Shenzhen Concox Information Technology Co., Ltd

GPS Tracker
Communication Protocol

(GT02)

Content

1.Communication Protocol

1.1 Introduction

This document defines instructions about interface protocol on application layer of vehicles GPS tracker and location-based service platform. Related interface protocol only applies in the interaction between the platform and the position terminal.

1.2 Network connection

The device adopts long connection of TCP via GPRS.

2.Terms, 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
CI	Cell ID	移动基站
RSSI	Received Signal Strength Indicator	接收信号强度
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

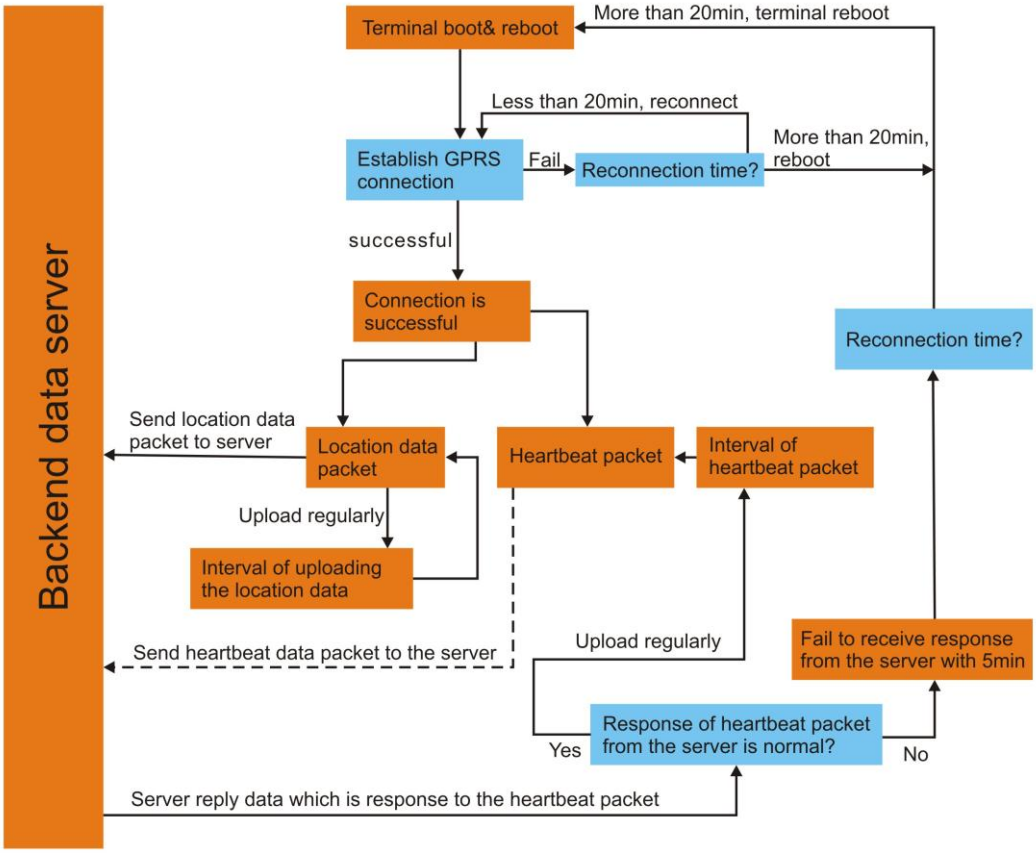
3.Basic rules

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 login message packet. The terminal will start schedule reboot in twenty minutes if the GPRS connection is failed three times. Within twenty minutes, if the terminal successfully connects to the server and receives the data packet from the server as the server's response to the login message packet sent by the terminal, the schedule reboot will be off and the terminal will not be rebooted; otherwise, the

terminal will be rebooted 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 retransmission function for GPS tracking data, which will cause the terminal to disconnect the current GPRS connection and rebuild a new GPRS connection.
4) If the connection is regarded to be abnormal, and the data packet as a response from the server is failed to be received three times after a connection is established and a login message packet or status information package is sent, the terminal will start schedule reboot and the scheduled time is ten minutes. Within ten minutes, if the terminal successfully connects to the server and receives the data packet responded by the server, the schedule reboot will be off and the terminal will not be rebooted; otherwise, the terminal will be rebooted automatically in ten minutes.
5) 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.
6) 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.
7) 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.

