MEITRACK TA255 series GPRS Protocol

Applicable Model: TA255 series

Change History

File Name	MEITRACK TA255 series GPRS Protocol		
Project	TA255 series	Creation Date Update Date	20220920 20230511
Subproject	GPRS Protocol	Total Pages	48
Version	V1.1	Confidential	Internal Documentation

Contents

1 TA255 series Command Format ..... - 5 -
1.1 GPRS Command Format ..... - 5 -
1.2 Tracker Command Format ..... - 5 -
1.3 Event Code ..... - 16 -
2 Command List ..... - 17 -
3 Command Details ..... - 20 -
3.1 Real-Time Location Query (GPRS) – A10 ..... - 20 -
3.2 Setting a Heartbeat Packet Reporting Interval (GPRS) – A11 ..... - 20 -
3.3 Tracking by Time Interval (GPRS) – A12 ..... - 20 -
3.4 Setting the Cornering Report (GPRS) – A13 ..... - 21 -
3.5 Tracking by Distance – A14 ..... - 21 -
3.6 Setting the Parking Scheduled Tracking Function (GPRS) – A15 ..... - 22 -
3.7 Enabling the Parking Scheduled Tracking Function (GPRS) – A16 ..... - 22 -
3.8 Setting GPRS Parameters – A21 ..... - 23 -
3.9 Setting the Standby GPRS Server – A23 ..... - 23 -
3.10 Reading All Authorized Phone Numbers – A70 ..... - 23 -
3.11 Setting Authorized Phone Numbers – A71 ..... - 23 -
3.12 Setting the Smart Sleep Mode – A73 ..... - 24 -
3.13 Obtaining Time from the Server – A82 ..... - 24 -
3.14 Setting up the BLE peripheral– ABD ..... - 25 -
3.15 Temporarily Enabling the Bluetooth Slave Mode – AD9 ..... - 27 -
3.16 Setting a Geo-Fence – B05 ..... - 27 -
3.17 Deleting a Geo-Fence – B06 ..... - 27 -
3.18 Setting the Speeding Alert – B07 ..... - 28 -
3.19 Setting the Towing Alert – B08 ..... - 28 -
3.20 Fast Setting the Towing Alert – B10 ..... - 28 -
3.21 Setting a Polygonal Geo-Fence – B11 ..... - 29 -
3.22 Setting the Idling Alert – B14 ..... - 29 -
3.23 Setting Driver Fatigue Parameters – B15 ..... - 30 -
3.24 Setting Filtering Time of an Input Port – B26 ..... - 30 -
3.25 Turning off the LED Indicator – B31 ..... - 30 -
3.26 Setting a Log Interval – B34 ..... - 31 -
3.27 Setting the SMS Time Zone – B35 ..... - 31 -
3.28 Setting the GPRS Time Zone – B36 ..... - 31 -
3.29 Setting the Auto Sleep Function – B37 ..... - 32 -
3.30 Determining Vehicle Status by ACC Status – B60 ..... - 32 -
3.31 Setting Event Authorization – B99 ..... - 32 -
3.32 Setting the Harsh Acceleration or Harsh Braking Alert – BBD ..... - 33 -
3.33 Setting the Sharp Left Turn or Sharp Right Turn Alert – BC6 ..... - 34 -
3.34 Controlling Output Status – C01 ..... - 34 -
3.35 Setting a GPRS Event Transmission Mode – C03 ..... - 35 -
3.36 Setting the Input Mode of an Input Port – C07 ..... - 35 -

3.37 Registering a Temperature Sensor Number – C40 ..... - 35 -
3.38 Deleting a Registered Temperature Sensor – C41 ..... - 36 -
3.39 Reading the Temperature Sensor SN and Number – C42 ..... - 36 -
3.40 Setting the Temperature Threshold and Logical Name – C43 ..... - 37 -
3.41 Reading Temperature Sensor Parameters – C44 ..... - 37 -
3.42 Checking Temperature Sensor Parameters – C46 ..... - 38 -
3.43 Setting Fuel Parameters – C47 ..... - 39 -
3.44 Reading Fuel Parameters – C48 ..... - 39 -
3.45 Setting the Fuel Theft Alert – C49 ..... - 39 -
3.46 Setting the Positioning Mode – C67 ..... - 40 -
3.47 Setting the RS232 Serial Port and Peripheral – C70 ..... - 40 -
3.48 Setting the Power-off Function of the Power Button – C77 ..... - 40 -
3.49 Locking the 4G Network – C82 ..... - 41 -
3.50 Locking the IoT Network – C94 ..... - 41 -
3.51 Transparently Transmitting Bluetooth Data – CC4 ..... - 41 -
3.52 Setting Bluetooth Pairing – CC5 ..... - 42 -
3.53 Deleting an Event in the Buffer – CFF ..... - 42 -
3.54 Authorizing an IButton Key – D10 ..... - 42 -
3.55 Authorizing iButton Keys in Batches – D11 ..... - 42 -
3.56 Checking iButton Authorization – D12 ..... - 43 -
3.57 Reading an Authorized IButton Key – D13 ..... - 43 -
3.58 Deleting an Authorized IButton Key – D14 ..... - 43 -
3.59 Deleting Authorized iButton Keys in Batches – D15 ..... - 44 -
3.60 Checking the Checksum of the Authorized iButton ID Database – D16 ..... - 44 -
3.61 Setting the Maintenance Mileage-D65 ..... - 44 -
3.62 Setting Maintenance Time-D66 ..... - 45 -
3.63 Setting GPS Data Filtering – D71 ..... - 45 -
3.64 Setting an Output Port– D72 ..... - 45 -
3.65 Allocating GPRS Cache and GPS Log Storage Space - D73 ..... - 46 -
3.66 Displaying the Actual Number of Satellites When the GPS Signal Strength is Low – DDB ..... - 46 -
3.67 Obtaining Real-time GNSS Data – DDC ..... - 46 -
3.68 Using the NITZ Time or Not – DDD ..... - 47 -
3.69 Detecting the Driving Speed (less than 5km/h) or not During Fuel Level Detection – DF3 ..... - 47 -
3.70 Obtaining the Terminal Command List - E04 ..... - 48 -
3.71 Reading Device's Firmware Version and SN – E91 ..... - 48 -
3.72 Restarting the GSM and GPS Module – F00 ..... - 48 -
3.73 Restarting the GSM Module – F01 ..... - 48 -
3.74 Restarting the GPS Module – F02 ..... - 48 -
3.75 Setting the Mileage and Run Time – F08 ..... - 49 -
3.76 Deleting SMS or GPRS Cache Data – F09 ..... - 49 -
3.77 Restoring Initial Settings – F11 ..... - 49 -
3.78 Changing the Device Password – F20 ..... - 49 -

1 TA255 series Command Format

1.1 GPRS Command Format

The GPRS command format is as follows:

GPRS command sent from the server to the tracker	*@@<Data identifier><Data length>,<IMEI>,<Parameter table No.><Command code>,<Command content><*Checksum>*\r\n
GPRS command sent from the tracker to the server	*$$<Data identifier><Data length>,<IMEI>,<Event code>,<Command content/Error code><*Checksum>*\r\n
Command description	

- **@@**: Indicates the packet header sent from the server to the tracker. Contains 2 characters.
- **Data identifier**: Contains 1 byte. The character type is hexadecimal, and its value ranges from **0x41** to **0x7A**. The data identifier in the reply command must be the same as that of the sending command. Otherwise, the command fails to be sent.
- A comma "," is used to separate data characters. The character type is the American Standard Code for Information Interchange (ASCII) (hexadecimal: 0x2C).
- **Data length**: Indicates the length of characters from the first separator "," to the ending character "\r\n" (including the first separator and the ending character). The character type is decimal.
  $$*<Data identifier><Data length>,<IMEI>,<Command type>,<Command content><*Checksum>*\r\n
- **IMEI**: Indicates the IMEI number of the GSM module. But the number stored on the flash memory can be changed.
- **Parameter table No.**: When the parameter value is **0** or not set, all parameter tables are modified. When the parameter value is **1**, the basic parameter table is modified. When the parameter value is **2**, roaming parameter table 1 is modified.
- **Command code**: Consists of letters and digits. For detail, see the chapter 3 "Command Details."
- **Command content**: no more than 1,024 bytes.
- *****: This is a fixed character. **Checksum**: Contains 2 hexadecimal characters; indicates the sum of characters from the packet header "$$" to the asterisk "*" (including the packet header and asterisk).
  $$*<Data identifier><Data length>,<IMEI>,<Command type>,<Command content><*Checksum>*\r\n
- **\r\n**: Contains 2 bytes. The parameter is an ending character. Hexadecimal: 0x0D 0x0A.
- **$$**: Indicates the packet header sent from the tracker to the server. Contains 2 bytes. Hexadecimal: 0x24 0x24.

If there are multiple commands, use the separator "," to separate them. If there is no command and the **Command content** parameter is required, the separator "," needs to be remained.

1.2 Tracker Command Format

The data format is as follows:

***$$<Data identifier><Data length>,<IMEI>,<Command type>,<Number of remaining cache records><Number of data packets><Data packet 1><Data packet 2><Data packet 3>…<Data packet N><*Checksum>*\r\n**

There is one or multiple data packets. When there are multiple data packets, the data is stored in the form of cache in the flash memory after the network is disconnected. After the network is connected normally, the tracker sends the cached data to the server in batches. When there is only one data packet, it means that this is a

piece of real-time data. The following is an example of data including only one data packet.

The command content in CCE format is as follows:

Parameter	Description	Example
@@ / $$	@@: Indicates the GPRS data packet header sent from the server to the tracker. The header type is ASCII (hexadecimal: 0x40). $$: Indicates the GPRS data packet header sent from the tracker to the server. The header type is ASCII (hexadecimal: 0x24).	Hexadecimal: 0x24 0x24 ASCII: $$
Data identifier	Contains 1 byte. The type is the ASCII, and its value ranges from **0x41** to **0x7A**.	Hexadecimal: 0x47 ASCII: G
Data length	Unit: byte. Type: decimal. Indicates the length of characters from the first separator "," to the ending character "\r\n" (including "," and "\r\n"), that is, the content underlined below. $$*<Data identifier><Data length>,<IMEI>,<Command type>,<Number of remaining cache records><Number of data packets><Data packet 1><Data packet 2><Data packet 3>…<Data packet N><*Checksum>*\r\n	Hexadecimal: 0x32 0x30 0x35 ASCII: 205
IMEI	Indicates the tracker's IMEI number. It has 15 digits generally.	Hexadecimal: 0x38 0x36 0x38 0x39 0x39 0x38 0x30 0x33 0x33 0x320x 34 0x32 0x33 0x33 0x36 ASCII: 868998033242336
Command type	Indicates the type of commands in the tracker protocol.	Hexadecimal: 0x43 0x43 0x45 ASCII: CCE
The following data is hexadecimal:		
Number of remaining cache records	Contains 4 bytes; hexadecimal; little-endian	0x00 0x00 0x00 0x00 The number of remaining cache records is 0.
Number of data packets	Contains 2 bytes; hexadecimal; little-endian	0x01 0x00 The entire message contains only data packet.
The following is the detailed data of each data packet which needs to be parsed by using hexadecimal numbers. This protocol only describes data including only one data packet.		
Length of the current data packet	Contains 2 bytes; hexadecimal; little-endian. Indicates the length of characters from the "total number of ID in the current data packet" parameter to the last parameter ID of the current data packet.	0xAB 0x00 The length of the current data packet is 171 bytes.

