Shenzhen Jimi Electronic Co., Ltd.

Communication Protocol

(3G Car DVR)

CONTENT

更新说明

1、在 6.16 中添加了上传照片的指令（PICTURE,OUT）

1. Communication Protocol

This document defines instructions about interface protocol on application layer of car rearview mirror, location-based service platform and image service platform. Related interface protocol only applies in the interaction between the platform and the position terminal.

2. Terms and Definitions

Terms, Abbreviation	Definition in English	Definition in Chinese
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	全球移动通信系统
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	循环冗余校验
NITZ	Network Identity and Time Zone,	时区
GIS	Geographic Information System	地理信息系统

3. Basic Rules

1). If a GPRS connection is established successfully, the terminal will send a first login message packet to the server and, within five seconds, if the terminal receives a data packet responded by the server, the connection is considered to be a normal connection. The terminal will begin to send location information (i.e., GPS, LBS information package). A status information package will be sent by the terminal after three minutes to regularly confirm the connection.

2). If the GPRS connection is established unsuccessfully, the terminal will not be able to send the first login message packet. The terminal will start scheduled reboot in twenty minutes if the GPRS connection failed for three times. Within twenty minutes, if the terminal successfully connected to the server and received the data packet from the server as the server's response to the login message packet sent by the terminal, the scheduled reboot will be off and the terminal will not reboot; otherwise, the terminal will reboot automatically in twenty minutes.

3). After receiving the login message packet, the server will return a response data packet. If the terminal doesn't receive packet from the server within five seconds after sending the login message packet or the status information package, the current connection is regarded as an abnormal connection. The terminal will start a re-transmission function for GPS tracking data, which will cause the terminal to disconnect the current GPRS connection, rebuild a new GPRS connection and send a login message packet again.

4). In case of the normal connection, the terminal will send a combined information package of GPS and LBS to the server after the GPS information is changed; and the server may set a default protocol for transmission by using commands.

5). To ensure the effectiveness of the connection, the terminal will send status information to the server at regular intervals, and the server will return response data packets to confirm the connection.

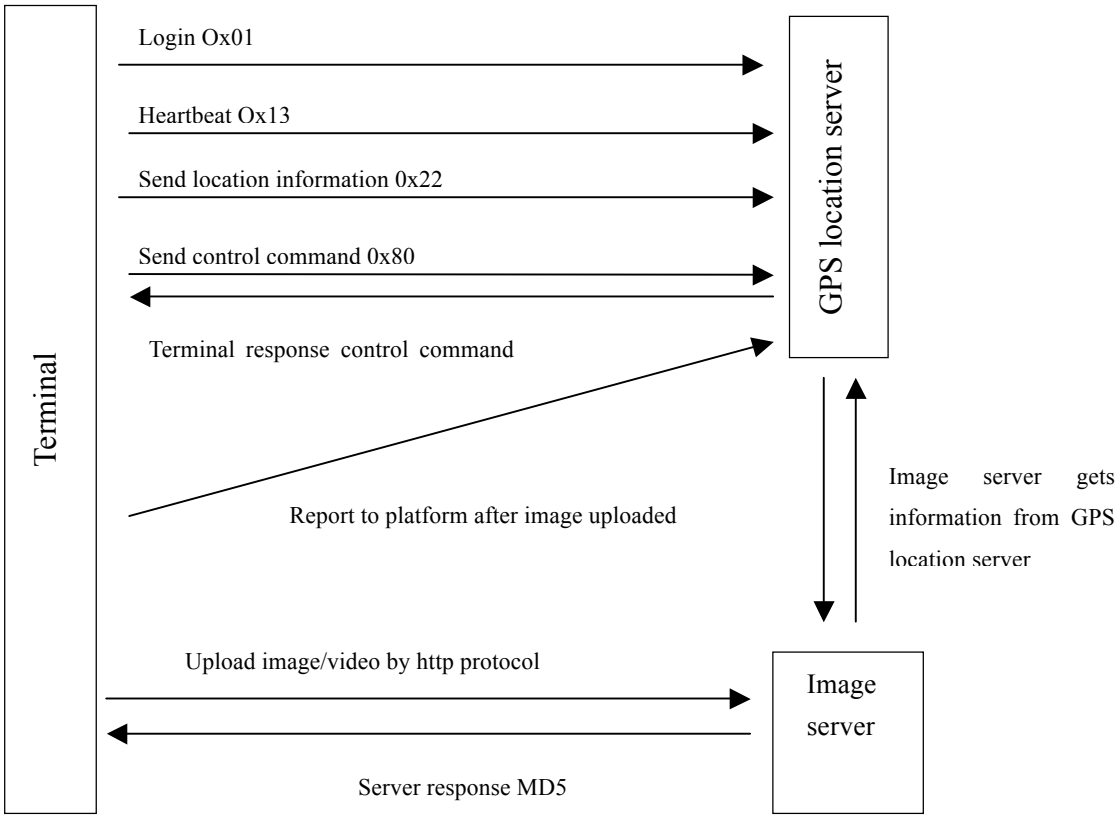
6). For the terminal which doesn't register an IMEI number, the server will reply the terminal with a login request response and heartbeat packet response, rather than directly disconnect the connection. (If the connection is directly disconnected or the server doesn't reply to the terminal, it will lead to a continuous reconnected by the terminal and the GPRS traffic will be consumed heavily.

Description: In order to rapidly develop compatible background and APP, the rearview mirror (C2) access to the original project "OBD (GT520_GT530_GT230_GT250_GT260) _GPS locator _ communication protocol, upload the positioning data protocol and GPS server, and keep the heartbeat, long established connection. Due to the protocol defines a 1-byte packet, it is impossible to extend the function required by uploading photo and video data for C2 rearview mirror. So we need to expand the other protocol and server (here referred to as the photo server). To simplify the design, the terminal no longer sends a separate login and heartbeat packet to the photo server, the photo server obtains the terminal is online and other related information.

Domain name of GPS location server: (It's a temporary IP address, the domain name will release later）
120.237.91.74:21100

Domain name of image server:
http://resource.jimicloud.com/upload

Data Flow Diagram



There are two ways to upload image and video:

1. Upload by APP

Process：

（1） Location server sends control command to terminal, asking for uploading image or video.

（2） Terminal responses the command.

（3） Terminal uploads image or video to image server through http protocol.

（4） Image server returns result message and the MD5 information of image/video. If the result is failure, the terminal will upload image/video again.

（5） Terminal reports IMEI, MD5 information of image/video and file type to location platform.

Terminal should send again if it receives the failure message returned by image server.

2. Upload by terminal （Used for future extensions）

Process：Same as (3), (4), (5); (1) and (2) are omitted

Note：Currently, C2, C6 only support to upload by APP. Terminal upload automatically for future extension.

4. Data Packet Format (GPS Location Server Protocol)

The communication is transferred asynchronously in bytes.
The total length of packets is (10+N) Bytes.

Format	Length(Byte)
Start Bit	2
Packet Length	1
Protocol Number	1
Information Content	N
Information Serial Number	2
Error Check	2
Stop Bit	2

4.1. Start Bit

Fixed value in HEX: 0x78 0x78

4.2. Packet Length

Length = Protocol Number + Information Content + Information Serial Number + Error Check, totally (5+N) Bytes, because the Information Content is in a variable length field.

4.3. Protocol

Type	Value
Login Message	0x01
Status information(Heartbeat)	0x13
Location Data Packet	0x22
Alarm Data Packet	0x95
General information transmission packet	0X94
LBS Multiple Base Station Data Packet（LBS Extended Information Packet）	0x28
Net Time Checking	8A
OBD Packet	8E
OBD Control Packet	8F
Command information sent by the server to the terminal	0x80
Image Sent by Terminal to Server MD5	0x30

4.4. Information Contents

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

4.5. Information Serial Number

The serial number of the first GPRS data (including status packet and data packet such as GPS, LBS) sent after booting is ‘1’, and the serial number of data sent later at each time will be automatically added ‘1’.

4.6. Error Checking

A check code may be used by the terminal or the server to distinguish whether the received information is wrong or not. To prevent errors occur during data transmission, error checking is added to against data inappropriate operation, so as to increase the security and efficiency of the system. The check code is generated by the CRC-ITU checking method.

The check codes of data in the structure of the protocol, from the Packet Length to the Information Serial Number (including “Packet Length” and “Information Serial Number”) , are the value of CRC-ITU.

If CRC error occurs when the received information is calculated; the receiver will ignore and discard the data packet.

