

S8.2/S8.3/S8.3D8/S8.5/FM devices

Communication protocol

Technical Documentation

1.44

2018-11-28

Device firmware version **SV: Nov 28 2018 15:12:51**

Device hardware version rev_14/rev_20

Attention!

[S8/FM protocol is compatible with S6 protocol]

[changes to last protocol version are marked on red]

Scope

This document contains technical documentation for new S 8th and FM generation series of GPRS telemetry. **It is important to know that it was priority to make device full compatible with previous S6 protocol. It's mean that S8/FM GPRS device can be connect immediately to server - if server was prepared to work with S6 device!**

This document main purpose is to provide detailed communication protocol description. First chapter contains brief information about unit electrical characteristics and IO description. The next chapter contains description of unit's synchronization packets and commands. Unit telemetry description can be found in "Telemetry Packets" chapter. This document also describes software update procedure.

To see installation details please check partnership website

Selected features of Albatross devices

Full description is available on website <http://www.albatross.com.pl> or in printed/digital version of catalogue

- GPRS S8/FM Terminals allows you to connect modules to support advanced transport fleet managing projects; full logistic data reading from the bus CAN/FMS/J1708/OBDII; remote driver cards DDD downloads and mas memory; read drivers status from the bus D8; communication and transport management on a connected GARMIN (FMI) navigation
- CAN module - reads data from the CAN bus (J1939, J1708, J1587, FMS) - over 550 vehicles supported! (Full CAN module description on a separate data sheet)
- TMR module - possibility of remote DDD files + many additional information from tachographs (driver name and surname, VIN, vehicle registration number **without company card authorization!**)
- Connecting to K-Line (D8) - for reading the status of the tachograph and driver card number (on the ignition switched off as well!) Support for all digital tachographs
- GARMIN FMI - Connects to Garmin navigation; Gamin offers a wide range of useful, versatile and economical solution for fleet management
- GSM/GPRS Module: 850/900/1800/1900 MHz SIMCOM/QUECTEL with JAMMING function (EVENT)
- GPS Quectel L70 receiver with ANTY-JAMMING function
- Maintaining the key operating parameters: time and GPS data - Data is never lost - even after a power cut
- Advanced filters of analog inputs (analog floats; fuel probes) with data support after the loss of the main power supply
- External active GPS antenna (3 meter / SMA connector)
- External quad band antenna (3 meter / SMA-RP connector)
- Internal sustaining clock RTC (Real Time Clock)
- External RS-232-TTL (optional; RS-232, RS-485)
- Internal battery 550 mAh (optional); Charging control
- ABS casing: 89x63x30 [mm]
- Memory archive - minimum of 24.000 events (up to 48.000)
- Signaling operation diodes of GSM and GPS
- Full support of 1-Wire DALLAS protocol (6 thermometers, DALLAS / RFID identification)

- Alarm, which informs that the GPS antenna is disconnected - real-time monitoring
- Operating a dedicated Albatross Wi-Fi module which enables a communication between GPRS-S8 terminal and a communicator / tablet using Wi-Fi networks
- Full support for Mobileye technology
- Full solutions support MDAS MOVON
- Authoring solution of protecting inputs of the device (power; ignition switch; analog inputs) - these inputs are damage resistant (e.g. surge in vehicle installation)
- Incremental firmware - backward compatible always contains all the device functions , which ensures ease of managing software versions; Files to update are delivered directly to a partner in order to further updates by using the FTP server
- Open communication TCP protocol and remote support of engineers in real time (Skype) facilitates the process of implementation; Record implementation of the protocol took place in a few days!
- Optional support of 3D sensor in order to detect accident; tows; car accident
- Immodallas - an additional security for customer vehicles. The device activates the output in case of detecting the absence of authorization (DALLAS, RFID). The device allows you to save up to 2047 identification numbers of drivers!
- Ignition signal detection using a dedicated analog input, the main power supply voltage or from CAN bus
- Ability to implement a dedicated logic functions; commands and a help of handling modules on partners demand in case of further cooperation
- Individual settings for partner
- Individual device configuration
- Dedicated commands, functions
- Individual device casing
- Individual device identification

Reading data

- Logistic data from CAN bus: total distance, total fuel consumption, fuel level, engine rev, vehicle speed, engine temperature, engine work time, drive time, operation time of the engine at a standstill, pressure on the gas pedal, vehicle range, instantaneous fuel consumption, axle load.
- Over 30 CAN indicators (check engine, lights, belts, airbag...)

- Sudden acceleration, sudden breakings
- Advanced ECODRIVING module
- Fully automatic vehicle synchronization procedure
- Data from tachograph - current STATUS
- Driver identification from tachograph

Technical specifications

- System supply from + 8V to + 31V
 - The average value of power consumption.
 - Offline terminal (5 minutes after the ignition is turned off):
 - 29 mA +/- 5% for power supply=12.7V
 - 20 mA +/- 5% for power supply=25.4V
- Online terminal (Ignition is on):
 - 54 mA +/- 5% for power supply=12.7V
 - 32 mA +/- 5% for power supply=25.4V
- Terminal in SLEEPMODE:
 - < 5 mA +/- 5% for power supply=12.7V
 - < 3 mA +/- 5% for power supply=25.4V
- Operating temperature -30°C to + 85°C

Inputs/Outputs

- 2 reacting to ground inputs
- 2 open collector outputs
- 3 analog inputs (for ignition voltage measurement; fuel probes)
- 1-Wire Input (identification of drivers; temperature measurement; DALLAS, RFID)
- RS-232-TTL communication port (default)
- RS-232 communication port (optional)
- RS-485 communication port (optional)

- Counting RPM/fuel used analogue input
- CAN J1939/J1708/J1587/FMS/OBDII/TACHO
- K-Line (D8)*

Protocol description

Communication packets can be divided into 3 different groups:

Synchronization Frames

These frames are periodically generated by unit – configured in COMM command. Their main purpose is to authorize protocol with server (should be integrated in first way) and sustain GPRS connection between unit and the GSM network.

Data Frames confirmation

If You will add synchronization frames support - You will receive data frame. For proper work S8 devices require confirmation to every data frame. See data frames description. For every received data frame server must generate value **0xFB 0xF9** - only then device will continue to send next data frame

Communication Commands

This group of packets provides possibility to query and configure unit parameters.

Telemetry Packets

This group of packets provides unit telemetry data.

This chapter contains description of Synchronization Packets and Communication Commands. Telemetry packets are described in chapter “Telemetry Packets”.

Synchronization frames

Synchronization frames are used by device for two main purposes:

Sustaining GPRS connection with network

Detecting TCP/IP protocol connection errors

Synchronization frames sending period can be defined in COMM command. After receiving synchronization frame, telemetry server should echo packet back to the device. If device does not receive echo of 10 following synchronization frames, it will perform emergency reboot operation. If situation repeats, emergency reboot operation will be performed only 3 times. After that unit will not reboot if server is not responding with echo of synchronization frames. This mechanism will be re enabled after units receive any echo of synchronization frame. It is strongly recommended to implement synchronization frames echo functionality in server application.

Unit provides two different synchronization frames formats. Selection of synchronization frame format is done by unit automatically and it depends on the content of UnitID field that can be defined by UNPM command. If UnitID field contains only numerical characters (digits: 0 - 9) unit will use Synchronization Frame Format 1 otherwise unit will use Synchronization Frame Format 2. Below you can find detailed description of Synchronization Frame formats.

Synchronization Frame Format

Below you can find description of Synchronization Frame Format 1. Example shows possible synchronization frame content divided into bytes written in HEX.

	Synchronization Header (16-bit)	Synchronization ID (16-bit)	UnitID (32-bit)
Received Frame Example	0xFA 0xF8	0x29 0x01	0x93 0x39 0x01 0x79

Synchronization Header – this field always contains values 0xFA 0xF8

Synchronization ID – this field contains synchronizations frame counter incremented by each frame, value 0x01 0x29 equals 297

UnitID – this field contains device UnitID, value 0x79 0x01 0x39 0x93 equals 2030123411

UDP Transmission

Devices do not support UDP transmission.

Communication commands

Below there is shown general command format. If not specified otherwise all command parameters are mandatory.

	Command	Parameters	End Markers
Command Example	\$AL+UNPM+03124=	abcd,2030123456,abcd,9998,7,14,15,16,1,50	\r\n

Command – this field consists of always present “\$AL+”, command name “UNPM” and optional command tag field “+03124”

Parameters – this field contains all command parameters

End Markers – command end markets with following values 0x0D 0x0A

Command	Description
3D	Start to generate asynchronous events from 3D sensor.
ACCAUTH	Enable feature of checking authorization after Ignition. Can be used with one of available input in S8 device.
ACTIVEIMMODALLAS	Causes activated JP2-2 output and deactivate this output after read the

	RFID card by RFID card reader
ANALOGFILTER	Configures Analogous filter function
ASYNCHFRAME	Configure special telemetry frame mode for F7 frames
AUTOSYNC (ALBAUTOSYNC)	Starts automatic car type configuration in CAN module
BLACKBOXENABLE	This command provides possibility to set up 3 events. When one of them occurs, the device dumps flash data from 10 seconds ago and 10 seconds after the event
BLACKBOXREAD	This command provides possibility to read data from BLACKBOX. There are 4 slots. Each have 20 frames, which describes every second of registration
CANCMD	Sends raw command to CAN module
CANDIAG	CAN interface diagnostic mode
CANOBD	Method of synchronization of the FM 15 CAN with the CAN bus
CANUPDATE	Update CAN module firmware or download configuration file
CAN	Query CAN module current firmware version
CARSET (ALBCARSET)	Changes current car type configuration in CAN module
CARID (ALBCARID)	Query for current car type configuration in CAN module
COMM	Unit communication parameters configuration
COPS	Check list of the available GSM networks
CHIPERASE	Clears archive memory
DALLASID	Set and Query Dallas ID codes for driver identification
DALLASMODE	Set and query Dallas ID algorithm mode
DRVID	Query for driver cards IDs inserted into tachograph
ECOROLL	ECOROLL feature
FMILIMITS	Set communication limits for FMI protocol

FUELFLAG	Set fuel flag
FRAMEFORMAT	Unit frame format configuration
FRIENDROAM	Forces specified network number or country code to be treated as home network
GEOSPEED	Runs the GEOSPEEDEVENT function
GEOSPEEDEVENT	Configure speed report in specified zone function
GETPOSITION	Query unit current position
GETSERIALNUMBER	Check device factory serial number
IGNITION	Source of ignition information (input, CAN module, main power)
INPUTDELAY	Set delays for device inputs
IMEI	GSM modem IMEI query
LIMITS	Set communication limits
NETLISTADD	Add another GSM operator code to database
NETLISTDELETE	Delete GSM operator code from database
NETLISTREAD	Read GSM operator codes from database
NETLISTENABLE	Enable working of GSM operator list (white list or black list)
OUTS	Set outputs configuration
POWERMODE	Set device power mode
QUST	Unit GSM status query
REBOOT	Reboot unit
SET3D	3D sensor calibration
SETECO	Set Eco driving control parameters
SETENGINEHOURS	Set value from which engine hours will be counting
SETGPSKM	Set mileage counter value

SETPOINTS	Set analog fuel characteristics points
SETRTC	Set RTC clock manually in case of lack GPS signal
SLEEPMODE	Sleeping device with accumulator during main power off
SMID	SIM card serial number query
SMSADDINFO	Set form of the SMS report
SMSPREFIX	Set prefix for SMS report
SMSREPORT	Set Event and SMS report configuration
SOFTUPDATE	Update unit software
SPEEDALARM	Alarm according to speed CAN or GPS – with activated output
STATUS	Query unit current status
SUPANALOG	This command replaces VoltageAnalog1 with VoltageAnalog5
SYNCFRAMEMODE	Set synchronization frame format other than standard
TACHOCARDBLOCK	Enable tachograph card blocking function
TEMPID	Set and query DS18B20 thermometer codes for multiple thermometer support on 1-wire line
TIMER	Enables periodic telemetry reporting
TOW	Set and query towing report settings
TRACKING	Set tracking configuration
TRACKROAM	Unit tracking in roaming configuration
UNPM	Unit primary configuration
VERSION	Query unit software version (communication subsystem)
VIRTUALACC	This command turning on ignition, when vehicle is on move
VOLTDETECT	Enables voltage dropout detection
ZCONFIG	Set various configuration options in I/O processor

ZCONFIG2	Set various configuration options in I/O processor
ZEROLIMITS	Clear communication limits
ZMASK	Set various configuration masks in I/O processor
ZVALUE	Set various configuration values in I/O processor

3D

Command	3D	
Description	Start to generate asynchronous events from 3D sensor. Event will be generated using algorithm – values from 3 axes > this one configured in command parameter.	
Command Syntax	\$AL+3D+[TAG]=[Password],[Limit_for_accident],[Limit_for_car_towing],[Limit_for_acceleration],[Limit_for_braking],[Limit_for_turning]	
Query Answer Syntax	\$QR:3D+[TAG]=[Limit_for_accident],[Limit_for_car_towing],[Limit_for_acceleration],[Limit_for_braking],[Limit_for_turning]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Limit_for_accident	Overload limit for accident. Range: 0 – 128 Value 16 = 1G You have to add 16 to your limit value. Please remember about that!
	Limit_for_car_towing	Overload limit for car towing. Range: 0 – 128 Value 16 = 1G
	Limit_for_acceleration	Overload limit for acceleration. Range: 0 – 128 Value 16 = 1G
	Limit_for_braking	Overload limit for braking. Range: 0 – 128