Total number of ID in the current data packet		Contains 2 bytes; hexadecimal; little-endian	0x2E 0x00 There are 46 ID numbers in the data packet.
Number of 1-byte parameter ID		Value range: 0x00–0xFF The length of the following parameter ID numbers is 1 byte.	0x08 There are 8 parameter ID numbers whose length is 1 byte. 0x00: The current data packet does not contain any parameter ID number whose length is 1 byte.
GPS positioning status	Parameter ID: 0x05	0x01: The GPS positioning is valid. 0x00: The GPS positioning is invalid. Data type: BYTE	0x01 The GPS positioning is valid.
Number of satellites	Parameter ID: 0x06	Indicates the number of received GPS satellites. Data type: BYTE	0x0A The number of received GPS satellites is 10.
GSM signal strength	Parameter ID: 0x07	Value range: 0x00–0x31 Data type: BYTE	0x1C The GSM signal strength is 28.
Output port status	Parameter ID: 0x14	Indicates the status values of eight output ports. Bits 0–7 correspond to status of output ports 1–8. Data type: BYTE	0x00 Converted to binary digits: 0000 0000 Output ports 1–8 is inactive.
input port status	Parameter ID:0x15	Indicates the status values of eight input ports. Bits 0–7 correspond to status of input ports 1–8. Data type: BYTE	0x00 Converted to binary digits: 0000 0000 input ports 1–8 is inactive.
Geo-fence number	Parameter ID: 0x1B	The data is available only when the GPRS event code is 20 or 21. Data type: BYTE	0x00 No Enter Geo-fence or Exit Geo-fence alert is generated.
Temperature sensor No.	Parameter ID: 0x27	07 Indicates temperature sensor 7. The data is available only when the GPRS event code is 50 or 51. Data type: BYTE	
Number of 2-byte parameter ID		Value range: 0x00–0xFF The length of the following parameter ID numbers is 2 bytes.	0x09 There are 9 parameter ID numbers whose length is 2 bytes. 0x00: The current data packet

			does not contain any parameter ID number whose length is 2 bytes.
Speed	Parameter ID: 0x08	Unit: km/h; little-endian Data type: WORD	0x15 0x00 The driving speed is 21 km/h.
Driving direction	Parameter ID: 0x09	The unit is degree. When the parameter value is **0**, the direction is due north. The parameter value ranges from **0** to **359**. Little-endian. Data type: WORD	0x66 0x00 The driving direction is 102 degrees.
Horizontal dilution of precision (HDOP)	Parameter ID: 0x0A	Value range: 5–999 Unit: 1/10; little-endian Data type: WORD	0x13 0x00 The HDOP is 1.9.
Altitude	Parameter ID: 0x0B	Unit: meter; little-endian Data type: SINT16	0x2D 0x00 The altitude is 45 meters.
AD1	Parameter ID: 0x16	Analog <AD1>; little-endian Voltage formula of analog: AD1/100 Data type: WORD	0x00 0x00 Convert the digits to decimal digits: 0 0/100 = 0 The AD1 voltage is 0V.
AD5	Parameter ID: 0x1A	External power analog <AD5>; little-endian Voltage formula of analog: AD5/100 Note: When the external power supply is disconnected, the voltage of AD5 is about 2 V instead of 0. Data type: WORD	0x51 0x05 Convert the digits to decimal digits: 1366 1366/100 = 13.66 The voltage of the external power supply is 13.66 V.
Fuel level (%)	Parameter ID: 0x29	Little-endian. After the digits are converted to decimal digits, the converted value divided by 10000 is the actual value. Data type: WORD	0x7A 0x0D Convert the digits to decimal digits: 3450 The fuel level is 34.50%.
New event code	Parameter ID: 0x40	For details, see the section 1.3 "Event Code." Data type: WORD	0x23 The event code is 35
Geofence Index	Parameter ID: 0xFE90	Little-endian Data type: WORD	Only available in event 20,21
Number of 4-byte parameter ID		Value range: 0x00–0xFF The length of the following parameter ID numbers is 4 bytes.	0x07 There are 7 parameter ID numbers whose length is 4 bytes. 0x00: The current data packet does not contain any parameter ID number whose length is 2 bytes.

Latitude	Parameter ID: 0x02	Unit: millionth of a degree; little-endian Data type: SINT32	0xE6 0x87 0x57 0x01 Convert the digits to decimal digits: 22513638 The latitude is 22.513638 degrees.
Longitude	Parameter ID: 0x03	Unit: millionth of a degree; little-endian Data type: SINT32	0XE6 0x5F 0xCC 0x06 Convert the digits to decimal digits: 114057190 The longitude is 114.057190 degrees.
Date and time	Parameter ID: 0x04	Contains 4 bytes; little-endian Unit: second Starting time: 1 January, 2000, 00:00:00 am. Data type: DWORD	0xEA 0x8D 0xA7 0x22 Convert the digits to decimal digits: 581406186
Mileage	Parameter ID: 0x0C	Indicates the total mileage. Unit: meter; little-endian Data type: DWORD	0x56 0x05 0x00 0x00 Convert the digits to decimal digits: 1366 The total mileage is 1366 meters.
Run time	Parameter ID: 0x0D	Indicates the total time. Unit: second; little-endian Data type: DWORD	0x96 0x1B 0x00 0x00 Convert the digits to decimal digits: 7062 The run time is 7062 seconds.
System flag	Parameter ID: 0x1C	The data is available only when the GPRS event code is 35. Bit 0: Whether to modify the EEP2 parameter. When the parameter value is **1**, the EEP2 parameter is modified. Bit 1: Indicates the ACC status. When the parameter value is **1**, the ACC is on. Bit 2: Indicates the anti-theft status. When the parameter value is **1**, the tracker is in the arming state. Bit 3: vibration flag. When the parameter value is **1**, the tracker is vibrating. Bit 4: moving flag. When the parameter value is **1**, the tracker is moving. Bit 5: Whether to connect the external power supply. When the parameter value is **1**, the external power supply is connected. Bit 6: Whether the tracker is charging. When the parameter value is **1**, the tracker is charging. Bit 7: Whether to enable the sleep mode.	0x00 0x00 0x03 0x00 Converted to binary digits: 0000 0000 0000 0000 0000 0011 0000 0000 The tracker supports the FMS function and is connected to the FMS.

		When the parameter value is **1**, the sleep mode is enabled. Bit 8: Whether to connect the FMS. When the parameter value is **1**, the FMS is connected. Bit 9: Whether to enable the FMS function. When the parameter value is **1**, the function is enabled. Bits 10–31: reserved. Data type: DWORD	
iButton ID	Parameter ID: 0x25	Indicates the ID number of a iButtonIButton. The data is available only when the GPRS event code is 37. Data type: DWORD	0xD7 0x9D 0xD1 0x00 The iButton ID number is 13737431.
Number of unfixed-byte parameter ID		Value range: 0x00–0xFF The length of the following parameter ID numbers is 8 or 12 bytes, or is unfixed. The ordering of parameter ID numbers is not fixed. For details, see the parameter ID table.	0x01 There is one unfixed-byte parameter ID number. 0x00: The current data packet does not contain any parameter ID number whose length is unfixed.
Current base station info	Parameter ID: 0x0E	<Data length><MCC><MNC><LAC><CELL_ID><RX_LEVEL> Data length: hexadecimal; indicates the length of the base station data. Unit: byte. MCC: 16-bit unsigned; little-endian; indicates the Mobile Country Code. MNC: 16-bit unsigned; little-endian; indicates the Mobile Network Code. LAC: 16-bit unsigned; little-endian; indicates the Location Area Code. CELL_ID: 32-bit unsigned; little-endian; indicates the cell ID. RX_LEVEL: 16-bit signed; little-endian; indicates the signal strength. Data type: STRUCT	0x0C 0xCC 0x01 0x01 0x00 0x45 0xA5 0x8B 0xD4 0xE9 0x01 0xBB 0xFF 0x0C: The data length is 12 bytes. 0xCC 0x01: The MCC is 460. 0x01 0x00: The MNC is 01. 0x45 0xA5: The LAC is 42309. 0x8B 0xD4 0xE9 0x01: The cell ID is 32101515. 0xBB 0xFF: The signal strength is -69 dbm.
Temperature sensor 1	Parameter ID: 0x2A	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 2	Parameter ID: 0x2B	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01.

			09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 3	Parameter ID: 0x2C	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 4	Parameter ID: 0x2D	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 5	Parameter ID: 0x2E	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 6	Parameter ID: 0x2F	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 7	Parameter ID: 0x30	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
Temperature sensor 8	Parameter ID: 0x31	Little-endian Data type: STRUCT	0x01 0x09 0x1A 01: Indicates sensor 01. 09 1A: signed; 2 bytes; little-endian. The temperature is 66.65°C.
WiFi info 1	Parameter ID: 0x1D	When no GPS signal is detected, enable the WiFi function. Then WiFi data can be obtained. <Data length><MAC><RSSI> Data length: hexadecimal; indicates the length of the MAC address and RSSI. Unit: byte. The fixed data length is 8 bytes. MAC: Contains 6 bytes; indicates the MAC address of the WiFi network. RSSI: signed; 2 bytes; little-endian; indicates the WiFi signal strength.	0x08 0x00 0x25 0x86 0xA7 0x0B 0x0A 0xDA 0xFF 0x08: The data length is 8 bytes. 0x00 0x25 0x86 0xA7 0x0B 0x0A: The MAC address is 00-25-86-A7-0B-0A. 0xCF 0xFF: The signal strength is -38 dbm.

WiFi info 2	Parameter ID: 0x1E	The description is the same as that of WiFi info 1.	0x08 0x38 0x83 0x45 0xE1 0xA6 0x36 0XC7 0xFF The description is the same as that of WiFi info 1.
WiFi info 3	Parameter ID: 0x1F	The description is the same as that of WiFi info 1.	0x08 0x8C 0x21 0x0A 0x78 0x30 0x56 0xC6 0xFF The description is the same as that of WiFi info 1.
WiFi info 4	Parameter ID: 0x20	The description is the same as that of WiFi info 1.	0x08 0xF0 0xB4 0x29 0x8B 0x4B 0xDD 0xBF 0xFF The description is the same as that of WiFi info 1.
WiFi info 5	Parameter ID: 0x21	The description is the same as that of WiFi info 1.	0x08 0x38 0x83 0x45 0xAD 0x89 0x72 0xBE 0xFF The description is the same as that of WiFi info 1.
WiFi info 6	Parameter ID: 0x22	The description is the same as that of WiFi info 1.	0x08 0x00 0x25 0x68 0x60 0x1F 0x10 0xB9 0xFF The description is the same as that of WiFi info 1.
WiFi info 7	Parameter ID: 0x23	The description is the same as that of WiFi info 1.	0x08 0x0A 0x18 0xD6 0x0B 0x15 0xAE 0xB8 0xFF The description is the same as that of WiFi info 1.
WiFi info 8	Parameter ID: 0x24	The description is the same as that of WiFi info 1.	0x08 0x7C 0x03 0xC9 0x10 0xF9 0xB0 0xB8 0xFF The description is the same as that of WiFi info 1.
network information	Parameter ID:0x4B	<ID_Len><version><Type><DescriptorLen><Descriptor> ID_Len：indicates the length of this ID. Contains one byte. version：indicates the data version. 0X01: indicates version 1. Contains one byte. version = 0x01：1：mobile network（default）; DescriptorLen：Network descriptor length，1 byte。range：0~32 Descriptor：Network descriptor，String format.	0x06 0x01 0x01 0x03 0x4C 0x54 0x45 0x06：6 bytes 0x01：version number is 1 0x01：mobile network 0x03：The network descriptor is 3 bytes 0x4C 0x54 0x45：To string format：LTE
Additional info of a Bluetooth device	Parameter ID: 0xFE70	<ID_Len><Version><Type><Data> ID_Len: indicates the length of this ID. Contains one byte. Version: indicates the data version. 0X01: indicates version 1. Contains one byte.	

		Type: indicates the alert type. Contains one byte. 01: low battery alert for the temperature and humidity sensor. 02: high temperature alert for the temperature and humidity sensor. 03: low temperature alert for the temperature and humidity sensor. 04: high humidity alert for the temperature and humidity sensor. 05: low humidity alert for the temperature and humidity sensor. 06: signal lost alert for the temperature and humidity sensor. 07: signal recovery alert for the temperature and humidity sensor. 08: low battery alert for the Bluetooth beacon. 09: Bluetooth beacon lost alert. 10: Bluetooth beacon found alert. Data: The data format varies depending on the alert type. Big-endian. Example: Temperature and humidity data analysis (signed 8.8 fixed-point) Alert type: 0x01\0x02\0x03\0x04\0x05: 0x31 0x32 0x33 0xAB 0xBC 0xB1 0x00 0x11 0x22 0x0A 0x64 0x48 0x63 0x48 Data analysis: 0x03: indicates the length of the device name. The device name contains three bytes as follows. 0x31 0x32 0x33: indicates the device name. The device name is **123**. It contains a maximum of 16 bytes. The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. 0x0A: indicates the battery power of the device. 0x64 0x48: indicates the temperature of the device. Little-endian. The final value needs to be divided by 256. 0x63 0x48: indicates the humidity of the device. Little-endian. The final value needs to be divided by 256. Alert type: 0x06\0x07\0x09\0x10: 0x03 0x31 0x32 0x33 0xAB 0xBC 0xB1 0x00 0x11 0x22 Data analysis: 0x03: indicates the length of the device name. The device name contains three bytes as follows. 0x31 0x32 0x33: indicates the device name. It contains a maximum of 16 bytes. The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. Alert type: 0x08: 0x03 0x31 0x32 0x33 0xAB 0xBC 0xB1 0x00 0x11 0x22 0x0A 0xF8 0x03: indicates the length of the device name. The device name contains three bytes as follows. 0x31 0x32 0x33: indicates the device name. It contains a maximum of 16 bytes.

