna status [1] - 0x00: normal; 0x01: module fault; 0x02: Antenna open circuit; 0x03: Antenna short circuit. Positioning PDOP [2] - PDOP range: 0.5 to 99.9; Replace decimal by enlarging 10, e.g., 0x0041 = 65 = 6.5 Real-time longitude coordinate [4]: directly output the coordinate value of the positioning module, For example: 0x06c916b0 is the longitude value in degrees multiplied by the 6th power of 10. Real-time longitude coordinate [4]: directly outputs the coordinate value of the positioning module, For example 0x0158f522: the latitude value in degrees is multiplied by the 6th power of 10. Total number of satellites Participating in the calculation [1] - 0x0b: 11 Number of BD satellites involved in the calculation [1] - 0x06: 6 Number of GPS satellites involved in the solution [1] - 0x05: 5 The CN values of the 10 satellites with the strongest signal are [10] - 0x32, 0x30, 0x2F, 0x22, 0x22, 0x20, 0x18, 0x17, 0x15, 0x14
Maintain diagnostic instructions	0x72	read-write	Byte [1]	LORA function status diagnosis: LORA status [1] Example description: LORA status [1] ----- 0x00: LORA chip is normal. 0x01: LORA chip is abnormal. 0x02: Without LORA
Gateway	0x82	read-write	WORD	Calibration mileage, unit: 0.1KM
Gateway	0x83	read-write	BYTE	Peripheral configuration: BIT0: oil sensor (0: enable 1: disable) BIT1: tank level meter (0: enable 1: disable)
Gateway	0x84	read-write	BYTE[N]	configure multiple load sensors, maximum support 4 sensors. Format: Load sensor ID1[3], Load sensor ID2[3] ....

Gateway	0x85	read-write	BYTE	Protocol reserved devices' output IO: Set the high- and low-level output of the expansion IO port. (Maximum 8 channels of extended IO output) Bit0 --- bit7: (0: low level; 1: high level;)
Gateway	0x86	read-write	BTYE	Protocol reserved devices' output IO: Read the high and low level of the input expansion IO port. (Maximum 8 extended IO inputs) Bit0 --- bit7: (0: low level; 1: high level;)

Scope	**ID**	**Read-Write**	**Type**	**Description**
Clients' customization instruction	0x91	read-write	BYTE[7+7+4+2+2] *N group (N<=12)	(Start date BCD [7] + end date BCD [7] + return interval [4] seconds + static timing threshold [2] seconds + static return interval [2] minutes) * N **Mode 1** (static timing threshold = 0, GPS does not turn off): 5 seconds < = return interval < = 30 seconds; Static timing threshold = 0; Short connection interval = 0 minutes. **Mode 2** (GPS works intermittently according to the return interval): 31 seconds < = return interval = < 600 seconds; Static timing threshold < 600 seconds; Short connection interval = 0 minutes. **Mode 3** (long-time stationary GSM off, motion active positioning upload): 5 seconds < = return interval < = 600 seconds; Static timing threshold > = 600 seconds; Short connection interval > = 10 minutes. **Mode 4** (short connection mode) 600 seconds < return interval; Static timing threshold < 600 seconds. be careful: 1) When N = 0, it means to clear the time work schedule. N<=12. 2) If it is not in the work schedule, it will work according to the current default return interval. 3) When the power is lower than the set threshold, it is working in the low power short connection mode. 4) If the power is too low, it will sleep automatically and wake up only when there is an event.

Platform message configuration	0x92	read-write	BYTE[N](N<=16)	Client ID (if this item is configured as 0x00, it indicates deletion) can support through IC card to swipe and write information. It can be set to 16 Byte at most and always will come with 0200 command for uploading.
Platform message configuration	0x93	read-write	BYTE [16]	Platform AES128 key encryption.
Platform message configuration	0x98	read-write	BYTE	Platform encryption method: 0x00：No Encryption 0x01：AES128 0x02：RSA1024 0x03：SM2 0x04：SM3
Universal	0x95	read	DWORD	Alarm status flag, as defined in Table 4
Universal	0x96	read	DWORD	Status flags, as defined in Table 5
Universal	0x97	read	BYTE [14]	GPS location information Format: Latitude (DWORD) + longitude (DWORD) + altitude (WORD) + Speed (WORD) + Direction (WORD) Note: Latitude: The value of latitude in degrees multiplied by 10 to the sixth power, accurate to millionths of a degree Longitude: The value of longitude in degrees multiplied by 10 to the sixth power, accurate to millionths of a degree Altitude: in meters (m). Speed: 0.1km/h Direction: 0-359, 0 due north, clockwise
Internal use only (not for external access)				
Internal use only	0XFF	Read-write	Byte[14]	Custom Device ID Format: Country code byte[2] + Customer abbreviation byte[2] + Device ID byte[10] Example: CNNT1234567890 Country code: CN Customer abbreviation: NT Device ID: 1234567890 Note: Since the device ID is only 10 digits, if it is less than 12 digits, automatically fill in with two leading zeros. The actual reported device ID is 001234567890.

