AVT110 RS485 Tracker Protocol

EGPRS / LTE Cat-1

R1.02

Contents

0. Revision History

Revision	Date	Author	Description of Change
1.01	2025-08-30	Sherlock	Initial protocol base on RS232 V4.02 protocol.
1.02	2025-10-26	Sherlock	1. Added "3: Boot firmware of main MCU" in <*Update Type*> of **AT@UPF** command. 2. Added Bit9: BOOT Firmware of Main MCU Mask in <*BMR Data Mask*> of **AT@BMS** command. 3. Added <*BOOT Firmware Version*> in **+BMR** Message.

TERMS AND ABBREVIATIONS

TERMS

IN THIS DOCUMENT, UNLESS SPECIFIED, "TERMINAL", "DEVICE", "EQUIPMENT" ALL REFER TO THE TERMINAL DEVICE DESIGNED BY BTTECH.

ABBREVIATIONS

ACK	Acknowledgement
AGPS	Assisted Global Position System
APN	Access Point Name
ASCII	American National Standard Code for Information Interchange
BER	Bit Error Rate
BLE	Bluetooth Low Energy
CPU	Central Processing Unit
GPRS	General Packet Radio Service
GPS	Global Position System
GSM	Global System for Mobile Communication
HDOP	Horizontal Dilution of Precision
HTTP	Hyper Text Transfer Protocol
ICCID	Integrated Circuit Card Identity
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
IoT	Internet of Things
IP	Internet Protocol
LAC	Location Area Code
LTE	Long Term Evolution
MCC	Mobile Country Code
MCU	Micro Controller Unit
MNC	Mobile Network Code
MQTT	Message Queuing Telemetry Transport
NTP	Network Time Protocol
PLMN	Public Land Mobile Network
RSSI	Received Signal Strength Indicator
SIM	Subscriber Identity Module
SMS	Short Message Service
SN	Serial Number
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UTC	Coordinated Universal Time
WAN	Wide Area Network
VG	Virtual Ignition
VGN	Virtual Ignition On
VGF	Virtual Ignition Off

1. Data Stream Format

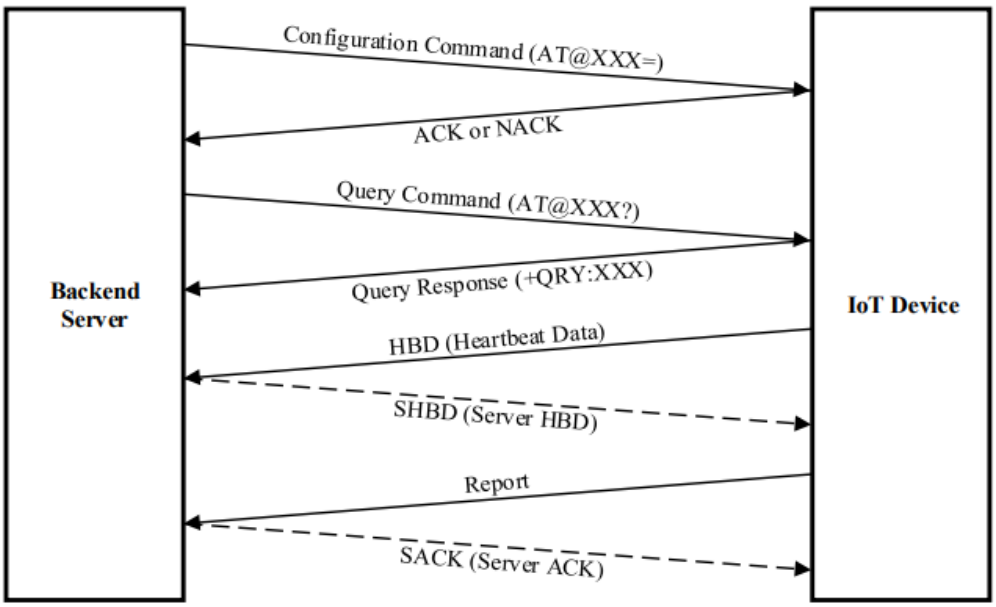
Data stream includes upstream and downstream.

Upstream refers to the data stream transmitted from the terminal device to the backend server via cellular network, which includes the response to the downlink command and some unsolicited messages.

Downstream refers to the data stream transmitted from the backend server to the terminal device via cellular network, which includes the commands to configure or operate the terminal and the response to the uplink message.

Upstream and downstream include the following types of frames:

Direction	Frame Encoding	Frame Type	Frame Usage
Downstream	Printable ASCII	Configuration Command	The backend server configures the parameters of the terminal.
		Query Command	The backend server queries the current parameters of the terminal.
		+SACK	Server Acknowledgement. The backend server send +SACK response to the terminal when it receives an +RST message from terminal.
		+SHBD	The backend server responds to the terminal **+SHBD** frame when it receives **+HBD** frame
Upstream	Printable ASCII	+QRY	The backend server sends a '**Query Command**' frame to the terminal, and the terminal responds with the current parameters.
		+HBD	Heartbeat Data. In order to maintain connectivity with the backend server, the terminal periodically sends a **+HBD** frame to the server.
		+ACK	Acknowledgement. When the terminal receives a **Configuration Command** from backend server/data cable/SMS/bluetooth, It responds with a **+ACK** frame indicating that a legitimate command has been received.
		+NACK	No Acknowledgement. When the backend server sends an illegal '**Configuration Command**' frame to the terminal, and the terminal responds with a **+NACK** frame indicating that an illegal command has been received.
	Hexadecimal Code	+RPT	A message that the terminal actively reports to the backend server.



The backend server issues commands to the terminal to not only configure and query its parameters, but also control it to perform certain actions. And the terminal sends heartbeat packets and unsolicited reports to the backend server appropriately.

All the upstream and downstream frames end with character "#" (i.e. 23H). In the frame of printable ASCII format, the character "," is used to separate the adjacent parameter characters. In the frame of the hexadecimal code format, there is no separator between the parameters. Please refer to the detailed descriptions of each frame format below.

2. Command Structure

2.1. Configuration Command

2.1.1. Network

2.1.1.1. SIS (Main Server Information Settings)

The command **AT@SIS** is used to configure main backend server parameters, such as IP, Port.

- **AT@SIS=**

Example: **AT@SIS=at,0,1,255.255.255.255,12345,,,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@SIS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Report Mode	1	0 - 3	0
4	Buffer Mode	1	0 - 2	1
5	Main Server IP / Domain Name	<=60		
6	Main Server Port	<=5	0 – 65535	
7	Reserved	0		
8	Reserved	0		
9	SACK Enable	1	0 - 1	0
10	Heartbeat Interval	<=3	0|2 – 360 (min)	0
11	PDP Enable	1	0 - 1	0
12	TLS Enable	1	0 - 1	0
13	TLS Verification Mode	1	0 - 2	0
14	Message Filter Mode	1	0 - 2	0
15	Message Filter Mask1	1	00000000 – FFFFFFFF	0
16	Resend Time No SACK	<=3	10 - 300	10
17	Serial Number	4	0000 – FFFF	
18	Tail	1	#	#

- <Report Mode>: This parameter defines the method of communication between the backend server and the device.

- 0: Stop mode.
- 1: TCP mode.
  The device will send data to the backend server by TCP protocol. Receiving protocol commands via TCP is supported if the EGPRS/LTE network allows it.
- 2: UDP mode.
  The device will send data to the backend server by UDP protocol. Receiving protocol commands via UDP is supported if the EGPRS/LTE network allows it.
- 3: MQTT mode.
  MQTT (Message Queuing Telemetry Transport) is a lightweight, publish subscribe based messaging protocol suitable for resource constrained devices and low bandwidth, high latency, or unstable network environments.

✧ <*Buffer Mode*>: The working mode of the buffer report function. If the buffer report function is enabled and the device goes into areas without EGPRS/LTE network coverage or without receiving +SACK message from server, the device will save all the reports locally. If the device goes back to areas with EGPRS/LTE network coverage, it will then send all the buffered reports through EGPRS/LTE.

- 0: Disable buffer function
- 1: Low priority - Enable the buffer report function: In this mode, the device will send the buffered messages after sending real time messages.
- 2: High priority - Enable the buffer report function: In this mode, the device will send the real time messages after sending buffered messages.

✧ <*Main Server IP / Domain Name*>: The IP address or the domain name of the primary server.

✧ <*Main Server Port*>: The port of the primary server.

✧ <*SACK Enable*>: This parameter defines whether the backend server should respond to the terminal with +SACK message when it receives a message from the terminal.

- 0: The backend server doesn't reply with +SACK message when receiving a message from the terminal.
- 1: The backend server replies with +SACK message when receiving a message from the terminal.

✧ <*Heartbeat Interval*>: The interval for sending heartbeat messages (**+HBD**) when report mode is TCP long-connection mode or UDP mode. If it is set to 0, no heartbeat message will be sent.

✧ <*PDP Enable*>: This parameter defines whether send "PDP Alarm Event" of **+RPT** message when the terminal socket connects to server.

- 0: Disable PDP alarm event report.
- 1: Enable PDP alarm event report.

✧ <*TLS Enable*>: The working mode of TLS.

- 0: Disable.
- 1: Enable.

✧ <*TLS Verification Mode*>: It expresses the certificate verification way for the terminal.

- 0: Don't verify the certificates, no certificates need to be built into the terminal.
- 1: Only server certification, CA file needs to be built into the terminal.
- 2: Server and client certifications, CA file, Client certificate, client key file needs to be built into the terminal.

✧ <*Message Filter Mode*>: it expresses the message filtering method to be used.

- 0: No need for filtering, all messages reporting.
- 1: Messages selected by <Filter Mask> will allow to reporting, other messages will not allow to reporting.
- 2: Messages selected by <Filter Mask> will not allow to reporting, other messages will allow to reporting.

✧ <*Message Filter Mask*>: this is bitmask to configure which message should be reported or filtered.

- Bit 0: Tachograph Common Event (25) event type of **+RPT** message.
- Bit 1: Tachograph DDD File Upload Event (26) event type of **+RPT** message.
- Bit 2: **+DFI** message.
- Bit 3: **+LDP** message.
- Bit 4: **+ACK: CAN** message.
- Bit 5: **+ACK: TAC** message.
- Bit 6: **+ACK: TCO** message.
- Bit 7: CANBUS Info Event (24) event type of **+RPT** message.

✧ <*Resend Time No SACK*>: It expresses the resend time of message, when the +SACK enable and unit sent message to server, the unit never received +SACK message from server, main server and second server all controlled by <*Resend Time No SACK*>.

2.1.1.2. SSI (Second Server Information Settings)

The command **AT@SSI** is used to configure second backend server parameters, such as IP, Port.

➢ **AT@SSI=**

Example: **AT@SIS=at,0,1,255.255.255.255,12345,,,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@SSI=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Report Mode	1	0 - 3	0
4	Buffer Mode	1	0 - 2	1
5	Second Server IP / Domain Name	<=60		
6	Second Server Port	<=5	0 – 65535	
7	Reserved	0		
8	Reserved	0		
9	SACK Enable	1	0 - 1	0
10	Heartbeat Interval	<=3	0|2 – 360 (min)	0
11	Reserved	0		

12	TLS Enable	1	0 - 1	0
13	TLS Verification Mode	1	0 - 2	0
14	Message Filter Mode	1	0 - 2	0
15	Message Filter Mask	<=8	00000000 – FFFFFFFF	0
16	Serial Number	4	0000 – FFFF	
17	Tail	1	#	#

- <Report Mode>: This parameter defines the method of communication between the backend server and the device.
  - 0: Stop mode.
  - 1: TCP mode.
    The device will send data to the backend server by TCP protocol. Receiving protocol commands via TCP is supported if the EGPRS/LTE network allows it.
  - 2: UDP mode.
    The device will send data to the backend server by UDP protocol. Receiving protocol commands via UDP is supported if the EGPRS/LTE network allows it.
  - 3: MQTT mode.
    MQTT (Message Queuing Telemetry Transport) is a lightweight, publish subscribe based messaging protocol suitable for resource constrained devices and low bandwidth, high latency, or unstable network environments.
- <*Buffer Mode*>: The working mode of the buffer report function. If the buffer report function is enabled and the device goes into areas without EGPRS/LTE network coverage or without receiving +SACK message from server, the device will save all the reports locally. If the device goes back to areas with EGPRS/LTE network coverage, it will then send all the buffered reports through EGPRS/LTE.
  - 0: Disable buffer function
  - 1: Low priority - Enable the buffer report function: In this mode, the device will send the buffered messages after sending real time messages.
  - 2: High priority - Enable the buffer report function: In this mode, the device will send the real time messages after sending buffered messages.
- <*Main Server IP / Domain Name*>: The IP address or the domain name of the primary server.
- <*Main Server Port*>: The port of the primary server.
- <*SACK Enable*>: This parameter defines whether the backend server should respond to the terminal with +SACK message when it receives a message from the terminal.
  - 0: The backend server doesn't reply with +SACK message when receiving a message from the terminal.
  - 1: The backend server replies with +SACK message when receiving a message from the terminal.
- <*Heartbeat Interval*>: The interval for sending heartbeat messages (**+HBD**) when report mode is TCP long-connection mode or UDP mode. If it is set to 0, no heartbeat message will be sent.
- <*PDP Enable*>: This parameter defines whether send "PDP Alarm Event" of **+RPT** message when the terminal socket connects to server.

- 0: Disable PDP alarm event report.
- 1: Enable PDP alarm event report.

✧ <*TLS Enable*>: The working mode of TLS.

- 0: Disable.
- 1: Enable.

✧ <*TLS Verification Mode*>: It expresses the certificate verification way for the terminal.

- 0: Don't verify the certificates, no certificates need to be built into the terminal.
- 1: Only server certification, CA file needs to be built into the terminal.
- 2: Server and client certifications, CA file, Client certificate, client key file needs to be built into the terminal.

✧ <*Message Filter Mode*>: it expresses the message filtering method to be used.

- 0: No need for filtering, all messages reporting.
- 1: Messages selected by <Filter Mask> will allow to reporting, other messages will not allow to reporting.
- 2: Messages selected by <Filter Mask> will not allow to reporting, other messages will allow to reporting.

✧ <*Message Filter Mask*>: this is bitmask to configure which message should be reported or filtered.

- Bit 0: Tachograph Common Event (25) event type of **+RPT** message.
- Bit 1: Tachograph DDD File Upload Event (26) event type of **+RPT** message.
- Bit 2: **+DFI** message.
- Bit 3: **+LDP** message.
- Bit 4: **+ACK: CAN** message.
- Bit 5: **+ACK: TAC** message.
- Bit 6: **+ACK: TCO** message.
- Bit 7: CANBUS Info Event (24) event type of **+RPT** message.

2.1.1.3. NIS (Network Information Setting)

The command **AT@NIS** is used to configure network information parameter.

➢ **AT@NIS=**

Example: **AT@NIS=at,apn,username,123456,,,,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@NIS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	APN	<=42		
4	APN User Name	<=32	NULL	
5	APN Password	<=32	NULL	
6	Reserved	0		

7	Reserved	0		
8	Reserved	0		
9	Network Scan Mode	1		0
10	Serial Number	4	0000 – FFFF	
11	Tail	1	#	#

- ✧ <*APN*>: Access point name (APN).
- ✧ <*APN User Name*>: The GPRS APN user name. If the parameter field is empty, the current value for this parameter will be cleared.
- ✧ <*APN Password*>: The GPRS APN password. If the parameter field is empty, the current value for this parameter will be cleared.
- ✧ <*Network Scan Mode*>: Configure RAT(s) to be Searched.
  - 0: Automatic (Cat1->GSM).
  - 1: Cat1 only.
  - 2: GSM only.

2.1.1.4. MQT (MQTT Information Setting)

The command **AT@MQT** is used to configure MQTT information parameter.

➢ **AT@MQT=**

Example: **AT@MQT=at,apn,username,123456,,,,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@MQT=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Server Index	1	0 - 1	0
4	Reserved			
5	Client ID	<=64	'0' -'9', 'a' -'z', 'A' -'Z' |'$'	$
6	User Name	<=64	ASCII String	
7	Password	<=64	ASCII String	
8	Subscribe Topic	<=64	ASCII String	
9	Subscribe QOS	1	0 - 2	
10	Publish Topic	<=64	ASCII String	
11	Publish QOS	1	0 - 2	
12	Keep Alive	<=4	0 - 3600	120
13	Serial Number	4	0000 – FFFF	

14	Tail	1	#	#

✧ <*Server Index*>: It indicates support MQTT server map index.
- 0: Main Server.
- 1: Second Server.

✧ <*Client ID*>: Each client connected to the server has a unique client identifier (Client ID). Both the client and the server must use the Client ID to identify the state associated with the MQTT session between them. The Client ID can contain only uppercase letters, lowercase letters and numeric characters.
In addition, a single character '$' is defined to indicate the use of the IMEI number that will be used internally as the <*Client ID*>.

✧ <*User Name*>: Client username, used for MQTT server authorization.

✧ <*Password*>: Client password, used for MQTT server authorization.

✧ <*Subscribe Topic*>: The client subscribes topic name.

✧ <*Subscribe QOS*>: the quality of service of subscribe topic.

✧ <*Publish Topic*>: The client publishes topic name.

✧ <*Publish QOS*>: the quality of service of publish topic.

✧ <*keep Alive*>: It is used to set keep alive mechanism for MQTT, Range: 0 - 3600; Default value: 120; Unit: seconds. This parameter defines the maximum interval time for receiving messages from the client. Within 1.5 times the set time, if the server does not receive a message from the client, it is assumed that the client has sent a DISCONNECT message, and therefore the server will disconnect the client connection.

2.1.1.5. FTP (FTP Server Setting)

The command **AT@FTP** is used to configure the FTP server information as to where and how to upload data packets.

➢ **AT@FTP=**

Example: **AT@FTP=at,,,,,,,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@FTP=
2	Password	4 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Server Address	<=60		
4	Server Port	<=5	0 – 65535	0
5	Full Name	<=60		
6	User Name	<=20		
7	Login Password	<=20		
8	Transfer Mode	1	0|1	0
9	Prefix File Name	<=20	'0' – '9' 'a' – 'z' 'A' – 'Z'	%IMEI

10	Serial Number	4	0000 – FFFF	
11	Tail	1	#	#

- ✧ <*Server Address*>: The IP address or the domain name of the FTP server.
- ✧ <*Server Port*>: The server port of the FTP server.
- ✧ <*Full Name*>: The file path to save pictures. To restore the default path after a path is set, set it to "\".
- ✧ <*User Name*>: The user name to login to the FTP server.
- ✧ <*Login Password*>: The password to login to the FTP server.
- ✧ <*Transfer Mode*>: The mode of transferring picture.
  - 0: Passive mode.
  - 1: Active mode.
- ✧ <*Prefix File Name*>: The file name prefix of transformed file. It should not include ".".

2.1.1.6. TCR (TLS Certificate Remote Download)

The command **AT@TCR** is used to configure TLS certificate remote download parameter.

- ➢ **AT@TCR=**

Example: **AT@TCR=at,0,0,abcd,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@TCR=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Server Protocol Type	1	0	0
4	Certificate File Type	1	0 - 2	
5	Download URL	<=80	legal URL Address	
6	Serial Number	4	0000 – FFFF	
7	Tail	1	#	#

- ✧ <*Server Protocol Type*>: The type of communication protocol using for obtain data from the server.
  - 0: HTTP.
- ✧ <*Certificate File Type*>: It expresses the type of certificate file to download from the server.
  - 0: CA certificate.
  - 1: Client certificate.
  - 2: Client Key.
- ✧ <*Download URL*>: The URL to download the certificate.

2.1.1.7. TCL (TLS Certificate Local Download)

The command **AT@TCL** is used to configure TLS certificate local download parameter.

➢ **AT@TCL=**

Example: **AT@TCL=1,0,13,2,ABCDEFGH,1234,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@TCL=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0 | 1	
4	Certificate File Type	1	0 - 2	
5	Total Package	<=3	1 - 255	
6	Current Package ID	<=3	1 - 255	
7	Certificate Data Length	<=3	1 - 768	
8	Certificate Data	<=768	HEX	
9	CRC16	4	0000 – FFFF	
10	Serial Number	4	0000 – FFFF	
11	Tail	1	#	#

✧ <*Mode*>: The working mode of local certificate file.
- 0: Delete file.
- 1: Write file.

✧ <*Certificate File Type*>: It expresses the type of certificate file to download from local
- 0: CA certificate.
- 1: Client certificate.
- 2: Client Key.

✧ <*Total Package*>: Need total package of write certificate data.

✧ <*Current Package ID*>: the current package ID of write certificate data.

✧ <*Certificate Data Length*>: the length of certificate data written to file.

✧ <*Certificate Data*>: certificate data written to file.

✧ <*CRC16*>: CRC verification current transfer data whether is correct.

2.1.2.General

2.1.2.1. GMT (Greenwich Mean Time Setting)

The command **AT@GMT** is used to configure time zone.

➢ **AT@GMT=**

Example: **AT@GMT=at,0,0,0001#**

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	7	ASCII	AT@GMT=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Time Zone Direction	1	0|1	
4	Time Zone	<=2	0 - 12	
5	Serial Number	4	0000 – FFFF	
6	Tail	1	#	#

✧ <*Time Zone Direction*>: *This indicates east time zone or the west time zone.*

- 0: East time zone.
- 1: West time zone.

✧ <*Time Zone*>: UTC offset in hours.

2.1.2.2. GCS (Global Configuration Setting)

The command **AT@GCS** is used to configure device parameter,

➢ **AT@GCS=**

Example: **AT@GCS=at,19C,,0,,,,FF,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@GCS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	New Password	2 - 19	'0' - '9', 'a' - 'z', 'A' - 'Z'	
4	Data Mask	<=8	(HEX)	
5	Event Data Mask	<=8	(HEX)	
6	LED Mode	1	0|1	0
7	Low Battery Threshold	<=2	5 – 30 (%)	10
8	Low Main Power Threshold	<=5	8000 – 24000 (mV)	9000
9	GNSS Working Mode	1	1|3|4	1
10	GNSS Info Mask	<=4	0000 - FFFF	FF
11	AGPS Mode	1	0|1	0
12	ODO Enable	1	0|1	0
13	ODO Initial Mileage	<=9	0.0 – 4294967.0 (Km)	0.0
14	Hour Meter Enable	1	0|1	0

15	Initial Hour Meter Count	11	00000:00:00 - 99999:59:59	00000:00:00
16	Power Saving Mode	1	0 – 2	0
16	Serial Number	4	0000 – FFFF	
17	Tail Character	1	#	#

✧ <*New Password*>: It is used to set up a new password.

✧ <*Data Mask*>: Component mask of the **+RPT/-RPT** message.

Mask Bit	Item
Bit 31	Reserved
…	…
Bit 21	RS485 Information Mask
Bit 20	Reserved
Bit 19	BLE Information Mask
Bit 18	NMEA2000 information Mask
Bit 17	Special Car Information Mask
Bit 16	Tachograph Information Mask
Bit 15	Tachograph Common Event Information Mask
Bit 14	Tachograph Driver2 Working Time Mask
Bit 13	Tachograph Driver1 Working Time Mask
Bit 12	Reserved
Bit 11	Electric CAN Information Mask
Bit 10	Motor CAN Information Mask2
Bit 9	Motor CAN Information Mask1
Bit 8	Event Data Mask
Bit 7	Upgrade information Mask
Bit 6	Reserved
Bit 5	Reserved
Bit 4	Cell Information Mask
Bit 3	GNSS Information Mask
Bit 2	Battery Information Mask
Bit 1	Network Type

Bit 0	Multi-Packet, this bit doesn't support configuration, when the length of message more than 256 Bytes, this bit will be enabled.

✧ <*Event Data Mask*>: Component sub mask of the **<***Data Mask*>.

Mask Bit	Item
Bit 31	Reserved
…	…
Bit 14	TLS Certificate Download Mask
Bit 13	Reserved
Bit 12	Crash Info Mask, this bit doesn't support configuration, when happen crash event, this bit will be enabled.
Bit 11	3Axis Self Calibration info Mask, this bit doesn't support configuration, when self-calibration success, this bit will be enabled.
Bit 10	Hour Meter Count Mask
Bit 9	1-wire Sensor Mask
Bit 8	ID Data Mask
Bit 7	GEO Status Mask
Bit 6	Reserved
Bit 5	Mileage Info Mask
Bit 4	Digital output Mask
Bit 3	Digital Input Mask
Bit 2	Ignition & Motion State Mask
Bit 1	Analog Input Mask
Bit 0	Main Power Mask

✧ <*LED Mode*>: This parameter is used to configures the LEDs' working mode:

- 0: All LEDs does not work after 6 minutes.
- 1: All LEDs always work.

✧ <*Low Battery Threshold*>: when battery voltage is lower than <*Low Battery Threshold*>, the device will send the low battery voltage alarm event to backend server.

✧ <*Low Main Power Threshold*>: when main power voltage is lower than <*Low Main Power Threshold*>, the device will send the low main power alarm event to backend server.

<*GNSS Working Mode*>: This parameter configures the GNSS working mode

- 1: GPS Only.
- 3: GPS, GLONASS and GALILEO.
- 4: GPS, BEIDOU and GALILEO.

✧ <*GNSS Info Mask*>: This parameter configures the GNSS information mask of **+RPT** message.

Mask Bit	Item
Bit 15	Reserved
…	…
Bit 8	GNSS Satellites Number
Bit 7	UTC Time
Bit 6	Longitude
Bit 5	Latitude
Bit 4	Altitude
Bit 3	Azimuth
Bit 2	Speed
Bit 1	HDOP
Bit 0	Fix Type

✧ <*AGPS Mode*>: AGPS helps increase the possibility to fix GNSS position and reduces the time to fix GNSS position. When AGPS is enabled, it will consume some data.

- 0: Disable the AGPS function.
- 1: Enable the AGPS function.

✧ <*ODO Enable*>: Enable/disable the ODO-graph function to calculate the total mileage.

- 0: Disable the odometer. The mileage will not be included in the position related messages.
- 1: Enable the ODO-graph function to calculate the total mileage by GNSS. The current mileage will be included in the position related message.

✧ <*ODO Initial Mileage*>: The initial value for calculating the total mileage.

✧ <*Hour Meter Enable*>: Enable/disable the hour meter count function. When the hour meter count function is enabled, the hour meter count will be calculated when the device detects the vehicle ignition is on.

- 0: Disable the HMC function.
- 1: Enable the HMC function.