		The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. 0x0A: indicates the battery power of the device. The device reaches 10% battery power left. 0xF8: indicates the signal strength. The signal strength is -8 dbm. Signed.
Bluetooth beacon A	Parameter ID: 0xFE71	<ID_Len><Version><Data 1><Data 2><Data 3><Data 4><Data 5><Data 6><Data 7><Data 8> ID_Len: indicates the length of this ID. Contains one byte. Version: indicates the data version. 0X01: indicates version 1. Contains one byte. <Data>: big-endian 0x03: indicates the length of the device name. The device name contains three bytes as follows. 0x31 0x32 0x33: indicates the device name. It contains a maximum of 16 bytes. The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. 0x0A: indicates the battery power of the device. The device reaches 10% battery power left. 0xF8: indicates the signal strength. The signal strength is -8 dbm. Signed.
Bluetooth beacon B	Parameter ID: 0xFE72	<ID_Len><Version><Data 1><Data 2><Data 3><Data 4><Data 5><Data 6><Data 7><Data 8> ID_Len: indicates the length of this ID. Contains one byte. Version: indicates the data version. 0X01: indicates version 1. Contains one byte. <Data>: big-endian 0x03: indicates the length of the device name. The device name contains three bytes as follows. 0x31 0x32 0x33: indicates the device name. It contains a maximum of 16 bytes. The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. 0x0A: indicates the battery power of the device. The device reaches 10% battery power left. 0xF8: indicates the signal strength. The signal strength is -8 dbm. Signed.
Temperature and humidity sensor	Parameter ID: 0xFE73	<ID_Len><Version><Data 1><Data 2><Data 3><Data 4> ID_Len: indicates the length of this ID. Contains one byte. Version: indicates the data version. 0X01: indicates version 1. Contains one byte. <Data>: big-endian Example: 0x03 0x31 0x32 0x33 0xAB 0xBC 0xB1 0x00 0x11 0x22 0x0A 0x64 0x48 0x63 0x48 0x64 0x48 0x63 0x48 0x64 0x48 0x63 0x48 Data analysis: 0x03: indicates the length of the device name. The device name contains three

		bytes as follows. 0x31 0x32 0x33: indicates the device name. It contains a maximum of 16 bytes. The length of the following data is fixed. 0xAB 0xBC 0xB1 0x00 0x11 0x22: indicates the MAC code of the device. 0x0A: indicates the battery power of the device. 0x64 0x48: indicates the temperature of the device. Little-endian. The final value needs to be divided by 256. 0x63 0x48: indicates the humidity of the device. Little-endian. The final value needs to be divided by 256. 0x64 0x48 0x63 0x48: indicates the device temperature alert threshold. (0x64 0x48: indicates the high temperature alert threshold. 0x63 0x48: indicates the low temperature alert threshold.) 0x64 0x48 0x63 0x48: indicates the device humidity alert threshold. (0x64 0x48: indicates the high humidity alert threshold. 0x63 0x48: indicates the low humidity alert threshold.)
Sensor data	Parameter ID: 0XFEA5	*<ID_LEN><Version><Atmospheric_pressure><Temperature><Humidity>* ID_Len: indicates the length of this ID. Contains one byte. Version: indicates the version of sensor data. The current version is 0x01. Atmospheric_pressure: indicates the atmospheric pressure. Unsigned; two bytes; little-endian. Unit: hPa. 0xFFFF: The atmospheric pressure is not obtained. Temperature: indicates the ambient temperature of the device. Signed; two bytes; little-endian. Unit: °C. 0x7FFF: The temperature is not obtained. Humidity: indicates the ambient humidity of the device. Unsigned; one byte. Unit: RH. 0xFF: The humidity is not obtained.
Battery status	Parameter ID: 0XFEA8	*<ID_LEN><Status of battery 1><Battery level of battery 1><Status of battery 2><Battery level of battery 2><Status of battery 3><Battery level of battery 3><Alert No.>* ID_Len: indicates the length of this ID. Contains one byte. Battery status: contains one byte. **0**: The battery cannot be detected. **1**: low battery. **2**: The battery status is normal. Battery level: contains one byte. Value range: 0–100%. Alert No.: contains one byte. Indicates the battery alert No. When the event code is 168. Value range: 1–3. When the event code is not 168, the fixed alert No. is 0x00.
Battery charging status	Parameter ID: 0XFEB2	<ID_LEN><version><flag> ID_Len: indicates the length of this ID. Contains one byte. version： 0x01 Flag: bit0: if 0 means discharging if 1 means charging bit1~bit31: reserve
The current data packet ends here.		
*	Contains 1 byte. It is used to separate the	*

	command content from the checksum. ASCII (hexadecimal: 0x2A)	
Checksum	Contains 2 bytes. Indicates the sum of hexadecimal characters from the packet header "$$" to the asterisk "*" (including the packet header and asterisk). *$$<Data identifier><Data length>,<IMEI>,<Command type>,<Hexadecimal data packet><*Checksum*>\r\n	If the sum result is **0x27 0x62**, send the checksum 62 (low byte 0x62) in ASCII format, that is, **0x36 0x32**. Hexadecimal: 0x36 0x32 ASCII: 62
\r\n	Contains 2 bytes. This is an ending character. The type is ASCII (hexadecimal: 0x0D,0x0A).	\r\n

1.3 Event Code

Event Code	Event	Default SMS Header (At Most 16 Bytes)
1	**Input 1 Active**	Ignition On
9	**Input 1 Inactive**	Ignition Off
18	**Low External Battery**	Low Ext-Battery
19	**Speeding**	Speeding
20	**Enter Geo-fence**	Enter Fence N (N means the number of the fence)
21	**Exit Geo-fence**	Exit Fence N (N means the number of the fence)
22	**External Battery On**	Ext-Battery On
23	**External Battery Cut**	Ext-Battery Cut
24	**GPS Signal Lost**	GPS Signal Lost
25	**GPS Signal Recovery**	GPS Recovery
26	**Enter Sleep**	Enter Sleep
27	**Exit Sleep**	Exit Sleep
29	**Device Reboot**	Power On
31	**Heartbeat**	/
32	**Cornering**	Cornering
33	**Track By Distance**	Distance
34	**Reply Current (Passive)**	Now
35	**Track By Time Interval**	Interval
36	**Tow**	Tow
37	**iButton (change uart rate)**	
40	**Power Off**	
41	**Stop Moving**	Stop Moving
42	**Start Moving**	Start Moving
50	**Temperature High**	Temp High
51	**Temperature Low**	Temp Low

52	**Full Fuel**	Full Fuel
53	**Low Fuel**	Low Fuel
54	**Fuel Theft**	Fuel Theft
78	**Impact**	Impact
80	**Install**	**Install**
81	**Drop Off**	**Drop Off**
82	**Refuelling**	Refuelling
83	**Ult-Sensor Drop**	**Ult-Sensor Drop**
90	**Sharp Turn to Left**	Sharp Turn to Left
91	**Sharp Turn to Right**	Sharp Turn to Right
94	**Output 1 Active**	Output 1 Active
99	**Output 1 Inactive**	Output 1 Inactive
129	**Harsh Braking**	Harsh Braking
130	**Harsh Acceleration**	Fast Accelerate
133	**Idle Overtime**	Idle Overtime
134	**Idle Recovery**	Idle Recovery
135	**Fatigue Driving**	Fatigue Driving
136	**Enough Rest after Fatigue Driving**	Enough Rest
139	**Maintenance Notice**	Maintenance
165	**BLE Alarm**	BLE Alarm
168	**Battery Alam**	**Battery Alam**

2 Command List

Command	Command Description
A10	Real-Time Location Query (GPRS)
A11	Setting a Heartbeat Packet Reporting Interval (GPRS)
A12	Tracking by Time Interval (GPRS)
A13	Setting the Cornering Report (GPRS)
A14	Tracking by Distance
A15	Setting the Parking Scheduled Tracking Function (GPRS)
A16	Enabling the Parking Scheduled Tracking Function (GPRS)
A21	Setting GPRS Parameters
A23	Setting the Standby GPRS Server
A70	Reading All Authorized Phone Numbers
A71	Setting Authorized Phone Numbers
A72	Setting Listen-in Phone Numbers
A73	Setting the Smart Sleep Mode
A82	Obtaining Time from the Server
ABD	Setting up the BLE peripheral
AD9	Temporarily Enabling the Bluetooth Slave Mode

ABD	Setting BLE peripherals (Bluetooth models only)
B05	Setting a Geo-Fence
B06	Deleting a Geo-Fence
B07	Setting the Speeding Alert
B08	Setting the Towing Alert
B10	Fast Setting the Towing Alert
B11	Setting a Polygonal Geo-Fence
B14	Setting the Idling Alert
B15	Setting Driver Fatigue Parameters
B26	Setting Filtering Time of an Input Port
B31	Turning off the LED Indicator
B34	Setting a Log Interval
B35	Setting the SMS Time Zone
B36	Setting the GPRS Time Zone
B37	Setting the Auto Sleep Function
B60	Determining Vehicle Status by ACC Status
B99	Setting Event Authorization
BBD	Setting the Harsh Acceleration or Harsh Braking Alert
BC6	Setting the Sharp Left Turn or Sharp Right Turn Alert
C01	Controlling Output Status
C03	Setting a GPRS Event Transmission Mode
C07	Setting the Input Mode of an Input Port
C40	Registering a Temperature Sensor Number
C41	Deleting a Registered Temperature Sensor
C42	Reading the Temperature Sensor SN and Number
C43	Setting the Temperature Threshold and Logical Name
C44	Reading Temperature Sensor Parameters
C46	Checking Temperature Sensor Parameters
C47	Setting Fuel Parameters
C48	Reading Fuel Parameters
C49	Setting the Fuel Theft Alert
C67	Setting the Positioning Mode
C70	Setting the RS232 Serial Port and Peripheral
C77	Setting the Power-off Function of the Power Button
C82	Locking the 4G Network （Only TA255 seriesL support）
C94	Locking the IoT Network （Only TA255 seriesE support）
CC4	Transparently Transmitting Bluetooth Data
CC5	Setting Bluetooth Pairing
CFF	Deleting an Event in the Buffer
D10	Authorizing an IButton Key
D11	Authorizing iButton Keys in Batches
D12	Checking iButton Authorization

D13	Reading an Authorized IButton Key
D14	Deleting an Authorized IButton Key
D15	Deleting Authorized iButton Keys in Batches
D16	Checking the Checksum of the Authorized iButton ID Database
D65	Setting the Maintenance Mileage
D66	Setting Maintenance Time
D71	Setting GPS Data Filtering
D72	Setting an Output Port
D73	Allocating GPRS Cache and GPS LOG Storage Space
DDB	Displaying the Actual Number of Satellites When the GPS Signal Strength is Low
DDC	Obtaining Real-time GNSS Data
DDD	Using the NITZ Time or Not
DF3	Detecting the Driving Speed (less than 5km/h) or not During Fuel Level Detection
E04	Obtaining the Terminal Command List
E91	Reading Device's Firmware Version and SN
F00	Restarting the GSM or GPS Module
F01	Restarting the GSM Module
F02	Restarting the GPS Module
F08	Setting the Mileage and Run Time
F09	Deleting SMS or GPRS Cache Data
F11	Restoring Initial Settings
F20	Changing the Device Password

3 Command Details

3.1 Real-Time Location Query (GPRS) – A10

GPRS Sending	A10
GPRS Reply	AAA,34,(-)*Latitude*,(-)*Longitude*,*Date and time*,*Positioning status*,*Number of satellites*,*GSM signal strength*,*Speed*,*Direction*,*HDOP*,*Altitude*,*Mileage*,*Run time*,*Base station info*,*I/O port status*,*Analog input value*
Description	**34**: Indicates the event code of the GPRS command.
Example	
GPRS Sending	@@Q25,353358017784062,A10*6A\r\n
GPRS Reply	$$Q128,353358017784062,AAA,34,22.543176,114.078448,100313093738,A,5,22,2,205,5,-14,0,60,0|0|10133|4110,0000,149|153|173|2707|914,*91\r\n

3.2 Setting a Heartbeat Packet Reporting Interval (GPRS) – A11

GPRS Sending	A11,*Interval*
GPRS Reply	A11,OK
Description	The heartbeat packet function is used to keep the Transmission Control Protocol (TCP) connection open when the interval of scheduled GPRS reporting is long. Interval = 0: function disabled (default). Interval = [1…65535]: function enabled. Set the heartbeat packet reporting interval. Unit: second. The heartbeat function is available only in conjunction with deep sleep mode. When the device enters the deep sleep mode, a heartbeat packet will be sent at the specified interval. A heartbeat packet is to confirm the device is online, and positioning data is invalid.
Example	
GPRS Sending	@@S28,353358017784062,A11,10*FD\r\n
GPRS Reply	$$S28,353358017784062,A11,OK*FE\r\n *After the above command is sent successfully, the device will send the following GPRS heartbeat packet to the platform every 10 minutes in sleep mode:* *$$a131,353358017784062,AAA,31,22.913458,114.083183,080229123628,V,9,23,21,83,1,18,1350,127,0|0|10133|4110,0000,169|181|184|2714|919,*60*

3.3 Tracking by Time Interval (GPRS) – A12

GPRS Sending	A12,*Interval*
GPRS Reply	A12,OK
Description	Interval unit: x10 seconds Interval = 0: function disabled. The maximum time interval is 65535 x 10 seconds.