Internal use only	0xFE	Write	Byte	LoRa test command. Default setting is 0. Return value: 0 - Success 1 - Failure

0X0310 Example：

7e 03 10 00 07 76 80 63 20 55 49 00 58 01 06 04 00 00 00 b4 52 7e

7e：Header Identifier

03 10 00 07 76 80 63 20 55 49 00 58：Data Message Header (Details please refer to table 2)

01: Total number of parameters

26：Parameter ID (Regular reporting interval setting)

04 ：Parameter length

00 00 00 b4: 180 seconds

52: Verification Code (Calculation methods refer to 5.5 Part for detail)

7e：Frame end identifier

6.5 Terminal parameter setting response（0X0311）

Table 15: Terminal parameter setting response Data Formats

Start Byte	Field	Data Types	Description
0	Answer serial number	WORD	Corresponding terminal data message serial number
2	Number of Parameters	BYTE	Total number of parameters
3	Parameter item list		See Table 16 for Data Format Definition of Setting Terminal Parameters

Table 16: Set Terminal Parameter Data Format

Field	Data Types	Description
Parameter ID	BYTE	Parameter ID definition and description refer to Table 14
Result Value	BYTE	0：Succeed; 1：Failure.

0X0311 Example:

7e 03 11 00 05 76 80 63 20 55 49 00 10 00 58 01 06 00 f1 7e

7e：Header Identifier

03 11 00 05 76 80 63 20 55 49 00 10：Data Message Header (Details please refer to table 2)

00 58 ： Answer serial number (0x0058) must correspond to the serial number in the data message head sent from the device

01: Total number of parameters

06：Parameter ID (0x06 Report interval)

00 ： Result value (0 success)

F1: Verification Code (Calculation Method Details please see Part 5.5)

7e：Frame end identifier

6.6 Terminal parameter query（0X0312）

Table 17: Terminal parameter query Data Format

Start Byte	Field	Data Types	Description
0	Number of Parameters	BYTE	Total number of parameters
1	Parameter ID which to be queried	BYTE [n]	The ID definitions of related parameters refer to Table 14

0X0312 Example:

7e 03 12 00 04 01 00 36 52 64 47 33 1b 03 03 05 06 78 7e

7e：Header Identifier

03 12 00 04 01 00 36 52 64 47 33 1b： Data Message Header (Details please refer to table 2)

03: Total number of parameters

03 05 06: ID definitions of related parameters refer to Table 14

78 ： Verification Code (Calculation Method Details please see Part 5.5)

7e：Frame end identifier

6.7 Terminal parameter query response（0X0313）

Table 18: Terminal parameter query response Data Formats

Start Byte	Field	Data Types	Description
0	Answer serial number	WORD	Corresponding terminal data message serial number

2	Number of Parameters	BYTE	
3	Parameter item list		See Table 19 for Terminal parameter query response Data Format Definition

Table 19: Terminal Parameter Querying Data Format

Field	Data Types	Description
Parameter ID	BYTE	parameter ID definition refer table 14 for detail
Parameter length	BYTE	Return each parameter's length
Parameter Value		Returns according to the ID at the query. If the multi valued parameter is used, the parameter items of multiple ID are returned in the data message, and the parameters to query ID are defined and explained in table 14

0X0313 Example:

7e 03 13 00 1d 01 00 36 52 64 47 09 01 33 1b 03 03 04 00 00 00 3c 05 0e 67 70 73 6c 6f 63 6b 2e 6f 6e 6c 69 6e 65 06 04 33 38 30 38 17 7e

7e：Header Identifier

03 13 00 1d 01 00 36 52 64 47 09 01：Data Message Header (Details please refer to table 2)

33 1b ： Must correspond to the serial number in the data message head sent from the device.