		Value 16 = 1G
	Limit_for_turning	Overload limit for turning. Range: 0 – 128 Value 16 = 1G
Example Query	Query: \$AL+3D=0000,? Example answer: \$QR:3D=24,2,5,5,6	
Error Answer	\$ER:3D+[TAG]=[ErrorCode]	

ACCAUTH

Command	ACCAUTH	
Description	This command provides possibility to set and query ACC authorization settings. Authorization is performed in the following way. After turning on ACC driver must generate active state on defined input.	
Command Syntax	\$AL+ACCAUTH+[TAG]=[Password],[Enable],[AuthInput],[Time]	
Query Answer Syntax	\$QR+ACCAUTH+[TAG]=[Enable],[AuthInput],[Time]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Enable	0 – Disable 1 – Enable
	AuthInput	1 – JP1-5 2 – JP2-8 4 – JP1-4 5 – JP2-1
	Time	Time to perform authorization. [s]
Example Query	Query: \$AL+ACCAUTH=0000,? Example answer: \$QR:ACCAUTH=1,5,15	
Error Answer	\$ER:ACCAUTH+[TAG]=[ErrorCode]	

ACTIVEIMMODALLAS

Command	ACTIVEIMMODALLAS	
Description	This command provides possibility to activate JP2-2 output and deactivate this output after read the RFID card by RFID card reader.	
Command Syntax	\$AL+ACTIVEIMMODALLAS+[TAG]=[Password]	
Query Answer Syntax	\$OK:ACTIVEIMMODALLAS+[TAG]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
Error Answer	\$ER:ACTIVEIMMODALLAS+[TAG]=[ErrorCode]	

ANALOGFILTER

Command	ANALOGFILTER	
Description	This command provides possibility to set and query Filtration parameters for analog inputs.	
Command Syntax	\$AL+ANALOGFILTER+[TAG]=[Password],[Mode],[Ignition],[BlockingVoltage],[SampleTime],[SampleCount]	
Query Answer Syntax	\$QR+ANALOGFILTER+[TAG]=[Mode],[Ignition],[BlockingVoltage],[SampleTime],[SampleCount]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Mode	0 – Median 1 – Average
	Ignition	If the value is set, ignition state drives analog input actualization system.
	BlockingVoltage	If main power is under this value, terminal is not reading analog inputs. [V]
	SampleCount	Number of samples used in filtration process.

	SampleTime	Time between samples. [s]
Example Query	Query: \$AL+ ANALOGFILTER =0000,? Example answer: \$QR: ANALOGFILTER =0,1,1,5,15	
Error Answer	\$ER: ANALOGFILTER +[TAG]=[ErrorCode]	

ASYNCHFRAME

Command	ASYNCHFRAME	
Description	This command provides possibility to configure special telemetry frame mode for F7 frames. When function is set and large amount of data is stored in archive memory (eg. after long time in roaming network) currently generated frames are send immediately as F8 frames (identical with F7 frames) and do not require frame confirmation. They will be sent later once again, but using standard archive and confirmation mechanism and as usual F7 frames.	
Command Syntax	\$AL+ASYNCHFRAME+[TAG]=[Password],[Enable],[Time]	
Query command	\$AL+ASYNCHFRAME+[TAG]=[Password],?	
Query Answer Syntax	\$QR:ASYNCHFRAME+[TAG]=[Enable],[Time]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Enable	0 – Function disabled 1 – Function enabled
	Time	Enable time. [min]
Example Query	Query: \$AL+ASYNCHFRAME=0000,? \$QR:ASYNCHFRAME=1,30	
Error Answer	\$ER:ASYNCHFRAME+[TAG]=[ErrorCode]	

AUTOSYNC

Command	AUTOSYNC
---------	----------

Description	This command provides possibility to auto detect car type in CAN BUS. Procedure is similar to operation by pressing button located on CAN module. During car detection ignition must be on.	
Command Syntax	\$AL+AUTOSYNC+[TAG]=Password	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
Example Command	\$AL+AUTOSYNC=0000	
Asynchronous report	<p>Unit after performing car auto detection procedure will send asynchronous report with the following syntax: AUTOSYNCH: <result> <result> possible values: 2 – car detection successful 3 – car detection failed (wrong CAN module connection to CAN bus, or car not recognized) 4 – car detection failed (wrong CAN module connection to CAN bus, or no ignition on)</p> <p>NOTE: Report will not be send if CAN module is not present or connected improperly.</p>	
Ok Answer	\$OK:AUTOSYNC+[TAG]	

BLACKBOXENABLE

Command	BLACKBOXENABLE	
Description	This command provides possibility to set up 3 events. When one of them occurs, the device dumps flash data from 10 seconds ago and 10 seconds after the event.	
Command Syntax	\$AL+BLACKBOXENABLE+[TAG]=[Password],[EVENT1],[EVENT2],[EVENT3]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
Example Command	\$AL+BLACKBOXENABLE=0000,3,,27	

CANDIAG

Command	CANDIAG	
Description	This command provides possibility to enable CAN interface diagnostic mode.	
Command Syntax	\$AL+CANDIAG+[TAG]=[Password],[DiagTime]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	DiagTime	Time to enable function. [s]
Example	Command: \$AL+CANDIAG=0000,20 Answer: \$OK:CANDIAG	
Error Answer	\$ER:CANDIAG+[TAG]=[ErrorCode]	

CANOBD

Command	CANOBD	
Description	Method of synchronization of the FM 15 CAN with the CAN bus	
Command Syntax	\$AL+CANOBD+[TAG]=[Password],[Mode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Mode	0 - turned off 1 - every time after turn on ignition or in case of none data while 60 seconds at least 2 - once after first turn on ignition (if you want to install the device in other car model you have to send \$AL+CANOBD=0000,2 before first turn on

		ignition in new car)
Example	Command: \$AL+CANOBD=0000,1 Answer: \$OK:CANOBD	
Error Answer	\$ER:CANOBD+[TAG]=[ErrorCode]	

CANUPDATE

Command	CANUPDATE	
Description	This command provides possibility to update or configure can module firmware.	
Command Syntax	\$AL+CANUPDATE+[TAG]=Password,ftpAdr,ftpUser,ftpPass,file,mode	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	ftpAdr	FTP server address. (max. 29 characters)
	ftpUser	FTP server login. (max. 14 characters)
	ftpPass	FTP server password. (max. 14 characters)
	file	File containing update or configuration. (max. 29 characters) File must be placed in root directory of ftp account.
	mode	1 – firmware update 2 – configuration download 4 – TMR device firmware update
Example Command	\$AL+CANUPDATE=0000,ftp.com.pl,user,pass,file.frm,1	
Asynchronous report	Unit after performing update will send asynchronous report with the following syntax: \$QR:CANUPDATESTATUS=[updateCode] updateCode – 0 – update or configuration successful, 1 – update error	
Ok Answer	\$OK:CANUPDATE+[TAG]	

Error Answer	\$ER:CANUPDATE+[TAG]=[ErrorCode]
--------------	----------------------------------

CANVERSION

Command	CANVERSION	
Description	This command provides possibility to check current firmware version of the CAN module.	
Command Syntax	\$AL+CANVERSION+[TAG]=[Password],?	
Query Answer Syntax	\$QR:CANVERSION+[TAG]=[Version_string]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	Version_string	Character string representing current version of the CAN module firmware. (max. 35 characters)
Example Query	<p>Query: \$AL+CANVERSION=0000,?</p> <p>Example answer: \$QR:CANVERSION=2.1.15</p> <p>NOTE: Answer will not be send if CAN module is not present or connected improperly.</p>	

CARSET

Command	CARSET	
Description	This command provides possibility to auto detect car type in CAN module device. Procedure is similar to operation by pressing button located on CAN module. During car detection ignition must be on. Detailed information is provided in can module documentation.	
Command Syntax	\$AL+CARET+[TAG]=Password,type,car_ID	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)

	type	<p>Command type:</p> <p>16 – setting car from manual selection group</p> <p>17 – setting car from automatic selection group (possible results of car auto detection)</p> <p>18 – erase car setting</p>
	car_ID	<p>Car or group specific id – refer to can module documentation for details and car table.</p> <p>This field must be 0 in case of car erase (18) command</p>
Example Command	\$AL+CARSET=0000,16,18	
Command result	<p>Unit will send command result using following syntax:</p> <p>\$QR:CARSET+[TAG]=type,result</p> <p>type:</p> <p>command type (16, 17 or 18) from corresponding request</p> <p>result:</p> <p>0 – Operation successful</p> <p>1 – Operation failed (unsupported command parameters)</p> <p>NOTE: Answer will not be send if CAN module is not present or connected improperly.</p>	

CARID

Command	CARID	
Description	This command provides possibility to check car type currently set in CAN module device.	
Command Syntax	\$AL+CARID+[TAG]=Password	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
Example Command	\$AL+CARID=0000	
Command result	Unit will send command result using following syntax:	

	<p>\$QR:CARID+[TAG]=type,car_ID</p> <p>type:</p> <p>0 – no car set</p> <p>16 – car set from manual selection group</p> <p>17 – car set from automatic selection group (possible results of car auto detection)</p> <p>car_ID:</p> <p>Car or group specific id – refer to can module documentation for details and car table</p> <p>NOTE: Answer will not be send if CAN module is not present or connected improperly.</p>
--	--

COMM

Command	COMM	
Description	This command provides possibility to set and query unit communication parameters. Unit will reboot in 10s. After changing configuration with this command.	
Command Syntax	\$AL+COMM+[TAG]=[Password],[CommType],[SMSBaseNumber],[R],[GPRS_AP N],[GPRS_Username],[GPRS_Password],[R],[GPRS_HostAddress],[GPRS_HostP ort],[GPRS_SyncInterval],[R]	
Query Answer Syntax	\$QR+COMM+[TAG]=[CommType],[R],[R],[GPRS_APN],[GPRS_Username],[GPRS _Password],[IP_Type],[GPRS_HostAddress],[GPRS_HostPort],[GPRS_SyncInterv al],[R]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
	CommType	Unit primary communication channel for sending telemetry data. Possible values: 0 – Serial RS232 1 – Reserved 2 – SMS 3 – GPRS

	SMSBaseNumber	Number to which SMS will be send with information about generated event (if SMSREPORT is set to report via SMS)
	GPRS_APN	GPRS access point. (max. 34 characters)
	GPRS_Username	GPRS user name. (max. 14 characters)
	GPRS_Password	GPRS password. (max. 14 characters)
	IP_Type	0 – UDP 1 – TCP
	GPRS_HostAddress	Telemetry server address. (max. 29 characters)
	GPRS_HostPort	Telemetry server port.
	GPRS_SyncInterval	Interval between sending synch packets. (min. 10, max. 65535)
	R	Reserved for further use.
Example Query	Query: \$AL+COMM=0000,? Example answer: \$QR:COMM=3,+48602951111,,erainternet,erainternet,erainternet,1,83.31.213.119,10873,40,	
Example Command	\$AL+COMM=0000,3,+48602951111,,erainternet,erainternet,erainternet,1,83.31.213.119,10873,40,	
Ok Answer	\$OK:COMM+[TAG]	
Error Answer	\$ER:COMM+[TAG]=[ErrorCode]	

COPS

Command	COPS	
Description	This command provides possibility to check list of the available GSM networks.	
Command Syntax	\$AL+COPS+[TAG]=[Password]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)

	Password	Unit current password. (max. 4 characters)
Example Query	Query: \$AL+COPS=0000 Example answer: \$QR:COPS=+COPS: (2,"IDEA Centertel","IDEA","26003"),(3,"Era","Era","26002"),(3,"Plus GSM","PLUS","26001"),(1,"26006","26006","26006"),,(0-4),(0-2)	
Error Answer	\$ER:COPS+[TAG]=[ErrorCode]	

CHIPERASE

Command	CHIPERASE	
Description	This command provides possibility to clear internal archive FLASH memory. NOTE: Device restarts after completion of erase	
Command Syntax	\$AL+CHIPERASE+[TAG]=[Password]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (max. 5 characters)
	Password	Unit current password. (max. 4 characters)
Example Command	Command: \$AL+CHIPERASE=0000 Answer: \$OK:CHIPERASE	
Error Answer	\$ER:CHIPERASE+[TAG]=[ErrorCode]	

DALLASID

Command	DALLASID	
Description	This command provides possibility to set and query Dallas IDs authorized for turning ignition on. Authorization checking can be enabled using proper ZCONFIG command setting.	
Command Syntax	\$AL+DALLASID+[TAG]=[Password],[index],[ID] NOTE: Device will respond sending query with selected index. If new ID setting is unsuccessful ID field will be empty or contain old value	
Query command	\$AL+DALLASID+[TAG]=[Password],[index],?	
Query and Command	\$QR:DALLASID+[TAG]=[index],[ID]	

Answer Syntax		
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	index	Number of slot to read/write ID code. Value range: 1-2047
	ID	Dallas Button ID code presented in ASCII hex format. Code is always 16 characters (8 bytes)
Example Command	Command: \$AL+DALLASID=0000,1,3100001375041A01 Response: \$QR:DALLASID=1,3100001375041A01	
Example Query	Query: \$AL+DALLASID=0000,1, \$QR:DALLASID=1,3100001375041A01	

DALLASMODE

Command	DALLASMODE	
Description	This command provides possibility to set and query DALLAS ID authorization algorithm mode	
Command Syntax	\$AL+DALLASMODE+[TAG]=[Password],[Mode]	
Query Answer Syntax	\$QR+DALLASMODE+[TAG]=[Mode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	mode	0 – default 1 – Dallas authorization is possible only when ignition is turned on.
Example Query	Query: \$AL+DALLASMODE=0000, Example answer: \$QR:DALLASMODE=0	
Example Command	\$AL+DALLASMODE=0000,1	
Ok Answer	\$OK: DALLASMODE +[TAG]	

Error Answer	\$ER: DALLASMODE+[TAG]=[ErrorCode]
--------------	------------------------------------

DRVID

Command	DRVID	
Description	This command provides possibility to query driver cards IDs from tachograph using CAN module. DRVID answer packets are generated asynchronously when card state is changed. DRVID answer packets are also stored in telemetry archive.	
Query command	\$AL+DRVID+[TAG]=[Password],?	
Query Answer Syntax	\$QR:DRVID+[TAG]=[Time],[ID1],[ID2]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Time	Current date and time (UTC)
	ID1	Driver card ID inserted into first slot (max 30 characters)
	ID2	Driver card ID inserted into second slot (max 30 characters)
Example Query	Query: \$AL+DRVID=0000,? \$QR:DRVID=20100624105054,PL123456789*,PL234567890* NOTE: Reports will not be sending if CAN module is not present or connected improperly.	
Error Answer	\$ER:DRVID+[TAG]=[ErrorCode]	

ECOROLL

Command	DRVID	
Description	This feature provides possibility to generating ECOROLL events when the vehicle is driving above the defined vehicle speed with lower engine speed that is defined for START ECOROLL, or STOP ECOROLL if one of the requirements will not be fulfilled.	
Query command	\$AL+ECOROLL+[TAG]=[Password],?	