4.7. Stop Bit

Fixed value in HEX: 0x0D 0x0A

5. Details about Data Packet Sent by Terminal to Server （GPS Location Server Protocol）

The commonly used information packages sent by the terminal and those responded by the server will be interpreted separately.

5.1. Login Message Packet (0x01)

5.1.1 Data Packet Sent by Terminal to Server

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

Format		Length
Login Message Packet(18 Bytes)	Start Bit	2
	Packet Length	1
	Protocol Number	1
	Terminal ID	8
	Type Identification Code	2
	Time Zone Language	2
	Information Serial Number	2
	Error Check	2
	Stop Bit	2

Example: 78 78 11 01 07 52 53 36 78 90 02 42 70 00 32 01 00 05 12 79 0D 0A

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

Protocol number: 0x01

5.1.1.4. Terminal ID

Example: if the IMEI is 123456789012345, then the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

5.1.1.5. Type Identification Code

Type identification code consumes two bytes, which shows the types of the terminals. The first three bits represent the type of the terminal, while the last bit represents the branch of the type.

5. 1. 1. 6. Time Zone Language

One and a half bits bit15—bit4	15	Time zone expands 100 value
	14	
	13	
	12	
	11	
	10	
	9	
	8	
	7	
	6	
	5	

	4		
Last nibble（bit4-bit0）	3	GMT	
	2	No definition	
	1	Language choice bit	1
	0	Language choice bit	0

Bit3 0------- Eastern time

1------- Western time

Example: Extended bit: 0x32 0x00 means GMT+8

Calculation method: 8*100=800 converts to HEX: 0X0320

Extended bit: 0x4D 0xD8 means GMT-12:45

Calculation method: 12.45*100=1245 converts to HEX: 0x04 0xDD

5.1.1.7. Information Serial Number

For details see Data Packet Format section 4.5.

5.1.1.8. Error Check

For details see Data Packet Format section 4.6.

5.1.1.9. Stop Bit

For details see Data Packet Format section 4.7.

5.1.2. Server Responds the Data Packet

Description		Bits
Login Message Packet (18 Byte)）	Start Bit	2
	Packet Length	1
	Protocol Number	1
	Information Serial Number	2
	Error Check	2
	Stop Bit	2

The response packet from the server to the terminal: the protocol number in the response packet is identical to the protocol number in the data packet sent by the terminal.
Example: 78 78 05 01 00 05 9F F8 0D 0A

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. Stop Bit

For details see Data Packet Format section 4.7.

5.2. Heartbeat Packet（Information Status Packet）（0x13）

Heartbeat Packet Sent by Terminal to Server

Format			Length (Byte)
Information Content	Start Bit		2
	Packet Length		1
	Protocol Number		1
	Status Information	Terminal Information Content	1
		Voltage Level	1
		GSM Signal Strength	1
		Alarm/Extended Port Status	2
	Serial Number		2
	Error Check		2
	Stop Bit		2

Example: 78 78 0A 13 40 04 04 00 01 00 0F DC EE 0D 0A

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

Protocol Number: 0x13

5.2.1.4. Terminal Information

1 byte is consumed defining for various status information of the terminal.

Note：For rearview mirror JC900 and JC600，terminal information is newly-defined as the following:

Bit		Code Meaning
BYTE	Bit7	1：Oil/Electricity cutoff
		0：Oil/Electricity on

	Bit6	1: GPS tracking is on
		0: GPS tracking is off
	Bit3～Bit5	Extended bit
	Bit2	1: Charge On
		0: Charge Off
	Bit1	1: ACC high
		0: ACC Low
	Bit0	1：Arm
		0: Disarm

For example： The binary encoding of 0x44 01000100

Note： Terminal information is newly defined for rearview mirror series：

Byte		Code Meaning
BYTE	Bit7	Extended bit
	Bit6	1: GPS tracking is on
		0: GPS tracking is off
	Bit3～Bit5	Extended bit
	Bit2	1: Charge On
		0: Charge Off
	Bit1	1: ACC high
		0: ACC Low
	Bit0	Extended bit

To JC900 and JC900R Rear view mirror: (For terminal firmware engineer)

Remark：when ACC is on，device will be in charging mode。

5.2.1.5. Voltage Grade

The range is 0~6 , defining 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 (Low Battery Alarm)

3: Low Battery (can be used normally)

4: Medium

5: High

6: Very High

For C2 and C6

0：Battery≤3%

1：Battery≤6%

2：Battery≤10%

3：Battery≤15%

4：Battery≤45%
5：Battery≤75%
6：Battery≤100%

5.2.1.6. GSM Signal Strength Level

0x00: no signal;
0x01: very weak signal;
0x02: weak signal;
0x03: good signal;
0x04: strong signal.

5.2.1.7. Language/Extended Port Status

0x00 (former bit) 0x01 (latter bit)
Former bit: terminal extended port status
Latter bit: the current language of the terminal

Former Bit	
Latter Bit	0x01 Chinese 0x02 English

5.2.1.8. Information Serial Number

For details see Data Packet Format section 4.5.

5.2.1.9. Error Check

For details see Data Packet Format section 4.6.

5.2.1.10. Stop Bit

For details see Data Packet Format section 4.7.

5.2.2. Server Responds the Data Packet

Format		Length
Login Message Packet (18 Byte)	Start Bit	2
	Packet Length	1
	Protocol Number	1
	Information Serial Number	2
	Error Check	2
	Stop Bit	2

The response packet from the server to the terminal: the protocol number in the response packet is identical to the protocol number in the data packet sent by the terminal.
Example: 78 78 05 13 00 0F 00 8F 0D 0A

5.2.2.1. Start Bit

For details see Data Packet Format section 4.1.

5.2.2.2. Packet Length

For details see Data Packet Format section 4.2.

5.2.2.3. Protocol Number

For details see Data Packet Format section 4.3.

5.2.2.4. Information Serial Number

For details see Data Packet Format section 4.5.

5.2.2.5. Error Checking

For details see Data Packet Format section 4.6.

5.2.2.6. Stop Bit

For details see Data Packet Format section 4.7.

5.2.3. Data Example

6. Terminal sends example

7. 78 78 08 13 4B 04 03 00 01 00 11 06 1F 0D 0A

8. Explanation

0x78 0x78	0x08 0	0x13	0x4B 0x04 0x03	0x00 0x01	0x00 0x11	0x06 0x1F	0x0D 0x0A
Start Bit	Packet Length	Protocol Number	Information Content	Reserved Bit(Language)	Information Serial Number	Error Check	Stop Bit

Server replies example

78 78 05 13 00 11 F9 70 0D 0A

Explanation

0x78 0x78	0x05	0x13	0x00 0x11	0xF9 0x70	0x0D 0x0A
Start Bit	Packet Length	Protocol Number	Information Serial Number	Error Check	Stop Bit

5.3. Location Data Packet (Combined Information Package of GPS and LBS)(Ox22)

5.3.1. Terminal Sending Location Data Packet to Server

Format			Length(Byte)
Information Content	Start Bit		2
	Packet Length		1
	Protocol Number		1
	GPS	Date Time	6

	Information	Quantity of GPS information satellites	1
		Latitude	4
		Longitude	4
		Speed	1
		Course, Status	2
	LBS Information	MCC	2
		MNC	1
		LAC	2
		Cell ID	3
	ACC		1
	Data upload mode		1
	GPS real time upload		1
	Serial Number		2
	Error Check		2
	Stop Bit		2

EXAMPLE: 78 78 22 22 0F 0C 1D 02 33 05 C9 02 7A C8 18 0C 46 58 60 00 14 00 01 CC 00 28 7D 00 1F 71 00 00 01 00 08 20 86 0D 0A

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

Protocol number 0x22

5.3.1.4. Date Time

Format	Length(Byte)	Example
Year	1	0x0A
Month	1	0x03
Day	1	0x17

Hour	1	0x0F
Minute	1	0x32
Second	1	0x17

Example: 2010-03-23 15:50:23

Calculated as follows: 10(Decimal)=0A(Hexadecimal)
3 (Decimal)=03(Hexadecimal)
23(Decimal)=17(Hexadecimal)
15(Decimal)=0F(Hexadecimal)
50(Decimal)=32(Hexadecimal)
23(Decimal)=17(Hexadecimal)

Then the value is: 0x0A 0x03 0x17 0x0F 0x32 0x17

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 0xCB, it means the length of GPS information is 12 and the number of the positioning satellites is 11. (C = 12Bit Length , B = 11 satellites)

5.3.1.6. Latitude

4 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 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: 22º32.7658’=(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

4 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.

