iStartek GPRS Protocol

Between GPS Tracker and Server

Version 2.8

Contents

I Packet Format

GPRS packet format is as follows:

From server to tracker:

@@<L><ID><command><data><checksum>\r\n

From tracker to server:

$$<L><ID><command><data><checksum>\r\n

Remarks:
'<' and '>' are the list separators, do not input '<' and '>' when writing a packet.
All multi-byte data complies with the sequence: High byte prior to low byte.

Item	Specification
@@	2 bytes. It is the header of packet from server to tracker. It is in ASCII code (Hex code is 0x40 0x40)
$$	2 bytes. It is the header of packet from tracker to server, It is in ASCII code (Hex code is 0x24 0x24)
L	2 bytes. It is the length of the whole packet including the header and ending character, it is in hex code
ID	7 bytes. It is in hex code, the unused byte will be stuffed by 'f' or '0xff'. For example, if ID is 12345678901, then it will be shown as follows: 0x12, 0x34, 0x56, 0x78, 0x90, 0x1f, 0xff.
Command	2 bytes. The command code is in hex code. Please refer to the command list below.
Data	Min 0 byte, max 200 bytes.
Checksum	2 bytes. It indicates CRC-CCITT(default is 0xffff) checksum of "$$<L><ID><command><data>". It is in hex code. For example: 24 24 00 11 12 34 56 78 90 1F FF 50 00 6F 35 0D 0A, which the checksum 0x6F35 = CRC-CCITT (24 24 00 11 12 34 56 78 90 1F FF 50 00)
\r\n	2 bytes. It is the ending character and in hex code (0x0d,0x0a)

II Command List

0x4101 - Track on Demand	
Packet	@@<L><ID><0x4101><checksum><\r\n>
Description	Upon receipt of the command, the tracker will send one message with its current location to the server.
Reply	$$<L><ID><0x9955><data><checksum>\r\n Please refer to annex 1 for detailed description of 'data'.
Example	40 40 00 11 12 34 56 FF FF FF FF 41 01 67 D9 0D 0A

0x4102 - Track by Time Interval	
Packet	@@<L><ID><0x4102><time interval><checksum><\r\n>
Description	time interval: 2 bytes, in hex code. Interval is in unit of 10 seconds, Max 65535*10 seconds. If set to 0x00 0x00, will stop tracking by interval. Once the interval is successfully set, the tracker will send the following position report to the server at the interval specified: $$<L><ID><0x9955><data><checksum>\r\n
Reply	$$<L><ID><0x5100><flag><timer interval><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response. timer interval: as above describe.
Example	40 40 00 13 12 34 56 FF FF FF FF 41 02 00 03 A7 30 0D 0A Set the time interval to (0x00 0x03)*10=30s

0x4103 - Set Authorization	
Packet	@@<L><ID><0x4103><input No><phone No for SMS><phone No for call><checksum><\r\n>
Description	input No: 1byte, in hex code. It supports up to 3 inputs. If input no = 0x01, set authorized phone number for input1; = 0x02, set authorized phone number for input2; = 0x03, set authorized phone number for input3. Phone No for SMS: Authorized phone number for receiving SMS alarm. Phone No for call: Authorized phone number for receiving phone call. Phone No is 16 bytes in ASCII. If the phone number is less than 16 bytes, the blank byte(s) should be stuffed by '0x00'. For example: If the authorized number is 1234567890, then it should be written as follows: 0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38 0x39 0x30 0x00 0x00 0x00 0x00 0x00 0x00 If all 16 bytes data are 0x00, the authorized number is invalid.
Reply	$$<L><ID><0x4103><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.

Example	40 40 00 32 12 34 56 FF FF FF FF 41 03 01 38 38 38 38 38 38 38 38 38 38 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 99 68 0D 0A Set authorized phone number for input1, '8888888888' is the Phone No for SMS, Phone No for call is invalid.

0x4105 - Set Speeding Alarm

Packet	@@<L><ID><0x4105><speed><checksum><\r\n>
Description	speed: 1 byte, in hex code. If speed = 0x00, to cancel over speed alarm; 0x01, speed limit is set to 10km/hour; 0x02, speed limit is set to 20km/hour; 0x03, speed limit is set to 30km/hour; 0x04, speed limit is set to 40km/hour; 0x05, speed limit is set to 50km/hour; ……. Max speed limit is 200km/hour When the tracker is over this preset speed limit, an alarm message will be sent to the server.
Reply	$$<L><ID><0x4105><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 41 05 0A D0 35 0D 0A Set the limit speed to 100 km/hour.