✧ <*Initial Hour Meter Count*>: The initial value of <*Hour Meter Count*> which range is 00000:00:00 – 99999:59:59. It is consisted of three parts that separated by ':', the first part is hour digits and the length of it is 5 bytes, the second part is 2 bytes minute digit, the last part is 2 bytes second digit. When the ignition is on, <*Hour Meter Count*> will be reported in regular reports and ignition events and calculated based on this value.

✧ <*Power Saving Mode*>: It is used to set up the different working mode to save the power consumption.

- 0: Normal Power Saving Mode: When the device detects ignition off and <*IGF Send Interval*> is less than 10 minutes, GPS chip will be always on. When <*IGF Send Interval*> is 10 minutes or more, the GPS chip will be on only when it is needed. The cellular module will be always on.

- 1: Low Power Saving Mode (G-sensor on): When the device detects ignition off and <*IGF Send Interval*> is less than 10 minutes, GPS module and cellular module will be always on. When <*IGF Send Interval*> is 10 minutes or more, the GPS chip will be on only when it is needed. The cellular module will be on when there is any message need to send to backend server, the unit can wake up by virtual ignition and wire ignition.
- 2: Deep Power Saving Mode (G-sensor off): When the device detects ignition off and <*IGF Send Interval*> is less than 10 minutes, GPS module and cellular module will be always on. When <*IGF Send Interval*> is 10 minutes or more, the GPS chip will be on only when it is needed. The cellular module will be on when there is any message need to send to backend server, the unit can wake up only by wire ignition.

2.1.2.3. RRS (Regular Report Setting)

The command **AT@RRS** is used to configure regular GNSS report parameters. It can be set to report multiple GNSS in one message when ignition is on. However, it will only report one GNSS in each message when ignition is off,

➢ **AT@RRS=**

Example: **AT@RRS=at,1,0,30,3600,1000,1000,30,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@RRS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0|1	0
4	IGN GNSS Fixing Interval	<=4	0 – 3600 (sec)	0
5	IGN Send Interval	<=4	1 – 3600 (sec)	30
6	IGF Send Interval	<=5	0 – 86400 (sec)	3600
7	Discard No Fix	<=1	0|1	1
8	Reserved	0		
9	Corner Angle	<=3	0|5– 90	30
10	Serial Number	4	0000 – FFFF	
11	Tail Character	1	#	#

✧ <*Mode*>: The working mode of the regular GNSS report function.

- 0: Disable this function.
- 1: Fixed timing GNSS report. The report message is sent to the backend server periodically according to the <*Send Interval*> Parameter.

✧ <*IGN GNSS Fixing Interval*>: It is GNSS fixing internal when ignition is on. Its range is 0 – 3600s. If it is set to be 0, the device will update GNSS position according to the value of <*IGN Send Interval*>. Please notice that <*IGN GNSS fixing Interval*> can not be more than < *IGN Send*

Interval>.

- ✧ <*IGN Send Interval*>: It is GNSS report internal when ignition is on. The value range is 1 – 3600s. It can be maximum 15 times higher than <IGN GNSS Fixing Interval>. When it is times of the <IGN GNSS Fixing Interval>, device will send the GNSS report with multiple GNSS positions.
- ✧ <*IGF Report Interval*>: The time interval for fixing and sending the position information when ignition is off. Its value range is 0s – 86400s.
- ✧ <*Turning Angle*>: It is the parameter to set up the turning angle GNSS report. 0 means "Disable the turning of angle". If the parameter is set to be higher than 0, the device will keep comparing the heading between the two latest GNSS fixing. If the difference is same or more than the set value, It will generate a GNSS of the turning possible and sent it within the regular GNSS report.

  <*Discard No Fix*>: It is used to set up the device whether the GNSS report need to be sent. when GNSS fixing is failed.
  - 0: Enable report.
  - 1: Disable report.

2.1.2.4. RTO (Real Time Operation)

The command **AT@RTO** is used to retrieve information from the device or control the device, when it executes certain actions.

- ➢ **AT@RTO=**

Example: **AT@RTO=at,1,,,,,FFFF#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@RTO=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Sub Command	<=1	1 - 16	
4	Reserved (Optional)	0		
5	Reserved (Optional)	0		
6	Serial Number	4	0000 – FFFF	
7	Tail	1	#	#

- ✧ <Sub Command>: A hex value to indicate the sub command to be executed.
  - 1: **Read Parameter**. Get the all configurations of the device via the message **+ALL**.
  - 2: **Reboot The Device**.
  - 3: **Power Off The Device.**
  - 4: **Reset All Parameters to factory settings and clear all buffered messages.**
  - 5: **Query Network Information**. Get APN, ICCID, RSSI, Cell-ID and IP via the message **+QNI**.
  - 6: **Query Version**. Get the version information of the device via the message **+VER**.
  - 7: **Reserved.**
  - 8: **Request All Messages.** It is used to request the device to generate all types of

messages and send to backend server for integration.

- 9: **Delete Buffer Message**, it will delete current all buffer messages.
- 10: **Reserved**.
- 11: **Query Satellite Information**. When the device receives the command **AT@RTO** to get the GPS, GLONASS and BEIDOU satellites information, it will send the satellite information via the message **+GSV (GPS satellites information), +LSV (GLONASS satellites information), +BSV (BEIDOU satellites information), +GAV (GALILEO satellites information)** to the backend server.
- 12: **CAN**. Get CAN information via the CAN message of **+RPT**.
- 13: **CVS**. Get CAN version and serial number via the message **+CVS**.
- 14: **CAN Module Operation**. Set car model for CAN Module or read car model from CAN Module.

Parameter	Length (Byte)	Range/Format	Default
CAN Module Operation Mode	1	0 – 3	
CAN Module Car Model ID	1 – 5	1 – 65535	

- 16: **DFI**. Get device full information via the message **+DFI**.

✧ <*CAN Module Operation Mode*>: If <*Sub Command*> is set to 14, this parameter will work as follows.

- 0: Read the current car model and report it via the message **+CMI**.
- 1: Set car model. Please use the <*CAN Module Car Model ID*> parameter to set car model.
- 2: Start CAN Module automatic synchronization. The synchronization result is CANBUS sync state of Event Type 24 (CANBUS Info Event) in **+RPT** message.
  Note: The entire synchronization takes about 10-40s, and CAN Module will restart immediately after the end of the synchronization regardless of the result. If automatic sync is enabled, please wait for the synchronization to finish before reading the current car model. If automatic synchronization has not ended, the subsequent synchronization command will be ignored.
- 3: Delate current supported car model.

2.1.2.5. PCL (Preset Command List)

The command **AT@PCL** is used to preset store command list, which is bind with trigger event id in **AT@ECL** command.

➢ **AT@PCL=**

Example: **AT@PCL=at,1,,,,,FFFF#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@PCL=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at

3	Preset Command ID	<=1	0 - 31	
4	Command String	<=200	AT Command | NULL	
5	Serial Number	4	0000 – FFFF	
6	Tail	1	#	#

✧ <*Preset Command ID*>: A numeral to identify the preset command.

✧ <*Command String*>: The whole content of the preset command.

2.1.2.6. ECL (Event Driver Command List)

The command **AT@ECL** is used to bind input events and preset commands, the input events will trigger the corresponding preset commands.

➢ **AT@ECL=**

Example: **AT@ECL=at,1,,,,,FFFF#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@ECL=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Group Number	<=2	0 - 15	0
4	Group Mode	1	0 - 1	0
5	Reserved	0		
6	Event ID Mask	<=8	0 - FFFFFFFF	0
7	Bind Command Mask	<=8	0 - FFFFFFFF	0
8	Command ACK	0	0 - 1	0
9	Serial Number	4	0000 – FFFF	
10	Tail	1	#	#

✧ <*Group Number*>: The group numbers to identify the input events and preset commands to be executed.

✧ <*Group Mode*>: The working mode of the event driver command group.

- 0: Disable the group function.
- 1: Enable the group function.

✧ <*Event Duration Time*>: The duration time for input events before the specified preset commands are executed.

✧ <*Event ID Mask*>: The bitwise mask to indicate the trigger events included in the group.

- Bit 0: Ignition on.
- Bit 1: Ignition off.
- Bit 2: No moving.
- Bit 3: Moving.

- Bit 4: Connect main power.
- Bit 5: Disconnect main power.
- Bit 6: Battery charging.
- Bit 7: Battery no charging.
- Bit 8: Digital input 1 off.
- Bit 9: Digital input 1 on.
- Bit 10: Digital output 1 off.
- Bit 11: Digital output 1 on.
- Bit 12: DRV ID no authorized.
- Bit 13: DRV ID authorized.

✧ <*Bind Command Mask*>: The bitwise mask of the preset commands which will be executed after the event state of the group becomes true.

✧ <*Command ACK*>: A numeral to indicate whether to return **+ACK** message after preset command is executed.

- 0: Don't send **+ACK** message when the preset command is executed.
- 1: Send **+ACK** message when the preset command is executed.

2.1.3. Alarm

2.1.3.1. MOV (Movement Alarm Setting)

The **AT@MOV** command is used to configure the movement alarm parameters.

➢ **AT@MOV=**

Example: **AT@MOV=at,3,1,30,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@MOV=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Motion Threshold	1	0 - 9	2
4	Motion Duration	<=2	1 – 10 (sec)	1
5	Stillness Duration	<=2	1 – 60 (sec)	30
6	Serial Number	4	0000 – FFFF	
7	Tail	1	#	#

✧ <*Motion Threshold*>: It indicates the threshold of motion detection. Its value range is 0-9.

- 0: Disable motion detection.
- 1: 0.03g (Highest sensitivity)
- 2: 0.06g
- 3: 0.09g
- 4: 0.12g
- 5: 0.15g
- 6: 0.18g

- 7: 0.21g
- 8: 0.24g
- 9: 0.27g (Lowest sensitivity)

✧ <*Motion Duration*>: This is the duration time of motion detection. If motion working mode is enabled, and the motion is above the sensitivity for <*Motion Duration*> seconds, the device considered to be in moving status.

✧ <*Stillness Duration*>: This is the duration time of stillness detection. If motion working mode is enabled, and the stillness is above the sensitivity for <Stillness Duration> seconds, the device considered to be in stillness status.

2.1.3.2. TOW (TOW Alarm Setting)

The **AT@TOW** command is used to configure the moving alarm parameters.

➢ **AT@TOW=**

Example: **AT@TOW=at,0,120,60,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@TOW=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0|1	0
4	Tow Duration	<=4	120 – 3600 (sec)	120
5	Tow Alarm Interval	<=4	60 – 3600 (sec)	60
6	Serial Number	4	0000 – FFFF	
7	Tail	1	#	#

✧ <*Mode*>: The working mode of tow alarm function.

- 0: Disable the tow alarm.
- 1: Enable the tow alarm.

✧ <*Tow Duration*>: This parameter is used to measure whether the vehicle is towed . when the device detects consistent movement for this period time when ignition is off, it will report tow alarm to backend server.

✧ <*Tow Alarm Interval*>: The sending interval of the tow alarm. If <*Tow Alarm Interval*> is less than 60s, the GNSS will be always on.

2.1.3.3. SPD (Speed Alarm Setting)

This command is used to set the over speed alarm.

AT@SPD=

Example: **AT@SPD=at,0,100,10,60,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**

1	Head	7	ASCII	AT@SPD=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0|1	0
4	Over Speed Threshold	<=3	30 – 400 (km/h)	100
5	Over Speed Duration	<=4	5 – 3600 (sec)	10
6	Report Interval	<=4	30 – 3600 (sec)	60
7	Serial Number	4	0000 – FFFF	
8	Tail	1	#	#

✧ <*Mode*>: The working mode of the speed alarm function.
- 0: Disable speed alarm.
- 1: Enable speed alarm.

✧ <*Over Speed Threshold*>: it's the over speed threshold.

✧ <*Over Speed Duration*>: When the device detect the speed to higher than the <Over Speed Threshold> consistently for more than this duration time, device will report over speed alarm to backend server.

✧ <*Report Interval*>: The send interval of the speed alarm message.

2.1.3.4. GEO (Geo-Fence Setting)

The command **AT@GEO** is used to configure the parameters of circular Geo-Fence. (Circular Geo-Fence is a virtual perimeter around a geographic area using a location-based service. When the geo fencing terminal enters or exits the area, a notification will be generated. The notification contains information about the location of the terminal and could be sent to the backend server.)

➢ **AT@GEO=**

Example: **AT@GEO=at,1,3,121.123333,32.123456,50,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@GEO=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	GEO ID	<=2	1 – 32	
4	Mode	1	0 – 1	0
5	Longitude	<=11	(–)xxx.xxxxxx	
6	Latitude	<=10	(–)xx.xxxxxx	
7	Radius	<=7	50 – 60000 (m)	50
8	Check Interval	<=5	0|5 – 3600 (sec)	0
9	Serial Number	4	0000 – FFFF	

10	Tail	1	#	#

- ✧ <*GEO ID*>: The ID of the circular Geo-Fence. It supports total 32 GEO fences.
- ✧ <*Mode*>: The working mode of the zone's circular Geo-Fence function.
  - 0: Disable Geo-Fence function.
  - 1: Enable Geo-Fence function.
- ✧ <*Longitude*>: The longitude of a point which is defined as the center of the circular Geo-Fence region. The unit is degree, and accuracy is 6 decimal places. West longitude is defined as negative starting with the sign "–" and east longitude is defined as positive without "+". The range is -180.xxxxxx ~ 180.xxxxxx.
- ✧ <*Latitude*>: The latitude of a point which is defined as the center of the circular Geo-Fence region. The unit is degree, and accuracy is 6 decimal places. South latitude is defined as negative starting with the minus sign "–" and north latitude is defined as positive without "+". The range is -90.xxxxxx ~ 90.xxxxxx.
- ✧ <*Radius*>: The radius of the circular Geo-Fence region. The value range is 50 – 6000000. Unit: meter.
- ✧ <*Check Interval*>: The interval of GNSS checking position information against the Geo-Fence alarm. It works for all of the GEO fences.

2.1.3.5. HBS (Harsh Behavior Setting)

The command **AT@HBS** is used to monitor the harsh driving behavior based on motion sensor.

- ➢ **AT@HBS=**

Example: **AT@HBS=at,1,3,121.123333,32.123456,50,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@HBS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0 - 1	0
4	Acceleration Threshold	<=3	10 - 100	20
5	Acceleration Duration	<=3	10 – 250 (*8ms)	25
6	Braking Threshold	2	10 - 100	25
7	Braking Duration	<=3	10 - 250(*8ms)	25
8	Cornering Threshold	2	10 - 100	20
9	Cornering Duration	<=3	10 - 250 (*8ms)	25
10	Serial Number	4	0000 – FFFF	

11	Tail	1	#	#

✧ <*Mode*>: The working mode of the harsh behavior monitoring function.
- 0: Disable this function.
- 1: Enable this function.

✧ <*Acceleration Threshold*>: The threshold for the 3-axial acceleration data measure whether the device is in harsh acceleration status.

✧ <*Acceleration Duration*>: The time parameter to measure whether the device enters harsh acceleration status. If the driving behavior is maintained for a period of time longer than the time defined by <*Acceleration Duration*>, harsh acceleration event will be triggered.

✧ <*Brake Threshold*>: The threshold for the 3-axial acceleration data measure whether the device is in harsh braking status.

✧ <*Brake Duration*>: The time parameter to measure whether the device enters harsh braking status. If the driving behavior is maintained for a period of time longer than <*Brake Duration*>, harsh braking event will be triggered.

✧ <*Cornering Threshold*>: The threshold for the 3-axial acceleration data measure whether the device is in harsh cornering.

✧ <*Cornering Duration*>: The time parameter to measure whether the device enters harsh cornering status. If the driving behaviors are maintained for a period of time longer than the time defined by <*Cornering Duration*>, harsh cornering event will be triggered.

2.1.3.6. CRA (Crash Detection Setting)

The command **AT@CRA** is used to configure crash detection parameters.

➢ **AT@CRA=**

Example: **AT@CRA=at,1,4.0,,,,,,,,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@CRA=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0 – 1	0
4	X Axis Threshold	<=2	0 – 160 (*0.1g)	50
5	Y Axis Threshold	<=2	0 – 160 (*0.1g)	50
6	Z Axis Threshold	<=2	0 – 160 (*0.1g)	50
7	Serial Number	4	0000 – FFFF	
8	Tail Character	1	#	#

✧ <*Mode*>: The working mode of the crash detection function.
- 0: Disable this function.
- 1: Enable this function, it only works when Ignition is on.

✧ <*X Axis Threshold*>: The acceleration threshold for crash in X direction. The smaller value it is,

the more sensitive to detect crash event. If <*X Axis Threshold*> is 0, the device will not monitor crash event in X axis. The unit is 0.1g.

- ✧ <*Y Axis Threshold*>: The acceleration threshold for crash in Y direction. The smaller value it is, the more sensitive to detect crash event. If <*Y Axis Threshold*> is 0, the device will not monitor crash event in Y axis. The unit is 0.1g.
- ✧ <*Z Axis Threshold*>: The acceleration threshold for crash in Z direction. The smaller value it is, the more sensitive to detect crash event. If <*Z Axis Threshold*> is 0, the device will not monitor crash event in Z axis. The unit is 0.1g.

2.1.3.7. JAM (Jamming Alarm Setting)

The command **AT@JAM** is used to configure jamming detection and alarm parameters. Jamming alarm function need work with normal power saving mode of <*Power Saving Mode*> in **AT@GCS** command, because other power saving mode, the network module is off when the ignition off state and effect jamming detection function.

- ➢ **AT@JAM=**

Example: **AT@JAM=at,1,5,5,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@JAM=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0 – 1	0
4	Enter Jamming Duration	<=3	0 – 60 (*Sec)	5
5	Quit Jamming Duration	<=2	0 – 60 (*Sec)	5
6	Serial Number	4	0000 – FFFF	
7	Tail Character	1	#	#

- ✧ <*Mode*>: The working mode of the Jamming detection function.
  - 0: Disable this function.
  - 1: Enable this function.
- ✧ <*Enter Jamming Duration*>: The time parameter to measure whether the device enters jamming alarm status. It's maintained for a period time longer than the time defined by <*Enter Jamming Duration*>, jamming on alarm event will be triggered.
- ✧ <*Quit Jamming Duration*>: The time parameter to measure whether the device quits jamming alarm status. It's maintained for a period time longer than the time defined by <*Quit Jamming Duration*>, jamming off alarm event will be triggered.

2.1.4. IO & ADC

2.1.4.1. IGN (Ignition Setting)

The command **AT@IGN** is used to configure the way of ignition detection.

➢ **AT@IGN=**

Example: **AT@IGN=at,1,5,60,,,,,,,,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@IGN=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Wire Ignition Mode	1	0|1	1
4	IGN Debounce Time	<=2	0 – 20 (*10ms)	5
5	Reserved			
6	Reserved			
7	Virtual Ignition Mode	1	0 – 2	1
8	VGF Rest Duration	<=3	5 - 180(sec)	60
9	VGN Motion Duration	<=3	5 – 180(sec)	5
10	Ignition On Voltage	<=5	250 – 28000 mV	13500
11	Voltage Offset	<=4	200 – 2000 mV	600
12	Ignition On Debounce	<=3	5 – 255 (× 1 sec)	10
13	Ignition Off Debounce	<=3	5 – 255 (× 1 sec)	5
14	Serial Number	4	0000 – FFFF	
15	Tail Character	1	#	#

✧ <*Wire Ignition Mode*>: The working mode of wire ignition.

- 0: Disable the wire ignition function.
- 1: Enable the wire ignition function.

✧ <*IGN Debounce Time*>: The time for interruptible ignition port debouncing.

✧ <*Virtual Ignition Mode*>: A numeral to define the working mode of detecting virtual ignition state.

- 0: Disable the virtual ignition detection function
- 1: Enable the virtual ignition detection function, detection by G-sensor and vehicle voltage combo.
- 2: Enable the virtual ignition detection function, detection by G-sensor only.

✧ <*VGF Rest Duration*>: A time parameter to determine whether the device enters virtual ignition off state. The device will be considered in virtual ignition off state after the motion sensor detects rest and the stationary state is maintained for a period of time specified by the

parameter <*VGF Rest Duration*>.

- ✧ <*VGN Motion Validity*>: A time parameter to determine whether the device enters virtual ignition on state. The device will be considered in virtual ignition on state after the motion sensor detects movement and the moving state is maintained for a period of time specified by the parameter <*VGN Motion Validity*>.
- ✧ <*Ignition On Voltage*>: The external power voltage in ignition on state. Different vehicles have different voltage in ignition on state. This parameter should be set very close to the original external power, so that the device can detect ignition event more accurately.
- ✧ <*Voltage Offset*>: The offset from <*Ignition On Voltage*> used to determine ignition on or ignition off state. If the voltage of the external power is higher than <*Ignition On Voltage*> - <*Voltage Offset*> and is maintained for <*Ignition On Debounce*> seconds, the device will consider it as virtual ignition on state. If the voltage of the external power is lower than <*Ignition On Voltage*> - <*Voltage Offset*> and is maintained for <*Ignition Off Debounce*> seconds, the device will consider it as virtual ignition off state.
- ✧ <*Ignition On Debounce*>: The debounce time before updating virtual ignition on state according to the external power voltage. Unit: second.

Note: The device may adjust the <*Ignition On Voltage*> and <*Voltage Offset*> value automatically by measuring the external power voltage data to make the ignition detection more precisely.

- ✧ <*Ignition Off Debounce*>: The debounce time of the virtual ignition off voltage.

2.1.4.2. DAM (Digital and Analog Input Multiplexing Setting)

The command **AT@DAM** is used to configure the input to work as analog input or digital input.

- ➢ **AT@DAM=**

Example: **AT@DAM=at,1,0,2,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@DAM=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Reserved	1		
4	Input Mode	1	0 – 2	0
5	Debounce Time	<=2	0 – 20 (*10ms)	2
6	Validity Time	<=2	0|1 – 12 (*2s)	0
7	Delta AIS Threshold	<=4	0 – 10 (*100mV)	0
8	Delta AIS Duration	<=2	0 – 12 (*1s)	0
9	Serial Number	4	0000 – FFFF	
10	Tail Character	1	#	#

- ✧ <*Input Mode*>: The working mode of multiplexing input port.

- 0: Disable input mode.
- 1: Digital input mode (negative trigger).
- 2: Analog input mode.

✧ <*Debounce Time*>: The time for interruptible input port debouncing.

✧ <*Validity Time*>: The validity time of the input port. 0 means "Do not check the validity time".

Note: <Debounce Time> and <Validity Time> parameters using for the digital input of <*Input Mode*>.

✧ <*Delta AIS Threshold*>: The threshold for the delta analog input. 0 means "Do not check delta AIS threshold".

✧ <*Delta AIS Duration*>: The duration for the delta analog input. 0 means "Do not check delta AIS duration", if the unit meet <*Delta AIS Threshold*>, the unit will trigger "delta AIS alarm event" of **+RPT** message at once.

Note: If the difference between the analog quantity read in the previous steady-state value and the current reading is greater than <*Delta AIS threshold*> parameter and this state lasts for <*Delta AIS Duration*>, the device will trigger "delta AIS alarm event" of **+RPT** message.

Note: <*Debounce Time*> and <*Validity Time*> parameters only use for digital input mode. <*Delta AIS Threshold*> and <*Delta AIS Duration*> parameter only use for analog input mode.

2.1.4.3. OUT (Digital Output Setting)

The **AT@OUT** command is used to output specified wave shape from the digital output ports. A total of four wave shapes are supported as shown below. If set to wave shape 1, the device will maintain this wave shape at the specified output port after power reset.

The digital output 1 is a latched output. The final status of the output will be latched during power off.

If a specified output port is set to wave shape 4, then the port will output square wave. When the main power is off, the port will stop outputting the wave; if the main power is turned on again, the port will start to output the wave again. If the device is rebooted, the port will still output the wave.

Wave Shape 1:

✓ <*Duration*> = 0ms, <*Toggle Times*> = 0

Figure 1: Wave Shape 1

Wave Shape 2:

✓ <*Duration*> = 500ms, <*Toggle Times*> = 1

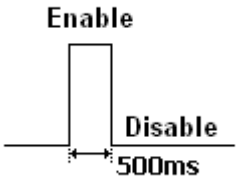


Figure 2: Wave Shape 2

Wave Shape 3:

✓ <*Duration*> = 800ms, <*Toggle Times*> = 3

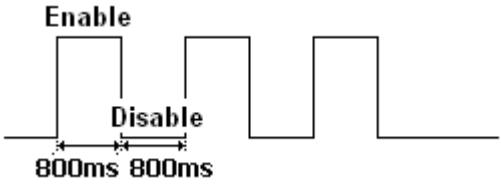


Figure 3: Wave Shape 3

Wave Shape 4:

✓ <*Duration*> = 800ms, <*Toggle Times*> = 0

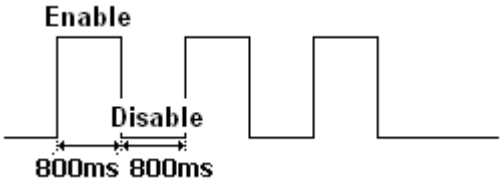


Figure 4: Wave Shape 4

➢ **AT@OUT=**

Example: **AT@OUT=at,1,0,2,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@OUT=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Reserved	0		
4	Output ID	1	1	1
5	Output Status	1	0 – 1	
6	Duration	<=3	0 – 255 (*100ms)	

7	Toggle Times	<=3	0 – 255	
8	Serial Number	4	0000 – FFFF	
9	Tail Character	1	#	#

✧ <*Output ID*>: A numerical to indicate the ID of output ports.

✧ <*Output Status*>: Valid only for the wave shape 1 as shown in **Figure 1** and used to set the final status of the output port.

- 0: Disable status.
- 1: Enable status.

✧ <*Duration*>: Please refer to **Figure 1, Figure 2, Figure 3** and **Figure 4**. Unit: 100ms.

✧ <*Toggle Times*>: Please refer to **Figure 1, Figure 2, Figure 3** and **Figure 4**.

When <*Duration*> is set to 0, <*Toggle Times*> must be set to 0 rather than other values. Otherwise, the command is invalid.

2.1.4.4. AOS (Alarm Output Setting)

The **AT@AOS** command is used to alarm event output specified wave shape from the digital output ports. Such as GEO fence alarm event, over speed alarm event, Harsh Behavior event and Driver ID authorization event.

➢ **AT@AOS=**

Example: **AT@AOS=at,7,1,0,2,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@AOS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Alarm OUT1 Trigger Mask	<=8	0 – FFFFFFFF	0
4	Output ID	1	1	1
5	Output Status	1	0 – 1	
6	Duration	<=3	0 – 255 (*100ms)	
7	Toggle Times	<=3	0 – 255	
8	Output Alarm Report	1	0 – 1	0
9	Serial Number	4	0000 – FFFF	
10	Tail Character	1	#	#

✧ <*Alarm OUT1 Trigger Mask*>: Component mask of the alarm output trigger.