Data flow diagram



Protocol number and start bit

Type	Protocol number	Start bit
IP request packet, IP respond packet	0x00	0x78 0x78
GPS data packet（GPS& LBS information）	0x10	0x68 0x68
Heartbeat respond packet	0x1A	0x54 0x68
Heartbeat packet	0x1A	0x68 0x68
Address respond packet(Chinese)	0x17	0x78 0x78
Address respond packet(English)	0x97	0x78 0x78
Address request packet (terminal to the server)	0x1B	0x68 0x68
Packet of issued instruction (server to the terminal)	0x1C	0x88 0x88
Respond packet of issued instruction (terminal to the server)	0x1C	0x68 0x68

4.GPS data packet format

The communication is transferred asynchronously in bytes. The length of each packet transmits between the terminal and the server is uncertain.

	Format		Length(Byte)
GPS data packet （42 Byte）	Start Bit		2
	Packet Length		1
	LAC		2
	Terminal ID		8
	Information Serial Number		2
	Protocol Number		1
	Information content	Date Time	6
		Latitude	4
		Longitude	4
		Speed	1
		Course	2
		MNC	1
		Cell ID	2
		Status bit	4
	Stop Bit		2

1）Start bit

2 bytes, 0x68 0x68

2）Packet length

The total byte from “LAC” to “information content”, and the default is 37 bytes.

3）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).

4）Terminal ID

The terminal ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345,

the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

5）Information Serial Number

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

6）Protocol Number

The protocol number is 0x10.

7）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 15:30:23

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）Latitude

Four bytes are consumed, defining the latitude value of location data. The range of the value is 0-162000000, indicating a range of 0 °-90 °. The conversion method thereof is as follow:

Converting the value of latitude and longitude output by GPS module into a decimal based on minute; multiplying the converted decimal by 30000; and converting the multiplied result into hexadecimal

Example: 22º32.7658'=(22X60+32.7658)X3000=40582974, then converted into a hexadecimal number

40582974(Decimal)= 26B3F3E(Hexadecimal)

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

9）Longitude

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

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

10）Speed

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

e.g. 0x00 represents 0 km/h.
0x10 represents 16km/h.
0xFF represents 255 km/h.

11）Course

Two 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	0
	Bit4	0
	Bit3	0
	Bit2	0
	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
BYTE_1 Bit4 0
BYTE_1 Bit3 0
BYTE_1 Bit2 0
BYTE_1 Bit1 0
BYTE_1 Bit0 1
BYTE_2 Bit7 0
BYTE_2 Bit7 1
BYTE_2 Bit7 0 → Course 332 °(0101001100 in Binary, or 332 in decimal)
BYTE_2 Bit7 0
BYTE_2 Bit7 1
BYTE_2 Bit7 1

BYTE_2 Bit7 0
BYTE_2 Bit7 0 ———

12）MNC

Mobile Network Code(MNC)

Example: Chinese MNC is 0x00.

13）Cell ID

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

14）Status

4 byte is consumed defining for various status information of the mobile phone.

Regards 4 bytes as 32 bits, the lowest bit is 0 while the highest is 31. In the process of data transmission, the higher one will be transmitted first.