0x4106 - Set Movement Alarm

Packet	@@<L><ID><0x4106><radii><checksum><\r\n>
Description	If radii: =0x00, to cancel movement alert function; =0x01, it is set in a circle with current location as center and with radii=30m; =0x02, it is set in a circle with current location as center and with radii =50m; =0x03, it is set in a circle with current location as center and with radii =100m; =0x04, it is set in a circle with current location as center and with radii =200m; =0x05, it is set in a circle with current location as center and with radii =300m; =0x06, it is set in a circle with current location as center and with radii =500m; =0x07, it is set in a circle with current location as center and with radii =1000m; =0x08, it is set in a circle with current location as center and with radii =2000m.
Reply	$$<L><ID><0x4106><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 41 06 07 54 CB 0D 0A Set in a circle with current location as center and with radii =1000m.

0x4108 - Extended Settings

Packet	@@<L><ID><0x4108><ABCDEFGHIJK><checksum><\r\n>
Description	A=0x00, turn off the function of replying with an SMS position report after a call; A=0x01, turn on the function of replying with an SMS position report after a call. B=0x00, SMS location data in normal format;

	B=0x01, SMS location data in google link format. C=0x00, turn off the function to automatically hang up an incoming call; C=0x01, turn on the function to automatically hang up an incoming call. D=0x00, turn off the function of sending alarms when the tracker is turned on; D=0x01, turn on the function of sending alarms when the tracker is turned on; E: reserved and defaulted as 0x01. F=0x00, turn off the function of sending alarms when the tracker enters GPS blind area; F=0x01, turn on the function of sending alarms when the tracker enters GPS blind area. G=0x00, all LED lights work normally; G=0x01, all LED lights stop flashing when the tracker is working. H: reserved and defaulted as 0x00. I=0x00, turn off the function of sending alarms when the external power of the vehicle tracker is cut; I=0x01, turn on the function of sending alarms when the external power of the vehicle tracker is cut. J: reserved and defaulted as 0x01. K=0, turn off the function that stopping when ACC Off; K=1, turn on the function that stopping when ACC Off. "ABCDEFGHIJK" defaulted as 0x01 0x00 0x00 0x00 0x01 0x00 0x00 0x00 0x00 0x01 0x00
Reply	$$<L><ID><0x4108><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 1C 12 34 56 FF FF FF FF 41 08 01 00 00 00 01 00 00 00 00 01 01 C6 7A 0D 0A

0x4110 - Initialization Parameter

Packet	@@<L><ID><0x4110><checksum><\r\n>
Description	Make all settings (except for the password, IP, Port, APN, ID and GPRS interval) back to factory default.
Reply	$$<L><ID><0x4110><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 11 12 34 56 FF FF FF FF 41 10 65 C9 0D 0A

0x4114 - Output Control (speed is below 10km/h)

Packet	@@<L><ID><0x4114><ABCDE><checksum><\r\n>
Description	A=0x00, close output (OUT1) -open drain; A=0x01, open output (OUT1) -connect to GND; A=0x02, remain previous status.

	B=0x00, close output (OUT2) -open drain; B=0x01, open output (OUT2) -connect to GND; B=0x02, remain previous status. C=0x00, close output (OUT3) -open drain; C=0x01, open output (OUT3) -connect to GND; C=0x02, remain previous status. D=0x00, close output (OUT4) -open drain; D=0x01, open output (OUT4) -connect to GND; D=0x02, remain previous status. E=0x00, close output (OUT5) -open drain; E=0x01, open output (OUT5) -connect to GND; E=0x02, remain previous status. Output is achievable when the speed is below 10km/h and GPS is available.
Reply	$$<L><ID><0x4114><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 41 14 01 51 1C 0D 0A Output1 is achievable when the speed is below 10km/h and GPS is available.

0x4115 - Output Control (Immediate)

Packet	@@<L><ID><0x4115><ABCDE><checksum><\r\n>
Description	A=0x00, close output (OUT1) -open drain; A=0x01, open output (OUT1) -connect to GND; A=0x02, remain previous status. B=0x00, close output (OUT2) -open drain; B=0x01, open output (OUT2) -connect to GND; B=0x02, remain previous status. C=0x00, close output (OUT3) -open drain; C=0x01, open output (OUT3) -connect to GND; C=0x02, remain previous status. D=0x00, close output (OUT4) -open drain; D=0x01, open output (OUT4) -connect to GND; D=0x02, remain previous status. E=0x00, close output (OUT5) -open drain; E=0x01, open output (OUT5) -connect to GND; E=0x02, remain previous status. Output is achievable with no speed Limit, Advised Caution to using this function.
Reply	$$<L><ID><0x4115><flag><checksum>\r\n flag: 1byte, =0x00, failure response;

	=0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 41 15 01 62 2D 0D 0A Output1 is achievable Immediate.