- Bit 0: (AT@GEO) Quit GEO fence alarm event.
- Bit 1: (AT@GEO) Enter GEO fence alarm event.
- Bit 2: (AT@SPD) Over speed alarm event.
- Bit 3: (AT@HBS) Harsh Behavior alarm event.

- Bit 4: (AT@DRV) Driver ID authorization event.
- Bit 5: (AT@DAM) negative trigger input 1 On event.
- Bit 6: (AT@DAM) negative trigger input 1 Off event.
- Bit 7: (AT@JAM) Jamming On event.

✧ <*Output Alarm Report*>: it will control "output alarm event" of **+RPT** whether report to server, when state of output wave 1 change.

- 0: Disable Report.
- 1: Enable Report.

2.1.5. 1wire & RS485

2.1.5.1. WAS (1wire Application Setting)

The command **AT@WAS** is used to configure the parameters of 1-wire devices including iButton and temperature sensors.

➢ **AT@WAS=**

Example: **AT@WAS=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@WAS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	iButton Timer	<=2	0 – 10 (sec)	0
4	Temperature Timer	<=3	0|10 – 255 (sec)	0
5	1wire Data Mask	<=2	00 – FF	0
6	Serial Number	4	0000 – FFFF	
7	Tail Character	1	#	#

✧ <*iButton Timer*>: The check interval of searching the iButton ID.
Note: The sequence of iButton ID is in reverse order of the ID on iButton. The ID structure is composed of <*1 byte family ID*> + <*6 bytes serial number*> + <*1 byte CRC*>. Thus, the first byte of the ID number represents iButton's family ID. For example, if the ID on iButton is 12 345678901234 01, then the actual iButton ID is 01 341290785634 12.

✧ <*Temperature Timer*>: The interval of reading the temperature sensor value. If more than one temperature sensors are connected, the device will read the temperature sensors one by one at the interval specified by <Temperature Timer>. 0 means "Disable temperature detection".

✧ <*1wire Data Mask*>: it controls which 1wire data will be reported when the bit 9 of <Event Data Mask> in the AT@GCS command is enabled.

- Bit 0: Temperature Mask.
- Bit 1: Humidity Mask.

2.1.5.2. ETS (External Temperature Setting)

The command **AT@ETS** is used to configure external temperature alarm parameters.

➢ **AT@ETS=**

Example: **AT@ETS=at,,,0,0,0,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@ETS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Sensor Number	1	0 - 5	0
4	Sensor Type	1	1 – 2	
5	Sensor ID	<=16		
6	Temperature Alarm Mode	1	0 – 4	0
7	High Temperature	<=3	-55 – 125 (℃)	0
8	Low Temperature	<=3	-55 – 125 (℃)	0
9	Reserved	0		
10	Temperature Duration	<=4	10 – 3600 (sec)	10
11	Serial Number	4	0000 – FFFF	
12	Tail	1	#	#

✧ <*Sensor Number*>: It indicates the amount of the sensors need to be set up.
If sensor number is blank, device will show all of the sensors which are connected.

✧ <*Sensor Type*>: This parameter defines the type of current sensors.

- 1: 1wire temperature sensor.
- 2: BLE temperature Sensor

✧ <Sensor ID>: The ID of the temperature sensor. Total of four sensors are supported.

✧ <*Temperature Alarm Mode*>: This parameter defines to enable temperature alarm or not.

- 0: Disable temperature alarm.
- 1: Report the alarm message when the current temperature is equal or higher than the <high *Temperature*>.
- 2: Report the alarm message when the current temperature is lower than the <Low *Temperature*>.
- 3: Report the alarm message when the current temperature is within the temperature range.
- 4: Report the alarm message when the current temperature is outside the temperature range.

- ✧ <*High Temperature*>: The upper limit of the temperature range.
- ✧ <*Low Temperature*>: The lower limit of the temperature range.
- ✧ <*Temperature Duration*>: If the temperature is within the specified temperature range and maintained for a period of time specified by <*Temperature* Duration>, the temperature alarm will be triggered.

2.1.5.1. EHS (External Humidity Setting)

The command **AT@EHS** is used to configure external humidity alarm parameters.

- ➢ **AT@EHS=**

Example: **AT@EHS=at,,,0,0,0,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@EHS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Sensor Number	1	0 - 5	0
4	Sensor Type	1	1 – 2	
5	Sensor ID	<=16		
6	Humidity Alarm Mode	1	0 – 4	0
7	High Humidity	<=2	5 – 95 %	0
8	Low Humidity	<=2	5 – 95 %	0
9	Reserved	0		
10	Humidity Duration	<=4	10 – 3600 (sec)	10
11	Serial Number	4	0000 – FFFF	
12	Tail	1	#	#

- ✧ <*Sensor Number*>: It indicates the amount of the sensors need to be set up.
  If sensor number is blank, device will show all of the sensors which are connected.
- ✧ <*Sensor Type*>: This parameter defines the type of current sensors.
  - 1: 1wire humidity sensor.
  - 2: BLE humidity sensor.
- ✧ <Sensor ID>: The 1wire ID of the humidity sensor or the MAC address of the BLE humidity sensor. Total of four sensors are supported.
- ✧ <*Humidity Alarm Mode*>: This parameter defines to enable humidity alarm or not.
  - 0: Disable humidity alarm.
  - 1: Report the alarm message when the current humidity is equal or higher than the <*high Humidity*>.

- 2: Report the alarm message when the current humidity is lower than the <*Low Humidity*>.
- 3: Report the alarm message when the current humidity is within the humidity range.
- 4: Report the alarm message when the current humidity is outside the humidity range.

✧ <*High Humidity*>: The upper limit of the humidity range.

✧ <*Low Humidity*>: The lower limit of the humidity range.

✧ <*Humidity Duration*>: If the humidity is within the specified humidity range and maintained for a period of time specified by <*Humidity* Duration>, the humidity alarm will be triggered.

2.1.5.2. DRV (Driver ID Setting)

The command **AT@DRV** is used to driver ID authorization. In order to us this function, we need get the driver ID by iButton/RFID reader/BLE Beacon and control the engine starter by external relay. When the device gets an ID, It will check if the ID is in the white list. If yes, It will be authorized to start the engine. At the same time, it will report this ID to the backend server. The drivers need to scan their IDs each time before they start the engine.

➢ **AT@DRV=**

Example: **AT@DRV=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@DRV=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0|1|2	0
4	Reserved	0		
5	Start Index	<=3	1 – 1024	
6	End Index	<=3	1 – 1024	
7	ID Number List	<=16*28	'0' – '9','a' – 'f', 'A' – 'F' | "NULL"	
8	ID Authorized Validity Time	<=3	15 – 600 (sec)	30
9	Authorized Timeout after IGF	<=3	0|15 – 600 (sec)	30
10	DRV Report Mode	1	0 – 7	7
11	Serial Number	4	0000 – FFFF	
12	Tail Character	1	#	#

✧ <*Mode*>: The working mode of the ID authentication function.

- 0: Disable function.
- 1: Enable function, only authorized ID card can unlock .
- 2: Enable function, any ID card can unlock.

- <*Start Index*>, <*End Index*>: The index range of the driver ID white list. For example, if <*Start Index*> is set to 1 and <*End Index*> is set to 2, then the first two ID numbers in the white list will be updated by the numbers provided in the parameter <*ID Number List*>. <*Start Index*> and <*End Index*> determine the total amount of ID numbers that will be updated. If any of them is blank, there should be no <*ID Number List*> updated in the blank value. Maximum of 8 numbers can be updated each time.
- <*ID Number List*>: A list of comma-separated ID numbers to be updated to the white list. The quantity of the ID numbers is determined by <*Start Index*> and <*End Index*>.
- **Note:** If more ID Numbers are needed, please adjust <*Start Index*> and <*End Index*> for appropriate setup. If some ID Numbers in <*ID Number List*> are blank, then the corresponding ID Numbers will be deleted. For example, to delete the 4th, 5th and 6th ID Numbers of the <*ID Number List*>, please set <*Start Index*> to 4 and set <*End Index*> to 6 and keep the three ID Numbers of <ID Number List> blank.
- <*ID Authorized Validity Time*>: It will remain authorized for this period time, when the ID is valid. If the driver has not trigger ignition on in the <*ID Authorized Validity Time*> period time of after authorized ID, the driver needs re-swipe card to be authorized.
- <*Authorized Timeout after IGF*>: It is the timeout to quit the authorization after ignition off. Drivers don't need to scan the ID within this period time to start the engine again.
- <*DRV Report Mode*>: The mode of reporting authorized ID.
  - Bit 0: Report the ID which is authorized.
  - Bit 1: Report the ID which is unauthorized.
  - Bit 2: Report the ID which has logged out.

2.1.5.3. RCS (RS485 Communication Setting)

The command **AT@RCS** is used to configure RS485 communication setting.

- **AT@RCS=**

Example: **AT@RCS=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@RCS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Reserved	0		
4	Baud Rate Index	<=2	1 – 12	12
5	Data Bits	1	7 – 8	8
6	Stop Bits	1	1	1
7	Parity Bits	1	0	0
8	Reserved	0		
9	Reserved	0		

10	Serial Number	4	0000 ~ FFFF	
11	Tail Character	1	#	#

✧ <*Baud Rate Index*>: The index of the supported baud rate of the serial port. All supported baud rates are listed below:

Baud Rate Index	Baud Rate
1	**1200**
2	**2400**
3	**4800**
4	**7200**
5	**9600**
6	**14400**
7	**19200**
8	**28800**
9	**33900**
10	**38400**
11	**57600**
12	**115200**

✧ <*Data Bits*>: The data bits of the UART, the parameter value can be 7 or 8.

✧ <*Stop Bits*>: The stop bits of the UART.

- 1: 1 stop bit

✧ <*Parity Bits*>: The parity bits of the UART.

- 0: None parity

2.1.5.4. RAS (RS485 Accessory Setting)

The command **AT@RAS** is used to configure RS485 accessory setting, it supports connected with multiple external accessories via RS485.

➢ **AT@RAS=**

Example: **AT@RAS=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@RAS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Accessory Type Number	1	0 – 1	0
4	Accessory Model	1	0 - 1	0
5	Address ID Number	1	1 - 5	1
6	Address ID	<=2	HEX String	
7	Serial Number	4	0000 ~ FFFF	

8	Tail Character	1	#	#

✧ <*Accessory Type Number*>: It expresses connected the number of RS485 accessory type:

✧ <*Accessory Model*>: It expresses connected accessory model.

- 0: No RS485 device.
- 1: DYP-U02 ultrasonic oil level sensor, 9600 baud rate, 8 data bits, 1 stop bit, and no parity.

✧ <*Address ID number*>: It expresses the number of accessory RS485 address.

✧ <*Address ID*>: It expresses accessory RS485 address.

2.1.6. CANbus & Tacho

2.1.6.1. CAN (CANbus Setting)

This command **AT@CAN** is used to configure the CAN function parameters, when the power saving mode is deep power saving mode and go into ignition off state, the CAN module will mandatory to close.

If the <*Mode*> is not equal to 0, the <*CAN Report Interval*> and <*CAN Report Interval IGF*> are all equal to 0, the CAN data will report by "Regular Report Event" of **+RPT** message.

➢ **AT@CAN=**

Example: **AT@CAN=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@CAN=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Mode	1	0 – 4	0
4	Special Car Model	<=5	158|160|161	160
5	CAN Report Interval	<=5	0|5 – 86400 (sec)	10
6	CAN Report Interval IGF	<=5	0|60 – 86400 (sec)	0
7	CAN Info Mask 1	<=8	0 – FFFFFFFF	C00FFFFF
8	CAN Info Mask 2	<=8	0 – FFFFFFFF	007FFFFF
9	Reserved	0		
10	Electric Info Mask 1	<=8	0 – FFFFFFFF	C0
11	Reserved	0		
12	Additional Event Mask	1	0 – 7F	0
13	CAN 120 ohms Mask	1	0 – 3	0
14	Tachograph Driver1 Working	<=8	0 – FFFFFFFF	0

	Time Mask			
15	Tachograph Driver2 Working Time Mask	<=8	0 – FFFFFFFF	0
16	Tachograph Info Mask	<=8	0 – FFFFFFFF	0
17	Special Car Info Mask	<=8	0 – FFFFFFFF	0
18	NMEA2000 Info Mask	<=8	0 – FFFFFFFF	0
19	Tacho Driver Time Mode	1	0 – 3	0
20	Serial Number	4	0000 – FFFF	
21	Tail Character	1	#	#

✧ <*Mode*>: The working mode of CANBUS function.

- 0: Disable CANBUS function.
- 1: Enable CANBUS function, only CANBUS device information will be reported.
- 2: Enable CANBUS function, only tachograph information will be reported
- 3: Enable CANBUS function, CANBUS device and tachograph information will be reported.
- 4: Enable the CANBUS function, the CAN chip will work according to <Machine Mode>.

✧ <*Special Car Model*>: A number indicates the car model of the machine to be set from the machine list, this parameter is valid when <*mode*> set to 4.

✧ <*CAN Report Interval*>: The time interval for sending the CANBUS information message to the backend server when the ignition is on. Its value range is 0|5 – 86400s. 0 means "Do not report the CANBUS information message in ignition on state.

✧ <*CAN Report Interval IGF*>: The time interval for sending the CANBUS information message to the backend server when the ignition is off. Its value range is 0|60 – 86400s. 0 means "Do not report the CANBUS information message in ignition off state.

✧ <*CAN Info Mask 1*>: Bitwise mask to configure the composition of CANBUS information 1 in the CANBUS info event of **+RPT** report message.

Mask Bit	**Item**	**Description**
Bit 31	Reserved	
Bit 30	<*Handbrake Applies During Ride Count*>	Counts events when handbrake is pulled-up while driving (speed is greater than 5 km/h).
Bit 29	<*Total Engine Cold Running Time*>	Total driving time with cold engine (engine coolant temperature below 70°C.
Bit 28	<*Engine Starts by Ignition Count*>	Total number of Engine starts by ignition.
Bit 27	<*Engine All Starts Count*>	Total number of Engine starts.
Bit 26	<*Engine Cold Starts Count*>	Number of cold Engine starts.
Bit 25	<*Total Vehicle Engine Overspeed Time*>	The total time when the vehicle engine speed is greater than the limit defined in CAN Module

		configuration.
Bit 24	<*Total Vehicle Overspeed Time*>	The total time when the vehicle speed is greater than the limit defined in CAN Module configuration.
Bit 23	<*Doors*>	An 8-bit hexadecimal number. Each bit indicates information of one door.
Bit 22	<*Lights*>	An 8-bit hexadecimal number. Each bit indicates information of one light.
Bit 21	<*Detailed Information / Indicators 2*>	A hexadecimal number. Each bit indicates information of one indicator.
Bit 20	<*Detailed Information / Indicators 1*>	A hexadecimal number. Each bit indicates information of one indicator.
Bit 19	<*Axle Weight 4th*>	Weight of vehicle’s fourth axle
Bit 18	<*Axle Weight 3rd*>	Weight of vehicle’s third axle
Bit 17	<*Axle Weight 2nd*>	Weight of vehicle’s second axle
Bit 16	<*Axle Weight 1st*>	Weight of vehicle’s first axle
Bit 15	<*Total Idle Fuel Used*>	Number of liters of fuel used since vehicle manufacture or device installation
Bit 14	<*Total Engine Idle Time*>	Time of engine running during idling status (vehicle at a standstill) since vehicle manufacture or device installation
Bit 13	<*Total Driving Time*>	Time of engine running during driving (non-zero speed) since vehicle manufacture or device installation
Bit 12	<*Total Engine Hours*>	Time of engine running since vehicle manufacture or device installation
Bit 11	<*Accelerator Pedal Pressure*>	Pressure applied onto accelerator pedal
Bit 10	<*Range*>	The number of kilometers to drive on remaining fuel
Bit 9	<*Fuel Level*>	The level of fuel in vehicle's tank (in liters or percentage)
Bit 8	<*Fuel Consumption*>	The fuel consumption of the engine
Bit 7	<*Engine Coolant Temperature*>	Engine coolant temperature
Bit 6	<*Engine RPM*>	Revolutions per minute of the engine
Bit 5	<*Vehicle Speed*>	Vehicle road speed

Bit 4	<*Total Fuel Used*>	Number of liters of fuel used since vehicle manufacture or device installation
Bit 3	<*Total Distance Impulses*>	Vehicle total distance measured in Impulses (if distance from dashboard is not available)
Bit 2	<*Total Distance*>	Vehicle total distance
Bit 1	<*Ignition Key*>	Ignition status
Bit 0	<*VIN*>	Vehicle identification number

✧ <*CAN info Mask 2*>: Bitwise mask to configure the composition of CANBUS information 2 in the CANBUS info event of **+RPT** report message.

Mask Bit	Item	Description
Bit 31	Reserved	
Bit 30	<*Time To Service*>	It's the remaining time to maintenance service
Bit 29	<*Tyres Information*>	It includes tyres pressures and temperatures.
Bit 28	<*Total Gaseous Fuel Usage*>	If the vehicle is powered by gas and provides gas usage counter (eighter for gas-powered or bifuel vehicles) – the value is placed here. Otherwise the parameter is unavailable.
Bit 27	<*Total Fuel From Vehicle*>	It's always total fuel counter read from the vehicle. If the vehicle does not provide total fuel counter- this parameter is unavailable.
Bit 26	<*Only Fuel Level Combustion*>	It shows fuel level only (never switches to battery charge level) – for combustion and hybrid cars (in percents or liters, depending on car model). When car doesn't have combustion engine – this parameter remains unavailable.
Bit 25	<*Gaseous Fuel Level*>	The alternative fuel levels.
Bit 24	<*DTC Codes*>	
Bit 23	<*Ambient Temperature*>	
Bit 22	<*Service Distance*>	The parameter describes distance left to diagnostic car review.
Bit 21	<*Engine Torque*>	The engine torque. Unit: percentage.
Bit 20	<*Rapid Accelerations*>	Total number of rapid accelerations since installation (calculation based on CANBUS settings of speed increase time and value)

Bit 19	<*Rapid Brakings*>	Total number of rapid brakings since installation (calculation based on CAN 100 settings of speed decrease time and value)
Bit 18	<*Registration Number*>	The vehicle registration number
Bit 17	Reserved	
Bit 16	Reserved	
Bit 15	<*Trailer VIN*>	VIN of trailer
Bit 14	<*Oil Temperature*>	Vehicle oil tempearture
Bit 13	<*Total Brake Applications*>	Count of applying brake pedal (braking process initiated by brake pedal)
Bit 12	<*Total Accelerator Kick-down Time*>	Total time when accelerator pedal is pressed over 90%
Bit 11	<*Total Cruise Control Time*>	Total time when the vehicle speed is controlled by cruise-control module
Bit 10	<*Total Effective Engine Speed Time*>	Total time when the vehicle engine speed is effective
Bit 9	<*Total Accelerator Kick-downs*>	Count of accelerator pedal kick-downs (with the pedal pressed over 90%)
Bit 8	<*Pedal Braking Factor*>	It measures how often the driver brakes with brake pedal or with engine and stores both counts (which are always increasing). Decreasing speed with brake pedal pressed causes increase of pedal braking factor.
Bit 7	<*Engine Braking Factor*>	It measures how often driver brakes with brake pedal or with engine and stores both counts (which are always increasing). Decreasing speed with no pedal pressed causes increase of engine braking factor.
Bit 6	<*Analog Input Value*>	Analog input value
Bit 5	*Axle Weight 7th*	Weight of vehicle’s seventh axle
Bit 4	*Axle Weight 6th*	Weight of vehicle’s sixth axle
Bit 3	*Axle Weight 5th*	Weight of vehicle’s fifth axle
Bit 2	<*Power Mode*>	Power mode.
Bit 1	<*Retarder Usage*>	The usage of Retarder.
Bit 0	<*Ad-Blue Level*>	The level of Ad-Blue

✧ <*Electric Info Mask 1*>: Bitwise mask to configure the composition of electric bus information in the CANBUS info event of **+RPT** report message.

Mask Bit	Item	Description
Bit 31	Reserved	
…	…	
Bit 13	Reserved	
Bit 12	<*Only Battery charge Level*>	It's used in electric and hybrid vehicles and shows battery charge level (state of charge), it's always unavailable in cars having only combustion engine.
Bit 11	<*Total Energy Charged*>	Sum of energy transmitted to battery from external charger.
Bit 10	<*Total Energy Used When Idling*>	With vehicle speed 0 km/h.
Bit 9	<*Total Energy Used*>	Sum of energy used by car's engine and equipment; recuperated energy does not affect this parameter.
Bit 8	<*Battery State of Health (SoH)*>	For electric cars this parameter stands for battery general condition (100% means brand new, 0% totally damaged).
Bit 7	<*Battery Instantaneous Power*>	For electric cars this parameter stands for instantaneous power used by (positive values) or recuperated from (negative values) the car.
Bit 6	<*Battery Charging Current*>	Indicates the electric car's battery charging current.
Bit 5	<*Battery Temperature*>	Indicates the battery temperature.
Bit 4	<*Charging States*>	The status of battery charging and Charging cable connected in electric cars.
Bit 3	<*Battery Level*>	The battery charge level.
Bit 2	<*Total Energy Recuperated*>	Sum of energy transmitted to battery with no charger connected.
Bit 1	<*Battery Charging Cycles Count*>	For electric cars this parameter counts cycles of battery charging (increments at every finished cycle of charging).
Bit 0	<*Battery Instantaneous Voltage*>	For electric cars this parameter stands for instantaneous high voltage of battery cells

		(Read from BMS)

✧ <*Additional Event Mask*>: A bitwise numeral to control whether to send the CAN information message by additional event mask, 0 means "Ignore all additional events".

- Bit 0: By ignition on/off event.
- Bit 1: When the <*Tachograph Driver 1 Card Number*> has changed.
- Bit 2: When the <*Tachograph Driver 2 Card Number*> has changed.
- Bit 3: When the Driver1 working status in the parameter <*Tachograph Information*> has changed.
- Bit 4: when the Driver2 working status in the parameter <*Tachograph Information*> has changed.
- Bit 5: when the "Out of Scope" condition in the parameter <*Tachograph Information Expand*> has changed.
- Bit 6: when the value of <*Ultima F2*> parameter has changed (Clamp unlocking, Clamp locking, stabilizer-roller moving down, stabilizer-roller moving up).

✧ <*CAN 120 ohms Mask*>: A bitwise numeral to control whether to enable 120 ohms resistor parallel between CAN1 L and CAN1 H or between CAN2 L and CAN2 H.

- Bit 0: CAN1 L and CAN1 H parallel 120 ohms resistor.
- Bit 1: CAN2 L and CAN2 H parallel 120 ohms resistor.

✧ <*Tachograph Driver1 Working Time Mask*>: It expands CANBUS information in "CANBUS Info Event" of **+RPT** message. Bitwise report mask to configure the composition of expanded CANBUS information of "CANBUS Info Event" message report.

Mask Bit	Item	Description
Bit 31	Reserved	
Bit 30	Reserved	
Bit 29	Reserved	
Bit 28	*<Drv1 Maximum Daily Driving Time>*	Driver 1: Maximum daily driving time.
Bit 27	*<Drv1 Cumulative Uninterrupted Rest Time>*	Driver 1: Cumulative uninterrupted rest time.
Bit 26	*<Drv1 Current Weekly Driving Time>*	Driver 1: Current weekly driving time.
Bit 25	*<Drv1 Current Daily Driving Time>*	Driver 1: Current daily driving time.
Bit 24	*<Drv1 Accumulated Driving Time Previous And Current Week>*	Driver 1: Accumulated driving time previous and current week.
Bit 23	*<Drv1 Current Duration Of Selected Activity>*	Driver 1: Current duration of selected activity.
Bit 22	*<Drv1 Cumulative Break Time>*	Driver 1: Cumulative break time.

Bit 21	*<Drv1 Continuous Driving Time>*	Driver 1: Continuous driving time.
Bit 20	*<Drv1 Open Compensation In 2nd Week Before Last>*	Driver 1: Open compensation in 2nd week before last.
Bit 19	*<Drv1 Open Compensation In Week Before Last>*	Driver 1: Open compensation in week before last.
Bit 18	*<Drv1 Open Compensation In The Last Week>*	Driver 1: Open compensation in the last week.
Bit 17	*<Drv1 Minimum Weekly Rest>*	Driver 1: Minimum weekly rest.
Bit 16	*<Drv1 Time Left Until New Weekly Rest Period>*	Driver 1: Time left until new weekly rest period.
Bit 15	*<Drv1 Remaining Driving Time Of Current Week>*	Driver 1: Remaining driving time of current week.
Bit 14	*<Drv1 Minimum Daily Rest>*	Driver 1: Minimum daily rest.
Bit 13	*<Drv1 Time Left Until New Daily Rest Period>*	Driver 1: Time left until new daily rest period.
Bit 12	*<Drv1 Remaining Driving Time On Current Shift>*	Driver 1: Remaining driving time on current shift.
Bit 11	*<Drv1 Duration Of Next Driving Period>*	Driver 1: Duration of next driving period.
Bit 10	*<Drv1 Time Left Until Next Driving Period>*	Driver 1: Time left until next driving period.
Bit 9	*<Drv1 Remaining Time Of Current Break Rest>*	Driver 1: Remaining time of current break rest.
Bit 8	*<Drv1 Duration Of Next Break Rest>*	Driver 1: Duration of next break rest.
Bit 7	*<Drv1 Remaining Time Until Next Break Or Rest>*	Driver 1: Remaining time until next break or rest.
Bit 6	*<Drv1 Remaining Current Drive Time>*	Driver 1: Remaining current drive time.
Bit 5	*<Drv1 Number Of Used Reduced Daily Rest Periods>*	Driver 1: Number of times 9h daily driving times exceeded.
Bit 4	*<Drv1 Number Of Times 9h Daily Driving Times Exceeded>*	Driver 1: Number of times 9h daily driving times exceeded.
Bit 3	*<Drv1 Maximum Daily Period>*	Driver 1: Tachograph drivers' maximum daily

		period.
Bit 2	*<Drv1 End Of Second Last Weekly Rest Period>*	Driver 1: end of second last weekly rest period.
Bit 1	*<Drv1 End Of Last Weekly Rest Period>*	Driver 1: end of last weekly rest period.
Bit 0	*<Drv1 End Of Last Daily Rest Period>*	Driver 1: end of last daily rest period.

