Vehicle GPS Tracker AT09 Protocol

Version 2.41

content

Document Revision

DATE	REVISION	CHANGE LIST
21ST MAY 2016	1.8	Update AA data type
23RD MAY 2016	1.9	Add temperature sensor 2 alarm in data type
22ND JUNE 2016	2.0	Add TPMS data format
30TH JULY 2016	2.1	Revise TPMS data string; Add idle alert command.
5TH AUG, 2016	2.2	Revise TPMS data string.
30TH AUG, 2016	2.3	Add 134 command to set fuel data reading
6TH SEP 2016	2.31	Update the command list
14TH OCT 2016	2.32	Explain Vehicle which one is 1st bit, which one is 32nd bit.
14TH DEC 2016	2.33	Add 104 command
18TH JAN 2017	2.33	ADD 140, 141 command and mobileye events
11TH APR 2017	2.34	Fix mistake; Data convert to decimal: -----→ Data convert to binary: in Page 9
3RD MAY 2017	2.35	Update temperature sensor alarm code Add 143, 144, 145 command
19THJUNE 2017	2.36	Specify data type for RFID reader, iButton reader, HID reader
28TH JULY 2017	2.37	Correct geofences in&out alarm data. The correct one: GEO in alarm: 04; GEO out alarm: 05
26TH AUG 2017	2.38	1. Revise sleep function description 2. revise harsh brake alarm description
14TH NOV 2017	2.39	Revise the description error: data string sent by data to server
23RD NOV 2017	2.40	Add 058 command: *000000,058,State,Content#
8TH FEB 2018	2.41	1. revise 19 bit meaning of VehicleStatus

1. Overview of AT09

Hardware Spec	
MCU	STM32F103VCT6 (ARM 32-bit Cortex™-M3 CPU)
GSM	GSM/GPRS; Quad-band: 850/900/1800/1900Mhz
GPS	*UBlox* G7020 KT
G-sensor/3-axis sensor	Built in for sleep and trembles alert
GSM antenna	External
GPS antenna	External
PIN IO interface	4 digital input
	4 digital output
	4 analog input
	1 panic button
	2 1-wire for iButton and temperature probe
	1 RS485 for capacitor fuel sensor (able to cut to suit the tank)

	2 RS232 for RFID, Camera, dispatch screen and other serial port device with customize firmware 1 speaker and microphone 1 Micro USB (No driver need, totally a HID/USB device)
Dimension	93mm*82mm*30mm
Firmware feature	
Firmware upgrade	By USB or by GPRS remotely(OTA)
Working parameter	Request the working remotely in configurator tool
Communication	TCP/UDP, SMS
Report interval	By time/distance combines with angle/veer
Sleep/deep sleep	Sleep
SMS alarm	Send out alarm when vehicle status has been changed Ignition on/off, GPS antenna drop, IO port changed, low battery
Voice monitoring	Two way conversation: build in authorized phone list.
Digital output/input	Enable / disable

2. AT09 communications

Data format summary: $$<length><data_type><data><checksum>

Remarks: (no "<" and ">" in real data strings)

2.1 checksum for all GPRS strings

String	Note
$$	Head (2 byte)
length	Length for whole string including checksum (4 byte)
data_type	Data type. Like AA, AB, AC
Data	Data

Checksum	Checksum (2 byte).
	Below is the function in C language how Checksum acquired:
	Unsigned char Checksum (const char *s,int nLength)
	{
	Unsigned char result;
	result=0;
	for(int i=0;i<Length;i++)
	{
	result ^=*s++
	}
	return result;
	}
	//**********
	Note: nLength here in this function starts from $ to checksum, but not including 2bytes checksum;
	And Length = nLength +2;
	e.g., data string: $$0030CF864244023027914011010 7
	char* s="$$0030CF8642440230279140110107";
	int nLength=strlen(s);
	
	checksum is "07" if we see the data string: $$0030CF864244023027914011010 7;
	unsigned char strChecksum=Checksum(s, Length);
	After strChecksum is known, you append strChecksum in **hex format** to string s:
	$$0030CF864244023027914011010 7;

2.2 Standard GPRS Data String Format

2.2.1 Standard Data Format Received in Server:

<$$><length><DataType><IMEI><|><VehicleStatus><Time><BatteryVoltage><SupplyVoltage><ADC1><ADC2><ADC3><ADC4><Temperature1><Temperature2><LACCI><CellID><GPSSatellite><CSQ><
Angle><Speed>< HDOP>
<Mile><Latitude><N/S><longtitude><E/W> <SerialNumber><checksum>

Remarks: (no "<" and ">" in real data strings)

Data String Example:
$$0128AA864244026065291|18001800140916020524401100000000000000000000000027BA0E57063100000001.20000000 2237.8119N11403.5075E05202D