0x4116 – Input triggered Alarms Settings

Packet	@@<L><ID><0x4116><ABCDE><checksum><\r\n>
Description	ABCDE is in ASCII code. A=IN1 B=IN2 C=IN3 D=IN4 E=IN5 When A or B or C or D or E =0(0x30), cancel the input alert; =1(0x31), to enable alert when input is active; =2(0x32), to enable alert when input is inactive; =3(0x33), to enable alert when input is either active or inactive.
Reply	$$<L><ID><0x4116><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 14 12 34 56 FF FF FF FF 41 16 31 32 33 2C 52 0D 0A enable alert when input1 is active; enable alert when input2 is inactive; enable alert when input3 is either active or inactive.

0x4126 – Set Power Down Mode

Packet	@@<L><ID><0x4126><time><checksum><\r\n>
Description	This command will set the tracker to enter Power Down mode after “time” minutes. In power down mode, GPS stops working, GSM enters standby mode and stop sending out message until it is activated or the tracker moves again. time: In ASCII code and in unit of minute. If time = 0(0x30), to close this function; = [1,255], to set this function.
Reply	$$<L><ID><0x4126><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 41 26 31 30 CD A2 0D 0A Set the tracker to enter Power Down mode after 10 minutes.

0x4130 – Set Listen-in Phone (Voice Monitoring)

Packet	@@<L><ID><0x4130><phone number><checksum><\r\n>
Description	The phone number is for wiretapping and it should be numbers or ‘+’ and numbers. Max 16 digits. In ASCII code.
Reply	$$<L><ID><0x4130><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 1B 12 34 56 FF FF FF FF 41 30 38 38 38 38 38 38 38 38 38 AD 3A 0D 0A Set authorize phone number ‘8888888888’ for wiretapping.

0x4131 – Set GPS Log by Interval	
Packet	@@<L><ID><0x4131><time><checksum><\r\n>
Description	time: ASCII code and in unit of second. If time = 0, to close this function; time = [1,65535], to set interval for logging when it gets GPS fix.
Reply	$$<L><ID><0x4131><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 41 31 31 35 5B F4 0D 0A Set interval time 15 seconds.

0x4132 – Set Time Zone for SMS	
Packet	@@<L><ID><0x4132><time><checksum><\r\n>
Description	SMS message default time is GMT. time is to set time difference in minutes to GMT. time = [-720,720], in ASCII code and in unit of minute.
Reply	$$<L><ID><0x4132><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 14 12 34 56 FF FF FF FF 41 32 34 38 30 E5 B5 0D 0A Set 480 minutes as time difference.

0x4135 – Set Sensitivity of Tremble Sensor	
Packet	@@<L><ID><0x4135><data><checksum><\r\n>
Description	data= [1,255], in ASCII code. The smaller the data, the more sensitive the sensor will be. Default is 2. Sensitivity of tremble sensor is the key parameter for sleep mode and wake up etc.
Reply	$$<L><ID><0x4135><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 41 35 35 12 1C 0D 0A Set Sensitivity to 5.

0x4136 – Set Heading Change Report	
Packet	@@<L><ID><0x4136><degree><checksum><\r\n>
Description	degree =[0,180], in ASCII code; If degree = 0, to close this function; When the heading direction of the tracker changes over the preset degree, a message with location data will be sent back to the server by GPRS. This enhances the accuracy and continuous trace when the tracker makes a direction change.
Reply	$$<L><ID><0x4136><flag><checksum>\r\n flag: 1byte,

	=0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 41 36 33 30 E8 A3 0D 0A Set degree to 30. When the tracker turns over 30 degrees, a message will be sent back to the server.

0x4142 – Set Speeding and output Alarm

Packet	@@<L><ID><0x4142><speed><checksum><\r\n>
Description	speed =[0,255], in ASCII code, and in unit of Km/h; If speed = 0, to close this function; Out1 connect to a buzzer, when the tracker is over this preset speed limit, an alarm message will be sent to the server, and Out1 will be activated to alarm.
Reply	$$<L><ID><0x4142><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 41 42 38 30 30 91 0D 0A Set limit speed to 80 Km/h.

