



GPS Tracker Communication protocol (GTP)
between GPS Tracker with Server
Version B – 6.0

Tel.: +598 2401 0444 | +1 305 328 3889 | +1 305 424 1494

E_mail: sales@cartrackgps.com

www.cartrackgps.com

Tel.: +598 2401 0444 | +1 305 328 3889 | +1 305 424 1494
E_mail: sales@cartrackgps.com
www.cartrackgps.com



GPS Tracker Communication protocol (GTP)
between GPS Tracker with Server
Version B – 6.0

Version B – 6.0.....	1
1.Introduce.....	4
2.Command Format Specification:.....	5
3.Command list.....	9
4. Detailed CMD (Part1: Link).....	11
CMD5000 (Login after power on).....	11
CMD4000 (confirm login).....	12
CMD 0000 (PING and PING reply).....	13
CMD 1111 (CMD data ERR notice).....	14
5 Detail CMD (Part2: GPS position).....	15
CMD 4101 (Single location request).....	15
CMD 9966 (Single location reply).....	16
CMD 9955 (Continuous location reply).....	19
CMD 9999 (Alarm Alert).....	20
6 Detail CMD (Part3: Configure).....	21
CMD 4102 (Set time interval of continuous tracking mode: MOVE state).....	21
CMD 4112 (Set time interval of continuous tracking mode: PARK state).....	22
CMD 4105 (Set speed limit for over speed alarm).....	23
CMD 4106 (Set GEO-fence).....	24
CMD 4108 (Set extend function).....	25
CMD 4110 (Init all parameter beside password, IP/PORT/APN setting ,ID,timer interval setting of continuous Tracking).....	26
CMD 4113 (Set power saving mode).....	27
CMD 4200 (Delete All GPS data to be stored).....	28
CMD 4307 (Set the init value for Mile Meter Data).....	29
CMD 9002 (Read preset time interval of continuous tracking).....	30
CMD 9005 (Read speed limitation setting of over speed alarm).....	31
CMD 9006 (Read setting of Geo-fence alarm).....	32
CMD 9008 (Read setting of extend function).....	33
CMD 9013 (Read setting of power saving mode).....	34
7 Detail CMD (Part4: IO Control).....	35
CMD 4115 -Drive Output Port to control external device.....	35
8 Detail CMD (Part5: UART(serial port) transmission by transparent mode).....	36
CMD 1001 (Receive or send data by UART Port A).....	36
CMD 1002 (Receive or send data by UART Port B).....	37
9 Detail CMD (Part6: Camera).....	38
CMD 8101 8106 8108 (Get Picture from Camera).....	38

1.Introduce

This GPS Tracker communication protocol (GTP) is defined to make available a datagram mode of packet communication between GPS tracker unit with Server. This protocol provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism. The protocol is transaction oriented, and delivery and duplicate protection are not guaranteed. Applications requiring ordered reliable delivery of streams of data should use the Transmission Control Protocol.

This document describes the functions and definition to be performed by the GTP, the program that implements it, and its interface to programs or users that require its services. This document represents a specification of the behavior required of GPS tracker unit implementation, both in its interactions with other higher level protocols and in its interactions with other Service Center.

2.Command Format Specification:

Depending on the direction of the GPRS package, the communication protocol is defined as following format

The GPRS package from server to track unit:

@@ + <ID (14byte)> + &A + <Command code(4byte)> + <Command data> + ##

The GPRS package from track unit to server:

\$\$ + <ID (14byte)> + &A + <Command (4byte)> + &B + <GPS DATA> + &C + <IO port state(8byte)> + &D + <Distance(8byte)> + ##

sequence: High bit character first and low bit character last

Remark	Explain
@@	2Bytes, means the package header from server to track unit, it is in ASCII code (in hex code ,it is 0x4040)
\$\$	2Bytes, means the package header from tracker unit to server ,it is in ASCII Code (in hex code ,it is 0x2424)
<ID>	14ytes,(Ascii, 0x30-0x39) the unused byte will be stuffed by 0x3f. For example when ID is 123456 ,then it will be show 123456?????????
&A	Command header
<Command Code>	4Bytes, As command list and command explain below.
<Command Data>	The parameter for command code. All data is in ASCII code(0x30~0x3f).
&B	GPS data header
<GPS data>	Like NMEA0183 format (All data is in ASCII code) For detail, pls read <Appendix 1>
&C	IO port state data header
<IO port state data>	10 byte (depend on different tracker) In general, byte5~byte9 is output state. (port1~port5) byte0~byte4 is input state. (port1~port5) Data sequence : low byte first and high byte last (&C+ byte0+byte1+byte2+byte3.....+byte9)
&D	Mile meter data header
<Mile meter data>	8 byte (0x30~0x3f) unit: meter Data sequence : high byte first and low byte last
&E	Alarm data header
<Alarm data>	Byte0: The alarm of external Power Supply be cut off Byte7: ='0' MOVE state ; ='1' PARK state Data sequence : low byte first and high byte last (&E+ byte0+byte1+byte2+.....byte7)
&Y	AD input data header
<AD input Data>	Byte4~byte7 is AD1 value Byte0~byte3 is AD2 value Data sequence : high byte first and low byte last
## or ####	End char ## : indicate the data is REAL-TIME gps data; ####: indicate the data is RECORD gps data be stored in Device Memory When GPRS is not available.

