Version: V3.0.17

Secret Level: Confidential

V3 Vehicle Terminal Communication Protocol

(V3.0.17)

Modified Record

Version	Modified Content	Date
V3.0.2	6.6. Add Checking The IMEI Number Of The Module And The ICCID Of The Sim Card	2016-11-25
V3.0.3	5.3.1.17. Alarm command, Alarm field, Add the alarm status of the pseudo base station 5.4.1.7. Heartbeat packet, alarm field, Add the alarm status of the pseudo base station	2016-12-10
V3.0.4	Modify the command issued by the checking module IMEI number and the SIM card ICCID number in 6.6	2016-12-22
V3.0.5	5.3.1.17. Alarm Command, Alarm Field, Add alarm status of illegal demolition 5.4.1.7. Heartbeat packet, Alarm field, Add alarm status of illegal demolition (For A5E-3)	2017-01-05
V3.0.6	Rearrange protocol	2017-03-11
V3.0.7	Modify the response format of terminal in 6.2	2017-03-16
V3.0.8	Add blind area data processing logic description	2017-03-22
V3.0.9	Modify The Description of the Terminal Response Format, delete content code	2017-05-17
V3.0.10	Add voltage level uploading to the positioning data packet (for A5E‐3)	2017-05-22
V3.0.11	Modify The format of positioning data packet, delete voltage level	2017-05-25
V3.0.12	Add the protocol number 0xA0 of the extended field Add the extended command 0xA0/0x00C5	2018-05-15
V3.0.13	Add the extended command 0xA0/0x0088 0xA0/0x0089	2018-05-22
V3.0.14	Add the extended command 0xA0/0x002D	2018-05-23
V3.0.15	Status of the light sensor is uploaded through the alarm field, delete extended command 0xA0/0x00C5	2018-05-30
V3.0.16	Add collision alarm, illegally start alarm, and seeking car alarm	2018-08-16
V3.0.17	Delete The Protocol number 0xA0 of the extended field	2018-09-04

Content

1. Communication Protocol

Foreword

This document defined description of the application layer interface protocol for vehicle GPS tracker and positioning service platform. The relevant interface protocol is only applicable to the interaction between the platform and the positioning terminal.

2. Terms, Definitions

Terms, Abbreviation	English Meaning	Chinese Meaning
CMPP	China Mobile Peer To Peer	中国移动点对点协议
GPS	Global Positioning System	全球卫星定位系统
GSM	Global System For Mobile Communication	全球移动通讯系统
GPRS	General Packet Radio Service	通用无线分组业务
TCP	Transport Control Protocol	传输控制协议
LBS	Location Based Services	辅助定位服务
IMEI	International Mobile Equipment Identity	国际移动设备识别码
MCC	Mobile Country Code	移动用户所属国家代号
MNC	Mobile Network Code	移动网号码
LAC	Location Area Code	位置区码
Cell ID	Cell Tower ID	移动基站
UDP	User Datagram Protocol	用户数据报协议
SOS	Save Our Ship/Save Our Souls	遇难求救信号
CRC	Cyclic Redundancy Check	循环冗余效验
NITZ	Network Identity And Time Zone	时区
GIS	Geographic Information System	地理信息系统

3. Basic Rules

1. The GPRS connection is successfully established and the first login packet is sent to the server. If the server responds to the data packet within 5 seconds, the connection is considered normal, and the location information (GPS, LBS packet) is sent. After 5 minutes, the status packet is sent. Regularly confirm the normal communication.

2. When the GPRS connection establishment is unsuccessful, the terminal cannot send the login packet. When the GPRS connection fails 3 times, the terminal starts the timing restart function, and the time is 20 minutes. If the terminal successfully establishes a connection with

the server within 20 minutes and receives a packet responding to the login packet sent by the server, the timing restart function is closed, the terminal does not restart, otherwise the terminal automatically restarts after 20 minutes.

3. After receiving the login packet sent by the terminal, the server returns a response packet to the terminal. If the terminal does not receive the return packet from the server after sending the login packet or status packet for more than 5 seconds, the current connection is considered abnormal, The terminal will start a GPS positioning data resend function which will disconnect the current GPRS connection and rebuild a new GPRS connection and send a login information packet.

4. The connection is judged to be abnormal. The login packet or status packet sent after the connection is established 3 times fails to receive the data packet responded by the server, and the terminal starts the timing restart function, and the timing time is 20 minutes. If the terminal and the server successfully establish a connection within 20 minutes and receive the data packet that the server responds to, the timing restart function is closed, the terminal does not restart, otherwise the terminal automatically restarts after 20 minutes.

