**Example of VT200L GPRS Protocol Integration（Standard version）**

**Data Format：**

&&<pack-no><pack-len>,<ID>,<cmd>,<alm-code>,<alm-data>,<date-time>,<fix\_flag>,<latitude>,<longitude>,<sat-quantity>,<HDOP>,<speed>,<course>,<altitude>,<odometer>,<MCC|MNC|LAC|CI>,<CSQ-quanlity>,<system-sta>,<in-sta>,<out-sta>,<ext-V|bat-V|ad1-V|…|adn-V>,<pro-code>,<fule\_liter>,<temp-sensor><checksum>\r\n

**Original data from tracker to server:**

26 26 3C 31 35 33 2C 38 36 38 36 31 38 30 35 32 31 30 38 39 30 39 2C 30 30 30 2C 30 2C 2C 32 31 30 35 32 36 30 36 33 34 35 33 2C 41 2C 31 39 2E 39 35 36 32 38 35 2C 39 39 2E 38 36 30 30 30 38 2C 31 34 2C 31 2E 33 2C 30 2C 36 31 2C 33 39 30 2C 37 34 32 39 2C 35 32 30 7C 33 7C 33 39 30 38 7C 30 32 36 37 32 36 36 36 2C 32 38 2C 30 30 30 30 30 30 42 43 2C 30 32 2C 30 30 2C 30 35 30 38 7C 30 31 41 30 7C 30 30 30 30 7C 30 30 30 30 2C 31 2C 30 31 30 30 30 30 2C 30 31 30 31 30 39 44 39 0d 0a

**Convert the data in red from HEX to ASCII, we got following** **ASCII data:**

&&<153,**868618052108909**,000,0,,**210526063453**,A,19.956285,99.860008,14,1.3,0,61,390,7429,520|3|3908|02672666,28,000000BC,02,00,0508|01A0|0000|0000,1,010000,010109D9

**Head=**&& (data from tracker to server)

**Pack number:** < (in hex is 0x3C, server reply to server shall use the same pack number)

**Data length=153** ( 153 bytes )

**tracker id** --> **868618052108909** (default IMEI number)

**Command**=000 (normal data from tracker to server, and the platform does not need to reply to receive confirmation)

**Alarm code= 0**( refer to alarm list, 0 means interval report)

**Alarm Data**= non in this data

**Time =** 210526063453 ( means 2021, May, 26th, 06:34:53)

**Status = A** ( GPS signal status, V=invalid, A= Valid)

**Latitude =** 19.956285(Latitude, the unit is degree, decimal character format, if it is south latitude, add-sign in front)

**Longitude =** 99.860008 Longitude, the unit is degree, decimal character format, if it is west longitude, add-sign in front

**Satellite Quantity =** 14 ( 14 satellites)

**HDOP = 1.3**  Horizontal positioning accuracy, decimal character format, the smaller the value, the more accurate the accuracy

**Speed =** 0(speed 0, unit in km/h )

**Course=**61 (heading is 61 degree)

**Altitude**=390 (height is 390 meters)

**Odometer**=7429 (7429 meters)

**MCC|MNC|LAC|CI**=520|3|3908|02672666 Base station information, ‘|’ is used to separate each data symbol; MCC, MNC: decimal character format; LAC, CI: hexadecimal character format)

MCC=520, MNC=3, LAC=0x3908, CI=0x02672666

**CSQ = 28** (telecommunication strength is 28)

**System Data**=000000BC =(The working status of tracker, hexadecimal format, each bit represents a working status. Hex to binary, 0x000000BC =10111100, Bit0=0, GPRS 1 disconnected; Bit1=0, GPRS 2 disconnected; Bit 2=1, GPS signal valid; bit 3=1, external power connected; bit 4=1, GPS antenna connected; bit 5=1, vehicle is stop status; bit 6=0, disarmed; bit 7=1, RFID/iButton logged in)

**Input status= 02, Hex to Binary, 0x02=0010, bit 0=0, input 1 inactive; Bit 1=1, input 2 activated; bit 2=0, input 3 inactive**

**Input status= 00, Hex to Binary, 0x00=0000, bit 0=0, output 1 inactive; Bit 1=0, output 2 inactive**

**<ext-V|bat-V|ad1-V|…|adn-V>**=0508|01A0|0000|0000

External voltage =0508, hex to decimal, 0x0508=1288, then 1288/100=12.88V

Battery voltage=01A0,hex to decimal, 0x01A0=416, then 416/100=4.16V

**AD1 = 0000**

**Hex to decimal,** the follow up fuel sensor calculation format

**Example:** The fuel tank height is 100cm and full fuel is 50 liters, GPRS data as below:

&&A147,021104023195429,000,0,,180106093046,A,22.646430,114.065730,8,0.9,54,86,76,326781,460|0|27B3|0EA7,27,0000000F,02,01,04E2|018C**|01C8 |0000**,1,0104B0,01013D|02813546\r\n