	Recommended value: 6 x 10 seconds
Example	
GPRS Sending	@@V27,353358017784062,A12,6*D5\r\n
GPRS Reply	$$V28,353358017784062,A12,OK*02\r\n *After the above command is sent successfully, the device will send the following GPRS data packet to the platform every one minute:* *$$W129,353358017784062,AAA,35,22.540113,114.076141,100313094354,A,5,22,1,17 4,4,129,0,435,0|0|10133|4110,0000,166|224|193|2704|916,*BE\r\n*

3.4 Setting the Cornering Report (GPRS) – A13

GPRS Sending	A13,*Angle*
GPRS Reply	A13,OK
Description	When the driving angle exceeds the preset value, the device will send a GPRS data packet with location information to the server, which ensures a smoother route on the platform. Angle = 0: function disabled (default). Angle = [1…359]: function enabled. Recommended value: 30
Example	
GPRS Sending	@@X29,353358017784062,A13,120*37\r\n
GPRS Reply	$$X28,353358017784062,A13,OK*05\r\n *After the above command is sent successfully, if the cornering angle is greater than 120 degrees, the device will send the following GPRS data packet to the server:* *$$Y129,353358017784062,AAA,32,22.540968,114.077455,100313094534,A,4,22,1,166, 3,175,0,534,0|0|10133|4110,0000,141|138|159|2691|904,*D9\r\n*

3.5 Tracking by Distance – A14

GPRS Sending	A14,*Distance*
GPRS Reply	A14,OK
Description	Distance = 0: function disabled (default). Distance = [1…65535]: function enabled. Set the distance. Unit: meter. Note: When both the GPRS time interval and distance tracking functions have been set, the "first reach first report" rule will be applied, and then both the time interval and distance counters will be reset to 0. For example, set the time interval to 6 x 10 seconds and distance to 200 meters. If the road is clear and the driving time is less than one minute, a distance data packet will be reported first; if there is heavy traffic on the road and the vehicle only drives 100 meters within one minute, a time interval data packet will be reported first. Recommended value: 300
Example	
GPRS Sending	@@D30,353358017784062,A14,1000*4A\r\n

GPRS Reply	$$D28,353358017784062,A14,OK*F2\r\n *After the above command is sent successfully, if the driving distance reaches 1000 meters, the device will send the following data packet to the server:* *$$D131,353358017784062,AAA,33,22.547271,114.047405,080310080929,A,8,21,13,89,1,12,8525,561,0|0|10133|4110,0000,163|185|186|2712|939,*31\r\n*

3.6 Setting the Parking Scheduled Tracking Function (GPRS) – A15

GPRS Sending	A15,*Interval*
GPRS Reply	A15,OK
Description	The function is available for vehicle trackers only. With the function, the number of GPRS messages is reduced, and thus GPRS traffic is saved. After the A15 function is set, the A16 function will be automatically enabled. For details about engine status, see the section 3.7 "Enabling the Parking Scheduled Tracking Function (GPRS) – A16." Interval unit: x10 seconds Interval = 0: function disabled. The maximum time interval is 65535 x 10 seconds. Note: If data needs to be sent at the specified time interval after the vehicle starts or stops, the function needs to work with the A12 function. For details, see A12 and A16 commands.
Example	
GPRS Sending	@@E27,353358017784062,A15,6*C7\r\n
GPRS Reply	$$E28,353358017784062,A15,OK*F4\r\n

3.7 Enabling the Parking Scheduled Tracking Function (GPRS) – A16

GPRS Sending	A16,*Status*
GPRS Reply	A16,OK
Description	**The function is available for vehicle trackers only. The first positive input port (high level) of the device must be connected to engine detection. Otherwise, the function is unavailable.** When the status value is **1**, the parking scheduled tracking function is enabled, and GPRS data is sent at the following interval: Engine on: Data is sent at the interval of the A12 command. Engine off: Data is sent at the interval of the A15 command. When the status value is **0**, the parking scheduled tracking function is disabled, and GPRS data is sent at the following interval: Engine on: Data is sent at the interval of the A12 command. Engine off: Data is sent at the interval of the A15 command.
Example	
GPRS Sending	@@F27,353358017784062,A16,0*C3\r\n
GPRS Reply	$$F28,353358017784062,A16,OK*F6\r\n

3.8 Setting GPRS Parameters – A21

GPRS Sending	A21,*Connection mode*,*IP address*,*Port*,*APN*,*APN user name*,*APN password*
GPRS Reply	A21,OK
Description	Connection mode = 0: function disabled. Connection mode = 1: function enabled; use the TCP/IP reporting mode. Connection mode = 2: function enabled; use the UDP reporting mode. IP address: IP address or domain name. A maximum of 32 bytes are supported. Port: a maximum of 5 digits APN/APN user name/APN password: a maximum of 32 bytes respectively If no user name and password are required, leave them blank.
Example	
GPRS Sending	@@H58,353358017784062,A21,1,server.meigps.com,8800,CMNET,,*A0
GPRS Reply	$$H28,353358017784062,A21,OK*F4\r\n

3.9 Setting the Standby GPRS Server – A23

GPRS Sending	A23,*IP address*,*Port*
GPRS Reply	A23,OK
Description	IP address: a maximum of 32 bytes Port: a maximum of 5 digits When the device fails to send data to the active server set by the A21 command, data will be automatically sent to the standby server to prevent data loss.
Example	
GPRS Sending	@@S44,353358017784062,A23,182.92.69.175,8800*35\r\n
GPRS Reply	$$S28,353358017784062,A23,OK*01\r\n

3.10 Reading All Authorized Phone Numbers – A70

GPRS Sending	A70
GPRS Reply	A70,*SOS phone number 1*,*SOS phone number 2*,*SOS phone number 3*,*Listen-in phone number 1*,*Listen-in phone number 2*
Description	Read all authorized phone numbers.
Example	
GPRS Sending	@@T25, 353358017784062,A70*93\r\n
GPRS Reply	$$T85,353358017784062,A70,13811111111,13822222222,13833333333,13844444444,13855555555*21\r\n

3.11 Setting Authorized Phone Numbers – A71

GPRS Sending	A71,*Phone number 1*,*Phone number 2*,*Phone number 3*

GPRS Reply	A71,OK
Description	Phone number: A phone number has a maximum of 16 bytes. If no phone numbers are set, leave them blank. Phone numbers are empty by default. Phone number 1: SOS phone number. When you call the device by using the phone number, you will receive an SMS notification about the location, geo-fence alert and low battery alert.
Example	
GPRS Sending	@@U61,353358017784062,A71,13811111111,13822222222,13833333333*7D\r\n
GPRS Reply	$$U28,353358017784062,A71,OK*06\r\n

3.12 Setting the Smart Sleep Mode – A73

GPRS Sending	A73,*Sleep level*
GPRS Reply	A73,OK
Description	Set the automatic smart sleep mode when the device is idle. Sleep level = 0: function disabled (default). Sleep level = 1: normal sleep. The GSM module always works, and the GPS module occasionally enters the sleep mode. The device works 25% longer in the normal sleep mode than that in the normal working mode. This mode is not recommended for short interval tracking because it will affect the route precision. Sleep level = 2: deep sleep. If no event is triggered after five minutes, the GPS module will stop working and the GSM module will enter the sleep mode. Once an event is triggered, the GPS and GSM modules will be woken up. The above actions will be cycled. A heartbeat event is triggered only in the deep sleep mode and is uploaded every one hour by default. Triggering events include the SOS alert, low power alert for internal battery, low power alert for external battery, external power status, GPS antenna cut-off alert, towing alert, high temperature, low temperature, fuel theft, vehicle theft, ACC ON, (button) changes in any input port, vibration, incoming call, SMS receiving, call, and heartbeat event (GPS invalid during heartbeat wakeup).
Example	
GPRS Sending	@@W27,353358017784062,A73,2*D9\r\n
GPRS Reply	$$W28,353358017784062,A73,OK*0A\r\n

3.13 Obtaining Time from the Server – A82

GPRS Sending	A82,*Time*
GPRS Reply	A82,OK
Description	1. X = 1: Obtain time from the server. 2. Time: When the server receives the A82 command and the value of **X** is **1**, the current UTC time is sent. Decimal. For example, 09:36:30, April 19, 2016, which means **160419093630** is

	sent. Note: This command is actively sent by the server. After the device receives time from the server, the time is updated automatically.
Example	
GPRS Sending	@@V27,353358017784062,A82,1*D7\r\n
GPRS Reply	$$S38,353358017784062,A82,160419093630*D7\r\n

3.14 Setting up the BLE peripheral– ABD

GPRS Sending	ABD, Struct info
GPRS Reply	ABD,OK

Description	01 The definitions of the struct information are as follows:
	typedef struct _PlayBackrequest
	{BYTE type; // type 0： set the temperature sensor 1： set ibeacon}
	{ // when type=1，when setting IBeacon, this parenthesis parameter does not need to be sent
	Byte operation; 0: Add a temperature sensor 1: Upload a peripheral for the specified Mac 2: Read the temperature and humidity state of the specified MAC 3： Read the state of all peripherals that have been added 4： Set temperature alarm threshold 5： Set humidity alarm threshold 6: Set the control output to lose sensitivity
	{ // operation:0 Add a temperature sensor
	BYTE temp_index; // The corresponding IBEACON position needs to be set
	BYTE upload_flag; // Enable upload flag
	BYTE temp_name[16]; // If the length is insufficient, add 0
	BYTE temp_mac[6];
	sint16 temp1_H; // Temperature, high temperature alarm -40 ~ +60
	sint16 temp1_L; // Temperature, low temperature alarm -40 ~ +60
	sint16 humidity_H; // Humidity. High humidity 0 ~ 100
	sint16 humidity_L; // Humidity. Low humidity 0 ~ 100}
	{ // operation:1 Specifies peripheral uploads for the MAC
	BYTE upload_flag; // Enable upload flag
	BYTE temp_mac[6];}
	{ // operation: 2 Gets the temperature and humidity status of the specified MAC
	BYTE temp_mac[6];}
	{ // operation: 3 Gets the state of all peripherals that have been added
	//空}
	{ // operation: 4 Set temperature alarm threshold
	BYTE temp_mac[6];
	sint16 temp1_H; // Temperature, high temperature alarm -40 ~

+60

sint16 temp1_L; // Temperature, low temperature alarm -40 ~ +60}

{ // operation: 5 Set humidity alarm threshold

BYTE temp_mac[6];

sint16 humidity_H; // Humidity. High humidity 0 ~ 100

sint16 humidity_L; // Humidity. Low humidity 0 ~ 100}

{// operation: 6 Set the control output to lose sensitivity

word lose_sec; // unit: S 0~65535

byte control_out_type; // bit:0 lose

// bit:1 high temperature

// bit:2 low temperature

// bit:3 High humidity

// bit:4 Low humidity

byte control_out; // bit:0 Control output 1

// bit:1 Control output 2}

{ // type When the type is 0 , that is ,when setting the temperature sensor, the parenthesis parameter does not need to be issued

Byte operation; 0: Add iBeacon sensor; 1: Pause uploading a peripheral for the specified Mac ; 2: Gets the state of the specified MAC ; 3 : Gets the state of all peripherals that have been added 4: Set the control output to lose sensitivity;