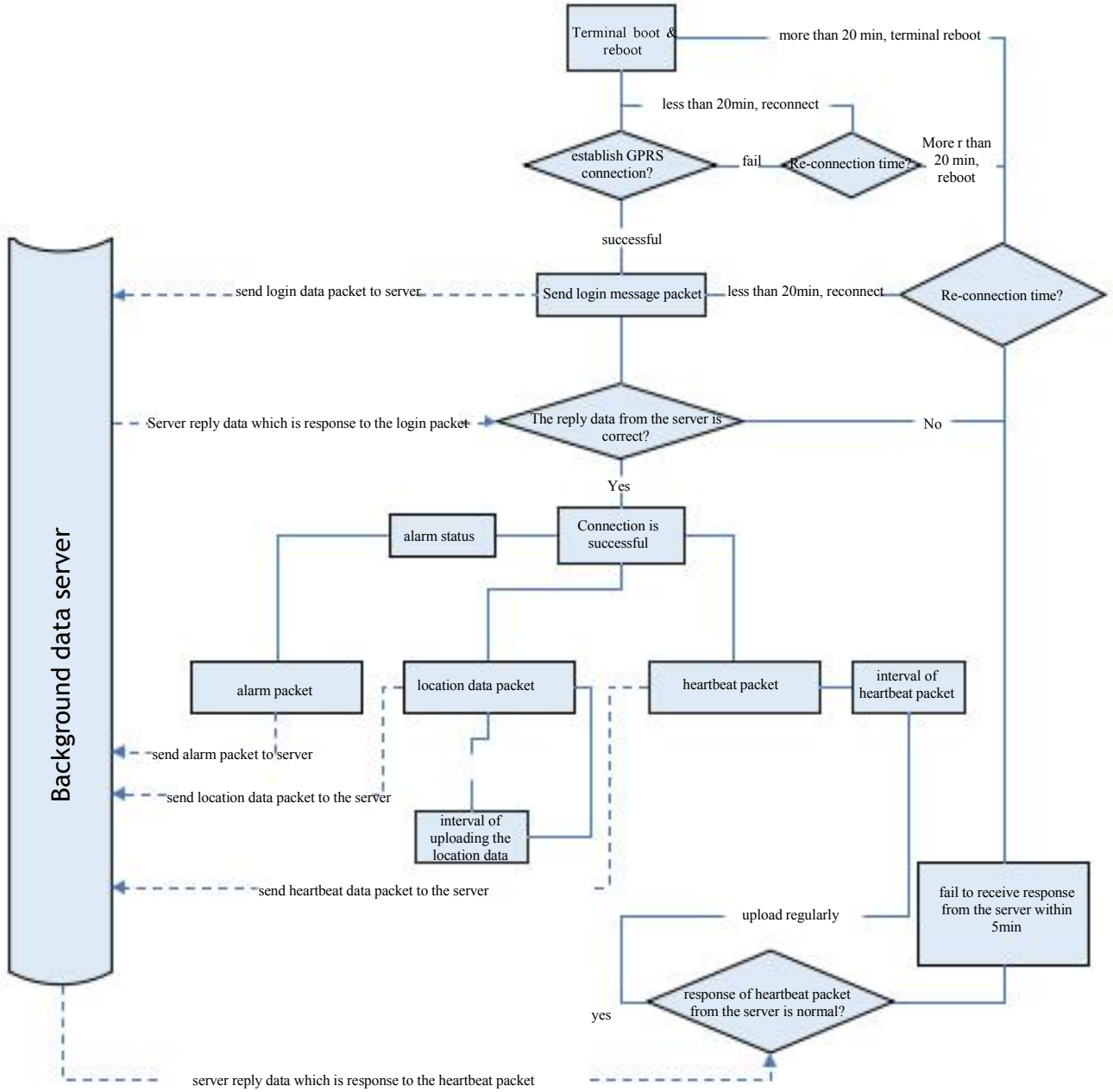
5. After the connection is established normally, the terminal sends the GPS and LBS merged packets to the server periodically after the GPS information is changed. The server can set the default sending protocol by command.

6. To ensure the effectiveness of the connection, the terminal will send status information to the server at a regular time interval, and the server will return response data packet to confirm.

7. For the terminal which doesn't register an IMEI number, please reply a login request response and heartbeat packet response rather than disconnecting the connection directly. (If the connection is disconnected or doesn't reply will cause a continuous reconnected and the GPRS traffic will be heavily consumed).

8. If ACK response is not received after positioning data is sent by the terminal over 180s, the blind area data occurs (blind area data occurs when the terminal is lack of network, connect server unsuccessfully or meet the interval of the location data report), the terminal disconnect from the network and reconnect with the server, starts resend blind area to the platform after receives the heartbeat response. The blind area data can be uploaded as a packet with a maximum of 10 location data. If no ACK response is received, resend 3 times in interval of 4 seconds and the blind area data will be cleared. The blind area data maximum support is 600, if the data is filled will not cover or save blind area data any longer. The data will be lost when terminal power off and reboot, soft reset will not lost.

9. Data Flow Diagram



4. Data Packet Format

The communication transmitted in asynchronous mode, unit is byte

The total length of packet is: (10+N) Byte

Format	Length (Byte)
Start bit	2
Packet length	1
Protocol number	1
Information content	N
Information serial number	2
Error check	2
End bit	2

4.1. Start bit

Fixed value, Unified to hex 0x78 0x78

4.2. Packet length

Length = Protocol number + Information content + Information serial number + error check, (5+N) byte in total, because the information content is a variable long field.

4.3. Protocol number

Type	Value
Login information	0x01
Positioning data	0x22
Status information	0x13
String information	0x15
Alarm data	0x26
Command information sent by the server to the terminal	0x80

4.4. Information content

Determine the specific content according to different applications and corresponding "protocol numbers".

4.5. Information serial number

The serial number of the first GPRS data (including status packet and GPS, LBS, etc.) sent after power-on is '0', and the serial number of each sent data (including status packet and GPS, LBS packet) is automatically added. '1'. Unless it is a power-down reset, the serial number continues to add "1" and does not start from "0".

4.6. Error Checking

The terminal or the server can use the check code to discriminate whether the received information is in error. In order to prevent data from being lost during transmission, error checking is added to prevent erroneous operation of data, which increases the security and efficiency of the system. The error checking code uses the CRC-ITU check method. From the "package length" to the "information serial number" (including "packet length", "information serial number") The CRC-ITU value of this part of the data. If CRC error occurs in the received information calculated, please ignore and discard this data packet.

4.7. End Bit

Fixed value, Unified to hex 0x0D 0x0A

5. Details About Data Packet Sent By Terminal To The Server

Explain the common packet sending and server return separately

5.1. Login Information Packet

5.1.1. Packet Sent By The Terminal To The Server

The login information packet is used to confirm to the server that the connection is established normally, and submit the terminal ID to the server.