03: Total number of parameters

03 04 00 00 00 3c: Report interval -----60 seconds

05 0e 67 70 73 6c 6f 63 6b 2e 6f 6e 6c 69 6e 65：Domain Name-----device.hhdlink.top

06 04 33 38 30 38: TCP Port-----3808

17: Verification Code (Calculation Method Details please see Part 5.5)

7e：Frame end identifier

6.8 IC card setting rules（0X0214）

Table 6.8: IC card setting data format

Start Byte	Field	Data Types	Description
0	IC card storage sector block number	BYTE	**0-9 (10 blocks in total), each storage sector block can store 100 IC cards.**
1	IC card storage	(WORD+BYTE	**IC card storage address**

	address No.: WORD + IC Card No.: byte [4]	[4])*n N<=100	00 block: 0-99 address Block 01: 100-199 address Block 02: 200-299 address Block 03: 300-399 address Block 04: 400-499 address Block 05: 500-599 address Block 06: 600-699 address Block 07: 700-799 address Block 08: 800-899 address Block 09: 900-999 address Note: 1) ASCII code of the card number is: 12345678, which changes to: 0x12345678 after converting 2) The letter "A" represents a value from 0 to 9. For example, 123456AA means that all 100 IC cards from 12345600 to 12345699 are valid. 3) If the card number is 0x00000000, it means that the IC card number on the address serial number is cleared and deleted.

0X0214 Example:

7E 02 14 00 0D 56 90 67 45 35 78 00 05 00 00 00 90 01 93 30 00 01 12 34 56 78 8C 7E

7E：Header Identifier

02 14 00 0D 56 90 67 45 35 78 00 05：Data Message Header (Details please refer to table 2)

00 ： IC card storage sector block number (block number 0)

00 00 90 01 93 30 00 01 12 34 56 78： IC card binding address and IC card number (IC card storage address 0000 + card ID 90019330 + IC card storage address 0001 + card ID 12345678)

8C ： Verification Code (Calculation Method Details please see Part 5.5)

7E ： Frame end identifier

6.9 Setting IC card Response（0X0215）

Table 6.9: IC card data setting response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Corresponding terminal data message serial number
2	Result Value	BYTE	0：Success 1：Failure

0X0215 Example:

7E 02 15 00 03 56 90 67 45 35 78 00 01 00 05 00 B9 7E

7E：Header Identifier

02 15 00 03 56 90 67 45 35 78 00 01 ：Data Message Header (Details please refer to table 2)

00 05 ： Answer serial number (0x0005)

00 ：Result value (0 success)

B9 ：Verification Code (Calculation Method Details please see Part 5.5)

7E ：Frame end identifier

6.10 IC card reading（0X0216）

Table 6.10: Reading bound IC card data format

Start Byte	Field	Data Types	Description
0	IC card storage sector block number	BYTE	0-9 (10 blocks in total), each storage sector block can store up to 200 IC cards.

0X0216 Example:

7E 02 16 00 01 56 90 67 45 35 78 00 01 00 BD 7E

7E：Header Identifier

02 16 00 01 56 90 67 45 35 78 00 01 ：Data Message Header (Details please refer to table 2)

00 ：IC card storage sector block number (Block 0)

BD ：Verification Code (Calculation Method Details please see Part 5.5)

7E ：Frame end identifier

6.11 IC card reading response（0X0217）

Table 6.11: Reading bound IC card data response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Corresponding terminal data message serial number
2	Current storage sector block number	BYTE	Current storage sector block number 0-9 (10 blocks in total)
3	Number of valid IC cards in the current storage sector block	WORD	Number of valid IC cards of the current storage sector block (the maximum 100 cards)
5	Total valid IC card numbers of all storage sector blocks	WORD	Total number of valid IC card numbers of all storage sector blocks (the device automatically scans 1000 address serial numbers to retrieve the total number of currently valid IC cards.)

7	IC card storage address No.: WORD + IC Card No.: BYTE [4]	(WORD+BYTE [4])....N	IC card storage address 00 block: 0-99 address Block 01: 100-199 address Block 02: 200-299 address Block 03: 300-399 address Block 04: 400-499 address Block 05: 500-599 address Block 06: 600-699 address Block 07: 700-799 address Block 08: 800-899 address Block 09: 900-999 address Note: 1) the ASCII code of the card number is: 12345678, which changes to: 0x12345678 after converting 2) The letter "A" represents a value from 0 to 9. For example, 123456AA means that all 100 IC cards from 12345600 to 12345699 are valid. 3) If the card number is 0x00000000, the card number at the address serial number is empty.