Code	Description	Length(byte)	e.g.
$$	Header	2	***$$***
Length	Total length is for the entire string including the Header, Length and Checksum.	4	***0128***
DataType	Package flag ***Refer to alarm code list***	2	***AA***
IMEI	**Device ID** IMEI number(15 digits) in default, or 1-17digts ID if you set your own ID)	N/A	***864244026065291***
|	separator	1	
VehicleStatus	Vehicle status; HEX code Detailed Alarm code please refer to ***below***	8	***18001800***
Date/Time	UTC Date/Time: YYMMDDHHMMSS Example: YY (year 00-99): 14 MM (month 01-12): 07 DD (day 01-31): 29 HH (hour 00-23): 08 MM (minute 00-59): 20 SS (second 00-59): 13	12	***140916020524***
Battery Voltage	Battery Voltage Format: BB example：42 42 means 4.2V	2	***40***
Supply Voltage	Supply voltage Format: CC 11 means 11V	2	***11***
ADC1	Collected voltage AAAA Actual voltage: AA.AA	4	***0000***

	0006 means 0.06V		
ADC2	BBBB Actual voltage: BB.BB 0006 means 0.06V	4	*0000*
ADC3	CCCC Actual voltage: CC.CC 0008 means 0.08V	4	*0000*
ADC4	DDDD Actual voltage: DD.DD 0002 means 0.02V	4	*0000*
Temperature A	1-Wire Temperature sensor Range -55.0~ 125.0 0293 means 29.3°C If temperature under 0°C, it will show like -100	4	*0000*
Temperature B	1-Wire Temperature sensor Range -55.0~ 125.0 0292 means 29.2°C If temperature under 0°C, it will show like -100	4	*0000*
LACCI	Location area code	4	*27BA*
Cell ID	Cell ID	4	*0E57*
GPS Satellites	Current available GPS satellites that can be used Value range: (00 ~ 12)	2	*06*
GSM signal	Strength of GSM signal (0 ~31)	2	*31*
Angle	Veer: direction in degrees Value range:(000 ~ 359)	3	*000*
Speed	Actual GPS speed when record is generated. Unit: KM/H	3	*000*
HDOP	HDOP Value range: (00.0 ~ 99.9)	4	*01.2*
Mileage	Mileage between current string and last string Unit: meter	7	*0000000*
Latitude	Latitude. Format: DDMM.MMMM 2237.8119 means 22degree 37.8119minute	9	*2237.8119*
NS	North/South N = North S = South	1	*N*
Longitude	Longitude Format: DDDMM.MMMM 11403.5075 means 114degree 3.5075minute	10	*11403.5075*

EW	East/West E = East W = West	1	*E*
SerialNumber	0001-9999	4	***0520***
Checksum	**Checksum**	**2**	***2D***

2.2.2 Alarm Code List

Normal interval data	AA
Angle data / course change data	AB
Distance data	AC
SOS alarm	01
Over speed alarm	02
Speed recover alarm	03
GEO in alarm	04
GEO out alarm	05
TOW alarm	06 (if digital input 1 not triggered, and move 200 meters, TOW alarm triggered.)
GPS disconnect alarm	07
GPS reconnect alarm	08
GPS Module Alarm	09
External Power disconnect alarm	10
External power reconnect alarm	11
External power low level alarm (<10V)	12
Battery low voltage alarm (<3.6V)	13
Entry sleep	14
EXIT sleep	15
Digital input 1 ON	21
Digital input 1 OFF	22
Digital input 2 ON	23
Digital input 2 OFF	24
Digital input 3 ON	25
Digital input 3 OFF	26
Digital input 4 ON	27
Digital input 4 OFF	28
Shock alarm	40
Idle alarm	41
Harsh acceleration alarm	42
Harsh brake alarm	43
Temperature sensor A High alarm	44
Temperature sensor A Low alarm	45
Temperature sensor A resume alarm	46
Temperature sensor B High alarm	47
Temperature sensor B Low alarm	48
Temperature sensor B resume alarm	49
MOBILEYE FCW	**80**
MOBILEYE UFCW	81
MOBILEYE PCW	82

MOBILEYE LLDW	83
MOBILEYE RLDW	84
MOBILEYE HMW	85
MOBILEYE SLI	86

2.2.3 VehicleStatus

Note: in HEX code, status code 01 is at high bit, status code 32 is at low bit. The leftmost bit is 01 status code, the rightmost bit is 32 status code.

Original data in example in HEX: 14012000
Data convert to binary:
14012000 (HEX) → 0001 0100 0000 0001 0010 0000 0000 0000 (Binary)

Status code	description	Note
01	Panic button (SOS)	1=SOS 0= without SOS
02	ACC	IN1 (digital input 1) 1= ignition ON 0= ignition OFF
03	Over speed	1= over speed 0= without over speed
04	External power	1= with supply power 0= without supply power
05	GEO fence（out）	1=out alarm 0= none out alarm
06	GEO fence（in）	1= GEO in alarm 0= without in alarm
07	GPS antenna drop	1= GPS antenna drop alarm 0= GPS antenna is ok
08	GPS module error	1= module error 0= GPS module is ok
09	Output1	1= digital output1 in high level 0= digital output1 in low level
10	Output2	1= digital output2 in high level 0= digital output2 in low level
11	Output3	1= digital output3 in high level 0= digital output3 in low level
12	Output4	1= digital output4 in high level 0= digital output4 in low level
13	IN2 (digital input 2)	1= digital intput2 has been triggered 0= digital input2 without triggering
14	IN3 (digital input 3)	1= digital intput3 has been triggered 0= digital input3 without triggering
15	IN4 (digital input 4)	1= digital intput4 has been triggered 0= digital input4 without triggering
16	Shocking	1= shocking alarm 0= without shocking