Format		Length	Example
Login information	Start bit	2	0x78 0x78
	Packet length	1	0x11

packet (18 bytes)	Protocol number		1	0x01
	Information content	Terminal ID	8	0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45
		Type identification code	2	0x01 0x18
		Time zone language	2	0x32 0x00
	Information serial number		2	0x00 0x01
	Error checking		2	0x8C 0xDD
	End bit		2	0x0D 0x00

5.1.1.1. Start Bit

Refer to the packet format 4.1 for details.

5.1.1.2. Packet Length

Refer to the packet format 4.2 for details.

5.1.1.3. Protocol Number

Refer to the packet format 4.3 for details.

5.1.1.4. Terminal ID

For example: 123456789012345
The terminal ID is: 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45

5.1.1.5. Type Identification Code

Type identification code occupied two bytes. Judge the terminal type according to this identification code. The first 3 bit of two bytes represents the model, and the last bit is the branch of the model.

Example:
Type identification code is:
0x20 0x00 with cutting off fuel function
0x20 0x01 without cutting off fuel function
Note: Fixed to 0x01 0x00

5.1.1.6. Time Zone Language

<table>
<tr><td rowspan="12">One and a half bytes
bit15-bit4</td><td>15</td><td colspan="2" rowspan="12">Value of time zone expanded 100</td></tr>
<tr><td>14</td></tr>
<tr><td>13</td></tr>
<tr><td>12</td></tr>
<tr><td>11</td></tr>
<tr><td>10</td></tr>
<tr><td>9</td></tr>
<tr><td>8</td></tr>
<tr><td>7</td></tr>
<tr><td>6</td></tr>
<tr><td>5</td></tr>
<tr><td>4</td></tr>
<tr><td rowspan="4">Low half bit3-bit0</td><td>3</td><td colspan="2">Time zone East and West</td></tr>
<tr><td>2</td><td colspan="2">No definition</td></tr>
<tr><td>1</td><td>Language chosen bit</td><td>1</td></tr>
<tr><td>0</td><td>Language chosen bit</td><td>0</td></tr>
</table>

Bit3 0------- East time zone

1-------West time zone

If: extension bit: 0x32 0x00 represents Eastern eight zones, GMT+8:00.

Calculation method: 8*100 = 800，Convert to hexadecimal, 0x0320

Extension bit: 0x4D 0xD8 represents Western twelve and 3/4 time zones, GMT-12:45.

Calculation method: 12.45*100 = 1245, Convert to hexadecimal, 0x04,0xDD

The algorithm here is to shift the calculated time zone value to the left by four bits and then combine the time zone east and west, and the language selection bit to save four bytes.

5.1.1.7. Information Serial Number

Refer to the packet format 4.5 for details

5.1.1.8. Error Checking

Refer to the packet format 4.6 for details

5.1.1.9. End Bit

Refer to the packet format 4.7 for details

5.1.2. Server Response Packet

Description		Length	Example
Login information packet (188 byte)	Start bit	2	0x78 0x78
	Packet length	1	0x05
	Protocol number	1	0x01
	Information serial number	2	0x00 0x01
	Error checking	2	0xD9 0xDC
	End bit	2	0x0D 0x0A

Server response packet: (The protocol number in the response packet is the same as the packet protocol number sent by the terminal.)

5.1.2.1. Start Bit

Refer to the packet format 4.1 for details.

5.1.2.2. Packet length

Refer to the packet format 4.2 for details.

5.1.2.3. Protocol Number

Refer to the packet format 4.3 for details.

5.1.2.4. Information Serial Number

Refer to the packet format 4.5 for details.

5.1.2.5. Error Checking

Refer to the packet format 4.6 for details.

5.1.2.6. End Bit

Refer to the packet format 4.7 for details.

5.2. Location Data Packet (Combined Information Packet Of GPS And LBS)

5.2.1. Location Data Packets Sent By The Terminal To The Server

Format			Length	Example
Start bit			2	0x78 0x78
Packet length			1	0x22
Protocol number			1	0x22
Information content	GPS information	Date and time	6	0x0B 0x08 0x1D 0x11 0x2E 0x10
		GPS information satellite	1	0Xcf
		Latitude	4	0x02 0x7A 0xC7 0xEB
		Longitude	4	0x0C 0x46 0x58 0x49
		Speed	1	0x00
		Course, status	2	0x14 0x8F
	LBS information	MCC	2	0x01 0xCC
		MNC	1	0x00
		LAC	2	0x28 0x7D
		Cell ID	3	0x00 0x1F 0xB8
	ACC		1	0x01
	Data upload mode		1	0x01
	GPS real-time uploading		1	0x00
Serial number			2	0x00 0x03
Error checking			2	0x80 0x81
End bit			2	0x0D 0x0A

5.2.1.1. Start Bit

Refer to the packet format 4.1 for details.

5.2.1.2. Packet length

Refer to the packet format 4.2 for details.

5.2.1.3. Protocol Number

Refer to the packet format 4.3 for details.

5.2.1.4. Date And Time

Format	Length (Byte)	Example
Year	1	0x0A
Month	1	0x03
Day	1	0x17
Hour	1	0x0F
Minute	1	0x32
Second	1	0x17

Example: March 23, 2010, 15:50:23
Calculation method: 10 (decimal) = 0A (hexadecimal)
3 (decimal) = 17 (hexadecimal)
23 (decimal) = 17 (hexadecimal)
15 (decimal) = 0F (hexadecimal)
50 (decimal) = 32 (hexadecimal)
23 (decimal) = 17 (hexadecimal)
The value is: 0x0A 0x03 0x17 0x0F 0x32 0x17

5.2.1.5. Length Of GPS Information, The Number Of Satellites Participating In Positioning

1 Byte Two hexadecimal characters displayed, the first character is length of GPS information, the second character is the number of satellites participating in positioning.