0X0217 Example：

7E 02 17 00 13 56 90 67 45 35 78 00 07 00 01 00 00 02 00 02 00 00 90 01 93 30 00 01 12 34 56 78 92 7E

7E：Header Identifier

02 17 00 13 56 90 67 45 35 78 00 07：Data Message Header (Details please refer to table 2)

00 01 ：Answer serial number（0x0001）

00 ： Current storage block number (Block 0)

00 02 ： Number of valid IC cards currently storing sector blocks (0x0002)

00 02 ： Total number of valid IC cards for all storage sector blocks (0x0002)

00 00 90 01 93 30 00 01 12 34 56 78： IC card binding address and IC card number (IC card storage address 0000 + card ID 90019330 + IC card storage address 0001 + card ID 12345678)

92 ： Verification Code (Calculation methods refer to 5.5 Part for detail)

7E ： Frame end identifier

6.12 Write business data to terminal（0X0218）

Table 6.12 Write business data format

Start Byte	Field	Data Types	Description
0	Business data sector block number	BYTE	0-9 (10 blocks in total), each storage sector block maximum can store 512 Byte.
1	Business data length value	WORD	Business data length value (the value does not exceed 512, if 0 means to clear the business data)
3	Business data content	BYTE[N]	Business data content

6.13 Response for writing business data to terminal（0X0219）

Table 6.13 write business data response format

Start Byte	Field	Data Types	Description
0	Answer serial number	WORD	Corresponding terminal data message serial number
2	Result Value	BYTE	0：Succeed; 1：Failure.

6.14 Read business data from terminal（0X021A）

Table 6.14: Read the business data format

Start Byte	Field	Data Types	Description
0	Business data sector block number	BYTE	0-9 (10 blocks in total), each storage sector block maximum can store 512 Byte.

6.15 Response for reading business data from terminal （0X021B）

Table 6.15: Read the business data response format

Start Byte	Field	Data Types	Description
0	Answer serial number	WORD	Corresponding terminal data message number
2	Current business data sector block number	BYTE	Current storage sector block number 0-9 (total 10 blocks)
3	Valid business data length value of the current sector block	WORD	Business data length value (the value does not exceed 512, if 0 indicates that the business data is empty)
5	Total business data length values of all sector blocks	WORD	Total valid business data length of 10 business data sector blocks. (The device adds up the valid length values of each business data sector block for upload)
7	Business data content	BYTE[N]	Business data content

6.16 Write circular fence data to the device（0X021C）

Table 6.16 Write circular fence data format

Start Byte	Field	Data Types	Description
0	Total amount of circular fences to be written to the device	WORD	maximum 500 groups of Circular fences

2	Circular fence data storage sector block number	BYTE	0-6 (total 7 blocks), each storage sector block can store maximum 80 sets of circular fence data. Note: The device limits the maximum number of circular fences to 500 sets, meaning the last block can store a maximum of 20 sets.
3	Amount of circular fences to be written to the current sector of the device	BYTE	Each storage sector block can store maximum 80 sets of circular fence data. Note: The device limits the maximum number of circular fences to 500 sets, meaning the last block can store a maximum of 20 sets.
4	(Fence number + center coordinate longitude value + center coordinate latitude value + radius)*N	(WORD+DWORD+DWORD+WORD)*N N<=80	(Fence number + center coordinate longitude value + center coordinate latitude value + radius)*N Fence number range: 0-999; Radius unit: meters (if 0 meters means the fence is invalid) Note: If this item is empty, it means clear this block of fence

6.17 Response for writing circular fence data to the device （0X021D）

Table 6.17 Write circular fence data to the device response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Number of the corresponding terminal message
2	Result Value	BYTE	00：Success ； 01：Failure

6.18 Read circular fence data from the device（0X021E）

Table 6.18 Read circular fence data format

Start Byte	Field	Data Types	Description
0	Circular fence data storage sector block number	BYTE	0-6（Total 7 blocks）, Each storage sector block can store a maximum of 80 sets of circular fence data. Note: The device limits the maximum number of circular fences to 500 sets, meaning the last block can store a maximum of 20 sets.

6.19 Response for reading circular fence data from the device（0X021F）

Table 6.19 Read circular fence data response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Serial number of the corresponding terminal message
2	Current storage sector block number	BYTE	Current storage sector block number 0-6（Total 7 blocks ）
3	Amount of valid fences for the current storage sector block	WORD	Amount of valid fences for the current storage sector block;The device limits the circular fence to a maximum of 50 sets. That is, the last block has a maximum of 20 sets.
5	Total Amount of valid fences for all storage sector blocks	WORD	Total Amount of valid fences for all storage sector blocks <=500 sets

7	(Fence number + center coordinate longitude value + center coordinate latitude value + radius)*N	(WORD+DWORD+DWORD+WORD)*N N<=80	(Fence number + center coordinate longitude value + center coordinate latitude value + radius)*N Fence number range: 0-999; Radius Unit: meters (if 0 meters means the fence is invalid) Note:If this item is empty, it means this block of fence is empty.