0x4155 – Set GPRS Parameters

Packet	@@<L><ID><0x4155><mode,IP,port,APN,APN user name,APN password><checksum><\r\n> @@<L><ID><0x4155><mode,IP,port><checksum><\r\n>
Description	mode: =0, disable GPRS Communication =1, TCP =2, UDP IP: could be IP or domain name; The command could set <mode,IP,port,APN,APN user name,APN password> or only <mode,IP,port>.
Reply	$$<L><ID><0x4155><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 2E 12 34 56 FF FF FF FF 41 55 31 2C 69 73 74 61 72 74 72 61 63 6B 65 72 2E 63 6F 6D 2C 38 30 31 31 2C 43 4D 4E 45 54 B7 96 0D 0A The command will set communication mode as TCP, IP as istartracker.com, port as 8011, APN as CMNET.

0x4302 – Set Geo-fence Alarm

Packet	@@<L><ID><0x4302><latitude,longitude,radius,in,out><checksum><\r\n>
Description	Set circle Geo-fence Alarm. When the tracker moves in or moves out of a preset circle scope, an SMS alarm will be sent to the authorized phone number for Input1, and a GPRS alarm will be sent to the server. All data are in ASCII code; Latitude is in unit of degree, ddd.dddddd format, '0' is needed to be stuffed if no value available. '-' should be added for south. Longitude is in unit of degree, dd.dddddd format, '0' is needed to be stuffed if no value available.

	'-' should be added for west. Radius=[1,4294967295], in unit of meter. in or out: =0, invalid; =1, valid. Only one alarm can be set in either Movement Alarm(0x4106) or Geo-fence Alarm(0x4302).
Reply	$$<L><ID><0x4302><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 2E 12 34 56 FF FF FF FF 43 02 32 33 2E 31 30 38 37 34 38 2C 31 30 34 2E 33 37 38 30 32 36 2C 31 30 30 30 2C 31 2C 31 A4 35 0D 0A The command will set center's latitude as 23.108748, longitude as 104.378026, radius as 1000 meters. Alarm will be triggered when entering or exiting preset scope.

0x4303 – Track by Distance Interval

Packet	@@<L><ID><0x4303><distance><checksum><\r\n>
Description	Set distance report as per pre-set interval. Tracker sends report when the it is moving, and stops sending the report when the it is stationary. Only for GPRS report. distance: ASCII code, in unit of meter; distance=[0,4294967295], if distance =0,cancel the distance interval tracking; Distance interval is suggested to be set above 300 meters.
Reply	$$<L><ID><0x4303><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 14 12 34 56 FF FF FF FF 43 03 35 30 30 45 F2 0D 0A The command will set distance interval as 500 meters.

0x4304 – Track by ACC Off Interval

Packet	@@<L><ID><0x4304><time><checksum><\r\n>
Description	time: 2 bytes, in hex code, in unit of 10 seconds, Max 65535*10 seconds. Default 0. If set to 0x00 0x00, will stop tracking by interval when ACC off.
Reply	$$<L><ID><0x4304><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 43 04 00 06 A8 5D 0D 0A The command will set time interval as (0x00 0x06)*10=60s.

0x4305 – Set ACC Off Interval Function

Packet	@@<L><ID><0x4305><flag><checksum><\r\n>
Description	flag: =0x00, cancel; =0x01, enable this function. Input3 is used as the ACC detect input. When enable this function, it will track by ACC off interval(0x4304) with ACC off, and track by time interval(0x4102) with ACC on.

	Only for GPRS report.
Reply	$$<L><ID><0x4305><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 43 05 01 0F 3E 0D 0A The command will enable this function.

0x4307 – Set GSM Jamming(Customized)

Packet	@@<L><ID><0x4307><flag><checksum><\r\n>
Description	flag: =0x00, cancel; =0x01, enable this function. When the Jamming function is enabled, when the GSM jamming is detected, there will be a GPRS event 0x57 alarm and SMS alarm, and output1 will turn on to cut the power of the oil pump or buzzer alarm. When GSM jamming disappeared, output1 will turn off.
Reply	$$<L><ID><0x4307><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 43 07 01 69 5C 0D 0A The command will enable this function.

0x4308 – Set Ext-Power Low Alarm

Packet	@@<L><ID><0x4308><Volt><checksum><\r\n>
Description	Volt=[0.0, 25.5], in ASCII code and in unit of volt, default 10.0V; =0.0, disable this function. When the Ext-power is low then volt, there is will be a GPRS alarm and a SMS alarm.
Reply	$$<L><ID><0x4308><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 15 12 34 56 FF FF FF FF 43 08 31 31 2E 30 96 11 0D 0A The command will set the volt to 11.0V.