{ When operation:0, add the IBeacon sensor

BYTE ibeacon_index; // The location of the corresponding ibeacon that needs to be set

BYTE upload_flag; // Enable upload flag

BYTE ibeacon_name[16]; // If the length is insufficient, add 0

BYTE ibeacon_mac[6];}

{ // operation:1 Pause the specified Mac upload

BYTE upload_flag; //Enable upload flag

BYTE temp_mac[6];}

{ // operation: 2 Read the state of the specified MAC//

BYTE temp_mac[6];}

{ // operation:3 Reads the state of all peripherals that have been added

// blank}

{// operation: 4 Set the sensitivity that controls the lost output

word lose_sec; // unit:S Lost alarm time, // unit: S 0~65535

byte control_out_type; // bit:0 lose

byte control_out; // bit:0 control output1

//bit:1 control output2}

}

3.15 Temporarily Enabling the Bluetooth Slave Mode – AD9

GPRS Sending	AD9,*Time*
GPRS Reply	AD9,OK/*Error code*
Description	Time: indicates the time for enabling the Bluetooth slave mode. Unit: minute. Value range: 1–10. (Used to connect to an app on a phone.)
Example	
GPRS Sending	@@V27,353358017784062,AD9,1*EA\r\n
GPRS Reply	$$W28,353358017784062,AD9,OK*1D\r\n

3.16 Setting a Geo-Fence – B05

GPRS Sending	B05,*Geo-fence number*,*Latitude*,*Longitude*,*Radius*,*Enter Geo-fence alert*,*Exit Geo-fence alert*
GPRS Reply	B05,OK
Description	Geo-fence number: The parameter value ranges from **1** to **8**. A maximum of eight geo-fences can be set. Latitude: Indicates the latitude of the geo-fence center; decimal; accurate to six digits placed after the decimal point. If there are only four digits placed after the decimal point, add two digits 0. Otherwise, the command cannot be used successfully. Longitude: Indicates the longitude of the geo-fence center; decimal; accurate to six digits placed after the decimal point. If there are only four digits placed after the decimal point, add two digits 0. Otherwise, the command cannot be used successfully. Radius: The parameter value ranges from **1** to **4294967295**. Unit: meter. Take coordinates of the above latitude and longitude as the center point and draw a circle with this radius. Enter Geo-fence alert = 0: function disabled. Enter Geo-fence alert = 1: function enabled. Exit Geo-fence alert = 0: function disabled. Exit Geo-fence alert = 1: function enabled.
Example	
GPRS Sending	@@H57,353358017784062,B05,1,22.913191,114.079882,1000,0,1*96\r\n
GPRS Reply	$$H28,353358017784062,B05,OK*F7\r\n *When the device exits the geo-fence (latitude: 22.913191; longitude: 114.079882; radius: 1000 meters), it will send the following GPRS data packet about an Exit Geo-fence alert to the server:* *$$J132,353358017784062,AAA,21,22.918046,114.089726,080229123812,A,10,22,12,32,1,21,6667,847,0|0|10133|4110,0000,124|181|183|2714|922,*5A\r\n*

3.17 Deleting a Geo-Fence – B06

GPRS Sending	B06,*Geo-fence number*
GPRS Reply	B06,OK
Description	Geo-fence number: The parameter value ranges from **1** to **8**. Only one geo-fence can be

	deleted each time by sending an SMS or GPRS command.
Example	
GPRS Sending	@@J27,353358017784062,B06,1*C8\r\n
GPRS Reply	$$J28,353358017784062,B06,OK*FA\r\n *After the above command is sent successfully, the first geo-fence will be deleted.*

3.18 Setting the Speeding Alert – B07

GPRS Sending	B07,*Driving speed*
GPRS Reply	B07,OK
Description	Driving speed = 0: function disabled (default). Driving speed = [1…255]: function enabled. When the driving speed reaches the preset value, a speeding alert will be sent. Unit: km/h.
Example	
GPRS Sending	@@P28,353358017784062,B07,60*05\r\n
GPRS Reply	$$P28,353358017784062,B07,OK*01\r\n

3.19 Setting the Towing Alert – B08

GPRS Sending	B08,*Consecutive vibration time*
GPRS Reply	B08,OK
Description	When the device's consecutive vibration time exceeds the preset value, the device will send an alert to an authorized phone number or the server. Before using the towing alert function, use the A73 command to set the smart sleep level to **2** and use the B08 command to set the consecutive vibration time. Otherwise, the towing alert function will be unavailable. Consecutive vibration time = 0: function disabled (default). Consecutive vibration time = [1…255]: function enabled. Set the consecutive vibration time. Unit: second.
Example	
GPRS Sending	@@I27,353358017784062,B08,3*CB\r\n
GPRS Reply	$$I28,353358017784062,B08,OK*FB\r\n *After the above command is sent successfully, if the device vibrates for more than three consecutive seconds, it will send the following GPRS data packet about a towing alert to the server:* *$$K133,353358017784062,AAA,36,22.916675,114.088813,080229123718,A,10,22,61,31,1,21,6635,395,460|0|1013|4110,0000,164|185|181|2712|915,*A2*

3.20 Fast Setting the Towing Alert – B10

GPRS Sending	B10,Vibration time,Idling time
GPRS Reply	B10,OK

Description	Consecutive vibration time = 0: function disabled (default). Consecutive vibration time = [1…255]: function enabled. Set the consecutive vibration time. Unit: second. Idling time: The default parameter value is 2. Unit: minute. Idling time = 0: The deep sleep mode is disabled. Idling time = [1…255]: The power-saving function is enabled. When the idling time exceeds the preset value, the device will enter deep sleep mode.
Example	
GPRS Sending	@@I30,353358017784062,B10,10,5*4D\r\n
GPRS Reply	$$I28,353358017784062,B10,OK*F4\r\n

3.21 Setting a Polygonal Geo-Fence – B11

GPRS Sending	B11,*Geo-fence number*,*Latitude 1*,*Longitude 1*,*Latitude 2*,*Longitude 2*...*Latitude N*,*Longitude N*,*Enter Geo-fence alert*,*Exit Geo-fence alert*
GPRS Reply	B11,OK
Description	Geo-fence number: The parameter value ranges from **1** to **8**. (The maximum value varies depending on customization projects.) Latitude: accurate to 6 digits placed after the decimal point. For example, 22.512517 or -22.512517. Longitude: accurate to 6 digits placed after the decimal point. For example, 114.057200 or -114.057200. Enter Geo-fence alert: The parameter value is **0** or **1**. **0**: An alert will not be generated when the device enters the geo-fence. **1**: An alert will be generated when the device enters the geo-fence. Exit Geo-fence alert: The parameter value is **0** or **1**. **0**: An alert will not be generated when the device exits the geo-fence. **1**: An alert will be generated when the device exits the geo-fence. If the command only contains the parameter **Geo-fence number**, related geo-fences will be deleted. If the geo-fence is circular, the command to be sent is **B11,*Geo-fence number*,*Latitude*,*Longitude*,*Radius (meter)*,*Enter Geo-fence alert*,*Exit Geo-fence alert***.
Example	
GPRS Sending	@@I59,353358017784062,B11,5,31,22.913458,114.083183,100,1,1*F5\r\n
GPRS Reply	$$I28,353358017784062,B11,OK*F5\r\n

3.22 Setting the Idling Alert – B14

GPRS Sending	B14,*Consecutive speed time (second)*,*Speed (km/h)*,*Alert time (second)*
GPRS Reply	B14,OK
Description	Consecutive speed time: The parameter value ranges from **0** to **60000**. Unit: second. The default parameter value is **180**. Speed: The parameter value ranges from **0** to **200**. Unit: km/h. The default parameter value is **5**.

	Alert time: The parameter value ranges from **0** to **60000**. Unit: second. The default parameter value is **120**. If you want to read the parameters, send **B14**.
Example	
GPRS Sending	@@I27,353358017784062,B14,180,5,120*AE\r\n
GPRS Reply	$$I28,353358017784062,B14,OK*F8\r\n

3.23 Setting Driver Fatigue Parameters – B15

GPRS Sending	B15,*Consecutive driving time (minute),Alert time (second),Rest time (minute)*
GPRS Reply	B15,OK
Description	Consecutive driving time: The parameter value ranges from **0** to **1000**. Unit: minute. The default parameter value is **240**. Alert time: The parameter value ranges from **0** to **60000**. Unit: second. The default parameter value is **300**. Rest time: The parameter value ranges from **0** to **1000**. Unit: minute. The default parameter value is **20**. If you want to read the parameters, send **B15**.
Example	
GPRS Sending	@@I27,353358017784062,B15,240,300,20*79\r\n
GPRS Reply	$$I28,353358017784062,B15,OK*F9\r\n

3.24 Setting Filtering Time of an Input Port – B26

GPRS Sending	B26,1:T1,2:T2,...n:Tn
GPRS Reply	B26,OK
Description	n: The parameter value ranges from 1 to 5, which corresponds to input ports 1–5. Tn: Indicates the filtering time. Value range: 0–65535; unit: x10ms You can set one or multiple input ports at a time. If you want to read filtering time of an input port, send B26.
Example	
GPRS Sending	@@C30,353358017784062,B26,1:10*58\r\n
GPRS Reply	$$C28,353358017784062,B26,OK*F5\r\n

3.25 Turning off the LED Indicator – B31

GPRS Sending	B31,*A*
GPRS Reply	B31,OK
Description	A = 00: The device's LED indicator is turned on (default). The LED indicator can be used to confirm the device's running status. A = 10: The device's LED indicator is turned off.
Example	

GPRS Sending	@@J28,353358017784062,B31,10*F7\r\n
GPRS Reply	$$J28,353358017784062,B31,OK*F8\r\n

3.26 Setting a Log Interval – B34

GPRS Sending	B34,*Log interval*
GPRS Reply	B34,OK
Description	Set the interval for recording data to device's memory when the GPS signal is valid. When there is no GPS signal, data will not be recorded. **Recorded logs can only be read by GPSLog or Meitrack Manager software.** Log interval = 0: function disabled (default). Log interval = [1…65535]: function enabled. Set the log interval. Unit: second.
Example	
GPRS Sending	@@N28,353358017784062,B34,60*03\r\n
GPRS Reply	$$N28,353358017784062,B34,OK*FF\r\n

3.27 Setting the SMS Time Zone – B35

GPRS Sending	B35,*SMS minute*
GPRS Reply	B35,OK
Description	The default time zone of the device is GMT 0. You can run the B35 command to change the time zone of an SMS report to the local time zone. The time zone of an SMS report is different from that of a GPRS data packet. SMS minute = 0: The time zone is GMT 0. SMS minute = [-32768...32767], set time zones.
Example	
GPRS Sending	@@O29,353358017784062,B35,480*3C\r\n
GPRS Reply	$$O28,353358017784062,B35,OK*01\r\n *After the above command is sent successfully, the device's SMS time zone will be changed to UTC+08:00 (China time zone).*

3.28 Setting the GPRS Time Zone – B36

GPRS Sending	B36,*GPRS minute*
GPRS Reply	B36,OK
Description	GPRS minute = 0: The time zone is GMT 0 (default). The MS02 platform can automatically detect users' time zone, so that the GPRS time zone does not need to be changed. Please maintain the default GPRS time zone at GMT 0. If the GPRS time zone is changed, data will be inaccurate. GPRS minute = [-32768...32767]: Set time zones.
Example	
GPRS Sending	@@P29,353358017784062,B36,480*3E\r\n

GPRS Reply	$$P28,353358017784062,B36,OK*03\r\n *After the above command is sent successfully, the GPRS time zone will be changed to UTC+08:00 (China time zone).*

3.29 Setting the Auto Sleep Function – B37

GPRS Sending	B37,*X*
GPRS Reply	B37,OK
Description	X: The parameter value is **0** or **1**. **0**: The auto sleep mode is disabled. The default parameter value is **1**. The device will automatically enter the deep sleep mode when the following conditions are met: (a) When the device detects that the ACC is off and it stops moving; and (b) The voltage of the external power supply is smaller than 11.4 V or 24.8 V.
Example	
GPRS Sending	@@P27,353358017784062,B37,0*D1\r\n
GPRS Reply	$$P28,353358017784062,B37,OK*04\r\n