		(Status will show '1' if it's in shocking even shock alarm has been disabled.)
17	Idle	1= Idle speed alarm (if the digital input 1 triggered and 10 minutes not moving, idle speed alarm generated.) 0= without idle speed
18	Inner battery in low level	1= inner battery voltage is low 0= inner battery voltage is ok
19	authorized Driving	1= Authorized driving 0= Unauthorized driving
20	GPS status	1= GPS fixed 0= No GPS fixed
21	Inner batter in charging	1= battery charge 0= batter not charging
22	GSM jamming detection	1= GSM jamming available 0= GSM jamming available
23	reserve	
24	reserve	
25	reserve	
26	reserve	
27	reserve	
28	reserve	
29	reserve	
30	reserve	
31	reserve	
32	reserve	
Other alarm events can be customized upon the detailed needs.		

2.2.3 Acknowledge Response for Normal Data String (From server to device):

Format: $$<Length><AA><SerialNumber><checksum>

Remarks: (no "<" and ">" in real data strings)

e.g.

data string sent by device to server:

($$0128AA867965020488181|1880180015091407364538110000000000000000000000027BA0E4E041200600000.00000002237.7611N11403.5196E00605B)

the server response for this data string: $$0014AA006003

e.g.	$$0014AA006003		
item	e.g.	length	description
Header	$$	2 bytes	Header
Length	0014	4 bytes	Length of whole string
Data Type	AA	2 bytes	Data Type
Serial number	0060	4 bytes	Serial number of received data string received in server.
CheckSum	03	2 byte	checksum

Remarks:

1. in default settings of standard firmware, users don't need to take care of this response format. This acknowledge response designed for for the purpose of making sure every data received in server. If you want to enable this function, please set command: *000000,019,0# (via SMS).
2. if ACK enabled, server should send Acknowledge Response via gprs if getting data including AA interval data, and alarm data(for alarm code, you can check here), excluding data E2, E3, E4, E5, E6 and heartbeat data BB.
3. The data type of ACK response is AA always even the data is AA, AB, AC or other alarm data.

2.3 iButton / RFID data Format received in server

Data Type: E2

2.3.1 RFID / iButton data format

e.g.	$$0077E2863835025116042|160621021914,114.058624,22.629075,0000000000000000|54		
item	**e.g.**	**length**	**description**
Header	$$	2 bytes	Header
Length	0077	4 bytes	Length of whole string
Data Type	E2	2 bytes	Data Type. E2: login authorized; (RFID reader; HID reader, iButton on with E2, iButton off with 000000; magnetic reader) E3: login unauthourized; (magnetic reader) E7: logout. (for HID reader; magnetic reader)
ID	863835025116042	variable length	Device ID. It's IMEI number normally.
Separator	|	1 byte	Separator
Time	160621021914	12 bytes	Time: YYMMDDHHMMSS. GMT TIME
Separator	,	1 byte	Separator
Longitude	114.058624	variable length	Longitude. In DDD.DDDDD format. "-" before longitude means West
Separator	,	1 byte	Separator
Latitude	22.629075	variable length	Latitude. In DD.DDDDD format. "-" before latitude means south
Separator	,	1 byte	Separator
RFID	0000000000000000	variable length	RFID / iButton ID
Separator	|	1 byte	Separator
Checksum	38	2 bytes	checksum

If you want to set or erase the iButton/RFID number in device, please use the 053 command.

2.4 Heart Beat/Keep-alive data format received in server

In order to always keep GPRS connection alive, device has provided another package for this purpose

Device send to server : <$$><0025><BB><IMEI><checksum>

Server response not required for heartbeat/keep-alive data.

2.5 Images Data Format Received in Server

Data Type: E4

e.g.	$$0494E4123456789012345|150425223945, 12,12,0,0,00112233445566778899AABBCC|38		
Item	**e.g.**	**Length**	**description**
Header	$$	2 bytes	Header
Length	0494	4 bytes	Length of whole string
Data Type	E4	2 bytes	Data type. E4 means Image data string
ID	123456789012345	15 bytes	IMEI or ID
Separator	|	1 byte	separator
Time	150425223945	12 bytes	YYMMDDHHMMSS. GMT time
Separator	,	1 byte	Separator
Length	12	Not fixed	The length of whole image packets
Separator	,	1 byte	Separator
Length	12	Not fixed	The length of current packet
Separator	,	1byte	Separator
PacketNum	0	Not fixed	Packet number
Separator	,	1 byte	Separator
Packet Status	0	1 byte	Image packet status: 0, the first packet of image; 1, the middle packet of image; 2, the last packet of image
Separator	,	1 byte	Separator
Image Data	├?[????XS?)??k??8? aO?┐?(?┐?(?┐?(?┐?*???j?? @?	Not fixed	Image data content string. In hex.
Separator	|	1 byte	separator
checksum	38	2 bytes	checksum

Remarks:

picture data is 450 bytes long in each data string except the last data string which may be less than 450 bytes.

