# **TR05** Protocol

GPS Tracker
Communication Protocol

V2.0

# Content

# 1. Commucation Protocol

## 1.1. Introduction

This document defines the instructions of GPS vehicle tracker platform. The reference interface protocol is only applicable for the platform and server transfer.

## 1.2. Compatibility

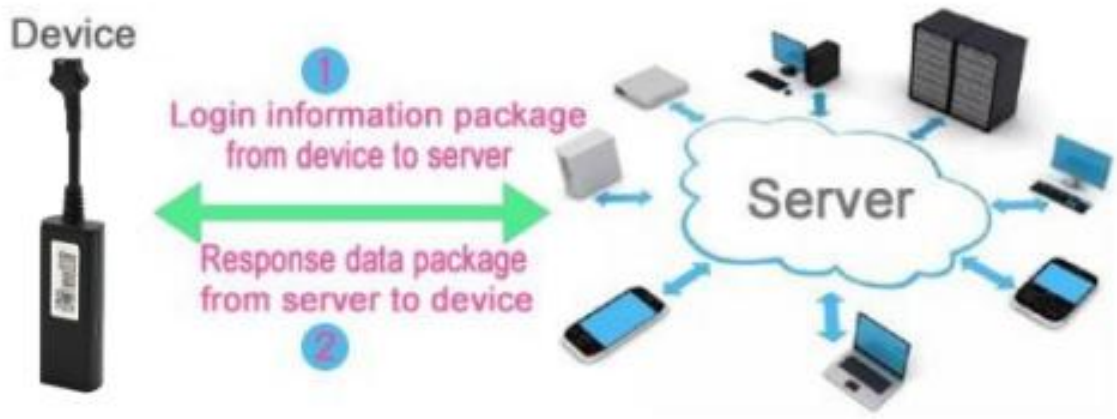
Suitable for Great-Will/Getrak TR05 equipment.

# 2. Terms/Meanings

| Terms/ Abbreviation | Meanings |
|---|---|
| CMPP | China Mobile Peer to Peer |
| GPS | Global Positioning System |
| GSM | Global System for Mobile Communication |
| GPRS | General Packet Radio Service |
| TCP | Transport Control Protocol |
| LBS | Location Based Services |
| IMEI | International Mobile Equipment Identity |
| MCC | Mobile Country Code |
| MNC | Mobile Network Code |
| LAC | Location Area Code |
| Cell ID | Cell Tower ID |
| UDP | User Datagram Protocol |
| SOS | Save Our Ship/Save Our Souls |
| CRC | Cyclic Redundancy Check |
| NITZ | Network Identity and Time Zone, |
| GIS | Geographic Information System |

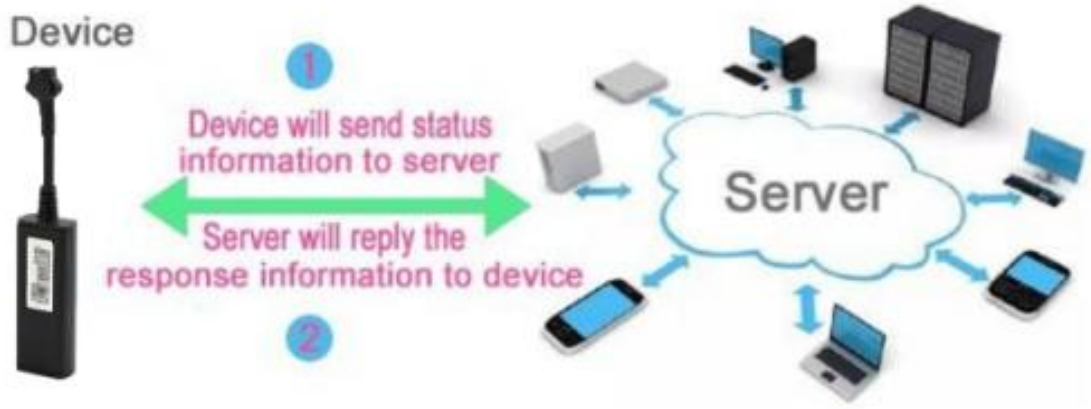
## 3. Basic Rules

3.1. After power on, device will send login information package by default and wait confirmation from the server.




3.2. After the establishment of the normal connection and changing the GPS information, device will send GPS/LBS Merged Information or GPS information package and LBS information package to the server at scheduled time intervals. The server can set default protocol via commands.


3.3. To ensure the effectiveness of the connection, the device will send state information to server during fixed interval and the server will reply the response information package to confirm.



# 4. Data package format

The communication is transferred asynchronously in bytes.

Data package length: (10+N) Byte

| Format | Start bit | Package Length | Protocol number | Information content | Information serial number | error checking | End Bit |
|---|---|---|---|---|---|---|---|
| Length(Byte) | 2 | 1 | 1 | N | 2 | 2 | 2 |

### 4.1. Start Bit

Fixed value, hexadecimal number **0x78 0x78**.

### 4.2. Package Length

Length= protocol number + Information content+ Information serial number + error checking, (5+N) Byte in all, as the information Content is uncertain length data.

### 4.3. Protocol number

Refer to different "information content" and correspond to the protocol number, As table:

| Type | Value |
|---|---|
| Login Information | 0x01 |
| Location Data(OLD) | 0x12 |
| Location Data(NEW) | **0x17** |
| Status Information (The heartbeat packets) | 0x13 |
| Strings Information | 0x15 |
| Alarm data | 0x16 |
| GPS, query address information by phone number | 0x1A |
| LBS、WIFI Merged Data | 0x2C |
| IMSI Information | 0x90 |
| ICCID Information | 0x94 |
| Server send command to device | 0x80 |

### 4.4.Information content

The specific contents are determined by the protocol numbers corresponding to different applications.

### 4.5. Information serial number

After turning on the device, it will send the first item of GPRS data (including heartbeat package and GPS/LBS data package); the serial number of this item is "1". After that, the serial number will be added on by 1 automatically at every sending process (including heartbeat package and GPS/LBS data package).

### 4.6. Error Checking

Device or server can judge the accuracy of data received with identifying code. Sometimes, because of the electronic noise or other interference, data will be changed a little in the transit process. In this case, identifying code can make sure the core or associated core do nothing with such kind of wrong data, which will strengthen the security and efficiency of system. This identifying code adopts CRC-ITU identifying method. The CRC-ITU value is from "Package Length' to "Information Serial Number" in the protocol (including "Package Length" and "Information Serial Number ").

If the receiver receives CRC wrong calculating information, then ignore it and discard this data package.

### 4.7. End bit

Fixed value by hexadecimal **0x0D 0x0A**.

# 5. Data Packet Sent From Device to Server

## 5.1. Login information Packet (0x01)

The login information packet is used to be sent to the server with the device ID so as to confirm the established connection is normal or not.

Content for 8 BYTE, A total of 18 bytes.

#### 5.1.1 Device Sending Login information Packet to Server

| | Description | Byte | Example |
|---|---|---|---|
| Login Message Packet(18 Byte) | Start Bit | 2 | 0x78 0x78 |
| | Length | 1 | 0x0D |
| | Protocol Number | 1 | 0x01 |
| | Device ID | 8 | 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45 |
| | Information Serial Number | 2 | 0x00 0x01 |
| | Error Check | 2 | 0x8C 0xDD |
| | End Bit | 2 | 0x0D 0x0A |

#### 5.1.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.1.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.1.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.1.1.4. Device ID

The Device ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345

The device ID is: **0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45**.