Example: when the value is 0xCB means length of GPS information is 12, the number of satellites participating in positioning is 11.(C = 12Byte length, B = 11 satellites)

5.2.1.6. Latitude

Occupies 4 bytes, represents the latitude value of the positioning data. The value range is from 0 to 324000000 which represent the range from 0 degree to 90 degree. The conversion method is as follows:

Convert the latitude and longitude values output by the GPS module into numbers in minutes which the unit is minutes; and then multiply that number by 30,000, and finally convert the result to a hexadecimal number.

Example: 22º 32.7658’ = (22*60 + 32.7658) * 30000 = 40582974，and then convert it into hexadecimal 40582974 (decimal) = 26B3F3E (hexadecimal)
At the last the value is The value is 0x02 0x6B 0x3F 0x3E.

5.2.1.7. Longitude

Occupies 4 bytes, represents the longitude value of the positioning data. The value range is from 0 to 324000000 which represent the range from 0 degree to 180 degree. Convert method is the same as latitude.

5.2.1.8. Speed

Occupies 1 byte, represents the running speed of the GPS. The value ranges from 0x00 to 0xFF. The range is 0 to 255 km/h.

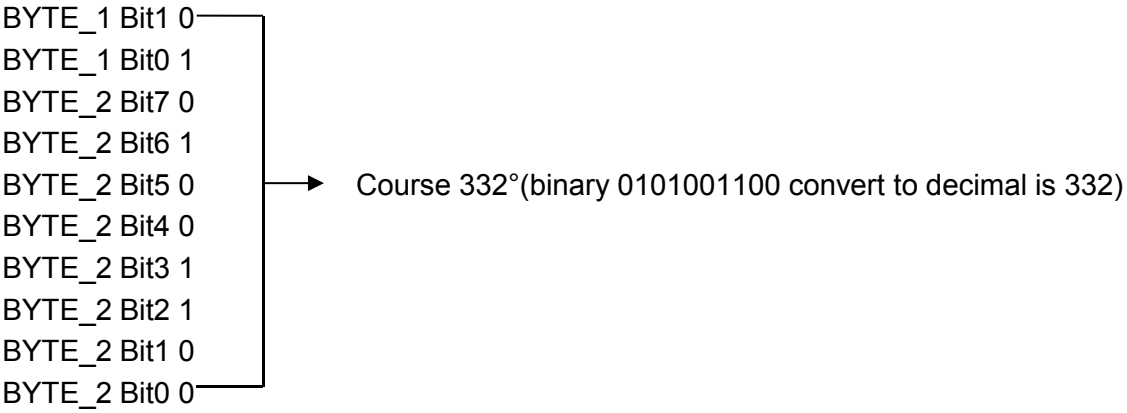
Example: 0x00 represents 0km/h
0x10 represents 16km/h
0xFF represents 255km/h

5.2.1.9. Course, Status

Occupies 2 bytes, represents the moving direction of GPS with a range of 0 to 360, unit is degree, true north is 0 degree, clockwise.

BYTE_1	Bit7	0
	Bit6	0
	Bit5	GPS real time positioning
	Bit4	Whether the GPS is positioning or not
	Bit3	East longitude, West longitude
	Bit2	South latitude, North latitude
	Bit1	Course
	Bit0	
BYTE_2	Bit7	
	Bit6	
	Bit5	
	Bit4	
	Bit3	
	Bit2	
	Bit1	
	Bit0	

Noted: The status information in the data packet is the moment in which the time bits are recorded in the data packet.
Example: Value is 0x15 0x4C, Convert to binary is 00010101 01001100
BYTE_1 Bit7 0
BYTE_1 Bit6 0
BYTE_1 Bit5 0 (Real-time GPS)
BYTE_1 Bit4 1 (GPS is positioned)
BYTE_1 Bit3 0 (East longitude)
BYTE_1 Bit2 1 (North latitude)



Means GPS is positioned, real time GPS, North longitude, East latitude, course 332°

5.2.1.10. MCC

Mobile Country Code(MCC)

Example: China Mobile Country Code is 460 (decimal) 0x01 0xCC (decimal 460 convert into hexadecimal, append 0 on the left when hexadecimal is less than 4 bits)

The value range is: 0x0000 to 0x03E7

5.2.1.11. MNC

Mobile Network Code (MNC)

Example: China Mobile is 0x00.

5.2.1.12. LAC

Location Area Code (LAC) is included in the LAI and consists of two bytes, encoded in hexadecimal. Available range is 0x0001 to 0xFFFE, code block 0x0000 and 0xFFFF is not allowed to use (Refer to GSM standard 03.03, 04.08 and 11.11).

5.2.1.13. Cell ID

Cell Tower ID (Cell ID), value range is 0x000000 to 0xFFFFFF.

5.2.1.14. ACC

ACC status, ACC low is 00, high is 01

5.2.1.15. Data Uploading Mode

GPS Data point uploading type

0x00 Timing uploads

0x01 Fixed distance upload (the terminal didn't do this function)

0x02 Inflection upload

0x03 ACC status' changes upload

0x04 Static mode last position uploads
0x05 Power on login successful uploads directly

5.2.1.16. GPS Real-time upload and blind area upload

0x00 Real-time upload
0x01 Blind area upload

5.2.1.17. Information serial number

Refer to the packet format 4.5 for details

5.2.1.18. Error check

Refer to the packet format 4.6 for details

5.2.1.19. End bit

Refer to the packet format 4.7 for details

5.3. Alarm Packet (GPS, LBS status merged data packet)

5.3.1. Terminal sends alarm packet to the server

Format			Length	Example
Start bit			2	0x78 0x78
Packet length			1	0x25
Protocol number			1	0x26
Date and time			6	0x0B 0x08 0x1D 0x11 0x2E 0x10
Information content	GPS information	Length of GPS information + number of satellite	1	0xCF
		Latitude	4	0x02 0x7A 0xC7 0xEB
		Longitude	4	0x0C 0x46 0x58 0x49
		Speed	1	0x00
		Course, status	2	0x14 0x8F
	LBS information	Length of LBS	1	0x08
		MCC	2	0x01 0xCC
		MNC	1	0x00
		LAC	2	0x28 0x7D
		Cell ID	3	0x00 0x1F 0xB8
	Status	Terminal information	1	0x40

	information	content		
		Voltage level	1	0x06
		GSM signal strength	1	0x04
		Alarm/voice/extension status	2	0x00 0x02
Serial number			2	0x00 0x1F
Error checking			2	0x00 0x1F
End bit			2	0x0D 0x0A

Remark: The alarm packet adds status information (alarm information) based on the positioning data packet, and the encoding protocol format also adds status information based on the positioning data packet.