Query Answer Syntax	\$QR:ECOROLL+[TAG]=[Speed],[RPM]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Speed	Minimum vehicle speed
	RPM	Maximum engine speed
Example Query	Query: \$AL+ECOROLL=0000,? \$QR:ECOROLL=90,650	
Error Answer	\$ER:ECOROLL+[TAG]=[ErrorCode]	

FMILIMITS

Command	FMILIMITS	
Description	This command provides possibility to set and query unit limits parameters. Setting limits values to 0 disables limits control. ZEROLIMITS command resets counter values.	
Command Syntax	\$AL+FMILIMITS+[TAG]=[Password],[HomeDayBytes],[HomeMonthBytes],[RoamDayBytes],[RoamMonthBytes]	
Query Answer Syntax	\$QR:FMILIMITS+[TAG]=[HomeDayBytes],[HomeMonthBytes],[RoamDayBytes],[RoamMonthBytes],[HomeDayBytesSend],[HomeMonthBytesSend],[RoamDayBytesSend],[RoamMonthBytesSend]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	HomeDayBytes	Day limit of FMI data (bytes) send by TerminalS8 in home network
	HomeMonthBytes	Month limit of FMI data (bytes) send by TerminalS8 in home network
	RoamDayByte	Day limit of FMI data (bytes) send by TerminalS8 in roaming

	s	network
	RoamMonthBytes	Month limit of FMI data (bytes) send by TerminalS8 in roaming network
	HomeDayBytesSend	Amount of FMI data bytes send by TerminalS8 during current day in home network.
	HomeMonthBytesSend	Amount of FMI data bytes send by TerminalS8 during current month in home network.
	RoamDayBytesSend	Amount of FMI data bytes send by TerminalS8 during current day in roaming network.
	RoamMonthBytesSend	Amount of FMI data bytes send by TerminalS8 during current month in roaming network.
Example Query	Query: \$AL+FMILIMITS=0000,? Example answer: \$QR:FMILIMITS=20,10,3,0,0,0,0	
Example Command	\$AL+FMILIMITS=0000,20,10,5	

FUELFLAG

Command	FUELFLAG	
Description	This command adds fuel unit indicator to telemetry frames.	
Command Syntax	\$AL+FUELFLAG+[TAG]=[Password],[enable]	
Query command	\$AL+FUELFLAG+[TAG]=[Password],?	
Query Answer Syntax	\$QR:FUELFLAG+[TAG]=[enable]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	enable	0 – Function disabled 1 – Function enabled

		<p>When enabled function will add to F2 and F4 frames additional field indicating unit of [Total_fuel] field.</p> <p>Values of additional field:</p> <p>P – fuel is in percent's</p> <p>L – fuel is in liters</p> <p>S – fuel is substituted from analog fuel probe</p> <p>empty field – no information</p>
Example Query	<p>Query: \$AL+FUELFLAG=0000,?</p> <p>Example answer: \$QR:FUELFLAG=1</p>	
Error Answer	\$ER:FUELFLAG+[TAG]=[ErrorCode]	

FRAMEFORMAT

Command	FRAMEFORMAT	
Description	This command provides possibility to set and query unit telemetry format settings.	
Command Syntax	\$AL+FRAMEFORMAT+[TAG]=[Password],[TmFormat]	
Query Answer Syntax	\$QR:FRAMEFORMAT+[TAG]=[TmFormat]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	TmFormat	<p>Possible telemetry formats:</p> <p>F0 – telemetry without CAN data</p> <p>F1 – telemetry with CAN data</p> <p>F2 – telemetry with extended CAN data</p> <p>F4 – extended telemetry with CAN data, 1-wire and analog inputs data</p> <p>F7 – more extended telemetry with full range of CAN data, 1-wire analog inputs data, eco driving, 3d sensors, tachograph drivers ID. [device will send F7 data frames over TCP. Via SMS will send F77 data frame. See description of F77 data frame in Telemetry packets chapter]</p>

		<p>F12 – telemetry the same like in the case of F7 frame format with additional eco driving fields and others</p> <p>F16 – telemetry the same like in the case of F7 frame format with four additional fields: coasting, pto_duration, pto_fuel_consumption, clutch_presses, retarder_usage.</p> <p>See frame format descriptions for telemetry details</p>
Example Query	<p>Query: \$AL+FRAMEFORMAT=0000,?</p> <p>Example answer: \$QR:FRAMEFORMAT=F1</p>	
Example Command	\$AL+FRAMEFORMAT=0000,F0	
Ok Answer	\$OK:FRAMEFORMAT+[TAG]	
Error Answer	\$ER:FRAMEFORMAT+[TAG]=[ErrorCode]	

FRIENDROAM

Command	FRIENDROAM	
Description	This command forces specified network number or country code to be treated as home network	
Command Syntax	\$AL+FRIENDROAM+[TAG]=[Password],[enable],[networkCode]	
Query Answer Syntax	\$QR+FRIENDROAM+[TAG]=[enable],[networkCode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	enable	<p>1 – Enable function (single network)</p> <p>2 – Enable function (all networks matching country code)</p> <p>0 – Disable function</p>
	Network Code	<p>For option 1:</p> <p>5 – digit GSM network code for network that will be treated as home network (e.g. 26001 for PLUS-GSM)</p> <p>For option 2:</p> <p>3 – digit prefix GSM country code for all networks in country</p>

		that will be treated as home network (e.g. 260 for Poland)
Error Answer	\$ER:FRIENDROAM+[TAG]=[ErrorCode]	

GEOSPEED

Command	GEOSPEED	
Description	Runs the GEOSPEEDEVENT function	
Command Syntax	\$AL+GEOSPEED+[TAG]=0000,[Mode]	
Query command	\$AL+GEOSPEED+[TAG]=[Password],?	
Query Answer Syntax	\$QR:GEOSPEED+[TAG]=[Mode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	0 – function disabled 1 – function enabled 2 – function enabled with activating outputs when vehicle speed is exceeded
Example Query	Query: \$AL+GEOSPEED=0000,? \$QR:GEOSPEEDEVENT=1	
Error Answer	\$ER:GEOSPEED+[TAG]=[ErrorCode]	

GEOSPEEDEVENT

Command	GEOSPEEDEVENT	
Description	This command provides possibility to enable and query speed report in specified zone function	
Command Syntax	\$AL+GEOSPEEDEVENT+[TAG]=[Password],[Event_ID][Mode],[Speed_source],[Speed],[output],[X1],[Y2],[X2],[Y2],...,[X10],[Y10]	
Query command	\$AL+GEOSPEEDEVENT+[TAG]=[Password],?	
Query Answer Syntax	\$QR:GEOSPEEDEVENT+[TAG]=[Event_ID][Mode],[Speed_source],[Speed],[out	

	put],[X1],[Y2],[X2],[Y2],...,[X10],[Y10]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Event_ID	Configurable Event_ID of generated frame Range: 110-150
	Mode	0 – disabled 1 – enabled with checking vehicle speed only eventid for start exceeding vehicle speed – 256 + inserted event_id eventid for end exceeding vehicle speed – 384 + inserted event_id 2 – enabled with checking presence in zone only eventid for enter to zone – 256 + inserted event_id eventid for exit from zone – 384 + inserted event_id 3 – enabled with checking vehicle speed and presence in zone (mode 1 + mode2) eventid for enter to zone – 256 + inserted event_id eventid for exit from zone – 384 + inserted event_id eventid for start exceeding vehicle speed – 512 + inserted event_id eventid for end exceeding vehicle speed – 640 + inserted event_id
	Speed_source	0 – speed from GPS 1 – speed from CAN
	Speed	Speed value in km/h above which event will be generated
	Output	Output which will be activated when speed limit is crossed bit 0 – JP2-2 (S8) + JP1-2 (FM10/15/23) bit 1 – JP2-1 (S8) bit 2 – JP1-2 (S8) bit 3 – Expander Output 1 bit 4 – Expander Output 2
	[X1],[Y2],[X2],[Y2],...,[X10],[Y10]	Up to 10 points defining speed limit zone. Note: all comas must be inserted even when less than 10 points are defined

Example Query	Query: \$AL+GEOSPEEDEVENT=0000,? \$QR:GEOSPEEDEVENT=110,1,1,50,2,20.123456,50.123456,20.223456,50.123456,20.223456,50.23456,20.123456,50.223456,,,,,,,,,
Error Answer	\$ER:GEOSPEEDEVENT+[TAG]=[ErrorCode]

GETPOSITION

Command	GETPOSITION	
Description	This command provides possibility to query unit current position and other telemetry parameters.	
Command Syntax	\$AL+GETPOSITION+[TAG]=[Password],?	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
Example Query	Query: \$AL+GETPOSITION=0000,? Example answer: 2030123457,20100217104434,F1,21.022840,52.205989,0,0,0,8,0,670,0,00.00,00.00,0,,,,,,,,,0	

GETSERIALNUMBER

Command	GETSERIALNUMBER	
Description	This command provides possibility to check device factory serial number	
Command Syntax	\$AL+GETSERIALNUMBER+[TAG]=[Password]	
Answer Syntax	\$QK:GETSERIALNUMBER=[SN]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	SN	Serial number string
Example	Query: \$AL+GETSERIALNUMBER=0000 \$QK:GETSERIALNUMBER=12345	

Error Answer	\$ER:GETSERIALNUMBER+[TAG]=[ErrorCode]
--------------	--

IGNITION

Command	IGNITION	
Description	This command provides possibility to read ignition information from different source.	
Command Syntax	\$AL+IGNITION+[TAG]=[Password],[mode],[time_delay],[voltage_value]	
Query Answer Syntax	\$QR:IGNITION+[TAG]=[mode],[time_delay],[voltage_value]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	mode	<p>0 – Default mode. Ignition will be read from ignition wire (JP1-3 ignition input (+)).</p> <p>1 – Ignition will be read from CAN BUS. Check supported list to see if ignition should available.</p> <p>2 – Ignition will be read from main power voltage. Set time of delay and voltage_value.</p> <p>3 – Dedicated customer feature.</p> <p>7 - If you set 7 mode then in the <i>[voltage_value]</i> field you should insert vehicle speed – 05.00 means 5 km/h. The speed is limit to turn ignition off. When vehicle speed will be lower than set in <i>[voltage_value]</i> within time defined in <i>[time_delay]</i>.</p>
	Time_delay	Delay value in 1s, e.g. value 50 = 50s (Min = 1, Max = 32767)
	Voltage_value	<p>Voltage value in xx.xx format.</p> <p>E.G.</p> <p>05.08 (5,08V)</p> <p>12.50 (12,50V)</p>
Example Command	\$AL+IGNITION=0000,1,0,00.00 \$AL+IGNITION=0000,2,5,12.50	
Example Query	Query: \$AL+IGNITION=0000,? \$QR:IGNITION=1,0,00.00	
Ok Answer	\$OK:IGNITION+[TAG]	

Error Answer	\$ER:IGNITION+[TAG]=[ErrorCode]
--------------	---------------------------------

INPUTDELAY

Command	INPUTDELAY	
Description	This command provides possibility to set delay for every input of the device	
Command Syntax	\$AL+INPUTDELAY+[TAG]=[Password],[Input],[Delay]	
Query Answer Syntax	\$QR:INPUTDELAY+[TAG]=[Delay0],[Delay1],[Delay2],[Delay3],[Delay4],[Delay5],[Delay6],[Delay7],[Delay8]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Input	Input bit number (see Inputs Field Description for details) Allowed values: 0,1,2,3,4,5,6,7,8
	Delay	Delay value in 100ms, e.g. value 50 = 5000ms (Min = 1, Max = 32767, Default = 7) Value 254 = state of input will be always 0 Value 255 = state of input will be always 1
Example Command	\$AL+INPUTDELAY=0000,1,35 (3500ms delay on bit number 1 input)	
Example Query	Query: \$AL+INPUTDELAY=0000,? \$QR:INPUTDELAY=7,7,7,7,7,7,7,7,7	
Ok Answer	\$OK:INPUTDELAY+[TAG]	
Error Answer	\$ER:INPUTDELAY+[TAG]=[ErrorCode]	

IMEI

Command	IMEI	
Description	This command provides possibility to query unit IMEI number	
Command Syntax	\$AL+IMEI+[TAG]=[Password],?	
Field Description	TAG	Command specific TAG number. If present unit answer

		must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
Example Query	Query: \$AL+IMEI=0000,? Example answer: \$QR:IMEI=356950005005364	

LIMITS

Command	LIMITS	
Description	This command provides possibility to set and query unit limits parameters. Setting limits values to 0 disables limits control. ZEROLIMITS command resets counter values.	
Command Syntax	\$AL+LIMITS+[TAG]=[Password],[HomeGPRS],[RoamGPRS],[SMS]	
Query Answer Syntax	\$QR:LIMITS+[TAG]=[HomeGPRS],[RoamGPRS],[SMS],[HomeGPRSCount],[RoamGPRSCount],[SMSCount],[SMSTotal]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	HomeGPRS	Day limit of GPRS session open attempts for home network. Max value = 200.
	RoamGPRS	Day limit of GPRS session open attempts for roaming networks. Max value = 200.
	SMS	Day limit of SMS. Max value = 200.
	HomeGPRSCo unt	GPRS session open attempts counter for home network
	RoamGPRSCo unt	GPRS session open attempts counter for roaming networks
	SMSCount	SMS counter
	SMSTotal	SMS send during current month. Device will not send more than 100 sms within one month.
Example Query	Query: \$AL+LIMITS=0000,?	

