
I Command Format

Command format of GPRS packet is as follows:

From server to tracking unit:

@@<L><ID (7 bytes)><command (2 bytes)><data><checksum (2 bytes)>\r\n

From tracking unit to server:

\$\$<L><ID (7 bytes)><command (2 bytes)><data><checksum (2 bytes)>\r\n

Note:

Do not input '<' and '>' when writing a command.

All multi-byte data complies with the following sequence:

High byte prior to low byte.

| Item | Specification |
|----------|--|
| @@ | 2 bytes. It means the header of packet from server to tracking unit. It is in ASCII code (Hex code: 0x40) |
| \$\$ | 2 bytes. It is the header of packet from tracking unit to server, It is in ASCII code (Hex code: 0x24) |
| L | 2 bytes. It means the length of the whole packet including the header and ending character and it is in hex code |
| ID | 7 bytes, the unused byte will be stuffed by 'f' or '0xff'. It is in the format of hex code. For example, if ID is 13612345678, then it will be shown as follows: 0x13, 0x61, 0x23, 0x45, 0x67, 0x8f, 0xff. If all 7 bytes are 0xff, it is a broadcasting command. ID is in hex code |
| Command | 2 bytes. The command code is in hex code. Please refer to the command list below. |
| Data | Min 0 byte, max 100 bytes. |
| Checksum | 2 bytes. It indicates CRC-CCITT(default is 0xffff) checksum of all data (not including CRC itself and the ending character). It is in hex code. For example: 24 24 00 11 13 61 23 45 67 8f ff 50 00 05 d8 0d 0a 0x05d8 = CRC-CCITT (24 24 00 11 13 61 23 45 67 8f ff 50 00) |
| \r\n | 2 bytes. It is the ending character and in hex code (0x0d,0x0a in hex code) |

II Command List

| Command | Definition |
|------------------|---|
| 0x4000 | Server confirms tracker's login |
| 0x5000 | Tracker's login |
| 0x4101 | Request one single location report |
| 0x4102 | Set time interval for continuous tracking |
| 0x4103 | Set authorized phone number |
| 0x4104 | Reserved |
| 0x4105 | Set speed limit for over speed alarm |
| 0x4106 | Set movement alert |
| 0x4107 0x4207 | Set Geo-fence |
| 0x4108 | Set extended functions |
| 0x4109 | Reserved |
| 0x4110 | Initialize all parameters except for password, IP/PORT/APN, ID and time interval for continuous tracking. |
| 0x4111 | Reserved |
| 0x4112 | Reserved |
| 0x4113 | Set sleep mode for power saving |
| 0x4114 | Output control (safe - 10Km/h) |
| 0x5114 | Output control (safe - 20Km/h) |
| 0x4115 | Output control (general) |
| 0x4116 | Set GPRS alert for buttons or inputs |
| 0x4126 | Set power saving when tracker is immobile |
| 0x4130 | Set telephone number for wiretapping |
| 0x4131 | Set interval for logging |
| 0x4132 | Set time zone |
| 0x9000 | Reserved |
| 0x9001 | Reserved |
| 0x9002 | Read time interval of continuous tracking |
| 0x9003 | Read authorized phone number |
| 0x9004 | Reserved |
| 0x9005 | Reserved |
| 0x9007 | Reserved |
| 0x9008 | Reserved |
| 0x9011 | Reserved |
| 0x9012 | Reserved |
| 0x9013 | Reserved |
| 0x9016 | Read logging waypoints |
| 0x9955 | Single location report |
| 0x9999 | Alarm command |

III Command Details

1. Tracker's Login

Command code: **0x5000**

Once the IP/PORT/APN is correctly set by SMS and GPRS function is enabled, the tracking unit will apply for a GPRS connection by sending the login command every 30 seconds to the server until the server confirms the tracking unit's login.

Tracking unit sends the following command to the server to apply for login and GPRS connection:

`$$<L><ID><0x5000><checksum><\r\n>`

2. Server Confirms Tracker's Login

Command code: **0x4000**

Following command will be sent back to tracking unit after server receives the login command:

`@<L><ID><0x4000><Flag><checksum>\r\n`

If Flag (1 byte) is

0x00, it means login failed and will try to login again. Or

0x01, it means login succeeded.