6.20 Write polygonal fence data to the device（0X0220）

Table 6.20 Write polygonal fence data to the device format

Start Byte	Field	Data Types	Description
0	Total Amount of polygonal fence	WORD	Total Amount of polygonal fences that need to be distributed to devices <=100 sets
2	Block number of the polygonal fence data storage sector that currently needs to be written	BYTE	0-9（total 10 blocks), each storage sector block can store a maximum of 10 sets of polygonal fence data.
3	Amount of polygonal fences that need to be written for the current sector block	BYTE	Amount of polygonal fences that need to be written to the current sector block, <=10 sets

4	（Fence number+sum of vertexes+((Vertexes1 longitude+Vertex1latitude)*M)）*N	((WORD+BYTE+((DWORD+DWORD）*M))*N	M: <=10 vertexes N: <=10 sets Polygonal fence number range：1000-1999; Note: If this item is empty, it means that this block of fencing will be emptied.

6.21 Response for writing polygonal fence data to the device（0X0221）

Table 6.21 Write polygonal fence data to the device response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Serial umber of the corresponding terminal message
2	Result Value	BYTE	00：Success 01：Failure

6.22 Read polygonal fence data from the device（0X0222）

Table 6.22 Read polygonal fence data from the device format

Start Byte	Field	Data Types	Description
0	Polygonal fence data storage sector block number	**BYTE**	0-9（total 10 blocks), each storage sector block can store a maximum of 10 sets of polygonal fence data.

6.23 Response for reading polygonal fence data from the device（0X0223）

Table 6.23 Read polygonal fence data from the device response format

Start Byte	Field	Data Types	Description
0	Answer serial number	WORD	Corresponding terminal data message number
2	Current storage sector block number	BYTE	Current storage sector block number 0-9 （Total 10 blocks ）
3	Amount of valid fences for the current storage sector block	WORD	Total Amount of valid fences for current storage sector blocks
5	Total Amount of valid fences for all storage sector blocks	WORD	Total Amount of valid fences for all storage sector blocks , <=100 sets
7	(fence number + sum of vertices + ((vertex 1 longitude + vertex 1 latitude)*M))*N	((WORD+BYTE +((DWORD+D WORD）*M))*N	M: <=10 vertexes N: <=10 sets Polygonal fence number range：1000-1999; Note:If this item is empty, it means this block of fence is empty.

6.24 Write fence sealing & unsealing rules to the device （0X0224）(old)

Table 6.24 Write fence sealing & unsealing rules to the device format

Start Byte	Field	Data Types	Description
0	Total amount of the fence sealing & unsealing rules that issued to the devices	WORD	Total amount <=1000

2	Storage sector block number for fence sealing & unsealing rules	BYTE	0-19（Total 20）,Each storage sector block can hold up to 50 groups of fence sealing & unsealing rules
3	Total amount of the fence sealing & unsealing rules that need to be written for current sector block	BYTE	Each storage sector block can hold up to 50 groups of fence sealing & unsealing rules
4	（Fence number+Sealing-unsealing Method & Parameter+Allowed time period per day+Deadline date and time）*N	(WORD+BYTE[9]+BCD[4]+BCD[5])*N	（Fence number+Sealing-unsealing Method & Parameter+Allowed time period per day+Deadline date and time）*N Fence number： Circle 0-999, Polygon 1000-1999; Sealing-unsealing Method & Parameter: 0x01---Touch sealing （Default Parameters eight 0x00） 0x02---Password Sealing (parameters such as: "12341234") 0x03---Password unsealing (parameters such as: "43214321") 0x04---Card sealing & unsealing （parameters such as:”800123AA” means in this fence swipe 80012300-80012399 is valid） 0x05---Timed Unsealing （parameters such as:0x0020201231235959,the first 0x00 is used for the default complement of 8 Byte） 0x06---SMS Sealing （parameters such as :eight 0x00 (Means not comparing cell phone number; cell phone number adopts BCD code method, less than 8 Byte will be supplemented by 0x00.) 0x07---SMS unsealing（parameters such as :eight 0x00 (Means not comparing cell phone number; cell phone number adopts

			BCD code method, less than 8 Byte will be supplemented by 0x00.) 0x08---BLE sealing（Parameters default eight 0x00） 0x09---BLE unsealing（Parameters default eight 0x00） 0x0A---Remote sealing（Parameters default eight 0x00） 0x0B---Remote unsealing（Parameters default eight 0x00） Allowed time period per day：starting hour minute BCD[2]+ending hour minute[2]; Deadline date & time[5]（year month date hour minute）：BCD[5]; **N：<=50 sets** Note: If this item is empty, it means clear this block fence rule.