	Example answer: \$QR:LIMITS=20,10,3,0,0,0,0
Example Command	\$AL+LIMITS=0000,20,10,5
Ok Answer	\$OK:LIMITS+[TAG]
Error Answer	\$ER:LIMITS+[TAG]=[ErrorCode]

NETLISTADD

Command	NETLISTADD	
Description	This command provides possibility to add GSM operator codes to white or black lists. To start controlling GSM operators code User need to enable option in command NETLISTENABLE.	
Command Syntax	\$AL+NETLISTADD+[TAG]=[Password],[list_number],[GSMOPERATOR_CODE],[GSMOPERATOR_CODE], [GSMOPERATOR_CODE],....	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	List_number	0 – list with GSM operators that will be blocked (black list) 1 – list with GSM operators whom device will connect to GPRS (white list)
	GSMOPERATOR_CODE	26002 – this is example of GSM operator code – mainly contain 5 digits there three first mean country code. 260 is Poland. 262 Germany and so on....
Example Command	\$AL+NETLISTADD=0000,1,26001,26002,26003	
Ok Answer	\$OK:NETLISTADD+[TAG]	
Error Answer	\$ER:NETLISTADD+[TAG]=[ErrorCode]	

NETLISTDELETE

Command	NETLISTDELETE
---------	---------------

Description	This command provides possibility to delete GSM operator codes from white or black list.	
Command Syntax to delete selected GSM operator code	\$AL+NETLISTADD+[TAG]=[Password],[list_number],[GSMOPERATOR_CODE],[GSMOPERATOR_CODE],[GSMOPERATOR_CODE],....	
Command Syntax to delete all GSM operator codes from list	\$AL+NETLISTADD+[TAG]=[Password],[list_number],0	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	List_number	0 – list with GSM operators that will be blocked (black list) 1 – list with GSM operators whom device will connect to GPRS (white list)
	GSMOPERATOR_CODE	26002 – this is example of GSM operator code – mainly contain 5 digits there three first mean country code. 260 is Poland. 262 Germany and so on....
Example Command	\$AL+NETLISTDELETE=0000,1,26001,26002,26003 \$AL+NETLISTDELETE=0000,1,0	
Ok Answer	\$OK:NETLISTDELETE+[TAG]	
Error Answer	\$ER:NETLISTDELETE+[TAG]=[ErrorCode]	

NETLISTREAD

Command	NETLISTREAD
Description	This command provides possibility to read saved GSM operator codes to white or black lists.
Command Syntax	\$AL+NETLISTREAD+[TAG]=[Password],[List_number],[Group_of_selected_8_codes]

Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	List_number	0 – list with GSM operators that will be blocked (black list) 1 – list with GSM operators whom device will connect to GPRS (white list)
	Group_of_selected_codes	0 – first group with 8 GSM codes 1 – second group with 8 GSM codes 2 – third group with 8 GSM codes ...
Example Command	\$AL+NETLISTREAD=0000,1,1	
Ok Answer	\$OK:NETLISTREAD+[TAG]	
Error Answer	\$ER:NETLISTREAD+[TAG]=[ErrorCode]	

NETLISTENABLE

Command	NETLISTENABLE	
Description	This command provides possibility to enable working of white or black list. User must select between white or black list. To operate with NETLIST use NETLISTADD or NETLISTDELETE commands. To read actual black or white list use NETLISTREAD command.	
Command Syntax	\$AL+NETLISTENABLE+[TAG]=[Password],[enable],[List_number],[mode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Enable	0 – NETLIST command not working 1 – NETLIST command is working 2 – Semi-automatic logging for 2G / 3G modems Principle of operation: Modems logs card into network automatically and tries to connect with GPRS. If the attempt fails, modem adds network temporarily to the blacklist and selects next available. Correct login into GPRS clears blacklist, trying to connect to all available networks also clears blacklist, so you can start browsing it again. This way guarantees GPRS

		browsing in all networks that are available for the SIM card. If you select this mode, the List_number and Mode fields do not matter (you can enter 0).
	List_number	0 – list with GSM operators that will be blocked (black list) 1 – list with GSM operators whom device will connect to GPRS (white list)
	Mode	0 – Device will use full code e.g. 26002 1 – Device will use country codes 26002
Example Command	\$AL+NETLISTENABLE=0000,1,1,1	
Ok Answer	\$OK:NETLISTENABLE+[TAG]	
Error Answer	\$ER:NETLISTENABLE+[TAG]=[ErrorCode]	

OUTS

Command	OUTS	
Description	This command provides possibility to control device outputs.	
Command Syntax	\$AL+OUTS+[TAG]=[Password],[OutputID],[OutputControl],[Duration],[Toggle Times]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	OutputID	Output number: 1 – output 1 JP2-2 2 – output 2 JP2-1 3 – output 3 not available 4 – output 4 JP4-3 (available if the EXPANDER is used) 5 – output 5 JP4-2 (available if the EXPANDER is used)
	OutputControl	0 – disabled 1 – enabled

	Duration	Output activation time in 100ms units. Setting to 0 means continuous activation. (Min. 1, Max. 255)
	Toggle Times	Output state toggle times. (Max. 255) – not used
Example Command	\$AL+OUTS=0000,1,1,50,0	
Ok Answer	\$OK:OUTS+[TAG]	
Error Answer	\$ER:OUTS+[TAG]=[ErrorCode]	

POWERMODE

Command	POWERMODE	
Description	<p>This command provides possibility to set device power mode.</p> <p>Important!</p> <p>Green diode will blinks every 3 seconds if the device is in the mode that causes that the GSM modem is turned off.</p>	
Command Syntax	\$AL+POWERMODE+[TAG]=[Password],[powerMode]	
Query Answer Syntax	\$QR:POWERMODE+[TAG]=[powerMode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	<p>0 – GPS=OFF, GSM=ON, CAN=ON, power consumption is 26mA when the main power is 12.8V</p> <p>1 – GPS=ON, GSM=ON, CAN=ON, power consumption is 39mA when the main power is 12.8V</p> <p>2 – GPS=OFF, GSM=OFF, CAN=ON, power consumption is 19mA when the main power is 12.8V</p> <p>3 – Dedicated customer feature</p> <p>4 – GPS=OFF, GSM=ON, CAN=OFF, power consumption is 21mA when the main power is 12.8V</p> <p>5 – GPS=ON, GSM=ON, CAN=OFF, power consumption is 34mA when the main power is 12.8V</p> <p>6 – GPS=OFF, GSM=OFF, CAN=OFF, power consumption is 14mA when the main power is 12.8V</p> <p>7 – GPS=ON, GSM=OFF, CAN=OFF, power consumption is 26mA when the main power is 12.8V</p>

		<p>Important!</p> <p>Remember please that choosing mode that will cause that the GSM module go to sleep cause that you have to set waking up the GSM module for events that can appear when the ignition is turned off. Otherwise, turning off the ignition will cause none of data frames.</p> <p>The same is in case of the GPS module. If you want to refresh position in each data frame on the stop you should set waking up the GPS module.</p>
Example Query	Query: \$AL+POWERMODE=0000,? \$QR:POWERMODE=1	
Error Answer	\$ER:POWERMODE+[TAG]=[ErrorCode]	

QUST

Command	QUST	
Description	This command provides possibility to query unit connection parameters.	
Command Syntax	\$AL+QUST+[TAG]=[Password],?	
Query Answer Syntax	\$QR+QUST+[TAG]=[OperatorCode],[Signal],[ConnectionState]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	OperatorCode	GSM operator code. (Max. 15 characters)
	Signal	GSM signal strength, range 0-31
	ConnectionState	Connection to server state: 0 – disconnected 1 – connected
Example Query	Query: \$AL+QUST=0000,? Example answer: \$QR:QUST=26001,29,1	
Error Answer	\$ER:QUST+[TAG]=[ErrorCode]	

REBOOT

Command	REBOOT
---------	--------

Description	This command provides possibility to perform unit reboot procedure. Command causes complete power down of GSM and GPS module for a few seconds. I/O and CAN module is not restarted.	
Command Syntax	\$AL+REBOOT+[TAG]=[Password]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
Example Command	\$AL+REBOOT=0000	
Ok Answer	\$OK:REBOOT+[TAG]	
Error Answer	\$ER:REBOOT+[TAG]=[ErrorCode]	

SET3D

Command	SET3D	
Description	This command provides possibility to calibrate 3D internal sensor. Just after sending this command 3D sensor will recognize directions while driving – forward, backwards, left and right. Calibration of 3D sensor is required only for AES feature.	
Command Syntax	\$AL+SET3D+[TAG]=[Password]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	mode	0 – Device will find itself the most horizontal surface 1 – Device will use current position as the most horizontal surface
Example Command	\$AL+SET3D=0000,0	
Ok Answer	\$OK:SET3D+[TAG]	
Error Answer	\$ER:SET3D+[TAG]=[ErrorCode]	

SETECO

Command	SETECO	
Description	This command configures Eco driving control parameters	
Command Syntax	\$AL+SETECO+[TAG]=[Password],[enable][AT],[AS],[BT],[BS]	
Query command	\$AL+SETECO+[TAG]=[Password],?	
Query Answer Syntax	\$QR:SETECO+[TAG]=[enable][AT],[AS],[BT],[BS]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	enable	0 – Eco driving default enabled 1 – Dedicated to Customer NOTE: setting ATAS, BT, BS parameters also possible when function disabled (0). This will affect [Rapid_accelerations] and [Rapid_brakings] value counting.
	AT,AS,BT,BS	CAN specific parameters used for rapid accelerations and braking are calculation. Refer to CAN Module documentation for details
Example Command	\$AL+SETECO=0000,1,10,50,10,50	
Ok Answer	\$OK:SETECO+[TAG]	
Error Answer	\$ER:SETECO+[TAG]=[ErrorCode]	

SETENGINEHOURS

Command	SETENGINEHOURS	
Description	This command configures a value from which engine hours will be counting.	
Command Syntax	\$AL+SETENGINEHOURS+[TAG]=[Password],[value]	
Query command	\$AL+SETENGINEHOURS+[TAG]=[Password],?	
Query Answer Syntax	\$QR:SETENGINEHOURS+[TAG]=[value]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)

	Password	Unit current password. (Max. 4 characters)
	value	Value from which engine hours will be counting (maximum value unsigned 32 bit)
Example Command	\$AL+SETENGINEHOURS=0000,1000	
Ok Answer	\$OK:SETENGINEHOURS+[TAG]	
Error Answer	\$ER:SETENGINEHOURS+[TAG]=[ErrorCode]	

SETGPSKM

Command	SETGPSKM	
Description	This command provides possibility to set GPS mileage counter value.	
Command Syntax	\$AL+SETGPSKM+[TAG]=[Password],[milValue],[mode]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mil Value	New mileage starting value in kilometers (for mode 1) meters (for mode 3)
	mode	GPS mileage counter mode: 0 – use mileage data from CAN module 1 – use incremental mileage data from GPS (unit: 1km) 2 – use differential mileage data from GPS (unit: 1m) Mileage value is counted from last frame generated. 3 – use incremental mileage data from GPS (unit: 1m) 4 – Use total mileage from CAN in total_distance, in mileageGPS use differential between generated data frames. 5 – Use total mileage from CAN in total_distance, use incremental mileage data from GPS (unit: 1m)
Example Command	\$AL+SETGPSKM=0000,47324,1	
Ok Answer	\$OK:SETGPSKM+[TAG]	
Error Answer	\$ER:SETGPSKM+[TAG]=[ErrorCode]	