5.3.1.1. Start Bit

Refer to the packet format 4.1 for details.

5.3.1.2. Packet length

Refer to the packet format 4.2 for details.

5.3.1.3. Protocol Number

Refer to the packet format 4.3 for details.

5.3.1.4. Date And Time

Refer to the positioning data format 5.2.1.4 for details.

5.3.1.5. Length of GPS information, number of satellite

Refer to the positioning data format 5.2.1.5 for details

5.3.1.6. Latitude

Refer to the positioning data format 5.2.1.6 for details

5.3.1.7. Longitude

Refer to the positioning data format 5.2.1.7 for details

5.3.1.8. Speed

Refer to the positioning data format 5.2.1.8 for details

5.3.1.9. Course, Status

Refer to the positioning data format 5.2.1.9 for details

5.3.1.10. MCC

Refer to the positioning data format 5.2.1.10 for details

5.3.1.11. MNC

Refer to the positioning data format 5.2.1.11 for details

5.3.1.12. LAC

Refer to the positioning data format 5.2.1.12 for details

5.3.1.13. Cell ID

Refer to the positioning data format 5.2.1.13 for details

5.3.1.14. Terminal information

Occupies 1 byte, indicate various status information

Bit		Code meaning
BYTE	Bit7	1: Cut off fuel
		0: Restore fuel
	Bit6	1: GPS is positioned
		0: GPS is not positioned
	Bit3~ Bit5	110: Over-speed alarm
		101: Timeout alarm (fatigue driving)
		100: SOS Alarm
		011: Low power alarm
		010: Power off alarm
		001: Vibration alarm
		000: Normal
	Bit2	1: Charging by connect to the power supply
		0: Not charging
	Bit1	1: ACC high
		0: ACC low
	Bit0	1: Set defense
		0: Cancel defence

For example: 0x44, The corresponding binary is 01000100
Indicates that the terminal status is: The fuel is turned on, the GPS is already positioned, the

alarm is not normal, the power is charged, and the ACC is Low, disarmed status.

5.3.1.15. Voltage level

The range is 0 to 6, indicating that the voltage is from low to high.
0: No power (shutdown)
1: Extremely low battery (not enough for calling or sending text messages, etc.)
2: Very low battery
3: Low battery (can be used normally)
4: Medium battery
5: High
6: Very high
E,g: 0x02 indicates very low battery

5.3.1.16. GSM signal strength level

0x00: No signal
0x01: Extremely weak signal
0x02: Very weak signal
0x03: Good signal
0x04: Strong signal
E.g.: 0x03 indicates the GSM signal is good

5.3.1.17. Alarm/Language

0x00 (former bit) 0x01 (latter bit)
Former bit: terminal alarm status
Latter bit: the language that terminal currently used

BYTE 1	0x00	Normal
	0x01	SOS Alarm
	0x02	Power off alarm
	0x03	Vibration alarm
	0x04	Enter geo-fence alarm
	0x05	Exit geo-fence alarm
	0x06	Over speed alarm
	0x09	Displacement alarm
	0x0A	LBS alarm
	0x0C	0x0C illegal removal alarm (highest priority)
	0x0D	Collision alarm
	0x0E	Illegal starting alarm
	0x0F	Car searching alarm
BYTE 2	State[0~7] defaults is 0x000 Bit0~Bit3 represents the language type	

	0x01 Chinese 0x02 English Bit4 Light sensor status 1: Detected light 0: No light detected For example: The language type is Chinese. If light is detected, upload 0x11. Bit5~Bit7 reserved.

Example:
No alarm Chinese: 0x00 0x01
No alarm English: 0x00 0x01
In order to increase the reliability of the alarm information, the alarm information is repeatedly marked. In most cases, the alarm information is consistent with the uploaded terminal information. The inconsistency is as follows:
A. Low battery alarm occurs in the terminal information
B. Geo-fence enters and exit alarm in the alarm/language information
C. LBS alarm
D. Illegal dismantle alarm
Remark: The illegal removal alarm has the highest priority level, triggers the alarm, and no other alarms are uploaded until the alarm is released.

5.3.1.18. Information serial number

Refer to the packet format 4.5 for details

5.3.1.19. Error check

Refer to the packet format 4.6 for details

5.3.1.20. End bit

Refer to the packet format 4.7 for details
Note: The status information in the data packet is the status at the moment of the time bit recorded in the data packet.

5.3.2. Server send alarm packet reply to the terminal.

Format	Length (Byte)	Example
Start bit	2	0x78 0x78
Packet length	1	0x05
Protocol number	1	0x26
Serial number	2	0x00 0x01
Error check	2	0xD9 0xDC
End bit	2	0x0D 0x0A

The alarm data packet is added with status information (alarm information) based on the positioning data packet, and the encoding protocol format also adds status information based on the positioning data packet.

5.3.2.1. Start Bit

Refer to the packet format 4.1 for details.

5.3.2.2. Packet length

Refer to the packet format 4.2 for details.

5.3.2.3. Protocol Number

Refer to the packet format 4.3 for details.

5.3.2.4. Information serial number

Refer to the packet format 4.5 for details.

5.3.2.5. Error check

Refer to the packet format 4.6 for details.

