RedView GPS

The Communication protocol (GTP) between GT30 Personal Tracking Device With Call Center

Version 3.0

Revision History

Version	Date	Author	Remark
V1.00	2006-6-25	Hai Bing Deng	Init draft
V2.00	2007-5-06	Hai Bing Deng	Add alarm function
V3.00	2007-09-17	Hai Bing Deng	Add extend function

1 Introduce

This GT30 GPS Tracker communication protocol (GTP) is defined to make available a datagram mode of packet communication between GT30 with Call Center. 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 GT30 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

A .The GPRS package from call center to track unit:

@@ + L +ID (7byte) + command (2byte) + DATA + checksum(2byte) +\r\n

B .The GPRS package from track unit to call center:

$$ + L + ID (7byte) + command (2byte) + DATA + checksum(2byte) + \r\n

All multi byte data is according to the sequence: High bit character first and low bit character last

Remark	Explain
@@	2Bytes, means the package header from call center to track unit, it is in ASCII code (in hex code ,it is 0x40)
$$	2Bytes, means the package header from tracker unit to call center ,it is in ASCII code (in hex code ,it is 0x24)
L	2Bytes, meaning the length of the whole package ,including the header and end character of the package, it is in hex code
ID	7Bytes, the unused byte will be stuffed by 0xff,it is in format of hex code. For example when ID is 13612345678 ,then it will be show as following: 0x13,0x61,0x023,0x45,0x67,0x8f,0xff.When all 7 bytes is 0xff,it is a broadcast command. ID is in hex code
Command	2Bytes, as command list and command explain shows. The command code is in hex code
Data	Support min 0 Bytes, max 100 bytes. All data is in ASCII code

Checksum	2Bytes,indicate CRC-CCITT(default is 0xffff) checksum of all data(not including CRC itself and the end 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	2Bytes,end character ,it is in hex code(0x0d,0x0a in hex code)

3.Command list

Command	Explain	REmark
0x4000	Message send from server to tracker unit to confirm logon success	
*********		
0x5000	Message from tracker unit to server to apply logon	
0x4101	Single location request	
0x4102	Set time interval of continuous tracking mode in GPRS working mode	
0x4103	Set authorized telephone number and SMS number	
0x4104	Reserved	
0x4105	Set speed limitation of over speed alarm	
0x4106	Set Geo-fence	
0x4107	Reserved	
0x4108	Set extend function	
0x4109	Reserved	
0x4110	Init all parameter beside password ,IP/PORT/APN,ID, time interval when continuous tracking,	
0x4111	Reserved	
0x4112	Reserved	
0x4113	Set power saving mode	
*********		
0x9000	Reserved	
0x9001	Reserved	
0x9002	Read time interval of continuous tracking mode in GPRS working mode	
0x9003	Read the authorized telephone number	
0x9004	Reserved	
0x9005	Read setting of over speed alarm	
0x9006	Read setting of geo-fence alarm	
0x9007	Reserved	
0x9008	Read setting of extend function	
0x9011	Reserved	
0x9012	Reserved	
0x9013	Read setting of power saving mode	
********		
0x9999	Alarm	

4. Detailed command

(1). Login after power on

Command code: 0x5000

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

For example:

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

$$ + L + ID + 0x5000 + Checksum(2byte) + \r\n

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

@@ + L + ID + 0x4000 + 1B Flag + Checksum(2byte) +\r\n

1B Flag:

= 0x00 means login failed

= 0x01 means login success

(2). Message from server to tracker unit to confirm login

Command code:0x4000

For example:

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

$$ + L + ID + 0x5000 + Checksum(2byte) + \r\n

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

@@ + L + ID + 0x4000 + 1B Flag + checksum(2byte) +\r\n

1B Flag：

= 0x00 means login failed

= 0x01 means login success

(3).Single location request

Command code:0x4101

For example:

Command from server to tracker units:

@@ + L + ID + 0x4101+ checksum(2byte) + \r\n

Message from tracker units to server:

$$ + L + ID + 0x9955 + GPRMC info + | + HDOP +

checksum(2byte) + \r\n

GPRMC info:

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

GPRMC info DATA Format is:

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

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, 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	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

Notes:

1. if xxmm.dddd or yyymm.dddd is " 0000.000",it means track unit cannot receive GPS info.
2. List separator | in ASC II is 0x7c
3. HDOP is in ASCII code, HDOP is empty when no GPS signal

(4).Set time interval of continuous tracking mode

Command code:0x4102

For example:

Command from server to tracker units:

@@ + L + ID + 0x4102 + 2Bytes (timer interval ,in hex code) + checksum(2byte) + \r\n

2 Byte timer interval: in format of 10 seconds
=0 mean stop tracking
The max timer interval =65535*10 seconds

Message from tracker units to server:

$$ + L + ID + 0x5100 + Flag(1Byte) + 2Bytes (timer interval) +

checksum(2byte) + \r\n

Flag： =0 Means setting failed
=1 Means setting success
Time interval is in Hex code