✧ <*Tachograph Driver2 Working Time Mask*>: It expands CANBUS information in "CANBUS Info Event" of **+RPT** message. Bitwise report mask to configure the composition of expanded CANBUS information of "CANBUS Info Event" message report.

Mask Bit	Item	Description
Bit 31	Reserved	
Bit 30	Reserved	
Bit 29	Reserved	
Bit 28	*<Drv2 Maximum Daily Driving Time>*	Driver 2: Maximum daily driving time.
Bit 27	*<Drv2 Cumulative Uninterrupted Rest Time>*	Driver 2: Cumulative uninterrupted rest time.
Bit 26	*<Drv2 Current Weekly Driving Time>*	Driver 2: Current weekly driving time.
Bit 25	*<Drv2 Current Daily Driving Time>*	Driver 2: Current daily driving time.
Bit 24	*<Drv2 Accumulated Driving Time Previous And Current Week>*	Driver 2: Accumulated driving time previous and current week.
Bit 23	*<Drv2 Current Duration Of Selected Activity>*	Driver 2: Current duration of selected activity.
Bit 22	*<Drv2 Cumulative Break Time>*	Driver 2: Cumulative break time.
Bit 21	*<Drv2 Continuous Driving Time>*	Driver 2: Continuous driving time.
Bit 20	*<Drv2 Open Compensation In 2nd Week Before Last>*	Driver 2: Open compensation in 2nd week before last.
Bit 19	*<Drv2 Open Compensation In Week Before Last>*	Driver 2: Open compensation in week before last.
Bit 18	*<Drv2 Open Compensation In The Last Week>*	Driver 2: Open compensation in the last week.
Bit 17	*<Drv2 Minimum Weekly Rest>*	Driver 2: Minimum weekly rest.
Bit 16	*<Drv2 Time Left Until New Weekly Rest Period>*	Driver 2: Time left until new weekly rest period.

Bit 15	*<Drv2 Remaining Driving Time Of Current Week>*	Driver 2: Remaining driving time of current week.
Bit 14	*<Drv2 Minimum Daily Rest>*	Driver 2: Minimum daily rest.
Bit 13	*<Drv2 Time Left Until New Daily Rest Period>*	Driver 2: Time left until new daily rest period.
Bit 12	*<Drv2 Remaining Driving Time On Current Shift>*	Driver 2: Remaining driving time on current shift.
Bit 11	*<Drv2 Duration Of Next Driving Period>*	Driver 2: Duration of next driving period.
Bit 10	*<Drv2 Time Left Until Next Driving Period>*	Driver 2: Time left until next driving period.
Bit 9	*<Drv2 Remaining Time Of Current Break Rest>*	Driver 2: Remaining time of current break rest.
Bit 8	*<Drv2 Duration Of Next Break Rest>*	Driver 2: Duration of next break rest.
Bit 7	*<Drv2 Remaining Time Until Next Break Or Rest>*	Driver 2: Remaining time until next break or rest.
Bit 6	*<Drv2 Remaining Current Drive Time>*	Driver 2: Remaining current drive time.
Bit 5	*<Drv2 Number Of Used Reduced Daily Rest Periods>*	Driver 2: Number of times 9h daily driving times exceeded.
Bit 4	*<Drv2 Number Of Times 9h Daily Driving Times Exceeded>*	Driver 2: Number of times 9h daily driving times exceeded.
Bit 3	*<Drv2 Maximum Daily Period>*	Driver 2: Tachograph drivers' maximum daily period.
Bit 2	*<Drv2 End Of Second Last Weekly Rest Period>*	Driver 2: end of second last weekly rest period.
Bit 1	*<Drv2 End Of Last Weekly Rest Period>*	Driver 2: end of last weekly rest period.
Bit 0	*<Drv2 End Of Last Daily Rest Period>*	Driver 2: end of last daily rest period.

✧ <*Tachograph Info Mask*>: Bitwise mask to configure the composition of tachograph information in the CANBUS info event of **+RPT** report message.

Mask Bit	Item	Description
Bit 31	Reserved	
…	…	
Bit 9	Reserved	

Bit 8	<*Real Time Clock Date and Time*>	Real time clock date and time display on car's dashboard
Bit 7	<*Tachograph Driver 2 Name*>	The name of tachograph driver 2
Bit 6	<*Tachograph Driver 1 Name*>	The name of tachograph driver 1
Bit 5	<*Tachograph Driver 2 Card Number*>	The card number of tachograph driver 2
Bit 4	<*Tachograph Driver 1 Card Number*>	The card number of tachograph driver 1
Bit 3	<*Tachograph Driving Direction*>	Vehicle driving direction from tachograph
Bit 2	<*Tachograph Vehicle Motion Signal*>	Vehicle motion signal from tachograph
Bit 1	<*Tachograph Overspeed Signal*>	Tachograph overspeed signal for the vehicle
Bit 0	<*Tachograph Information*>	It consists of Tachograph Information

✧ <*Special Car Info Mask*>: Bitwise mask to configure the composition of special car information in the CANBUS info event of **+RPT** report message.

Mask Bit	Item	Description
Bit 31	Reserved	
…	…	
Bit 1	<*Multilift Ultima Information*>	Data from special equipment "Multilift Ultima 21s", <*Special Car Model*> must be set to 160 through AT@CAN command. Data from special equipment "Multilift Ultima 26s", <*Special Car Model*> must be set to 161 through AT@CAN command. (automatic synchronization would lose effectiveness at that time)
Bit 0	<*e-GENSET Information*>	Data from special equipment "Etteplan Thriatlon e-GENSET", <*Special Car Model*> must be set to 158 through AT@CAN command. (automatic synchronization would lose effectiveness at that time)

✧ <*NMEA2000 Info Mask*>: NMEA2000 information read from NMEA CANbus of boat, bitwise mask to configure the composition of NMEA2000 information in the CANBUS info event of **+RPT** report message.

Mask Bit	Item	Description
Bit 31	Reserved	
…	…	

Bit 4	Wind Speed	The wind speed of NMEA2000
Bit 3	Wind Angle	The wind angle of NMEA2000
Bit 2	Speed Through Water	The speed through water of NMEA2000
Bit 1	Heading	The heading of NMEA2000
Bit 0	Depth Blow Transducer	The depth blow transducer of NMEA2000

✧ <*Tacho Driver Time Mode*>: Querying tachograph about the drivers' working times.

- 0: Querying is disabled at all. Drivers' working times are not available.
- 1: Querying is enabled only when ignition on (for vehicles which don't support ignition signal – when engine is on).
- 2: "Smart mode": querying is enabled when CANbus is active.
- 3: Querying is always enabled, even if car is in sleep mode.

2.1.6.2. CFU (CAN Module FOTA Upgrade)

The command AT@CFU is used to upgrade the firmware in CANBUS over the air.

➢ **AT@CFU=**

Example: **AT@CFU=at,1,10,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@CFU=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Working Mode	1	0|1	1
4	Reserved	1	0	0
5	Reserved	0		
6	Server URL	100	Legal URL	
7	Update Type	1	0	0
8	Serial Number	4	0000 – FFFF	
9	Tail Character	1	#	#

✧ <*Working Mode*>: The working mode of CANBUS Module firmware and configuration upgrade.

- 0: Stop firmware upgrade.
- 1: Start firmware upgrade.

✧ <*Server URL*>: The URL to download the package.

✧ <*Update Type*>: The type of CAN Module update over the air.

- 0: CAN Module firmware update

2.1.6.3. TAC (Tachograph DDD File Request Setting)

The command **AT@TAC** is used to tachograph DDD file request setting.

➢ **AT@TAC=**

Example: **AT@TAC=at,0001,0,1,20240730153000,20240731153000,F,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@TAC=
2	Password	4 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Request ID	4	0000 – FFFF	
4	Report Mode	1	0|1|2	
5	Read file type	1	1 – 7	
6	Start Time	14	YYYYMMDDHHMMSS	
7	End Time	14	YYYYMMDDHHMMSS	
8	Segment Mask	1	0 – F	F
9	Serial Number	4	0000 – FFFF	
10	Tail Character	1	#	#

✧ <*Request ID*>: Authorized request's identifier. It is different to set the request each time.

✧ <*Report Mode*>: It indicates how the DDD file is reported to the end user.

- 0: DDD file upload to server by **+LDP** message.
- 1: DDD file upload by FTP, **AT@FTP** command should be pre-set.
- 2: Diagnostic file upload by FTP, **AT@FTP** command should be pre-set.

2: Diagnostic file upload by FTP, AT@FTP command should be pre-set.

✧ <*Read file Type*>: The DDD file needs to be read.

- Bit 0: Request to read from tachograph memory.
- Bit 1: Request to read from driver card in slot 1.
- Bit 2: Request to read from driver card in slot 2.

Note: Here are the timeout instructions for file reading.

◆ <*Read file Type*> of **AT@TAC** is 0x01

CAN Logistic getting file from Tachograph timeout is 15 minutes.

Device getting file from CAN Logistic timeout is 5 minutes.

◆ <Read file Type> of **AT@TAC** is 0x02 or 0x04

CAN Logistic getting file from Tachograph timeout is 5 minutes.

Device getting file from CAN Logistic timeout is 3 minutes.

◆ <Read file Type> of **AT@TAC** is 0x03 or 0x05

CAN Logistic getting file from Tachograph timeout is 20 minutes.

Device getting file from CAN Logistic timeout is 10 minutes.

◆ <Read file Type> of **AT@TAC** is 0x06

CAN Logistic getting file from Tachograph timeout is 10 minutes.

Device getting file from CAN Logistic timeout is 5 minutes.

- ◆ <Read file Type> of **AT@TAC** is 0x07

  CAN Logistic getting file from Tachograph timeout is 20 minutes.

  Device getting file from CAN Logistic timeout is 10 minutes.

✧ <*Start Time*>: Date and time at the beginning of the file to read from Tachograph. The value will transform in Unix Timestamp. If the memory of the Tachograph is not read, the data can be any value. If this parameter is not set, the memory of the Tachograph will be read from the date of the last reading. If this parameter is an invalid value, the memory of the tachograph will be read from the data, when previous read is performed.

✧ <*End Time*>: Date and time at the end of the file to read from Tachograph. The value will transform in Unix Timestamp. If the memory of the Tachograph is not read, the data can be any value. If this parameter is not set or invalid value, the memory of the Tachograph will be read from the current date.

Note: If <*Start Time*> and <*End Time*> are set to the future time, the memory of the Tachograph will be read from the date of the last reading.

✧ <*Segment Mask*>: This parameter is additional markers for reading tachograph memory (<*Read file Type*> bit0 is set 1), excluding certain blocks from download so that the downloaded file size may be smaller.

- Bit 0: Don't Download activities.
- Bit 1: Don't Download errors.
- Bit 2: Don't Download detailed speed.
- Bit 3: Don't Download technical data.

2.1.6.4. TCO (Tachograph Common Operation)

The command **AT@TCO** is used to query tachograph state, driver 1 id, name and surname, send authorization APDU packet and so on.

Note: Please do not issue the query command when requesting DDD file.

➢ **AT@TCO=**

Example: **AT@TCO=at,11,5,,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@TCO=
2	Password	4 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Sub Command	<=2	1 – 11	
4	Reserved (Optional)	0		
5	Reserved (Optional)	0		
6	Serial Number	4	0000 – FFFF	
7	Tail Character	1	#	#

- 1: Send authorization APDU to the device. Two reserved parameters are used as follows.

APDU Sequence number	4	0000 – FFFF	
APDU DATA	<=256		

- 2: DDD file request cancel. Cancel the older request.
- 3: Debug mode. Read the state of the device. Please do not send this command during the interaction of DDD file.
- 5: Inquiry Tachograph model.
- 6: Inquiry status of the device.
- 7: Inquiry Tachograph driver 1 id, name and surname.
- 8: Inquiry Tachograph driver 2 id, name and surname.
- 9: Check Tachograph interfaces status.
- 10: Setting tachograph file format. Two reserved parameters are used as follows.

Tachograph File Format	1	0 – 2	
Reserved	0		

- 11: Changes the type of communication with the tachograph used for supporting Tachograph Stoneridge gen 2. Two reserved parameters are used as follows.

Tacho Communication Type		1	0 | 5	
Reserved		0		

✧ <*APDU Sequence Number*>: The serial number of APDU, it should keep the same as that sent from the terminal to server.

✧ <*APDU Data*>: It is the APDU Data for CAN-Logistic. The value is in ASCII Hexadecimal Format.

✧ <*Tachograph File Format*>: It indicates the file format of tachograph.

- 0 - DDD (most common European format).
- 1 - TGD (format required in Spain).
- 2 - V1B/C1B (according to French "Vehicle appendix 1 B").

✧ <*Tacho Communication Type*>: It gives the possibility to change the type of communication with the tachograph.

- 0 - Standard type of tachograph communication.
- 5 - Additional Stoneridge gen 2 communication.

2.1.7. BLE

2.1.7.1. BSS (BLE Scan Setting)

The command AT@BSS is used to BLE scanning setting.

➢ **AT@BSS=**

Example: **AT@BSS=at,1,,,,,,,,,FFFF#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@BSS=

2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Scan Working Mode	1	0|1|2|3	0
4	Scan Frequency	<=5	1 – 65535 (sec)	10/
5	Scan Duration	<=2	1 – 300 (100 msec)	1
6	Filter Group	1	1 – 4	
7	Group Working Mode	1	0|1	0
8	MAC address	0 - 12		
9	Data Clear Period	<=5	0 – 65535	0
10	Match & Save Group	2	<=10	10
11	EIR Data Type	2	HEX	
12	Data offset	<=2		0
13	Data Size	<=2		0
14	Action	1	0|1	0
15	I/O	<=2	0 – 13	0
16	Match	<=20	HEX	
17	Endian Type	1	0 – 1	0
18	Multiplier	<=8		0
19	Offset	<=6	-32768 – 32767	0
20	BSS Report Interval Motion	<=5	0|5 – 86400 (sec)	60
21	BSS Report Interval Motionless	<=5	0|5 – 86400 (sec)	60
22	Serial Number	4	0000 – FFFF	
23	Tail	1	#	#

✧ <*Scan Working Mode*>: It expresses scan working mode of BLE.

- 0: Disable BLE Scanning.
- 1: Enable BLE Scanning, when the unit move, scan only once.
- 2: Enable BLE Scanning, the unit start moving, always scanning, the unit stop moving, stop scanning.
- 3: Enable BLE Scanning, always scanning.

✧ <*Scan Frequency*>: It expresses BLE scanning frequency, the unit is second.

✧ <*Scan Duration*>: It expresses BLE scanning Duration, the unit is 100 milliseconds.

✧ <*Filter Group*>: It expresses need filter group numbers of BLE accessor.

✧ <*MAC Address*>: It expresses MAC address of current filter's BLE accessory.

✧ <*Data Clear Period*>: It expresses 0-disable, if value equal or higher than 1, it will clear sensors

BLE IO elements values if they were not updated for configured amount of time.

- <*Match & Save Group*>: It expresses the group of matched and required data from broadcast package of BLE accessory.
- <*EIR Data Type*>: It expresses this parameter will indicate which type to look for and work with.
- <*Data Offset*>: It expresses start index of data we are interested in.
- <*Data Size*>: It expresses size of data we are interested in.
- <*Action*>: It expresses action to perform with data
  - 0: Match.
  - 1: Save.
- <*I/O*>: It expresses need get parameter, this parameter is used only when save active is selected, tells which elements data will be saved to.
  - 0: None.
  - 1: Battery.
  - 2: Temperature.
  - 3: Humidity.
  - 4: Door State.
  - 5: Luminosity.
  - 6: Fuel.
  - 7: Fuel Frequency.
  - 8: Custom1.
  - 9: Custom2.
  - 10: Custom3.
  - 11: Custom4.
  - 12: Custom5.
  - 13: Custom6.
- <*Match*>: It expresses hexadecimal string to be matched with sensor data. This parameter is used only when match action is selected.
- <*Endian Type*>: It expresses the endian type of save data, this parameter is used only when save action is selected.
  - 0: Little Endian.
  - 1: Big Endian.
- <*Multiplier*>: It expresses set multiplier by which read data value will be multiplied. This parameter is used only when save action is selected.
- <*Offset*>: It expresses set value to add to or subtract from the register before saving it. This parameter is used only when save action is selected.
- <*BSS Report Interval Motion*>: It expresses the “BLE Common info Event” of +RPT message report interval when the unit enter into motion state, the unit is second.
- <*BSS Report Interval Motionless*>: It expresses the “BLE Common info Event” of +RPT message report interval when the unit enter into motionless state, the unit is second.

2.1.8. OTA

2.1.8.1. UPF (OTA Upgrade Firmware)

The command AT@UPF is used to update the firmware over the air.

➢ **AT@UPF=**

Example: **AT@UPF=at,1,,,,,,,,,FFFF#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@UPF=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Working Mode	1	0|1	1
4	Reserved	1	0	0
5	Reserved	0		
6	Download URL	<=80	legal URL Address	
7	Update Type	1	0 - 3	0
8	Serial Number	4	0000 – FFFF	
9	Tail	1	#	#

✧ <*Working Mode*>: The working mode of the firmware upgrade.

- 0: Stop firmware upgrade.
- 1: Start firmware upgrade.

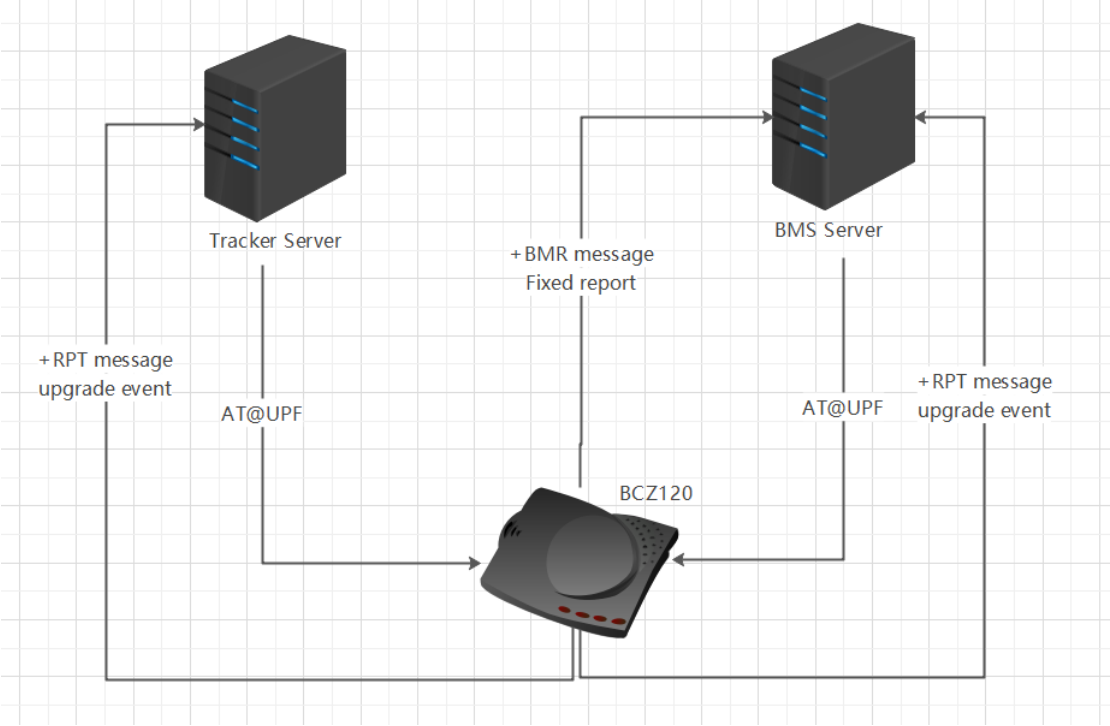
✧ <*Download URL*>: The URL to download the package.

✧ <*Update Type*>: The type of terminal update over the air.

- 0: APP firmware of main MCU.
- 1: Auxiliary MCU.
- 2: Network Module.
- 3: Boot firmware of main MCU.

2.1.8.2. BMS (Device Manage System Setting)

The command **AT@BMS** is used to configure the device to report **+BMR** message to the BMS server. BMS server is a separate backend server for device management, firmware OTA and configuration OTA.



➢ **AT@BMS=**

Example: **AT@BMS=at,0,1,255.255.255.255,12345,30,,,1440,1,,,,,,,,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@BMS=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Report Mode	1	0 - 1	0
4	Buffer Mode	1	0 - 2	1
5	BMS Server IP / Domain Name	<=60		
6	BMS Server Port	<=5	0 – 65535	
7	Connection Life	<=3	5 – 300 (sec)	5
8	Reserved	0		
9	Reserved	0		
10	Report Interval	<=5	1 – 2880 (min)	1440
11	BMR Data Mask	<=8	0 – FFFFFFFF	1
12	Serial Number	4	0000 – FFFF	
13	Tail	1	#	#

✧ <*Report Mode*>: This parameter defines the method of communication between the backend server and the device. Supported report modes are as follows:

- 0: Stop mode.
- 1: TCP mode. The device and BMS server will transmit data by TCP protocol.

✧ <*Buffer Mode*>: The working mode of the buffer report function. If the buffer report function is enabled and the device goes into areas without EGPRS network coverage or couldn't receive +SACK message from server, the device will store all reports locally. If the device goes back to areas with EGPRS network coverage, it will then send all the buffered reports through EGPRS.

- 0: Disable buffer function
- 1: Low priority - Enable the buffer report function: In this mode, the device will send the buffered messages after sending real time messages.
- 2: High priority - Enable the buffer report function: In this mode, the device will send the real time messages after sending buffered messages.

✧ <*BMS Server IP / Domain Name*>: The IP address or the domain name of the BMS server.

✧ <*BMS Server Port*>: The port of the BMS server.

✧ <*Connection Life*>: A numeral to indicate the time to maintain TCP connection for receiving commands from the BMS server. If there is no data transmission within the time specified by <*Connection Life*>, the TCP connection will be closed. The unit is second.

✧ <*Report Interval*>: The time interval for sending **+BMR** message. The value range is 1 – 2880. Unit: minute.

✧ <*BMR Data Mask*>: Component mask of the **+BMR/-BMR** message.

Mask Bit	Item
Bit 31	Reserved
…	…
Bit 10	Reserved
Bit 9	BOOT Firmware of Main MCU Mask
Bit 8	Module Version Mask
Bit 7	SIM Info Mask
Bit 6	GNSS Info Mask
Bit 5	IGN State Mask
Bit 4	CAN Info Mask
Bit 3	Config Version Mask
Bit 2	Auxiliary MCU Mask
Bit 1	Battery Level Mask
Bit 0	Main Power Mask

2.1.9.DEBUG

2.1.9.1. DBG (OTA Debug Log)

The command **AT@DBG** is used to configure catch debug log and report to backend server, default TCP connection.

➢ **AT@DBG=**

Example: **AT@DBG=at,1,1,255.255.255.255,12345,,,0,0001#**				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	7	ASCII	AT@DBG=
2	Password	2 - 19	'0' -'9', 'a' -'z', 'A' -'Z'	at
3	Working Mode	1	0 - 1	0
4	Debug Info Mask	8	00000000 – FFFFFFFF	0
5	Server IP / Domain Name	<=60		
6	Server Port	<=5	0 – 65535	0
7	Serial Number	4	0000 – FFFF	
8	Tail	1	#	#

✧ <*Working Mode*>: The working mode of OTA debug log.

- 0: Stop catching log.
- 1: Start catching log.

✧ <*Debug info Mask*>: Component mask of catch logging.

Mask Bit	**Item**
Bit 31	Reserved
…	…
Bit 14	UART Log
Bit 13	Reserved
Bit 12	1wire Log
Bit 11	Harsh Behavior Log
Bit 10	Crash Log
Bit 9	Self-correcting log
Bit 8	Reserved
Bit 7	Network Log
Bit 6	HTTP Log

Bit 5	G-sensor Log
Bit 4	Input and output Log
Bit 3	GPS Log
Bit 2	Report Message Log
Bit 1	Protocol Log
Bit 0	NMEA Log

- <*Server IP / Domain Name*>: The IP address or the domain name of the debug logging server.
- <*Server Port*>: The port of the debug logging server.

2.2. Query Command

The query commands are used to query the current working parameters of the terminal. All query commands are encoded using printable ASCII code, and the character "," is used to separate the adjacent parameter characters, their frame format is:

Example:
AT@SIS?at,,#
AT@NIS?at,,#
AT@GMT?at,,#
AT@GCS?at,,#
AT@RRS?at,,#

No	Parameter	Length (Byte)	Range / Format	Default
1	Header	3	ASCII	AT@
2	Command Key	<=N		
3	Leading Symbol	1	ASCII	?
4	Password	2 - 20	'0' - '9', 'a' - 'z', 'A' - 'Z'	at
5	Reserved	0		
6	Serial Number	4	0000 – FFFF	
7	Tail Character	1	#	#

3. Message Structure

3.1. Message Format

+/-RPT , +/-BMR and +HBD message uses the following data format:

Header Zone				Data Zone	Tail Zone		
Head	Length	IMEI	Device ID	Data	G-time	SN	Tail
5 Bytes	2 Bytes	8 Bytes	1 Byte	Variable	7 Bytes	2 Bytes	1 Byte
ASCII	HEX	HEX	HEX	HEX	HEX	HEX	ASCII

Data Zone is encoded using HEX code.

+SACK , +SHBD message is encoded using ASCII code.

The other messages (+ACK, +QRY, +ALL, +QNI, +VER , +GSV, +LSV, +BSV, +CVS, +CMI)use the following data format:

Header Zone				Data Zone	Tail Zone		
Head	Length	IMEI	Device ID	Data	G-Time	SN	Tail
5 Bytes	2 Bytes	8 Bytes	1 Byte	Variable	7 Bytes	2 Bytes	1 Byte
ASCII	HEX	HEX	HEX	ASCII	HEX	HEX	ASCII

Data Zone is encoded using ASCII code, and the character "," is used to separate the adjacent parameter characters.

3.2. Report Message

‘**+RSP’** is real-time reports, which is generated and sent successfully in 3 minutes.
‘-RSP’ is historical reports, which are saved in buffer.

3.2.1.+RPT/-RPT

This message is the main data generated by different conditions and settings. All of the GNSS related messages and events share the same format.
In order to reduce the frame size and data consumption, all reports frames are uniformly encoded using Hexadecimal codes. And the big-endian byte order is used for multi-byte data types (int, float, etc). For example, for the integer 305419896 (i.e., 12345678H), the byte 12H will be sent first, and then 34H, 56H,78H.