SETPOINTS

Command	SETPOINTS	
Description	This command provides possibility to set and query fuel analog characteristics points. It is possible do define max 50 points at a time. Defining total 32 points is possible First 16 points are reserved for analog input 1, while another 16 for Analog input 2. Points are sorted automatically by the system and placed in appropriate order in the internal memory. Defining point with voltage value that currently exists will replace existing point.	
Command Syntax	<ul style="list-style-type: none"> – \$AL+SETPOINTS+[TAG]=[Password],[X1],[Y1],[X2],[Y2]... - sets calibration points – \$AL+SETPOINTS+[TAG]=[Password],[index] - resets stored fuel probe points – \$AL+SETPOINTS+[TAG]=[Password],[index],? - query all calibration data points for fuel probe 	
Command Answer	<ul style="list-style-type: none"> – \$OK:SETPOINTS+[TAG] – \$OK:SETPOINTS+[TAG] – \$QR:SETPOINTS+[TAG]= [X1],[Y1],[X2],[Y2]... 	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	X1, Y1, X2, Y2, ...	<p>Pairs of points coordinate (Up to 16 pairs).</p> <p>X1 – point voltage e.g. 1234 means 12,34V</p> <p>Y1 – corresponding percent of fuel level range 0 – 1000 for first input range 2000-3000 for second input (2000 added) e.g. 450 means 45,0% (for first input) 2450 means 45,0% (for second input)</p> <p>Single SETPOINTS command can contain mixed points for both inputs.</p>
	index	<p>Number of fuel tank:</p> <p>0 – fuel probe connected to JP1-5 9 – fuel probe connected to JP2-8</p>
Error Answer	\$ER:SETPOINTS+[TAG]=[ErrorCode]	

SETRTC

Command	SETRTC	
Description	This command provides possibility to set RTC clock manually in case of lack GPS signal.	
Command Syntax	\$AL+SETRTC+[TAG]=[Password],[Time]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Time	RTC time in format similar to field [DateTime] in data frames (YYYYMMDDHHMMSS). Time is UTC (GMT+0 time zone)
Example Query	Query: \$AL+SETRTC=0000,20141212090823 \$QR:SETRTC	
Error Answer	\$ER:SETRTC+[TAG]=[ErrorCode]	

SLEEPMODE

Command	SLEEPMODE	
Description	Device can go to deep sleep mode, where power consumption is minimum. Device will wake up after time set in command to send one actual data frame (with refreshed GPS position, CAN data...).	
Command Syntax	\$AL+SLEEPMODE+[TAG]=[Password],[enable],[time],[sleep_cause],[3D_sensor_wakeup_threshold]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	mode	0 – Disabled 1 – Device will generate data frames only according to 3D overloads and in case of sabotage (sabotage means that JP1-5 input is active and data frames will be generated every 30 seconds while this) 2 – Device will generate data frames only at defined time 3 – Device will generate data frames only according to 3D overloads and in case of

		change state of one of inputs 5 - If you set 5 mode then a device wake up according to 3D sensor.
	time	Time of sleep duration. Value in minutes. Minimum time 10 minutes.
	sleep_cause	0 – Device will go to sleep when the main power will turn off 1 – Device will go to sleep when the ignition will turn off
	3D_sensor_wakeup_threshold	3D sensor threshold to generate asynchronous data frame when threshold will be exceeded. One of values 3DX, 3DY, 3DZ have to be exceeded at least by value that is inserted into 3D_sensor_wakeup_threshold.
	Password	Unit current password. (Max. 4 characters)
Example Query	Query: \$AL+SLEEPMODE=0000,? Example answer: \$QR:SLEEPMODE=1,10,1,3	

SMID

Command	SMID	
Description	This command provides possibility to query SIM card SMID number	
Command Syntax	\$AL+SMID+[TAG]=[Password],?	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
Example Query	Query: \$AL+SMID=0000,? Example answer: \$QR:SMID=8948020509036806915	

SMSADDINFO

Command	SMSADDINFO
Description	This command provides possibility to set and query unit tracking SMS and

	tracking event reporting.	
Command Syntax	\$AL+SMSADDINFO+[TAG]=[Password],[Enable]	
Query Answer Syntax	\$QR:SMSADDINFO+[TAG]=[Enable]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Enable	0 – Disabled 1 – Enabled 2 – Enabled (sending sms contains event id instead of telemetry frame)
Example Query	Query: \$AL+SMSADDINFO=[Password],? Example answer: \$QR:SMSADDINFO=1	
Example Command	\$AL+SMSADDINFO=0000,0	
Ok Answer	\$OK:SMSADDINFO+[TAG]	
Error Answer	\$ER:SMSADDINFO+[TAG]=[ErrorCode]	

SMSPREFIX

Command	SMSPREFIX	
Description	This command provides possibility to set and query unit tracking SMS and tracking event reporting.	
Command Syntax	\$AL+SMSPREFIX+[TAG]=[Password],[Reserved],[Prefix]	
Query Answer Syntax	\$QR:SMSPREFIX+[TAG]=[Reserved],[Prefix]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Prefix	String (max 20 characters)
Example Query	Query: \$AL+SMSPREFIX=[Password],? Example answer: \$QR:SMSPREFIX=0,test1_	

Example Command	\$AL+SMSADDINFO=0000,0,test1_
Ok Answer	\$OK:SMSADDINFO+[TAG]
Error Answer	\$ER:SMSADDINFO+[TAG]=[ErrorCode]

SMSREPORT

Command	SMSREPORT	
Description	This command provides possibility to set and query unit tracking SMS and tracking event reporting. See EVENTID [Report ID] Codes Description to check meaning of each EVENT.	
Command Syntax	\$AL+SMSREPORT+[TAG]=[Password],[index],[Enable],[SMSHomeEnable],[SMSRoamEnable]	
Query Answer Syntax	\$QR:SMSREPORT+[TAG]=[Enable],[SMSHomeEnable],[SMSRoamEnable]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	index	Event id (Min. 0, Max. 1364)
	Enable	<p>Bit 0 = data frame generating 0 – not allowed 1 – allowed Bit 1 = not used Bit 2 = impulse generating over 1 second on the output 2 Bit 3 = not used Bit 4 = GPS waking up 0 – not allowed 1 – allowed Bit 5 = GSM waking up 0 – not allowed 1 – allowed Bit 6 = not used Bit 7 = not used</p> <p>For example: 00110001 (2) -> 49 (10) \$AL+SMSREPORT=0000,3,49,0,0 The command that is given above cause that the TIMER feature has permission to generate data frames and waking</p>

		up GPS and GSM modules.
	SMSTHomeEnable	0 – Disabled 1 – Enabled (sending sms in home network when event [index] appears) SMS is generate only where there is no connection with TCP server 2 – Enabled (sending sms in home network when event [index] appears) SMS from this event is always generated
	SMSTRoamEnable	0 – Disabled 1 – Enabled (sending sms in Roaming network when event [index] appears) SMS is generate only where there is no connection with TCP server 2 – Enabled (sending sms in home network when event [index] appears) SMS from this event is always generated
Example Query	Query: \$AL+SMSREPORT=[Password],11, Example answer: \$QR:SMSREPORT=1,0,0	
Example Command	\$AL+SMSREPORT=0000,11,1,1,1	
Ok Answer	\$OK:SMSREPORT+[TAG]	
Error Answer	\$ER:SMSREPORT+[TAG]=[ErrorCode]	

SOFTUPDATE

Command	SOFTUPDATE	
Description	This command provides possibility to update unit firmware. Parameters in this command are optional and if not provided unit will use default factory settings. For detailed description of software update procedure	
Command Syntax	\$AL+SOFTUPDATE+[TAG]=[Password],[ftpAdr],[ftpUser],[ftpPass]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	ftpAdr	FTP server address (Max. 29 characters)
	ftpUser	FTP login (Max. 14 characters)
	ftpPass	FTP password (Max. 14 characters)

Example Command	\$AL+SOFTUPDATE=0000,ftp.com.pl,login,password
Asynchronous report	<p>Unit after performing update will send asynchronous report with the following syntax:</p> <p>UPDATESTATUS: w: [ModemUpdate], z: [IOProcUpdate]</p> <p>Modem Update – 1 – successful modem firmware update, 0 – modem firmware update error</p> <p>IOProcUpdate – 1 – successful IO processor update, 0 – IO processor update error</p> <p>After sending Ok answer unit can send another asynchronous error answer with error code (e.g. ftp download timeout, file corrupted, see ErrorCode details)</p>
Ok Answer	\$OK:SOFTUPDATE+[TAG]
Error Answer	\$ER:SOFTUPDATE+[TAG]=[ErrorCode]

SPEEDALARM

Command	SPEEDALARM	
Description	Device is activating output JP2-2 when speed limits is reached. Choose speed source, value and minimum time.	
Command Syntax	\$AL+ SPEEDALARM +[TAG]=[Password],[source],[value],[minimum_time]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit password
	source	0 – Disabled 1 – Enabled (GPS speed) 2 - Enabled (CAN speed)
	value	Limit in km/h
	Minimum_time	Value set to filter minimum over speed events
Example Query	<p>Command: \$AL+SPEEDALARM =0000,1,100,5</p> <p>Query: \$AL+ SPEEDALARM =0000,?</p> <p>Example answer: \$QR:SPEEDALARM=1,100,5</p>	

STATUS

Command	STATUS	
Description	This command provides possibility to query unit status.	
Command Syntax	\$AL+STATUS+[TAG]=[Password],?	
Query Answer Syntax	\$QR:STATUS+[TAG]=[ModemSoftware],[GPS],[TCP],[GPRSr],[GPRSm],[Gd],[Ge],[Zs],[Lr],[Gs],[Cs],[Tr],[Ss],[Reg],[Gsup]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Modem Software	Modem software compilation data
	GPS	Field describing GPS status. This field is composed of two following fields: GPS FIX indicator: 1 – FIX 0 – no FIX GPS data indicator: 1 – data ok 0 – no data from GPS unit
	TCP	Connection to server status: 1 – connected 0 – disconnected
	GPRSr	Number of sequential GPRS session attempts
	GPRSm	Current GPRS connectivity mode: 0 –device is working with mode of continuously connection to GPRS using internal algorithm.
	Gd	Number of disconnection events caused by server.
	Ge	Number of disconnection events caused by GPRS errors.
	Zs	IO processor operation status: 1 – ok 0 – IO processor failure, software update required

	Lr	Last unit reset source: 1 – power on or internal unit safety mechanism 2 – internal software mechanism 3 – successful modem software update 4 – unsuccessful modem software update
	Gs	GSM signal strength with range 0-31 (31 = best signal)
	Cs	CAN module status: 0 – disconnected 1 – connected (CAN Module) 2 – connected (CAN SEO)
	Tr	Unit total reset count
	Ss	Unit security status
	Reg	GSM registration status 0 – searching for network 1 – registered to home network 5 – registered to roaming network
	Gsup	Current network GPRS support 0 – GPRS not supported 1 – GPRS supported 2 – GPRS and EDGE supported
Example Query	Query: \$AL+STATUS=0000,? Example answer: \$QR:STATUS=SV:Feb 25 2010 11:43:17, GPS: 0,1,TCP: 1,GPRSr: 0,GPRSm: 0,Gd: 0,Ge: 0,Zs: 0,Lr: 1,Gs: 27,Cs: 0,Tr: 1186,Ss: 1,Reg: 1,Gsup:2	
Error Answer	\$ER:STATUS+[TAG]=[ErrorCode]	

SUPANALOG

Command	SUPANALOG	
Description	This command replaces VoltageAnalog1 with VoltageAnalog5.	
Command Syntax	\$AL+SUPANALOG+[TAG]=[Password],[enable]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)

	Password	Unit current password. (Max. 4 characters)
	enable	0 – disable 1 – enable
Example Command	\$AL+SUPANALOG=0000,1	
Ok Answer	\$OK: SUPANALOG+[TAG]	
Error Answer	\$ER: SUPANALOG+[TAG]=[ErrorCode]	

SYNCFRAMEMODE

Command	SYNCFRAMEMODE				
Description	This command provides possibility to change synchronization frame format.				
		Synchronization Header (16-bit)	Synchronization ID (16-bit)	Unit ID (64-bit)	End Markers (16-bit)
	Received Frame Example	0xFA 0xF9	0x29 0x01	0x93 0x39 0x01 0x79 0x78 0x12	0x13 0x10
Command Syntax	\$AL+SYNCFRAMEMODE+[TAG]=[Password],[enable]				
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)			
	Password	Unit current password. (Max. 4 characters)			
	enable	0 – disable 1 – enable			
Example Command	\$AL+SYNCFRAMEMODE=0000,1				
Ok Answer	\$OK: SYNCFRAMEMODE+[TAG]				
Error Answer	\$ER: SYNCFRAMEMODE+[TAG]=[ErrorCode]				

TACHOCARDBLOCK

Command	TACHOCARDBLOCK
Description	This command provides possibility to disable inserting tachograph card id in data frame.

Command Syntax	\$AL+TACHOCARDBLOCK+[TAG]=[Password],[Enabled]	
Query Answer Syntax	\$QR: TACHOCARDBLOCK+[TAG]=[Enabled]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Enabled	0 – function disabled- Card Id is inserted in data frame 1 – function enabled
Example Query	Query: \$AL+ TACHOCARDBLOCK =0000,? \$QR: TACHOCARDBLOCK =1	
Error Answer	\$ER: TACHOCARDBLOCK+[TAG]=[ErrorCode]	

TEMPID

Command	TEMPID	
Description	This command provides possibility to set and query DS18B20 thermometer codes for multiple thermometer support on 1-wire line	
Command Syntax	\$AL+TEMPID+[TAG]=[Password],[index],[ID] NOTE: Device will respond sending query with selected index. If new ID setting is unsuccessful ID field will be empty or contain old value	
Query command	\$AL+TEMPID+[TAG]=[Password],[index],?	
Query and Command Answer Syntax	\$QR:TEMPID+[TAG]=[index],[ID]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	index	Number of slot to read/write thermometer ID code. Value range: 1 – 6 For query also index 0 is possible. This reads currently connected thermometer code if there is only one connected.