Picture related commands:

Taking picture command: *000000,802#

Pixel setting of 320*240: *000000,803,0#

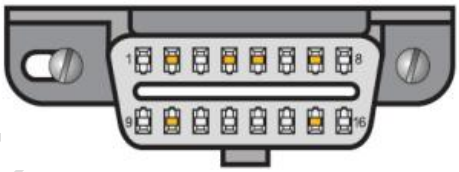
Pixel setting of : 640*480: *000000,803,1#

e.g.

$$0952E4867965023081371|160407025754,6659,900,5,1,c?Em#?????i? E!;~??Q@¶Q@G<?M#t↔?h┐?
??,?q'Rx?(?┐?(?┐?(?┐?(?┐??¶??┤h????n???Ej3?\?Is@◀=D?????└□???r?:u?????(???????◀?? ?t??/?)???
4P E¶Px├?[????XS?)??k??8?
aO?┐?(?┐?(?┐?(?┐?*???j??@?>U?+?→???|t?j?P┘mP?0?□UU???Bb?M?)?┼G?? ┴?iZ??Q¶?↕?? ?E┴V???◀?
%_?}}?┐??v???|?^??*??┐?(??F?$m└y?x??h2?+u↔(└^?(?┐?(?┐?(?┐?(?┐?4bh?6$─↑??q??M?y$a????→?"+
e`8$??1?")??Z???
aQ53¶?P┘???m"?"?~?O?E^B?O??:??:?:????????‼?V???)???>tC↕N?%????n$^????,Q@G<?(????9????
??[???n?S@‼Q@¶Q@¶Q@¶Q@¶Q@¶ ??@???└?□:???@┴??a@◀?i?
? E?izt?(%?D↕x??Q?{→??@0}(┐[↔Q?`hB─┘?[?w?kV?TIp?└7??i? ??E┴┘??8?/???@↕?n\?
??P Y???HV◀????(┐=┬,┬v└????┐????8=?┐??^#???;2????E?? ??
Q@┴/.'?←?└??*?Z?????>?(┐?^?'(?????d'??0=
-!8?"?┬?)??\ H??└?┘u┘?}N?7O?o8←??\??@◀0???"c?0??<2??+?d└????*??┬??<c?(└*?A#-o&□?j???{f???
???@→Zu??vI?'??m?↑┐A?h h?┘?9¶? ┴??o◀??N???-?}F??'??[?zP─?????└??@¶???
??2Y?‼??B0@@0015E1SEND OK@@0012B30001@@0116B5$$0108AA86

For server response of image data string, device will take care of the data sending loss issue. So no need to do anything by users.

2.6 OBD Data Format Received in Server



Data Type: E5

e.g.	$$0129E5867965026718441|160518070415,BD$V13.6;R1290;S32;P34.5;O92.5;C80;L40.4;XH4.94;XM15.43;A0;B14;D0;MS1.40;FS0.300;TS0.10;SS1;@6M|3A		
Item	**e.g.**	**Length**	**description**
Header	$$	2 bytes	Header
Length	0129	4 bytes	Length of whole string
Data Type	E5	2 bytes	Data type. E5means OBD data string
ID	867965026718441	15 bytes	IMEI or ID
	|	1 byte	separator
Time	160518070415	12 bytes	YYMMDDHHMMSS. GMT time
	,	1 byte	Separator
Car battery volt	BD$V13.6	Not fixed	Car battery volt: 13.6v
	;	1 byte	Separator
RPM	R1290	Not fixed	RPM is 1290
	;	1byte	Separator
Speed	S32	Not fixed	Speed is 32KM/H
	;	1 byte	Separator
Throttle position	P34.5	Not fixed	throttle position is 34.5
	;	1 byte	Separator
Engine Loading	O92.5	Not fixed	Engine loading is 92.5
	;	1 byte	
Coolant temperature	C80	Not fixed	Coolant temperature is 80 celsius degree

	;	1 byte	separator
Fuel	L40.4	Not fixed	Fuel remain in percentage : 40.4%
	;	1 byte	separator
Fuel consumption on idle time	XH4.94	Not fixed	Fuel consumption on idle time is 4.94L/H
	;	1 byte	separator
Fuel consumption on 100KM	XM15.43	Not fixed	Fuel consumption on 100KM: 15.43 L/H (instantaneous value)
	;	1 byte	separator
Harsh acceleration amount	A0	Not fixed	Harsh acceleration amount
	;	1 byte	separator
Harsh brake amount	B14	Not fixed	Harsh brake amount: 14
	;	1 byte	separator
Error amount	D0	Not fixed	Error amount: 0
	;	1 byte	separator
Mileage accumulation	MS1.40	Not fixed	Mileage accumulation: 1.40KM
	;	1 byte	separator
Fuel consumed accumulation	FS0.300	Not fixed	Fuel consumed accumulation: 0.3L
	;	1 byte	separator
Ignition time	TS0.10	Not fixed	Ignition time: 0.1hour
		1 byte	separator
Ignition amount	SS1	Not fixed	Ignition amount: 1 time
	;	1 byte	separator
	@6M	Not fixed	Checksum from OBD module, which can be ignored.
	|	1 byte	separator
checksum	3A	2 bytes	checksum

Remarks:

• For server response of image data string, device will take care of the data sending loss issue. So no need to do anything by users.