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Header	5	ASCII	+RPT:/-RPT:
2	Length	2		
3	IMEI	8		
4	Device ID	1	0x24	0x24
5	Protocol Version	2		
6	Event Type	1		
7	Event State	1		
8	Data Mask	4		
1	Frame Count	1		**Bit 0:** Multi-Packet
2	Frame ID	1		
1	Network Type	1	0|3|8	**Bit 1:** Network Type
1	Battery Voltage	2	4200 (mV)	**Bit2:** Battery Information
2	Battery Level	1	0 – 100 (%)	
1	GNSS Info Mask	2	0000 - FFFF	**Bit 3:** GNSS Information
2	GNSS number	1	1 - 15	
3	**Bit0 :** GNSS Fixing Type	1	(0,1) | (0,2,3)	
4	**Bit1 :** HDOP	1		
5	**Bit2 :** Speed	2	0 – 999 (Km/h)	
6	**Bit3 :** Azimuth	2	0 - 359	
7	**Bit4 :** Altitude	2		
8	**Bit5 :** Latitude	4		
9	**Bit6 :** Longitude	4		

10	**Bit7:** UTC Time	7	YYYYMMDDHHMMSS	
11	**Bit8:** Satellites Number Mask	1		
12	**Bit8:** GPS Satellite number	1		
13	**Bit8:** BEIDOU Satellite number	1		
14	**Bit8:** GALILEO Satellite number	1		
15	**Bit8:** GLONASS Satellite number	1		
1	MCC	2	0000 – FFFF	**Bit 4:** Cell Information
2	MNC	2	0000 – FFFF	
3	LAC	2	0000 – FFFF	
4	Cell ID	4	00000000 – FFFFFFFF	
5	CSQ	1	0 – 31	
1	Upgrade Code	1		**Bit 7:** Upgrade information
2	Reserved	2		
	Event Data Mask	4		**Bit 8:** Event Data Mask
1	**Bit 0:** Main Power VCC	2	0 – 32000 (mV)	
1	**Bit 1:** Analog Input Number	1	1	
2	**Bit 1:** Analog Input Voltage	2	0 – 16000 (mV)	
1	**Bit 2:** Ignition & Motion State	1		
1	**Bit 3:** Digital Input State	1	0x00 – 0x03	
1	**Bit 4:** Digital Output State	1	0x00 – 0x01	
1	**Bit 5:** Current Mileage	2	0 – 65535 (hm)	

2	**Bit 5:** Total Mileage	4	0 – 42949679 (hm)	
1	**Bit 7:** All GEO Status	4	00000000 - FFFFFFFF	
1	**Bit 8** ID Length	1		
2	**Bit 8:** ID Data	n		
1	**Bit 9:** 1wire Number	1	0 – 4	
2	**Bit 9:** 1wire Sensor ID	8		
3	**Bit 9:** 1wire Data Mask	1	00 – FF	
4	**Bit 9:** 1wire Temperature (Optional)	2	-550 – 1250 (* 0.1℃)	
5	**Bit 9:** 1wire Humidity (Optional)	1	1 – 100 (%)	
1	**Bit 10:** Current Hour Meter Count	3	HHMMSS	
2	**Bit 10:** Total Hour Meter Count	5	HHHHHHMMSS	
1	**Bit 11:** ASC X Forward	1	-100 – 100	
2	**Bit 11:** ASC Y Forward	1	-100 – 100	
3	**Bit 11:** ASC Z Forward	1	-100 – 100	
4	**Bit 11:** ASC X Horizontal	1	-100 – 100	
5	**Bit 11:** ASC Y Horizontal	1	-100 – 100	
6	**Bit 11:** ASC Z Horizontal	1	-100 – 100	
7	**Bit 11:** ASC X Gravity	1	-100 – 100	

8	**Bit 11:** ASC Y Gravity	1	-100 – 100	
9	**Bit 11:** ASC Z Gravity	1	-100 – 100	
1	**Bit 12:** Crash Counter	1	00 – FF	
2	**Bit 12:** ASC Status	1	0 – 2	
3	**Bit 12:** Crash X Value	2	-1311 – 1312	
4	**Bit 12:** Crash Y Value	2	-1311 – 1312	
5	**Bit 12:** Crash Z Value	2	-1311 – 1312	
1	**Bit 14:** Server Protocol Type	1	0	
2	**Bit 14:** Certificate File Type	1	0 - 2	
3	**Bit 14:** Download Code	1	11 | 12 | 21 | 22	
	CAN Info Mask 1	4	0x00000000 – 0xFFFFFFFF	**Bit 9:** Motor CAN Information Mask 1
0	Length of VIN	1		
	VIN	<=17		
1	Ignition Key	1	0|1|2	
2	Total Distance	4	0 – 99999999 (hm)	
3	Total Distance Impulses	4	0 – 2147483648	
4	Total Fuel Used	4	0 – 999999999 (mL)	
5	Vehicle Speed	2	0 – 455 (Km/h)	
6	Engine RPM	2	0 – 16383 (rpm)	
7	Engine Coolant Temperature	2	-40 – +215 (℃)	
8	Fuel Consumption (L/100km)	2	0 – 9999 (*0.1L/100km)	
	Fuel Consumption (L/H)	2	0 – 9999 (*0.05L/H)	
	Fuel Level Liter	1	0 – 256 (L)	

9	Fuel Level Percent	1	0 – 250 (*0.4%)	
10	Range	3	0 – 9999999 (km)	
11	Accelerator Pedal Pressure	1	0 – 250 (*0.4%)	
12	Total Engine Hours	4	0 – 999999999 (s)	
13	Total Driving Time	4	0 – 999999999 (s)	
14	Total Engine Idle Time	4	0 – 999999999 (s)	
15	Total Idle Fuel Used	4	0 – 999999999 (mL)	
16	Axle Weight 1st	2	0 – 65535 (*0.5kg)	
17	Axle Weight 2nd	2	0 – 65535 (*0.5kg)	
18	Axle Weight 3rd	2	0 – 65535 (*0.5kg)	
19	Axle Weight 4th	2	0 – 65535 (*0.5kg)	
20	Detailed Information | Indicators 1	4	00000000 – FFFFFFFF	
21	Detailed Information | Indicators 2	4	00000000 – FFFFFFFF	
22	Lights	1	00 – FF	
23	Doors	1	00 – FF	
24	Total Vehicle Overspeed Time	4	0 – 999999999 (s)	
25	Total Vehicle Engine Overspeed Time	4	0 – 999999999 (s)	
26	Engine Cold Starts Count	3	0 – 16449535	
27	Engine All Starts Count	3	0 – 16449535	
28	Engine Starts by Ignition Count	3	0 – 16449535	
29	Total Engine Cold Running Time	4	0 – 421108120 (s)	
30	Handbrake Applies During Ride Count	2	0 – 64255	
	CAN Info Mask 2	4	0x00000000 – 0xFFFFFFFF	**Bit 10:** Motor CAN Information Mask 2
0	Ad-Blue Level	1	0 – 250 (*0.4%)	
1	Retarder Usage	1	0 – 125 (*1%)	
2	Power Mode	1	0 – FF	
3	Axle Weight 5th	2	0 – 65535 (*0.5kg)	

4	Axle Weight 6th	2	0 – 65535 (*0.5kg)	
5	Axle Weight 7th	2	0 – 65535 (*0.5kg)	
6	Analog Input Value	2	0 – 15000 (mV)	
7	Engine Braking Factor	4	0 – 4278190079	
8	Pedal Braking Factor	4	0 – 4278190079	
9	Total Accelerator Kick-downs	3	0 – 999999	
10	Total Effective Engine Speed Time	4	0 – 999999999 (s)	
11	Total Cruise Control Time	4	0 – 999999999 (s)	
12	Total Accelerator Kick-down Time	4	0 – 999999999 (s)	
13	Total Brake Applications	3	0 – 999999	
14	Oil Temperature	2	-40 – +215 (℃)	
15	Length of Trailer VIN	1		
	Trailer VIN	<=17		
16	Reserved			
17	Reserved			
18	Length of Registration Number	1		
	Registration Number	<=40	'0' – '9' 'a' – 'z' 'A' – 'Z' '-' '_'	
19	Rapid Brakings	3	0 – 16711679	
20	Rapid Accelerations	3	0 – 16711679	
21	Engine Torque	1	-125 – 125 (%)	
22	Service distance	4	-160635 – 327675 (km)	
23	Ambient Temperature	2	-40 – +215 (℃)	
24	DTC Number	1	00 – 99	
	DTC 1	3	'0' – '9' 'a' – 'z' 'A' – 'Z'	
	…			
	DTC N	3	'0' – '9' 'a' – 'z' 'A' – 'Z'	
25	Gaseous Fuel Level Liter	1	0 – 250 (L)	
	Gaseous Fuel Level Percent	1	0 – 250 (*0.4%)	

26	Fuel Level Combustion Liter	1	0 – 250 (L)	
	Fuel Level Combustion Percent	1	0 – 250 (*0.4%)	
27	Total Fuel From Vehicle	4	0 – 999999999 (mL)	
28	Total Gaseous Fuel Usage	4	0 – 999999999 (*0.5kg)	
29	Tyres Count	1	0 - 256	
	Tyres Location	1	00 – FF	
	Tyres Pressure	2	0 - 64250 (*kPa)	
	Tyres Temperature	3	0 – 999999 (*0.01℃)	
	Tyres State	1	00 - FF	
30	Time To Service	2	0000 - FFFF (*day)	
	Electric Info Mask 1	4	0x00000000 – 0xFFFFFFFF	**Bit 11:** Electric CAN Information Mask
0	Battery Instantaneous Voltage	2	0 – 64255 (V)	
1	Battery Charging Cycles Count	2	0 – 64255 (Cycle)	
2	Total Energy Recuperated	4	0 – 4211081215 (Wh)	
3	Battery Level	1	0 – 250 (*0.4%)	
4	Charging State	1	0 – FF	
5	Battery Temperature	2	-40 – +210 (℃)	
6	Battery Charging Current	1	0 – 250 (*0.5A)	
7	Battery Instantaneous Power	4	-327680 – 325110 (W)	
8	Battery State of Health (SoH)	1	0 – 250 (*0.4%)	
9	Total Energy Used	4	0 – 4211081215 (Wh)	
10	Total Energy Used When Idling	4	0 – 4211081215 (Wh)	
11	Total Energy Charged	4	0 – 4211081215 (Wh)	
12	Only Battery Charge Level	1	0 – 200 (*0.5%)	
	Tachograph Driver1 Working Time Mask	4	0x00000000 – 0xFFFFFFFF	**Bit 13:** Tachograph Driver1 Working Time Mask
0	Drv1 End Of Last Daily Rest Period	6	YYYYMMDDHHMM	
1	Drv1 End Of Last Weekly Rest Period	6	YYYYMMDDHHMM	

2	Drv1 End Of Second Last Weekly Rest Period	6	YYYYMMDDHHMM	
3	Drv1 Maximum Daily Period	1	00 – FF (Hour)	
4	Drv1 Number Of Times 9h Daily Driving Times Exceeded	1	00 – FF	
5	Drv1 Number Of Used Reduced Daily Rest Periods	1	00 – FF	
6	Drv1 Remaining Current Drive Time	2	0000 – FFFF (min)	
7	Drv1 Remaining Time Until Next Break Or Rest	2	0000 – FFFF (min)	
8	Drv1 Duration Of Next Break Rest	2	0000 – FFFF (min)	
9	Drv1 Remaining Time Of Current Break Rest	2	0000 – FFFF (min)	
10	Drv1 Time Left Until Next Driving Period	2	0000 – FFFF (min)	
11	Drv1 Duration Of Next Driving Period	2	0000 – FFFF (min)	
12	Drv1 Remaining Driving Time On Current Shift	2	0000 – FFFF (min)	
13	Drv1 Time Left Until New Daily Rest Period	2	0000 – FFFF (min)	
14	Drv1 Minimum Daily Rest	2	0000 – FFFF (min)	
15	Drv1 Remaining Driving Time Of Current Week	2	0000 – FFFF (min)	
16	Drv1 Time Left Until New Weekly Rest Period	2	0000 – FFFF (min)	
17	Drv1 Minimum Weekly Rest	2	0000 – FFFF (min)	
18	Drv1 Open Compensation In The Last Week	2	0000 – FFFF (min)	
19	Drv1 Open Compensation In Week Before Last	2	0000 – FFFF (min)	
20	Drv1 Open Compensation In 2nd Week Before Last	2	0000 – FFFF (min)	

21	Drv1 Continuous Driving Time	2	0000 – FFFF (min)	
22	Drv1 Cumulative Break Time	2	0000 – FFFF (min)	
23	Drv1 Current Duration Of Selected Activity	2	0000 – FFFF (min)	
24	Drv1 Accumulated Driving Time Previous And Current Week	2	0000 – FFFF (min)	
25	Drv1 Current Daily Driving Time	2	0000 – FFFF (min)	
26	Drv1 Current Weekly Driving Time	2	0000 – FFFF (min)	
27	Drv1 Cumulative Uninterrupted Rest Time	2	0000 – FFFF (min)	
28	Drv1 Maximum Daily Driving Time	2	0000 – FFFF (min)	
	Tachograph Driver2 Working Time Mask	4	0x00000000 – 0xFFFFFFFF	**Bit 14:** Tachograph Driver2 Working Time Mask
0	Drv2 End Of Last Daily Rest Period	6	YYYYMMDDHHMM	
1	Drv2 End Of Last Weekly Rest Period	6	YYYYMMDDHHMM	
2	Drv2 End Of Second Last Weekly Rest Period	6	YYYYMMDDHHMM	
3	Drv2 Maximum Daily Period	1	00 – FF	
4	Drv2 Number Of Times 9h Daily Driving Times Exceeded	1	00 – FF	
5	Drv2 Number Of Used Reduced Daily Rest Periods	1	00 – FF	
6	Drv2 Remaining Current Drive Time	2	0000 – FFFF (min)	
7	Drv2 Remaining Time Until Next Break Or Rest	2	0000 – FFFF (min)	
8	Drv2 Duration Of Next Break Rest	2	0000 – FFFF (min)	
9	Drv2 Remaining Time Of Current Break Rest	2	0000 – FFFF (min)	

10	Drv2 Time Left Until Next Driving Period	2	0000 – FFFF (min)	
11	Drv2 Duration Of Next Driving Period	2	0000 – FFFF (min)	
12	Drv2 Remaining Driving Time On Current Shift	2	0000 – FFFF (min)	
13	Drv2 Time Left Until New Daily Rest Period	2	0000 – FFFF (min)	
14	Drv2 Minimum Daily Rest	2	0000 – FFFF (min)	
15	Drv2 Remaining Driving Time Of Current Week	2	0000 – FFFF (min)	
16	Drv2 Time Left Until New Weekly Rest Period	2	0000 – FFFF (min)	
17	Drv2 Minimum Weekly Rest	2	0000 – FFFF (min)	
18	Drv2 Open Compensation In The Last Week	2	0000 – FFFF (min)	
19	Drv2 Open Compensation In Week Before Last	2	0000 – FFFF (min)	
20	Drv2 Open Compensation In 2nd Week Before Last	2	0000 – FFFF (min)	
21	Drv2 Continuous Driving Time	2	0000 – FFFF (min)	
22	Drv2 Cumulative Break Time	2	0000 – FFFF (min)	
23	Drv2 Current Duration Of Selected Activity	2	0000 – FFFF (min)	
24	Drv2 Accumulated Driving Time Previous And Current Week	2	0000 – FFFF (min)	
25	Drv2 Current Daily Driving Time	2	0000 – FFFF (min)	
26	Drv2 Current Weekly Driving Time	2	0000 – FFFF (min)	
27	Drv2 Cumulative Uninterrupted Rest Time	2	0000 – FFFF (min)	
28	Drv2 Maximum Daily Driving Time	2	0000 – FFFF (min)	
	Tachograph Common Event Mask	4	00000000 – FFFFFFFF	**Bit15:** Tachograph Common

0	**Bit 0:** Request ID	2	0000 – FFFF	Event Information Mask
1	**Bit 1:** Reply Result	1	00 – FF	
2	**Bit 2:** Device Status	1	00 – FF	
3	**Bit 3:** File State In Memory	1	00 – FF	
4	**Bit 4:** Error Code	1	00 – FF	
5	**Bit 5:** Error Code Memory	1	00 – FF	
6	**Bit 6:** APDU Serial Number	2	0000 – FFFF	
7	**Bit 6:** APDU Data Length	1	00 – FF	
8	**Bit 6:** APDU Data	<=N		
9	**Bit 7:** Tachograph Producer	1		
10	**Bit 7:** Tachograph Model Length	1		
11	**Bit 7:** Tachograph Model	<=N		
12	**Bit 8:** File In Device Memory	1	00 – FF	
13	**Bit 9:** Tachograph driver time related states	1	00 – FF	
14	**Bit 9:** Tachograph driver 1 id Length	1		
15	**Bit 9:** Tachograph driver 1 id	<= N		
16	**Bit 9:** Tachograph driver 1 name and surname Length	1		

17	**Bit 9:** Tachograph driver 1 name and surname	<= N		
18	**Bit 10:** Tachograph driver time related states	1	00 – FF	
19	**Bit 10:** Tachograph driver 2 id Length	1		
20	**Bit 10:** Tachograph driver 2 id	<= N		
21	**Bit 10:** Tachograph driver 2 name and surname Length	1		
22	**Bit 10:** Tachograph driver 2 name and surname	<= N		
23	**Bit 11:** CAN1-bus Status	1	0 – 3	
24	**Bit 11:** CAN2-bus or J1708-bus Status	1	0 – 3	
25	**Bit 11:** Check Information	1		
	Tachograph Info Mask	4	00000000 – FFFFFFFF	**Bit16:** Tachograph Information Mask
0	Tachograph Information	4	00000000 – FFFFFFFF	
1	Tachograph Overspeed Signal	1	0|1	
2	Tachograph Vehicle Motion Signal	1	0|1	
3	Tachograph Driving Direction	1	0|1	
4	Length of Tachograph Driver 1 Card Number	1		
	Tachograph Driver 1 Card Number	<=40	'0' – '9' 'a' – 'z' 'A' – 'Z' '-' '_'	
5	Length of Tachograph Driver 2 Card Number	1		
	Tachograph Driver 2 Card Number	<=40	'0' – '9' 'a' – 'z' 'A' – 'Z' '-' '_'	

6	Length of Tachograph Driver 1 Name	1		
	Tachograph Driver 1 Name	<=40	'0' – '9' 'a' – 'z' 'A' – 'Z' '-' '_'	
7	Length of Tachograph Driver 2 Name	1		
	Tachograph Driver 2 Name	<=40	'0' – '9' 'a' – 'z' 'A' – 'Z' '-' '_'	
8	Real Time Clock Date and Time	7	YYYYMMDDHHMMSS	
	Special Car Info Mask	4	00000000 – FFFFFFFF	**Bit17:** Special Car Information Mask
0	**Bit 0: e-GENSET Info** Hours Until Service	2	0 – 65535 (h)	
	Bit 0: System Status Health	2	0000 – FFFF	
	Bit 0: Status Inverter Mode	1	0|1|2|8|16	
	Bit 0: Remaining Runtime	1	0 – 255 (h)	
1	**Bit 1: Ultima Info** Ultima F1	1	00 – FF	
	Bit 1: Ultima F2	1	00 – FF	
	NMEA2000 Info Mask	4	00000000 – FFFFFFFF	**Bit18:** NMEA2000 Information Mask
0	**Bit 0:** Depth Blow Transducer	4	00000000 – FFFFFFFF (0.01m)	
1	**Bit 1:** Heading	2	0000 – FFFF (0.0001 rad)	
2	**Bit 2:** Speed Through Water	2	0000 – FFFF (0.01 m/s)	
3	**Bit 3:** Wind Angle	2	0000 – FFFF (0.0001 rad)	
4	**Bit 4:** Wind Speed	2	0000 – FFFF (0.01 m/s)	
	BLE Sensor Number	1	0 – 128	**Bit19:** BLE Information Mask
	MAC Address	6		

	BLE RSSI	1		
	BLE Info Mask	2	0000 – FFFF	
0	BLE Battery Voltage	2		
1	BLE Temperature	2		
2	BLE Humidity	2		
3	BLE Door State	1		
4	BLE Luminous	2		
5	BLE Fuel	2		
6	BLE Fuel Frequency	4		
7	Custom1 Length	1		
	Custom1 Data	N		
8	Custom2 Length	1		
	Custom2 Data	N		
9	Custom3 Length	1		
	Custom3 Data	N		
10	Custom4 Length	1		
	Custom4 Data	N		
11	Custom5 Length	1		
	Custom5 Data	N		
12	Custom6 Length	1		
	Custom6 Data	N		
1	RS485 Accessory Number	1	0 - 5	**Bit21** RS485 Information Mask
2	RS485 Accessory Model	1	1	
3	RS485 Accessory Address	1	1 - 255	
4	RS485 Data Mask	2		
5	Volume	2	0 - 65535	
6	Temperature	2	-400 – 850 (* 0.1℃)	
1	Generated Time	7	YYYYMMDDHHMMSS	
2	Serial Number	2	0000 – FFFF	
3	Tail Character	1	#	#

✧

<*IMEI*>: the IMEI of the device is used as the unique ID of the device. IMEI is a 15-digit string. In the HEX format message, each 2 digits are encoded into one byte as an integer.

IMEI	13	57	90	24	68	11	22	5
HEX	0D	39	5A	18	44	0B	16	05

✧ <*Device ID*>: The ID of the terminal project.

✧ <*Protocol version*>: First byte indicates the major version number of the protocol. the last byte indicates the minor version number of the protocol, for example: **0x0B01** means version **11.01**.

✧ <*Event Type*>: The type of event trigger message.

Event Type	Item
0	Terminal Power Event
1	Regular Report Event
2	Moving Event
3	Battery Alarm Event
4 ~ 6	Reserved
7	Crash Alarm Event
8 – 10	Reserved
11	Upgrade Firmware Event
12	External Temperature Alarm Event
13	Ignition Event
14	Main Power Alarm Event
15	TOW Alarm Event
16	Over Speed Alarm Event
17	Engine Idle Alarm Event
18	Start Stop Alarm Event
19	Harsh Behavior Alarm Event
20	G-Sensor Self-Calibration Event
21	Geo Fence Alarm Event
22	Digital Input 1 Event
23	ID Authorized Event
24	CANBUS Info Event
25	Tachograph Common Event

26	Tachograph DDD File Upload Event
27	Output Alarm Event
28	Delta AIS Alarm Event
29	PDP Alarm Event
30	BLE Common info Event
31	External Humidity Alarm Event
32	Jamming Alarm Event
35	TLS Certificate File Download Event

✧ <*Event State*>: The state of event trigger message.

- 0: Nothing.
- Others, please refer to the following event type corresponding event state:

(**Event Type 0)** Terminal Power Event

Event State	Item
1	Power on with normal
2	Power on with abnormal
3	Power off with RTO Reboot
4	Power off with RTO power-off
5	Power off with Battery Low
10	Power on with upgrade firmware

(**Event Type 1)** Regular Report Event:

Event State	Item
0	Regular Report

(**Event Type 2)** Moving Event:

Event State	Item
1	Start Moving
2	Stop Moving

(**Event Type 3)** Battery Alarm Event:

Event State	Item
1	Low Battery Alarm
2	Battery Start Charge

3	Battery Stop Charge
4	Battery Full Charge

(**Event Type 11)** Upgrade Firmware Event:

Event State	Item
1	Main MCU Upgrade
2	Auxiliary MCU upgrade
3	CAN Firmware Upgrade
4	CAN Configuration Upgrade
5	Network Module Upgrade
6	BOOT Firmware of Main MCU Upgrade

(**Event Type 12)** External Temperature Alarm Event:

Event state data zone consists of sensor type and alarm type. First half byte is sensor type, second half byte is alarm type.

Sensor Type	Item	Alarm Type	Item
0	Reserved	0	Temperature alarm cancel
1	1wire Sensor	1	Report the temperature alarm event of +RPT message when the current temperature is more or equal than the temperature specified by <*High Temperature*>.
2	BLE Sensor	2	Report the temperature alarm event of +RPT message when the current temperature is less than the temperature specified by <*Low Temperature*>.
3	.....	3	Report the temperature alarm event of +RPT message when the current temperature is inside the temperature range.

(**Event Type 13)** Ignition Event:

Event State	Item
1	Wired ignition off
2	Wired ignition on
3	Virtual ignition off
4	Virtual ignition on

(**Event Type 14)** Main Power Alarm Event:

Event State	Item
1	Disconnect the main power supply
2	Connect the main power supply
3	The low main power alarm

(**Event Type 15)** TOW Alarm Event:

Event State	Item
1	Enter TOW State
2	Quit TOW State

(**Event Type 17)** Engine Idle Alarm Event:

Event State	Item
1	Enter idling status
2	Quit idling status

(**Event Type 18)** Start Stop Alarm Event:

Event State	Item
1	Start Status
2	Stop Status
3	Long Stop Status

(**Event Type 19)** Harsh Behavior Alarm Event:

Event State	Item
1	Harsh braking behavior
2	Harsh acceleration behavior
3	Harsh cornering behavior
4	Harsh braking and cornering behavior
5	Harsh acceleration and cornering behavior

(**Event Type 21)** Geo Fence Alarm Event:

Event state data zone consists of GEO trigger id and GEO trigger state. Bit 1 to Bit 7 is GEO trigger id, bit 0 is GEO trigger state.

Bit 1 ~ Bit 7 GEO Trigger ID	Item	Bit 0 GEO Trigger State	Item
0	GEO ID 1	0	Quit GEO-Fence
1 ……	GEO ID 2 ……	1	Enter GEO-Fence

(**Event Type 22)** Digital Input 1 Event:

Event State	Item
1	Digital Input 1 On
2	Digital Input 1 Off

(**Event Type 23)** ID Authorized Event:

Event State	Item
1	ID Authorized Disable
2	ID Unauthorized
3	ID Authorized
4	ID Authorized-timeout
5	ID Authorized-logout

(**Event Type 24)** CANBUS Info Event:

Event state data zone consists of CAN info report event and CANBUS sync state. First half byte is CAN info report event, second half byte is CANBUS sync state.