	ID	DS18B20 ID code presented in ASCII hex format. Code is always 16 characters (8 bytes)
Example Command	Command: \$AL+TEMPID=0000,1,FB000003293AB428 Response: \$QR:TEMPID=1,FB000003293AB428	
Example Query	Query: \$AL+TEMPID=0000,1,? \$QR:TEMPID=1,FB000003293AB428	

TIMER

Command	TIMER	
Description	This command provides possibility to enable periodic telemetry reports or heartbeat frames	
Command Syntax	\$AL+TIMER+[TAG]=[Password],[Mode],[Time],[Period]	
Query Answer Syntax	\$QR:TIMER+[TAG]=[Mode],[Time],[Period]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	0 – function disabled 1 – periodic timer enabled 2 – periodic timer enabled (uses parameter Period) 3 – mode 2 with blocking updating gps data in TIMER frames
	Time	Time of report generation in format HHMMSS. Used in mode 1
	Period	Period (seconds) for heartbeat frames. Next frame will be generated after specified period from last frame. Used in mode 2
Example Query To set heartbeat every 1h	Query: \$AL+TIMER=0000,? \$QR:TIMER=2,000000,3600	
Error Answer	\$ER:TIMER+[TAG]=[ErrorCode]	

TOW

Command	TOW	
Description	This command provides possibility to enable and query towing report settings. In case of detecting towing unit will generate telemetry report 9.	
Command Syntax	\$AL+TOW+[TAG]=[Password],[Enable],[SatUsed],[MinSpeed],[MaxSpeed],[Duration]	
Query command	\$AL+TOW+[TAG]=[Password],?	
Query Answer Syntax	\$QR:TOW+[TAG]=[Enable],[SatUsed],[MinSpeed],[MaxSpeed],[Duration]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Enable	0 – disables report 1 – enables report
	SatUsed	Minimum number of satellites used for measurement
	MinSpeed	Minimum speed [km/h]
	MaxSpeed	Maximum speed [km/h]
	Duration	Duration of vehicle movement [s]
Example Query	Query: \$AL+TOW=0000, \$QR:TOW=1,5,0,255	
Error Answer	\$ER:TOW+[TAG]=[ErrorCode]	

TRACKING

Command	TRACKING	
Description	This command provides possibility to set and query unit tracking parameters.	
Command Syntax	\$AL+TRACKING+[TAG]=[Password],[Mode],[Time],[Distance],[Times],[Basis],[CommType]	

Query Answer Syntax	\$QR:TRACKING+[TAG]=[Mode],[Time],[Distance],[Times],[Basis],[CommType]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	<p>Tracking mode:</p> <p>1 – Time Mode Telemetry report is sent if the time elapsed from last report is equal or greater then defined time.</p> <p>2 – Distance Mode Telemetry report is sent if the distance change from last position is equal or greater then defined.</p> <p>3 – Intelligent Mode Telemetry report is sent if the distance and the time or movement angle change equal or are greater than defined in UNPM and TRACKING commands.</p> <p>5 – Time Mode + ACC ON The same as mode 1 but additional ignition input active state required.</p> <p>6 – Distance Mode + ACC ON The same as mode 2 but additional ignition input active state required.</p> <p>7 – Intelligent Mode + ACC ON The same as mode 3 but additional ignition input active state required.</p> <p>8 – Sensitive Mode + ACC ON Telemetry report is sent if the distance or the time or movement angle change equal or are greater than defined in UNPM and TRACKING commands.</p>
	Time	Tracking time value [s]. (Min. 0, Max. 65535)
	Distance	Tracking distance value [m]. (Min. 0, Max. 65535)
	Times	Number of packages to be sent. 0 values mean continuous tracking. (Min. 0, Max. 65535)
	Basis	<p>0 – Unit will generate only if GPS acquired proper position data</p> <p>1 – Unit will generate packets regardless to presence of GPS signal</p>

	CommType	0 – Serial Port 1 – Reserved 2 – SMS 3 – GPRS
Example Query	Query: \$AL+TRACKING=[Password],? Example answer: \$QR:TRACKING=7,15,500,0,1,3	
Example Command	\$AL+TRACKING=0000,7,15,500,0,1,3	
Ok Answer	\$OK:TRACKING+[TAG]	
Error Answer	\$ER:TRACKING+[TAG]=[ErrorCode]	

TRACKROAM

Command	TRACKROAM	
Description	This command provides possibility to set and query unit tracking parameters used in roaming when gprsRoamingMode is set to value 4 or 5.	
Command Syntax	\$AL+TRACKROAM+[TAG]=[Password],[Mode],[Time],[Distance],[Times],[Basis], [CommType]	
Query Answer Syntax	\$QR:TRACKROAM+[TAG]=[Mode],[Time],[Distance],[Times],[Basis], [CommType]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	Tracking mode: 1 – Time Mode Telemetry report is sent if the time elapsed from last report is equal or greater then defined time. 2 – Distance Mode Telemetry report is sent if the distance change from last position is equal or greater then defined. 3 – Intelligent Mode Telemetry report is sent if the distance and the time or movement angle change equal or are greater than defined in UNPM and TRACKIN commands. 5 – Time Mode + ACC ON The same as mode 1 but additional ignition input active state required.

		<p>6 – Distance Mode + ACC ON The same as mode 2 but additional ignition input active state required.</p> <p>7 – Intelligent Mode + ACC ON The same as mode 3 but additional ignition input active state required.</p> <p>8 – Sensitive Mode + ACC ON Telemetry report is sent if the distance or the time or movement angle change equal or are greater than defined in UNPM and TRAROAM commands.</p>
	Time	Tracking time value [s]. (Min. 0, Max. 65535)
	Distance	Tracking distance value [m]. (Min. 0, Max. 65535)
	Times	Number of packages to be sent. 0 values mean continuous tracking. (Min. 0, Max. 65535)
	Basis	<p>0 – Unit will generate only if GPS acquired proper position data</p> <p>1 – Unit will generate packets regardless to presence of GPS signal</p>
	CommType	<p>0 – Serial Port</p> <p>1 – Reserved</p> <p>2 – SMS</p> <p>3 – GPRS</p>
Example Query	<p>Query: \$AL+TRACKROAM=[Password],?</p> <p>Example answer: \$QR:TRACKROAM=6,500,5000,0,1,3</p>	
Example Command	\$AL+TRACKROAM=0000,7,15,500,0,1,3	
Ok Answer	\$OK:TRACKROAM+[TAG]	
Error Answer	\$ER:TRACKROAM+[TAG]=[ErrorCode]	

UNPM

Command	UNPM
Description	This command provides possibility to set and query primary unit configuration parameters. Unit will reboot in 10s. After changing configuration with this command.

Command Syntax	\$AL+UNPM+[TAG]=[Password],[UnitID],[NewPassword],[simPin],[R],[R],[R],[R],[gprsRoamingMode],[trackingAngle]	
Query Answer Syntax	\$QR:UNPM+[TAG]=[UnitID],[Password],[simPin],[R],[R],[R],[R],[gprsRoamingMode],[trackingAngle]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	UnitID	Unit current number. (Max. 19 characters)
	NewPassword	Unit new password. If command is not changing password, field "NewPassword" should be the same as "Password". (Max. 4 characters)
	simPin	SIM card pin.(Max. 6 characters)
	R	Reserved for further use.
	gprsRoamingMode	Field defining unit behavior in network roaming mode. Below you can find possible values with description: 0 – Not available in GPRS S8 1 – Not available in GPRS S8 2 – Not available in GPRS S8 3 – Not available in GPRS S8 4 – Unit disconnect in roaming and store all date to telemetry archive. Unit will send all telemetry data in chronological order requiring confirmation of delivery of every frame 5 – Similar to 4 mode, but additionally unit will operate in Roaming using TRACKROAM settings
	trackingAngle	Decimal angle value. Unit heading change by this value will generate telemetry packet if unit is configured in proper tracking mode. See TRACKING command. (Min. value 5, Max. value 359)
Example Query	Query: \$AL+UNPM=0000,? Example answer: \$QR:UNPM=210000123,0000,9998,7,7,7,7,4,20	
Example Command	\$AL+UNPM=0000,210000123,abcd,9998,7,7,7,7,4,50	
Ok Answer	\$OK:UNPM+[TAG]	

Error Answer	\$ER:UNPM+[TAG]=[ErrorCode]
--------------	-----------------------------

VERSION

Command	VERSION	
Description	This command provides possibility to query unit software version (communication subsystem)	
Command Syntax	\$AL+VERSION+[TAG]=[Password],?	
Query Answer Syntax	\$QR:VERSION+[TAG]=[Software Version]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Software Version	Software version string (Max. 35 characters)
Example Query	Query: \$AL+VERSION=0000,? Example answer: \$QR:VERSION=Feb 23 2010 08:34:36	
Error Answer	\$ER:VERSION+[TAG]=[ErrorCode]	

VIRTUALACC

Command	VIRTUALACC	
Description	This command turning on ignition, when vehicle is on move.	
Command Syntax	\$AL+VIRTUALACC+[TAG]=[Password],[Mode],[Time],[SpeedGPS],[Samples]	
Query Answer Syntax	\$QR: VIRTUALACC+[TAG]=[Mode],[Time],[SpeedGPS],[Samples]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	Mode	0 – Disabled 1 – Enabled 2- Enabled (For FM15)

	Time	Duration of stop to turn off ignition (seconds)
	SpeedGPS	Minimal speed to turn on ignition (km/h)
	Samples	Minimal quantity of frames with speed from GPS above defined in SpeedGPS to turn on ignition

VOLTDETECT

Command	VOLTDETECT	
Description	This command enables voltage dropout detection. Voltage drop event generates telemetry frame with reportID = 47.	
Command Syntax	\$AL+VOLTDETECT+[TAG]=[Password],[enable],[threshold],[time],[delay],[send_option]	
Query command	\$AL+VOLTDETECT+[TAG]=[Password],?	
Query Answer Syntax	\$QR:VOLTDETECT+[TAG]=[enable],[threshold],[time],[delay],[send_option]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	enable	0 – Function disabled 1 – Function enabled then ignition ON 2 – Function enabled regardless of ignition state
	hreshold	Threshold value below which report is generated (unit: 10mV) example: 800 = 8V 1256 = 12,56V
	time	Minimum duration (seconds) of low voltage to generate report (seconds)
	delay	Delay (seconds) to generate next report in case of continuous low voltage Note: value 0 = generate report only once per threshold crossing
	sending	Not supported in S8 (see SMSREPORT command)

Example Query	Query: \$AL+VOLTDETECT=0000,? Example answer: \$QR:VOLTDETECT=1,800,10,0,0
Error Answer	\$ER:VOLTDETECT+[TAG]=[ErrorCode]

ZCONFIG

Command	ZCONFIG	
Description	This command provides possibility to set various configuration options in I/O processor	
Command Syntax	\$AL+ZCONFIG+[TAG]=[Password],[State],[ConfigOption]	
Query command	\$AL+ZCONFIG+[TAG]=[Password],?	
Query Answer Syntax	\$QR:ZCONFIG+[TAG]=[OptionStates]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	State	0 – disable option 1 – enable option default: all options disabled
	ConfigOption	Number of configuration function: 2 – replace CAN [engine_temp] by Dallas digital thermometer temperature 3 – replace CAN [engine_speed] by value calculated from counting input. 4 – enable CAN units unification (default CAN value units are describes is Telemetry Packets section) 5 – enable Dallas Immobilizer features 7 – enable CAN [total_distance] conversion to miles 8 – enable verification of Dallas ID defined on DALLASID list 9 – enable GARMIN FMI protocol support (exhibit some other functionality) 10 – enable 3D sensor (if installed on board) 12 – enable CAN [Total_Fuel] [Total_distance] [Fuel_level] values to be buffered by the device 14 – enable CAN asynchronous events blocking

Command Syntax	\$AL+ZCONFIG2+[TAG]=[Password],[State],[ConfigOption]	
Query command	\$AL+ZCONFIG2+[TAG]=[Password],?	
Query Answer Syntax	\$QR:ZCONFIG2+[TAG]=[OptionStates]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	State	0 – disable option 1 – enable option default: all options disabled
	ConfigOption	<p>Number of configuration function:</p> <ul style="list-style-type: none"> 1 – Support internal CAN module with internal TMR module 2 – Insert total vehicle weight into [axleweight4] field 5 – Insert impulses from JP2-7 into [range] field 6 – Insert temperature into [wiretemp6] field instead of Jamming/CREG/CME/CMS value 7 – Enable EXPANDER (more information is available in the section about EXPANDER) 8 – Dedicated customer feature 9 – Dedicated customer feature 10 – Dedicated customer feature 11 – Dedicated customer feature 12 – Dedicated customer feature 13 – Dedicated customer feature 14 – Dedicated customer feature 15 – Dedicated customer feature 16 – Bluetooth 4.0 Low Energy module 17 – Refresh CAN data always – on stops also 18 – TTL logic for inputs (0 when < 2V, 1 when > 2V) 19 – Enable CAN bus data in case of FM15 CAN 20 – No refreshing GPS position when drivercode field is empty 21 – Use output 2 in case of use Dallas Immobilizer feature 22 – Use output 3 in case of use Dallas Immobilizer feature 23 – The feature that allows sending and receiving commands via Rx, Tx using UARTDATA command 25 – Dedicated customer feature 26 – Sending data frames via RS232TTL also with default baudrate 9600 (other baudrate can be set using ZVALUE30)