5.3.2.6. End bit

Refer to the packet format 4.7 for details.

5.4. Heartbeat Packet (Status Packet)

The heartbeat packet is a packet that maintains the connection between the terminal and the server.

5.4.1. Terminal send heartbeat packet to the server.

Format		Length (Byte)	Example
Start bit		2	0x78 0x78
Packet length		1	0x0A
Protocol number		1	0x13
Information content	Terminal information content	1	0x40
	Voltage level	1	0x06
	GSM signal strength	1	0x04
	Status of alarm/language/extension port	2	0x00 0x01
Serial number		2	0x00 0x1F
Error check		2	0xC4 0x39
End bit		2	0x0D 0x0A

5.4.1.1. Start Bit

Refer to the packet format 4.1 for details.

5.3.1.2. Packet length

Refer to the packet format 4.2 for details.

5.4.1.3. Protocol Number

Refer to the packet format 4.3 for details.

5.4.1.4. Terminal information (Heartbeat data does not contain alarm information)

Occupies 1 byte, indicate various status information

Bit		Code meaning
BYTE	Bit7	1: Cut off fuel
		0: Restore fuel
	Bit6	1: GPS is positioned
		0: GPS is not positioned
	Bit3~ Bit5	100: Reserved
		101: Timeout alarm
		110: Over-speed alarm
		011: Low power alarm
		010: Power off alarm
		001: Vibration alarm
		000: Normal

	Bit2	1: Charging by connect to the power supply
		0: Not charging
	Bit1	1: ACC high
		0: ACC low
	Bit0	1: Set defense
		0: Cancel defence

For example: 0x44, The corresponding binary is 01000100

Indicates that the terminal status is: The fuel is turned on, the GPS is already positioned, the alarm is not normal, the power is charged, and the ACC is Low.

5.4.1.5. Voltage level

The range is 0 to 6, indicating that the voltage is from low to high.
0: No power (shutdown)
1: Extremely low battery (not enough for calling or sending text messages, etc.)
2: Very low battery
3: Low battery (can be used normally)
4: Medium battery
5: High
6: Very high
E,g: 0x02 indicates very low battery

5.4.1.6. GSM signal strength level

0x00: No signal
0x01: Extremely weak signal
0x02: Very weak signal
0x03: Good signal
0x04: Strong signal
E.g.: 0x03 indicates the GSM signal is good

5.4.1.7. Alarm/Language (Upload terminal alarm information)

0x00 (former bit) 0x01 (latter bit)
Former bit: terminal alarm status
Latter bit: the language that terminal currently used

Former bit	0x00	Normal
	0x01	SOS Alarm
	0x02	Power off alarm
	0x03	Vibration alarm
	0x04	Enter geo-fence alarm
	0x05	Exit geo-fence alarm

	0x06	Over speed alarm
	0x09	Displacement alarm
	0x0A	Pseudo base station alarm
	0x0C	0x0C illegal removal alarm (highest priority)
	0x0D	Collision alarm
	0x0E	Illegal starting alarm
	0x0F	Car searching alarm
Latter bit	State[0~7] defaults is 0x000 Bit0~Bit3 represents the language type 0x01 Chinese 0x02 English Bit4 Light sensor status 1: Detected light 0: No light detected For example: The language type is Chinese. If light is detected, upload 0x11. Bit5~Bit7 reserved.	

Example:
No alarm Chinese: 0x00 0x01
No alarm English: 0x00 0x01
Remark: The illegal removal alarm has the highest priority level, triggers the alarm, and no other alarms are uploaded until the alarm is released.

5.4.1.8. Information serial number

Refer to the packet format 4.5 for details

5.4.1.9. Error check

Refer to the packet format 4.6 for details

5.4.1.10. End bit

Refer to the packet format 4.7 for details

5.4.2. Server response packet

Format		Length	Example
Heartbeat Response Packet (10Byte)	Start bit	2	0x78 0x78
	Packet length	1	0x05
	Protocol number	1	0x13
	Information serial number	2	0x00 0x01
	Error check	2	0xD9 0xDC

	End bit	2	0x0D 0x0A

The response packet from the server to the terminal: (The protocol number in the response packet is the same as the packet protocol number sent by the terminal)

5.4.2.1. Start Bit

Refer to the packet format 4.1 for details.

5.4.2.2. Packet length

Refer to the packet format 4.2 for details.

5.4.2.3. Protocol Number

Refer to the packet format 4.3 for details.

5.4.2.4. Information serial number

Refer to the packet format 4.5 for details.

5.4.2.5. Error check

Refer to the packet format 4.6 for details.

5.4.2.6. End bit

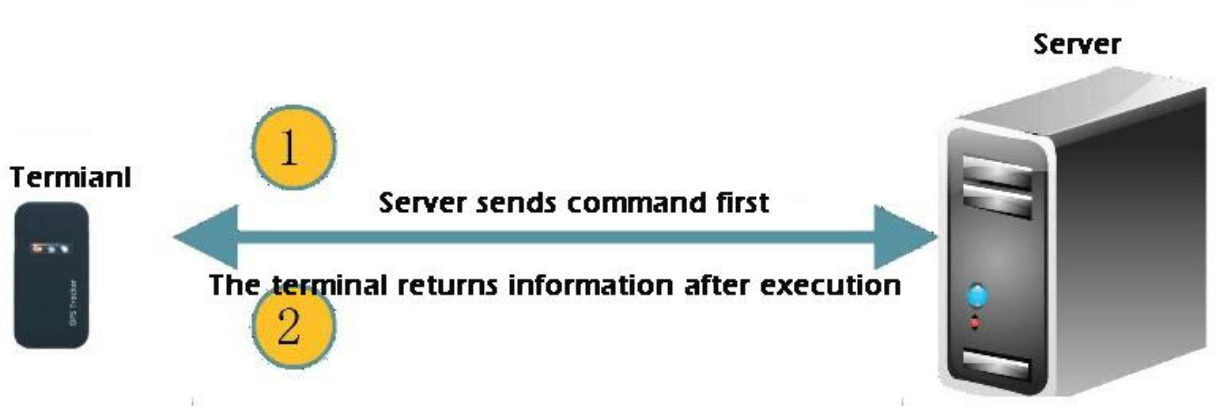
Refer to the packet format 4.7 for details.