3. Request One Single Location Report

Command code: **0x4101**

Command from server to tracking unit:

`@<L><ID><0x4101><checksum>\r\n`

Upon receipt of the above command, the tracking unit will send the following message with its current location to the server:

`$$<L><ID><0x9955><data><checksum>\r\n`

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

4. Set Time Interval for Continuous Tracking

Command code: **0x4102**

Command from server to tracking unit:

`@<L><ID><0x4102><timer interval in hex code (2 bytes)><checksum>\r\n`

Note:

2 bytes of timer interval is in format of 10 seconds and if

=0, it means stop tracking

Max time interval = 65535*10 seconds

Message from tracking unit to server:

`$$<L><ID><0x5100><Flag (1byte)> <timer interval (2 bytes)><checksum>\r\n`

Note:

If Flag

=0, it means setting failed,

=1, it means setting succeeded

When time interval is set successfully, the tracking unit will send the following message with its current location to the server at the interval specified:

\$\$<L><ID><0x9955><data><checksum>\r\n

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

5. Set Authorized Phone Number

Command code: **0x4103**

Command from server to tracking unit:

@<L><ID><0x4103><button number><authorized phone number for receiving SMS><authorized phone number for receiving call><checksum>\r\n

Remarks:

Button number - 1 byte and in hex code. It can support up to 3 buttons. Currently our G100 only have three speed dial buttons (SOS, Call B and Call C). Therefore for these trackers, if button number,
= 0x01, to set authorized phone number for SOS button;
= 0x02, to set authorized phone number for Call B button;
= 0x03, to set authorized phone number for Call C button.

Authorized phone number - 16 bytes in ASCII. If the phone number is less than 16 bytes, the blank byte(s) should read as '0x00'. For example:

If the authorized number is 8613612345678, then it should be written as follows:

0x38 0x36 0x31 0x33 0x36 0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38 0x00 0x00 0x00

If all 16 bytes data are 0x00, it means the authorized number is invalid.

After the above command is received, the tracking unit will send the following message back to the server:

\$\$<L><ID><0x4103><Flag><checksum>\r\n

If Flag (1byte)

=0x00, it means setting failed;

=0x01, it means setting succeeded.

6. Set Speed Limit for Over Speed Alarm

Command code: **0x4105**

Command from server to tracking unit:

@<L><ID><0x4105><speed><checksum>\r\n

Speed - 1 byte and in hex code. When the tracker is over this preset speed limit, an alarm message will be sent to the server.

If speed is

0, 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

After the above command is received, the tracking unit will send the following message back to the server:

\$\$<L><ID><0x4105><Flag><checksum>\r\n

If Flag (1byte)

=0x00, it means setting of speed limit failed;
=0x01, it means setting of speed limit succeeded.

7. Set Movement Alert

Command code: **0x4106**

Command from server to tracking unit:

@<L><ID><0x4106><area (1byte in hex code)><checksum>\r\n

If area:

=0, to cancel movement alert function
=0x01, it is set in a square with current location as center and with side length=2x30m;
=0x02, it is set in a square with current location as center and with side length=2x50m;
=0x03, it is set in a square with current location as center and with side length=2x100m;
=0x04, it is set in a square with current location as center and with side length=2x200m;
=0x05, it is set in a square with current location as center and with side length=2x300m;
=0x06, it is set in a square with current location as center and with side length=2x500m;
=0x07, it is set in a square with current location as center and with side length=2x1000m;
=0x08, it is set in a square with current location as center and with side length=2x2000m.

Message replied from tracking unit to server:

\$\$<L><ID><0x4106><Flag><checksum>\r\n

If Flag

=0, it means setting failed;
=1, it means setting succeeded.

8. Set Geo-fence

Command code: **0x4107, 0x4127**

Command from server to tracking unit:

@<L><ID><0x4107 or 0x4127><coordinates><checksum>\r\n

Remarks:

0x4107 is the command for exiting the geo-fencing area.

0x4127 is the command for entering the geo-fencing area

Coordinates includes:

Bottom left longitude,bottom left latitude,top right longitude, top right latitude

The value of bottom left longitude and latitude should be less than that of top right. And all longitudes and latitudes should be in ASCII format as follows:-

Longitude: DDDFF.FFFF,E/W, 4 places of decimal. '0' is needed to be stuffed if no value available.