Command	ZMASK	
Description	This command provides possibility to set various configuration masks in I/O processor	
Command Syntax	\$AL+ZMASK+[TAG]=[Password],[MaskNumber],[Mask]	
Query command	\$AL+ZMASK+[TAG]=[Password],?	
Query Answer Syntax	\$QR:ZMASK+[TAG]=[Mask]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	MaskNumber	Always 1
	Mask	Mask value e.g. 0011000000000000
Example Query	Query: \$AL+ZMASK=0000,? Example answer: \$QR:ZMASK=0011000000000000	
Example Command	\$AL+ZMASK=0000,1,0011000000000000 [please notice that you have to insert all 16 bits! In other case device will return with \$ER:ZMASK=2)	
Ok Answer	Command is forwarded do I/O processor and generates no OK answer.	
Error Answer	\$ER:ZMASK+[TAG]=[ErrorCode]	

ZVALUE

Command	ZVALUE	
Description	This command provides possibility to set various configuration values in I/O processor	
Command Syntax	\$AL+ZVALUE+[TAG]=[Password],[ValueNumber],[Value]	
Query command	\$AL+ZVALUE+[TAG]=[Password],?	
Query Answer Syntax	\$QR:ZVALUE+[TAG]=[Value1],[Value2],[Value3],[Value4],[Value5],[Value6],[Value7],[Value8],[Value9],[Value10],[Value11],[Value12],[Value13],[Value14],[Value15],[Value20],[Value21],[Value22],[Value23],[Value24],[Value25],[Value	

	26],[Value27],[R]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	ValueNumber	<p>Number of configuration value:</p> <p>1 – Correction multiplier for Total Fuel CAN parameter</p> <p>3 – Fuel tank capacity for CAN unit unification</p> <p>6 – Analog fuel tank capacity</p> <p>8 – Engine RPM threshold for SETECO function</p> <p>9 – CAN [Total_distance] correction multiplier</p> <p>10 – Max accelerator pressure threshold for [Rapid_pedal_pressure] counter</p> <p>11 – Max car speed threshold for [Drive_time_over_speedlimit] counter</p> <p>12 – Max engine RPM threshold for [Drive_time_over_rpmlimit] counter</p> <p>13 – Analog probe: Sample per filtration calculation.</p> <p>14 – Analog probe: Sample period</p> <p>15 – Divider for counting input</p> <p>16 – Digital probe address no. 1.</p> <p>17 – Digital probe address no. 2.</p> <p>18 – Low Pass filter Threshold [ms]</p> <p>20 – Correction value for CAN [Vehicle_speed]</p> <p>21 – Accelerator pressure limit</p> <p>22 – Drive speed</p> <p>23 – Dedicated customer feature</p> <p>24 – Dedicated customer feature</p> <p>25 – Delay to stop impulse counting (if you want to use this feature please set 1 in ZCONFIG 18)</p> <p>26 – Delay to authorization expiring</p> <p>27 – Delay to remove MAC address of Bluetooth TAG from a data frame</p> <p>28 – Dedicated customer feature</p> <p>29 – Dedicated customer feature</p> <p>30 – You can set baud rate here 0-115200. Default is 0 treated as 9600. You can also set unusual values.</p> <p>31 – Delay for SENT-GEO feature</p>
	Value	<p>Value to set</p> <p>1)</p> <p>range: 1 – 2000 (0.1% – 200%)</p> <p>default: 0 – correction disabled</p> <p>3)</p>

		<p>range: 1 – 9999 (1 – 9999 liters) default: 0 – disabled</p> <p>6)</p> <p>range: 1 – 9999 (1 – 9999 liters) default: 0 – calculation disabled</p> <p>8)</p> <p>range: 1 – 9999 (1 – 9999 RPM) default: 0 – disabled</p> <p>9)</p> <p>range: 1 – 2000 (0.1% – 200%) default: 0 – correction disabled</p> <p>10)</p> <p>range: 0 – 100 (0 – 100 %) default: 0 – counting total time above 0%</p> <p>11)</p> <p>range: 0 – 250 (0 – 250 km/h) default: 0 – counting total drive time over speed limit</p> <p>12)</p> <p>range: 0 – 9999 (0 – 9999 RPM) default: 0 – counting total time above speed limit</p> <p>13)</p> <p>range: 0 – 50 (0-50 quantity of samples – filter for analog measure – 10 as default)</p> <p>14)</p> <p>range: 0 – 50 (0 – 50 time of samples)</p> <p>15)</p> <p>range: 1 – 100 (impulses will be divided by typed value)</p> <p>20)</p> <p>range: 1 – 200 (1 – 200%) default: 0 – correction disabled</p> <p>21)</p> <p>range: 0 – 100 (0 – 100%) default: 0 – function disabled</p> <p>22)</p> <p>range: 0 – 250 (0 – 250 km/h) default: 0 – function disabled</p> <p>23)</p> <p>range: 0 – 9999 (0 – 9999 s) default: 0 – function disabled</p> <p>24)</p> <p>range: 0 – 9999 (0 – 9999 RPM) default: 0 – function disabled</p> <p>25)</p> <p>range: 0 – 9999 (0 – 9999 s) default: 0 – function disabled</p> <p>26)</p>
--	--	---

		<p>range: 0 – 9999 (0 – 9999 s) default: 0 – function disabled 27)</p> <p>range: 0 – 9999 (0 – 9999 s) default: 0 – 60 seconds 28)</p> <p>range: 0 – 250 (0 – 250 s) default: 0 – function disabled 29)</p> <p>range: 0 – 9999 (quantity of samples) default: 0 – function disabled 30)</p> <p>range: 0 – 115200 (quantity of samples) default: 0 – 9600 baudrate 31)</p> <p>range: 0 – 3600 (0 – 3600 s) default: 0 – function disabled</p>
	Value1, Value2, Value3, Value4, Value5, Value6, Value7, Value8, Value9, Value10, Value11, Value12, Value13, Value14, Value15, Value20, Value21, Value22, Value23, Value24, Value25, Value26, Value27, Value28, Value29, Value30, Value31	Configuration values currently set in device

Remove of device – to use in other vehicle

In case of remove of device there is a necessary to reset CAN data that was read from previous car, to do that send:

RESETCANBUFFER

Command	RESETCANBUFFER	
Description	This command reset data buffer.	
Command Syntax	\$AL+RESETCANBUFFER+[TAG]=[Password]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
Example Command	\$AL+RESETCANBUFFER=0000	
Ok Answer	\$OK:RESETCANBUFFER+[TAG]	
Error Answer	\$ER:RESETCANBUFFER+[TAG]=[ErrorCode]	

CANCMD

Command	CANCMD	
Description	This command provides possibility to send raw binary command directly to CAN module. For detailed CAN module commands and responses description see Can module protocol documentation.	
Command Syntax	\$AL+CANCMD+[TAG]=[Password],[CMD_STRING]	
Answer Syntax	\$QR:CANCMD+[TAG]=[CAN_RESPONSE]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	CMD_STRING	B34104FF08 – reset CAN-Logistic data

	CAN_RESPONSE	B442041F00E6 – correct response for finished CAN-Logistic data reset
Example Command	\$AL+CANCMD=0000,B34104FF08	
Ok Answer	\$QR:CANCMD=B442041F <u>00</u> E6 – marked as underline should be 00	
Error Answer	\$ER:CANCMD+[TAG]=[ErrorCode]	

Telemetry Packets

Device supports a few different formats of telemetry packets containing various sets of telemetry data. Telemetry format can be changed by executing FRAMEFORMAT command. All telemetry packets send on server request have optional TAG field. It can be helpful to pair request with corresponding data answer. Frames with TAG have the following prefix at the beginning:

\$RP:[TAG]

Frames without TAG field have no prefix. All fields in frames marked as [] are optional. They may come empty (no characters between commas) meaning requested data is unavailable at the moment or invalid. All telemetry packets are finished by \r\n (binary: 0x0D 0x0A).

Data Frames confirmation

For proper work S8 devices require confirmation to every single data frame that will send to TCP server. See data frames description. For every received data frame server must generate value **0xFB 0xF9** - only then device will continue to send next data frame

Attention! GPRS S8 devices supporting only UNPM 4th and 5th GPRS Roaming mode!

See that last two fields of data frames are: [seq_num] and [CRC]

[seq_num] is 16 bit unsigned value representing packed sequence number.

[CRC] is 16 bit CRC16-CCIT (0x1D0F) checksum presented in ASCII hex format (always 4 characters). This CRC is calculated from all fields including comma before [CRC] field

Example of data frame with CRC marked RED:

123456789012345,20130701115426,F7,20.939029,52.191005,0,6,24,176,26002,11,65,0,14.42,14.49,00.17,00.14,0,,1234717186,1234717186,,,,,,,,,,,,,0,,,,,,,,,0,,,0,,,,,,,,,27472,8041

Frame Format 0

[DeviceID],[DateTime],[Longitude],[Latitude],[SpeedGPS],[Heading],[Altitude],[Satellite],[EventID],[inputs],[outputs],[seq_num],[CRC]

Frame Format 1

[DeviceID],[DateTime],F1,[Longitude],[Latitude],[Speed-GPS],[Heading],[Altitude],[Satellite],[EventID],[Mileage-GPS],[InputStatus],[VoltageAnalog1],[VoltageAnalog2],[Outputs],[total_distance],[total_fuel],[vehicle_speed],[engine_speed],[fuel_level],[fuel_consump],[accelerator],[tachograph],[axleweight],[indicators],[seq_num],[CRC]

Frame Format 2

[DeviceID],[DateTime],F2,[Longitude],[Latitude],[EventID],[Speed-GPS],[Heading],[Satellite],[Inputs],[Outputs],[total_distance],[total_fuel],[engine_speed],[fuel_level],

[indicators_C],[engine_hours],[engine_temp],[tachograph],[accelerator],[axleweight2],[fuel_consump],[seq_num],[CRC]

Frame Format 4

[DeviceID],[DateTime],F4,[Longitude],[Latitude],[SpeedGPS],[Heading],[Altitude],[Satellite],[EventID],[MileageGPS],[InputStatus],[VoltageAnalog1],[VoltageAnalog2],[VoltageAnalog3],[VoltageAnalog4],[VoltageAnalog5],[outputs],[total_distance],[total_fuel],[vehicle_speed],[engine_speed],[fuel_level],[fuelconsump],[accelerator],[tachograph],[axleweight2],[indicators],[DriverCode],[1WireTemp_1],[1WireTemp_2],[1WireTemp_3],[1WireTemp_4],[1WireTemp_5],[1WireTemp_6],[seq_num],[CRC]

Frame Format 7

[DeviceID],[DateTime],F7,[Longitude],[Latitude],[Heading],[Satellite],[GSM_signal],[Altitude],[GSM_operator],[EventID],[Inputs],[Outputs],[VoltageAnalog1],[VoltageAnalog2],[VoltageAnalog3],[VoltageAnalog4],[Speed-GPS],[Vehicle_speed],[Mileage-GPS],[Total_distance],[Total_fuel],[Total_idle_fuel],[Fuel_level_percents],[Fuel_level_litres],[Fuel_Flag],[Fuel_consump],[Accelerator],[Engine_speed],[Engine_temperature],[Engine_hours],[Engine_idle_time],[Oil_temperature],[Hydraulic_oil_temperature],[Tachograph],[Axleweight1],[Axleweight2],[Axleweight3],[Axleweight4],[Indicators_C],[1WireTemp_1],[WireCode_1],[1WireTemp_2],[WireCode_2],[1WireTemp_3],[WireCode_3],[1WireTemp_4],[WireCode_4],[1WireTemp_5],[WireCode_5],[1WireTemp_6],[WireCode_6],[Rapid_pedal_pressure],[Rapid_accelerations],[Rapid_brakings],[Engine_speed_overspeed_counter],[RANGE],[Drive_time_over_rpmlimit],[Drive_time_over_speedlimit],[Driver1_IDcard_tacho],[Driver2_IDcard_tacho],[DriverCode],[3DX],[3DY],[3DZ],[seq_num],[CRC]