Currently only the lowest three bits are used, which indicates whether GPS is located, south/ north latitude, and east/ west longitude

Byte	Bit	Code Meaning
BYTE_1~BYTE3	Bit7	0
	Bit6	0
	Bit5	0
	Bit4	0
	Bit3	0
	Bit2	0
	Bit1	0
	Bit0	0
BYTE_0	Bit7	0
	Bit6	0
	Bit5	0
	Bit4	0
	Bit3	0
	Bit2	0：west longitude 1：East Longitude
	Bit1	0：South Latitude 1：North Latitude
	Bit0	0：GPS is not located, 1：GPS is located

15）Stop Bit

Fixed value: 0x0D 0x0A

5.Heartbeat packet and response format

Heartbeat	Format	Length (Byte)

packet from terminal to the server （20+N Byte）	Start Bit		2
	Packet Length		1
	Voltage Level		1
	GSM Signal Strength		1
	Terminal ID		8
	Serial Number		2
	Protocol Number		1
	Information content	Positioning state	1
		quantity of positioning satellites	1
		SNR of Satellite 1	N
		SNR of Satellite 2	
		SNR of Satellite 3	
		SNR of Satellite ……	
		SNR of Satellite N	
	Stop Bit		2

5.1 Heartbeat packet from terminal to the server

1）Start bit

2 bytes: 0x68 0x68

2）Packet Length

The number of bytes from voltage level to SNR of satellite.

The total length is 15+N, in which N indicates the number of satellite.

3）Voltage level

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

4）GSM Signal Strength

0x00: no signal;

0x01: extremely weak signal;

0x02: very weak signal;

0x03: good signal;

0x04: strong signal.

5）Terminal ID

The terminal ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345,

the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

6）Serial Number

The serial number of the first GPRS data (including GPS data packet, heartbeat packet, command packet) sent after booting is '1', and the serial number of data sent later at each time will be automatically added '1'.

7）Protocol Number

0x1A

8）Positioning state（0x00-0x04）

0x00— not positioned

0x01—real time GPS

0x03— GPS enable

0x04— GPS searching state

9）Quantity of positioning satellites

e.g.: there are 12 satellites, then the value is 0x0C

10）SNR of Satellite

Range: 0x00~0x63（0~99dBHZ）。

Each satellite occupies one byte.

11）Stop bit

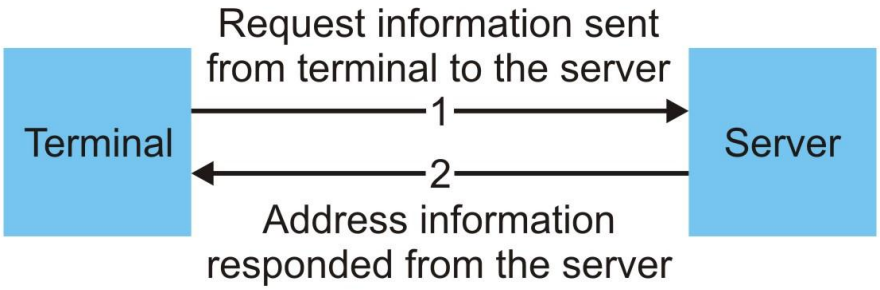
Fixed value: 0x0D 0x0A

5.2 Format of the response from server to the terminal

Format	content	Length (Byte)
Message header	0x54 0x68	2
Protocol Number	0x1A	1
Stop bit	0x0D 0x0A	2

Each time the terminal sends a heartbeat packet to the server, the server needs to respond.

6.Address requests and replies



6.1（terminal to the server）address request packet format

	Format		Length(Byte)
Address request packet（64 Byte）	Start Bit		2
	Packet Length		1
	LAC		2
	Terminal ID		8
	Serial Number		2
	Protocol Number		1
	Information Content	Date Time	6
		Latitude	4
		Longitude	4
		Speed	1
		Course	2
		MNC	1
		Cell ID	2
		Status	4
	Phone number		21
	Language		1
	Stop Bit		2

1）Start bit

0x68 0x68

2）Packet length

The default length is 59.