• there are 16 parts of OBD data in standard, but sometimes it will be 15 parts in data if it doesn't fetch data from car successfully. So programmer should check each part of OBD data first when you parse the data. Like sometimes O92.5 **will not show in data.**

2.7 Tire pressure monitoring data Format received in server

Data type: E6

e.g.	$$0100E6863835025116042|150425223945,113.925525,22.55814,4,10,013006036030;023006036030;033006036030;223006036030|38		
item	**e.g.**	**length**	**description**
Header	$$	2 bytes	Header
Length	0100	4 bytes	Length of whole string

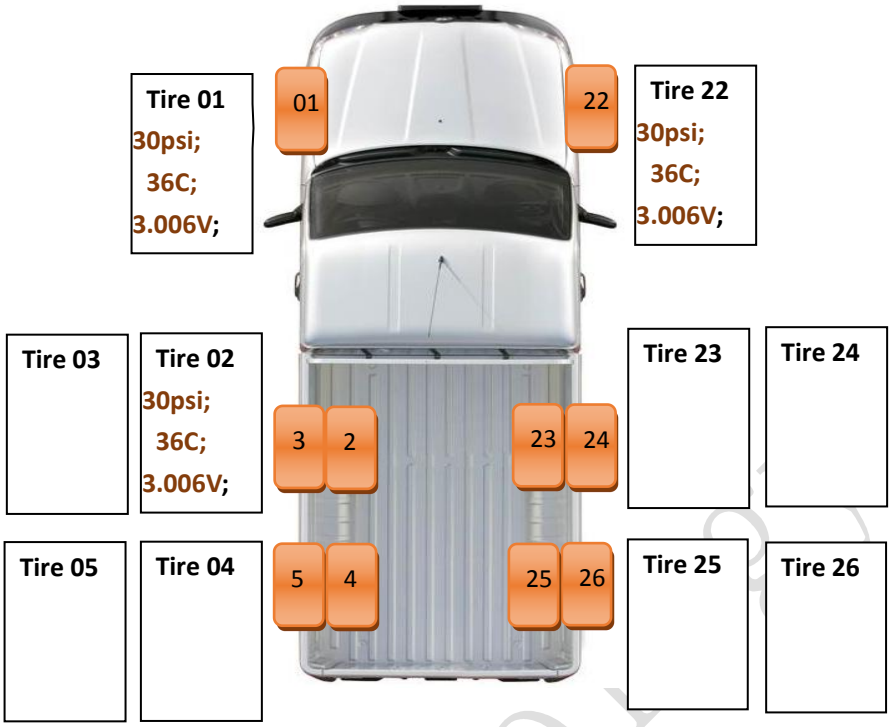
Data Type	E6	2 bytes	Data Type. E2 means the RFID data
IMEI	863835025116042	15 bytes	IMEI
	|	1 byte	Separator
Time	150425223945	12 bytes	Time: YYMMDDHHMMSS. GMT TIME
	,	1 byte	Separator
Longitude	113.925525	No fixed length	Longitude. In DDD.DDDDD format. "-" before longitude means West
	,	1 byte	Separator
Latitude	22.55814	No fixed length	Latitude. In DD.DDDDD format. "-" before latitude means south
	,	1 byte	Separator
RealCount	3	Not fixed length	real count tire sensors from which the GPS receives data
	,	1 byte	Separator
TotalTireCount	10	Not fixed length	the count of tires in vehicle
	,	1 byte	separator
Tire 01 data	01	2 bytes	Tire code number:
	3006	4 bytes	3.006V for battery in sensor
	036	3 bytes	36°C for tire. If -36, means 36°C below zero
	030	3 bytes	30psi for tire pressure
	;	1 byte	separator
Tire 02 data	02	2 bytes	Tire code number:
	3006	4 bytes	3.006V for battery in sensor
	036	3 bytes	36°C for left front tire. If -36, means 36°C below zero
	030	3 bytes	30psi for tire pressure
	;	1 byte	separator
Tire 22 data	22	2 bytes	Tire code number:
	3006	4 bytes	3.006V for battery in sensor
	036	3 bytes	36°C for left front tire. If -36, means 36°C below zero
	030	3 bytes	30psi for tire pressure
	;	1 byte	separator
Separator	|	1 byte	Separator
Checksum	38	2 bytes	checksum

Note:

1. If no TPMS data for one tire received in GPS tracker, data about this tire including tire code number is not shown in GPRS data string. In the string example, only Tire01, Tire02, Tire22 TPMS data available in data string.