0x4312 – Set Harsh Accelerate and Deceleration Alarm

Packet	@@<L><ID><0x4312><a1,a2,T1,T2><checksum><\r\n>
Description	a1 is the accelerated variation, default as 7 km/h/s; a2 is the decelerated variation, default as 7 km/h/s; T1 is the time for Continuous acceleration, in unit of second, default as 3s; T2 is the time for continuous deceleration, in unit of second, default as 3s; When the a1 variation is continuous for T1 time or the a2 variation is continuous for T2 time, there will be an alarm.
Reply	$$<L><ID><0x4312><flag><checksum>\r\n flag: 1byte, =0x00, failure response;

	=0x01, success response.
Example	40 40 00 18 12 34 56 FF FF FF FF 43 12 36 2C 36 2C 32 2C 32 63 57 0D 0A The command will set the a1 and a2 to 6 km/h/s, and set T1 and T2 to 2s.

0x4352 – Set Mileage

Packet	@@<L><ID><0x4352><mileage><checksum><\r\n>
Description	Set the total mileage of GPRS packets. When mileage is deleted, the server should have a corresponding program to avoid calculation mistake. Mileage=[0, 4294967295], in unit of meter and in decimal string.
Reply	$$<L><ID><0x4352><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 15 12 34 56 FF FF FF FF 43 52 31 30 30 30 D2 C1 0D 0A Set the total mileage of GPRS packets to 1000m.

0x4563– Set RFID/Ibutton ID(Customized)

Packet	@@<L><ID><0x4563><index,ID1,ID2,ID3,ID4,ID5……ID10><checksum><\r\n>
Description	Max 100 IDs can be set, one command can set 10 IDs at most. Index: decimal string, set the IDs from index position. RFID ID: decimal string, 10 digitals; Ibutton ID: Hex string, 12 characters.
Reply	$$<L><ID><0x4563><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 39 12 34 56 FF FF FF FF 45 63 33 2C 30 30 30 30 30 31 30 34 38 38 42 34 2C 30 30 30 30 30 31 32 38 35 45 34 46 2C 30 30 30 30 30 31 33 45 39 45 42 42 41 23 0D 0A Set IDs(0000010488B4,000001285E4F,0000013E9EBB) from the third position.

0x4564– Set RFID/Ibutton to control out1(Customized)

Packet	@@<L><ID><0x4564><flag><checksum><\r\n>
Description	flag: =0x00, cancel; =0x01, enable this function. When enable this function, out1 will control the power of the oil pump, it need the authorized ID to drive car.
Reply	$$<L><ID><0x4564><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 45 64 31 B3 D6 0D 0A

0x4565– Set Buzzer Time(DLT Customized)

Packet	@@<L><ID><0x4565><time><checksum><\r\n>
Description	time: decimal string.

	Time = 0: no alarm; time = [1,255]: the minutes for the alarm. Default as 10 minutes.
Reply	$$<L><ID><0x4565><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 45 65 32 30 16 FD 0D 0A Set the buzzer time to 20 minutes.

0x4566– Set Magnetic Card ID(DLT Customized)

Packet	@@<L><ID><0x4566><ID1,ID2,…IDn><checksum><\r\n>
Description	ID =[0,65535], decimal string, Support max 16 IDs settings. All ID default as 0 . Once set, the new value will cover all the previous values.
Reply	$$<L><ID><0x4566><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 16 12 34 56 FF FF FF FF 45 66 32 30 2C 32 34 89 77 0D 0A Set the ID 20 and 24.

0x4570– Get SIM Card CCID(Customized)

Packet	@@<L><ID><0x4570><checksum><\r\n>
Description	Read the CCID of SIM card.
Reply	$$<L><ID><0x4570><CCID><checksum>\r\n CCID: the CCID of SIM card.
Example	40 40 00 11 12 34 56 FF FF FF FF 45 70 C5 AB 0D 0A

0x4571– Set Only Use Track3 Data(DLT Customized)

Packet	@@<L><ID><0x4571><flag><checksum><\r\n>
Description	flag=0: read the data of track1, track2 and track3, default 0. flag=1: only read the data of track3.
Reply	$$<L><ID><0x4571><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 45 71 31 4F 50 0D 0A

0x4572 – Delete RFID/Ibutton ID(Customized)

Packet	@@<L><ID><0x4572><ID1,ID2,ID3,ID4,ID5……ID10><checksum><\r\n>
Description	One command can delete 10 IDs at most. RFID ID: decimal string, 10 digitals; Ibutton ID: Hex string, 12 characters;
Reply	$$<L><ID><0x4572><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.