5.3.1.8. Speed

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

e.g. 0x00 represents 0 km/h.

0x10 represents 16km/h.
0xFF represents 255 km/h

5.3.1.9. Course & Status

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

BYTE_1	Bit7	0
	Bit6	0
	Bit5	GPS real-time/differential positioning
	Bit4	GPS has been 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	

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 0x15 0x4C, the corresponding binary is 00010101 01001100,

BYTE_1 Bit7 0
BYTE_1 Bit6 0
BYTE_1 Bit5 0 (real time GPS)
BYTE_1 Bit4 1 (GPS has been positioned)
BYTE_1 Bit3 0 (East Longitude)
BYTE_1 Bit2 1 (North Latitude)
BYTE_1 Bit1 0
BYTE_1 Bit0 1
BYTE_2 Bit7 0
BYTE_2 Bit6 1
BYTE_2 Bit5 0
BYTE_2 Bit4 0

→ Course 332° (0101001100 in Binary, or 332 in decimal)

BYTE_2 Bit3 1
BYTE_2 Bit2 1
BYTE_2 Bit1 0
BYTE_2 Bit0 0 ————

which means real-time GPS tracking is on. The location is at north latitude, east longitude and the course is 332°

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

The Value of Cell Tower ID (Cell ID) ranges from 0x000000 to 0xFFFFFF.

5.3.1.14.ACC

The stature of ACC: 00 means low ACC; 01 means high ACC.

5.3.1.15. GPS Data Upload Mode

0x00 upload by time interval
0x01 upload by distance

0x02 upload for corner correction
0x03 upload by ACC status change
0x04 Re-upload last valid locating point after shift from dynamic motion to static position.
0x05 Report last valid locating point after network reconnected.
0x06 Update the ephemeris to upload locating point.
0x07 Press to upload locating point
0x08 Power on to upload locating point
0x09 Power on and change locating point by LBS based station.

5.3.1.16.GPS real-time upload/re-upload

GPS real-time re- upload
0x00 GPS real-time upload
0x01 GPS data re-upload

5.3.1.17.Serial Number

For details see Data Packet Format section 4.5

5.3.1.18.Error Check

For details see Data Packet Format section 4.6.

5.3.1.19.Stop Bit

For details see Data Packet Format section 4.7.

5.4. Alarm Packet

5.4.1. Alarm data packet sent by terminal to server.

Format			Length (Byte)
Information Content	Start Bit		2
	Packet Length		1
	Protocol Number		1
	Date Time		6
	Alarm Information	Alarm Type	1
		Alarm Value	N
	Serial Number		2
	Error Check		2
	Stop Bit		2

Example: 78 78 0C 95 11 02 0A 03 32 00 80 00 01 82 24 0D 0A

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

95 protocol number

5.4.1.4. Date Time

Time of alarm data packet, UTC time

5.4.1.5. Alarm Type

Byte 1	0x80: vibration alarm
	0X87: over-speed alarm
	0x90: harsh acceleration alarm
	0x91,quick deceleration alarm
	0x92, sharp turn alarm

5.4.1.6. Alert Value

Byte N	When alarm type us 0x87, alert value is, speed of over-speed, km/g (Byte N is two bytes, byte show by short)

5.4.1.7. Alarm Value

It means threshold value of alarm. If no value, there is no need to write. Bit number is 0 and used as reserved character. For example if there is over-speed alarm, then the speed value will be sent to platform.

5.5. OBD Protocol Report（0x8E）

5.5.1. OBD packet sent by terminal to server

Format		Length(Byte)
Information content	Start Bit	2
	Packet Length	2
	Protocol Number	1
	Date and Time（UTC）	6
	OBD Information	N
	Numbers of GPS Satellite	1
	Latitude	4
	Longitude	4
	Speed	1
	Direction, Status	2
	Serial Number	2
	Error Check	2
	Stop Bit	2

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

Protocol number: 0x8E

5.5.1.4. Date and Time

For details see Data Packet Format section 5.2.1.4

5.5.1.5. OBD information(Hex) transfer one or more Hex packets.

Data Sequence		Content	Note
Type		0x21	Common information shown
Data Length		0x08	Length of Byte
Data	Data 0	Doors & Windows status	Bit7: Left front door status 0:OFF 1:ON
			Bit6: Right front door status 0:OFF 1:ON
			Bit5:Left behind door status 0:OFF 1:ON
			Bit4:Right behind door status 0:OFF 1:ON
			Bit3:Footbrake status 0:Loosen 1:Depress
			Bit2:Engine status 0:OFF 1:ON
			Bit1:Left turn light status 0:OFF 1:ON
			Bit0:Right turn light status 0:OFF 1:ON
	Data1	Car basic information	Bit7:Small light status 0:OFF 1:ON
			Bit6:Door lock status 0:Unlocked 1:Locked
			Bit5:Trunk status 0:OFF 1:ON
			Bit4:Hood status 0:OFF 1:ON
			Bit3:Handbrake status 0:Down 1:Up
			Bit2~1:Car anti-theft status 0:Disarm 1:Arm 2：Alarm
			Bit0:Ignition status 0:OFF 1:ON
	Data2	Average fuel	Average fuel consumption =（HEX_TO_DEC（Data2））/10 (L/100KM)

		consumption	
	Data3	Average speed	Average speed=HEX_TO_DEC（Data3） (KM/H)
	Data4	Engine speed low byte	Engine rotate speed=HEX_TO_DEC(Data5)*256 + HEX_TO_DEC（Data4）
	Data5	Engine speed high byte	
	Data6	Common remind	Bit7:Right behind window unclosed remind 0: Closed 1: Unclosed
			Bit6: Left behind window unclosed remind 0: Closed 1: Unclosed
			Bit5: Right front window unclosed remind 0: Closed 1: Unclosed
			Bit5: Left front window unclosed remind 0: Closed 1: Unclosed
			Bit3:No reserve start remind 0: No remind 1: Remind
			Bit2:Car unlock remind 0: Invalid 1: Unlocked
			Bit1:Vibration alarm status 0: Alarm 1: No alarm
			Bit0:Door unlocked remind 0: Locked 1: Unlocked
	Data7	Tyre pressure show	Bit7: Driving a seat belt 0: Not wear 1: Wear
			Bit6:Wiper status 0: Off 1: On
			Bit5:Hard breaking remind 0: Normal 1: Unnormal
			Bit4:Harsh acceleration remind 0: Normal 1: Unnormal
			Bit3: Right behind tyre pressure 0: Normal 1: Unnormal
			Bit2: Left behind tyre pressure 0: Normal 1: Unnormal
			Bit1: Right front tyre pressure 0: Normal 1: Unnormal
			Bit0: Left front tyre pressure 0: Normal 1: Unnormal
Error Check			Negated after all data is summed together
Type		0x22	Common information shown
Data length		0x08	Length of Byte
Data	Data0	Mileage low byte 1	Mileage=HEX_TO_DEC(Data2)*65536 +HEX_TO_DEC(Data1)*256 + HEX_TO_DEC（Data0） (Unit: KM)
	Data1	Mileage middle byte 2	
	Data2	Mileage high byte 1	
	Data3	Battery volt	Battery Volt=（HEX_TO_DEC（Data3））/10 (Unit: V)
	Data4	Breakdown status	0：No 1：Yes
	Data5	Hardware version	Bit7~Bit4 Software version
			Bit1：Car model 0：low-end car 1：high-end car
			Bit0：model 0：cloud start 1：cloud anti-theft

	Data6	Balance fuel volume low byte	Balance fuel volume=HEX_to_DEC(data7&data6)/10 (Unit: L)
	Data7	Balance fuel volume high byte	
Error Check			Negated after all data is summed together
Type		0x23	Common information shown
Data Length		0x08	Length of byte
Data	Data0	Temperature low byte	~~Temperature=（HEX_TO_DEC(Data1)*256 + HEX_TO_DEC（Data0））-1000）/10~~
	Data1	Temperature high byte	
	Data2	PM2.5 low byte	PM2.5=HEX_TO_DEC(Data3)*256 + HEX_TO_DEC（Data2）
	Data3	PM2.5 high byte	
	Data4	Renew mileage high byte	Renew mileage=HEX_TO_DEC(Data5)*256 + HEX_TO_DEC（Data4）
	Data5	Renew mileage low byte	
	Data6	steering wheel angle low byte	Steering wheel angle information is 2 byte,the highest bit in first byte means direction, 1 means right, 0 means left. The biggest angle is 40 degree. Low 7 bits in first byte means integer part second byte means fractional part; Example: 20.5 degree Left turn first byte 0x20 second byte 0x50 Right turn first byte 0xA0 second byte 0x50
	Data7	steering wheel angle high byte	
Error Check			Negated after all data is summed together