3）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).

4）Terminal ID

The terminal ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345,

the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

5）Serial number

The serial number of the first GPRS data (including GPS data packet, heartbeat packet, command packet) sent after booting is '1', and the serial number of data sent later at each time will be automatically added '1'.

6）Protocol number

0X1B

7）Date & time

For details see GPS Data Packet Format section 4.

8）Latitude

For details see GPS Data Packet Format section 4.

9）Longitude

For details see GPS Data Packet Format section 4.

10）Speed

For details see GPS Data Packet Format section 4.

11）Course

For details see GPS Data Packet Format section 4.

12）MNC

For details see GPS Data Packet Format section 4.

13）Cell ID

For details see GPS Data Packet Format section 4.

14）Status

For details see GPS Data Packet Format section 4.

15）Phone number

Convert by ASCII, add 0 to the right side if less than 21 bits.

16）Language

0x01: Chinese

0x02: English

17）Stop bit

Fixed value: 0x0D 0x0A

6.2（server to the terminal）address reply packet format

Format			Length(Byte)
Command packet sent from the server to the terminal（15+M/17+M Byte）	Start Bit		2
	Length of data bit		2 Byte for English, 1 Byte for Chinese
	Protocol Number		1
	Information Content	Length of Command	2 Byte for English, 1 Byte for Chinese
		Server Flag Bit	4
		Command Content	M
	Information Serial Number		2
	Check Bit		2
	Stop Bit		2

1）Start bit

The default: 0x78 0x78

2） Length of data bit

The number of bytes from "protocol number" to "check bit"

3）Length of command

Server Flag Bit + Length of Command Content

The length for Chinese command and English command is not the same, 2 bytes for English, 1 byte for Chinese.

4） Protocol number

For English: 0x97;

For Chinese: 0x17

5） Server Flag Bit

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

6）Command content

The server receives address request packet from the terminal, it will check the Chinese address information according to the loaded GPS information and then reply the address information and phone number to the terminal.

Command content: ADDRESS&&address content &&phone number ##(ADDRESS, && and ## are fixed string)

（address content is in Unicode)

7）Serial number

The serial number of the first GPRS data (including GPS data packet, heartbeat packet) sent after booting is '1', and the serial number of data sent later at each time will be automatically added '1'.

8） Error check

A check code may be used by the terminal or the server to distinguish whether the received information is error or not. To prevent errors occur during data transmission, error check is added to against data disoperation, 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 values of CRC-ITU.

CRC error occur when the received information is calculated, the receiver will ignore and discard the data packet.

9）Stop bit

Fixed value in HEX 0x0D 0x0A

7. Packet format of issue instruction online(server to the terminal)

7.1 Packet format of issue instruction online(server to the terminal)

Format			Length(Byte)
Command packet sent from the server to the terminal（15+M+N Byte）	Start Bit		2
	Packet Length		1
	Protocol Number		1
	Information Content	Length of Command	1
		Server Flag Bit	4
		Command Content	M

	Stop Bit	2

1）Start bit

The default: 0x88 0x88

2）Packet length

The length of "protocol number" + the length of information content

3）Length of command

Server Flag Bit + Length of Command Content

4）Protocol number

0x1C

5）Server flag bit

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

6）Command content

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

e.g.: STATUS, 666666#

7）Stop bit

Fixed value in HEX 0x0D 0x0A

7.2 Packet format of command response（terminal to the server）

	Format		Length(Byte)
Command response Packet （15+M+5 Byte）	Start Bit		2
	Packet Length		1
	Reserved bit		2
	Terminal ID		8
	Information Serial Number		2
	Protocol Number		1
	Command information	Length of Command	1
		Server Flag Bit	4
		Command Content	M
	Stop bit		2

1）Start bit

The default: 0x68 0x68

2）Packet length

The length from “reserved bit” to “command information”

3）Reserved bit

0x00 0x00

4）Terminal ID

The terminal ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345,

the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

5）Serial number

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

6）Protocol number

0x1C

7）Length of command

Server Flag Bit + Length of Command Content

8）Server flag bit

Server flag bit is returned without change.