Example	40 40 00 31 12 34 56 FF FF FF FF 45 72 30 30 30 30 30 31 30 34 38 38 2C 30 30 30 30 30 31 32 38 35 34 2C 30 30 30 30 30 31 33 38 34 37 09 75 0D 0A Delete IDs：0000010488,0000012854,0000013847, and other ID invariable.

0x4573 – Check RFID/Ibutton ID(Customized)

Packet	@@<L><ID><0x4573><ID><checksum><\r\n>
Description	Check the ID whether has been authorized, and each command can only check one ID. RFID ID: decimal string, 10 digitals; Ibutton ID: Hex string, 12 characters;
Reply	$$<L><ID><0x4573><flag><checksum>\r\n flag: 1byte, =0x00, unauthorized; =0x01, authorised.
Example	40 40 00 1B 12 34 56 FF FF FF FF 45 73 30 30 30 30 30 31 30 34 38 38 27 9E 0D 0A Check ID: 0000010488

0x4607 – Get USSD(Customized)

Packet	@@<L><ID><0x4607><USSD command><checksum><\r\n>
Description	Check the expenses information of SIM card.
Reply	$$<L><ID><0x4607><USSD data<checksum>\r\n USSD data: the expenses information of SIM card.
Example	40 40 00 16 12 34 56 FF FF FF FF 46 07 2A 31 32 31 23 54 69 0D 0A USSD command is ‘*121#‘

0x4901 – Reboot GSM

Packet	@@<L><ID><0x4901><checksum><\r\n>
Description	Reboot the GSM module of the tracker.
Reply	$$<L><ID><0x4901><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 11 12 34 56 FF FF FF FF 49 01 EE 70 0D 0A Reboot the GPS module of the tracker.

0x4902 – Reboot GPS

Packet	@@<L><ID><0x4902><checksum><\r\n>
Description	Reboot the GPS module of the tracker.
Reply	$$<L><ID><0x4902><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 11 12 34 56 FF FF FF FF 49 02 DE 13 0D 0A Reboot the GPS module of the tracker.

0x4903 – Reboot device

Packet	@@<L><ID><0x4903><checksum><\r\n>
Description	Reboot the device.
Reply	No reply.
Example	40 40 00 11 12 34 56 FF FF FF FF 49 03 CE 32 0D 0A Reboot the device.

0x5000 - Tracker Login Server

Packet	$$<L><ID><0x5000><checksum><\r\n>
Description	Once the IP/PORT/APN is set correctly and GPRS function is enabled, the tracker will apply for a GPRS connection by sending the login command.
Reply	/
Example	24 24 00 11 12 34 56 FF FF FF FF 50 00 8B 9B 0D 0A

0x5114 – Output Control (speed is below 20km/h)

Packet	@@<L><ID><0x5114><ABCDE><checksum><\r\n>
Description	A=0x00, close output (OUT1) -open drain; A=0x01, open output (OUT1) -connect to GND; A=0x02, remain previous status. B=0x00, close output (OUT2) -open drain; B=0x01, open output (OUT2) -connect to GND; B=0x02, remain previous status. C=0x00, close output (OUT3) -open drain; C=0x01, open output (OUT3) -connect to GND; C=0x02, remain previous status. D=0x00, close output (OUT4) -open drain; D=0x01, open output (OUT4) -connect to GND; D=0x02, remain previous status. E=0x00, close output (OUT5) -open drain; E=0x01, open output (OUT5) -connect to GND; E=0x02, remain previous status. Output is achievable when the speed is below 20km/h and GPS is available.
Reply	$$<L><ID><0x5114><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 12 12 34 56 FF FF FF FF 51 14 01 12 7F 0D 0A Output1 is achievable when the speed is below 20km/h and GPS is available.

0x5199 – Set Heartbeat Interval

Packet	@@<L><ID><0x5199><time><checksum><\r\n>
Description	Set time interval for heartbeat mode. Heartbeat will be sent when sleeping. time=[0,255], in ASCII code, in unit of minute. Default 0. If set to 0, there will be no heartbeat when sleeping.

	Heartbeat will be sent with 0x9961 command: $$<L><ID><0x9961><data><checksum>\r\n Please refer to annex 1 for detailed description of 'data'.
Reply	$$<L><ID><0x5199><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 13 12 34 56 FF FF FF FF 51 99 36 30 9D 5C 0D 0A The command will set heartbeat time interval as 60 minutes. The tracker will send heartbeat to the server every 60 minutes when sleeping.