#### 5.1.1.5. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.1.1.6. Error Check

For details see Data Packet Format section 4.6.

#### 5.1.1.7. End Bit

For details see Data Packet Format section 4.7.

### 5.1.2. Server Responds the Login information Packet

| | Description | Byte | Example |
|---|---|---|---|
| Server Response Packet(10 Byte) | Start Bit | 2 | 0x78 0x78 |
| | Length | 1 | 0x05 |
| | Protocol Number | 1 | 0x01 |
| | Information Serial Number | 2 | 0x00 0x01 |
| | Error Check | 2 | 0xD9 0xDC |
| | End Bit | 2 | 0x0D 0x0A |

The response packet from the server to the device: the protocol number in the response packet is identical to the protocol number in the data packet sent by the device.

#### 5.1.2.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.1.2.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.1.2.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.1.2.4. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.1.2.5. Error Check

For details see Data Packet Format section 4.6.

#### 5.1.2.6. End Bit

For details see Data Packet Format section 4.7.

## 5.2. Location data Packet(GPS and LBS) (0x12)

### 5.2.1 Device Sending Location Data Packet to Server

| | Description | | Byte | Example |
|---|---|---|---|---|
| Information Content | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x1F |
| | Protocol Number | | 1 | 0x12 |
| | GPS information | Date time | 6 | 0x13 0x01 0x08 0x09 0x1E 0x0A |
| | | Quantity of GPS information satellites | 1 | 0xCD |
| | | Latitude | 4 | 0x02 0x6B 0x3F 0x3E |
| | | Longitude | 4 | 0x0C 0x45 0x49 0x53 |
| | | Speed | 1 | 0x00 |
| | | Course, Status | 2 | 0x14 0x8F |
| | LBS information | MCC | 2 | 0x01 0xCC |
| | | MNC | 1 | 0x00 |
| | | LAC | 2 | 0x27 0xAC |
| | | Cell ID | 3 | 0x00 0x20 0xC7 |
| | Information Serial Number | | 2 | 0x00 0x02 |
| | Error Check | | 2 | 0xF1 0xFF |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.2.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.2.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.2.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.2.1.4. Date Time

| Format | Length(Byte) | Example |
|---|---|---|
| Year | 1 | 0x13 |
| Month | 1 | 0x01 |
| Day | 1 | 0x08 |
| Hour | 1 | 0x09 |
| Minute | 1 | 0x1E |
| Second | 1 | 0x0A |

Example: 2019-01-08 09:30:10

Calculated as follows: 19(Decimal)=13(Hexadecimal)
01(Decimal)=01(Hexadecimal)
08(Decimal)=08(Hexadecimal)
09(Decimal)=09(Hexadecimal)
30(Decimal)=1E(Hexadecimal)
10(Decimal)=0A(Hexadecimal)

Then the value is: 0x13 0x01 0x08 0x09 0x1E 0x0A

#### 5.2.1.5. Length of GPS information, quantity of positioning satellites

The field is 1 Byte displayed by two hex digits, wherein the first one is for the length of GPS information and the second one for the number of the satellites join in positioning.

Example: if the value is 0xCD, it means the length of GPS information is 12 and the number of the positioning satellites is 13. (C = 12Bit Lenght， D = 13 satellites)

#### 5.2.1.6. Latitude

Four bytes are consumed, defining the latitude value of location data. The range of the value is 0-162000000, indicating a range of 0°-90°. The conversion method thereof is as follow:Converting the value of latitude and longitude output by GPS module into a decimal based on minute;multiplying the converted decimal by 30000; and converting the multiplied result into hexadecimal

Example: 22°32.7658′=(22X60+32.7658)X30000=40582974, then converted into a hexadecimal number. 40582974(Decimal)= 26B3F3E(Hexadecimal)

at last the value is: **0x02 0x6B 0x3F 0x3E**.

#### 5.2.1.7. Longitude

Four bytes are consumed, defining the longitude value of location data. The range of the value is 0-324000000, indicating a range of 0°-180°.

The conversion method herein is same to the method mentioned in Latitude (see section 5.2.1.6).

#### 5.2.1.8. Speed

One byte is consumed, defining the running Speed of GPS. The value ranges from 0x00 to 0xFF. Indicating a range from 0 to 225km/h.

Example: 0x00 represents 0 km/h. 0x13 represents 19km/h.

#### 5.2.1.9. Course & Status

Two bytes are consumed, defining the running direction of GPS. The value ranges from 0° to 360° measured clockwise from north of 0°.

<table>
<tr><td rowspan="8">BYTE1</td><td>Bit7</td><td>0</td></tr>
<tr><td>Bit6</td><td>0(or ACC status)</td></tr>
<tr><td>Bit5</td><td>GPS real-time/differential positioning</td></tr>
<tr><td>Bit4</td><td>GPS having been positioning or not</td></tr>
<tr><td>Bit3</td><td>East Longitude, West Longitude</td></tr>
<tr><td>Bit2</td><td>South Latitude, North Latitude</td></tr>
<tr><td>Bit1</td><td rowspan="16">Course</td></tr>
<tr><td>Bit0</td></tr>
<tr><td rowspan="8">BYTE2</td><td>Bit7</td></tr>
<tr><td>Bit6</td></tr>
<tr><td>Bit5</td></tr>
<tr><td>Bit4</td></tr>
<tr><td>Bit3</td></tr>
<tr><td>Bit2</td></tr>
<tr><td>Bit1</td></tr>
<tr><td>Bit0</td></tr>
</table>

Note: The status information in the data packet is the status corresponding to the time bit recorded in the data packet.

For example: the value is 0x14 0x8F, the corresponding binary is 00010100 10001111,

BYTE_1 Bit7 0

BYTE_1 Bit6 0 (or ACC status)

BYTE_1 Bit5 0 (real time GPS)

BYTE_1 Bit4 1 (GPS has been positioned) (1:Fixed; 0:Not fixed)

BYTE_1 Bit3 0 (East Longitude) (0:East Longitude; 1:West Longitude)

BYTE_1 Bit2 1 (North Latitude)  (0:South Latitude; 1:North Latitude)

BYTE_1 Bit1 0
BYTE_1 Bit0 0
BYTE_2 Bit7 1
BYTE_2 Bit6 0
BYTE_2 Bit5 0
BYTE_2 Bit4 0  } Course 143° (0010001111 in Binary, or 143 in decimal)
BYTE_2 Bit3 1
BYTE_2 Bit2 1
BYTE_2 Bit1 1
BYTE_2 Bit0 1

Which means GPS tracking is on, real time GPS, location at north latitude, east longitude and the course is 143°.

#### 5.2.1.10. MCC

The country code to which a mobile user belongs, i.e., Mobile Country Code(MCC).
Example: Chinese MCC is 460 in decimal, or 0x01 0xCC in Hex (that is, a decimal value of 460 converting into a hexadecimal value, and 0 is added at the left side because the converted hexadecimal value is less than four digits).

Herein the range is 0x0000 ~ 0x03E7.

#### 5.2.1.11. MNC

Mobile Network Code(MNC)
Example: Chinese MNC is 0x00.