2. The tires rows are marked at zero based. It means the headmost row is 0th row.

3. the default maximum tires amount is 42. Tires on left side starting from number 1 to number



3. Commands via GPRS and SMS

You can send commands via GPRS or SMS.
e.g., command 016 setting output A state:
in GPRS mode: **$$0024CF000000,016,A,168**
in SMS mode: ***000000,016,A,1#**

3.1 Commands via GPRS:

3.1.1 Format of command sent from server to device:

$$<length><CF><******,Command_content><checksum>

Remarks: (no "<" and ">" in real command)

****** represents the password (6 digits)

e.g.	$$0029CF000000,052,1234567831		
Item	**e.g.**	**Length**	**description**
Header	$$	2 bytes	Header
Length	0029	4 bytes	Length of whole string in decimal format
Data Type	CF	2 bytes	Command type
Command	000000,052,12345678	15 bytes	Command content
Checksum	31	2 byte	checksum

3.1.2 Format of device response of command to server:

Command received ok: $$<0030>CF<IMEI><CommandNumber>01<checksum>
Command password error: $$<0030>CF<IMEI><CommandNumber>02<checksum>
Command format error: $$<0030>CF<IMEI><CommandNumber>03<checksum>
Note: the length is 30 bytes always.

e.g.	**$$0030CF864244023027914011010 7**		
Item	**e.g.**	**Length**	**description**
Header	$$	2 bytes	Header
Length	0030	4 bytes	Length of whole string in decimal format
Data Type	CF	2 bytes	Data type
ID	864244023027914	15 bytes	ID of device(normally it's IMEI number)
Command Number	011	3 byte	Command_code_number
Execute_Status	01	2 bytes	Status code of command: 01: command OK 02: command password error 03: command format error
CheckSum	**07**	**2 bytes**	**checksum**

e.g.
server sends command to device via GPRS: $$0020CF000000,0111B
the response of device is: $$0030CF8642440230279140110107

3.2 Commands via SMS:

3.2.1 Format from User to device:

<*><Password><,><Command Content><#>

Remarks: (no "<" and ">" in real command)

e.g., *000000,016,A,1#

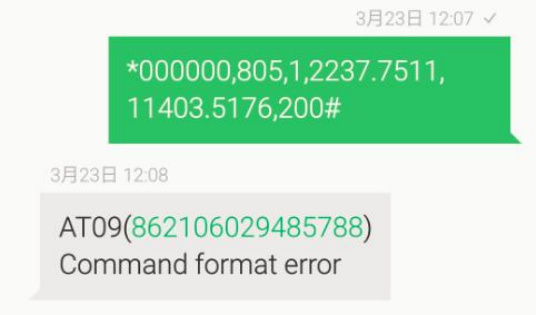
e.g.	*000000,016,A,1#		
Item	**e.g.**	**Length**	**description**
Header	*	1 byte	Header
Password	000000	6 bytes	Password. Default password is 000000
separator	,	1 bytes	separator
Command Content	016,A,1	Not fixed	Command content. Refers to here
Tail Separator	#	1 byte	Tail separator

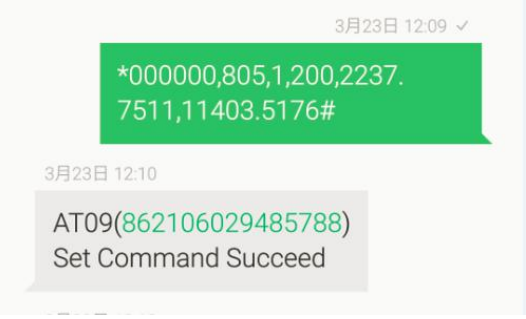
3.2.2 Format of device response to user cellphone via SMS:

"Set Command Succeed" (command ok)
"Command password Error" (Password error)

"Command Format error" (format error)

e.g.,





3.3 Commands List

(in column item: item with prefix * can not be used in OTA mode. Commands with prefix $ can be used in gprs. Without any prefix, means it's universal in OTA mode, gprs mode, sms mode)

Note: Default password is 000000 (6 zero)

No.	Command Description	Command Content	Remarks
001	SIM PIN	001,PinCode	PIN code should <=4 digitals numbers
*002	APN	002,APN,UserName,Password	APN[1,30] User Name: [1,20] Password[1,20]
*003	GPRS IP setting	003,ip,port,protocol,X	**IP(or DNS)**: [1,30]; **Port**: [1,5] **Protocol**= 0: TCP = 1: UDP **X** = 0: DNS; = 1: IP
$004	OTA upgrade	004	
$005	OTA disable	005	
*006	Restart device	006	*@@@@@@,006#（sent by SMS. It works only when GSM module is in working status.）
*007	Initialize device	007	*000000,007#
008	Read parameter of each command	008,X	X means command number. Read APN: 008, 002