ASCII Format:

When we say "ASCII format" here, we means as following:

HEX -> ASCII

0x1 -> '1'
0x2 -> '2'
0x3 -> '3'
0x4 -> '4'
0x5 -> '5'
0x6 -> '6'
0x7 -> '7'
0x8 -> '8'
0x9 -> '9'
0xA -> ':'
0xB -> ','
0xC -> '<'
0xD -> '='
0xE -> '>'
0xF -> '?'

Example: 34?>:4 is 0x34FEA4 =3473060

Example:
[TCP data]

\$\$222234??????&A9955&B102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17|6.
3|&C0100000100&D000024?>&E10000000&Y00100020####

INFORMATION	QTY BYTES	EXAMPLE	MEANING
Header	2	\$\$	from tracker to server
Device ID	14	2222234????????	ID is 2222234
Command Header	2	&A	
Command Code	4	9955	Indicate the packet is gps position report
GPS Data Header	2	&B	
GPS Data	Chang	102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17	Like NMEA0183 format (All data is in ASCII code) For detail, pls read <Appendix 1>
IO port state data header	2	&C	
IO port state data	10	0100000100	Byte1= =1 Indicate input port 2 be Activated Byte7= =1 Indicate device output 2 be enable
Mile meter data header	2	&D	
Mile meter data	8	000024?>	000024?> : is stand for 0x000024fe ==9470 meters
Alarm data header	2	&E	
Alarm data	8	10000000	Byte0 = =1 The alarm of external Power Supply be cut off Byte7= =0: Device is in MOVE state
AD input data header	2	&Y	
AD input data	8	00100020	0010: value of AD1 input is 0x0010 0020: value of AD2 input is 0x0020
End char	2 or 4 bytes	####	Indicate the data is RECORD gps data be stored in Device Memory when GPRS is not Available.

3.Command List:

Command	Explain	Direction
Part 1: Link		
4000	Confirm logon success	Down
5000	Tracker logon	Up
0000	PING message	Up/Down
1111	CMD data ERR notice	Up
Part 2: GPS Position		
4101	Single location request	Down
4102	Set time interval of continuous tracking mode in GPRS working mode	Down
9955	Continue GPS data reply	Up
9966	Signal GPS data reply	Up
9999	Alarm alert + GPS data reply	Up
Part 3: Configure		
4102	Set time interval of continues tracking mode (MOVE state)	Down
4112	Set time interval of continues tracking mode (PARK state)	Down
4103	Set authorized telephone number and SMS number	Down
4105	Set speed limitation of over speed alarm	Down
4106	Set Geo-fence	Down
4108	Set extend function	Down
4110	Init all parameter beside password , IP/PORT/APN,ID, time interval when continuous tracking,	Down
4113	Set power saving mode	Down

3.Command List:

Command	Explain	Direction
4200	Delete All GPS data to be stored	Down
4307	Set the init value for Mile Meter Data	Down
9002	Read preset time interval of continuous tracking mode	Up
9005	Read speed limitation setting of over speed alarm	Up
9006	Read setting of geo-fence alarm	Up
9008	Read setting of extend function	Up
9013	Read setting of power saving mode	Up
9014	Read the status of output port	Up
Part 4: IO control		
4115	Drive output port to control external device	Down
Part5: UART(serial port) transmission by transparent mode UART For the specific UART external device of Customer		
1001	Receive or send data by UART A (Only effective for iTrackPro AVL)	User can control their own UART device by the CMD
1002	Receive or send data by UART B (Only effective for iTrackPro AVL)	User can control their own UART device by the CMD
Part6: Camera		
8101		
8106		
8108		

4. Detailed CMD (Part1: Link)

CMD5000 (Login after power on)

Command code: 5000

After the IP/PORT/APN is set ,the tracker unit apply for GPRS service by sending command every about 30 second until it login the server.