5.4.3. Data example

Terminal transmission example:					
78 78 0A 13 04 06 03 00 02 00 04 1D A3 0D 0A					
Explanation:					
0x78 0x78 Start bit	0x0A Length	0x13 Protocol number	0x04 0x06 0x03 Information content	0x00 0x02 Reserved bit (language)	
0x00 0x04 Serial number			0x1D 0XA3 Error check	0x0D 0x0A End bit	
Server response example:					
78 78 05 13 00 11 F9 70 0D 0A					
Explanation:					
0x78 0x78 Start bit	0x05 Length	0x13 Protocol number	0x00 0x11 Serial number	0xF9 0x70 Error check	0x0D 0x0A End bit

An example of a heartbeat response for reference

6. Server send packet to the terminal.



6.1 Packet sent by server

Format		Length (Byte)	Example
Start bit		2	0x78 0x78
Packet length		1	0x0F
Protocol number		1	0x80
Information content	Command Length	1	0xCA
	Server mark bit	4	0x00 0x00 0x00 0x01
	Command content	M	
	Language	2	
Information serial number		2	0x00 0x01
Error check		2	0xD9 0xDC
End bit		2	0x0D 0x0A

6.1.1. Start Bit

Refer to the packet format 4.1 for details.

6.1.2. Packet length

Refer to the packet format 4.2 for details.

6.1.3. Protocol Number

Server send protocol number: 0x80

6.1.4. Command length

Server mark bit + command content length (4+M)

For example, in the unit of byte length, 0x0A, that means the command content occupies 6 bytes.

6.1.5. Server mark bit

It is reserved to the server for identification, the terminal will return the received data binary in the returned packet without changing it.

6.1.6. Command content

Indicates with ASC II of string, the command content is compatible with text message command.

6.1.7 Language

Current language bit of the terminal.

Chinese: 0x00 0x01

English: 0x00 0x02

6.1.8. Information serial number

Refer to the packet format 4.5 for details.

6.1.9. Error check

Refer to the packet format 4.6 for details.

5.1.10. End bit

Refer to the packet format 4.7 for details.

6.2 Terminal response

Format	Length (Byte)	Example
Start bit	2	0x78 0x78
Packet length	1	0x0F
Protocol number	1	0x15

Information content	Command Length	1	0xCA
	Server mark bit	4	0x00 0x00 0x00 0x01
	Command content	M	
	Language	2	
Information serial number		2	0x00 0x01
Error check		2	0xD9 0xDC
End bit		2	0x0D 0x0A

6.2.1. Start Bit

Fixed value 0x78 0x78

6.2.2. Packet length

Occupies 1 byte

6.2.3. Protocol Number

Use 0x15

6.2.4. Length of command

Server mark bit + command content length (4+M)

For example, in the unit of byte length, 0x0A, that means the command content occupies 6 bytes.

6.2.5. Server mark bit

It is reserved to the server for identification, the terminal will return the received data binary in the returned packet without changing it.

6.2.6. Command content

Data to send

6.2.7. Information serial number

Refer to the packet format 4.5 for details.

6.2.8. Error check

Refer to the packet format 4.6 for details.

6.2.9. End bit

Refer to the packet format 4.7 for details.

6.2.10. Data example

<table>
<tr><td colspan="5">Server sending example:</td></tr>
<tr><td colspan="5">78 78 12 80 0A 00 00 00 01 52 45 53 45 54 23 00 02 00 34 73 2E 0D 0A</td></tr>
<tr><td colspan="5">Explanation:</td></tr>
<tr><td>0x78 0x78
Start bit</td><td>0x12
Length</td><td>0x80
Protocol number</td><td>0x0A
Command length</td><td>0x00 0x00 0x00 x01
Server mark bit</td></tr>
<tr><td colspan="3">0x52 0x45 0x53 0x45 0x54 0x23
Command content</td><td>0x00 0x02
Language</td><td>0x00 0x34
Serial number</td></tr>
<tr><td colspan="3">0x73 0x2E
Error check</td><td colspan="2">0x0D 0x0A
End bit</td></tr>
<tr><td colspan="3">Terminal response example:</td><td colspan="2"></td></tr>
<tr><td colspan="5">78 78 32 15 2A 00 00 00 01 54 68 65 20 74 65 72 6D 69 6E 61 6C 20 77 69 6C 6C 20 72 65 73 74 61 72 74 20 61 66 74 65 72 20 32 30 73 65 63 21 00 02 00 0A 1B F0 0D 0A</td></tr>
<tr><td colspan="5">Explanation:</td></tr>
<tr><td>0x78 0x78
Start bit</td><td>0x32
Length</td><td>0x15
Protocol number</td><td colspan="2">0x00 0x00 0x00 0x01
Server mark bit</td></tr>
<tr><td colspan="5">0x54 0x68 0x65 0x20 0x74 0x65 0x72 0x6D 0x69 0x6E 0x61 0x6C 0x20 0x77 0x69 0x6C 0x6C 0x20 0x72 0x65 0x73 0x74 0x61 0x72 0x74 0x20 0x61 0x66 0x74 0x65 0x72 0x20 0x32 0x30 0x73 0x65 0x63 0x21
Command content</td></tr>
<tr><td>0x00 0x02
Language</td><td>0x00 0x0A
Serial number</td><td>0x1B 0xF0
Error check</td><td colspan="2">0x0D 0x0A
End bit</td></tr>
</table>