(5).Set authorized telephone number

Command code:0x4103

For example:

Command from server to tracker units:

@@ + L + ID + 0x4103

+ 1B name of key(hex code) ,which set to link the following SMS and telephone number

+ 16Byte cell phone number to receive SMS message when the key was pressed (in ASCII code)

+ 16Byte telephone number to receive the call when the key was pressed (in ASCII code)

+ checksum(2byte) + \r\n

Name of key: it can support up to 9 key. At present in GT30,there are 3 keys so the min name is 0x01 and max name is 0x03

16 byte ASCII authorized number, for number less than 16 bytes ,the empty bytes should be stuffed by 0x00
For example 8613612345678 should be shown as following:

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

If all the 16 bytes data is 0x00,this means the authorized code is invalid

After the tracker units get the setting command ,following message will send back to server:

$$ + L + ID + 0x4103 + Flag(1B) + checksum(2byte) + \r\n

Flag： =0 means setting failed
=1 means setting success

(6).Setting speed limit of over speed alarm

Command code:0x4105

For example:

Command from server to tracker units:

@@ + L + ID + 0x4105 + 1Byte speed limitation(Hex code) + checksum(2byte) + \r\n

1Byte speed limitation: when speed over speed limitation, a alarm message will send to server

Speed limitation=0: cancel over speed alarm function
0x01:speed limitation set to 10km/hour
0x02:speed limitation set to 20km/hour
0x03:speed limitation set to 30km/hour
0x04:speed limitation set to 40km/hour
0x05:speed limitation set to 50km/hour
…….
Max speed limitation is 0x14,which means speed limitation to 200km/hour

Message from tracker unit to server

$$ + L + ID + 0x4105 + Flag(1byte) + checksum(2bytes) + \r\n

Flag： =0 means setting speed limitation failed
=1 means setting speed limitation success

(7).Set Geo-fence

Command code:0x4106

For example:

Command from call center to tracker units

@@ + L + ID + 0x4106 + 1Byte Geo-fence area (in hex code)+ checksum(2byte) + \r\n

1 Byte Geo-fence area:
=0: cancel geo-fence function
=1:geo-fence area is set in a square with Side length=2*30m,set current point as center
=2: geo-fence area is set in a square with side length=2*50m,set current point as center
=3 : geo-fence area is set in a square with side length=2*100m,set current point as center
=4: geo-fence area is set in a square with side length=2*200m,set current point as center
=5: geo-fence area is set in a square with side length=2*300m ,set current point as center
=6 :geo-fence area is set in a square with side length=2*500m,set current point as center
=7 :geo-fence area is set in a square with side length=2*1000m,set current point as center
=8 :geo-fence area is set in a square with side length=2*2000m,set current point as center

Message from track unit to server

$$ + L + ID + 0x4106 + Flag(1B) + checksum(2byte) + \r\n

Flag： =0 means setting speed limitation failed
=1 means setting speed limitation success

(8).Set extend function

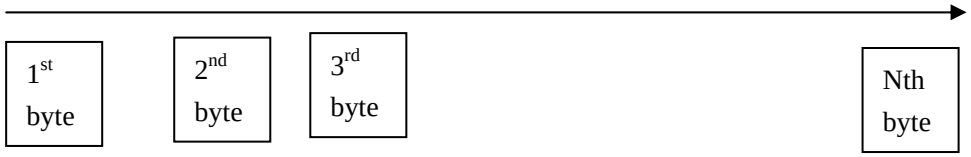
Command code:0x4108

For example:

Command from server to tracker units

@@ + L + ID + 0x4108 + extend function flag + checksum(2byte) + \r\n

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 NMEA 0183 GPRMC protocol

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

3rd btye=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

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

$$ + L + ID + 0x4108 + Flag(1B) + checksum(2byte) + \r\n

Flag： =0 means setting speed limitation failed
=1 means setting speed limitation success

(9). Init all parameter beside password, IP/PORT/APN setting ,ID,GPS enable/disable setting, timer interval setting of continuous tracking

Command code:0x4110

For example:

Command from server to tracker units:

@@ + L + ID + 0x4110 + checksum(2byte) + \r\n

Message from tracker units to server to confirm setting:

$$ + L + ID + 0x4110 + Flag(1B) + checksum(2byte) + \r\n

Flag： =0 means setting speed limitation failed
=1 means setting speed limitation success

(10).Set power saving mode

Command line:0x4113

For example:

Command from server to tracker units:

@@ + L + ID + 0x4113 + 1B power saving level(in hex code) + check sum(2byte) + \r\n

1Byte power saving level
=0 close power saving function
=0x01 low power saving level
=0x02 middle power saving level
=0x03 high power saving level