Login command from tracker unit to server to apply GPRS service:

(Tracker - Server)

\$\$ + ID + &A+5000 + ##

Following message will be send back from server when the server receive the login command

(Tracker - Server)

@@ + ID + &A+4000 + 2B Flag + ##

2B Flag:

= 0x0030 means login failed

= 0x0031 means login success

For example:

\$\$1234567890????&A5000##

CMD4000 (Login after power on)

Command code: 4000

Login command from tracker unit to server to apply GPRS service:

(Tracker - Server)

\$\$ + ID + &A+5000 + ##

Following message will be send back from server when the server receive the login command:

(Tracker - Server)

@@ + ID + &A + 4000 + 2B Flag + ##

2B Flag:

= 0x0030 means login failed

= 0x0031 means login success

For example: **@@1234567890????&A400001##**

CMD0000 (PING and PING reply)

Command code: 0000

Function:

Maintain TCP connection, and avoid TCP connection be disconnected for timeout.

(When there no data communication between server and tracker for some minutes, TCP connection often be disconnected by network provider.)

PING from server to tracker unit

(Server - Tracker)

@@ + ID + &A + 0000 + ##

PING Reply from tracker unit to server

(Tracker - Server)

\$\$ + ID + &A + 0000 + ##

CMD 1111 (CMD data ERR notice)

If Server send tracker one TCP CMD with WRONG data, the tracker reply CMD 1111 to notice the server that this TCP CMD is wrong or not useful.

Command code: 1111

Notice from server to tracker unit :
(Tracker - Server)

\$\$ + ID + &A + 1111 + ##

5 Detail CMD (Part2: GPS position)

CMD 4101 (Single location request)

Command code: 4101

Command from server to tracker units:
(Server - Tracker)

@@+ ID + &A+4101+ ##

Message from tracker units to server:
(Tracker - Server)

**\$\$ + ID + &A+9966 +&B+ <GPS Data> +&C+ <IO data>+&D+ <Distance data>
+&E + <Alarm data> + &Y + <AD data> + ##**

For example: **\$\$1234567890????&A4101##**

5 Detail CMD (Part2: GPS)

Command code: 9966

Message from tracker units to server:
(Tracker - Server)

**\$\$ + ID + &A+9966 +&B+ <GPS Data> +&C+ <IO data>+&D+ <Distance data>
+&E + <Alarm data> + &Y + <AD data> + ##**

<GPS Data>

GPRMC info + |+ HDOP +

|

[GPRMC info]

Include: (**Time, date, position, course and speed data**)

[GPRMC info Format]: (like NMEA)

**hhmmss.dd,S,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,s.s,h.h,ddm
myy,d.d,<E|W>,M*hh**

[GPRMC info Example]:

134829.486,A,1126.6639,S,11133.3299,W,58.31,309.62,110200,,,A*14

Parameter	Description	Example
Hhmmss.dd	UTC time, h = hours, mm = minutes, ss = seconds, dd = decimal part of seconds	13:48:29.486
S	Status indicator, A = valid, V = invalid	Valid
Xxmm.dddd	Latitude, xx = degrees, mm = minutes, dddd = decimal part of minutes	11 deg.
<N S>	Either character N or character S, N = North, S = South	26.6639 min.
Yyymm.dddd	Longitude, yy = 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	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
D.d	Magnetic variation	
<E W>	Declination. Either character E or character W, E = East, W = West	
M	Mode indicator, A = autonomous, N = data not valid	Autonomous
Hh	Check sum	14

Notes:

1. List separator | in ASC II is 0x7c
2. HDOP is in ASCII code, HDOP is empty when no GPS signal

Example:

[TCP data]

\$\$222234??????&A9966&B102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17|6.3|&C010000000&D000024?>&E10000000&Y00100020##

[Explain]

\$\$ ---- header of one package (from tracker to server)

222234?????? ---- ID is 222234

&A9966 --- CMD 9966

<Blue string> ---GPS data

(ASCII: 102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17|6.3|)

&C01000000 --- IO port state

&D000024?> --- Mile meter data

(000024?> : is stand for 0x000024fe ==9470 meters)

&E1000000 --- Alarm data

(byte7 is 1: alarm of external power supply be cut off)

&Y00100020 --- AD input data

(0010: value of AD1 input is 0x0010;
0020: value of AD2 input is 0x0020)

---- end char

CMD 9955 (Continuous Location Reply)

Command code: 9955

Message from tracker units to server:
(Tracker - Server)

**\$\$ + ID + &A+9966 +&B+ <GPS Data> +&C+ <IO data>+&D+ <Distance data>
+&E + <Alarm data> + &Y + <AD data> + ##**

<GPS Data> : Please refer to Command 9966.