CAN info report event	Item	CANBUS sync state	Item
0	Sync State Change Report	0	Sync just started (ongoing)
1	Periodic Report	1	Sync success
2	RTO CAN Report	2	Sync failed (connect but have not been recognized)
3	Ignition Event Report	3	Sync failed (disconnect or ignition off)
4	Tacho Drv1 Card Changed	4	Sync not allowed
5	Tacho Drv2 Card Changed	5	Sync Unknown

6	Tacho Drv1 State Changed	6	Reserved
7	Tacho Drv2 State Changed	7	Reserved
8	Tacho Out of Scope Changed	8	Reserved
9	Ultima F2 State Changed	9	Reserved

- (**Event Type 25)** Tachograph Common Event:

Event State	Item
1	Reply for DDD file request
2	Authorization result
3	CAN logistic APDU data result
4	The result process of read file from tachograph
5	Communication timeout
6	the state of CAN logistic debug mode
7	FTP transfer event
8	Reserved
9	Reserved
10	Tachograph details
11	Inquiry status of device
12	Inquiry tachograph driver 1 ID, name and surname
13	Inquiry tachograph driver 2 ID, name and surname
14	Reserved
15	Reply setting tachograph file format
16	Reply changing the type of communication with the tachograph.

- **(Event Type 26) Tachograph DDD File Upload Event:**

Event State	Item
1	Start upload DDD file
2	Upload DDD file success
3	Upload DDD file failed

- **(Event Type 30) BLE Common info Event:**

Event State	Item
0	Regular Report
1	Door State Change Event

- **(Event Type 31) External Humidity Alarm Event:**

Event state data zone consists of sensor type and alarm type. First half byte is sensor type, second half byte is alarm type.

Sensor Type	Item	Alarm Type	Item
0	Reserved	0	Humidity alarm cancel
1	1wire Sensor	1	Report the humidity alarm event of **+RPT** message when the current humidity is more or equal than the humidity specified by <*High Humidity*>.
2	BLE Sensor	2	Report the humidity alarm event of **+RPT** message when the current humidity is less than the humidity specified by <*Low Humidity*>.
3	.....	3	Report the humidity alarm event of **+RPT** message when the current humidity is inside the humidity range.

- **(Event Type 32) Jamming Alarm Event:**

Event State	Item
1	Jamming On
2	Jamming Off

✧ <*Data Mask*>: Please refer to the <*Data Mask*> in AT@GCS command.

✧ <*Frame Count*>: It indicates the number of reports generated by dividing the message. When the length of the message is too long, the terminal will automatically split it into multiple reports to send to the backend server.

✧ <*Frame ID*>: It indicate the sequence of the current multiple reports.

✧ <*Network Type*>: Current network type.

- 0: EGPRS
- 3: CAT-1
- 8: Unknow

✧ <*Battery Voltage*>: Voltage of battery, the unit is mV.

✧ <*Battery Level*>: Battery Percent Information, the unit is 1%.

✧ <*GNSS Info Mask*>: This is GNSS information mask, control which part of GNSS data report.

Mask Bit	Item
Bit 15	Reserved
…	…
Bit 8	GNSS Satellites Number
Bit 7	UTC Time
Bit 6	Longitude
Bit 5	Latitude
Bit 4	Altitude
Bit 3	Azimuth
Bit 2	Speed
Bit 1	HDOP
Bit 0	Fix Type

- <*GNSS Number*>: the number of GNSS information report.
- <*GNSS Fixing Type*>: GNSS fixing type consists of GNSS generated type and fixing result. First half byte is GNSS generated type, second half byte is fixing result.

GNSS Generated Type	Item	Fixing Result	Item
0	Periodic Fix	0	No fixed
1	Corner Turning	1	Reserved
2	Reserved	2	2D fixed type.
3	Reserved	3	3D fixed type.

- <*HDOP*>: The current HDOP of GNSS, it expresses GNSS fixing accuracy, the smaller the value, the higher the positioning accuracy.
  - 1 - 2: The satellite distribution is good, and the horizontal positioning accuracy is high.
  - 3 - 5: The satellite distribution is general, and the horizontal positioning accuracy is moderate.
  - 6 - 8: The satellite distribution is poor and horizontal positioning accuracy is low.
  - >=9: The satellite distribution is very poor, and the horizontal positioning accuracy is very low.
- <*Speed*>: The current speed. The unit is km/h.
- <*Azimuth*>: The azimuth of the GNSS fix.
- <*Altitude*>: The altitude from GNSS. If the altitude is negative, it is represented in 2's complement format, the unit is meter.
- <*Latitude*>: The latitude of the current position. Total 4 bytes. The device converts the latitude to an integer with 6 implicit decimals and reports this integer in HEX format. If the value of the latitude is negative, it is represented in 2's complement format.

Latitude 31.164503	31164503			
HEX	01	DB	88	57

✧ <*Longitude*>: The longitude of the current position. 4 bytes in total. The device converts the longitude to an integer with 6 implicit decimals and reports this integer in HEX format. If the value of the longitude is negative, it is represented in 2's complement format.

Longitude 121.390847	121390847			
HEX	07	3C	46	FF

✧ <*GNSS UTC Time*>: UTC time is the GNSS fixing time. 7 bytes in total. The first 2 bytes are for year, and the other 5 bytes are for month, day, hour, minute and second respectively.

GNSS UTC Time	2011		07	14	08	24	13
HEX	07	DB	07	0E	08	18	0D

✧ <*MCC*>: Mobile country code.

✧ <*MNC*>: Mobile network code.

✧ <*LAC*>: Location area code.

✧ <*Cell ID*>: Cell ID of base station.

✧ <*CSQ*>: The signal strength level.

CSQ RSSI	Signal Strength (dBm)
0	<-113
1	-111
2 - 30	-109 - -53
31	>-51
99	Unknown

✧ <*Upgrade Code*>: The code of the upgrade status.

- 0: The device prepares to download the package.
- 1: The device starts downloading the package.
- 2: The device finishes downloading the package successfully.
- 3: The firmware package download failed.
- 4: The device refuse to start the downloading process because the battery is low.
- 11: The device starts upgrading the firmware.
- 12: The device finishes upgrading the firmware successfully.
- 13: The firmware update is failed.
- 14: The upgrade process has been postponded because the battery is low.

✧ <*Main Power VCC*>: The voltage of the main power supply.

✧ <*Analog Input Number*>: The number of the analog input.

✧ <*Analog Input Voltage*>: The voltage of the analog input.

✧ <Ignition & Motion State>: The state of ignition and motion, please refer to the following:

- 0x11 (Ignition Off & Rest): The device status is ignition off and motionless.
- 0x12 (Ignition Off & Motion): The device status is ignition off and moving before it is considered as being towed.
- 0x13 (Tow): The device status is ignition off and it is being towed.
- 0x21 (Ignition On & Rest): The device status is ignition on and motionless.

- 0x22 (Ignition On & Motion): The device status is ignition on and moving.

✧ <Digital Input State>: The status of ignition detection and digital input1

Digital Input State	
Bit 0	Ignition Detection
Bit 1	Digital Input1

✧ <Digital Output State>: The status mask of digital output1.

Digital Output State	
Bit 0	Digital Output1

✧ <*Current Mileage*>: It's the mileage traveled of current journey. The unit is hectometer.

✧ <*Total Mileage*>: It's the total mileage traveled. The unit is hectometer.

✧ <*Satellites Number Mask*>: The mask of satellites number, it consists of GPS, GLONASS, BEIDOU, GALILEO Satellites.

Satellites Number Mask	Description
Bit 0	GPS Satellite number
Bit 1	BEIDOU Satellite number
Bit 2	GALILEO Satellite number
Bit 3	GLONASS Satellite number

✧ <*ALL GEO Status*>: The bitwise mask for trigger condition composition of the corresponding GEO ID. Each bit from bit 0 to bit 31 represents the logical state of the corresponding GEO ID to trigger the entering or exiting event. 1 means "The event of the GEO ID set is triggered", and 0 means "The event of the GEO ID set is not triggered".
For example, if the Area Mask is 03, it means entering or exiting events of GEO-ID0 and GEO-ID1 occur at the same time.

✧ <*ID Length*>: The length of RFID ID.

✧ <*ID Data*>: The data of RFID ID.

✧ <*1wire Number*>: The total amount of 1wire accessories connected, it Contains multiple groups of <*1wire Sensor ID*>, <*1wire Data Mask*>, <*1wire Temperature*> and <*1wire Humidity*>.

✧ <*1wire Sensor ID*>: The sensor id of 1wire sensor, for example DS18B20, TH200 etc. the 1wire sensor ID is HEX format.

✧ <*1wire Data Mask*>: It refers to the <*1wire Data Mask*> parameter of the AT@WAS command.

✧ <*1wire Temperature*>: The temperature value of 1wire sensor, it's controlled by <*1wire Data Mask*>.

✧ <*1wire Humidity*>: The Humidity value of 1wire sensor, it's controlled by <*1wire Data Mask*>.

✧ <*Current Hour Meter Count*>: it is the engine on time of the current trip. 3 bytes in total, the first byte represents the hour part, the second byte represents the minute part, and the third byte represents the second part.

Current Hour Meter Count	99	59	59
HEX	63	3B	3B

✧ <*Total Hour Meter Count*>: It is the total engine on time. 5 bytes in total, the first 3 bytes

represent the hour part, the fourth byte represents the minute part, and the fifth byte represents the second part.

Total Hour Meter Count	**99999**			**59**	**59**
HEX	**01**	**86**	**9F**	**3B**	**3B**

- <*ASC X Forward*>, <*ASC Y Forward*>, <*ASC Z Forward*>: These parameters are used to calculate the new acceleration in forward direction.
- <*ASC X Horizontal*>, <*ASC Y Horizontal*>, <*ASC Z Horizontal*>: These parameters are used to calculate the new acceleration in horizontal direction.
- <*ASC X Gravity*>, <*ASC Y Gravity*>, <*ASC Z Gravity*>: These parameters are used to calculate the new acceleration in gravity direction.
- <*Crash Counter*>: It indicates how long time the crash happens. It circles from 0x00 to 0xFF.
- <*ASC Status*>: The status of the self-calibration status.
  - 1: ASC Doing.
  - 2: ASC Finish.
- <*Crash X Value*>: It indicates X axis value when crash event happens.
- <*Crash Y Value*>: It indicates Y axis value when crash event happens.
- <*Crash Z Value*>: It indicates Z axis value when crash event happens.
- <*Server Protocol Type*>: The type of communication protocol using for obtain data from the server.
  - 0: HTTP.
- <*Certificate File Type*>: It expresses the type of certificate file to download from the server.
  - 0: CA certificate.
  - 1: Client certificate.
  - 2: Client Key.
- <*Download Code*>: The code of the download TLS certificate status
  - 11 – start to download the certificate file.
  - 12 – download command error
  - 21 – download success.
  - 22 – download fail.
- <*CAN Info Mask 1*>: Please refer to <*CAN Info Mask 1*> parameter in **AT@CAN**.
- <*Length of VIN*>: The length of vehicle identification number.
- <*VIN*>: Vehicle identification number, it's ASCII type.
- <*Ignition Key*>: A numeral to indicate the ignition status.
  - 0: Ignition off
  - 1: Ignition on
  - 2: Engine on
- <*Total Distance*>: Vehicle total distance, this number is always increasing, the unit is hectometer (Hm).
- <*Total Distance* Impulses >: Vehicle distance in impulses, this number is always increasing, the unit is imp. If it is set to 0, the distance in imp will not be available.
- <*Total Fuel Used*>: The number of liters of fuel used since vehicle manufacture or device installation, the unit is Milliliter.
- <*Vehicle Speed*>: The vehicle speed based on wheel, the unit is km/h.
- <*Engine RPM*>: The revolutions per minute, the unit is rpm.

- <*Engine Coolant Temperature*>: the temperature of engine coolant, the unit is Celsius.
- <*Fuel Consumption (L/100km)*>: The fuel consumption is calculated based on values read from vehicle, the unit is L/100Km.
- <*Fuel Consumption (L/H)*>: The fuel consumption is calculated based on values read from vehicle, the unit is L/H.
- <*Fuel Level Liter*>: The level of fuel in vehicle tank, the unit is liter.
- <*Fuel Level Percent*>: The level of fuel in vehicle tank, the unit is 0.4%.
- <*Range*>: The mileage of drive on remaining fuel, the unit is kilometer.
- <*Accelerator Pedal Pressure*>: The unit is 0.4%.
- <*Total Engine Hours*>: Time of engine running since vehicle manufacture or device installation., the unit is second.
- <*Total Driving Time*>: Time of engine running (non-zero speed) since vehicle manufacture or device installation, the unit is second.
- <*Total Engine Idle Time*>: Time of engine running during idling status (vehicle at rest) since vehicle manufacture or device installation, the unit is second.
- <*Total Idle Fuel Used*>: The number of liters of fuel used since vehicle manufacture or device installation, the unit is milliliter.
- <*Axle Weight 1st*>: Vehicle first axle weight, the unit is 0.5kg.
- <*Axle Weight 2nd* >: Vehicle second axle weight, the unit is 0.5kg.
- <*Axle Weight 3rd*>: Vehicle third axle weight, the unit is 0.5Kg.
- <*Axle Weight 4th*>: Vehicle fourth axle weight, the unit is 0.5Kg.
- <*Axle Weight 5th*>: Vehicle fifth axle weight, the unit is 0.5Kg.
- <*Axle Weight 6th*>: Vehicle sixth axle weight, the unit is 0.5Kg.
- <*Axle Weight 7th*>: Vehicle seventh axle weight, the unit is 0.5Kg.
- <*Detailed Information / Indicators 1*>: Four bytes. Each bit contains information of one indicator.
    - Bit 0: FL – fuel low indicator (1 – indicator on, 0 – indicator off)
    - Bit 1: DS – driver seatbelt indicator (1 – indicator on, 0 – indicator off)
    - Bit 2: AC – air conditioning (1 – on, 0 – off)
    - Bit 3: CC – cruise control (1 – active, 0 – disabled)
    - Bit 4: B – brake pedal (1 – pressed, 0 – released)
    - Bit 5: C – clutch pedal (1 – pressed, 0 – released)
    - Bit 6: H – handbrake (1 – pulled-up, 0 – released)
    - Bit 7: CL – central lock (1 – locked, 0 – unlocked)
    - Bit 8: R – reverse gear (1 – on, 0 – off)
    - Bit 9: RL – running lights (1 – on, 0 – off)
    - Bit 10: LB – low beams (1 – on, 0 – off)
    - Bit 11: HB – high beams (1 – on, 0 – off)
    - Bit 12: RFL – rear fog lights (1 – on, 0 – off)
    - Bit 13: FFL – front fog lights (1 – on, 0 – off)
    - Bit 14: BFL – brake fluid low indicator (1 – on, 0 – off or not available)
    - Bit 15: CLL – coolant level low indicator (1 – on, 0 – off or not available)
    - Bit 16: BAT – battery indicator (1 – on, 0 – off or not available)
    - Bit 17: BF – brake system failure indicator (1 – on, 0 – off or not available)

- Bit 18: OP – oil pressure indicator (1 – on, 0 – off or not available)
- Bit 19: EH – engine hot indicator (1 – on, 0 – off or not available)
- Bit 20: ABS – ABS failure indicator (1 – on, 0 – off or not available)
- Bit 21: EPS – EPS failure indicator (1 – on, 0 – off or not available)
- Bit 22: CHK – "check engine" indicator (1 – on, 0 – off or not available)
- Bit 23: AIR – airbag indicator (1 – on, 0 – off or not available)
- Bit 24: SC – service call indicator (1 – on, 0 – off or not available)
- Bit 25: OLL – oil level low indicator (1 – on, 0 – off or not available)
- Bit 26: CHG –battery charging for electric cars (1 – battery is being charged, 0 – no charging)
- Bit 27: FS – fuel source for vehicles equipped with factory gas installation (1 – engine powered by gas, 0 – engine powered by petrol)
- Bit 28: BLW – brake linings worn off indicator (1 – indicator on, 0 – indicator off)
- Bit 29: DPF – DPF burning lock (1 – lock on, 0 – lock off)
- Bit 30: HAZ – Hazard Lights (1 – indicator on, 0 – indicator off)
- Bit 31: ACC – Accelerator pedal in "kick-down" position (1 – accelerator pedal in „kick-down" position, 0 – accelerator pedal in operating range (kick-down switch off))

✧ <*Detailed Information / Indicators 2*>: Four bytes. Each bit contains information of one indicator.

- Bit 0: TPMS – Tire Pressure Monitoring System indicator (1 – indicator on, 0 – indicator off)

✧ <*Lights*>: A hexadecimal number. Each bit contains information of one type of light.

- Bit 0: Running Lights (1 – on, 0 – off)
- Bit 1: Low Beam (1 – on, 0 – off)
- Bit 2: High Beam (1 – on, 0 – off)
- Bit 3: Front Fog Light (1 – on, 0 – off)
- Bit 4: Rear Fog Light (1 – on, 0 – off)
- Bit 5: Hazard Lights (1 – on, 0 – off)
- Bit 6: Reserved
- Bit 7: Reserved

✧ <*Doors*>: A hexadecimal number. Each bit contains information of one door.

- Bit 0: Driver Door (1 – open, 0 – closed)
- Bit 1: Passenger Door (1 – open, 0 – closed)
- Bit 2: Rear Left Door (1 – open, 0 – closed)
- Bit 3: Rear Right Door (1 – open, 0 – closed)
- Bit 4: Trunk (1 – open, 0 – closed)
- Bit 5: Hood (1 – open, 0 – closed)
- Bit 6: Reserved
- Bit 7: Reserved

✧ <*Total Vehicle Overspeed Time*>: The total time when the vehicle speed is greater than the limit defined in CAN Module's configuration, the unit is second.

✧ <*Total Vehicle Engine Overspeed Time*>: The total time when the vehicle engine speed is greater than the limit defined in CAN Module's configuration, the unit is second.

✧ <*Engine Cold Starts Count*>: the Number of cold Engine starts.

- <*Engine All Starts Count*>: Total number of Engine starts.
- <*Engine Starts by Ignition Count*>: Total number of Engine starts by ignition.
- <*Total Engine Cold Running Time*>: Total driving time with cold engine (engine coolant temperature below 70°C, the unit is second.
- <*Handbrake Applies During Ride Count*>: Counts events when handbrake is pulled-up while driving (speed is greater than 5 km/h).
- <*CAN Info Mask 2*>: Please refer to <*CAN Info Mask 2*> parameter in **AT@CAN**.
- <*Ad-Blue Level*>: The level of Ad-Blue, the unit is 0.4%.
- <*Retarder Usage*>: The usage of Retarder, the unit is 1%.
- <*Power Mode*>: The power mode.

MSB									LSB
	R	R	PTO1	PTO0	R	R	R	R	

R – Reserved for future use (set to 1)

PTO1:PTO0 – Power Take-Off

  - b'00' – PTO not engaged
  - b'01' – PTO engaged
  - b'10' – unknown state of PTO (not compliant with FMS 3.0 standard)
  - b'11' – parameter currently not available

- <*Analog Input Value*>: The value of analog input, the unit is mV.
- <*Engine Braking Factor*>: It measures how often driver brakes with brake pedal or with engine and stores both counts (always increasing). Decreasing speed with no pedal pressed causes an increase in engine braking factor.
- <*Pedal Braking Factor*>: It measures how often driver brakes with brake pedal or with engine and stores both counts (which are always increasing). Decreasing speed with brake pedal pressed causes an increase in pedal braking factor.
- <*Total Accelerator Kick-downs*>: The count of accelerator pedal kick-downs (with the pedal pressed over 90%).
- <*Total Effective Engine Speed Time*>: Total time when the vehicle engine speed is effective, the unit is second.
- <*Total Cruise Control Time*>: Total time when vehicle speed is controlled by cruise-control module, the unit is second.
- <*Total Accelerator Kick-down Time*>: Total time when accelerator pedal is pressed over 90% the unit is second.
- <*Total Brake Applications*>: The total number of braking processes initiated by brake pedal.
- <*Oil Temperature*>: the temperature of oil, the unit is Celsius.
- <*Length of Trailer VIN*>: The length of Trailer identification number.
- <*Trailer VIN*>: Trailer identification number, it's ASCII type.
- <*Length of Registration Number*>: The length of Registration Number.
- <*Registration Number*>: The vehicle registration number, it's ASCII type.
- <*Rapid Brakings*>: The number of total rapid brakings since installation (calculation based on CAN 100 settings of speed decrease time and value).
- <*Rapid Accelerations*>: The number of total rapid accelerations since installation (calculation based on CAN 100 settings of speed increase time and value).
- <*Engine Torque*>: The engine torque, offset -125%, operating range -125% to 125%, the unit is 1%.

- ✧ <*Service distance*>: The parameter describes distance left to diagnostic car review, the unit is kilometer.
- ✧ <*Ambient Temperature*>: A numerical value is used to indicate the ambient temperature.
- ✧ <*DTC*>: Diagnostic trouble codes read from the vehicle. The protocol is OBD II/SAE J2012.

Each diagnostic trouble code is a 3-byte element in following format:

Byte m									Byte m+1								Byte m+2							
7	6	5	4	3	2	1	0		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
C1		C2		C3					C4				C5				-	-	-	-	-	T	P	C

- ➢ C1 – first DTC character
  - b'00' – P – powertrain
  - b'01' – C – chassis
  - b'10' – B – body
  - b'11' – U – network
- ➢ C2 – second DTC character (digit 0 to 3)
- ➢ C3, C4, C5 – consecutive DTC characters (hexadecimal digits 0 to F)
- ➢ T – DTC status flag: permanent (stored into non-volatile memory)
- ➢ P – DTC status flag: pending (detected during current or last driving cycle)
- ➢ C – DTC status flag: confirmed (stored)

Each code may have one or multiple status flags set.

E.g.

0x02 0x2E 0x03 is a code P022E with status pending and confirmed.

0x61 0x99 0x02 is a code C2199 with status pending.

- ✧ <*Gaseous Fuel Level Liter*>: The alternative fuel levels. (when gas installation, i.e. LPG, is factory equipment and information is provided by vehicle). The unit is liter (L).
- ✧ <*Gaseous Fuel Level Percent*>: The alternative fuel levels. (when gas installation, i.e. LPG, is factory equipment and information is provided by vehicle). The unit is 0.4%.
- ✧ <*Fuel Level Combustion Liter*>: The liter of combustion fuel levels, the unit is liter (L).
- ✧ <*Fuel Level Combustion Percent*>: The percent of combustion fuel levels, the unit is 0.4%.
- ✧ <*Total Fuel From Vehicle*>: It expresses total fuel counter read from the vehicle, the unit is mL.
- ✧ <*Total Gaseous Fuel Usage*>: It expresses gaseous total fuel, the unit is 0.5Kg.
- ✧ <*Tyres Count*>: It expresses the count of tyre records.
- ✧ <*Tyres Location*>: It expresses the location of the tyre, the byte is encoded in following way:

MSB	AX3	AX2	AX1	AX0	WH3	WH2	WH1	WH0	LSB

AX..AX0 is a position of an axle (min.0, max.14) representing axles from front (0) to back.

WH3..WH0 is a position of a wheel on the axle (min.0, max.14) representing wheels from left to right.

LOC=0xFF(AX3..AX0 and WH3..WH0 is b'1111') indicates not available.

- ✧ <*Tyres Pressure*>: It expresses the pressure of tyre, the unit is 1kPa.
- ✧ <*Tyres Temperature*>: It expresses the temperature of tyre, the unit is 0.01℃.
- ✧ <*Tyres State*>: it expresses the type of tyres, please refer to the following state:
  - 0: OK.
  - 1: Low pressure (yellow indicator).
  - 2: Very Low Pressure (red indicator).

- 0xFF: status unavailable.

✧ <*Time To Service*>: this is remaining time, unit is 1day, offset -1000, positive value is when service is expected to be done within given number of days, negative value shows how many days ago the service shout be done.
For example: 0x0000 = 1000 day ago, 0x3E8 = today, 0x0555 = in 365 days.

✧ <*Electric info Mask 1*>: Please refer to <*Electric info Mask 1*> parameter in **AT@CAN**.

✧ <*Battery Instantaneous Voltage*>: For electric cars this parameter stands for instantaneous high voltage of battery cells (read from BMS), the unit is voltage (V).

✧ <*Battery Charging Cycles Count*>: For electric cars this parameter counts cycles of battery charging (increments at every finished cycle of charging), the unit is cycle.

✧ <*Total Energy Recuperated*>: Sum of energy transmitted to battery with no charger connected, the unit is Wh.

✧ <*Battery Level*>: The battery charge level for electric vehicles, the unit is 0.4%.

✧ <*Charging State*>: The status of battery charging and Charging cable connected in electric cars.

- 0x00: Plug disconnected
- 0x01: Plug connected, not charging
- 0x02: Connecting (plug connected, but charging not started yet)
- 0x03: Charging in progress
- 0x04: Charging failure

✧ <*Battery Temperature*>: A numerical value is used to indicate the battery temperature.

✧ <*Battery Charging Current*>: A numerical value is used to indicate the electric car's battery charging current, the unit is 0.5A.

✧ <*Battery Instantaneous Power*>: For electric cars this parameter stands for instantaneous power used by (positive values) or recuperated from (negative values) the car, the unit is W.

✧ <*Battery State of Health (SoH)*>: For electric cars this parameter stands for battery general condition (100% means brand new, 0% totally damaged), the unit is 0.4%.

✧ <*Total Energy Used*>: The sum of energy used by car's engine and equipment, recuperated energy does not affect this parameter, the unit is Wh.

✧ <*Total Energy Used When Idling*>: The sum of energy used by car's engine and equipment with vehicle speed 0 km/h, recuperated energy does not affect this parameter, the unit is Wh.

✧ <*Total Energy Charged*>: The sum of energy transmitted to battery from external charger, the unit is Wh.

✧ <*Drv1 End Of Last Daily Rest Period*>: Stands for card in tachograph's slot 1 end of last daily rest period, Dates and times are provided in universal time (UTC). 6 bytes in total. The first 2 bytes are for year, and the other 5 bytes are for month, day, hour and minute respectively.

Send Time	**2011**		**01**	**31**	**06**	**29**
HEX	**07**	**DB**	**01**	**1F**	**06**	**1D**

✧ <*Drv1 End Of Last Weekly Rest Period*>: End Of Last Weekly Rest Period, Dates and times are provided in universal time (UTC). 6 bytes in total. The first 2 bytes are for year, and the other 5 bytes are for month, day, hour and minute respectively.

✧ <*Drv1 End Of Second Last Weekly Rest Period*>: End Of Second Last Weekly Rest Period, Dates and times are provided in universal time (UTC). 6 bytes in total. The first 2 bytes are for year, and the other 5 bytes are for month, day, hour and minute respectively

✧ <*Tachograph Common Event Mask*>: it expresses tachograph common event mask of

tachograph common event of **+RPT** message.