#### 5.2.1.12. LAC

Location Area Code (LAC) included in LAI consists of two bytes and is encoded in hexadecimal.The available range is 0x0001-0xFFFE, and the code group 0x0000 and 0xFFFF cannot be used. (See GSM specification 03.03, 04.08 and 11.11).

#### 5.2.1.13. Cell ID

Cell Tower ID (Cell ID), which value ranges from 0x000000 to 0xFFFFFF.

#### 5.2.1.14. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.2.1.15. Error Check

For details see Data Packet Format section 4.6.

#### 5.2.1.16. End Bit

For details see Data Packet Format section 4.7.

### 5.2.2. Server Responds the Location Data Packet

Server does not need to Respond

## 5.3. Location data Packet (0x17)

### 5.3.1 Device Sending Location Data Packet to Server

Send old version location of frame 0*12
Send location with new frame 0*17

| | Description | | Byte | Example |
|---|---|---|---|---|
| Information Content | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x27 |
| | Protocol Number | | 1 | 0x17 |
| | GPS information | Date time | 6 | 0x16 0x01 0x08 0x09 0x1E 0x0A |
| | | Quantity of GPS information satellites | 1 | 0xCD |
| | | Latitude | 4 | 0x02 0x6B 0x3F 0x3E |
| | | Longitude | 4 | 0x0C 0x45 0x49 0x53 |
| | | Speed | 1 | 0x00 |
| | | Course, Status | 2 | 0x14 0x8F |
| | LBS information | MCC | 2 | 0x01 0xCC |
| | | MNC | 1 | 0x00 |
| | | LAC | 2 | 0x27 0xAC |
| | | Cell ID | 3 | 0x00 0x20 0xC7 |
| | Status information | Device Information | 1 | 0x4E |
| | | Battery Voltage Level | 1 | 0x06 |
| | | GSM Signal Strength | 1 | 0x64 |
| | | External Vol | 1 | 0x0F |
| | | Language | 1 | 0x02 |
| | Mileage | | 3 | 0x00 0x13 0x88 |
| | Information Serial Number | | 2 | 0x00 0x02 |
| | Error Check | | 2 | 0xF1 0xFF |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.3.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.3.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.3.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.3.1.4. Date Time

| Format | Length(Byte) | Example |
|---|---|---|
| Year | 1 | 0x13 |
| Month | 1 | 0x01 |
| Day | 1 | 0x08 |
| Hour | 1 | 0x09 |
| Minute | 1 | 0x1E |
| Second | 1 | 0x0A |

Example: 2019-01-08 09:30:10

Calculated as follows: 19(Decimal)=13(Hexadecimal)
01(Decimal)=01(Hexadecimal)
08(Decimal)=08(Hexadecimal)
09(Decimal)=09(Hexadecimal)
30(Decimal)=1E(Hexadecimal)
10(Decimal)=0A(Hexadecimal)

Then the value is: 0x13 0x01 0x08 0x09 0x1E 0x0A

#### 5.3.1.5. Length of GPS information, quantity of positioning satellites

The field is 1 Byte displayed by two hex digits, wherein the first one is for the length of GPS information and the second one for the number of the satellites join in positioning.

Example: if the value is 0xCD, it means the length of GPS information is 12 and the number of the positioning satellites is 13. (C = 12Bit Lenght， D = 13 satellites)

#### 5.3.1.6. Latitude

Four bytes are consumed, defining the latitude value of location data. The range of the value is 0-162000000, indicating a range of 0°-90°. The conversion method thereof is as follow:Converting the value of latitude and longitude output by GPS module into a decimal based on minute;multiplying the converted decimal by 30000; and converting the multiplied result into hexadecimal

Example: 22°32.7658′=(22X60+32.7658)X30000=40582974, then converted into a hexadecimal number. 40582974(Decimal)= 26B3F3E(Hexadecimal)

at last the value is: **0x02 0x6B 0x3F 0x3E**.

#### 5.3.1.7. Longitude

Four bytes are consumed, defining the longitude value of location data. The range of the value is 0-324000000, indicating a range of 0°-180°.

The conversion method herein is same to the method mentioned in Latitude (see section 5.2.1.6).

#### 5.3.1.8. Speed

One byte is consumed, defining the running Speed of GPS. The value ranges from 0x00 to 0xFF. Indicating a range from 0 to 225km/h.

Example: 0x00 represents 0 km/h. 0x13 represents 19km/h.

#### 5.3.1.9. Course & Status

Two bytes are consumed, defining the running direction of GPS. The value ranges from 0° to 360° measured clockwise from north of 0°.

| | | |
|---|---|---|
| BYTE1 | Bit7 | 0 |
| | Bit6 | 0(or ACC status) |
| | Bit5 | GPS real-time/differential positioning |
| | Bit4 | GPS having been positioning or not |
| | Bit3 | East Longitude, West Longitude |
| | Bit2 | South Latitude, North Latitude |
| | Bit1 | Course |
| | Bit0 | |
| BYTE2 | Bit7 | |
| | Bit6 | |
| | Bit5 | |
| | Bit4 | |
| | Bit3 | |
| | Bit2 | |
| | Bit1 | |
| | Bit0 | |

Note: The status information in the data packet is the status corresponding to the time bit recorded in the data packet.

For example: the value is 0x14 0x8F, the corresponding binary is 00010100 10001111,

BYTE_1 Bit7 0

BYTE_1 Bit6 0 (or ACC status)

BYTE_1 Bit5 0 (real time GPS)

BYTE_1 Bit4 1 (GPS has been positioned) (1:Fixed; 0:Not fixed)

BYTE_1 Bit3 0 (East Longitude) (0:East Longitude; 1:West Longitude)

BYTE_1 Bit2 1 (North Latitude) (0:South Latitude; 1:North Latitude)

BYTE_1 Bit1 0

BYTE_1 Bit0 0

BYTE_2 Bit7 1

BYTE_2 Bit6 0

BYTE_2 Bit5 0

BYTE_2 Bit4 0 } Course 143° (0010001111 in Binary, or 143 in decimal)

BYTE_2 Bit3 1

BYTE_2 Bit2 1

BYTE_2 Bit1 1

BYTE_2 Bit0 1

Which means GPS tracking is on, real time GPS, location at north latitude, east longitude and the course is 143°.

#### 5.3.1.10. MCC

The country code to which a mobile user belongs, i.e., Mobile Country Code(MCC).

Example: Chinese MCC is 460 in decimal, or 0x01 0xCC in Hex (that is, a decimal value of 460 converting into a hexadecimal value, and 0 is added at the left side because the converted hexadecimal value is less than four digits).

Herein the range is 0x0000 ~ 0x03E7.

#### 5.3.1.11. MNC

Mobile Network Code(MNC)

Example: Chinese MNC is 0x00.

#### 5.3.1.12. LAC

Location Area Code (LAC) included in LAI consists of two bytes and is encoded in hexadecimal.The available range is 0x0001-0xFFFE, and the code group 0x0000 and 0xFFFF cannot be used. (See GSM specification 03.03, 04.08 and 11.11).

#### 5.3.1.13. Cell ID

Cell Tower ID (Cell ID), which value ranges from 0x000000 to 0xFFFFFF.