0x5503 – Clear Message Queue

Packet	@@<L><ID><0x5503><checksum><\r\n>
Description	Clear all message queues logged when there is no GPRS coverage.
Reply	$$<L><ID><0x5503><flag><checksum>\r\n flag: 1byte, =0x00, failure response; =0x01, success response.
Example	40 40 00 11 12 34 56 FF FF FF FF 55 03 88 2C 0D 0A

0x9001 – Get device's SN, IMEI & firmware version

Packet	@@<L><ID><0x9001><checksum><\r\n>
Description	Get device's Serial Number, IMEI and firmware version.
Reply	$$<L><ID><0x9001><SN,IMEI,Version><checksum>\r\n SN,IMEI,Version: in ASCII code.
Example	40 40 00 11 12 34 56 FF FF FF FF 90 01 41 CF 0D 0A 24 24 00 31 12 34 56 FF FF FF FF 90 01 33 30 33 37 38 31 35 30 35 39 2C 33 35 33 33 35 38 30 31 31 30 31 38 31 34 35 2C 56 31 2E 33 30 FE D0 0D 0A SN=3037815059 IMEI=353358011018145 Firmware Version= V1.30

0x9002 – Get GPRS Time Interval

Packet	@@<L><ID><0x9002><checksum><\r\n>
Description	Get the GPRS Time Interval (0x4102 set).
Reply	$$<L><ID><0x9002><time><checksum>\r\n time: 2 bytes, in hex code, in unit of 10 seconds.
Example	40 40 00 11 12 34 56 FF FF FF FF 90 02 71 AC 0D 0A

0x9003 – Get Authorization number

Packet	@@<L><ID><0x9003><Input number><checksum><\r\n>
Description	Input number: 1 byte and in hex code, it should be in the range form 0x01 to 0x03.
Reply	$$<L><ID><0x9003><authorized phone number for receiving SMS><authorized phone number for receiving call><checksum>\r\n Authorized phone number: 16 bytes, in ASCII. If the phone number is less than 16 bytes, the blank byte(s) read as '0x00'.
Example	40 40 00 12 12 34 56 FF FF FF FF 90 03 01 99 5C 0D 0A

0x9016 – Get GPS Logged Data	
Packet	@@<L><ID><0x9016><device random><PC random><number(2B)><checksum>\r\n
Description	Device random and PC random are random data, 1byte, in hex code. for example, 0x03, 0x2A, etc. Number: 2 bytes, is the logged points that device delete.
Reply	$$<L><ID><0x9016>< device random><PC random<waypoints(4B)><data><checksum>\r\n Waypoints: 4bytes, in hex code, are the total number of waypoints that have been saved in memory. High byte prior to low byte. Data is the first record of the logged waypoints which is in GPRMC format.
Example	If you first send: 40 40 00 15 12 34 56 FF FF FF FF 90 16 03 15 00 01 AF AB 0D 0A Then device maybe return: 24 24 00 4B 12 34 56 FF FF FF FF 90 16 2B 15 00 00 00 59 30 33 34 34 33 33 2E 30 30 30 2C 41 2C 32 32 33 32 2E 35 32 30 39 2C 4E 2C 31 31 34 30 34 2E 36 36 35 39 2C 45 2C 30 2C 30 2C 30 31 30 38 30 39 2C 41 30 7C A0 18 0D 0A Next time you should send: 40 40 00 15 12 34 56 FF FF FF FF 90 16 2B 15 00 01 1D 26 0D 0A Where device random (0x2B) must equal to the latest received.

Annex 1: Description of data

data consists of:
hhmmss.ddd,S,xxmm.dddd,<N/S>,yyymm.dddd,<E/W>,s.s,h,ddmmyy,d.d,D*HH|HDOP|Altitude|State|AD1,AD2,Ext-AD,Bat-AD|Base ID|CSQ|Mileage|Satellite|Customized data

Remarks: ',' and '|' are list separators in ASCII.

For example:
095948.000,A,1126.6639,N,11133.3299,E,51.71,259,010318,,*08|1.4|8|2400|0000,0000,0105,029F|01A4000101D26F11|15|000001D3|06|0052E080

Details:

Parameter	Description	Example

hhmmss.dd	GMT time
	hh = hours;
	mm = minutes;
	ss = seconds;
	ddd = decimal part of seconds
		09:59:48.000

S	GPS status indicator, A = valid, V = invalid	A=Valid

xxmm.dddd	Latitude
	xx = degrees;
	mm = minutes;
	dddd = decimal part of minutes
		11 deg. 26.6639 min.

<N/S>	Either character N or character S N = North, S = South	N = North

yyymm.dddd	Longitude
	yyy = degrees;
	mm = minutes;
	dddd = decimal part of minutes
		111 deg. 33.3299 min.