6.25 Response for writing fence sealing & unsealing rules to the device（0X0225）(old)

Table 6.25 Write fence sealing & unsealing rules to the device Response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Serial number of the corresponding terminal message
2	Result value	BYTE	00：success 01：failure

6.26 Read fence sealing & unsealing rules from the device（0X0226）(old)

Table 6.26 Read fence sealing & unsealing rules from the device format

Start Byte	Field	Data Types	Description

0	Storage sector block number for fence sealing & unsealing rules	**BYTE**	0-19（total 20 blocks), Each storage sector block can hold up to 50 groups of fence sealing & unsealing rules

6.27 Response for reading fence seal/unseal rules from device（0X0227）(old)

Table 6.27 Reading fence seal/unseal rules response format

Start Byte	Field	Data Types	Description
0	Response serial number	WORD	Serial number of the corresponding terminal message
2	Current storage sector block number	BYTE	Current storage sector block number 0-19（total 20）
3	Amount of valid rules for Current storage sector blocks	WORD	Amount of valid rules for Current storage sector block
5	Total Amount of valid rules for all storage sector blocks	WORD	Total Amount of valid rules for all storage sector blocks
7	（Fence number+Sealing-unsealing Method & Parameter+Allowed time period per day+Deadline date and time ）*N	(WORD+BYTE[9]+BCD[4]+BCD[5])*N	（Fence number+Sealing-unsealing Method & Parameter+Allowed time period per day+Deadline date and time）*N Fence number：Circle 0-999, Polygon 1000-1999; Sealing-unsealing Method & Parameter: 0x01---Touch sealing （ Default Parameters eight 0x00） 0x02---Password Sealing (parameters such as: "12341234")

			0x03---Password unsealing (parameters such as: "43214321") 0x04---Card sealing & unsealing （parameters such as:”800123AA” means in this fence swipe 80012300-80012399 is valid） 0x05---Timed Unsealing （parameters such as:0x0020201231235959,the first 0x00 is used for the default complement of 8 Byte） 0x06---SMS Sealing（parameters such as :eight 0x00 (Means not comparing cell phone number; cell phone number adopts BCD code method, less than 8 Byte will be supplemented by 0x00.) 0x07---SMS unsealing （ parameters such as :eight 0x00 (Means not comparing cell phone number; cell phone number adopts BCD code method, less than 8 Byte will be supplemented by 0x00.) 0x08---BLE sealing（Parameters default eight 0x00） 0x09---BLE unsealing （ Parameters default eight 0x00） 0x0A---Remote sealing（Parameters default eight 0x00） 0x0B---Remote unsealing（Parameters default eight 0x00） Allowed time period per day：starting hour minute BCD[2]+ending hour minute[2]; Deadline date & time[5]（year month date hour minute）：BCD[5]; **N：<=50 sets** Note: 1. If this item is empty, it means clear this block fence rules. 2. The device limits polygonal fences to a maximum of 1000 groups.

6.32 Terminal AGPS data request message（0x0400）

Message ID：0x0400。

The terminal requests AGPS to send the specified data to the platform. The 0x0400 command has basically the same format as the 0x0200 command, see Table 6.19 for details.

Table 6.19 Terminal data request message format

The location information reporting message body consists of a list of location basic information and location additional information items, as shown in the figure:

Position basic information (see Table 3, Table 4, Table 5) until the time code	The additional information list (see Table 6.19.1) extracts the necessary Parts of the 0x0200 additional information.

Table 6.19.1 AGPS Parameters（Additional Information List）

0x01	4	Mileage, DWORD, 0.1km, corresponding to the odometer reading on the car
0x50	WIFI_MAC length=N*7;（N<=20）	In case of invalid satellite positioning, the terminal actively uploads the current WIFI_MAC to the platform for auxiliary positioning, and the maximum group of WIFI_MAC information does not exceed 8. The format of each group is as follows: WIFI_MAC[6]+signal value[byte] (Universal version does not have WIFI positioning function, this function needs to be customized)
0x65	Coordinates after LBS parsing (some versions support this method) Length value is 0x08	Latitude [DWORD] + Longitude [DWORD] Value in degrees multiplied by 1000000, accuracy 0.000001 degree
0x66	By whole message length (LBS base station)	In case of invalid positioning, the terminal takes the initiative to upload the current base station information to the platform for auxiliary positioning. The maximum group of base station information is not allowed to exceed 10 groups, and the first field is mcc mobile country code. In order to save the protocol length, each group of base stations will not be repeated with mcc mobile country code later. Mcc:Mobile Country Code [WORD] The information for each group of base stations is as follows (up to 10 groups) rxl:receive field strength [BYTE] mnc:mobile network code [WORD]

		cellid:location area number[DWORD] lac:cell number[WORD]
0x68	1	Battery power percentage 1Byte (0%-100%)
0x69	2	Battery voltage value: 2Byte (unit: 10mV)
0x6a	1	Network CSQ signal value: 1Byte (range: 0-31)
0x6b	1	Amount of satellites used: 1Byte
0x6C	SIM_IMSI value [N]	IMSI[N] SIM IMSI code----Trigger upload on network re-connection
0x6F	STRING	Hardware Version Information
0x70	STRING	Software Version Information
0x71	STRING	Terminal SIM ICCID number
0x72	STRING	TERM_ID（IMEI Number）
0x73	STRING	Bluetooth name
0x74	BYTE[6]	Chip's own Bluetooth MAC address: MAC [6]
0x75	BYTE[6]	Chip's own WIFI_MAC address: MAC[6]
0x76	BYTE[1]	GNSS Module Supplier 0x01: zkw------中科微 0x02: ublox------U-BLOX