Latitude: DDFF.FFFF,N/S, 4 places of decimal. '0' is needed to be stuffed if no value available.

Example:

11404.0000,E,2232.0010,N,11505.1234,E,2333.5678,N

Command '@<L><ID><0x4107 or 0x4127><checksum>\r\n' to cancel geo-fencing.

Following message will be sent from tracking unit to the server to confirm above settings:

\$\$<L><ID><0x4107 or 0x4127><Flag><checksum>\r\n

If Flag

=0, it means setting failed;

=1, it means setting succeeded.

Note: You can only set either movement alert or geo-fencing.

9. Set Extended Functions

Command code: **0x4108**

Command from server to tracking unit:

@<L><ID><0x4108><ABCDEFGG><checksum>\r\n

Remarks for ABCDEFG:

A=0, Turn off the function of sending an SMS location report to the authorized phone number when it makes a call to the tracking unit.

A=1, Turn on the function of sending an SMS location report to the authorized phone number when it makes a call to the tracking unit.

B=0, location data of NMEA 0183 GPRMC will be interpreted into normal text for easy reading.

B=1, location data complies with NMEA 0183 GPRMC protocol.

C=0, Turn off the function to automatically hang up an incoming call after 5 rings.

C=1, Turn on the function to automatically hang up an incoming call after 5 rings.

D=0, Turn off the function of sending a message via SMS/GPRS when the tracking unit is turned on.

D=1, Turn on the function of sending a message via SMS/GPRS when the tracking unit is turned on. SMS will be sent to preset SOS number.

E: reserved. Either 0 or 1 doesn't affect the other settings.

F=0: Turn off the alarm when the tracking unit enters GPS blind area.

F=1: Turn on the alarm when the tracking unit enters GPS blind area. The alarm message is to be sent by SMS/GPRS.

G=0, 3 LED lights work normally.

G=1, 3 LED lights stop flashing automatically after the tracking unit is rebooted and initialized.

Following message will be sent from tracking unit to the server to confirm above settings:

\$\$<L><ID><0x4108><Flag><checksum>\r\n

If Flag

=0x00, it means settings failed;

=0x01, it means settings succeeded.

10. Initialize All Parameters Except for Password, IP/PORT/APN, ID and Time Interval for Continuous Tracking

Command code: **0x4110**

Command from server to tracking unit:

@&&<L><ID><0x4110><checksum>\r\n

Message from tracking unit to server to confirm the settings:

\$\$<L><ID><0x4110><Flag><checksum>\r\n

If Flag

=0x00, it means setting failed;

=0x01, it means setting succeeded.

11. Set Sleep Mode for Power Saving

Command code: **0x4113**

Command from server to tracking unit:

@&&<L><ID><0x4113><Power-saving level (1 byte in hex code)><checksum>\r\n

If Power-saving level

=0, to close power-saving function;

=0x01, sleep level;

=0x02, deep sleep level.

Message from tracking unit to server:

\$\$<L><ID><0x4113><Flag><checksum>\r\n

If Flag

=0x00, it means setting failed;

=0x01, it means setting succeeded.

12. Output Control (safe) - Advised Caution in Using this Function

Command code: **0x4114 or 0x5114**

This command is used for VT300 and VT310 only.

--VT300

Command from server to tracking unit:

@<L><ID><0x4114 or 0x5114><A><checksum>\r\n

A=0, to close output (OUT1);

A=1, to open output (OUT1);

--VT310

Command from server to tracking unit:

@<L><ID><0x4114 or 0x5114><ABCDE><checksum>\r\n

A=0, to close output (OUT1);

A=1, to open output (OUT1);

A=2, to remain previous status.

B=0, to close output (OUT2);

B=1, to open output (OUT2);

B=2, to remain previous status.

C=0, to close output (OUT3);

C=1, to open output (OUT3);

C=2, to remain previous status.

D=0, to close output (OUT4);

D=1, to open output (OUT4);

D=2, to remain previous status.

D=0, to close output (OUT5);

D=1, to open output (OUT5);

D=2, to remain previous status.

Message from tracking unit to server:

\$\$<L><ID><0x4114 or 0x5114><Flag><checksum>\r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

Note: this function is achievable when the speed is below 10km/h (0x4114) or 20km/h (0x5114) and GPS is available.