CMD 9999 (Alarm Alert)

Command code: 9999

Message from tracker units to server:
(Tracker - Server)

**\$\$ + ID +&A+ 9999 + <2B alarm types> +&B+ <GPS info>
+ &C+ <IO data>+&D+ <Distance data> +&E + <Alarm data> + &Y + <AD data> + ##**

2B Alarm types:

- =01 SOS button is pressed
- =02 2nd button(call B)is pressed
- =03 3rd button(call C) is pressed
- =04 Reserved
- =10 Low battery alarm
- =11 over speed alarm
- =12 Geo fence alarm
- =13 Reserved
- =14 Tracker units reset or power up alarm
- =15 No GPS signal alarm
- =16 Out of no GPS signal area alarm
- =21 Input port low to high edge alarm, voltage of input port rise from 0V to 10V-----
- =31 Input port high to low edge alarm, voltage of input port rise from 10V to 0V-----

- = 51 Cut off External Power Supply
- =52 Direction Chang Alarm

6 Detail CMD (Part3: Configure)

CMD 4102 (Set time interval of continuous tracking mode: MOVE state)

Command code:4102

Command from server to tracker units:
(Server - Tracker)

@@+&A+4102+<4Bytes (timer interval ,in hex code 0x30~0x3f)>+##

For example: **\$\$1234567890????&A41020003##**
(note: time interval == 30s)

4 Byte timer interval: unit: 10 seconds
=0000 mean STOP tracking and STOP GPRS.
The max timer interval =65535*10 seconds

Message from tracker units to server: -----

\$\$+ID+&A+5100+Flag(2Byte)+data: 4Bytes (timer interval)+##
Flag: =0x0030 Means setting failed
=0x0031 Means setting success

Data: return time interval to be set.(in hex code 0x30~0x3f)
(note: if cmd data is 0000, the GPRS function is STOPPED.)

CMD 4112 (Set time interval of continuous tracking mode: PARK state)

Command code:4112

Command from server to tracker units:
(Server - Tracker)

@@+&A+4112+<4Bytes (timer interval ,in hex code 0x30~0x3f)>+##

For example: **\$\$1234567890????&A4112000<##**
(note: 000< is stand for 0x000C; It say: time interval == 120s)

4 Byte timer interval: unit: 10 seconds

=0000 mean STOP reply in PARK state

The max timer interval =65535* 10 seconds

Message from tracker units to server: -----

\$\$+ID+&A+4112+Flag(2Byte)+data: 4Bytes (timer interval)+##

Flag: =0x0030 Means setting failed

=0x0031 Means setting success

Data: return time interval to be set.(in hex code 0x30~0x3f)

CMD 4105 (Set speed limit for over speed alarm)

Command code: 4105

Command from server to tracker units:
(Server - Tracker)

@@ + ID + &A + 4105 + <2Byte speed limitation (in hex code: 0x30~0x3f)> + ##

2Byte speed limitation: when speed over speed limitation, a alarm message will send to server

Speed limitation= 00: cancel over speed alarm function (unit: 10km/hour)

01: speed limitation set to 10km/hour

02: speed limitation set to 20km/hour

03: speed limitation set to 30km/hour

04: speed limitation set to 40km/hour

05: speed limitation set to 50km/hour

0?: speed limitation set to 150km/hour

.....

Max speed limitation is 14, which means speed limitation to 200km/hour

Message from tracker unit to server:

\$\$ + ID + &A + 4105 + <Flag(2B)> + <Data(2B)> + ##

Flag: =00 means setting speed limitation failed

=01 means setting speed limitation success

Data: return the speed limitation to be set.

CMD4106 (Set GEO-fence)

Command code:4106

Command from call center to tracker units
(Server - Tracker)

@@ + ID + &A+ 4106 + <2Byte Geo-fence area (in hex code:0x30~0x3f)> + ##

2 Byte Geo-fence area:

=00: cancel geo-fence function

=01: geo-fence area is set in a square with Side length=2*30m,set current point as center

=02: geo-fence area is set in a square with side length=2*50m,set current point as center

=03: geo-fence area is set in a square with side length=2*100m,set current point as center

=04: geo-fence area is set in a square with side length=2*200m,set current point as center

=05: geo-fence area is set in a square with side length=2*300m ,set current point as center

=06: geo-fence area is set in a square with side length=2*500m,set current point as center

=07: geo-fence area is set in a square with side length=2*1000m,set current point as center

=08: geo-fence area is set in a square with side length=2*2000m,set current point as center

Message from track unit to server -----

\$\$ + ID + &A+ 4106 + <Flag(2B)> + <Data(2B)> + ##

Flag: =0x0030 means setting geo-fence failed

=0x0031 means setting geo-fence success

Data: return geo-fence area to be set.