6.33 Response for terminal AGPS data request（0x8400）

Message ID：0x8400

Terminal data request response message body data format refer to Table 6.20

Table 6.20 Definition of AGPS response data

Start Byte	Field	Data Type	Description
0	Answer serial number	WORD	Number of the corresponding terminal message
2	Result	BYTE	0: success; 1: failure; 2: parameter error; 3: not supported
3	AGPS data length value	DWORD	AGPS data length value, the length value contains the last two Byte of CRC-16 Verification Code

7	AGPS data	BYTE[n]	1. AGPS data issued by the server (the last unterminated packet needs to contain CRC-16 verification code, the maximum value of each packet is 1024Byte) 2. The platform issue one data packet every 1 second until it is finished. Note that this command requires the use of sub-packet. (GPS module AGPS data generally takes about 4KB, BD+GPS takes about 6KB.)

6.38 Terminal reports RSA public key（0X0610）

Data Direction: Terminal ---> Platform

Terminal reports RSA public key

Message ID: 0x0610.

Terminal RSA public key message body data format is shown in Table 31

Table 31 Terminal RSA Public Key Message Body Data Format

Start Byte	Field	Data Types	Description
0	Terminal ID	BCD[6]	Terminal ID
6	Real-time Timestamp	BCD[6]	Real-time Timestamp
12	Random Code	DWORD	Random Code
16	e	DWORD	Terminal RSA Public Key {e, n}, where e is the exponent.
20	n	BYTE[128]	RSA Public Key {e, n}, where n is the modulus.

The above message body is encrypted using AES128.

6.39 Response for terminal RSA public key（0X0611）

Data Direction: Platform ---> Terminal

Terminal reports RSA public key response

Message ID: 0x0611.

Terminal RSA public key message body data format is shown in Table 32

Table 32 Terminal RSA Public Key Message Body Data Format

Start Byte	Field	Data Types	Description
0	Terminal ID	BCD[6]	Terminal ID
6	Real-time Timestamp	BCD[6]	Real-time Timestamp
12	Random Code	DWORD	Random Code

16	e	DWORD	Platform RSA Public Key {e, n}, where e is the exponent.
20	n	BYTE[128]	RSA Public Key {e, n}, where n is the modulus.

The above message body is encrypted using AES128.

7. AES128 Encryption Description

7.1 AES128 Terminal Uplink Data Message Encryption

Message before encryption:

7e 02 00 00 52 08 60 31 79 17 05 00 71 00 00 00 00 00 00 00 02 01 be 5d 1e 06 bd 9d f0 00 00 00 00 00 a3 18 07 04 12 00 35 33 34 2a 4d 30 30 2c 34 33 2c 31 31 34 30 39 30 30 31 32 33 34 35 36 26 30 30 30 30 30 30 30 30 30 30 30 30 26 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 23 dc 7e

Step 1: (Modify message properties and length, and insert timestamp for Anti-counterfeiting Code)

7e 02 00 80 58 08 60 31 79 17 05 00 71 18 07 04 12 00 35 00 00 00 00 00 00 00 02 01 be 5d 1e 06 bd 9d f0 00 00 00 00 00 a3 18 07 04 12 00 35 33 34 2a 4d 30 30 2c 34 33 2c 31 31 34 30 39 30 30 31 32 33 34 35 36 26 30 30 30 30 30 30 30 30 30 30 30 30 26 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 23 xx 7e

1）80 (Modify the message properties in the message header; if 0X80 is set, it indicates encryption)

2）58 (The original data length before encryption is 0X52; when encrypted, an additional 6 bytes for the timestamp anti-counterfeiting code are added, making it 0X58.)

3）18 07 04 12 00 35 (Insert the time " anti-counterfeiting code " in front of the message body)

Step 2 :(Encrypt the message body using a 16-byte key, selecting ECB mode.)

7e 02 00 80 58 08 60 31 79 17 05 00 71 (96-byte ciphertext of the message body) xx 7e

The original message body is only 88 Byte, which is not a multiple of 16. Therefore, 0X00 should be filled in during encryption, and the multiple of 16 should be expanded to perform encryption calculation. The ciphertext is 96 Byte in total after generation.)