AD1 voltage is **0x01C8** (hexadecimal)=456 (decimal), **AD1=456/100** (fixed value)=4.56V;

AD2 voltage is **0x0000=0**, **AD2=0/100=0V**;

**Example:** Connect the GPS Tracker analog input to capactive fuel sensor, the blue wire of the sensor are connected to the blue wires (AD) of the VT200 L.

When the fuel is empty, the sensor output voltage is 0V, when the fuel is full, the sensor output voltage is 5V

Calculate the percentage of remaining fuel:

**Fuel percentage=(AD/5)\*100%=(4.56/5)\*100%=91.2%.**

Calculate the remaining fuel in liters:

The remaining fuel in liters **= (AD/5) \* 50 liters = 45.6 liters**

**AD2 = 0000**

Reserved, normally 0

**pro-code**=1, Extended protocol version number to distinguish different extended protocol functions. Decimal format.

**<fule\_liter>**＝010000，the number of liters of the fuel tank. 010000 means 01 fuel tank with 0X0000 fuel liter (convert 0X0000 from Hex to decimal)

01# Fuel tank: 0x0000, the decimal value is 0000, the fuel volume is 0 liters

The 138, 139, and 140 commands must be used to set the fuel tank parameters to detect the liters.

**<temp-sensor>=010109** 01#Temperature sensor: 0x0109, the highest digit is 0 for positive temperature value, decimal value is 265, 265/10= 26.5℃

**<checksum>=**D9

The checksum of the data packet, 2-digit hexadecimal character format, all the data before the checksum

{*&&<pack-no><pack-len>,<ID>,000,<alm-code>,<alm-data>,<date-time>,<fix\_flag>,<latitude>,<longitude>,<sat-quantity>,<speed>,<course>,<altitude>,<odometer>,<MCC|MNC|LAC|CI>,<CSQ-quanlity>,<status>,<in-sta>,<out-sta>,<bat-ad|ext-ad|ad1|…|adn>,<pro-code>,<fule\_data>,<temp-sensor>*} accumulation and calculation result

**Example of data from server to tracker:**

**GPRS command packet format（from server to tracker）：**

$$<pack-no><pack-len>,<ID>,<cmd-code>,<cmd-data><checksum>\r\n

$$:44,868618052108909,100,1,istartracker.com,8011E5 （The pack-no must be the same as the pack-no sent by the device）

**Example of VT200L GPRS Protocol Integration（RFID）**

After connected wires, set the RFID authorization code firstly, and then set OUT1 to control the vehicles’ oil and electricity. And use the 212 command to set the 53 event of OUT1. When the RFID card is swiped, the 0x53 event is generated and the RFID data is reported.

**Data Format：**&&<pack-no><pack-len>,<ID>,<cmd>,<alm-code>,<alm-data>,<date-time>,<fix\_flag>,<latitude>,<longitude>,<sat-quantity>,<HDOP>,<speed>,<course>,<altitude>,<odometer>,<MCC|MNC|LAC|CI>,<CSQ-quanlity>,<system-sta>,<in-sta>,<out-sta>,<ext-V|bat-V|ad1-V|…|adn-V>,<pro-code>,<fule\_liter>,<temp-sensor><checksum>\r\n

Original data from tracker to server:

26 26 4D 31 34 37 2C 38 36 38 34 35 30 30 34 30 34 35 32 33 39 31 2C 30 30 30 2C 35 33 2C 36 34 37 31 31 35 2C 32 31 30 36 30 32 30 32 30 33 33 32 2C 41 2C 32 32 2E 36 37 39 30 34 35 2C 31 31 34 2E 30 34 36 32 39 31 2C 36 2C 31 2E 31 2C 30 2C 31 33 35 2C 37 30 2C 33 38 30 2C 34 36 30 7C 30 7C 32 34 39 46 7C 30 41 43 32 36 32 30 44 2C 32 37 2C 30 30 30 30 30 30 31 44 2C 30 30 2C 30 30 2C 30 34 41 32 7C 30 31 39 38 7C 30 30 30 30 7C 30 30 30 30 2C 31 2C 2C 43 37 0D 0A

Convert the data from HEX to ASCII, we got following ASCII data:

&&M147,868450040452391,000,53,647115,210602020332,A,22.679045,114.046291,6,1.1,0,135,70,380,460|0|249F|0AC2620D,27,0000001D,00,00,04A2|0198|0000|0000,1,,C7

**RFID=**647115 (Convert 647115 from Hex to decimal = 6582549 )

**Example of VT200L GPRS Protocol Integration（Ibutton）**

After connected wires, set the iButton authorization code firstly, and then set OUT1 to control the vehicles’ oil and electricity. And use the 212 command to set the 53 event of OUT1. When the RFID card is swiped, the 0x53 event is generated and the RFID data is reported.