Mask Bit	Item
Bit 31	Reserved
…	…
Bit 12	Reserved
Bit 11	CANBUS Information Mask
Bit 10	Tachograph driver 2 Information Mask
Bit 9	Tachograph driver 1 Information Mask
Bit 8	File In Device Memory Mask
Bit 7	Tachograph Model Mask
Bit 6	APDU Information Mask
Bit 5	Error Code Memory Mask
Bit 4	Error Code Mask
Bit 3	File State In Memory Mask
Bit 2	Device Status Mask
Bit 1	Reply Result Mask
Bit 0	Request ID Mask

✧ <*Request ID*>: It is the response to the server with the same field <*Request ID*> in the command **AT@TAC**.

✧ <*Reply Result*>: it's reply on inquiry read request, different event state, different reply result, refer to the (**Event Type 25)** Tachograph Common Event, refer to the following reply result defined:

- **Event State (1): Reply for DDD file request.**
  - 0: Request OK.
  - 1: Request busy: Advanced test.
  - 2: Request busy: CAN Logistic is executing precious order.
  - 3: Request busy: Configuration of the cancel order.
  - 4: Request busy: The order is forbidden as the device is downloading files now.
- **Event State (2): Authorization result.**
  - 0: Authorization OK.
  - 1: Authorization failed.
  - 2: Authorization timeout.
  - 3: Authorization data error.
- **Event State (4): The result process of read file from tachograph.**
  - 0: CAN Logistic getting file from tachograph success.
  - 1: CAN Logistic getting file from tachograph fail.
  - 2: File mismatch.

- 3: CAN Logistic getting file from Tachograph timeout.
- 4: Device getting file from CAN Logistic timeout.
- 5: Device getting file from CAN Logistic fail.
- 6: DDD file size error

- **Event State (7):** FTP transfer event.
  - 0: FTP open server OK.
  - 1: FTP open server error.
  - 2: FTP sent data OK.
  - 3: FTP sent data error.
- **Event State (14)**: Reply setting tachograph file format.
  - 0: Tachograph file format set successfully.
  - 9: Reading tachograph files is disabled.
  - 255: Setting failure.
- **Event State (15)**: Reply changing the type of communication with the tachograph.
  - 0: Standard type of tachograph communication change successfully.
  - 5: New type of tachograph communication change successfully.

✧ <Device Status>:

- Bit 0 = 1 – Authentication in progress,
- Bit 1 = 1 – Authentication OK (bit is cleared at next read request),
- Bit 2 = 1 – Authentication ERROR (bit cleared at next read request),
- Bit 3 = 1 – CAN Logistic is downloading files from tachograph,
- Bit 4 = 1 – Data is ready to read from CAN Logistic by master device. Bit is cleared after all awaiting data is read (If all bytes are sent in one frame, this bit will be cleared after sending this frame. If all bytes are sent in 3 frames, this bit will be cleared after sending 3 frames).
- Bit 7 ~ Bit 5 = Details of the error. Flags bit 7 ~ bit 5 and flag bit 2 must be treated as combined information. The information is split to maintain protocol's backward compatibility. Flags bit 7 ~ bit 5 are cleared at next read request.
- 000 – No error
- 001 – 24 hours have passed after turning off the ignition, tachograph is in power down mode. (This error code is generated based on the timer of CAN Logistic, there is no attempt of connection with tachograph)
- 010 – 2hours have passed after last authentication session, tachograph is in power down mode. (This error code is generated based on the timer of CAN Logistic, there is no attempt of connection with tachograph)
- 011 – No answer from tachograph (wrongly connected wires or CAN-C is off in tachograph)
- 100 – Cannot open communication session with tachograph (remote download function is off in tachograph)
- 101 – No answer or incorrect answer on company card inquiry (remote download function is off in tachograph).
- 110 – Error during file transfer.
- 111 – Reserved for future use.

✧ <File State In Memory>:

- Bit 0 = 1 tachograph file is in CAN Logistic memory.
- Bit 1 = 1 file from card in slot 1 is in CAN Logistic memory.
- Bit 2 = 1 file from card in slot 2 is in CAN Logistic memory.
- Bit 3 = not used (to be ignored),
- Bit 4 = 1 file from tachograph is already read from CAN Logistic (it is possible to read once more, bit 0 will be still set), this flag is set after read whole file from CAN Logistic.
- Bit 5 = 1 file from card in slot 1 is already read from CAN Logistic (it is possible to read once more, bit 1 will be still set), this flag is set after reading whole file from CAN Logistic.
- Bit 6 = 1 file from card in slot 2 is already read from CAN Logistic (it is possible to read once more, bit 2 will be still set), this flag is set after reading whole file from CAN Logistic.
- Bit 7 = not used (to be ignored).

✧ <Error Code>: Extended error code. A detailed description of the value of this byte below:

Note: Following error codes can be found in protocol (CAN-Logistic v3 protocol XON-XOFF).

- 0x00 – No error detected. Report to the device producer.
- 0x02 – No communication with tachograph. Switch the ignition on and try again. If it does not help, check out CANBUS connection to tachograph.
- 0x0A – Invalid timestamps in read request (7.3.1), start time cannot be later than end time. Correct the request (7.3.1) content.
- 0x0B – CAN-Logistic does not receive real time clock from tachograph, so TSE in read request (7.3.1) cannot be set to 0. Provide the TSE date in request command 7.3.1.
- 0x0C – Invalid read request command (7.3.1). No file requested to download. Correct the request content.
- 0x0E – Error opening session. Report to the device producer.
- 0x0F – No response after sending ATR. Report to the device producer.
- 0x11 – Timeout on waiting for data from company card. CAN-Logistic expected to get reply from company card, but it did not come. Check data transmission path between application server, AVL Terminal and CAN-Logistic and try again.
- 0x18 – No data for company card from tachograph. Report to the device producer.
- 0x1B – Data transfer error. Report to the device producer.
- 0x20 – Error during the authorization process. Report to the device producer.
- 0x21 – Error while downloading the file from the tachograph to the CAN-Logistic memory. Report to the device producer.
- 0x22 – Requested to read activities part since date of last reading, but the date is not available. Correct the request content – set start date.
- 0x23 – Requested to read activities part since date of last reading, but the date is later than requested end time. Correct the request content – change end date.
- 0x30 – Error initializing authorization. Report to the device producer.
- 0x31 – Error during the authorization process. Report to the device producer.
- 0x32 – Error during the authorization process. Report to the device producer.
- 0x33 – Error during the authorization process. Report to the device producer.
- 0x34 – Error during the authorization process. Report to the device producer.

- 0x35 – Error closing session. Report to the device producer.
- 0x39 – Requested cards download, but no cards in tachograph's slots Insert card(s) into tachograph or correct the request (7.3.1) content.
- 0x3A – Error during the authorization process. Report to the device producer.
- 0x3B – Error initializing authorization. Report to the device producer.
- 0x3C – Error during the authorization process. Report to the device producer.
- 0x3D – Error while downloading the file from the tachograph to the CAN-Logistic memory. Report to the device producer.
- 0x3E – Error during the authorization process. Report to the device producer.
- 0x3F – Error opening session. Report to the device producer.
- 0x40 – Downloading finished with no files, because of missing data (i.e. requested cards was removed) Try again. If the error persists, report to the device producer.
- 0x41 – Error opening session. Report to the device producer.
- 0x42 – Error downloading file – invalid package sequence. Report to the device producer.
- 0x55 – Data access denied or communication error. Report to the device producer.
- 0x6A – Remote Authentication Closed. Report to the device producer.
- 0x6C – APDU error. Report to the device producer.
- 0x6E – Authentication error. Check if company card is not expired. If not, report to the device producer.
- 0x70 – Too many authentication errors. Report to the device producer.
- 0x80 – Error opening session. Report to the device producer.
- 0x90 – Error while sending UDS packet. Report to the device producer.
- 0xE1 – There is no operation to perform. Bits SD, SC, SA in byte FL are set. Correct the request (7.3.1) content.
- 0xEE – Downloaded files exceed memory of CAN Logistic. Repeat request with narrower activities range of tachograph DDD file.
- 0xEF – Failed to write the file to internal memory of CAN Logistic. Report to the device producer.
- 0xFE – Request cancelled by command 7.3.2. Start new request whenever you are ready.
- 0xFF – No reading requested since device restart. Start new request whenever you are ready.

✧ <Error Code Memory>: This is the error code byte <Error Code> stored when the next order of the download is sent.

✧ <*APDU Serial Number*>: It is the serial number of APDU received from CAN Logistic. Numbering starts from 0000 when receiving a new DDD file request.

✧ <*APDU DATA Length*>: It is the length of APDU received from CAN Logistic.

✧ <*APDU DATA*>: It is the APDU received from CAN Logistic.

✧ <*Tachograph producer*>:

- 0x00: None
- 0x01: VDO/Siemens
- 0x02: Efas
- 0x03: Stoneridge

- 0x04: Actia
- 0x80: Error of connection on D8 wire
- 0x81: Not supported format „2400" analogue tachographs

✧ <*Tachograph model Length*>: It is the length of Tachograph mode.

✧ <*Tachograph model*>: Tachograph name given by the manufacturer.

✧ <*Tachograph driver time related states*>: Driver time related state.

- 0x00: normal/no limits reached
- 0x01: 15 min before 4½ h
- 0x02: 4½ h reached
- 0x03: 15 min before 9 h
- 0x04: 9 h reached
- 0x05: 15 minutes before 16 h (not having 8h rest during the last 24h)
- 0x06: 16 h reached
- 0x07: week's driving time limit about to be reached
- 0x08: week's driving time limit exceeded
- 0x09: fortnight's driving time limit about to be reached
- 0x0A: fortnight's driving time limit exceeded
- 0x0B: driver's card validity term about to end
- 0x0C: driver's card reading time approaching soon
- 0x0D: other
- 0x0E: error on CAN-bus
- 0x0F: parameter currently not available

✧ <*Tachograph driver 1 id Length*>: It is the length of Tachograph driver 1 id.

✧ <*Tachograph driver 1 id*>: Card number given in ASCII string terminated with „*". First two or three letters are country code. Following is a SPACE character (0x20) and the 16-character long card number e.g.

PL 1820625133460000*

In addition to standard country codes, the following abbreviations also apply:

- „EC" - European Union,
- „EUR" – Rest of Europe,
- „WLD" – rest of the world.

✧ <*Tachograph driver 1 name and surname Length*>: It is the length of Tachograph driver 1 name and surname.

✧ <*Tachograph driver 1 name and surname*>: Surname and name (names) of driver given in an ASCII string terminated with „*". Each word is separated by a SPACE character (0x20).

Note: Number K will be returned if the card is not inserted into tachograph slot (or parameter currently not available; or card inserted, but no information read).

- K = 0x00 card not inserted,
- K = 0x01 card inserted, but no information read,
- K = 0xFF parameter currently not available.

✧ <*CAN1-bus Status*>: It indicates the status of the CAN1-bus.

- 0: bus in sleep mode
- 1: bus active
- 2: bus error

- 3: 2nd CAN-bus or J1708-bus not used

✧ <*CAN2-bus or J1708-bus Status*>: It indicates the status of the CAN2-bus or J1708-bus.

- 0: bus in sleep mode
- 1: bus active
- 2: bus error
- 3: 2nd CAN-bus or J1708-bus not used

✧ <*Check Information*>: It indicates the set of states associated with the tachograph.

MSB	R	R	F1	F0	T1	T0	D1	D0	LSB

R – Reserved

F1:F0 – remote DDD files downloading from tachograph

- b'00' – remote download disabled
- b'01' – remote download enabled
- b'10' – communication error
- b'11' – communication with tachograph not supported with current device settings

T1:T0 – communication with tachograph

- b'00' – no communication with tachograph
- b'01' – tachograph online
- b'10' – communication error
- b'11' – communication with tachograph not supported with current device settings

D1:D0 – D8 (K-Line) bus activity

- b'00' – bus in sleep mode
- b'01' – bus active
- b'10' – bus error
- b'11' –bus not used

Note: If there is no data returned within 2s after the query command is issued due to abnormal communication, 00 or Reserved will be reported (Hex will report 00, string will report Reserved).

✧ <*Tachograph Info Mask*>: Please refer to <*Tachograph Info Mask*> parameter in **AT@CAN**.

✧ <*Tachograph Information*>: Four bytes. The high byte describes driver working states, while the low byte describes tachograph other information and the middle two bytes describe driver 1 information and driver 2 information respectively.

Driver working states	Driver 1 information	Driver 2 information	Tachograph other information
High byte(T)	K1	K2	Low Byte(I)

T: Driver working states:

MSB	OUT1	OUT0	TD2	TD1	TD0	TP2	TP1	TP0	LSB

OUT1:OUT0 – “out of scope” condition (2 bits)

- b'00' – no out of scope condition opened, normal operation
- b’01’ – out of scope condition opened
- b’10’, b’11’– Invalid data.

TD2:TD0 – Driver 2 working state (3 bits)

TP2:TP0 – Driver 1 working state (3 bits)

- b'000': Rest - sleeping.
- b'001': Driver available – short break.
- b'010': Work – loading, unloading, working in an office.
- b'011': Drive – behind the wheel.
- b'100', b'101' – reserved.
- b'110', b'111'– Invalid data.

K1 – Driver 1 information (card in slot 1)

K2 – Driver 2 information (card in slot 2)

MSB	1	1	KK1	KK0	CZ3	CZ2	CZ1	CZ0	LSB

KK1:KK0 – driver card

- b'00' – no card in slot.
- b'01' – card in slot.
- b'10', b'11' – Invalid data.

CZ3:CZ0 – driver time related states (4 bits)

- b'0000' – normal/no limits reached
- b'0001' – 15 min before 4½ h
- b'0010' – 4½ h reached
- b'0011' – 15 min before 9 h
- b'0100' – 9 h reached
- b'0101' – 15 minute before 16 h (not having 8h rest during the last 24h)
- b'0110' – 16 h reached
- b'0111' – weekly driving time pre-warning active
- b'1000' – weekly driving time warning active
- b'1001' – 2 weeks driving time pre-warning active
- b'1010' – 2 weeks driving time warning active
- b'1011' – driver 1 card expiry warning active
- b'1100' – next mandatory driver 1 card download warning active
- b'1101' – other
- b'1110', b'1111' – Invalid data

I – tachograph other information

MSB	1	1	P1	P0	K1	K0	R1	R0	LSB

R1:R0 – vehicle motion

- b'00' – vehicle motion not detected.
- b'01' – vehicle motion detected.
- b'10' – Invalid data.

K1:K0 – direction indicator

- b'00' – forward.
- b'01' – reverse.
- b'10', b'11' – Invalid data.

P1:P0 – vehicle overspeed (it indicates whether the vehicle is exceeding the legal speed limit set in the tachograph)

- b'00' – no overspeed
- b'01' – overspeed
- b'10', b'11' – Invalid data

- <*Tachograph Overspeed Signal*>: Vehicle overspeed signal from the tachograph.
  - 0: Overspeed is not detected.
  - 1: Overspeed is detected.
- <*Tachograph Vehicle Motion Signal*>: The vehicle motion signal in the tachograph.
  - 0: Motion is not detected
  - 1: Motion is detected.
- <*Tachograph Driving Direction*>: Vehicle driving direction from the tachograph.
  - 0: Driving forward.
  - 1: Driving backward.
- <*Length of Tachograph Driver 1 Card Number*>: The length of Tachograph Driver 1 Card Number.
- <*Tachograph Driver 1 Card Number*>: The card number of tachograph driver 1, it's ASCII type.
- <*Length of Tachograph Driver 2 Card Number*>: The length of Tachograph Driver 2 Card Number.
- <*Tachograph Driver 2 Card Number*>: The card number of tachograph driver 2, it's ASCII type.
- <*Length of Tachograph Driver 1 Name*>: The length of Tachograph Driver 1 Name.
- <*Tachograph Driver 1 Name*>: The name of tachograph driver 1, it's ASCII type.
- <*Length of Tachograph Driver 2 Name*>: The length of Tachograph Driver 2 Name.
- <*Tachograph Driver 2 Name*>: The name of tachograph driver 2, it's ASCII type.
- <*Real Time Clock Date and Time*>: Real time clock is a date and time displayed on the car's dashboard, usually read from tachograph.
- <*Special Car Info Mask*>: Please refer to <*Special car Info Mask*> parameter in **AT@CAN**.
- <*Hours Until Service*>: it expresses hour until service of e-GENSET, the unit is hour.
- <*System Status Health*>: It expresses system status health of e-GENSET, 16 bits flag (bit 0 is least significant):
  - Bit 0: F00 Maxtemp Bel Inverter.
  - Bit 1: F01 Maxtemp Bel DC/DC Converter.
  - Bit 2: F02 Maxtemp Battery Cells (BMS).
  - Bit 3: F03 Insulation fault (Bender IMD).
  - Bit 4: F04 Fuses in CC1.
  - Bit 5 to 15: Reserved.
- <*Status Inverter Mode*>: it expresses status inverter mode of e-GENSET.
  - 0: Idle status.
  - 1 or 2: charging status.
  - 8 or 16: power export status.
- <*Remaining Runtime*>: it expresses remaining runtime of e-GENSET, the unit is hour.
- <*Ultima F1*>: it expresses F1 value of multilift ultima.
  - Bit 0 (SWB): Lift swing back.
  - Bit 1 (SWF): Lift swing forward.
  - Bit 2 (UP): Lift motion up.
  - Bit 3 (DN): Lift motion down.
  - Bit 4 (FRW): Lift motion forward.
  - Bit 5 (BCK): Lift motion back.
  - Bit 6: Reserved.

- Bit 7 (INV): it determines if entire data set is valid (INV=1- all data invalid (ignore all other bits, data is unavailable), INV=0 - other bits contain valid status of the machines)

✧ <*Ultima F2*>: it expresses F2 value of multilift ultima.

- Bit 0 (CLU): Clamp unlocking.
- Bit 1 (CLL): Clamp locking.
- Bit 2 (RDN): stabilizer-roller moving down.
- Bit 3 (RUP): stabilizer-roller moving up.

✧ <*Depth Blow Transducer*>: it expresses the depth blow transducer of boat read from NMEA2000. the unit is 0.01m.

✧ <*Heading*>: it expresses the heading of boat read from NMEA2000. the unit is 0.0001 rad.

✧ <*Speed Through Water*>: it expresses the speed through water of boat read from NMEA2000. the unit is 0.01 m/s.

✧ <*Wind Angle*>: it expresses the wind angle of boat read from NMEA2000. the unit is 0.0001 rad.

✧ <*Wind Speed*>: it expresses the wind speed of boat. the unit is 0.01 m/s.

✧ <*BLE Info Mask*>: BLE information read from BLE accessories, bitwise masks refer to the following:

Mask Bit	Item	Description
Bit 15	Reserved	
Bit 14	Reserved	
Bit 13	Reserved	
Bit 12	Custom6	
Bit 11	Custom5	
Bit 10	Custom4	
Bit 9	Custom3	
Bit 8	Custom2	
Bit 7	Custom1	
Bit 6	BLE Fuel Frequency	
Bit 5	BLE Fuel	
Bit 4	BLE Luminous	
Bit 3	BLE Door State	
Bit 2	BLE Humidity	
Bit 1	BLE Temperature	
Bit 0	BLE Battery Voltage	

✧ <*BLE Battery Voltage*>: the battery voltage of BLE accessory, the data size is 2 bytes, the range is from 0 to 65535, the unit is mV.

- <*BLE Temperature*>: the temperature of BLE accessory, the data size is 2 bytes, the range is from 0 to 65535, the unit is 0.01°C.
- <*BLE Humidity*>: the humidity of BLE accessory, the data size is 2 bytes, the range is from 0 to 65535, the unit is 0.01%.
- <*BLE Door State*>: the door state of BLE accessory, the data size is 1 byte, the range is from 0 to 1.
- <*BLE Luminous*>: the Luminous of BLE accessory, the data size is 2 bytes, the range is from 0 to 65535.
- <*BLE Fuel*>: the fuel of BLE accessory, the data size is 2 bytes, the range is from 0 to 65535.
- <*BLE Fuel Frequency*>: the fuel frequency of BLE accessory, the data size is 4 bytes, the range is from 0 to 4294967295.
- <*Custom1 Length*>: the length of custom1 data.
- <*Custom1 Data*>: the custom1 data of BLE accessory, the data size max is 20 bytes.
- <*Custom2 Length*>: the length of custom2 data.
- <*Custom2 Data*>: the custom2 data of BLE accessory, the data size max is 20 bytes.
- <*Custom3 Length*>: the length of custom3 data.
- <*Custom3 Data*>: the custom3 data of BLE accessory, the data size max is 20 bytes.
- <*Custom4 Length*>: the length of custom4 data.
- <*Custom4 Data*>: the custom4 data of BLE accessory, the data size max is 20 bytes.
- <*Custom5 Length*>: the length of custom5 data.
- <*Custom5 Data*>: the custom5 data of BLE accessory, the data size max is 20 bytes.
- <*Custom6 Length*>: the length of custom6 data.
- <*Custom6 Data*>: the custom6 data of BLE accessory, the data size max is 20 bytes.
- <*RS485 Accessory Number*>: The number of RS485 accessory and max support 25 group of RS485 accessory data, it includes <*RS485 Accessory Model*>, <*RS485 Accessory Address*>, <*RS485 Data Mask*>, <*Fuel Volume*>, <*Temperature*> information.
- <*RS485 Accessory Model*>: the model of RS485 accessory.
  - 1: DYP-U02 ultrasonic oil level sensor.
- <*RS485 Accessory Address*>: the communication address of RS485 accessory in RS485 bus.
- <*RS485 Data Mask*>: The bitwise masks of RS485 data mask refer to the following:

Mask Bit	Item	Description
Bit 15	Reserved	
…	…	
Bit 2	Reserved	
Bit 1	Temperature	
Bit 0	Volume	

- <*Fuel Volume*>: the volume of fuel tank.
  - DYP-U02 ultrasonic oil level sensor, output fuel volume value is height of fuel, the unit is 0.1mm.
- <*Temperature*>: the environment temperature of fuel tank.

3.2.2.+LDP/-LDP

'**+LDP**' is use for large data packet report.

'**-LDP**' is historical report which are saved in buffer.

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+LDP:/-LDP:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Large Data Packet Mask	4		Large Data Packet information
1	**Bit 0:** Total Frame	2	0001 – FFFF	
2	**Bit 0:** Current Frame ID	2	0001 – FFFF	
1	**Bit 1:** File Download Count	2		
2	**Bit 1:** DDD File Name Length	1		
3	**Bit 1:** DDD File Name	<=N		
1	**Bit 2:** DDD Data Length	2		
2	**Bit 2:** DDD Data	<=N		
13	Generated Time	7	YYYYMMDDHHMMSS	
14	Serial Number	2	0000 – FFFF	
15	Tail	1	#	#

✧ <*Large Data Mask*>: Component mask of the **+LDP/-LDP** message.

Mask Bit	**Item**
Bit 31	Reserved
…	…
Bit 4	Reserved

Bit 3	Reserved
Bit 2	Download DDD Data Mask
Bit 1	Download DDD File information Mask
Bit 0	Multi-Packet Mask

- <*File Download Count*>: It is a number that marks DDD file download.
- <*Total Frame*>: It indicates the total number of reports generated by dividing the message. When the length of the message is too long, the terminal will automatically split it into multiple reports to send to the backend server.
- <*Current Frame ID*>: It indicate the sequence of the current multiple reports.
- <*DDD Data Length*>: It indicates the length of DDD data in current frame.
- <*DDD Data*>: It indicates the DDD data of tachograph.

3.2.3.+BMR/-BMR

'**+BMR'** is real-time report. These are the reports sent to the device management server.
'-BMR' is historical report which are saved in buffer.

Example:
2B 41 43 4B 3A 00 26 56 14 2C 06 32 38 17 01 10 42 4D 53 2C 30 31 30 31 2C 2C 30 2C 07 E7 08 1F 10 16 0C 14 41 23

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	5		+BMR:/-BMR:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	BMS Protocol Version	2		
6	Track Protocol Version	2		
7	BB Hardware Version	1		
8	BB Firmware Version	2		
9	BMR Data Mask	4		
1	Main Power VCC	2	0 – 32000 (mV)	**Bit 0:** Main Power Mask
1	Battery Level	1	0 – 100 (%)	**Bit 1:** Battery Level Mask
1	Auxiliary MCU Version	2		**Bit 2:** Auxiliary MCU Mask

1	Config Version	2		**Bit 3:** Config Version Mask
1	CAN Serial Number	4		**Bit 4:** CAN Info Mask
2	CAN FW Version	4		
1	Ignition & Motion State	1		**Bit 5:** IGN State Mask
1	GNSS On	1	0|1	**Bit 6:** GNSS Mask
2	Satellite Number	1	0 - 6	
3	Satellite ID	1		
4	Satellite Power	1		
5	Last GNSS HDOP	1		
6	GNSS UTC Time	7	YYYYMMDDHHMMSS	
1	IMSI	8		**Bit 7:** SIM Info Mask
2	ICCID	10		
1	Module Version Length	1		**Bit 8:** Module Version Mask
2	Module Version	<=50	String	
1	BOOT Firmware Version	2		**Bit 9:** BOOT Firmware of Main MCU Mask
1	Generated Time	7	YYYYMMDDHHMMSS	
2	Serial Number	2	0000 – FFFF	
3	Tail	1	#	#

- <*Hardware Version*>: The hardware version. The first two characters represent the major version and the last two characters represent the minor version. For example, **010B** means the version **1.11**.
- <*Firmware Version*>: The firmware version. The first two characters represent the major version and the last two characters represent the minor version. For example, **010B** means the version**1.11**.
- <*BMR Data Mask*>: Component mask of the **+BMR/-BMR** message, set this parameter with the **AT@BMS** command.
- <*Battery Level*>: Battery Percent Information.
- <*BOOT Firmware Version*>: The BOOT firmware version of main MCU. The first byte represents the boot type and the last byte represent the BOOT version. For example, **0102** means 0x01 of the boot type express AVT boot. 0x02 express Boot version is V02.

3.3. Heart Beat Data

3.3.1.+SHBD

The **+SHBD** is encoded using printable ASCII code, its frame format is:

Example: +SHBD:04FF# +SHBD:0502# +SHBD:04F9#				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	6	ASCII	+SHBD:
2	Serial Number	4	0000 – FFFF	
3	Tail	1	#	#

✧ <*Serial Number*>: It corresponds to the serial number in the report. For example, the serial number in the report is 0x01F7, here it is ASCII "01F7".

3.3.2.+HBD

In order to maintain the connection between backend server and terminal, the terminal provides this heartbeat mechanism: the terminal periodically sends a **+HBD** frame to the server when there is no data transmitted between server and terminal in this periodical time. The backend server should responds with a **+SHBD** frame to the terminal when it receives a +HBD message. The heartbeat feature is configured by the **AT@SIS** command. please refer to the frame format below.