5.5.1.6. GPS information length, number of satellites

For details see section 5.3.1.5

5.5.1.7. Latitude

For details see section 5.3.1.6

5.5.1.8. Longitude

For details see section 5.3.1.7

5.5.1.9. Speed

For details see section 5.3.1.8

5.5.1.10. Course and Status

For details see section 5.3.1.9

5.5.1.11. Information serial number

For details see section 4.5

5.5.1.12. Error Check

For details see section 4.6

5.5.1.13. Stop Bit

For details see section 4.7

5.5.2. Server responses to data packet

Format		Length
Login packet（18 Byte）	Start bit	2
	Packet Length	1
	Protocol Number	1
	Information Serial number	2

	Check error	2
	Stop bit	2

Server responding packet to terminal: (the protocol number of responding packet is the same as the protocol number of data packet sent by terminal)

5.5.2.1. Start Bit

See section 4.1

5.5.2.2. Packet length

See section 4.2

5.5.2.3. Protocol number

See section 4.3

5.5.2.4. Information serial number

See section 4.5

5.5.2.5. Error check

See section 4.6

5.5.2.6. Stop Bit

See section 4.7

5.5.2.7. OBD Control（0x8F）

5.5.2.8. Server send OBD packet to terminal

Format		Length(Byte)
Packet content	Start bit	2
	Packet length	1

	Protocol number	1
	OBD controlling information	N
	Serial number	2
	Error check	2
	Stop bit	2

5.5.2.9. Start bit

Please find more details in Data Packet 4.1

5.5.2.10.Packet length

Please find more details in Data Packet 4.2

5.5.2.11.Protocol Number

Protocol Number: 0x8F

5.5.3. OBD Control Information（using hex coding）

Vehicle Control Packet 1

Data serial		Data content	Remark
Start Bit		0x82	Vehicle controlling information
Packet length		0x02	Packet length
Data	Data0	Control Commander	Bit7: Lift window 0: Void 1: lift window
			Bit6:Lock the car 0: Void 1: lock the car
			Bit5:request data 21 0: OFF 1: send 0x21 data
			Bit4:request data 22 0: OFF 1: send 0x22 data
			Bit3:Unlock the car 0: Void 1: unlock the car
			Bit2:Control indicator light 0: OFF 1: ON
			Bit1:Control speaker 0: OFF 1: ON
			Bit0: Control Alarm 0: OFF 1: ON
	Data1	Basic Information of Vehicle	Bit7:Unlock the trunk 0: Void 1: Open the trunk
			Bit6: reserved
			Bit5: reserved
			Bit4:Engine On/Engine OFF 0: Void 1: Valid
			Bit3: reserved
			Bit2: reserved

			Bit1: reserved
			Bit0: reserved
Error check			Negated after all data is summed together

Vehicle Control Packet 2

Data Serial		Data content	Remark
Start Bit		0xE0	Notice of setting vehicle model
Packet length		0x02	Packet length
Data	Data 0	Vehicle model	For more details about code of each vehicle model , please check the Appendix E
	Data 1	reserved	reserved
Error check			Negated after all data is summed together

Vehicle Control Packet 3

Data serial		Data content	remark
Start Bit		0xE1	Information of Vehicle controlling
Packet length		0x02	Packet length
Data	Data0	Functions settings	Bit7-Bit5: set the start time Engine off time =（Bit7- Bit5）×5 minutes When Bit7- Bit5=0, default time is 5mins WhenBit7- Bit5=, default time is 10mins …… When Bit7- Bit5=7, default time is 40 mins
			Bit4: set the start time. 0: Void time setting 1: Valid time setting
			Bit3 - Bit2: 0：High sensitivity 1 ：Moderate sensitivity 2：Low sensitivity 3：turn off the vibration alarm
			Bit1: set the sensitivity 0: Void sensitivity setting 1: Valid sensitivity setting
	Data1	Functions settings	Bit7:reserved
			Bit6: reserved
			Bit5: reserved
			Bit4: reserved
			Bit3: reserved
			Bit2: reserved
			Bit1:set auto-lock car 0: close 1: open

			Bit0:auto-lock car settings 0: void car auto-locking setting 1: Valid car auto-locking setting
Check error			Negated after all data is summed together

Example：

Command of lifting window: 82 02 80 00 FB

Command of locking the car: 82 02 40 00 3B

Command of reading Packet 21: 82 02 20 00 5B

Command of reading Packet 2: 82 02 10 00 6B

Command of locking & lifting window: 82 02 C0 00 BB

Command of unlocking the car: 82 02 08 00 74

Command of controlling indicator light: 82 02 04 00 77

Command of controlling speaker: 82 02 02 00 79

Command of controlling alarm: 82 02 01 00 7A

Command of opening the trunk: 82 02 00 80 FB

Remark: To control the light and speaker together, please send the command of controlling alarm.

5.5.3.1. Information Serial Number

See Detail at Data Packet Format4.5

5.5.3.2. Error Check

See Detail at Data Packet Format4.6

5.5.3.3. Stop Bit

See Detail at Data Packet Form at 4.7

5.5.4. Terminal Responds Data Packet

Format		Length(Byte)
Information Detail	Start Bit	2
	Packet Length	2
	Protocol Number	1
	Date & Time（UTC）	6

	OBD Information	N
	GPS Satellite Quantity	1
	Latitude	4
	Longitude	4
	Speed	1
	Course & Status	2
	Serial Number	2
	Error Check	2
	Stop Bit	2

The server respond packet to the terminal:（Server respond packet protocol is same as the terminal packet protocol)

5.5.4.1. Start Bit

See Detail at Data Packet Format 4.1

5.5.4.2. Packet Length

See Detail at Data Packet Format 4.2

5.5.4.3. Protocol Number

Protocol Number:0x8F

5.5.4.4. Date & Time

See detail at Location Data Packet 5.3.1.4

5.5.5. OBD information respond

（hexadecimal code）transmit one or several hexadecimal data packet

FF：Command execution success F0:Check Sum Error F3:Do not support this command FC:Busy，command execution failure

The server respond data packet to terminal:(Data packet protocol from server same as Data packet protocol from terminal）

5.5.5.1. Start Bit

See Detail at Data Packet Format 4.1

5.5.5.2. Packet Length

See Detail at Data Packet Format 4.2

5.5.5.3. Protocol Number

See Detail at Data Packet Format 4.3

5.5.5.4. Information Serial Number

See Detail at Data Packet Format 4.5

5.5.5.5. Error-Checking

See Detail at Data Packet Format 4.6

5.5.5.6. Stop Bit

See Detail at Data Packet Format 4.7

5.6. Network Time Checking Packet（0x8A）

5.6.1. Terminal Transmit Data Packet to Terminal

Format		Length
Llogin Info Packet（18 Byte）	Start Bit	2
	Packet Length	1
	Protocol Number	1
	Info Serial Number	2
	Error Check	2
	Stop Bit	2

Example：78 78 05 8A 00 06 88 29 0D 0A

5.6.1.1. Start Bit

See Detail at Data Packet Format4.1

5.6.1.2. Packet Length

See Detail at Data Packet Format4.2

5.6.1.3. Protocol Number

Protocol Number：0x8A

5.6.1.4. Information Serial Number

See Detail at Data Packet Format4.5

5.6.1.5. Error Check

See Detail at Data Packet Format4.6

5.6.1.6. Stop Bit

See Detail at Data Packet Format4.7

5.6.2. Server Responds Data Packet

Format		Length(Byte)
Information Detail	Start Bit	2
	Packet Length	1
	Protocol Number	1
	Time & Date	6
	Serial Number	2
	Error Check	2
	Stop Bit	2

Example：78 78 0B 8A 0F 0C 1D 00 00 15 00 06 F0 86 0D 0A

5.6.2.1. Start Bit

See Detail at Data Packet Format4.1

5.6.2.2. Packet Length

See Detail at Data Packet Format4.2

5.6.2.3. Protocol Number

Protocol Number：0x8A

5.6.2.4. Time & Date

Format	Length(Byte)	e.g
Year	1	0x0A
Month	1	0x03
Date	1	0x17
Hour	1	0x0F
Minute	1	0x32
Second	1	0x17

E.g：15:50:23 March 23th 2010

Computing Method :