CMD 4108 (Set extend function)

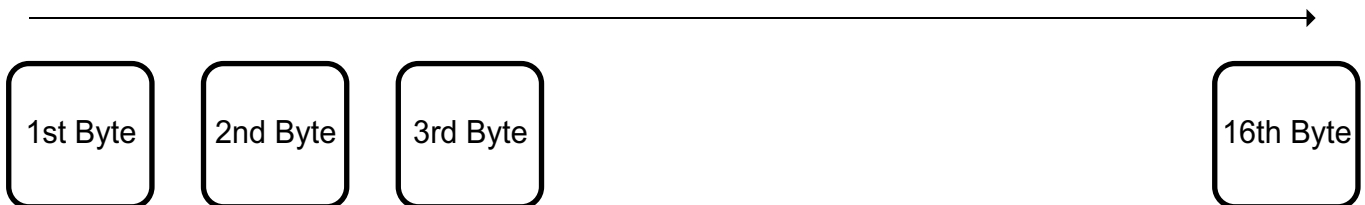
Command code: 4108

Command from server to tracker units-----

(Server - Tracker)

@@ + ID + &A+ 4108 + <extend function flag(16B)> + ##

Extend function flag:



1st byte=0: close the function of send back coordinate info to authorized cell phone by SMS when receive call

1st byte=1: open the function of send back coordinate info to authorized cell phone by SMS when receive call

2nd byte=0: coordinate info is in ASCII code ,the former NMEA code will be translated by tracker units to ASCII code and easier to read

2nd byte=1: coordinate info is obey NMEA0183 GPRMC protocol

3rd byte=0: close the function of automatic close the call

3rd byte=1: open the function of automatic close the call

4th byte=0: close the function of SMS/GPRS message when the tracker unit is turn on

4th byte=1: open the function of SMS/GPRS message when the tracker unit is turn on

5th byte: reserved

6th byte=0: close the alarm function when entering GPS signal blind area, the alarm message send by SMS/GPRS

6th byte=1: open the alarm function when exit GPS signal blind area, the alarm message send by SMS/GPRS

7th byte=0: 3 LED works normally

7th byte=1: 3 LED closed automatically when the track unit is working

Other byte: reserved

Following message will send from tracker units to server to confirm setting:

\$\$ ++ ID + &A+4108 + <Flag(2B)> + ##

Flag: =0x0030 means setting failed

=0x0031 means setting success

CMD 4110 (Init all parameter beside password, IP/PORT/APN setting ,ID,timer interval setting of continuous tracking)

Command code:4110

Command from server to tracker units:
(Server - Tracker)

@@ + ID +&A+ 4110 + ##

Message from tracker units to server to confirm setting:

\$\$ + ID +&A+ 4110 + <Flag(2B)> + ##

Flag: =0x0030 means setting fails,
=0x0031 means setting success.

CMD 4113 (Set power saving mode)

Command line:4113

Command from server to tracker units:
(Server - Tracker)

@@ + ID +&A+ 4113 + <power saving level (2B)(in hex code:0x30~0x3f)> + ##

2Byte power saving level

=00 close power saving function
=01 low power saving level
=02 middle power saving level
=03 high power saving level

Message from tracker units to server:

\$\$ + ID +&A+ 4113 + <Flag(2B)> + ##

Flag: =0x0030 means setting failed
=0x0031 means setting success

CMD 4200 (Delete All GPS data to be stored)

Command line:4200

Function: Delete All info in storage
Command from server to tracker units:
(Server - Tracker)

@@ + ID +&A+ 4200 + ##

Message from tracker units to server:
\$\$ + ID +&A+ 4200 + <Flag(2B)> + ##

Flag: =0x0030 means delete fail
=0x0031 means delete success

CMD 4307 (Set the init value for Mile Meter Data)

Command line:4307

Command from server to tracker units:
(Server - Tracker)

@@ + ID +&A+ 4307 + <Mile Meter Data (in hex code:0x30~0x3f)> + ##

Mile Meter Data:
(unit: Km)

note: data max < 40000 km (or 0x9c40 km)