### 5.3.1.14. Device Information

Occupy 1 byte，representing each information of the device。Regard 1 byte as 8bits, the lowest bit is 0, the highest is 7. In the process of the data transmitting,the high one comes first and the low one follows. Each bit represents the detailed meaning as follows:

High bit low bit

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

| | |
|---|---|
| 0 | 0：Not fortified<br>1：Fortified |
| 1 | 0：ACC Off<br>1：ACC On |
| 2 | 0：Not charged<br>1：Charged |
| 3、4、5 | 000：Normal 001：<br>Vibration alarm 010：<br>Cut-off alarm 011：<br>Low-power alarm<br>100：SOS alarm |
| 6 | 0：GPS not Fixed<br>1：GPS Fixed |
| 7 | 0：Petrol(Oil)/Electricity on<br>1：Petrol(Oil)/Electricity off |

**Note:** The status information refers to the status in a certain time

For example: **0x4B** converts to binary 01001011, which means fortified/high ACC/not charged/vibration alarm/GPS Fixed/petrol/electricity on.

### 5.3.1.15. Battery Voltage Level

The range is 0~6 defining the Battery voltage is from low to high.

0：Lowest power and power off

1：No enough power to dial a call or send messages.

2：Low power and alarm

3：Lower power but can work normally

3~6：Work in good condition

### 5.3.1.16.GSM Signal degree

The GSM information range: 0 ~ 100; The stronger the number, the greater the GSM signal
0: no signal
100: signal is full

### 5.3.1.17. External Voltage

For example：External Voltage 30V，As：0x1E

For example：External Voltage 12V，As：0x0C

#### 5.3.1.18. Language

Chinese：0x01　　English：0x02

#### 5.3.1.19. Mileage

Car mileage, If the mileage value is 9000 km, the display data:0x00 0x23 0x28
Mileage range:0~990000;

#### 5.3.1.20. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.3.1.21. Error Check

For details see Data Packet Format section 4.6.

#### 5.3.1.22. End Bit

For details see Data Packet Format section 4.7.

### 5.3.2. Server Responds the Location Data Packet

Server does not need to Respond

## 5.4. Status information Packet (Heartbeat Packet)(0x13)

Status information packet is a data packet to maintain the connection between the device and the server.

### 5.4.1 Device Sending Status information Packet to Server

| | Description | | Length(Byte) | Example |
|---|---|---|---|---|
| Information Content | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x0A |
| | Protocol Number | | 1 | 0x13 |
| | Status information | Device Information | 1 | 0x4E |
| | | Battery Voltage Level | 1 | 0x06 |
| | | GSM Signal Strength | 1 | 0x64 |
| | | External Voltage | 1 | 0x0C |
| | | Language | 1 | 0x02 |
| | Information Serial Number | | 2 | 0x00 0x03 |
| | Error Check | | 2 | 0xFF 0x4F |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.4.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.4.1.2. Packet Length

For details see Data Packet Format section 4.2.

### 5.4.1.3. Protocol Number

For details see Data Packet Format section 4.3.

### 5.4.1.4. Device Information

Occupy 1 byte，representing each information of the device。Regard 1 byte as 8bits, the lowest bit is 0, the highest is 7. In the process of the data transmitting,the high one comes first and the low one follows. Each bit represents the detailed meaning as follows:

High bit low bit

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

| | |
|---|---|
| 0 | 0：Not fortified<br>1：Fortified |
| 1 | 0：ACC Off<br>1：ACC On |
| 2 | 0：Not charged<br>1：Charged |
| 3、4、5 | 000：Normal 001：<br>Vibration alarm 010：<br>Cut-off alarm 011：<br>Low-power alarm<br>100：SOS alarm |
| 6 | 0：GPS not Fixed<br>1：GPS Fixed |
| 7 | 0：Petrol(Oil)/Electricity on<br>1：Petrol(Oil)/Electricity off |

**Note:** The status information refers to the status in a certain time

For example: **0x4B** converts to binary 01001011, which means fortified/high ACC/not charged/vibration alarm/GPS Fixed/petrol/electricity on.

### 5.4.1.5. Battery Voltage Level

The range is 0~6 defining the Battery voltage is from low to high.

0：Lowest power and power off

1：No enough power to dial a call or send messages.

2：Low power and alarm

3：Lower power but can work normally

3~6：Work in good condition

### 5.4.1.6.GSM Signal degree

The GSM information range: 0 ~ 100; The stronger the number, the greater the GSM signal
0: no signal
100: signal is full

#### 5.4.1.7. External Voltage

For example：External Voltage 30V，As：0x1E

For example：External Voltage 12V，As：0x0C

#### 5.4.1.8. Language

Chinese：0x01　　English：0x02

#### 5.4.1.9. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.4.1.10. Error Check

For details see Data Packet Format section 4.6.

#### 5.4.1.11. End Bit

For details see Data Packet Format section 4.7.

### 5.4.2. Server Responds the Status information Packet

| | Description | Bits | Example |
|---|---|---|---|
| Information Content | Start Bit | 2 | 0x78 0x78 |
| | Length | 1 | 0x05 |
| | Protocol Number | 1 | 0x13 |
| | Information Serial Number | 2 | 0x00 0x01 |
| | Error Check | 2 | 0xE9 0xF1 |
| | End Bit | 2 | 0x0D 0x0A |

The response packet from the server to the device: the protocol number in the response packet is identical to the protocol number in the data packet sent by the device.

#### 5.4.2.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.4.2.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.4.2.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.4.2.4. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.4.2.5. Error Check

For details see Data Packet Format section 4.6.

#### 5.4.2.6. End Bit

For details see Data Packet Format section 4.7.

## 5.5. Alarm Packet (0x16)

### 5.5.1 Device Sending Alarm Packet to Server

| | Description | | Byte | Example |
|---|---|---|---|---|
| Information Content | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x25 |
| | Protocol Number | | 1 | 0x16 |
| | GPS information | Date time | 6 | 0x13 0x01 0x08 0x09 0x1E 0x0A |
| | | Quantity of GPS Information satellites | 1 | 0xCD |
| | | Latitude | 4 | 0x02 0x6B 0x3F 0x3E |
| | | Longitude | 4 | 0x0C 0x45 0x49 0x53 |
| | | Speed | 1 | 0x00 |
| | | Course,Status | 2 | 0x14 0x8F |
| | LBS information | LBS Length | 1 | 0x09 |
| | | MCC | 2 | 0x01 0xCC |
| | | MNC | 1 | 0x00 |
| | | LAC | 2 | 0x27 0xAC |
| | | Cell ID | 3 | 0x00 0x20 0xC7 |
| | Status information | Device Information Content | 1 | 0x65 |
| | | Battery Voltage Level | 1 | 0x06 |
| | | GSM Signal Strength | 1 | 0x64 |
| | | Alarm Type | 1 | 0x01 |
| | | Language | 1 | 0x02 |
| | Information Serial Number | | 2 | 0x00 0x03 |
| | Error Check | | 2 | 0xCA 0xA3 |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.5.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.5.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.5.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.5.1.4. Date Time

For details see Location Data Packet Format section 5.2.1.4.

#### 5.5.1.5. Quantity of positioning satellites

For details see Location Data Packet Format section 5.2.1.5.

#### 5.5.1.6. Latitude

For details see Location Data Packet Format section 5.2.1.6.

#### 5.5.1.7. Longitude

For details see Location Data Packet Format section 5.2.1.7.

#### 5.5.1.8. Speed

For details see Location Data Packet Format section 5.2.1.8.

#### 5.5.1.9. Course & Status

For details see Location Data Packet Format section 5.2.1.9.