10(Decimalism)=0A（Hexadecimal）

3 (Decimalism)=03（Hexadecimal）

23(Decimalism)=17（Hexadecimal）

15(Decimalism)=0F（Hexadecimal）

50(Decimalism)=32（Hexadecimal）

23(Decimalism)=17（Hexadecimal）

Then Value:0x0A 0x03 0x17 0x0F 0x32 0x17

5.6.2.5. Information Serial Number

See Detail at Data Packet Format 4.5

5.6.2.6. Error Check4

See Detail at Data Packet Format 4.6

5.6.2.7. Stop Bit

See Detail at Data Packet Format 4.7

~~5.7. LBS Expand Information Packet(0X28)~~

~~Format~~			~~Length(Byte)~~
~~Information Detail~~	~~Time & Date（UTC）~~		~~6~~
	~~LBS Info~~	~~MCC~~	~~2~~
		~~MNC~~	~~1~~
		~~LAC~~	~~2~~
		~~CI~~	~~2~~
		~~RSSI~~	~~1~~
		~~NLAC1~~	~~2~~
		~~NCI1~~	~~2~~
		~~NRSSI1~~	~~1~~
		~~NLAC2~~	~~2~~
		~~NCI2~~	~~2~~
		~~NRSSI2~~	~~1~~
		~~NLAC3~~	~~2~~
		~~NCI3~~	~~2~~
		~~NRSSI3~~	~~1~~
		~~NLAC4~~	~~2~~
		~~NCI4~~	~~2~~
		~~NRSSI4~~	~~1~~
		~~NLAC5~~	~~2~~
		~~NCI5~~	~~2~~
		~~NRSSI5~~	~~1~~
		~~NLAC6~~	~~2~~
		~~NCI6~~	~~2~~
		~~NRSSI6~~	~~1~~
	~~Reserved Expand Bit~~		~~N~~

~~Example78 78 3B 28 10 01 0D 02 02 02 01 CC 00 28 7D 00 1F 71 3E 28 7D 00 1F 72 31 28 7D 00 1E 23 2D 28 7D 00 1F 40 18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF 00 02 00 05 B1 4B 0D 0A~~

5.5.2 Terminal Transmit Data Packet to Server

Time & Date

Same as above description

MCC

Same as above description

MNC

Same as above description

LAC

Same as above description

CI（Cell ID）

Area Heading Code,Range 0x0000 ~ 0xFFFF.

RSSI（Received Signal Strength Indicator）

Signal Strength of Area, Range 0x00~0xFF, 0x00 weak, 0xFF strong.

NLAC1~6

Location Area Code of Adjacent Area, totally 6

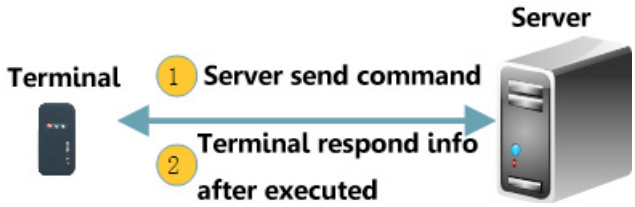
NCI1~6（Neighbouring Cell ID）

Heading Code of Adjacent Area,same as 6 NLAC.

NRSSI1~6（Near Cell ID Signal Strength）

Signal Strength of Adjacent Area ,same as 6 NLAC.

6. Server Sends Data Packet to Terminal (GPS Location Server Protocol)(0x80)



6.1. Control Command Sent by Server

Format	Length(Byte)
Start Bit	2
Packet Length	1
Protocol Number	1

Information Detail	Command Length	1
	Server Zone Bit	4
	Command Content	M
	Language	2
Info Serial Number		2
Error Check		2
Stop Bit		2

Example：78 78 0E 80 08 00 00 00 00 73 6F 73 23 00 01 6D 6A 0D 0A

6.1.1 Start Bit

See Detail at Data Packet Format4.1

6.1.2 Packet Length

See Detail at Data Packet Format4.2

6.1.3 Protocol Number

Terminal send Protocol Number by：0x80

6.1.4 Command Length

Server Zone Bit + Command Length
E.g: Byte Length as a unit, OxOA, this command occupies 10 bytes

6.1.5 Server Zone Bit

For server detecting, terminal respond packet with the received original binary sample

6.1.6 Command Content

Express as string ASC II:

HOTSPOT,ON --------- Open WIFI hotspot. The account is the IMEI; Passwords are the last eight number of IMEI.

PICTURE,OUT ------ get one picture from the outward camera

WIFICON,C8,88888888

Settings for WIFICON wifi connecting: ***WIFICON,<A>,<B>***

A = WiFi name;

B = WiFi passwords;

Device reply formate:

OK

SPEED,ON,10,80

Overspeed remind setting SPEED,<A>,<B>,<C>

A=ON/OFF, open or close over speed alarm, default: OFF

B=5～600 (second), time interval, default:10 (second) ()

C=1～255(km/h), speed threshold range, default: 80(km/h);

Remark: over-speed alarm triggrting mechanisam: device detects the speed is more than C for B period, then, alarm is triggered

FLIST ------ File list, device receive request and reply OK, then transfer file list to WEB by HTTP

HFILE, 2017-08-01 15:52:24

HFILE History file division ***HFILE,<A>***

A, time, server send request for the demanded division file's time

RTMP,ON,IN

Settings to start RTMP, ***RTMP ,<A>,<B>***

A=ON; RTMP starts; default: ON

B=IN or OUT; IN for inward camera，OUTfor outward camera, default is IN

Device reply format:

OK

RTMP,OFF -------- close RTMP

RLOGIN,jimi,88888888

RLOGIN Set the login info，***RLOGIN, <A>,<B>***

A=jimi; account; default jimi；

B=88888888; passwords; default 88888888；

Device reply format

OK

UPDATE, http://www.test.com/update.zip
UPDATE,<A> update firmware
<A>= download link; for example, http://www.test.com/update.zip

TCPSERVICE,gpsdev.hditcloud.com
Socket Server setting: ***TCPSERVICE ,<A>***
A is socket server address
Device reply format:
OK

HERVICE,1,gpsdev.hditcloud.com,21100
Socket Server setting: ***HSERVICE ,<A>,<B>***
A= 0 or 1; 1 means domain, 0 means IP
B is socket server address
Device reply format:
OK

RSERVICE,gpsdev.hditcloud.com
RSERVICE RTMP setting: ***RSERVICE ,<A>***
Ais RTMP server address
Device reply format:
OK

APNSET,<name>,<apn>,<mcc>,<mnc>,<type>,<proxy>,<port>,<user>,<server>,<password>,
<mmsc>,<mmsproxy>,<mmsport>,<numeric>
name: default is apn byten numbers
numeric: mcc+mnc, mcc3cbytes, mnc 2bytes;
If mnc is less than 2 bytes, please use 0 as alternative byte; if there is no value behind, then no need to input.
If there are values some part and some part no value, use space as alternative for the non-value part.
Remark: the command is complicated, please follow the rules strictly.
e.g.:
apn#666666#unim2m.gzm2mapn#unim2m.gzm2mapn#460#06

e.g.:

apn#666666# #unim2m.njm2mapn#460#06

SENALM	Vibration alarm command setting	Server send command format; SENALM,<A>,<B>,<M># A=ON/OFF; default: ON SENALM, ON # Open vibration alarm SENALM,OFF# Close vibration alarm B=0,1; Video upload or not; Default: not upload video, value is 0; M=1,2,3 sensitivity. 1 means low, 2 means middle, 3 means high. Default is 2. Device reply format: OK
RAPIDACC	Harsh acceleration alarm command setting	Server send command format; RAPIDACC,<A>,<B>,<M># A=ON/OFF; default: ON B=0,1; Video upload or not; Default: not upload video, value is 0; M=1,2,3 sensitivity. 1 means low, 2 means middle, 3 means high. Default is 2. RAPIDACC,OFF# Close harsh acceleration alarm RAPIDACC,ON# Open harsh acceleration Device reply format: OK