7. Appendix A: C language code fragment of the CRC-ITU lookup table algorithm

CRC-ITU lookup table algorithm C language code fragment

```
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 bits CRC of data with a given 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; // opposite
}
```

8. Appendix B: Example of communication protocol data packet fragments

The following data displayed in hexadecimal are intercepted from the communication between terminal and server, transmission means sent be the terminal and reception means returned from server:
Login packet:
Transmission: 78 78 11 01 00 00 01 31 22 33 44 55 01 00 32 00 00 00 21 AD 0D 0A
Reception: 78 78 05 01 00 01 D9 DC 0D 0A

GPS data packet (use combined information packet of GPS, LBS)
Transmission: 78 78 22 22 11 02 1B 06 01 0D C5 02 6D DE C0 0C 3B FE E6 25 14 00 01 CC 00 26 2C 00 0E BA 00 00 00 00 06 2B 7D 0D 0A

Status packet:
Transmission: 78 78 0A 13 04 06 03 00 02 00 03 69 1C 0D 0A
Reception: 78 78 05 13 00 11 F9 70 0D 0A

Alarm packet:
Transmission: 78 78 25 26 11 02 04 06 11 39 C5 02 6D DE C0 0C 3B FE E6 23 14 49 08 01 CC 00 26 2C 00 0E BA 54 00 03 02 02 00 6B FB EE 0D 0A (power off alarm)
Transmission: 78 78 25 26 11 02 04 06 11 01 C5 02 6D DE C0 0C 3B FE E6 5B 14 49 08 01 CC 00 26 2C 00 0E BA 74 06 04 06 02 00 67 0B 36 0D 0A (Over speed alarm)

Disconnect oil and electricity online:
Reception: 78 78 14 80 0C 00 00 00 01 52 45 4C 41 59 2C 31 23 00 02 00 3C 04 2B 0D 0A
Transmission: 78 78 19 15 11 00 00 00 01 C7 D0 B6 CF D3 CD B5 E7 B3 C9 B9 A6 21 00 02 10 22 AB 33 0D 0A
Server send down RELAY,1#
Respond: Disconnect oil and electricity successful

Send down command at the situation of oil and electricity are already disconnected:
Reception: 78 78 14 80 0C 00 00 00 01 52 45 4C 41 59 2C 31 23 00 02 01 30 D7 9F 0D 0A
Transmission: 78 78 31 15 29 00 00 00 01 D6 D5 B6 CB D2 D1 B4 A6 D3 DA B6 CF D3 CD B5 E7 D7 B4 CC AC A3 AC B1 BE D6 B8 C1 EE B2 BB D4 D9 D6 B4 D0 D0 21 00 02 00 06 EF C6 0D 0A
Server send down RELAY,1#
Respond: Terminal is already in the state of oil and electricity disconnected, this command will not execute!

Restore oil and electricity online:

Reception: 78 78 14 80 0C 00 00 00 01 52 45 4C 41 59 2C 30 23 00 02 01 31 C2 3D 0D 0A
Transmission: 78 78 19 15 11 00 00 00 01 BB D6 B8 B4 D3 CD B5 E7 B3 C9 B9 A6 21 00 02 00 07 3B 78 0D 0A
Server send down RELAY,0#
Respond: Restore oil and electricity successful

Send down command at the situation of oil and electricity are already restored:
Reception: 78 78 14 80 0C 00 00 00 01 52 45 4C 41 59 2C 30 23 00 02 01 32 F0 A6 0D 0A
Transmission: 78 78 33 15 2B 00 00 00 01 D6 D5 B6 CB D2 D1 B4 A6 D3 DA D3 CD B5 E7 BD D3 CD A8 D7 B4 CC AC A3 AC B1 BE D6 B8 C1 EE B2 BB D4 D9 D6 B4 D0 D0 21 00 02 00 08 28 59 0D 0A
Server send down RELAY,0#
Respond: Terminal is already in the state of oil and electricity connected, this command will not execute!

9. Appendix C: Complete format of information packet

A. Data packet sent by the terminal to server

Login information packet (22 Byte)								
Start bit	Packet length	Protocol number	Terminal ID	Type identification code	Time zone language	Information serial number	Check bit	End bit
2	1	1	8	2	2	2	2	2

GPS, LBS information packet (36+ M+N Byte)	Start bit			1
	Packet length			1
	Protocol Number			2
	Information content	Date and time		6
		GPS information	Length of GPS information, the number of satellites involved in positioning	1
			Latitude	4
			Longitude	4
			Speed	1
			Course, status	2
			Reserved extension bit	M
		LBS information	MCC	2
			MNC	1
			LAC	2
			Cell ID	3
		Reserve extension		M
	Information serial number			2

	Check bit	2
	End bit	2

Status packet (15Byte)	Start bit		2
	Packet length		1
	Protocol number		1
	Information content	Terminal information content	1
		Voltage level	1
		GSM signal strength level	1
		Reserve extension bit (language)	2
	Information serial number		2
	Check bit		2
	End bit		2

Terminal responds command sent by server (15+M+N Byte)	Start bit		2
	Packet length		1
	Protocol number		1
	String content	Length of command	1
		Server sign bit	4
		Command content	M
		Reserve extension bit (language)	N
	Information serial number		2
	Check bit		2
	End bit		2

B. Data packet sent by server to the terminal

Server's response after received the status packet sent by the terminal (10 Byte)					
Start bit	Packet length	Protocol No.	Information serial number	Check bit	End bit
2	1	1	2	2	2

Command packet sent by server to the terminal (15+M+N Byte)	Start bit		2
	Packet length		1
	Protocol number		1
	String content	Command length	1
		Server sign bit	4
		Command content	M
		Reserve extension bit (language)	N
	Information serial number		2
	Check bit		2
	End bit		2