9）Command content

It is represented in ASC II of string; the status of command response (e.g. Success) is stored.

10）Stop bit

Fixed value in HEX 0x0D 0x0A

8.The packet of abnormal state upload

	Format	Length(Byte)
The packet of abnormal state upload	Start Bit	2
	Packet Length	1
	LAC	2

	Terminal ID		8
	Information Serial Number		2
	Protocol Number		1
	Information Content	Date Time	6
		Latitude	4
		Longitude	4
		Speed	1
		Course	2
		MNC	1
		Cell ID	2
		Status bit	4
	Stop Bit		2

1）Start bit

2 bytes: 0x68 0x68

2）Packet length

The number of bytes from “LAC” to “information content”, its default value is 37.

3）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).

4）Terminal ID

The terminal ID applies IMEI number of 15 bits.

Example: if the IMEI is 123456789012345,

the terminal ID is 0x01 0x23 0x45 0x67 0x89 0x01 0x23 0x45.

5）Serial number

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

6）Protocol number

0x10

7）Date &time

For details see GPS Data Packet Format section 4.

8）Latitude

For details see GPS Data Packet Format section 4.

9）Longitude

For details see GPS Data Packet Format section 4.

10）Speed

For details see GPS Data Packet Format section 4.

11）Course

For details see GPS Data Packet Format section 4.

12）MNC

For details see GPS Data Packet Format section 4.

13）Cell ID

For details see GPS Data Packet Format section 4.

14）Status

For details see GPS Data Packet Format section 4.

15）Stop bit

Fixed value in HEX 0x0D 0x0A

9.Code fragment of the CRC-ITU lookup table algorithm implemented based on C language

9.1 Error check

A check code may be used by the terminal or the server to distinguish whether the received information is error or not. To prevent errors occur during data transmission, error check is added to against data misoperation, 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 values of CRC-ITU.

9.2 CRC code:

CRC error occur when the received information is calculated, the receiver will ignore and discard the data packet.

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}
```

9.3 Examples of data packet

GPS data packet: 686825266A03586880000001580001100A0C1E0A2E05027AC8390C46
57C5000156001DF1000000060D0A

Address request packet:68683B266A03586880000001580005 1B0A0C1E0A2E05027AC8390C46
57C5000156001DF1000000063132353230313530313339383134363100 49004F00020D0A

English address packet: 787800AF9700A800000001414444524553532626003200300031 00
30002D00310032002D00330030002000310030003A00340036003A00300035002000365 3F
7002000590075006E00730068006100 6E00200057006500730074002000520064002C00480
075006900630068006500 6E0067002C004800750069007A0068006F0075002C00470075006
1006E00670064006F006E006726263132353230313530313339383134363100 49004F00232
3000103020D0A

Chinese address packet: 787872176C0000000141444452455353262600320030003100 3000
2D00310032002D00330030002000310031003A00340031003A0030003500205E7F4E1C77
0160E05DDE5E0260E057CE533A4E915C71897F8DEF003653F726263132353230313530 3
13339383134363100360036002323000 1ffa80D0A

Heartbeat packet: 68680F0304035868800000 0158E5181A012A0D0A

Heartbeat response packet: 54681A0D0A

IP request packet: 787817000358688000000158300101CC00266A001DF10000DD310D0A

IP response packet: 78780B00716C44082275000025fc0D0A

ShenZhen Concox Information Technology CO.,Ltd
Tel: +86 755 2912 1200
Fax: +86 755 2912 1290
E-mail: aven@concox.cn.
Add: 4/F, Building B, Gaoxinqi Industrial Park,
Liuxian 1st Road, No.67 Bao'an District, Shenzhen