No.	Command Description	Command Content	Remarks
*011	Clear inner memory	011	*000000,011#
012	Request coordinate	012	*000000,012#（works in sms mode）
013	Request the configuration	013	*000000,013#(works only in configuration software and through gprs), kindly note the command format in configuration software and gprs is different.
014	Change password	014,new_password	*000000,014,$$$$$$# password: 6 digits
016	Digital output control	016,OutputNumber,state	**OutputNumber**: (A,B,C,D) **state**: (1 enable,0 disable) (only works in user define mode, or it will return format error)
018	Set mileage	018	Reserved, no effect if device receives this command.
019	Set ACK	019,state	**State**: (0 enable,1 disable)
021	Request Google map link	021	
051	Set sensitivity of 3D sensor	051,SenseValue	**SenseValue**: sensitivity value[1, 30]
052	set the ID of device	052,ID	**ID**: letter and number, 1 to 17 digits
053	manage iButton/RFID number	053,SequenceNumber,State,Content	**SequenceNumber**: 01-100 **State**: 0, enable it; 1, disable it (set iButton/RFID number by setting state 1; erase iButton/RFID number by setting state 0.) **Content**: iButton number. 16 digits required. e.g., erase the 1st iButton/RFID number, you can set like this: *000000,053,1,1,0000000000000000# or even not type the iButton/RFID number: *000000,053,1,1,# (but don't forget the comma"," at the end)
054	Set DNS	054，DNS1,DNS2	
056	Request firmware version	056	
058	Set Ibutton/RFID ID	058,State,Content	**State**: 0, enable it; 1, disable it (set iButton/RFID number by setting state 1; erase iButton/RFID number by setting state 0.) **Content**: iButton number. 16 digits required. i.e. erase the an iButton/RFID number 12345678, you can set like this: *000000,058,1,0000000012345678#
*101	Set GSM Band	101,GSMBand	**GSMBand** =0: 900/1800 =1: 850/1900 =2: 850/900/1800/1900
102	Data sending mode	102,move,distance,angel,rest	**move**: interval time when vehicle is moving [1, 99999] secs. **distance**: data sending by distance: [1, 99999] meters **angel**: data sending by angel: [1, 359] degree. **rest**: interval time when vehicle is powered off and not moving. [1, 99999] secs
103	Set the sleep mode	103,mode,time	**mode**=0: no sleep mode =1: sleep mode. gps module turned off, gsm module are turned on and only heartbeat and alarm data available. =2; deep sleep mode. no data generated, and both gps module and GSM/3G module turned off.

No.	Command Description	Command Content	Remarks
			time: time after which device gets into sleep mode: [1, 9999]. device will wake up from sleep mode if tremble is intense.
104	Set over speeding alarm	104,state,OverSpeed	**state** =0: enable, =1: disable **OverSpeed**: over speeding value: [1-999] meter
105	Set the phone numbers for call	105, state,Num1,Num2,Num3	**state** =0: enable, (if no settings for Num1, Num2 and Num3, then device doesn't answer any numbers; if numbers set for Num1, Num2 and Num3, device will answer the call from Num1, Num2 and Num3) =1: disable. Means any numbers can call and listen in. digits of phone number(Num1, Num2, Num3): [6, 25] Can set at most three phone numbers: Num1, Num2, Num3. e.g., ● set one number:*000000,105,0,18675642745,,# ● set three number:*000000,105,0,123456,123457,123458# ● any numbers can be answered by device: *000000,105,1,,,# ● set no any numbers can be answered: *000000,105,0,,,#
106	enable GEO fence alarm	106,in_GEO,out_GEO	**in_GEO** =0: alarm for entering into geo fence; =1: disable it. **out_GEO** =0: alarm for out of geo fence; =1: disable it.
107	Set GEO fence alarm location and scope	107,radius,CenterLat,CenterLon	**radius**: GEO radius: [1, 9999]meter **CenterLat**: Latitude must be 9 digits. e.g., 2837.7172, (in degree+minute format) **CenterLon**: Longitude, must be 9 digits. e.g., 07722.6363. (in degree+minute format)
	ENABLE / DISABLE SMS FUNCTIONS (from command 109 to 122) e.g., SMS SOS(command 111) enabled, if SOS button pressed, device will send sms alarm to preset cellphone number; if disabled, no sms for SOS alarm to preset cellphone number		
108	GPS antenna disconnect alert	108,mode	**mode**=0 enable it; =1 disable it.
109	External power disconnect alert	109,mode	**mode**=0 enable it; =1 disable it.
110	SMS TOW alarm	110,mode	**mode**=0 enable it; =1 disable it.
111	SOS	111,mode	**mode**=0 enable it; =1 disable it.
112	SMS GeoIn	112,mode	**mode**=0 enable it; =1 disable it.
113	SMS GeoOut	113,mode	**mode**=0 enable it; =1 disable it.

No.	Command Description	Command Content	Remarks
114	SMS IN1_ON	114,mode	**mode**=0 enable it; =1 disable it.
115	SMS IN1_OFF	115,mode	**mode**=0 enable it; =1 disable it.
116	SMS IN2_ON	116,mode	**mode**=0 enable it; =1 disable it.
117	SMS IN2_OFF	117,mode	**mode**=0 enable it; =1 disable it.
118	SMS IN3_ON	118,mode	**mode**=0 enable it; =1 disable it.
119	SMS IN3_OFF	119, mode	**mode**=0 enable it; =1 disable it.
120	SMS Low battery	120, mode	**mode**=0 enable it; =1 disable it.
121	SMS IN4_ON	121,mode	**mode**=0 enable it; =1 disable it.
122	SMS IN4_OFF	122,mode	**mode**=0 enable it; =1 disable it.
123	set the phone number for receiving alarms by sms	123,PhoneNumber	**PhoneNumber**: cell number, (<= 25digits)
141	Set interval time for OBD data, harsh acceleration and harsh brake.	141,interval_OBD,acceleration,brake	**Interval_OBD:** [0, 999999] unit: seconds; if =0, means no OBD data sent out. **Acceleration:** [20, 100] unit: m/s^2*10 **Brake:** [20, 100] unit: m/s^2*10
	ENABLE / DISABLE IO FUNCTIONS(from command 125 to 134) e.g., IN1 enabled, if IN1 triggered, device will process the IN1 triggered; if IN1 disabled, no any actions taken by device as if IN1 defective		
125	Set IN1	125,mode	**mode**=0 enable it; =1 disable it.
126	Set IN2	126,mode	**mode**=0 enable it; =1 disable it.
127	Set IN3	127,mode	**mode**=0 enable it; =1 disable it.
128	Set IN4	128,mode	**mode**=0 enable it; =1 disable it.
129	Set SMS SOS	129,mode	**mode**=0 enable it; =1 disable it.
130	Set digital output1	130,mode	**Mode**= 0 default; for authorized driving; 1 defined by user; 2 disable it.