Message from tracker units to server:

\$\$ + ID +&A+ 4307 + <Flag(2B)>+ <Data be set> + ##

Flag: =0x0030 means setting failed
=0x0031 means setting success

Data: the data to be set successfully
example :

@@20000????????&A43071<:?##

Set init mile meter data is 0x7ab1 km (or 31409km)

CMD 9002 (Read preset time interval of continuous tracking)

Command code:9002

Message from server to tracker units
(Server - Tracker)

@@ + ID + &A+9002 + ##

Message from tracker units to server

\$\$ + ID + &A+9002 + <(4Bytes) preset time interval of continuous tracking(in hex code: 0x30~0x3f)> + ##

4Bytes time interval: (unit: 10 seconds)
=0 mean stop tracking

The max timer interval =65535*10 seconds

CMD 9005 (Read speed limitation setting of over speed alarm)

Command code:9005

Message from server to tracker units
(Server - Tracker)

@@ + ID + &A+ 9005 + ##

Message from track unit to server

\$\$ + ID + &A+9005 + <(2B) speed limitation of over speed alarm> + ##

Speed limitation =0: cancel over speed alarm function

01:speed limitation set to 10km/hour

02:speed limitation set to 20km/hour

03:speed limitation set to 30km/hour

04:speed limitation set to 40km/hour

05:speed limitation set to 50km/hour

.....

Max speed limitation is 14, which means speed limitation to 200km/hour

CMD 9006 (Read setting of Geo-fence alarm)

Command code:0x9006

Message from server to tracker units
(Server - Tracker)

@@ + ID + &A+9006 + ##

Message from track units to server

\$\$ + ID + &A+ 9006 + <(2B) Geo-fence setting> + ##

1 Byte Geo-fence area:

=00: cancel geo-fence function

=01: geo-fence area is set in a square with Side length=2*30m, set current point as center

=02: geo-fence area is set in a square with side length=2*50m, set current point as center

=03: geo-fence area is set in a square with side length=2*100m, set current point as center

=04: geo-fence area is set in a square with side length=2*200m, set current point as center

=05: geo-fence area is set in a square with side length=2*300m, set current point as center

=06: geo-fence area is set in a square with side length=2*500m, set current point as center

=07: geo-fence area is set in a square with side length=2*1000m, set current point as center

=08: geo-fence area is set in a square with side length=2*2000m, set current point as center

CMD 9008 (Read setting of extend function)

Command code:9008

Message from server to tracker units
(Server - Tracker)

@@ + ID + &A+ 9008 + ##

Message from track units to server:

\$\$ + ID + &A+9008 + <extend function flags> + ##

Extend function flag:



1st byte=0: close the function of send back coordinate info to authorized cell phone by SMS when receive call

1st byte=1: open the function of send back coordinate info to authorized cell phone by SMS when receive call

2nd byte =0 :coordinate info is in ASCII code ,the former NMEA code will be translated by tracker units to ASCII code and easier to read

2nd byte=1:coordinate info is obey NMEA0183 GPRMC protocol

3rd byte=0: close the function of automatic close the call

3rd byte=1: open the function of automatic close the call

4th byte=0: close the function of SMS/GPRS message when the tracker unit is turn on

4th byte=1: Open the function of SMS/GPRS message when the tracker unit is turn on

5th byte: reserved

6th byte=0:close the alarm function when entering GPS signal blind area, the alarm message send by SMS/GPRS

6th byte=1:open the alarm function when exit GPS signal blind area, the alarm message send by SMS/GPRS

7th byte=0: 3 LED works normally

7th byte=1: 3 LED closed automatically after reset

CMD 9013 (Read setting of power saving mode)

Command code:9013

Message from server to tracker units:
(Server - Tracker)

@@ + ID + &A + 9013 + ##

Message from tracker units to server

\$\$ + ID + &A + 9013 + <(2B) power saving mode> + ##

2B power saving level

- =00 close power saving function
- =01 low power saving mode
- =02 middle power saving mode
- =03 high power saving mode

7.Detail CMD (Part4: IO Control)

CMD 4115 -Drive Output Port to control external device

Command code:4115

Message from server to tracker unit:
(Server - Tracker)

@@+ ID + &A + 4115 + <AABBCCDDEE (10bytes) > + ##

Data: The Output Port give the current to drive extern device (such as relay)
(note: Inside Tracker, the port will be connect to GND for giving the current)