The **+HBD** frame format is:

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+ACK:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
6	Protocol Version	2		

10	Generated Time	7	YYYYMMDDHHMMSS	
11	Serial Number	2	0000 – FFFF	
12	Tail	1	#	#

✧ <*Protocol version*>: First byte indicates the major version number of the protocol. the last byte indicates the minor version number of the protocol, for example: **0x0B01** means version **11.01**.

3.4. Acknowledgement

When the terminal receives a command, it responds an ACK to the backend server as an acknowledgement of receiving the command.

In addition, when the backend server successfully receives a report sent by terminal, it can also respond with a **+SACK** to the terminal when necessary, indicating that the report is received.

The frame formats of **+ACK**, and **+SACK** are shown below.

3.4.1.+ACK

The **+ACK** frame format is:

Example: 2B 41 43 4B 3A 00 26 56 14 2C 06 32 37 51 01 10 43 43 53 2C 30 31 30 31 2C 2C 30 2C 07 E7 08 18 09 0D 0E 05 01 23				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+ACK:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Command Key	<=N		
6	Protocol Version	4		
7	Sub Command	<=N		
8	Result	<=2		
9	Reserved	0		
10	Generated Time	7	YYYYMMDDHHMMSS	
11	Serial Number	2	0000 – FFFF	
12	Tail	1	#	#

- ✧ <*Command Key*>: Please refer to the "Command Key" in Configuration Command.
- ✧ <*Sub Command*>: Please refer to sub command in the **AT@RTO** command.
- ✧ <*Result*>: it is used to indicate the specific reason for the **+ACK** to be triggered.
  - 0: Success.
  - 1: Password error.
  - 2: Parameters error.
  - 3: Command is not supported.

3.4.2.+SACK

The **+SACK** is encoded using printable ASCII code, its frame format is:

Example:
+SACK:04FF#
+SACK:0502#
+SACK:04F9#

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	6	ASCII	+SACK:
2	Serial Number	4	0000 – FFFF	
3	Tail	1	#	#

- ✧ <*Serial Number*>: It corresponds to the serial number in the report. For example, the serial number in the report is 0x01F7, here it is ASCII "01F7".

3.5. Command Query Data

3.5.1. +QRY

When the terminal receives a **Query Command** from the backend server, it will use the following frame format to respond with the current parameters of the command to the server. The **+QRY** frame format as the following:

Example:
+QRY:RTO,862044065055811,10,0101#

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	5		+QRY:
2	Length	2		
3	IMEI	8		

4	Device ID	1		0x24
5	Protocol Version	4		
6	Command Key	<=N		
7	Parameters	<=N		
8	Generated Time	7	YYYYMMDDHHMMSS	
9	Serial Number	2	0000 – FFFF	
10	Tail	1	#	#

✧ <*Parameters*>: The current working parameters of the terminal device. Its content is determined by 'Command Key', which is consistent with 'Parameters' in 'Configuration Command'.

3.5.2. +ALL

When the terminal receives a query all commands (**AT@RTO**) from the backend server, it will use the following frame format to respond with the current parameters of the command to the server. The **+ALL** frame format as the following:

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	5		+ALL:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
1	Protocol Version	4		
2	Last Packet Flag	1	0|1	
	SIS	3		SIS
1	Report Mode	1	0 - 3	0
2	Buffer Mode	1	0 - 2	1
3	Main Server IP / Domain Name	<=60		
4	Main Server Port	<=5	0 – 65535	
5	Reserved	0		
6	Reserved	0		
7	SACK Enable	1	0 - 1	0
8	Heartbeat Interval	<=3	0|2 – 360 (min)	0

9	PDP Enable	1	0 - 1	0
10	TLS Enable	1	0 - 1	0
11	TLS Verification Mode	1	0 - 2	0
12	Message Filter Mode	1	0 - 2	0
13	Message Filter Mask1	1	00000000 – FFFFFFFF	0
14	Resend Time No SACK	<=3	10 - 300	10
	SSI	3		SSI
1	Report Mode	1	0 - 3	0
2	Buffer Mode	1	0 - 2	1
3	Second Server IP / Domain Name	<=60		
4	Second Server Port	<=5	0 – 65535	
5	Reserved	0		
6	Reserved	0		
7	SACK Enable	1	0 - 1	0
8	Heartbeat Interval	<=3	0|2 – 360 (min)	0
9	Reserved	0		
10	TLS Enable	1	0 - 1	0
11	TLS Verification Mode	1	0 - 2	0
12	Message Filter Mode	1	0 - 2	0
13	Message Filter Mask1	<=8	00000000 – FFFFFFFF	0
	NIS	3		NIS
1	APN	<=42		
2	APN User Name	<=32	NULL	
3	APN Password	<=32	NULL	
4	Reserved	0		
5	Reserved	0		
6	Reserved	0		
7	Network Scan Mode	1		0

	GMT	3		GMT
1	Time Zone Direction	1	0|1	
2	Time Zone	<=2	0 - 12	
	GCS	3		GCS
1	New Password	4 - 19	'0' - '9', 'a' - 'z', 'A' - 'Z'	
2	Data Mask	<=8	(HEX)	
3	Event Data Mask	<=8	(HEX)	
4	LED Mode	1	0|1	0
5	Low Battery Threshold	<=2	5 – 30 (%)	10
6	Low Main Power Threshold	<=5	8000 – 24000 (mV)	9000
7	GNSS Working Mode	1	1|3|4	1
8	GNSS Info Mask	2	0000 - FFFF	FF
9	AGPS Mode	1	0|1	0
10	ODO Enable	1	0|1	0
11	ODO Initial Mileage	<=9	0.0 – 4294967.0 (Km)	0.0
12	Hour Meter Enable	1	0|1	0
13	Initial Hour Meter Count	11	00000:00:00 - 99999:59:59	00000:00:00
14	Power Saving Mode	1	0 – 2	0
	RRS	3		RRS
1	Mode	1	0|1	0
2	IGN GNSS Sample Interval	<=4	0 – 3600 (sec)	0
3	IGN Send Interval	<=4	1 – 3600 (sec)	30
4	IGF Send Interval	<=4	0 – 86400 (sec)	3600
5	Discard No Fix	<=1	0|1	1
6	Reserved	0		
7	Corner Angle	<=3	0|5– 90	30
	MOV	3		MOV
1	Motion Threshold	1	0 - 9	2
2	Motion Duration	<=2	1 – 10 (sec)	1

3	Stillness Duration	<=2	1 – 60 (sec)	30
	TOW	3		TOW
1	Mode	1	0|1	0
2	Tow Duration	<=4	120 – 3600 (sec)	120
3	Tow Alarm Interval	<=4	60 – 3600 (sec)	60
	SPD	3		SPD
1	Mode	1	0|1	0
2	Over Speed Threshold	<=3	30 – 400 (km/h)	100
3	Over Speed Duration	<=4	5 – 3600 (sec)	10
4	Report Interval	<=4	30 – 3600 (sec)	60
	GEO	3		GEO
1	GEO ID	<=2	1 – 32	
2	Mode	1	0 – 1	0
3	Longitude	<=11	(–)xxx.xxxxxx	
4	Latitude	<=10	(–)xx.xxxxxx	
5	Radius	<=7	50 – 60000 (m)	50
6	Check Interval	<=5	0|5 – 3600 (sec)	0
	HBS	3		HBS
1	Mode	1	0 - 1	0
2	Acceleration Threshold	<=3	10 - 100	20
3	Acceleration Duration	<=3	10 – 250 (*8ms)	25
4	Braking Threshold	2	10 - 100	25
5	Braking Duration	<=3	10 - 250(*8ms)	25
6	Cornering Threshold	2	10 - 100	20
7	Cornering Duration	<=3	10 - 250 (*8ms)	25
	CRA	3		CRA
1	Mode	1	0 – 1	0
2	X Axis Threshold	<=2	0 – 160 (*0.1g)	50
3	Y Axis Threshold	<=2	0 – 160 (*0.1g)	50
4	Z Axis Threshold	<=2	0 – 160 (*0.1g)	50

	IGN	3		IGN
1	Wire Ignition Mode	1	0|1	1
2	IGN Debounce Time	<=2	0 – 20 (*10ms)	5
3	Reserved			
4	Reserved			
5	Virtual Ignition Mode	1	0|1	1
6	VGF Rest Duration	<=3	5 - 180(sec)	5
7	VGN Motion Duration	<=3	5 – 180(sec)	20
8	Ignition On Voltage	<=5	250 – 28000 mV	13500
9	Voltage Offset	<=4	200 – 2000 mV	600
10	Ignition On Debounce	<=3	5 – 255 (× 1 sec)	10
11	Ignition Off Debounce	<=3	5 – 255 (× 1 sec)	5
	DAM	3		DAM
1	Reserved	1		
2	Input Mode	1	0 – 2	0
3	Debounce Time	<=2	0 – 20 (*10ms)	2
4	Validity Time	<=2	0|1 – 12 (*2s)	0
	AOS	3		AOS
1	Alarm OUT1 Trigger Mask	<=8	0 – FFFFFFFF	0
2	Output ID	1	1	1
3	Output Status	1	0 – 1	
4	Duration	<=3	0 – 255 (*100ms)	
5	Toggle Times	<=3	0 – 255	
	WAS	3		WAS
1	iButton Timer	<=2	0 – 10 (sec)	0
2	Temperature Timer	<=3	0|10– 255 (sec)	0
3	1wire Data Mask	<=2	00 – FF	0
	ETS	3		ETS
1	Sensor Number	1	0 - 5	
2	Reserved	0		

3	Sensor ID	<=16		
4	Temperature Alarm Mode	1	0 – 4	0
5	High Temperature	<=3	-55 – 125 (℃)	0
6	Low Temperature	<=3	-55 – 125 (℃)	0
7	Temperature Sample Interval	1	10 – 600 (sec)	10
8	Temperature Duration	<=4	10 – 3600 (sec)	10
	RCS	3		RCS
1	Reserved	0		
2	Baud Rate Index	<=2	1 – 12	12
3	Data Bits	1	7 – 8	8
4	Stop Bits	1	1 – 3	1
5	Parity Bits	1	0 – 2	0
6	Reserved	0		
7	Reserved	0		
	RAS	3		RAS
1	Accessory Type Number	1	0 – 5	0
2	Accessory Model	1	0 - 1	0
3	Address ID Number	1	1 - 5	1
4	Address ID	<=2	HEX String	
	DRV	3		DRV
1	Mode	1	0|1|2	0
2	Reserved	0		
3	Start Index	<=3	1 – 250	
4	End Index	<=3	1 – 250	
5	ID Number List	<=16*28	'0' – '9','a' – 'f', 'A' – 'F'	
6	ID Authorized Validity Time	<=3	15 – 600 (sec)	30
7	Authorized Timeout after IGF	<=3	0|15 – 600 (sec)	30

	CAN	3		CAN
1	Mode	1	0|1|6	0
2	Special Car Model	<=5	158|160|161	160
3	CAN Report Interval	<=5	0|1 – 86400 (sec)	0
4	CAN Report Interval IGF	<=5	0|60 – 86400 (sec)	0
5	CAN Info Mask 1	8	0 - FFFFFFFF	C00FFFFF
6	CAN Info Mask 2	8	0 - FFFFFFFF	007FFFFF
7	Reserved	0		
8	Electric Info Mask 1	<=8	0 – FFFFFFFF	C0
9	Reserved	0		
10	Additional Event Mask	1	0|1	0
11	CAN 120 ohms Mask	1	0 – 3	0
12	Tachograph Driver1 Working Time Mask	<=8	0 – FFFFFFFF	0
13	Tachograph Driver2 Working Time Mask	<=8	0 – FFFFFFFF	0
14	Tachograph Info Mask	<=8	0 – FFFFFFFF	0
15	Special Car Info Mask	<=8	0 – FFFFFFFF	0
16	NMEA2000 Info Mask	<=8	0 – FFFFFFFF	0
17	Tacho Driver Time Mode	1	0 – 3	0
	BMS	3		BMS
1	Report Mode	1	0 - 1	0
2	Buffer Mode	1	0 - 2	1
3	BMS Server IP / Domain Name	<=60		
4	BMS Server Port	<=5	0 – 65535	
5	Connection Life	<=3	0|5 – 300 (sec)	30
6	Reserved	0		
7	Reserved	0		
8	Report Interval	<=5	1 – 2880 (min)	1440
9	BMR Data Mask	<=8	0 – FFFFFFFF	1

	DBG	3		DBG
1	Working Mode	1	0 - 1	0
2	Debug Info Mask	8	00000000 – FFFFFFFF	0
3	Server IP / Domain Name	<=60		
4	Server Port	<=5	0 – 65535	0
	ECL	3		ECL
1	Group Number	<=2	0 - 15	0
2	Group Mode	1	0 - 1	0
3	Reserved	0		
4	Event ID Mask	<=8	0 - FFFFFFFF	0
5	Bind Command Mask	<=8	0 - FFFFFFFF	0
6	Command ACK	0	0 - 1	0
	PCL	3		PCL
1	Preset Command ID	<=1	0 - 31	
2	Command String	<=200	AT Command | NULL	
	BSS			BSS
1	Scan Working Mode	1	0|1|2|3	0
2	Scan Frequency	<=5	1 – 65535 (sec)	10
3	Scan Duration	<=2	1 – 300 (100 msec)	1
4	Filter Group	1	1 – 4	
5	Group Working Mode	1	0|1	0
6	MAC address	0 - 12		
7	Data Clear Period	<=5	0 – 65535	0
8	Match & Save Group	2	<=10	10
9	EIR Data Type	2	HEX	
10	Data offset	<=2		0
11	Data Size	<=2		0
12	Action	1	0|1	0
13	I/O	<=2	0 – 11	0
14	Match	<=20	HEX	

15	Endian Type	1	0 – 1	0
16	Multiplier	<=8		0
17	Offset	<=6	-32768 – 32767	0
18	BSS Report Interval Motion	<=5	0|5 – 86400 (sec)	60
19	BSS Report Interval Motionless	<=5	0|5 – 86400 (sec)	60
	JAM	3		JAM
1	Mode	1	0 – 1	0
2	Enter Jamming Duration	<=3	0 – 60 (*Sec)	5
3	Quit Jamming Duration	<=2	0 – 60 (*Sec)	5
	SSI			SSI
1	Report Mode	1	0 - 3	0
2	Buffer Mode	1	0 - 2	1
3	Second Server IP / Domain Name	<=60		
4	Second Server Port	<=5	0 – 65535	
5	Reserved	0		
6	Reserved	0		
7	SACK Enable	1	0 - 1	0
8	Heartbeat Interval	<=3	0|2 – 360 (min)	0
9	Reserved	0		
10	Reserved	0		
11	Reserved	0		
12	Message Filter Mode	1	0 - 2	0
13	Message Filter Mask	<=8	00000000 – FFFFFFFF	0
1	Generated Time	7	YYYYMMDDHHMMSS	
2	Serial Number	2	0000 – FFFF	
3	Tail	1	#	#

3.5.3. +QNI

When the terminal receives a query network information command (**AT@RTO**) from the backend server, it will use the following frame format to respond with the current parameters of the command to the server. The **+QNI** frame format as the following:

Example:

2B 51 4E 49 3A 00 71 56 14 2C 06 32 37 51 01 10 30 31 30 31 2C 2C 2C 2C 2C 2C 34 36 30 30 38 34 33 38 37 35 30 39 34 33 33 2C 38 39 38 36 30 34 43 33 32 31 32 32 32 30 32 31 39 34 34 38 2C 30 2C 39 39 2C 30 2C 31 30 2E 36 31 2E 33 30 2E 35 37 2C 31 33 39 2E 31 39 36 2E 32 31 39 2E 35 2C 2C 32 2C 2C 2C 2C 07 E7 08 18 08 15 32 04 DD 23

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	5		+QNI:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	APN	<=42		
7	APN User Name	<=32		
8	APN Password	<=32		
9	Reserved			
10	Reserved			
11	Reserved			
12	IMSI	<=15		
13	ICCID	20		
14	CSQ RSSI	<=2	0 – 31 | 99	
15	CSQ BER	<=2	0 - 7 | 99	
16	Cell ID	4 | 8	XXXX | XXXXXXXX	
17	IP Address	<=15	(IP)	
18	Reserved			
19	Reserved			
20	Network Type	1	0|3|8	
21	Reserved			
22	Reserved			

23	Reserved			
24	Reserved			
25	Generated Time	7	YYYYMMDDHHMMSS	
26	Serial Number	2	0000 – FFFF	
27	Tail	1	#	#

✧ <*CSQ RSSI*>: The signal strength level.

CSQ RSSI	Signal Strength (dBm)
0	<-133
1	-111
2 - 30	-109 - -53
31	>-51
99	Unknow

✧ <*CSQ BER*>: The strength of the signal. The range is 0-7 and 99 means unknown.

✧ <*Cell ID*>: Cell ID in hex format.

✧ <*IP Address*>: The IP address of the device.

✧ <*Main DNS*>: The main DNS server.

✧ <*Network Type*>: Current network type.

- 0: EGPRS
- 3: CAT-1
- 8: Unknow

3.5.4. +VER

When the terminal receives a query version information command (**AT@RTO**) from the backend server, it will use the following frame format to respond with the current parameters of the command to the server. The **+VER** frame format is:

Example:
2B 56 45 52 3A 00 2F 56 14 2C 06 32 37 51 01 10 30 31 30 31 2C 30 31 30 31 31 30 2C 30 31 2C 2C 2C 2C 2C 2C 2C 2C 07 E7 08 18 06 39 33 04 B6 23

No	Parameter	Length (Byte)	Range / Format	Default
1	Head	5		+VER:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	Firmware Version	6		

7	Hardware Version	2		
8	Reserved	0		
9	Modem Software Version	<=50		
10	Reserved			
11	Reserved			
12	Reserved			
13	Reserved			
14	Reserved			
15	Generated Time	7	YYYYMMDDHHMMSS	
16	Serial Number	2	0000 – FFFF	
17	Tail	1	#	#

- ✧ <*Firmware Version*>: The firmware version. The first two characters represent the major version and the last two characters represent the minor version. For example, **010B** means the version**1.11**.
- ✧ <*Hardware Version*>: The hardware version. The first two characters represent the major version and the last two characters represent the minor version. For example, **010B** means the version **1.11**.
- ✧ <*Modem Software Version*>: It gives the modem software version information of this device.
- ✧ <*Sensor ID*>: It indicates the type of the sensor currently used by the device.

3.5.5. +GSV

After the device receives the command **AT@RTO** to get the GPS satellite information, it will send the GPS satellite information via the message **+GSV** to the backend server. The **+GSV** frame format is:

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+GSV:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	GPS SV Count	2	0-24	

7	SV ID	2	> =0	
8	SV Power	2	> =0	
9	…			
10	SV ID	2	> =0	
11	SV Power	2	> =0	
12	Reserved			
13	Reserved			
14	Generated Time	7	YYYYMMDDHHMMSS	
15	Serial Number	2	0000 – FFFF	
16	Tail	1	#	#

- ✧ <*GPS SV Count*>: The count of satellites the GPS finds.
- ✧ <*SV ID*>: The satellite ID. In case of no satellite, the field is filled with zero.
- ✧ <*SV Power*>: Satellite power. In case of no satellite, the field is filled with zero.

3.5.6. +LSV

After the device receives the command **AT@RTO** to get the GLONASS satellite information, it will send the GLONASS satellite information via the message **+LSV** to the backend server. The **+LSV** frame format is:

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+LSV:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	GLONASS SV Count	2	0-24	
7	SV ID	2	> =0	
8	SV Power	2	> =0	
9	…			
10	SV ID	2	> =0	
11	SV Power	2	> =0	

12	Reserved			
13	Reserved			
14	Generated Time	7	YYYYMMDDHHMMSS	
15	Serial Number	2	0000 – FFFF	
16	Tail	1	#	#

- ✧ <*GLONASS SV Count*>: The count of satellites the GLONASS finds.
- ✧ <*SV ID*>: The satellite ID. In case of no satellite, the field is filled with zero.
- ✧ <*SV Power*>: Satellite power. In case of no satellite, the field is filled with zero.

3.5.7. +BSV

After the device receives the command **AT@RTO** to get the BEIDOU satellite information, it will send the BEIDOU satellite information via the message **+BSV** to the backend server. The **+BSV** frame format is:

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+BSV:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4	(HEX)	
6	BEIDOU SV Count	2	0-24	
7	SV ID	2	> =0	
8	SV Power	2	> =0	
9	…			
10	SV ID	2	> =0	
11	SV Power	2	> =0	
12	Reserved			
13	Reserved			
14	Generated Time	7	YYYYMMDDHHMMSS	
15	Serial Number	2	0000 – FFFF	
16	Tail	1	#	#

- ✧ <*BEIDOU SV Count*>: The count of satellites the BEIDOU finds.
- ✧ <*SV ID*>: The satellite ID. In case of no satellite, the field is filled with zero.
- ✧ <*SV Power*>: Satellite power. In case of no satellite, the field is filled with zero.

3.5.8. +GAV

After the device receives the command **AT@RTO** to get the GALILEO satellite information, it will send the GALILEO satellite information via the message **+GAV** to the backend server. The **+GAV** frame format is:

Example:				
No	**Parameter**	**Length (Byte)**	**Range / Format**	**Default**
1	Head	5		+GAV:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4	(HEX)	
6	GALILEO SV Count	2	0-24	
7	SV ID	2	> =0	
8	SV Power	2	> =0	
9	…			
10	SV ID	2	> =0	
11	SV Power	2	> =0	
12	Reserved			
13	Reserved			
14	Generated Time	7	YYYYMMDDHHMMSS	
15	Serial Number	2	0000 – FFFF	
16	Tail	1	#	#

- ✧ <*GALILEO SV Count*>: The count of satellites the GALILEO finds.
- ✧ <*SV ID*>: The satellite ID. In case of no satellite, the field is filled with zero.
- ✧ <*SV Power*>: Satellite power. In case of no satellite, the field is filled with zero.

3.5.9. +CVS

After the device receives the command **AT@RTO** to get the version number and serial number of

the CAN Module, it will send the information to the backend server via the message **+CVS**.

- **+CVS,**

Example:				
No	**Parameter**	**Length (Byte)**	**Range/Format**	**Default**
1	Head	5		+CVS:
2	Length	2		
3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	CAN Version Number	<=7	ASCII	
7	CAN Serial Number	<=10	'0' – '9' 'a' – 'z'	
8	Reserved			
9	Reserved			
10	Reserved			
11	Reserved			
12	Generated Time	7	YYYYMMDDHHMMSS	
13	Serial Number	2	0000 – FFFF	
14	Tail	1	#	#

- <*CAN Version Number*>: The SW version of the CAN device.
- <*CAN Serial Number*>: The Serial Number of the CAN device.

3.5.10. +CMI

After the device receives the command **AT@RTO** to get the car model ID of the CAN Module, it will send the information to the backend server via the message **+CMI**.

- **+CMI,**

Example:				
No	**Parameter**	**Length (Byte)**	**Range/Format**	**Default**
1	Head	5		+CMI:
2	Length	2		

3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	CAN Module Car Model ID	<=5	0-65535	
7	CAN Module Car Name	<=50	ASCII	
8	CAN Module Sync Status	1	1-4	
9	Reserved			
10	Reserved			
11	Reserved			
12	Reserved			
13	Generated Time	7	YYYYMMDDHHMMSS	
14	Serial Number	2	0000 – FFFF	
15	Tail	1	#	#

- <*CAN Module Car Model ID*>: The car model ID of the CAN Module device. If the value is 0, it means that no model has been obtained.
- <*CAN Module Car Name*>: Car Name is human readable make and model of the car. If this string value is equal to "sync", then the <*CAN Module Car Model ID*> of the query is a class ID that has been synchronized.
- <*CAN Module Sync Status*>: The CAN Module synchronization status.
  - 1: The synchronization is successful.
  - 2: CAN Module is not properly connected to the CAN bus or ignition is switched off.
  - 3: The car is not supported by the firmware version of CAN Module or CAN-bus codes has not been recognized.
  - 4: CAN Module is not responding.

3.5.11. +DFI

After the device receives the command **AT@RTO** to get the device full information, it will send the information to the backend server via the message **+DFI**.

- **+DFI,**

Example:				
No	**Parameter**	**Length (Byte)**	**Range/Format**	**Default**
1	Head	5		+DFI:
2	Length	2		

3	IMEI	8		
4	Device ID	1		0x24
5	Protocol Version	4		
6	Last Packet Flag	1	0|1	
7	Firmware Version	6		
8	Hardware Version	2		
9	Network Type	1	0|3|8	
10	CSQ RSSI	<=2	0 – 31 | 99	
11	GNSS State	1		
12	GNSS HDOP	<=2		
13	GNSS Speed	<=3		
14	GNSS Corner	<=3		
15	GNSS Latitude	<=10		
16	GNSS Longitude	<=10		
17	GNSS UTC Time	14		
18	GPS SV Number	<=2	0 – 24	
19	BEIDOU SV Number	<=2	0 – 24	
20	GALILEO SV Number	<=2	0 – 24	
21	CAN HW Version	<=4	ASCII	
22	CAN SW Version	<=7	ASCII	
23	CAN Serial Number	<=10	'0' – '9' 'a' – 'z'	
24	CAN Sync Model ID	<=5	0 - 65535	
25	CAN Module Car Name	<=50	ASCII	
26	CAN Module Sync Status	1	1 - 4	
27	Tacho Support	1	0 - 1	
28	Tacho CAN1-bus State	1	0 - 3	
29	Tacho CAN2-bus State	1	0 - 3	
30	Tacho Check information	2	00 - FF	
31	Tacho Information	<=8	00000000 – FFFFFFFF	
32	Tacho Driver 1 Card Number	<=40		

33	Tacho Driver 2 Card Number	<=40		
34	Tacho Driver 1 Name	<=40		
35	Tacho Driver 2 Name	<=40		
36	VIN	<=17		
37	Range (ODO)	<=7	0 – 9999999 (km)	
38	Fuel Level Liter	<=3	0 – 256 (L)	
39	Fuel Level Percent	<=3	0 – 250 (*0.4%)	
40	Total Fuel Used	<=9	0 – 999999999 (mL)	
41	Engine RPM	<=5	0 – 16383 (rpm)	
42	Engine Coolant Temperature	<=4	-40 – +215 (℃)	
43	Reserved			
44	Reserved			
45	Reserved			
46	Reserved			
47	Reserved			
48	Reserved			
49	Reserved			
50	Reserved			
51	Reserved			
52	Reserved			
53	Reserved			
54	Reserved			
55	Generated Time	7	YYYYMMDDHHMMSS	
56	Serial Number	2	0000 – FFFF	
57	Tail	1	#	#