#### 5.5.1.10. Lbs Length

Fixed value: 0x09

#### 5.5.1.11. MCC

For details see Location Data Packet Format section 5.2.1.10.

#### 5.5.1.12. MNC

For details see Location Data Packet Format section 5.2.1.11.

#### 5.5.1.13. LAC

For details see Location Data Packet Format section 5.2.1.12.

#### 5.5.1.14. Cell ID

For details see Location Data Packet Format section 5.2.1.13.

#### 5.5.1.15. Device Information Content

For details see Location Data Packet Format section 5.3.1.4.

#### 5.5.1.16. Battery Voltage Level

For details see Location Data Packet Format section 5.3.1.5.

#### 5.5.1.17. GSM Signal Strength

For details see Location Data Packet Format section 5.3.1.6.

#### 5.5.1.18. Alarm Type

| | |
|---|---|
| Alarm | 0x00:Normal |
| | 0x01: SOS |
| | 0x02: Power Cut Alarm |
| | 0x03: Shock Alarm |
| | 0x04: Fence In Alarm |
| | 0x05: Fence Out Alarm |
| | 0x06:Speed Alarm |
| | 0x09:Move Alarm |
| | 0x0E:LowBattery Alarm |

| | |
|---|---|
| | 0x13: Disassemble Alarm |
| | 0xFE:ACC On Alarm |
| | 0xFF:ACC Off Alarm |

#### 5.5.1.19. Language

For details see Location Data Packet Format section 5.3.1.7.

#### 5.5.1.20. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.5.1.21. Error Check

For details see Data Packet Format section 4.6.

#### 5.5.1.22. End Bit

For details see Data Packet Format section 4.7.

### 5.5.2. Server Responds Alarm Data Packet to Device

| | Description | Length | Example |
|---|---|---|---|
| Information Content | Start Bit | 2 | 0x78 0x78 |
| | Packet Length | 1 | 0x05 |
| | Protocol Number | 1 | 0x16 |
| | Serial Number | 2 | 0x00 0x05 |
| | Error Check | 2 | 0x96 0x68 |
| | End Bit | 2 | 0x0D 0x0A |

#### 5.5.2.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.5.2.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.5.2.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.5.2.4. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.5.2.5. Error Check

For details see Data Packet Format section 4.6.

#### 5.5.2.6. End Bit

For details see Data Packet Format section 4.7.

## 5.6. Address Packet (0x1A)

### 5.6.1 Device Sending Address Packet to Server

| | Description | | Byte | Example |
|---|---|---|---|---|
| Information Content | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x2E |
| | Protocol Number | | 1 | 0x1A |
| | GPS information | Date time | 6 | 0x13 0x01 0x08 0x09 0x1E 0x0A |
| | | Quantity of GPS Information satellites | 1 | 0xCD |
| | | Latitude | 4 | 0x02 0x6B 0x3F 0x3E |
| | | Longitude | 4 | 0x0C 0x45 0x49 0x53 |
| | | Speed | 1 | 0x00 |
| | | Course,Status | 2 | 0x14 0x8F |
| | Phone Number | The phone number | 21 | 0x31 0x33 0x38 0x30 0x30 0x31 0x39 0x39 0x38 0x38 0x35 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 |
| | Status | The reserved | 1 | 0x00 |
| | | Language | 1 | 0x02 |
| | Information Serial Number | | 2 | 0x00 0x06 |
| | Error Check | | 2 | 0x29 0xD1 |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.6.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.6.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.6.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.6.1.4. Date Time

For details see Location Data Packet Format section 5.2.1.4.

#### 5.6.1.5. Quantity of positioning satellites

For details see Location Data Packet Format section 5.2.1.5.

#### 5.6.1.6. Latitude

For details see Location Data Packet Format section 5.2.1.6.

#### 5.6.1.7. Longitude

For details see Location Data Packet Format section 5.2.1.7.

#### 5.6.1.8. Speed

For details see Location Data Packet Format section 5.2.1.8.

#### 5.6.1.9. Course & Status

For details see Location Data Packet Format section 5.2.1.9.

#### 5.6.1.10. Phone Number

Fixed 21 byte, If the phone number less than 21 num, added by Spaces: 0x20
Example:13800199885,Device upload:0x31 0x33 0x38 0x30 0x30 0x31 0x39 0x39 0x38 0x38 0x35 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20 0x20

#### 5.6.1.11. The reserved

Currently unused, fixed 0x00

#### 5.6.1.12. Language

Chinese: 0x01

English: 0x02

#### 5.6.1.13. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.6.1.14. Error Check

For details see Data Packet Format section 4.6.

#### 5.6.1.15. End Bit

For details see Data Packet Format section 4.7.

### 5.6.2. Server Responds Address Packet to Device

#### 5.6.2.1. Responds English

```
7878                                    //Start bts
00 9B                                   //Length(2bytes)
97                                      //Protocol Number
00 96                                   //Content length
00000001                                //Sever flag
41444452455353                          //ADDRESS(Fixed char)
2626                                    //&& (Fixed char)
//Address Content（UNICODE）
004C006F00630061007400690006F006E003A
00310039002000440061007900610E002000520064002C
00590075006C007500200043006F006D006D0075006E006900740079002C
005300680065006E007A00680065006E002C
004700750061006E00670064006F006E0067
2626                                    //&& (Fixed char)
313338303031393938383520202020202020202020     //PhoneNumber,13800199885
2323                                    //##  (End char)
0106                                    //Serial number
8C56                                    //Error check
0D0A                                    //End bit
```

#### 5.6.2.2. Responds Chinese

```
7878                                    //Start bts
52                                      //Length(1byte)
17                                      //Protocol Number
4E                                      //Content length(1 个字节)
00000001                                //Server flag
41444452455353                          //ADDRESS(Fixed char)
2626                                    //&&
//Address contents（UNICODE）
4F4D7F6E003A                            //位置：
5E7F4E1C77016DF157335E02                //广东省深圳市
73895F8B793E533A                        //玉律社区
59276D0B4E008DEF                        //大洋一路
0031003953F7                            //19 号
2626                                    //&&
313338303031333830303020202020202020202020      //PhoneNubmer,13800199885
2323                                    //##
0106                                    // Serial number
2B24                                    //Error check
0D0A                                    //End bit
```