13. Output Control (general) - Advised Caution in Using this Function

Command code: **0x4115**

This command is used for VT300 and VT310 only.

--VT300

Command from server to tracking unit:

@<L><ID><0x4115><A><checksum>\r\n

A=0, to close output (OUT1);

A=1, to open output (OUT1);

--VT310

Command from server to tracking unit:

@<L><ID><0x4115><ABCDE><checksum>\r\n

A=0, to close output (OUT1);

A=1, to open output (OUT1);

A=2, to remain previous status.

B=0, to close output (OUT2);

B=1, to open output (OUT2);

B=2, to remain previous status.

C=0, to close output (OUT3);

C=1, to open output (OUT3);

C=2, to remain previous status.

D=0, to close output (OUT4);

D=1, to open output (OUT4);

D=2, to remain previous status.

D=0, to close output (OUT5);

D=1, to open output (OUT5);

D=2, to remain previous status.

Message from tracking unit to server:

\$\$<L><ID><0x4115><Flag><checksum>\r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

14. Set GPRS Alert for Buttons or Inputs

Command code: **0x4116**

For GT100 _____

Command from server to tracking unit:

@<L><ID><0x4116><ABC><checksum>\r\n

A=SOS button

B=Call B

C=Call C

When A or B or C

=1, to enable alert when button is pressed

=2, to enable alert when button is released

=3, to enable alert when button is pressed or released

For VT300

Command from server to tracking unit:

@@<L><ID><0x4116><A><checksum>\r\n

As IN1 or VT300 is linked with SOS button, only effective alert (SOS button is pressed) is available. So A=1.

For VT310

Command from server to tracking unit:

@@<L><ID><0x4116><ABCDE><checksum>\r\n

A=IN1

B=IN2

C=IN3

D=IN4

E=IN5

When A or B or C or D or E

=1, to enable alert when input port is close

=2, to enable alert when input port is open

=3, to enable alert when input port is close or open

IN1, IN2, IN3 are detected as low level input. IN4 and IN5 are detected as high level input.

Message from tracking unit to server:

\$\$<L><ID><0x4116><Flag><checksum> \r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

15. Set Power Saving When Tracker is Inactive

Command code: **0x4126**

Command from server to tracking unit:

@@<L><ID><0x4126><data><checksum>\r\n

Data: In ASCII code and in unit of minute

If data = 00, to close this function;

= [01,99], to set this function.

Example:

If data = 10, the tracker will enter power saving mode in ten minutes after it is immobile.

In power saving mode, GPS stops working. GSM enters standby mode and stop sending out message until it is activated or the tracker moves again.

Message from tracking unit to server:

\$\$<L><ID><0x4126><Flag><checksum> \r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

16. Set Telephone Number for Wiretapping (Listen Only)

Command code: **0x4130**

Command from server to tracking unit:

@<L><ID><0x4130><data><checksum>\r\n

Data is the telephone number for wiretapping and it should be numbers or '+' and numbers. Max 16 digits. In ASCII code.

Message from tracking unit to server:

\$\$<L><ID><0x4130><Flag><checksum> \r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

17. Set Interval for Logging

Command code: **0x4131**

Command from server to tracking unit:

@<L><ID><0x4131><data><checksum>\r\n

Data: ASCII code and in unit of second.

If data = 0, to close this function

Data = [1,65535], to set interval for logging.

Message from tracking unit to server:

\$\$<L><ID><0x4131><Flag><checksum> \r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

18. Set Time Zone

Command code: **0x4132**

Command from server to tracking unit:

@<L><ID><0x4132><data><checksum>\r\n

Default time is GMT.

Data is to set time difference in minutes to GMT

Data = [0,65535]/[-1,-65535], In ASCII code and in unit of minute

Message from tracking unit to server:

\$\$<L><ID><0x4132><Flag><checksum> \r\n

If Flag

=0x00, it means operation failed;

=0x01, it means operation succeeded.

19. Read Preset Time Interval of Continuous Tracking

Command code: **0x9002**

Command from server to tracking unit:

@<L><ID><0x9002><checksum>\r\n

Message from tracking unit to server:

\$\$<L><ID><0x9002><Preset time interval (2 bytes in hex code)><checksum> \r\n

Preset time interval: in format of 10 seconds and if

=0, it means stop tracking.