<E/W>	Either character E or character W E = East, W = West	E = East
s.s	Speed, in unit of knot.	51.71 Knots
h	Heading	259 deg.
ddmmyy	Date, dd = date; mm = month; yy = year	010318=2018/03/01
d.d	Magnetic variation	Normal empty
D	Either character W or character E W = West ,E=East	Normal empty
*****	The end of the GPRMC sentence	*
HH	Checksum of the GPRMC sentence	08,
HDOP	In ASCII code, 0.1-99.9. HDOP is blank when the device has no GPS fix.	1.4
Altitude	in unit of meter	8
State	The status of input and output. in HEX String. 2 bytes with 16 bits binary: bit15...bit0. bit0: Status of Out1 If bit0=0: Out1 is closed; bit0=1:Out1 is open. bit1: Status of Out2 If bit1=0: Out2 is closed; bit1=1:Out2 is open. bit2: Status of Out3 If bit2=0: Out3 is closed; bit2=1:Out3 is open. bit3: Status of Out4 If bit3=0: Out4 is closed;bit3=1:Out4 is open. bit4: Status of Out5 If bit4=0: Out5 is closed; bit4=1:Out5 is open. bit5~bit7: Reserved and default as '0'. bit8: Status of Input1 If bit8=0: Input1 is inactive; bit8=1: Input1 is active bit9: Status of Input2 If bit9=0: Input2 is inactive; bit9=1: Input2 is active bit10: Status of Input3 If bit10=0: Input3 is inactive; bit10=1: Input3 is active bit11: Status of Input4 If bit11=0: Input4 is inactive; bit11=1: Input4 is active	2400 In binary is 0010010000000000 bit15……bit0 External power is connected, Input3 is active, input1and input2 are inactive, all output is closed.

	bit12: Status of Input5 If bit12=0: Input5 is inactive; bit12=1: Input5 is active bit13: Status of external power If bit13=0: External power disconnected; bit13=1: External power connected bit14~bit15: Reserved and default as '0'	
AD1, AD2, Ext-AD,Bat-AD	10 bit analog input, in HEX String, separated by ',' (comma). From 0000…03FF. AD1, AD2: are ADC input, Vat=(AD/1024)*6 for VT600; Vat=(AD/1024)*24 for VT900. Ext-AD: is the external power AD, Vat=(Ext-AD/1024)*48 Bat-AD: is the inside battery AD, Vat=(AD/1024)*6	0000,0000,0105,029F
Base ID	The ID of the base station . VT600: MCC MNC LAC CI each other in 4 HEX String. VT900: MCC MNC LAC in 4 HEX String, CI in 8 HEX String.	VT600: 01A4000101D26F11 VT900: 0208000332D7028A4584
CSQ	GSM CSQ value. In HEX string. From 00 to 1F	15(HEX string) = 21(decimal)
Mileage	In unit of meter. In HEX string. The total accumulated Mileage, max FFFFFFFF (4294967295) meters. Could delete by command 0x4351.	000001D3 = 467 meters
Satellite	Using Satellites. In HEX string. From 00 to 1F	06
Customized data	The data for customization. Different customization with different data.	RFID: 0052E080

Annex 2: Alarms message

Alarms message from device to server:

$$<L><ID><0x9999><Alarm code><data><checksum>\r\n

data: Please refer to annex 1 for detailed description of 'data'.

Alarm code: 1 byte in hex code, details as follows:

Alarm code	Definition
0x01	SOS button is pressed / Input 1 active
0x02	Input 2 active
0x03	Input 3 active
0x04	Input 4 active

0x05	Input 5 active
0x10	Low battery alarm
0x11	Over speed alarm
0x12	Movement alarm or exiting Geo-fence alarm
0x13	Enter Geo-fence alarm
0x14	Reboot device alarm
0x15	Enter blind area alarm
0x16	Exit blind area alarm
0x17	Ext-power low alarm
0x31	SOS button is released/Input 1 inactive
0x32	Input 2 inactive
0x33	Input 3 inactive
0x34	Input 4 inactive
0x35	Input 5 inactive
0x50	External power cut alarm
0x52	Heading change report
0x53	GPS antenna cut alarm(Customized)
0x56	Tow alarm(Customized)
0x57	GSM jamming alarm(Customized)
0x59	GSM jamming lost alarm(Customized)
0x63	Distance interval report
0x64	Fuel Theft Alarm(Customized)
0x73	Turn off alarm(Customized)
0x74	RFID/ibutton event report(Customized)
0x75	Magnetic card event report(Customized)
0x78	Illegal driving(Customized)
0x80	Harsh accelerate alarm
0x81	Harsh deceleration alarm