3.30 Determining Vehicle Status by ACC Status – B60

GPRS Sending	B60,*X*
GPRS Reply	B60,OK
Description	X = 1: function enabled. When the device detects that the ACC is off, the device's longitude and latitude will not be updated. This helps avoid static drift. X = 0: function disabled (default). The first positive input of the device is connected to engine detection by default.
Example	
GPRS Sending	@@U27,353358017784062,B60,1*D3\r\n
GPRS Reply	$$U28,353358017784062,B60,OK*05\r\n

3.31 Setting Event Authorization – B99

GPRS Sending	B99,<SMS>/<0>,<*Phone number location*>,<*Authorized phone number*>,<*Operation code*>, [*Event code 1*]…[*Event code n*] B99,<CALL>/<1>,<*Phone number location*>,<*Authorized phone number*>,<*Operation code*>, [*Event code 1*]…[*Event code n*] B99,<GPRS>/<2>,<*Operation code*>, [*Event code 1*]…[*Event code n*] 0000,B99,<CAMERA>/<3>,<*Operation code*>, [*Event code 1*]…[*Event code n*] B99,<BUZZER>/<4>,<*Operation code*>, [*Event code 1*]…[*Event code n*]. B99,<OUT1>/<5>,<*Operation code*>, [*Event code 1*]…[*Event code n*] B99,<OUT2>/<6>,<*Operation code*>, [*Event code 1*]…[*Event code* n]
GPRS Reply	B99,<SMS>/<0>,<*Phone number location*>,<*Authorized phone number*>, [*Event code 1*]…[*Event code* n]

	B99,<CALL>/<1>,*<Phone number location>*,*<Authorized phone number>*, [*Event code 1*]…[*Event code* n] B99,<GPRS>/<2>,[*Event code 1*]…[*Event code* n] B99,<CAMERA>/<3>,[*Event code 1*]…[*Event code* n] B99,<BUZZER>/<4>,[*Event code 1*]…[*Event code* n] B99,<OUT1>/<5>,*<Operation code>*, [*Event code 1*]…[*Event code* n] B99,<OUT2>/<6>,*<Operation code>*, [*Event code 1*]…[*Event code* n]
Description	Fields SMS, CALL, CAMERA, GPRS, BUZZER, OUT1, and OUT2 can be presented by 0–6 in decimal string. Phone number location: value range 1~8 Operation codes GET, SET, ADD, and DEL can be presented by 0–3 in decimal string. These characters are not case-sensitive. Note: Before using the B99 command to set the SMS or CALL event code, ensure that an authorized phone number is set in advance by using the A71 command or the parameter configuration tool. The device will compare the authorized phone number included in the B99 command with the authorized phone number (excluding +86 characters) set before. If the phone numbers are the same, the new event code will be stored. If not, an SMS with error information will be sent.
Example	
GPRS Sending	@@B34,863070010825791,B99,gprs,get*BC\r\n
GPRS Reply	$$B33,863070010825791,B99,1,17,18*B5\r\n

3.32 Setting the Harsh Acceleration or Harsh Braking Alert – BBD

GPRS Sending	BBD,*X1*,*Y1*,*Z1*,*X2*,*Y2*,*Z2*
GPRS Reply	BBD,OK/*<Error code>*
Description	X1: indicates the initial speed after the vehicle accelerates suddenly. Unit: km/h. The maximum parameter value is **480**. Y1: indicates the increased speed after the vehicle accelerates suddenly. Unit: km/hr/sec. The parameter value ranges from **0** to **1000**. Z1: indicates the detection time of the harsh acceleration alert. Unit: second. The parameter value ranges from **1** to **255**. X2: indicates the initial speed after the driver brakes sharply. Unit: km/h. The maximum parameter value is **480**. Y2: indicates the increased speed after the driver brakes sharply. Unit: km/hr/sec. The parameter value ranges from **-1000** to **0**. Z2: indicates the detection time of the harsh braking alert. Unit: second. The parameter value ranges from **1** to **255**. If you want to read the command settings, send **BBD**.
Example	
GPRS Sending	@@V27,353358017784062,BBD,200,100,10,200,-100,10*D5\r\n
GPRS Reply	$$S28,353358017784062,BBD,OK*FE\r\n

3.33 Setting the Sharp Left Turn or Sharp Right Turn Alert – BC6

GPRS Sending	BC6,*A*,*B*,*C*
GPRS Reply	BC6,OK/<*Error code*>
Description	The parameter value is a decimal character. A: indicates the angle. The parameter value ranges from **0** to **359**. B: indicates the consecutive cornering time. The parameter value ranges from **2** to **100**. Unit: second. C: indicates the driving speed. The parameter value ranges from **0** to **255**. If you want to read the command settings, send **BC6**.
Example	
GPRS Sending	@@V27,353358017784062,BC6,90,10,60*D5\r\n
GPRS Reply	$$S28,353358017784062,BC6,OK*FE\r\n

3.34 Controlling Output Status – C01

GPRS Sending	C01,*Speed*,*ABCDE*
GPRS Reply	C01,OK
Description	Speed = 0: No speed limit exists. When the device receives the command, the function will take effect immediately. Speed = [1...255]: set the speed limit. When the driving speed is lower than the speed limit, the function will take effect. Unit: km/h. A = 0: Close output (output 1) - open drain. A = 1: Open output (output 1) - connect to GND. A = 2: Remain previous status. B = 0: Close output (output 2) - open drain. B = 1: Open output (output 2) - connect to GND. B = 2: Remain previous status. C = 0: Close output (output 3) - open drain. C = 1: Open output (output 3) - connect to GND. C = 2: Remain previous status. D = 0: Close output (output 4) - open drain. D = 1: Open output (output 4) - connect to GND. D = 2: Remain previous status. E = 0: Close output (output 5) - open drain. E = 1: Open output (output 5) - connect to GND. E = 2: Remain previous status.
Example	
GPRS Sending	@@M34,353358017784062,C01,20,10122*18\r\n
GPRS Reply	$$M28,353358017784062,C01,OK*F9\r\n

3.35 Setting a GPRS Event Transmission Mode – C03

GPRS Sending	C03,*X*
GPRS Reply	C03,OK
Description	X = 0: automatic event report (default) X = 1: Before another event is transmitted, existing event reports need to be confirmed and deleted on the server by using the AFF command. Select this mode when the GPRS connection uses UDP.
Example	
GPRS Sending	@@f27,353358017784062,C03,0*E1\r\n
GPRS Reply	$$f28,353358017784062,C03,OK*14\r\n

3.36 Setting the Input Mode of an Input Port – C07

GPRS Sending	C07,IN1:M1,IN2:M2 … INn:Mn
GPRS Reply	C07,IN1:C1,IN2:C2 … INn:Cn
Description	**n: indicates the input port number. The parameter value varies depending on the product model. Value range: [1...n].** **Mn: indicates the I/O port status. The parameter value is as follows:** **0: low trigger** **1: high trigger** **2: AD input** **Cn: indicates the current input mode of an input port. The parameter value is the same as that of Mn.** 1. **You can set one or multiple input ports simultaneously. If you want to read the command settings, send 0000,C07.**
Example	
GPRS Sending	@@P31,353358017784062,C07,IN1:1*CD\r\n
GPRS Reply	$$P42,353358017784062,C07,IN1:1,IN2:0,IN3:1*57\r\n

3.37 Registering a Temperature Sensor Number – C40

GPRS Sending	C40,*SN1 & Number 1,SN2 & Number 2,…,SNn & Number n*
GPRS Reply	C40,*SN1 & Number 1 & Registration result,SN2 & Number 2 & Registration result,…SNn & Number n & Registration result*
Description	Commands C40 to C46 are used to read or set a temperature sensor. Perform the following steps to connect to a temperature sensor: 1) Check whether the temperature sensor number included in GPRS data in AAA format is 0. 2) If the number is 0, it means the temperature sensor is not numbered. Then send the C42 command to read the mappings of sensor SNs and numbers. 3) Use the C40 command to number all sensors and bind related information in the database, such as the IMEI number, SN, number, and customized name, so that you can view the mappings from the platform.

	4) If a high or low temperature alert is required, send the C43 command to set the temperature threshold and customize a name. You are advised to use the installation path as the name and save the name to the database. 5) If the sensor is pulled out or replaced when the device is online, use the C46 command to check the sensor. If data is inconsistent, use the C40 and C43 commands to set data. The device uploads the current temperature data by the AAA event. If the number including in temperature data is 0, it means the temperature sensor is not registered. The platform will automatically send the C42 command to obtain the temperature sensor SN and number list. Then find out the sensor whose number is 0, and register it. n: The maximum parameter value is **8**. SN: Indicates the unique number, which is used to identify the temperature sensor; contains eight bytes; hexadecimal string. The SN is displayed on the platform like 28 1B D5 23 04 00 00 57, which is the same as that displayed on the sensor. Number: Contains one byte; hexadecimal. The parameter value ranges from **1** to **254**. Registration result: The parameter value is **0x01**, **0x02**, **0x03**, and **0x04**. **0x01**: Registration succeeded. **0x02**: The number or SN has already exists. **0x03**: The number of registered sensors has reached the upper limit. **0x04**: Registration failed. Hexadecimal.
Example (ASCII is used to display examples because hexadecimal characters cannot be displayed.)	
GPRS Sending	@@q35,012896001078259,C40,(1BD5#040000W02*50\r\n
GPRS Reply	$$q36,012896001078259,C40,(1BD5#040000W0201*1B \r\n

3.38 Deleting a Registered Temperature Sensor – C41

GPRS Sending	C41,*Number 1*,*Number 2*,…*Number n*
GPRS Reply	C41,*Number 1*,*Result*,*Number 2*,*Result*,…*Number n*,*Result*
Description	Number: Indicates the number of the registered sensor; hexadecimal. The parameter value ranges from **1** to **254**. Result: Decimal. The parameter value is **1**, **2**, and **3**. **1**: Deletion succeeded. **2**: The number does not exist. **3**: Deletion failed. (If you want to delete all registered temperature sensors, send **C41**. If deletion is successful, **OK** is returned. If not, **Error** is returned.)
Example	
GPRS Sending	@@n28,012896001078259,C41,01*19\r\n
GPRS Reply	$$n30,012896001078259,C41,01,1*37\r\n

3.39 Reading the Temperature Sensor SN and Number – C42

GPRS Sending	C42
GPRS Reply	C42,*SN1 & Number 1*,*SN2 & Number 2*,…*SNn & Number n*
Description	SNn: Indicates the unique SN of temperature sensor *n*; contains eight bytes in hexadecimal format.

	Number n: Indicates the number of sensor *n*; contains one byte in hexadecimal format. The parameter value ranges from **0** to **255**. If the parameter value is **0**, the temperature sensor is not registered.
Example (ASCII is used to display examples because hexadecimal characters cannot be displayed.)	
GPRS Sending	@@m25,012896001078259,C42*89\r\n
GPRS Reply	$$t45,012896001078259,C42,(B4v#040000R00,(1BD5#040000W00*13\r\n

3.40 Setting the Temperature Threshold and Logical Name – C43

GPRS Sending	C43,*Number 1/SN1/High temperature value 1/Low temperature value 1/High temperature alert 1/Low temperature alert 1/Logical name 1/…Number n/SNn/High temperature value n/Low temperature value n/High temperature alert n/Low temperature alert n/Logical name n*
GPRS Reply	C43,*Number 1/Result 1/Number 2/Result 2…/Number n/Result n*
Description	n: The maximum parameter value is **8**. Number: Contains one byte in hexadecimal format. SN: Indicates the unique SN of the temperature sensor; contains eight bytes in hexadecimal format. High/Low temperature value: Contains two bytes in hexadecimal format. The first byte indicates the integer part of the temperature value. When the high bit is 1, the first byte is a negative integer. When the high bit is 0, the first byte is a positive integer. The second byte indicates the decimal part of the temperature value. High temperature alert: flag; contains one byte in hexadecimal format. Low temperature alert: flag; contains one byte in hexadecimal format. Logical name (customized name): Contains 16 bytes in hexadecimal format. If the name length is less than 16 bytes, add 0x00. There are 15 English characters, and # is located at the end of English characters to distinguish the Unicode and English characters. A maximum of eight Chinese characters can be supported. Chinese characters must be Unicode characters. Result: Contains one byte in hexadecimal format. **0x01**: Settings succeeded. **0x02**: The number is not found. **0x03**: Setting failed due to incorrect parameters. Note: Separators "/" are not required between parameters.
Example (ASCII is used to display examples because hexadecimal characters cannot be displayed.)	
GPRS Sending	@@o57,012896001078259,C43,01(1BD5#040000W<0005000101T1#0000000000000000000000000000*3F
GPRS Reply	$$o28,012896001078259,C43,0101*85