Note:

1）58 The length of 58 refers to the actual length of the decrypted message body, not the ciphertext length. If the platform needs to use this length field to receive the message during parsing, it needs to perform the following operations to calculate the cipher-text length.0x58 is converted to 88 characters in decimal system. Since the cipher-text length must be a multiple of 16, the cipher-text length shall be calculated by the following formula:

88/16=(5 + 1)*16=96

INT + 1 to make up a multiple of 16

2）If the length of the plain-text is exactly a multiple of 16, then the extension padding 0X00 is also required

Example：

15-byte plain-text = 16-byte cipher-text (equivalent to filling 1 0x00)

16-byte plain-text = 32-byte cipher-text (equivalent to filling 16 0x00)

17-byte plain-text = 32-byte cipher-text (equivalent to filling 15 0x00)

7.2 AES128 Platform Downlink Data Message Encryption

Message before encryption:

7e 80 01 00 05 01 00 36 52 64 47 24 5f 51 47 02 00 00 ad 7e

Step 1 :(Modify message properties and length, and insert a timestamp for anti-counterfeiting purposes)

7e

80 01 （Command）

00 0b (Modify message properties, modify the original length of 5 + 6 Byte time anti-counterfeiting code =11)

01 00 36 52 64 47 （Terminal ID）

24 5f (Itself serial number)

21 12 30 23 59 59 (Insert the time “anti-counterfeiting code “in front of the message body)

51 47 (response serial number)

02 00

00

XX

7e

Message before encryption:

7e 80 01 00 05 08 60 31 79 17 05 00 01 00 71 02 00 00 xx 7e

Step 1:(Change message properties and length and insert time anti-counterfeiting code code)

7e 80 01 80 0b 08 60 31 79 17 05 00 01 18 07 04 12 00 35 00 71 02 00 00 xx 7e

1）80(Modify message properties in the message header,0X80 indicates encryption;)

2） 0b (the original length of data before encryption is 0X05, and it needs to add 6 Byte of time anti-counterfeiting code during encryption, so it is 0X0b;)

3）18 07 04 12 00 35(Insert the time " anti-counterfeiting code " in front of the message body)

Step 2: (Encrypt the message body using a 16-byte key, selecting ECB mode)

7e 02 00 80 0b 08 60 31 79 17 05 00 71 （16-byte message body cipher text） xx 7e

(The original message body is only 11 Byte, which is not a multiple of 16. Therefore, 0X00 should be filled in during encryption, and the multiple of 16 should be expanded to perform encryption calculation. The cipher-text is 16 Byte in total after generation.)

Note:

1）0x0b its length refers to the actual length of the decrypted message body, not the cipher-text length.

8. RSA Encryption Explanation

1) Preparation Before Communication:

Before the device is produced and delivered, the terminal ID and corresponding AES128-KEY will be provided to the platform for import and filing.

2) RSA Key Negotiation Exchange Process:

[1] Prior to each connection to the server, the device generates a random temporary session key pair using RSA1024. The device's RSA public key {e, n} is uploaded to the platform through AES symmetric encryption. The platform decrypts the device's RSA public key using AES. Similarly, the platform generates a random temporary session key pair using RSA1024, encrypts it with AES, and sends it to the device. This process completes the exchange of temporary session public keys between both parties under the protection of symmetric encryption.

[2] After decryption with AES, it is necessary to verify whether the device ID in the ciphertext is consistent with the device ID in the message header. Additionally, the real-time timestamp and the time deviation from the server system time should not exceed 5 minutes, or the real-time timestamp should not be less than or equal to the current latest timestamp. If any of these conditions are not met, the data is considered invalid, and the RSA exchange will not proceed.

3) **RSA Cipher Communication Process:**

The device encrypts the upstream data using the platform's temporary session public key, and the platform decrypts it using its RSA private key to obtain the plaintext.

Conversely, the platform encrypts the data using the device's RSA public key for downstream transmission, and the device decrypts it using its RSA private key to obtain the plaintext. Note that the encryption flag in the message header should be set to RSA.

4) **AES Encryption Mode:**

ECB mode is used with a data block size of 128 bits and a key size of 16 bytes. If the data is less than 16 bytes, it is padded with 0x00 and uses PaddingMode.None.

Note: For RSA encryption, the data is split into 117-byte segments, and for decryption, it is split into 128-byte segments. Manual padding with 0 is not required as the algorithm itself handles padding and removal of zeros. The length of the message body is based on the plaintext length.

RSA encryption test URL: http://tool.chacuo.net/cryptrsamodulus2pkey