AA==00 Close port 1; (disconnect to GND) (Port in High impedance state)
==01 Drive port 1; (connect to GND)
==02 or other No function

+

BB==00 Close port 2; (disconnect to GND) (Port in High impedance state)
==01 Drive port 2; (connect to GND)
==02 or other No function

CC==00 Close port 3; (disconnect to GND) (Port in High impedance state)
==01 Drive port 3; (connect to GND)
==02 or other No function

DD==00 Close port 4; (disconnect to GND) (Port in High impedance state)
==01 Drive port 4; (connect to GND)
==02 or other No function

EE ==00 Close port 5; (disconnect to GND) (Port in High impedance state)
==01 Drive port 5; (connect to GND)
==02 or other No function

For example: **\$\$1234567890????&A41150200010202+##**
(it is mean: close port 2; drive port3; other port be not changed)

Message from tracker units to server:
\$\$ + ID + &A+4115 + <Flag(2B)> +##

Flag: =00 means operation failed
=01 means operation success

Data: return the data to be set.

8.Detail CMD (Part5: UART(serial port) transmission by transparent mode)

CMD 1001 (Receive or send data by UART Port A)

(Only effective for iTrackPro AVL)

UART(serial port) transmission by transparent mode

User can operate/use their own UART(serial port) device by the CMD, such as card reader or some sensor.

UART Port A receive data from external device such as sensor or card reader. -----

(Data direction: External device - Tracker - Server)

\$\$+ID+&A1001+ <2B Data length> +&R+

<Data (which be received by UART Port A from extern device)> + ##

Example:

\$\$123????????????&A100114&RXXXXXXXXXXXXXXXXXXXXXX##

Data length: Data length which be received

XXXXXXXXXXXXXX: data which be received by UART Port A from external device.

UART Port A send data to external device such as sensor or card reader.

(Data direction: Server - Tracker - External Device)

@@+ID+&A1001+ <2B Data length> +&S+

<Data (which be sent by UART Port A to external device)> + ##

Example:

@@123????????????&A100114&SYYYYYYYYYYYYYYYYYYYY##

Data length: Data length which be sent

YYYYYYYYYYYYYYYYYYYY: data which be sent by USRT Port A to extern device.

CMD 1002 (Receive or send data by UART Port B)

(Only effective for iTrackPro AVL Advance)

UART(serial port) transmission by transparent mode

User can operate/use their own UART(serial port) device by the CMD, such as card reader or some sensor.

UART Port B receive data from external device such as sensor or card reader.

(Data direction: External device - Tracker - Server)

\$\$+ID+&A1002+ <2B Data length> +&R+

<Data (which be received by UART Port B from extern device)> + ##

Example:

\$\$123????????????&A100214&RXXXXXXXXXXXXXXXXXXXXXX##

Data length: the length of data which be received

XXXXXXXXXXXXXX: data which be received by USRT Port B from external device.

UART Port B send data to external device such as sensor or card reader.

(Data direction: Server - Tracker - External Device)

@@+ID+&A1002+ <2B Data length> +&S+

<Data (which be sent by UART Port B to external device)> + ##

Example:

@@123????????????&A100214&SYYYYYYYYYYYYYYYYYYYY##

Data length: Data length which be sent

YYYYYYYYYYYYYYYYYYYY: data which be sent by UART Port B to external device

9.Detail CMD (Part6: Camera)

CMD 8101 8106 8108 (Get Picture from Camera)

Firstly, Please check Hardware's wiring

```
// Camera GPS Tracker iTrackPro ADVANCE
// Red Line <-----> PIN1(5V Power, in red too)
// Black Line <-----> PIN4(Ground, also in black)
// Yellow Line <-----> PIN2(TXD2)
// Green Line <-----> PIN5(RXD2)
//
// or
// Red Line <-----> PIN1(5V Power, in red too)
// Black Line <-----> PIN4(Ground, also in black)
// Yellow Line <-----> PIN3(TXD3)
// Green Line <-----> PIN6(RXD3)
```

Picture Capturing

In order to get the picture, you need to do two things as following:

- A) Start the camera to capture one picture;
- B) Get the data content of the captured picture from the camera.