Server Responds Contents：位置：广东省深圳市玉律社区大洋一路 19 号

## 5.7. Wifi data Packet (0x2C)

### 5.7.1 Device Sending Data Packet to Server

| | Description | | Byte | Example |
|---|---|---|---|---|
| | Start Bit | | 2 | 0x78 0x78 |
| | Packet Length | | 1 | 0x66 |
| | Protocol Number | | 1 | 0x2C |
| | Date time | | 6 | 0x13 0x01 0x08 0x09 0x1E 0x0A |
| | LBS information | MCC | 2 | 0x01 0xCC |
| | | MNC | 1 | 0x00 |
| | | LAC | 2 | 0x27 0xAC |
| | | Cell ID | 3 | 0x00 0x20 0xC7 |
| | | Rxlev | 1 | 0x38 |
| | | LAC2 | 2 | 0x21 0xEB |
| | | Cell ID2 | 3 | 0x00 0x19 0xF7 |
| | | Rxlev2 | 1 | 0x31 |
| | | LAC3 | 2 | 0x29 0xC3 |
| | | Cell ID3 | 3 | 0x00 0x11 0xE6 |
| | | Rxlev3 | 1 | 0x33 |

| | | | | |
|---|---|---|---|---|
| Information Content | | LAC4 | 2 | 0x21 0xA5 |
| | | Cell ID4 | 3 | 0x00 0x20 0xA7 |
| | | Rxlev4 | 1 | 0x28 |
| | | LAC5 | 2 | 0x1e 0xF9 |
| | | Cell ID5 | 3 | 0x00 0x1A 0x36 |
| | | Rxlev5 | 1 | 0x23 |
| | | LAC6 | 2 | 0x00 0x00 |
| | | Cell ID6 | 3 | 0x00 0x00 0x00 |
| | | Rxlev6 | 1 | 0x00 |
| | | LAC7 | 2 | 0x00 0x00 |
| | | Cell ID7 | 3 | 0x00 0x00 0x00 |
| | | Rxlev7 | 1 | 0x00 |
| | | TA | 1 | 0x00 |
| | Wifi information | Wifi Number | 1 | 0x06 |
| | | MAC | 6 | 0xE4 0xF4 0xC6 0x00 0x2A 0x61 |
| | | Rxlev | 1 | 0x49 |
| | | MAC2 | 6 | 0x06 0x11 0xB5 0x26 0x2C 0x71 |
| | | Rxlev2 | 1 | 0x38 |
| | | ..... | ..... | ..... |
| | Serial Number | | 2 | 0x00 0x09 |
| | Error Check | | 2 | 0x3C 0x5F |
| | End Bit | | 2 | 0x0D 0x0A |

#### 5.7.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.7.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.7.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.7.1.4. Date Time

For details see Data Packet Format section 5.2.1.4

#### 5.7.1.5. MCC

For details see Data Packet Format section 5.2.1.10

#### 5.7.1.6. MNC

For details see Data Packet Format section 5.2.1.11

#### 5.7.1.7. LAC

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.8. Cell ID

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.9. Rxlev

The base station signal strength.

Rang:0~100

#### 5.7.1.10. LAC2

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.11. Cell ID2

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.12. Rxlev2

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.13. LAC3

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.14. Cell ID3

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.15. Rxlev3

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.16. LAC4

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.17. Cell ID4

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.18. Rxlev4

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.19. LAC5

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.20. Cell ID5

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.21. Rxlev5

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.22. LAC6

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.23. Cell ID6

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.24. Rxlev6

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.25. LAC7

For details see Data Packet Format section 5.2.1.12

#### 5.7.1.26. Cell ID7

For details see Data Packet Format section 5.2.1.13

#### 5.7.1.27. Rxlev7

For details see Data Packet Format section 5.6.1.9

#### 5.7.1.28. TA

The GSM signal of schedule.Current use 0x00;

#### 5.7.1.29. Wifi Number

The number of WIFI in the search. Not more than eight.

#### 5.7.1.30. MAC

The current WIFI MAC address.

Example: MAC is 06:11:B5:26:2C:71 Upload: 0x06 0x11 0xB5 0x26 0x2C 0x71

#### 5.7.1.31. Rxlev

The current WIFI signal strength

Rang:0~100

#### 5.7.1.32. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.7.1.33. Error Check

For details see Data Packet Format section 4.6.

#### 5.7.1.34. End Bit

For details see Data Packet Format section 4.7.

### 5.7.2. Server Responds Data Packet

Server does not need to Respond

## 5.8. IMSI Packet (0x090)

### 5.8.1 Device Sending Imsi information Packet to Server

| | Description | Byte | Example |
|---|---|---|---|
| Information Content | Start Bit | 2 | 0x78 0x78 |
| | Length | 1 | 0x10 |
| | Protocol Number | 1 | 0x90 |
| | IMSI:( CHAR) | 5 | 0x49 0x4D 0x53 0x49 0x3A |
| | IMS Content | 15 | 0x31 0x33 0x33 0x34 0x35 0x36 0x37 0x38 0x39 0x30 0x31 0x32 0x33 0x34 0x35 |
| | CHAR | 3 | 0x01 0x01 0x9F |
| | Information Serial Number | 2 | 0x00 0x09 |
| | Error Check | 2 | 0xCC 0x51 |
| | End Bit | 2 | 0x0D 0x0A |

#### 5.8.1.1. Start Bit

For details see Data Packet Format section 4.1.

#### 5.8.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.8.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.8.1.4. IMSI:(CHAR)

Fixed Value: 0x49 0x4D 0x53 0x49 0x3A (IMSI:)

#### 5.8.1.5. IMSI CONTENT

SimCard IMSI Number,15 bytes

Example: if the IMSI is 123456789012345

The Imsi Contents is: 0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38 0x39 0x30 0x31 0x32 0x33 0x34 0x35.

#### 5.8.1.6. CHAR

Fixed Value: 0x01 0x01 0x9F

#### 5.8.1.7. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.8.1.8. Error Check

For details see Data Packet Format section 4.6.

#### 5.8.1.9. End Bit

For details see Data Packet Format section 4.7.

### 5.8.2. Server Responds Data Packet

Server does not need to Respond

## 5.9. ICCID Packet (0x094)

### 5.9.1 Device Sending ICCID information Packet to Server

| | Description | Byte | Example |
|---|---|---|---|
| Information Content | Start Bit | 2 | 0x79 0x79 |
| | Length | 2 | 0x00 0x20 |
| | Protocol Number | 1 | 0x94 |
| | FLAG | 1 | 0x0A |
| | IMEI | 8 | 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45 |
| | IMSI | 8 | 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45 |
| | ICCID | 10 | 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45 0x67 0x89 |
| | Serial Number | 2 | 0x00 0x0A |
| | Error Check | 2 | 0x81 0xF0 |
| | End Bit | 2 | 0x0D 0x0A |

#### 5.9.1.1. Start Bit

Fixed value，hexadecimal number **0x79 0x79**

#### 5.9.1.2. Packet Length

For details see Data Packet Format section 4.2.

#### 5.9.1.3. Protocol Number

For details see Data Packet Format section 4.3.

#### 5.9.1.4. Flag

Fixed Value: 0x0A

#### 5.9.1.5. IMEI

IMEI Number,15 bytes

Example: if the IMEI is 123456789012345

The Imei Contents is: 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45

#### 5.9.1.6. IMSI

SimCard IMSI Number,15 bytes

Example: if the IMSI is 123456789012345

The Imsi Contents is: 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45

#### 5.9.1.7. ICCID

SimCard ICCID Number,20 bytes

Example: if the ICCID is 01234567890123456789

The ICCID Contents is: 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45 0x67 0x89

#### 5.9.1.8. Information Serial Number

For details see Data Packet Format section 4.5.

#### 5.9.1.9. Error Check

For details see Data Packet Format section 4.6.

#### 5.9.1.10. End Bit

For details see Data Packet Format section 4.7.

### 5.9.2. Server Responds Data Packet

Server does not need to Respond

# 6. Data Packet Sent From Server to Device

## 6.1. Packet Sent by Server(0x80)

| Description | | Length(Byte) |
|---|---|---|
| Information Content | Start Bit | 2 |
| | Picket Length | 1 |
| | Protocol Number | 1 |
| | Length of Command | 1 |
| | Server Flag Bit | 4 |
| | Command Content | M |
| | Serial Number | 2 |
| | Error Check | 2 |
| | End Bit | 2 |

(The sample, please refer to Appendix A or B)

### 6.1.1. Start Bit

For details see Data Packet Format section 4.1.