For more info about power saving level, please look at GT30 user manual

Message from tracker units to server:

$$ + L + ID + 0x4113 + Flag(1B) + checksum(2byte) + \r\n

Flag： =0 means setting speed limitation failed
=1 means setting speed limitation success

(11).Read preset time interval of continuous tracking

Command code:0x9002

For example:

Message from server to tracker units

@@ + L + ID + 0x9002 + checksum(2byte) + \r\n

Message from tracker units to server

$$ + L + ID + 0x9002 + 2Bytes preset time interval of continuous tracking (in hex code)+ checksum(2byte) + \r\n

2Bytes time interval: 2 Byte timer interval: in format of 10 seconds
=0 mean stop tracking
The max timer interval =65535*10 seconds

(12).Read authorized telephone number

Command code:0x9003

For example:

Message from server to tracker units:

@@ + L + ID + 0x9003 + 1B the name of authorized number(hex code) + checksum(2byte) + \r\n

1Bytes the name of authorized number, the name should from 1 to 9.When name=0xff,means read all authorized number

Message from track unit to server:

$$ + L + ID + 0x9003

+ 16Bytes SMS authorized number(in ASC II code)

+ 16Bytes Call authorized number (in ASCII code)

+ checksum(2byte) + \r\n

The authorized number is in ASCII code and the empty byte will be stuffed by 0x00

(13).Read speed limitation setting of over speed alarm

Command code:0x9005

For example:

Message from server to tracker units

@@ + L + ID + 0x9005 + checksum(2byte) + \r\n

Message from track unit to server

$$ + L + ID + 0x9005 + 1B speed limitation of over speed alarm

+checksum(2byte) + \r\n

Speed limitation =0: cancel over speed alarm function
0x01:speed limitation set to 10km/hour
0x02:speed limitation set to 20km/hour
0x03:speed limitation set to 30km/hour
0x04:speed limitation set to 40km/hour
0x05:speed limitation set to 50km/hour
…….
Max speed limitation is 0x14,which means speed limitation to 200km/hour

(14).Read setting of Geo-fence alarm

Command code:0x9006

For example:

Message from server to tracker units

@@ + L + ID + 0x9006 + checksum(2byte) + \r\n

Message from track units to server

$$ + L + ID + 0x9006 + 1B Geo-fence setting + checksum(2byte) + \r\n

1 Byte Geo-fence area:
=0: cancel geo-fence function
=1: geo-fence area is set in a square with Side length=2*30m,set current point as center
=2: geo-fence area is set in a square with side length=2*50m,set current point as center
=3 : geo-fence area is set in a square with side length=2*100m,set current

point as center

=4: geo-fence area is set in a square with side length=2*200m,set current point as center

=5: geo-fence area is set in a square with side length=2*300m ,set current point as center

=6 : geo-fence area is set in a square with side length=2*500m,set current point as center

=7 : geo-fence area is set in a square with side length=2*1000m,set current point as center

=8 :geo-fence area is set in a square with side length=2*2000m,set current point as center

(15). Read setting of extend function

Command code:0x9008

For example:

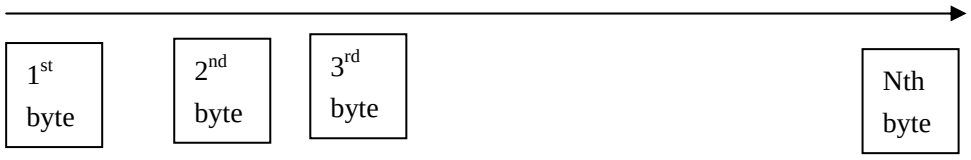
Message from server to tracker units

@@ + L + ID + 0x9008 + checksum(2byte) + \r\n

Message from track units to server:

$$ + L + ID + 0x9008 + extend function flags + checksum(2byte) + \r\n

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 NMEA 0183 GPRMC protocol

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

(16).Read setting of power saving mode

Command code:0x9013

For example:

Message from server to tracker units:

@@ + L + ID + 0x9013 + checksum(2byte) + \r\n

Message from tracker units to server

$$ + L + ID + 0x9013 + 1B power saving mode + checksum(2byte) +

\r\n

1B power saving level
=0 close power saving function
=1 low power saving mode
=2 middle power saving mode
=3 high power saving mode

(17). Alarm

Command code:0x9999

Message from tracker units to server:

$$ + L + ID + 0x9999 + 1B alarm types + GPRMC info + | + HDOP

+ checksum(2byte) + \r\n

1B Alarm types:
=0x01 SOS button is pressed
=0x02 2nd button(call B)is pressed
=0x03 3rd button(call C) is pressed
=0x04 Reserved
=0x10 Low battery alarm
=0x11 over speed alarm
=0x12 Geo fence alarm
=0x13 Reserved
=0x14 Tracker units reset or power up alarm
=0x15 No GPS signal alarm
=0x16 Out of no GPS signal area alarm