3.41 Reading Temperature Sensor Parameters – C44

GPRS Sending	C44
GPRS Reply	C44,*Number 1/SN1/High temperature value 1/Low temperature value 1/High temperature alert 1/Low temperature alert 1/Logical name 1/…Number n/SNn/High temperature value n/Low temperature value n/High temperature alert n/Low*

	temperature alert n/Logical name n
Description	n: The maximum parameter value is **8**. Number: Contains one byte in hexadecimal format. SN: Indicates the unique SN of the temperature sensor; contains eight bytes in hexadecimal format. High/Low temperature value: Contains two bytes in hexadecimal format. The first byte indicates the integer part of the temperature value. When the high bit is 1, the first byte is a negative integer. When the high bit is 0, the first byte is a positive integer. The second byte indicates the decimal part of the temperature value. High temperature alert: flag; contains one byte in hexadecimal format. Low temperature alert: flag; contains one byte in hexadecimal format. Logical name (customized name): Contains 16 bytes in hexadecimal format. If the name length is less than 16 bytes, add 0x00. There are 15 English characters, and # is located at the end of English characters to distinguish the Unicode and English characters. A maximum of eight Chinese characters can be supported. Chinese characters must be Unicode characters. Note: Separators "/" are not required between parameters.
Example (ASCII is used to display examples because hexadecimal characters cannot be displayed.)	
GPRS Sending	@@r25,012896001078259,C44*90\r\n
GPRS Reply	$$r274,012896001078259,C44,01(B4v#040000R00000000000000000000000 00000000000000000000002(1BD5#040000W00000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 00000000000000000000000000000000000000000000000000000000000000000000 0000000000000000000*1E\r\n

3.42 Checking Temperature Sensor Parameters – C46

GPRS Sending	C46
GPRS Reply	C46,*Checksum*
Description	Checksum: Contains two bytes in hexadecimal format. Use CRC-CCITT to calculate parameters of eight temperature sensors (in sequence, number, SN, high temperature value, low temperature value, high temperature alert, low temperature alert, and logical name). The calculation result is used as the temperature sensor checksum.
Example	
GPRS Sending	@@i25,012896001078259,C46*89\r\n
GPRS Reply	$$i28,012896001078259,C46,12_*F1\r\n

3.43 Setting Fuel Parameters – C47

GPRS Sending	C47,*Sensor type*,*Alert percentage upper limit*,*Alert percentage lower limit*
GPRS Reply	C47,OK
Description	Sensor type: The parameter value is **0**, **1**, **2**, and **3**. **0**: No fuel level sensor is connected. **1**: A C-type fuel level sensor (AD2) is connected. **2**: A R-type fuel level sensor (AD2) is connected. **3**: A V-type fuel level sensor (AD2) is connected. The AD2 of the MVT600 and T1 is connected to the fuel level sensor by default. Alert percentage upper limit: When the parameter value is **0**, the alert will be disabled. When the parameter value is not 0, GPRS and SMS event flags will take effect automatically. When the fuel percentage is higher than or equal to the preset value, an alert will be generated and the alert event code is 52. Alert percentage lower limit: When the parameter value is **0**, the alert will be disabled. When the parameter value is not 0, GPRS and SMS event flags will take effect automatically. When the fuel percentage is lower than or equal to the preset value, an alert is generated and the alert event code is 53. If you want to modify a parameter, other parameters need to be left blank and separators "," must be remained. If you only send **C47**, all parameter values will be initialized to 0. All the parameter values are decimal characters. R-type fuel level sensor: resistive fuel level sensor C-type fuel level sensor: capacitive fuel level sensor V-type fuel level sensor: voltage-type fuel level sensor A53 and A54 are V-type fuel level sensors.
Example	
GPRS Sending	@@f33,353358017784062,C47,2,90,10*0A\r\n
GPRS Reply	$$f28,353358017784062,C47,OK*1C\r\n

3.44 Reading Fuel Parameters – C48

GPRS Sending	C48
GPRS Reply	C48,*Sensor type*,*Alert percentage upper limit*,*Alert percentage lower limit*
Description	The format of returned parameters is the same as that of the C47 command. All the parameter values are decimal characters.
Example	
GPRS Sending	@@c25,353358017784062,C48*89\r\n
GPRS Reply	$$c33,353358017784062,C48,2,90,10*D0\r\n

3.45 Setting the Fuel Theft Alert – C49

GPRS Sending	C49,*Fuel theft alert detection time*,*Fuel decrease percentage*
GPRS Reply	C49,OK
Description	Fuel theft alert detection time: Decimal; unit: minute. The parameter value ranges from **0** to **255**. The default parameter value is **3**. When the parameter value is **0**, the fuel theft

	alert is disabled. Fuel decrease percentage: Decimal. The parameter value ranges from **0** to **100**. The default parameter value is **2**. When the parameter value is **0**, the fuel theft alert is disabled. By default, when the fuel decrease percentage is 2% within three minutes, a fuel theft alert will be generated (for example, **C49,3,2**). Note: The fuel decrease percentage must be more than two times larger than that of the fuel level sensor accuracy. For example, if the fuel level sensor accuracy is 10 mm and its height is 500 mm, the recommended fuel decrease percentage is 4% (10/500 x 2).
Example	
GPRS Sending	@@c29,353358017784062,C49,3,2*4B\r\n
GPRS Reply	$$c28,353358017784062,C49,ok*5B\r\n

3.46 Setting the Positioning Mode – C67

GPRS Sending	C67,*Positioning mode*
GPRS Reply	C67,ok
Description	Positioning mode: The parameter value ranges from **0** to **3** Decimal. Positioning mode = 2: GPS positioning Positioning mode = 3: GPS+BEIDOU
Example	
GPRS Sending	@@f27,353358017784062,C67,2*ED\r\n
GPRS Reply	$$f28,353358017784062,C67,OK*1E\r\n

3.47 Setting the RS232 Serial Port and Peripheral – C70

GPRS Sending	C70,*X*,*Y*
GPRS Reply	C70,OK
Description	X: Select a serial port. X = 1: RS232-1 port X = 2: RS232-2 port Y: Select a peripheral; decimal. (The default peripheral is the camera.) Y = 0: camera Y = 1: handset Y = 2: LED display Y = 3: A21 Y = 4: RFID Y = 11: magnetic card reader
Example	
GPRS Sending	@@c31,353358017784062,C70,1,1*3B\r\n
GPRS Reply	$$c28,353358017784062,C70,ok*55\r\n

3.48 Setting the Power-off Function of the Power Button – C77

GPRS Sending	C77,*Value*

GPRS Reply	C77,OK
Description	Value = 1: You can turn off the device by its power button. Value = 0: You cannot turn off the device by its power button.
Example	
GPRS Sending	@@c27,353358017784062,C77,1*EA\r\n
GPRS Reply	$$c28,353358017784062,C77,ok*5C\r\n

3.49 Locking the 4G Network – C82

GPRS Sending	C82,*A*
GPRS Reply	C82,OK/<*Error code*>
Description	1. A = 0: Automatically select a network. Decimal. 2. A = 1: Select the 2G network. Decimal. 3. A = 2: Select the 3G network. Decimal. 4. A = 3: Select the 4G network. Decimal.
Example	
GPRS Sending	@@V27,353358017784062,C82,1*D9\r\n
GPRS Reply	$$S28,353358017784062,C82,OK*08\r\n

3.50 Locking the IoT Network – C94

GPRS Sending	C94,*A*
GPRS Reply	C94,OK/<*Error code*>
Description	1. A = 0: GSM. Decimal. 2. A = 1: LTE-CAT-M1. Decimal. 3. A = 2: LTE-CAT-NB. Decimal. 4. If you want to read the command settings, send **C94**.
Example	
GPRS Sending	@@P27,353358017784062,C94,1*D6\r\n
GPRS Reply	$$P28,353358017784062,C94,OK*08\r\n

3.51 Transparently Transmitting Bluetooth Data – CC4

GPRS Sending	CC4,<*Transparently transmitted data*>
GPRS Reply	CC4,OK/<*Error code*>/<*Transparently transmitted data*>
Description	Transparently transmitted data: supports up to 1,394 bytes.
Example	
GPRS Sending	@@P29,353358017784062,CC4,123*47\r\n
GPRS Reply	$$P28,353358017784062,CC4,OK*47\r\n

3.52 Setting Bluetooth Pairing – CC5

GPRS Sending	CC5,*Broadcast name*,*Broadcast interval*,*Pairing password*
GPRS Reply	CC5,OK/*<Error code>*
Description	1. Broadcast name: contains a maximum of 16 bytes. 2. Broadcast interval: 20–10240. Unit: ms. 3. Pairing password: contains a maximum of six digits. 4. The Bluetooth information can be cleared after the command **CC5,,,*Pairing password*** is sent. 5. If you want to clear all parameters, leave all parameters blank, remain commas, and reset Bluetooth. 6. If you want to read the command settings, send **CC5**.
Example	
GPRS Sending	@@P39,353358017784062,CC5,123,20,000000*23\r\n
GPRS Reply	$$P28,353358017784062,CC5,OK*13\r\n

3.53 Deleting an Event in the Buffer – CFF

GPRS Sending	CFF,*Quantity of deleted data*
GPRS Reply	CFF,*CFF data packet*
Description	Quantity of deleted data: hexadecimal. In general, the number is 1. The data identifiers from the device and server must be consistent. Otherwise, data will not be deleted from the device. If data is transmitted in CFF format, send **CFF,FFFF** to delete all cache records and ensure that the data packet number sent from the server is consistent with that sent from the device. When the GPRS connection mode is UDP, send the CFF command to confirm that the server has received the data.

3.54 Authorizing an IButton Key – D10

GPRS Sending	D10,*ID(1)*,*ID(2)*,...,*ID(n)*
GPRS Reply	D10,OK
Description	ID(n): Indicates the number of the ID to be authorized. The parameter value ranges from **1** to **4294967295**. Decimal. A maximum of 50 ID can be authorized at a time.
Example	
GPRS Sending	@@f43,353358017784062,D10,13737431,13737461*17\r\n
GPRS Reply	$$f28,353358017784062,D10,OK*13\r\n

3.55 Authorizing iButton Keys in Batches – D11

GPRS Sending	D11,*IButton start number*,*n*
GPRS Reply	D11,OK

Description	IButton ID start number: The parameter value ranges from **1** to **4294967295**. Decimal. n: Indicates the number of ID to be authorized in batches. Decimal. The parameter value ranges from **1** to **128**.
Example	
GPRS Sending	@@e36,353358017784062,D11,13737431,1*AA\r\n
GPRS Reply	$$e28,353358017784062,D11,OK*13\r\n

3.56 Checking iButton Authorization – D12

GPRS Sending	D12,iButton ID
GPRS Reply	D12,*n*
Description	ID: The parameter value ranges from **1** to **4294967295**. Decimal. n: When the parameter value is not 0, the IButton is authorized. When the parameter value is **0**, the IButton is not authorized.
Example	
GPRS Sending	@@C34,353358017784062,D12,13737431*2A\r\n
GPRS Reply	$$C27,353358017784062,D12,0*87\r\n

3.57 Reading an Authorized IButton Key – D13

GPRS Sending	D13,*iButton data packet start number*
GPRS Reply	D13,*Number of ID data packets*,*Current ID data packet number*,*ID(1)ID(2)…ID(n)*
Description	ID data packet start number: Indicates the start number of an ID data packet. The minimum parameter value is **0**. For example, when the parameter value is **0**, the package list will be obtained from the first ID data packet. When the parameter value is **4**, the package list will be obtained from the fifth ID data packet. Number of ID data packets: Indicates the number of authorized ID data packets. One ID data packet includes a maximum of 100 ID ID numbers. The minimum parameter value is **0**. ID(n): An ID ID number contains eight hexadecimal characters.
Example	
GPRS Sending	@@w27,353358017784062,D13,0*F4\r\n
GPRS Reply	The example cannot be displayed here because of hexadecimal characters. Please do a test based on actual conditions.