### 6.1.2. Packet Length

For details see Data Packet Format section 4.2.

### 6.1.3. Protocol Number

The Protocol Number of device transmission is 0x80.

### 6.1.4. Length of Command

Server Flag Bit + Length of Command Content

Example: measured in bytes, 0x0A means the content of command occupied ten bytes.

### 6.1.5. Server Flag Bit

It is reserved to the identification of the server. The binary data received by the device is returned without change.

### 6.1.6. Command Content

It is represented in ASC II of string, and the command content is compatible with text message command.

### 6.1.7. Serial Number

For details see Data Packet Format section 4.5.

### 6.1.8. Error Check

For details see Data Packet Format section 4.6.

### 6.1.9. End Bit

For details see Data Packet Format section 4.7.

## 6.2. Packet Replied by Device (0x15)

| Description | | Length(Byte) |
|---|---|---|
| Information Content | Start Bit | 2 |
| | Picket Length | 1 |
| | Protocol Number | 1 |
| | Length of Command | 1 |
| | Server Flag Bit | 4 |
| | Command Content | M |
| | Reserved | 2 |
| | Serial Number | 2 |
| | Error Check | 2 |
| | End Bit | 2 |

(The sample, please refer to Appendix A or B)

### 6.2.1. Start Bit

For details see Data Packet Format section 4.1.

### 6.2.2. Packet Length

For details see Data Packet Format section 4.2.

### 6.2.3. Protocol Number

The device responds to the command sent by the server. The format of data packet is consistent with "the command sent by the server to the device", but the Protocol Number herein is different and is 0x15.

### 6.2.4. Length of Command

Server Flag Bit + Length of Command Content

Example: measured in bytes, 0x0A means the content of command occupied ten bytes.

### 6.2.5. Server Flag Bit

It is reserved to the identification of the server. The binary data received by the device is returned without change.

### 6.2.6. Command Content

It is represented in ASC II of string, and the command content is compatible with text message command.

### 6.2.7. Reserved

Not currently used, a fixed value: 0x00 0x01

### 6.2.8. Serial Number

For details see Data Packet Format section 4.5.

### 6.2.9. Error Check

For details see Data Packet Format section 4.6.

### 6.2.10. End Bit

For details see Data Packet Format section 4.7.

## 7. CRC-ITU lookup table algorithm Code

```
static const U16 crctab16[ ] =
{
    0X0000, 0X1189, 0X2312, 0X329B, 0X4624, 0X57AD, 0X6536, 0X74BF,
    0X8C48, 0X9DC1, 0XAF5A, 0XBED3, 0XCA6C, 0XDBE5, 0XE97E, 0XF8F7,
    0X1081, 0X0108, 0X3393, 0X221A, 0X56A5, 0X472C, 0X75B7, 0X643E,
    0X9CC9, 0X8D40, 0XBFDB, 0XAE52, 0XDAED, 0XCB64, 0XF9FF, 0XE876,
    0X2102, 0X308B, 0X0210, 0X1399, 0X6726, 0X76AF, 0X4434, 0X55BD,
    0XAD4A, 0XBCC3, 0X8E58, 0X9FD1, 0XEB6E, 0XFAE7, 0XC87C, 0XD9F5,
    0X3183, 0X200A, 0X1291, 0X0318, 0X77A7, 0X662E, 0X54B5, 0X453C,
    0XBDCB, 0XAC42, 0X9ED9, 0X8F50, 0XFBEF, 0XEA66, 0XD8FD, 0XC974,
    0X4204, 0X538D, 0X6116, 0X709F, 0X0420, 0X15A9, 0X2732, 0X36BB,
    0XCE4C, 0XDFC5, 0XED5E, 0XFCD7, 0X8868, 0X99E1, 0XAB7A, 0XBAF3,
    0X5285, 0X430C, 0X7197, 0X601E, 0X14A1, 0X0528, 0X37B3, 0X263A,
    0XDECD, 0XCF44, 0XFDDF, 0XEC56, 0X98E9, 0X8960, 0XBBFB, 0XAA72,
    0X6306, 0X728F, 0X4014, 0X519D, 0X2522, 0X34AB, 0X0630, 0X17B9,
    0XEF4E, 0XFEC7, 0XCC5C, 0XDDD5, 0XA96A, 0XB8E3, 0X8A78, 0X9BF1,
    0X7387, 0X620E, 0X5095, 0X411C, 0X35A3, 0X242A, 0X16B1, 0X0738,
    0XFFCF, 0XEE46, 0XDCDD, 0XCD54, 0XB9EB, 0XA862, 0X9AF9, 0X8B70,
    0X8408, 0X9581, 0XA71A, 0XB693, 0XC22C, 0XD3A5, 0XE13E, 0XF0B7,
    0X0840, 0X19C9, 0X2B52, 0X3ADB, 0X4E64, 0X5FED, 0X6D76, 0X7CFF,
    0X9489, 0X8500, 0XB79B, 0XA612, 0XD2AD, 0XC324, 0XF1BF, 0XE036,
    0X18C1, 0X0948, 0X3BD3, 0X2A5A, 0X5EE5, 0X4F6C, 0X7DF7, 0X6C7E,
    0XA50A, 0XB483, 0X8618, 0X9791, 0XE32E, 0XF2A7, 0XC03C, 0XD1B5,
    0X2942, 0X38CB, 0X0A50, 0X1BD9, 0X6F66, 0X7EEF, 0X4C74, 0X5DFD,
    0XB58B, 0XA402, 0X9699, 0X8710, 0XF3AF, 0XE226, 0XD0BD, 0XC134,
    0X39C3, 0X284A, 0X1AD1, 0X0B58, 0X7FE7, 0X6E6E, 0X5CF5, 0X4D7C,
    0XC60C, 0XD785, 0XE51E, 0XF497, 0X8028, 0X91A1, 0XA33A, 0XB2B3,
    0X4A44, 0X5BCD, 0X6956, 0X78DF, 0X0C60, 0X1DE9, 0X2F72, 0X3EFB,
    0XD68D, 0XC704, 0XF59F, 0XE416, 0X90A9, 0X8120, 0XB3BB, 0XA232,
    0X5AC5, 0X4B4C, 0X79D7, 0X685E, 0X1CE1, 0X0D68, 0X3FF3, 0X2E7A,
    0XE70E, 0XF687, 0XC41C, 0XD595, 0XA12A, 0XB0A3, 0X8238, 0X93B1,
    0X6B46, 0X7ACF, 0X4854, 0X59DD, 0X2D62, 0X3CEB, 0X0E70, 0X1FF9,
    0XF78F, 0XE606, 0XD49D, 0XC514, 0XB1AB, 0XA022, 0X92B9, 0X8330,
    0X7BC7, 0X6A4E, 0X58D5, 0X495C, 0X3DE3, 0X2C6A, 0X1EF1, 0X0F78,
};

// Calculate the 16-bit CRC of data with predetermined length.
U16 GetCrc16(const U8* pData, int nLength)
{
    U16 fcs = 0xffff;      // initialization

    while(nLength>0)
    {
        fcs = (fcs >> 8) ^ crctab16[(fcs ^ *pData) & 0xff];
        nLength--;
        pData++;
    }
    return ~fcs; // negated
}
```