The max timer interval = 65535x10 seconds.

20. Read Authorized Phone Number

Command code: **0x9003**

Command from server to tracking unit:

@<L><ID><0x9003><button number><checksum>\r\n

Button number: 1 byte and in hex code, it should be in the range form 1 to 3. When button name is 0xff, it is to read all authorized numbers.

Message from tracking unit to server:

\$\$<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'.

21. Read Logging Waypoints

Command code: **0x9016**

Command from server to tracking unit:

@<L><ID><0x9016><device random(1B)><PC random(1B)><flag(2B)><checksum>\r\n

Message from tracking unit to server:

\$\$<L><ID><0x9016>< device random(1B)><PC random(1B)><waypoints(4B)><data><checksum>\r\n

Note: Flag = 0x00 0x01 will delete the received waypoint from memory on the condition that sent device

random(1B)=latest received device random(1B)

Waypoints: the number of waypoints that have been saved in memory. High byte prior to low byte.

Data: it is the first record of the logged waypoints. In GPRMC format.

22. Alarm Command

Command: **0x9999**

Command from server to tracking unit:

```
@<L><ID><0x9999><checksum>\r\n
```

Message from tracking unit to server:

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

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

| Alarm code | Definition |
|-------------------|---|
| =0x01 | SOS button is pressed / Input port 1 close |
| =0x02 | 2nd button (Call B) is pressed / Input port 2 close |
| =0x03 | 3rd button (Call C) is pressed / Input port 3 close |
| =0x04 | Input port 4 close |
| =0x05 | Input port 5 close |
| =0x10 | Low battery alarm |
| =0x11 | Over speed alarm |
| =0x12 | movement alarm |
| =0x13 | Geo fence alarm |
| =0x14 | Alarm of tracking unit being turned on |
| =0x15 | Alarm of tracking unit entering blind area |
| =0x16 | Alarm of tracking unit leaving blind area |
| =0x31 | SOS button is released/Input port 1 open |
| =0x32 | Call B button is released/Input port 2 open |
| =0x33 | Call C button is released/Input port 3 open |
| =0x34 | Input port 4 open |
| =0x35 | Input port 5 open |

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

Annex 1: Description of data

Data consists of: *GPRMC | HDOP | Altitude | State | AD1,AD2*

(1) GPRMC includes:

hhmmss.dd,S,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,s.s,h.h,ddmmyy

For example:

134829.486,A,1126.6639,S,11133.3299,W,58.31,309.62,110200

Details:

| Parameter | Description | Example |
|--------------------|---|-----------------------|
| hhmmss.dd | UTC time hh = hours; mm = minutes; ss = seconds; dd = decimal part of seconds | 13:48:29.486 |
| S | 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 | S = South |
| 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 | W=West |
| s.s | Speed, knots. | 58.31 Knots |
| h.h | Heading | 309.62 deg. |
| ddmmyy | Date dd = date; mm = month' yy = year | 11th, Aug. 2000 |

(2) | is list separator in ASCII (=0x7c)

(3) HDOP, in ASCII code, 0.5-99.9. HDOP is blank when the tracking unit has no GPS fix.

(4) Altitude, in algorism.

(5) State: Status of input and output:

--For VT300

Bit 0: Status of Out1.

If Bit0=0: Out1 is closed; Bit0=1:Out1 is open.

Bit8: Status of Input1

If Bit8=0: Input1 is invalid; Bit8=1: Input1 is valid

Bit1~Bit7 and Bit8~Bit15: Reserved and default as '0'

--For VT310

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.

Bit5~7: Reserved.

Bit8: Status of Input1

If Bit8=0: Input1 is invalid; Bit8=1: Input1 is valid

Bit9: Status of Input2

If Bit9=0: Input2 is invalid; Bit9=1: Input2 is valid

Bit10: Status of Input3

If Bit10=0: Input3 is invalid; Bit10=1: Input3 is valid

Bit11: Status of Input4

If Bit11=0: Input4 is invalid; Bit11=1: Input4 is valid

Bit12: Status of Input5

If Bit12=0: Input5 is invalid; Bit12=1: Input5 is valid

Bit13~Bit15: Reserved and default as '0'

(6) AD1, AD2: 10 bit analog input for VT310, 0x0000...0x03ff in HEX, separated by ',' (comma).