RAPIDDEC	Harsh brake alarm command setting	Server send command format; RAPIDDEC,<A>,<B>,<M># A=ON/OFF; Default: ON B=0,1; Video upload or not; Default: not upload video, value is 0; M=1,2,3 sensitivity. 1 means low, 2 means middle, 3 means high. Default is 2. RAPIDDEC,OFF# Close harsh brake alarm RAPIDDEC,ON# Open harsh brake alarm Device reply format: OK
RAPIDTURN	Sharp turn alarm command setting	Server send command format; RAPIDTURN,<A>,<B>,<M># A=ON/OFF; Default: ON B=0,1; Video upload or not; Default: not upload video, value is 0; M=1,2,3 sensitivity. 1 means low, 2 means middle, 3 means high. Default is 2. RAPIDTURN,OFF# Close sharp turn alarm RAPIDTURN,ON# Open sharp turn alarm Device reply format: OK
MOVING	Moving alarm	Server send command format;

	command setting	MOVING,<A>[,R]# A=ON/OFF; Default: OFF R=100～1000; Moving radius, unite is meter, default is 300 MOVING,OFF# Close moving alarm MOVING# Inquiry status of current moving alarm, radius, alarm type, drift location. Device reply format: OK

6.1.7 Language

Terminal Current Language.
Chinese:0x00 0x01
English:0x00 0x02

6.1.8 Information Serial Number

See Detail at Data Packet Format 4.5

6.1.9 Error Check

See Detail at Data Packet Format 4.6

6.1.10 Stop Bit

See Detail at Data Packet Format 4.7

6.2. Terminal Return Data(Terminal Respond)

Format		Length (Byte)
Start Bit		2
Packet Length		2
Protocol Number		1
Information Detail	Server Zone Bit	4
	Content Code	1

	Content	M
Info Serial Number		2
Error Correction		2
End Bit		2

Example：79 79 00 9D 21 00 00 00 00 01 42 61 74 74 65 72 79 3A 34 2E 31 36 56 2C 4E 4F 52 4D 41 4C 3B 20 47 50 52 53 3A 4C 69 6E 6B 20 55 70 3B 20 47 53 4D 20 53 69 67 6E 61 6C 20 4C 65 76 65 6C 3A 53 74 72 6F 6E 67 3B 20 47 50 53 3A 53 65 61 72 63 68 69 6E 67 20 73 61 74 65 6C 6C 69 74 65 2C 20 53 56 53 20 55 73 65 64 20 69 6E 20 66 69 78 3A 30 28 30 29 2C 20 47 50 53 20 53 69 67 6E 61 6C 20 4C 65 76 65 6C 3A 3B 20 41 43 43 3A 4F 46 46 3B 20 44 65 66 65 6E 73 65 3A 4F 46 46 00 2E 26 DF 0D 0A

6.2.1 Start Bit

Fixed Value 0x79 0x79

6.2.2 Packet Length

Occupy 2 bytes

6.2.3 Protocol Number

Use 0x21

6.2.4 Server Zone Bit

For server detecting, terminal respond packet with the received original binary sample

6.2.5 Content Code

0x01 ASC IICode
0x02 UTF16-BE Code

6.2.6 Content

The data needed to transmit
For transmitting image and video, this protocol is unavailable to upload image nor video, after terminal got command respond from server, terminal should link to image server to upload packet. This respond content fixed as ASCII code: picture(upload image) or video(upload video).

6.2.7 Information Serial Number

See Detail at Data Packet Format4.5

6.2.8 Error Check

See Detail at Data Packet Format4.6

6.2.9 Stop Bit

See Detail at Data Packet Format4.7

Instruction of uploading video file

Code example for uploading:

public static String postFile(String httpUrl,String fileId,String filePath, String format,String imei)

httpUrl for the uploading server address；

filePath for the path of uploading file;

"filetime = " + conn.getRequestProperty("filetime")
+ " filesize = " + conn.getRequestProperty("filesize")
+ " imei = " + conn.getRequestProperty("imei")
+ " fileext = " + conn.getRequestProperty("fileext")
+ " fileMD5 = " + conn.getRequestProperty("fileMD5")
+ "fileId= " + conn.getRequestProperty("fileId")

filetime for time of uploading file；

filesize for the size of uploading file；

imei for device's IMEI numbers；

fileext for formate of the uploading file

fileMD5 for checking the file integrality

fileId for command ID sent by server

7. Image/Video upload protocol(Image server)—Only for sold devices

Image Upload

Interface Address	http://resource.car-matrix.com/upload?type=1or 2 (type=1:Image ; type=2:Video)
Request Method	post
Respond Format	Json [Type of Data]
Input Parameter	File file stream method(http head description)

Output	{
	"code":0, --result code,0:succeed 1:upload failure
	"type":1 ,
	"md5":"" -- 16 bit image MD5 code asset server switch
	}

8. Image Successfully Reported to Platform[TCP]（GPS Location Server）

Interface Address	Tracker address 120.237.91.74:21100 (for test)
Request Mode	Tcp
Return Format	No, server reply is user-defined.
Input Parameter	Byte 8 imei(device IMEI or other unique identify number) Byte 8 file md5(defined by 16 bits MD5, namely 8 bytes) Byte 1 file type(1:image 2:recording)
Output	No

Format		Length (Byte)
Resource report information packet（30 Byte）	Start Bit	2
	Packet Length	1
	Protocol Number	1

	Server Flag Bit	4
	Uploading Result	1
	File MD5	8
	File Type	1
	File Generation Time	6
	Information Serial Number	2
	Error Check	2
	Stop Bit	2

8.1 Start Bit

For details see Data Packet Format section 4.1.

8.2 Packet Length

For details see Data Packet Format section 4.2.

8.3 Protocol Number

Protocol number：0x30

~~8.4 Terminal ID~~

~~Example：123456789012345，~~
~~Then terminal ID is：0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45~~

8.4 Server Flag Bit

Server flag bit, 0X80 protocol, the server flag bit of image command.

8.5 Uploading Result

Success ：0x01
Failure：0x02（Fail to upload，file MD5，file type，byte occupied by file generation time，fill all 0x00）

8.6 MD5

Information of MD5 (8 bytes) returned by image server.

8.7 File Type

0x01: Photo
0x02: Video

8.8 File Generation Time

Actual shooting time of device

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

Example: 2010-03-23 15:30:23
Calculated as follows: 10(Decimal)=0A(Hexadecimal)
3 (Decimal)=03(Hexadecimal)
23(Decimal)=17(Hexadecimal)
15(Decimal)=0F(Hexadecimal)
50(Decimal)=32(Hexadecimal)
23(Decimal)=17(Hexadecimal)
Then the value is: 0x0A 0x03 0x17 0x0F 0x32 0x17

8.9 Information Serial Number

For details see Data Packet Format section 4.5.

8.10 Error Check

For details see Data Packet Format section 4.6.

8.11 Stop Bit

For details see Data Packet Format section 4.7.

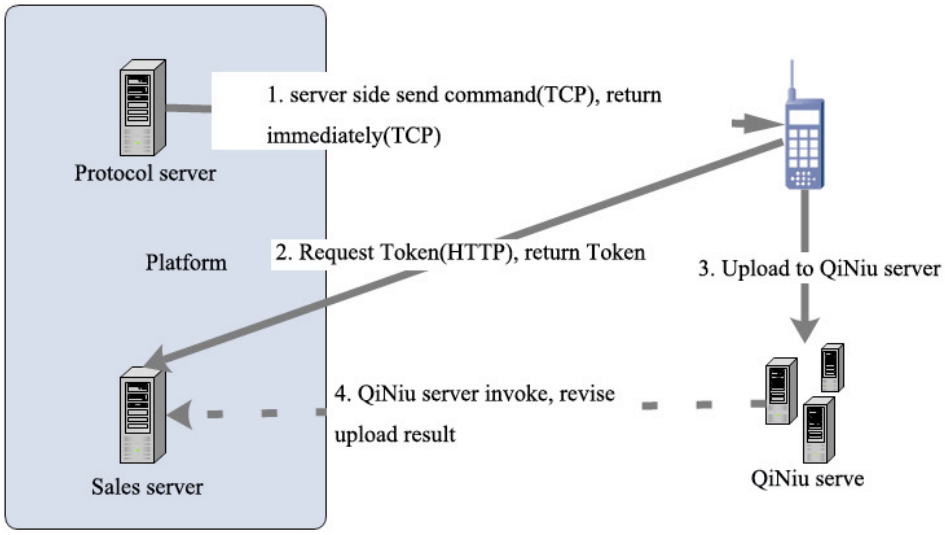
Note: To save space and resource, only 30 images and 20 videos can be saved. If surpassed, fresh ones will cover the previous.

9. Remote Image and Video Protocol

Note: QiNiu server is a brand of famous Chinese cloud server.