3.58 Deleting an Authorized IButton Key – D14

GPRS Sending	D14,*ID(1)*,*ID(2)*,…,*ID(n)*
GPRS Reply	D14,OK
Description	ID(n): Indicates the ID number to be deleted. The parameter value ranges from **1** to **4294967295**. Decimal. A maximum of 50 IButtons can be deleted at a time. One SMS (including the protocol

	part) cannot exceed 140 bytes.
Example	
GPRS Sending	@@Q34,353358017784062,D14,13723455*3B\r\n
GPRS Reply	$$Q28,353358017784062,D14,OK*02\r\n

3.59 Deleting Authorized iButton Keys in Batches – D15

GPRS Sending	D15,*IButton start number*,*n*
GPRS Reply	D15,OK
Description	IButton start number: The parameter value ranges from **1** to **4294967295**. Decimal. n: Indicates the number of IButtons to be deleted in batches. Decimal. The parameter value ranges from **1** to **128**. When the IButton start number is a value ranging from **1** to **4294967295** and **n** is greater than or equal to **65536**, all authorized IButtons will be deleted.
Example	
GPRS Sending	@@K36,353358017784062,D15,13723455,3*97\r\n
GPRS Reply	$$K28,353358017784062,D15,OK*FD\r\n

3.60 Checking the Checksum of the Authorized iButton ID Database – D16

GPRS Sending	D16
GPRS Reply	D15,XOR
Description	This command is used to check whether the existing authorized ID database is consistent with that recorded in the server. When the device receives the D16 command, the XOR result of all authorized ID numbers is regarded as the database checksum for responding. After the server receives the checksum, compare it with the XOR result of all authorized ID numbers recorded in the server. If the result is the same, the existing authorized ID database is consistent with that recorded in the server. Otherwise, data errors occur in the authorized ID database.
Example	
GPRS Sending	@@u25,353358017784062,D16*97\r\n
GPRS Reply	$$u28,353358017784062,D16,18*F7\r\n

3.61 Setting the Maintenance Mileage-D65

GPRS Sending	D65, Mileage point 1<,Mileage point 2><,Mileage point 3><,Mileage point 4><,Mileage point 5><,Mileage point 6><,Mileage point 7><,Mileage point 8>
GPRS Reply	D65,OK
Description	Set 8 mileage points. Otherwise, the function will be unavailable. Mileage point: The parameter value ranges from 0 to 4294967295. Unit: meter. Note: When the tracker detects that the mileage nearly reaches the preset value, a maintenance mileage reminder will be generated in advance. Users will be alerted by

	default when the mileage is 300 km ahead of the preset value. If a maintenance time alert is generated in advance, the maintenance mileage will skip to the next one automatically.
Example	Set 8 mileage points： 30000, 50000,60000,70000,80000,90000,100000,110000
GPRS Sending	@@V75,353358017784062,D65,30000,50000,60000,70000,80000,90000,100000,110000*EA\r\n
GPRS Reply	$$V28,353358017784062,D65,OK*OD\r\n

3.62 Setting Maintenance Time-D66

GPRS Sending	D66, Time point 1<,Time point 2><,Time point 3><,Time point 4><,Time point 5><,Time point 6><,Time point 7><,Time point 8>
GPRS Reply	D66,OK
Description	Set 8 time points. Otherwise, the function will be unavailable. Time point: The parameter value ranges from 0 to 4294967295. Unit: second. Note: When the tracker detects that the time nearly reaches the preset value, a maintenance time reminder will be generated in advance. Users will be alerted by default when the time is 7 days ahead of the preset value. If a maintenance mileage alert is generated in advance, the maintenance time will skip to the next one automatically.
Example	
GPRS Sending	@@V65,353358017784062,D66,8726,8816,8906,8996,9086,9176,9266,9356*A2\r\n
GPRS Reply	$$V28,353358017784062,D66OK*E2\r\n

3.63 Setting GPS Data Filtering – D71

GPRS Sending	D71,X,Y1,Y2,Y3,Y4
GPRS Reply	D71,OK/<Error code>
Description	X: Whether to enable the GPS data filtering function. 1: Enable the function. 0: Disable the function (default). Y1: Indicates the minimum value of the driving speed. Value range: 0–999 km/h. Y2: Indicates the maximum value of the driving speed. Value range: 0–999 km/h. Y3: Indicates the number of satellites. Value range: 0–99. When the number of satellites is greater than Y3, GPS data will be updated. Y4: Indicates the positioning accuracy. Unit: x10. Value range: 0–999. When the positioning accuracy value is less than Y4, GPS data will be updated. Note: When the GPS data filtering function is enabled, if all conditions of Y1, Y2, Y3 and Y4 are met, GPS data will be updated.
Example	
GPRS Sending	@@I40,865328022075252,0D71,1,5,255,4,0.4*38\r\n
GPRS Reply	$$I28,865328022075252,D71,OK*F8\r\n

3.64 Setting an Output Port– D72

GPRS Sending	D72,X,Y1,Y2,Y3,Y4

GPRS Reply	D72,OK/<错误代码>
Description	X: Select an output port. 1: output 1. 2: output 2. Y1: indicates the output time when an event is triggered. Unit: 10 ms. Value range: 0–4294967295. Y2: The parameter value is 0, 1, and 2. 0: High level output. 1: Low level output. 2: PWM wave output. Y3: indicates the PWM duty cycle. Value range: 0–100. Y4: indicates the PWM period. Unit: μs. Value range: 2000–50000000.
Example	
GPRS Sending	@@s42,865328022075252,0D72,1,100,0,0,10000*B0\r\n
GPRS Reply	$$s28,865328022075252,D72,OK*23\r\n

3.65 Allocating GPRS Cache and GPS Log Storage Space - D73

GPRS Sending	D73,X,Y
GPRS Reply	D73,OK/<Error code>
Description	X: Set the storage percentage of GPRS cache. The parameter value is a decimal character. Y: Set the storage percentage of GPS logs. The parameter value is a decimal character. The sum of X and Y must be 100.
Example	
GPRS Sending	@@Q31,865328022075252,D73,50,50*90\r\n
GPRS Reply	$$Q28,865328022075252,D73,OK*02\r\n

3.66 Displaying the Actual Number of Satellites When the GPS Signal Strength is Low – DDB

GPRS Sending	DDB,*A*
GPRS Reply	DDB,OK
Description	A: Whether to enable the function. **0**: Disable the function. **1**: Enable the function. Note: When the function is disabled and the GPS signal strength is lower than 4, the number of satellites is 0 by default and the actual number of satellites is not uploaded. Because the number of GPS satellites is less than 4 and GPS signals are ultra-low.
Example	
GPRS Sending	@@F27,56554895644558545,DDB,1*65\r\n
GPRS Reply	$$F28,56554895644558545,DDB,OK*97\r\n

3.67 Obtaining Real-time GNSS Data – DDC

GPRS Sending	DDC,*A*
GPRS Reply	DDC,*GPS data*
Description	1. A: indicates the time for obtaining real-time GNSS raw data. Value range: 0–600. Unit: second. 2. If you want to obtain the remaining time of the current window, send **DDC**.

Example	
GPRS Sending	@@F29,56554895644558545,DDC,120*CA\r\n
GPRS Reply	$$F998,864507031322448,DDC,$GPGGA,060413.000,2230.5973,N,11403.3781,E,1,5,2.72,31.4,M,-2.0,M,,*3E\r\n

3.68 Using the NITZ Time or Not – DDD

GPRS Sending	DDD,*Mode*,[*NTP synchronization timeout time*,*IP1*,*PORT1*,[*IP2*,*PORT2*]]
GPRS Reply	DDD,OK
Description	1. This command is used to set the system time calibration mode. The value ranges from **0** to **6**. **0**: GNSS **1**: NITZ + NTP **2**: NTP **3**: NITZ **4**: GNSS + NITZ **5**: GNSS + NTP **6**: GNSS + NTP + NITZ 2. NTP synchronization timeout time: The value ranges from **1** to **65535**. Unit: second. 3. IP1/IP2: indicates the IP address or domain name. The parameter contains a maximum of 32 characters. 4. PORT1/PORT2: indicates the port to be connected. Decimal. The value ranges from **2** to **65534**.
Example	
GPRS Sending	@@F27,863921032192554,DDD,1*E6\r\n
GPRS Reply	$$F28,863921032192554,DDD,OK*18\r\n

3.69 Detecting the Driving Speed (less than 5km/h) or not During Fuel Level Detection – DF3

GPRS Sending	DF3,*A*,*B*,*C*,*D*
GPRS Reply	DF3,OK/ERROR
Description	1. A: Detect the driving speed or not during fuel theft detection. **1**: Detect the driving speed during fuel theft detection. **0**: Do not detect the driving speed during fuel theft detection. 2. B: Detect the driving speed or not during fuel filling detection. **1**: Detect the driving speed during fuel filling detection. **0**: Do not detect the driving speed during fuel filling detection. 3. C: Detect the driving speed or not during high fuel detection. **1**: Detect the driving speed during high fuel detection. **0**: Do not detect the driving speed during high fuel detection. 4. D: Detect the driving speed or not during low fuel detection. **1**: Detect the driving speed during low fuel detection. **0**: Do not detect the driving speed during low fuel detection.
Example	
GPRS Sending	@@P33,896556551551545,DF3,1,0,1,1*03\r\n
GPRS Reply	$$P28,896556551551545,DF3,OK*22\r\n

3.70 Obtaining the Terminal Command List - E04

GPRS Sending	E04
GPRS Reply	E04,<command1>,<command2>,...
Description	Reply to the list of supported commands
Example	
GPRS Sending	@@P27,353358017784062,E04*D1\r\n
GPRS Reply	$$P28,353358017784062,E04,A10,A11....*03\r\n

3.71 Reading Device's Firmware Version and SN – E91

GPRS Sending	E91
GPRS Reply	E91,*Version*,*SN*
Description	This command is used to read the device's firmware version and SN.
Example	
GPRS Sending	@@W25,353358017784062,E91*7D\r\n
GPRS Reply	$$W38,353358017784062,FWV1.00,12345678*1C\r\n

3.72 Restarting the GSM and GPS Module – F00

GPRS Sending	F00,*GSM*,*GPS*
GPRS Reply	F00,OK
Description	GSM: The parameter value is **0** or **1**. **0**: no operation. **1**: Restart the GSM module. GPS: The parameter value is **0** or **1**. **0**: no operation. **1**: Restart the GPS module. This command is used to restart the GSM and GPS modules.
Example	
GPRS Sending	@@j25,353358017784062,F00*87\r\n
GPRS Reply	$$j28,353358017784062,F00,OK*18\r\n

3.73 Restarting the GSM Module – F01

GPRS Sending	F01
GPRS Reply	F01,OK
Description	This command is used to restart the GSM module.
Example	
GPRS Sending	@@j25,353358017784062,F01*88\r\n
GPRS Reply	$$j28,353358017784062,F01,OK*19\r\n

3.74 Restarting the GPS Module – F02

GPRS Sending	F02
GPRS Reply	F02,OK
Description	This command is used to restart the GPS module.
Example	

GPRS Sending	@@Z25,353358017784062,F02*79\r\n
GPRS Reply	$$Z28,353358017784062,F02,OK*0A\r\n

3.75 Setting the Mileage and Run Time – F08

GPRS Sending	F08,*Run time*,*Mileage*
GPRS Reply	F08,OK
Description	Run time: The parameter value ranges from **0** to **4294967295**. Decimal; unit: second. If you do not want to set the parameter, leave it blank. Mileage: The parameter value ranges from **0** to **4294967295**.Decimal; unit: meter. If you do not want to set the parameter, leave it blank.
Example	
GPRS Sending	@@D40,353358017784062,F08,0,4825000*51\r\n
GPRS Reply	$$D28,353358017784062,F08,OK*FA\r\n

3.76 Deleting SMS or GPRS Cache Data – F09

GPRS Sending	F09,*Number*
GPRS Reply	F09,OK
Description	Number = 1: SMS cache data to be sent is deleted. Number = 2: GPRS cache data to be sent is deleted. Number= 3: SMS and GPRS cache data to be sent is deleted.
Example	
GPRS Sending	@@E27,353358017784062,F09,1*CA\r\n
GPRS Reply	$$E28,353358017784062,F09,OK*FC\r\n

3.77 Restoring Initial Settings – F11

GPRS Sending	F11
GPRS Reply	F11,OK
Description	This command is used to restore initial settings except the SMS password.
Example	
GPRS Sending	@@[25,353358017784062,F11*7A\r\n
GPRS Reply	$$[28,353358017784062,F11,OK*0B\r\n

3.78 Changing the Device Password – F20

GPRS Sending	F20,New password
GPRS Reply	F20,OK
Description	This command is used to change the SMS password. Note: The password contains four decimal digits only.
Example	
GPRS Sending	@@P30,353358017784062,F20,1234*61\r\n

GPRS Reply	$$P28,353358017784062,F20,OK*00\r\n

If you have any questions, do not hesitate to email us at info@meitrack.com.