No.	Command Description	Command Content	Remarks
131	Set digital output2	131,mode	**Mode=** 0 default; for Buzzer controlling; if overspeeding, it will trigger the buzzer; =1 defined by user; =2 disable it.
132	Set digital output3	132, mode	**Mode=** 0 default; for LED; =1 defined by user; = 2 disable it.
133	Set digital output4	133,mode	**Mode**=0 default; for LED; =1 defined by user; =2 disable it.
134	Set fuel sensors data reading e.g., if your fuel sensors volt output is 0v to 5v, and you want to set fuel data shown in percent, and analog input 1 connected to fuel sensor, you can set like: *000000,134,0,1,0,5000# (in sms format)	134,InputNum,mode,DataFormat,Low,High	**InputNum=** 1 analog input1; = 2 analog input2; = 3 analog input3; = 4 analog input4; **Mode**=0 Enable in default; it will read data from analog input. =1 disable; it doesn't read data from analog input. **DataFormat** : 0 data in real value; : 1 data in percent; **Low**: lowest data [0, 60000]mV; **High**: highest data [0, 60000]mV;
135	Set server response time & heartbeat time	135,ServerResponse,Heartbeat	**ServerResponse**: [0, 60] second **HeartBeat**: [0, 999] second
136	Set distance for tow alarm	136,distance	**Distance**:[0, 5000] meter
140	Switch for data interval change by engine or tremble staus If you set 0 in 140 command, means device defines moving or parking by trembles: trembles(detected by 3d accelerometer and speed), it's moving; no trembles(detected by 3d accelerometer and speed),, it's parking. If you set 1 in 140 command, means device defines moving or parking by engine on /off: engine on(detected by digital input 1), it's moving; engine off(detected by digital input 1), it's parking.	140,switch	**Switch**=0 : data interval change by tremble status. =1: data interval change by engine status. There are two intevals: interval for moving status(referring to the move parameter in command: *000000,102,**move**,distance,angel,rest#); interval for parking status(referring to the **rest** parameter in command: *000000,102,move,distance,angel,**rest#**).
141	Set OBD, harsh acceleration and harsh brake	141,time,acc,brake	**time**=[0,99999]: device sends OBD data in the interval of time **acc=** [20,100]: 30 in default. Set value for harsh acceleration event **brake**=[20,100]:30 in default. Set value for harsh brake event.
142			
143	Tremble sensor data alarm	143,mode	**Mode: 0 = enable.** (tremble alarm 40 alarm type enabled) **1 = disable.** (tremble alarm 40 alarm type disabled)

No.	Command Description	Command Content	Remarks
144	Set temperature alarm value	144, tempID,mode,Low,High	**tempID** =1; temperature sensor A (related to 1-Wire-T in label) =2; temperature sensor B(related to 1-Wire-T in label) **Mode** =0, temperature alarm enabled. =1, temperature alarm disabled. **Low:** low alarm value (unit: ˚C) [-55 ˚C to 125 ˚C] **High:** High alarm value (unit: ˚C) [-55 ˚C to 125 ˚C] **e.g.,** *000000,2,0,-2,50# enable temperature B sensor alarm, low temperature at -2 degree, high temperature at 50 degree
145	Reboot GPS module	145	Reboot GPS module, just in case sometimes we received some wrong coordinate information.
200	Request IMSI	200	IMSI：International Mobile Subscriber Identification Number
201	Request CCID	201	**SIM CCID**
203	Set dual IP in the firmware	203,ip,port,protocol,X	
230	Set time for idle alert	230,Time	**Time:** [0,250] minute If digital input1 is triggered (means engine on), and speed is zero(not moving), the two conditions last certain time, the idle alert will be generated.
300	Time to reboot device	300,time	**Time in mins**
Set TPMS function			
301	Set tire pressure ID	301,X,ID	**X：1-42** **ID=8 byte** e.g., *000000,301,01, 1C9E8ED9#
303	Time to send tire pressure data to server	303,X	**Unit：s**
801	Device sends one sms to the typed phone number	801,CellNumber	Text message: "Read SIM Serial Number" **CellNumber**：Cellphone number. Device will send a sms to this number
Picture Taking function			
802	Take image	802	Device will take 1 image and send to server by gprs
803	Image pixel setting	803,pixel	**Pixel** =0: 320*240; =1: 648*480;
997	Send a RFID test data, ID number is 12345678	997	
998	Send a SOS data	998	
999	Send a normal AA data	999	

4. OTA upgrade

This part has been integrated in configuration tool
Working parameter requested remotely.
This part has been integrated in configuration manager.