After image server transferred to QiNiu server, the process is as below. First three steps involved:

1) Server sends command and terminal unconditionally responds immediately after receiving command.
2) Terminal requests to upload Token. Token includes access invoking etc. It must send request when upload every time.
3) Using the obtained Token to upload data to QiNiu server
4) QiNiu server will upload result, and notice sales server (including resources date, upload result etc.)



9.1 Server send command, terminal unconditionally respond immediately after receiving command.

1). Server send command protocol 80(see 6.1 Server Sending Control Command)

2). Terminal unconditionally and immediately responds protocol 21 (see 6.2 Terminal Return Data（Terminal Response）

3). Command server IP and Port: gpsdev.car-matrix.com : 21100

9.2 Terminal request upload Token

1). Request address http://www.car-matrix.com/common/getQiLiuToken?imei
IMEI= Device IMEI
2). Response:
{
Token:"content string",//token
expiresSecond: 86400,//expires time
code: 0
msg: "QiNiu teken generated successfully"
}
3) Parameter is device IMEI

9.3 Using the obtained Token to upload data via QiNiu SDK

1) Terminal and sales server appoint unique identified resource key as IMEI command serial No. time, for example: device (IMEI 358740054268379) send command serial No. 14214 at 7:02:46 on 15th. Mar. 2016, then terminal upload key is 358740054268379_14214_2016_03_15_07_02_46
2) Please note command serial No is generated by server and can't repeat.
3) Upload example code

```
UploadManager uploadManager = new UploadManager();
Response        response = uploader.put("d:/tmp/max.png","resource name Key", token);
```

9.4 QiNiu server will upload result, and notice sales server

Invoking address is auto written in by sales server send token, and telecommunication by QiNiu server and sales server, terminal will not take part. Just using sales server and terminal appointed key to upload.
1). Resource invoking address: http://www.car-matrix.com/common/qiliuCallbackApi?
2). Mircograph incoking address: http://www.car-matrix.com/common/ qiliuVideoCallbackApi?

10. Data Transfer Common Packet (0x94)

Terminal send data packet to server

Format		Length (Byte)
Data packet (16 Bytes)	Start Bit	2
	Packet length	2
	Protocol number	1

	File Type	1
	Information Content	N
	Serial Number	2
	Error Check	2
	Stop Bit	2

Example: 79 79 00 7F 94 04 41 4C 4D 31 3D 43 34 3B 41 4C 4D 32 3D 43 43 3B 41 4C 4D 33 3D 34 43 3B 53 54 41 31 3D 43 30 3B 44 59 44 3D 30 31 3B 53 4F 53 3D 2C 2C 3B 43 45 4E 54 45 52 3D 3B 46 45 4E 43 45 3D 46 65 6E 63 65 2C 4F 4E 2C 30 2C 32 33 2E 31 31 31 38 30 39 2C 31 31 34 2E 34 30 39 32 36 34 2C 34 30 30 2C 49 4E 20 6F 72 20 4F 55 54 2C 30 3B 4D 49 46 49 3D 4D 49 46 49 2C 4F 46 46 00 0A 06 1E 0D 0A

10.1 Start Bit

0x79 0x79

10.2 Packet Length

For details see Data Packet Format section 4.2.
Packet length=protocol number+ information type + information content +information serial number + error check

10.3 Protocol Number

For details see Data Packet Format section 4.3

10.4 File Type

File type stands for the type of uploading file
……
07 irror terminal report UUID

10.5 Information Content

When type is 07, this bit transfer mirror terminal report UUID. By ASCII

10.6 Information Serial Number

For details see Data Packet Format section 4.5

10.7 Error Check

For details see Data Packet Format section 4.6

10.8 Stop Bit

For details see Data Packet Format section 4.7

11. LBS Multiple Cell Tower ID Extension Packet(0x28)

Introduction of LBS Multiple Cell Tower ID extension packet:

The packet to transmit location packet when the terminal failed to locate

a). Terminal send LBS Multiple Cell Tower ID packet

		Length	Descrition
Start Bit		2	0x78 0x78
Packet Length		1	Length=Protocol number +Packet contents + Serial number of packet+ error check
Protocol Length		1	0x28
Information Content	Date（UTC）	6	Year（1byte）Month（1byte）Date（1byte）Hour（1byte）Minute（1byte）Second（1byte）（transform to Decimal ）
	MCC	2	Mobile Country Code
	MNC	1	Mobile Network Code(MNC)
	LAC	2	Mobile Network Code(MNC)
	CI	3	Cell Tower ID(Cell ID)
	RSSI	1	Signal intensity of the area， Range :0x00～0xFF， 0x00 the weakest signal，0xFF the strongest signal.
	NLAC1	2	The same as LAC
	NCI1	3	The same as CI
	NRSSI1	1	The same as RSSI
	NLAC2	2	The same as LAC
	NCI2	3	The same as CI
	NRSSI2	1	The same as RSSI
	NLAC3	2	The same as LAC
	NCI3	3	The same as CI
	NRSSI3	1	The same as RSSI
	NLAC4	2	The same as LAC

	NCI4	3	The same as CI
	NRSSI4	1	The same as RSSI
	NLAC5	2	The same as LAC
	NCI5	3	The same as CI
	NRSSI5	1	The same as RSSI
	NLAC6	2	The same as LAC
	NCI6	3	The same as CI
	NRSSI6	1	The same as RSSI
	Timing Advance	1	It is the difference value between Time A and Time B. Time A = the real time of the Cell Tower signal reaching the Cell Tower ID Time B = Assuming the distance between the Cell Tower and Base Station is 0, the time of Cell Tower signal reaching Cell Tower ID.
	Language	2	0x00 0x01 Chinese 0x00 0x02 English
Information Serial Number		2	The serial number will automatically add 1 each time sending information since the device is powered up
Error Check		2	The CRC-ITU value between "the packet length" and "Information Serial Number". Please delete this packet if there is CRC error of information calculation of the receiver. (check the algorithm in appendix A)
Stop Bit		2	Fixed value: 0x0D 0x0A

Example: 78 78 3B 28 10 01 0D 02 02 02 01 CC 00 28 7D 00 1F 71 3E 28 7D 00 1F 72 31 28 7D 00 1E 23 2D 28 7D 00 1F 40 18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF 00 02 00 05 B1 4B 0D 0A

b). The server response of LBS Multiple Cell Tower ID

LBS Multiple Cell Tower ID Packet server no need to response

12. WIFI Data Packet (0x2C)

About WIFI Information Packet:

To transmit the WIFI data packet which is received by terminal

a). The terminal send WIFI data packet

		Length	Description
Start bit		2	0x78 0x78
Packet length		1	Length =Protocol number +information content+ Information serial number +Error check
Protocol Number		1	0x2C
Information Content	Date and time （UTC）	6	Year（1byte）Month（1byte）Date（1byte）Hour（1byte）Minute（1byte）second（1byte）（transformed to Decimal）

	MCC	2	Mobile Country Code
	MNC	1	Mobile Network Code(MNC)
	LAC	2	Mobile Network Code(MNC)
	CI	3	Cell Tower ID(Cell ID)
	RSSI	1	Signal intensity of the area， Range :0x00～0xFF， 0x00 the weakest signal，0xFF the strongest signal.
	NLAC1	2	The same as above LAC
	NCI1	3	The same as above CI
	NRSSI1	1	The same as above RSSI
	NLAC2	2	The same as above LAC
	NCI2	3	The same as above CI
	NRSSI2	1	The same as above RSSI
	WIFI MAC1	6	The MAC to receive the 1WIFIsignal
	WIFI intensity value 1	1	The signal intensity of 1WIFI signal
	WIFI MAC2	6	The same as above
	WIFI intensity value 2	1	The same as above
	WIFI MAC3	6	The same as above
	WIFI intensity value 3	1	The same as above
	WIFI MAC4	6	The same as above
	WIFI intensity value 4	1	The same as above
	WIFI MAC5	6	The same as above
	WIFI intensity value 5	1	The same as above
Information Serial Number		2	The serial number will automatically add 1 each time sending information since the device is powered up
Error Check		2	The CRC-ITU value between “the packet length” and “Information Serial Number”. Please delete this packet if there is CRC error of information calculation of the receiver. (check the algorithm in Appendix A)
Stop Bit		2	Fixed value: 0x0D 0x0A

Example： 78 78 4D 2C 10 01 0D 02 02 02 01 CC 00 28 7D 00 1F 71 3E 28 7D 00 1F 72 31 28 7D 00 1E 23 2D 11 11 11 11 11 11 11 11 05 22 22 22 22 22 22 22 22 05 33 33 33 33 33 33 33 33 04 44 44 44 44 44 44 44 44 03 55 55 55 55 55 55 55 55 02 00 05 08 7B 0D 0A

a). The response to WIFI data packet from Server

Server no need to response to WIFI data packet

13. Appendix A

Code Fragment of the CRC-ITU Lookup Table Algorithm Implemented Based on C Language