Data Format：

&&<pack-no><pack-len>,<ID>,<cmd>,<alm-code>,<alm-data>,<date-time>,<fix\_flag>,<latitude>,<longitude>,<sat-quantity>,<HDOP>,<speed>,<course>,<altitude>,<odometer>,<MCC|MNC|LAC|CI>,<CSQ-quanlity>,<system-sta>,<in-sta>,<out-sta>,<ext-V|bat-V|ad1-V|…|adn-V>,<pro-code>,<fule\_liter>,<temp-sensor><checksum>\r\n

Original data from tracker to server:

26 26 51 31 34 39 2C 38 36 30 32 36 32 30 35 30 30 31 35 33 37 34 2C 30 30 30 2C 35 33 2C 30 31 33 39 32 38 31 31 2C 32 31 30 36 30 32 30 38 31 30 35 33 2C 41 2C 32 32 2E 36 37 37 36 36 31 2C 31 31 34 2E 30 34 36 32 37 31 2C 35 2C 32 2E 31 2C 30 2C 33 32 37 2C 31 30 30 2C 36 33 2C 34 36 30 7C 30 7C 32 34 39 46 7C 30 41 43 32 36 32 31 30 2C 33 31 2C 30 30 30 30 30 30 33 44 2C 30 32 2C 30 32 2C 30 34 41 32 7C 30 31 37 44 7C 30 30 30 30 7C 30 30 30 30 2C 31 2C 2C 31 45 0D 0A

Convert the data from HEX to ASCII, we got following ASCII data:

&&Q149,860262050015374,000,53,01392811,210602081053,A,22.677661,114.046271,5,2.1,0,327,100,63,460|0|249F|0AC26210,31,0000003D,02,02,04A2|017D|0000|0000,1,,1E

**Ibutton=**01392811

**Example of VT200L GPRS Protocol Integration（DLT）**

The card swiping event of the DLT version needs to be generated when ACC is ON. When the magnetic card is swiped, the 0x53 event is generated and the magnetic card data is reported.

Data Format：

&&<pack-no><pack-len>,<ID>,<cmd>,<alm-code>,<alm-data>,<date-time>,<fix\_flag>,<latitude>,<longitude>,<sat-quantity>,<HDOP>,<speed>,<course>,<altitude>,<odometer>,<MCC|MNC|LAC|CI>,<CSQ-quanlity>,<system-sta>,<in-sta>,<out-sta>,<ext-V|bat-V|ad1-V|…|adn-V>,<pro-code>,<fule\_liter>,<temp-sensor><checksum>\r\n

Note: Magnetic card data only exists in the data of 53 events, other events do not contain magnetic card data.

**53 event data example:**

&&F164,860262050009146,000,53,21 1 9999958 00100 ?,210325083017,A,22.678540,114.046106,16,1.1,0,0,72,374,460|0|249F|0099C257,28,000000BD,02,00,04CA|01A1|0000|0000,1,,B4

Original data from tracker to server:

26 26 46 31 36 34 2C 38 36 30 32 36 32 30 35 30 30 30 39 31 34 36 2C 30 30 30 2C 35 33 2C 32 31 20 20 31 20 20 39 39 39 39 39 35 38 20 20 30 30 31 30 30 20 20 3F 2C 32 31 30 33 32 35 30 38 33 30 31 37 2C 41 2C 32 32 2E 36 37 38 35 34 30 2C 31 31 34 2E 30 34 36 31 30 36 2C 31 36 2C 31 2E 31 2C 30 2C 30 2C 37 32 2C 33 37 34 2C 34 36 30 7C 30 7C 32 34 39 46 7C 30 30 39 39 43 32 35 37 2C 32 38 2C 30 30 30 30 30 30 42 44 2C 30 32 2C 30 30 2C 30 34 43 41 7C 30 31 41 31 7C 30 30 30 30 7C 30 30 30 30 2C 31 2C 2C 42 34 0D 0A

Track#3= 21 1 9999958 00100 ?

<driver's license type><gender><driver's license number><driver's license location>? "