# 8.Appendix A: a fragment of example of data packet of communication protocol

The following data displayed in hexadecimal are intercepted from the communication between a device and a server, wherein transmission means sending by the device and reception means returned from the server:

**Login packet:**

Transmission: 78 78 0D 01 03 53 41 35 32 15 03 62 00 02 2D 06 0D 0A

Reception: 78 78 05 01 00 02 EB 47 0D 0A

**GPS data packet (GPS and LBS):**

Transmission: 78 78 1F 12 0B 08 1D 11 2E 10 CF 02 7A C7 EB 0C 46 58 49 00 14 8F 01 CC 00 28 7D 00 1F B8 00 03 80 81 0D 0A

**Status packet(Heartbeat packet):**

Transmission: 78 78 0A 13 44 01 04 00 01 00 05 08 45 0D 0A

Reception: 78 78 05 13 00 05 AF D5 0D 0A

**Disconnect oil and electricity online:**

Reception: 78 78 15 80 0F 00 01 A9 58 44 59 44 2C 30 30 30 30 30 30 23 00 A0 DC F1 0D 0A

Transmission: 78 78 18 15 10 00 01 A9 58 44 59 44 3D 53 75 63 63 65 73 73 21 00 02 00 18 91 77 0D 0A

The server sending: DYD,000000#

Reply: DYD=Success!

**Command sent during disconnection of oil and electricity:**

Reception: 78 78 15 80 0F 00 01 A9 61 44 59 44 2C 30 30 30 30 30 30 23 00 A0 3E 10 0D 0A

Transmission: 78 78 53 15 4B 00 01 A9 61 41 6C 72 65 61 64 79 20 69 6E 20 74 68 65 20 73 74 61 74 65 20 6F 66 20 66 75 65 6C 20 73 75 70 70 6C 79 20 63 75 74 20 6F 66 66 2C 74 68 65 20 63 6F 6D 6D 61 6E 64 20 69 73 20 6E 6F 74 20 72 75 6E 6E 69 6E 67 21 00 02 00 1C F3 0D 0D 0A

The server sending: DYD,000000#

Reply: Already in the state of fuel supply cut off, the command is not running!

**Connect oil and electricity online:**

Reception: 78 78 16 80 10 00 01 A9 63 48 46 59 44 2C 30 30 30 30 30 30 23 00 A0 7B DC 0D 0A

Transmission: 78 78 19 15 11 00 01 A9 63 48 46 59 44 3D 53 75 63 63 65 73 73 21 00 02 00 1E F8 93 0D 0A

The server sending: HFYD,000000#

Reply: HFYD=Success!

**Command sent during connection of oil and electricity:**

Reception: 78 78 16 80 10 00 01 A9 64 48 46 59 44 2C 30 30 30 30 30 30 23 00 A0 8B 1B 0D 0A

Transmission: 78 78 55 15 4D 00 01 A9 64 41 6C 72 65 61 64 79 20 69 6E 20 74 68 65 20 73 74 61 74 65 20 6F 66 20 66 75 65 6C 20 73 75 70 70 6C 79 20 74 6F 20 72 65 73 75 6D 65 2C 74 68 65 20 63 6F 6D 6D 61 6E 64 20 69 73 20 6E 6F 74 20 72 75 6E 6E 69 6E 67 21 00 02 00 1F DB BF 0D 0A

the server sending: HFYD,000000#

reply: Already in the state of fuel supply to resume, the command is not running!

**Querying address information online:**

Reception: 78 78 16 80 10 00 01 A9 67 44 57 58 58 2C 30 30 30 30 30 30 23 00 A0 06 2D 0D 0A

Transmission: 78 78 64 15 5C 00 01 A9 67 44 57 58 58 3D 4C 61 74 3A 4E 32 33 2E 31 31 31 36 38 32 2C 4C 6F 6E 3A 45 31 31 34 2E 34 30 39 32 31 37 2C 43 6F 75 72 73 65 3A 30 2E 30 30 2C 53 70 65 65 64 3A 30 2E 33 35 31 38 2C 44 61 74 65 54 69 6D 65 3A 31 31 2D 31 31 2D 31 35 20 20 31 31 3A 35 33 3A 34 33 00 02 00 23 07 AE 0D 0A

Content sent by the device:
DWXX=Lat:N23.111682,Lon:E114.409217,Course:0.00,Speed:0.3518,DateTime:11-11-15 11:53:43

**Process of Alarm packet:**

Transmission: 78 78 25 16 0B 0B 0F 0E 24 1D CF 02 7A C8 87 0C 46 57 E6 00 14 02 09 01 CC 00 28 7D 00 1F 72 65 06 04 01 01 00 36 56 A4 0D 0A

Reception: 78 78 05 16 00 36 95 70 0D 0A

## 9.Appendix B:Other online command

**Center cmd online**

Text information: CENTER,A,13800138000# (ADD CENTER NUMBER)

Reception: 78 78 1F 80 19 00 01 A9 58 43 45 4E 54 45 52 2C 41 2C 31 33 38 30 30 31 33 38 30 30 30 23 00 08 59 5F 0D 0A

Transmission: 78 78 1B 15 13 00 01 A9 58 43 45 4E 54 45 52 3D 53 75 63 63 65 73 73 21 00 01 00 09 EC E0 0D 0A

Reply: CENTER=Success!

Text information: CENTER,D# (DEL CENTER NUMBER)

Reception: 78 78 13 80 0D 00 01 A9 58 43 45 4E 54 45 52 2C 44 23 00 08 59 5F 0D 0A

Transmission: 78 78 1B 15 13 00 01 A9 58 43 45 4E 54 45 52 3D 53 75 63 63 65 73 73 21 00 01 00 09 EC E0 0D 0A

Reply: CENTER=Success!

**Timer cmd online**

Text information: TIMER,30#

Reception: 78 78 13 80 0D 00 01 A9 58 54 49 4D 45 52 2C 33 30 23 00 08 FA 74 0D 0A

Transmission: 78 78 1A 15 12 00 01 A9 58 54 49 4D 45 52 3D 53 75 63 63 65 73 73 21 00 01 00 09 8F A0 0D 0A

Reply: TIMER=Success!

**Reset cmd online**

Text information: RESET#

Reception: 78 78 10 80 0A 00 01 A9 58 52 45 53 45 54 23 00 08 28 A0 0D 0A

Transmission: 78 78 1A 15 12 00 01 A9 58 52 45 53 45 54 3D 53 75 63 63 65 73 73 21 00 01 00 09 8E E6 0D 0A

Reply: RESET=Success!

**Factory cmd online**

Text information: FACTORY#

Reception: 78 78 12 80 0C 00 01 A9 58 46 41 43 54 4F 52 59 23 00 08 42 CC 0D 0A

Transmission: 78 78 1C 15 14 00 01 A9 58 46 41 43 54 4F 52 59 3D 53 75 63 63 65 73 73 21 00 01 00 09 74 00 0D 0A

Reply: FACTORY=Success!

**Other online command include:**

VIBRATION,LEVEL,MODE# (Open the vibration alarm)

VIBRATION,0# (Close the vibration alarm)

SPEEDING,VALUE# (Set the speeding alarm)

STATUS# (Show status information)

PARAM# (Show param parameter)

WHERE# (Check the latitude and longitude information)

(Notes:The command format same as the above)