Code fragment of the CRC-ITU lookup table algorithm implemented based on C language is as follow:

```
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
```

}

14. Appendix B

Complete Format of The Information Package

A. The packet sent to server by terminal

Login Message Packet（18 Byte）						
Start Bit	Packet length	Protocol number	Terminal ID	Information Serial number	Check error	Stop bit
2	1	1	8	2	2	2

GPS packet（26+N Byte）												
			Information Content									
				GPS information								
Start Bit	Packet length	Protocol Number	Date and Time	Length of GPS information, quantity of positioning satellites	Latitude	Longitude	Sped	Course, Status	Reserved extended bit	Information serial number	Check Bit	S
2	1	1	6	1	4	4	1	2	N	2	2	2

LBS information package (23+N Byte)											
			Information Content								
				LBS Information							
Start Bit	Packet length	Protocol Number	Date Time	MCC	MNC	LAC	Cell ID	Reserved extended bit	Information serial number	check bit	stop bit
2	1	1	6	2	1	2	3	N	2	2	2

LBS complete information package (42+N Byte)																								
			Information Content																					
				LBS Information																				
Start Bit	Packet length	Protocol Number	Date Time	MCC	MNC	LAC	MCI	MCISS	NCI1	NCISS1	NCI2	NCISS2	NCI3	NCISS3	NCI4	NCISS4	NCI5	NCISS5	NCI6	NCISS6	Reserved extended bit	Information serial number	check bit	stop bit
2	1	1	6	2	1	2	2	1	2	1	2	1	2	1	2	1	2	1	2	1	N	2	2	2

GPS、LBS information package (34+M+N Byte)																	
Start Bit	Packet length	Protocol Number	Information Content												Information serial number	check bit	stop bit
			Date Time	GPS Information						LBS Information				Reserved and extended			
				Length of GPS information, quantity of positioning satellites	Latitude	Longitude	Speed	Course, Status	Reserved extended bit	MCC	MNC	LAC	Cell ID				
2	1	1	6	1	4	4	1	2	M	2	1	2	3	M	2	2	2

Status Packet(13+N Byte)									
Start Bit	Packet Length	Protocol Number	Information Content				Information Serial Number	Check Bit	Stop Bit
			Terminal Information Content	Voltage Level	GSM Signal Strength Level	Reserved and Extended Bit (language)			
2	1	1	1	1	1	2	2	2	2

SNR information of satellite (11+M+N Byte)								
Start Bit	Packet Length	Protocol Number	Information Content			Information Serial Number	Check Bit	Stop Bit
			Quantity of positioning satellites	SNR of Satellite	Reserved and Extended Bit			
				1 2 3 …… n				
2	1	1	1	M	N	2	2	2

terminal responds to the command sent by server (15+M+N Byte)									
Start Bit	Packet Length	Protocol Number	String Content				Information Serial Number	Check Bit	Stop Bit
			Length of Command	Server Flag Bit	Command Content	Reserved and Extended Bit (language)			
2	1	1	1	4	M	2	2	2	2

GPS, LBS, Status Information Package (40+M+N+L Byte)																							
Start Bit	Packet Length	Protocol Number	Information Content																	Reserved and Extended Bit (language)	Information Serial Number	Check Bit	Stop Bit
			Date Time	GPS Information						LBS Information							Status Information						
				Length of GPS information, quantity of positioning satellites	Latitude	Longitude	Speed	Course, Status	Reserved and Extended Bit	LBS Length	MCC	MNC	LAC	Cell ID	Reserved and Extended Bit	Terminal Information Content	Voltage Level	GSM Signal Strength Level					
2	1	1	6	1	4	4	1	2	M	1	2	1	2	3	N	1	1	1	2	2	2	2	

B. Response of Server after receiving Status Packet from Terminal

Response of Server after receiving Status Packet from Terminal (10 Bytes)					
Start Bit	Packet Length	Protocol Number	Information Serial Number	Check Bit	Stop Bit
2	1	1	2	2	2

Command Packet Sent by Server to Terminal (15+M+N Byte)									
Start Bit	Packet Length	Protocol Number	Information Content				Information Serial Number	Check Bit	Stop Bit
			Length of Command	Server Flag Bit	Command Content	Reserved extended bit			
2	1	1	1	4	M	N	2	2	2

15. Appendix C

OBD Vehicle Model Code

0x00 No vehicle model set ， 0x01 General Motors ， 0x02 Old A6L ， 0x03 Peugeot Citroen ， 0x04 Audi A4L ，0x05 Changan，0x06 Mercedes ，0x07 BMW ，0x08 General Motors ，0x09 Hyundai ，0x0A Old Fox，0x0B Nissan series ， 0x0C Old Mercedes-Benz，0x0D Opel，0x0E 2012 Audi A6L ， 0x0F Toyota，0x10 Mazda ， 0x11 BYD ， 0x12 New Mondeo ， 0x13 High-end Honda ， 0x14 Chery Motors ， 0x15 Geely Automobile ， 0x16 Mitsubishi V5 ， 0x17 Roewe & Morris Garages ， 0x18 Sega ， 0x19 Fiat ， 0x1A new Mazda 3 ，0x1B Mondeo ，0x1C Low-end Honda，0x1D 2012 Fox，0x1E BesTurn，0x1F Low configuration Bora，0x20 New CRV ， 0x21 Great Wall Motors ， 0x22 Trumpchi，0x23 Mitsubishi Pajero ， 0x24 Subarus ， 0x25 Golf 7,Audi new A3，New Octavia ，0x26 Roewe 350、Morris Garages 3、W5、Huatai ，0x27 Mitsubishi ASX，0x28 ZotyeT600 ，0x29 Beijing Auto ，0x30 Luxgen，0x31 Ford Ecosport、WINBO、Sharp world Ford ， 0x32 SAAB ， 0x33 Fengshen ， 0x34 Changan Suzuki

16. Appendix D:

Tracker Function:

1、 Push message: default is ON

2 、Push message OFF: From the setting, after the Location is closed, turn off Push Message, the terminal will not send heartbeat packet and GPS packet.

3、 Push message ON:

3.1 the status of 5mins after the device is powered on, or 5mins before G-SENSOR detects vibration or 5minsbeforestatic status: if the network is disconnected, Push message is running on, but the device won't send heartbeat packet and GPS packet, and it will check whether GPRS Connection is connected or or not every 3 mins; If the GPRS Connection is connected, the device will send heartbeat packet every 3 mins, GPS located successfully, the device will send GPS packet every 10s. If the GPS failed to locate, the device won't send GPS data.

3.2 If the car is stopped for more than 5 mins:

If the GPRS connection is disconnected, push message is running on, but the device won't send heartbeat packet and GPS packet, and it will check whether GPRS network is connected or not every 3 mins; If the GPRS connection is connected, the device will send heartbeat packet every 3 mins, GPS located successfully, the device will send GPS packet every 10s. If the GPS failed to locate, the device won't send GPS data.

Remark：To change the running status of car, the interval of successful Located GPS: use SMS commander or engineering commander: *#*#89#*#*

4. If no response from server after the terminal sent 5 times of heart beat packet, please turn off and reboot the GPRS connection. Reboot 1 time at most per day. The method as follows:

If no response from server after the terminal sent 5 times of heart beat packet

a). if the defined the status value is 0, then turn off and reboot the GPRS connection, and set the status value to 1, and record the current date

b). if the defined the status value is 1, check whether the recorded date and current date is the same or not. If these two dates are different, then turn off and reboot the GPRS connection, and set the status value to 1, and record the current date. If these two dates are the same, then the terminal won't turn on the GPRS connection again.

17. Appendix E:

Type	Value
Login Information	0x01
~~Location Data（UTC）~~	~~0x12~~
Status Information（Heartbeat Packet）	0x13
LBS Base Stations Data Packet（UTC）	0x28
~~String Information~~	~~0x21~~
Alarm Data（UTC）（Single Fence）	0x26
Alarm Data（UTC）（Multiple Fence）	0x27
~~GPS and Telephone Query Location Information(UTC)~~	~~0x2A~~
Demand Sent to Terminal by Server	0x80
Network Time Checking	8A
OBD Packet(260)	8E
OBD Control Packet(260)	8F