Step A: Capture a picture

Capture a picture with a specified size

Command code: "8101" (Server -> Tracker)
Command data: "03" + "43" + <PS> + <TS> (ASCII format)
or "04" + "43" + <PS">+ <TS> (ASCII format) + <DD>
"03" or "04" means the length of parameters is 3 or 4,
"43" is the ASCII code(in HEX) of character 'C', is the CMD char of capture
"PS" means picture size, for now, we have three options:

PS = "00", capture a 160 * 128 size picture
PS = "01", capture a 320 * 240 size picture
PS = "02", capture a 640 * 480 size picture

"TS" means transfer size of one packet , which is between 64 and 227

"DD" means transfer delay, unit is 20ms, default value is 5, i.e. $5 * 20 = 100\text{ms}$

For example: "03430040" means capture a $160 * 128$ size picture, transfer size is 64

For example: "0343015:" means capture a $320 * 240$ size picture, transfer size is 90(0x5A)

For example: "0343026?" means capture a $640 * 480$ size picture, transfer size is 111(0x6F)

One suggestion example : **@@123456????????&A81010443004020##**

If success, a picture info packet(command code is "8106") will be return to the server.
the content of the picture info packet(6 bytes in hex) is as following:

0xLL 0xLL 0xLL 0xLL 0xPP 0xPP(all in hex format, NOT ASCII)
0xLL 0xLL 0xLL 0xLL means the total size of the captured picture (network endian)
0xPP 0xPP (network endian) means how many data packets the picture contain,

Step B: Transfer the picture

Device send picture data packet to Server.
the content of the picture(jpg format) will be transferred in small packets per request

Command code: "8101" (Server -> Tracker)
Command data: "03" + "47" + "PP" + "PP"(ASCII format)

"03" means the length of parameters is 3,
"47" is the ASCII code(in HEX) of character 'G', CMD char of data request.
"PP" "PP" means the packet number we're going to fetch, which is start from 1 in network endian

For example: @@123456????????&A810103470001##, get the first data packet
For example: @@123456????????&A81010347001;##, get the 27th(0x1B) data packet

If success, a picture data packet(command code is "8108") will be send to the server
Command code: "8108" (Tracker->Server)
Command data: 0xPP 0xPP 0xLL 0xLL 0xDD...0xDD (HEX code)

0xPP 0xPP is the packet number you request, in network endian
0xLL 0xLL is the length of the current picture data packet(not including "0xPP 0xPP 0xLL 0xLL"), also in network endian

0xDD...0xDD is the data content of the picture packet (HEX code)

Appendix 1 (GPS Data Info)

<GPS Data>

GPRMC info + | + HDOP + |

[GPRMC info]

Include: (Time, date, position, course and speed data)

[GPRMC info Format]: (like NMEA0183)

hhmmss.dd,S,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,s.s,h.h,ddm
 myy,d.d,<E|W>,M*hh

[GPRMC info Example]:

134829.486,A,1126.6639,S,11133.3299,W,58.31,309.62,110200,,,A*14

Parameter	Description	Example
Hhmmss.dd	UTC time, h = hours, mm = minutes, ss = seconds, dd = decimal part of seconds	13:48:29.486
S	Status indicator, A= valid, V = invalid	Valid
Xxmm.dddd	Latitude, xx = degrees, mm = minutes, dddd = decimal part of minutes	11 deg.
<N S>	Either character N or character S, N = North, S = South	26.6639 min.
Yyymm.dddd	Longitude, yy = 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	West
S.s	Speed, knots.	58.31 Knots
H.h	Heading	309.62 deg.
Ddmyy	Date, dd = date, mm = month, yy = year	11th, Aug. 2000
D.d	Magnetic variation	
<E W>	Declination. Either character E or character W, E = East, W = West	
M	Mode indicator, A = autonomous, N = data not valid	Autonomous
Hh	Check sum	14

Notes:

1. List separator | in ASC II is 0x7c
2. HDOP is in ASCII code, HDOP is empty when no GPS signal

Example:

[TCP data]

\$\$222234??????&A9966&B102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17|6.3|&C010000000&D000024?>&E10000000&Y00100020##

[Explain]

\$\$ ---- header of one package (from tracker to server)

222234??????? ---- ID is **222234**

&A9966 --- CMD 9966

<Blue string> ---GPS data

(ASCII: 102904.000,A,2233.0655,N,11404.9440,E,0.00,,030109,,*17|6.3|)

&C01000000 --- IO port state

&D000024?> --- Mile meter data

(000024?> : is stand for 0x000024fe ==9470 meters)

&E10000000 --- Alarm data

(byte7 is 1: alarm of external power supply be cut off)

&Y00100020 --- AD input data

(0010: value of AD1 input is 0x0010; 0020: value of AD2 input is 0x0020)

---- end char

Tel.: +598 2401 0444 | +1 305 328 3889 | +1 305 424 1494

E_mail: sales@cartrackgps.com

www.cartrackgps.com