Frame Format F12

[DeviceID],[DateTime],F12,[Longitude],[Latitude],[Heading],[Satellite],[GSM_signal],[Altitude],[GSM_operator],[EventID],[Inputs],[Outputs],[VoltageAnalog1],[VoltageAnalog2],[VoltageAnalog3],[VoltageAnalog4],[SpeedGPS],[Vehicle_speed],[MileageGPS],[Total_distance],[Total_fuel],[Total_idle_fuel],[Fuel_level_percents],[Fuel_level_litres],[Fuel_Flag],[Fuel_consump],[Accelerator],[Engine_speed],[Engine_temperature],[Engine_hours],[Engine_idle_time],[Oil_temperature],[Hydraulic_oil_temperature],[Tachograph],[Axleweight1],[Axleweight2],[Axleweight3],[Axleweight4],[Indicators_C],[1WireTemp_1],[WireCode_1],[1WireTemp_2],[WireCode_2],[1WireTemp_3],[WireCode_3],[1WireTemp_4],[WireCode_4],[1WireTemp_5],[WireCode_5],[1WireTemp_6],[WireCode_6],[Rapid_pedal_pressure],[Rapid_accelerations],[Rapid_brakings],[Engine_speed_overspeed_counter],[RANGE],[Drive_time_over_rpmlimit],[Drive_time_over_speedlimit],[Driver1_IDcard_tacho],[Driver2_IDcard_tacho],[DriverCode],[3DX],[3DY],[3DZ],[Total_number_of_brake applies],[Pedal_braking_factor],[Engine_braking_factor],[Total_number_of_accelerator_kick_downs],[Total_driving_time_with_accelerator_kick_down],[Total_driving_time_when_cruise_control_is_on],[ecospeed_1],[ecospeed_2],[ecospeed_3],[ecospeed_4],[ecospeed_5],[ecospeed_6],[ecoRPM_1],[ecoRPM_2],[ecoRPM_3],[ecoRPM_4],[ecoRPM_5],[ecoRPM_6],[eco_acc_1],[eco_acc_2],[eco_acc_3],[eco_acc_4],[eco_acc_5],[eco_acc_6],[eco_braking_1],[eco_braking_2],[eco_braking_3],[eco_braking_4],[eco_braking_5],[eco_braking_6],[eco_pedalpress_1],[eco_pedalpress_2],[eco_pedalpress_3],[eco_pedalpress_4],[eco_pedalpress_5],[eco_pedalpress_6],[adblue],[engine_oil_level],[alternative_gasoline],[axle_weight5],[engine_torque],[distance_to_service],[axle_brutto],[seq_num],[CRC]

Frame Format F16

[DeviceID],[DateTime],F16,[Longitude],[Latitude],[Heading],[Satellite],[GSM_signal],[Altitude],[GSM_operator],[EventID],[Inputs],[Outputs],[VoltageAnalog1],[VoltageAnalog2],[VoltageAnalog3],[VoltageAnalog4],[SpeedGPS],[Vehicle_speed],[MileageGPS],[Total_distance],[Total_fuel],[Total_idle_fuel],[Fuel_level_percents],[Fuel_level_litres],[Fuel_Flag],[Fuel_consump],[Accelerator],[Engine_speed],[Engine_temperature],[Engine_hours],[Engine_idle_time],[Oil_temperature],[Hydraulic_oil_temperature],[Tachograph],[Axleweight1],[Axleweight2],[Axleweight3],[Axleweight4],[Indicators_C],[1WireTemp_1],[WireCode_1],[1WireTemp_2],[WireCode_2],[1WireTemp_3],[WireCode_3],[1WireTemp_4],[WireCode_4],[1WireTemp_5],[WireCode_5],[1WireTemp_6],[WireCode_6],[Rapid_pedal_pressure],[Rapid_accelerations],[Rapid_brakings],[Engine_speed_overspeed_counter],[RANGE],[Drive_time_over_rpm_limit],[Drive_time_over_speedlimit],[Driver1_IDcard_tacho],[Driver2_IDcard_tacho],[DriverCode],[3DX],[3DY],[3DZ],[Total_number_of_brake_applies],[Pedal_braking_factor],[Engine_braking_factor],[Total_number_of_accelerator_kick_downs],[Total_driving_time_with_accelerator_kick_down],[Total_driving_time_when_cruise_control_is_on],[ecospeed_1],[ecospeed_2],[ecospeed_3],[ecospeed_4],[ecospeed_5],[ecospeed_6],[ecoRPM_1],[ecoRPM_2],[ecoRPM_3],[ecoRPM_4],[ecoRPM_5],[ecoRPM_6],[eco_acc_1],[eco_acc_2],[eco_acc_3],[eco_acc_4],[eco_acc_5],[eco_acc_6],[eco_braking_1],[eco_braking_2],[eco_braking_3],[eco_braking_4],[eco_braking_5],[eco_braking_6],[eco_pedalpress_1],[eco_pedalpress_2],[eco_pedalpress_3],[eco_pedalpress_4],[eco_pedalpress_5],[eco_pedalpress_6],[adblue],[engine_oil_level],[alternative_gasoline],[axle_weight5],[engine_torque],[distance_to_service],[axle_brutto],[coasting],[pto_duration],[pto_fuel_consumption],[clutch_presses],[retarder_usage],[R1],[R2],[R3],[R4],[R5],[R6],[R7],[seq_num],[CRC]

Frame Format 77 – SMS only format (send as SMS instead of F7 data frame)

This frame contains following fields:

[DeviceID],[DateTime],F77,[Longitude],[Latitude],[Heading],[Satellite],[Altitude],[EventID],[Inputs],[Outputs],[VoltageAnalog2],[Speed-GPS],[3DX],[3DY],[3DZ],[VoltageAnalog1]

Field	Description	Type (total/differential /not applicable)	Data format
DeviceID	Unique ID of the device, it can contain up to 19 characters	not applicable	Up to 19 characters
DateTime	UTC date and time, up to 15 characters containing: year, month, day, hour, minutes, seconds	not applicable	Up to 15 characters
Longitude	Longitude of the position containing: 1	not applicable	32-bit float (ASCII text)

	optional minus (-) character, 1 to 3 characters of main value, dot, 6 characters of fraction		
Latitude	Latitude of the position containing: 1 optional minus (-) character, 1 to 3 characters of main value, dot, 6 characters of fraction	not applicable	32-bit float (ASCII text)
Speed-GPS	Speed of the vehicle read from GPS device	not applicable	16-bit unsigned (ASCII text)
Heading	Angle of the vehicle movement	not applicable	16-bit unsigned (ASCII text)
Altitude	Altitude of the vehicle from GPS receiver	not applicable	16-bit unsigned (ASCII text)
Satellite	Number of satellites used	not applicable	8-bit unsigned (ASCII text)
EventID	ID of the event which caused current frame to be send	not applicable	8-bit unsigned (ASCII text)
Inputs	State of the device inputs – see Inputs Field description chapter for details	not applicable	16-bit unsigned (ASCII text)
Outputs	State of the device outputs – see table Outputs Field description chapter for details	not applicable	8-bit unsigned (ASCII text)
Mileage-GPS	mileage counter based on GPS device	total or differential according to	32-bit unsigned (ASCII text)

		configuration by SETGPSKM	
VoltageAnalog1	Voltage of backup power supply (external accumulator), containing always: 2 digits of value, dot and 2 digits of fraction	not applicable	5 characters
VoltageAnalog2	Voltage of the main power supply, containing always: 2 digits of value, dot and 2 digits of fraction	not applicable	5 characters
Total_distance (CAN Data)	Total mileage of the vehicle from CAN bus unified CAN unit: 1m/bit	total	32-bit unsigned (ASCII text)
Total_fuel (CAN Data)	Total fuel consumed by the vehicle from the installation of CAN Module device unified CAN unit: 1ml/bit	total or differential in case of use analog impulses	32-bit unsigned (ASCII text)
Vehicle_speed (CAN Data)	Vehicle speed from CAN bus unified CAN unit: 1kmh/bit	not applicable	16-bit unsigned (ASCII text)
Engine_speed (CAN Data)	Engine speed from CAN bus unified CAN unit: 1rpm/bit	not applicable	16-bit unsigned (ASCII text)
Fuel_level (CAN Data)	Current fuel level of the vehicle unified CAN unit: 1l/bit or 1%/bit	not applicable	16-bit unsigned (ASCII text)

Fuel_consump (CAN Data)	Current fuel consumption of the vehicle	not applicable	16-bit unsigned (ASCII text)
Accelerator (CAN Data)	Current acceleration pedal position unified CAN unit: 1%/bit	not applicable	8-bit unsigned (ASCII text)
Tachograph (CAN Data)	Tachograph data – see Tachograph Field description chapter for details	not applicable	16-bit unsigned (ASCII HEX format)
Indicators (CAN Data)	Set of various indicators status – see Indicators Field description chapter for details	not applicable	16-bit unsigned (ASCII text)
Indicators „C” (CAN Data)	Set of various indicators status – see Indicators “C” Field description chapter for details This field is send as up to 8 hexadecimal characters with no leading zeros	not applicable	32-bit unsigned (ASCII HEX format)
Engine_hours (CAN Data)	Engine total work time unified CAN unit: 1min/bit	total	32-bit unsigned (ASCII text)
Engine_temperature (CAN Data)	Temperature of the engine unified CAN unit: 1C/bit	not applicable	8-bit unsigned (ASCII text)
VoltageAnalog3	Voltage of JP1-5 (analog input), containing always: 2 digits of value, dot and 2 digits of fraction	not applicable	5 characters

VoltageAnalog4	Voltage of JP2-8 (analog input), containing always: 2 digits of value, dot and 2 digits of fraction	not applicable	5 characters
VoltageAnalog5	Voltage of backup power supply (internal accumulator), containing always: 2 digits of value, dot and 2 digits of fraction	not applicable	5 characters
DriverCode	64-bit unique driver code read from 1-wire identification device, and presented as hexadecimal characters (e.g. FD347834A2B3D101)	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
1WireTemp_1	Temperature read from 1 st one-wire temperature sensor detailed temperature calculation method is presented in Dallas Temperature description section	not applicable	12-bit signed (ASCII HEX format 3 characters)
1WireTemp_2	Temperature read from 2 nd one-wire temperature sensor detailed temperature calculation method is presented in Dallas Temperature description section	not applicable	12-bit signed (ASCII HEX format 3 characters)
1WireTemp_3	Temperature read from 3 rd one-wire temperature sensor	not applicable	12-bit signed (ASCII HEX format 3 characters)

	detailed temperature calculation method is presented in Dallas Temperature description section		
1WireTemp_4	Temperature read from 4 th one-wire temperature sensor detailed temperature calculation method is presented in Dallas Temperature description section	not applicable	12-bit signed (ASCII HEX format 3 characters)
1WireTemp_5	Temperature read from 5 th one-wire temperature sensor detailed temperature calculation method is presented in Dallas Temperature description section	not applicable	12-bit signed (ASCII HEX format 3 characters)
1WireTemp_6	Value that inform about CME/CMS error, Jamming and CREG state.	not applicable	12-bit signed (ASCII HEX format 4 characters)
GSM_signal	GSM signal strength (range: 0 - 33)	not applicable	8-bit unsigned (ASCII text)
GSM_operator	Identification number of network GSM operator	not applicable	32-bit unsigned (ASCII text)
Total_idle_fuel (CAN Data)	Total fuel used when idle value read from CAN device unified CAN unit: 1ml/bit	total	32-bit unsigned (ASCII text)
Fuel_level_litres (CAN Data)	Current fuel level of the vehicle always in liters	not applicable	16-bit unsigned (ASCII text)

Fuel_level_percents (CAN Data)	Current fuel level of the vehicle always in percent's	not applicable	16-bit unsigned (ASCII text)
Fuel_Flag	Information about [fuel_level_litres] source	not applicable	1 character
Engine_idle_time (CAN Data)	Engine total idle time unified CAN unit: 1min/bit	total	32-bit unsigned (ASCII text)
Oil_temperature (CAN Data)	Current Oil temperature unified CAN unit: 1C/bit	not applicable	8-bit unsigned (ASCII text)
Hydraulic_oil_temperature (CAN Data)	Current hydraulic oil temperature unified CAN unit: 1C/bit	not applicable	8-bit unsigned (ASCII text)
Axleweight1 (CAN Data)	Current axle 1 weight unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
Axleweight2 (CAN Data)	Current axle 2 weight unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
Axleweight3 (CAN Data)	Current axle 3 weight unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
Axleweight4 (CAN Data)	Current axle 4 weight unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
WireCode_1	Identification number of Dallas thermometer 1	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
WireCode_2	Identification number of Dallas thermometer 2	not applicable	64-bit unsigned (ASCII HEX format 16 characters)

WireCode_3	Identification number of Dallas thermometer 3	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
WireCode_4	Identification number of Dallas thermometer 4	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
WireCode_5	Identification number of Dallas thermometer 5	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
WireCode_6	Identification number of Dallas thermometer 6	not applicable	64-bit unsigned (ASCII HEX format 16 characters)
Rapid_pedal_pressure (CAN Data)	Time counter of pressure above configured - in seconds for Eco driving	differential	32-bit unsigned (ASCII text)
Rapid_accelerations (CAN Data)	Counter of dynamic accelerations Value for Eco driving	total	32-bit unsigned (ASCII text)
Rapid_brakings (CAN Data)	Counter of dynamic brakes Value for Eco driving	total	32-bit unsigned (ASCII text)
Engine_speed_overspeed_counter (CAN Data)	Counter of reached engine speed limits Value for Eco driving	differential	32-bit unsigned (ASCII text)
RANGE (CAN Data)	Actual car range value	not applicable	32-bit unsigned (ASCII text)
Drive_time_over_rpm_limit (CAN Data)	Total time of drive with over engine speed limit for Eco driving unified CAN unit: 1min/bit	Total	32-bit unsigned (ASCII text)
Drive_time_over_speed_limit	Total time of drive with over speed limit	total	32-bit unsigned (ASCII text)

(CAN Data)	for Eco driving unified CAN unit: 1min/bit		
Driver1_IDcard_tacho (CAN Data)	Driver 1 card ID (Tachograph)	not applicable	max 30 characters
Driver2_IDcard_tacho (CAN Data)	Driver 2 card ID (Tachograph)	not applicable	max 30 characters
[3DX]	3D sensor X axis acceleration	not applicable	8-bit signed (ASCII text)
[3DY]	3D sensor Y axis acceleration	not applicable	8-bit signed (ASCII text)
[3DZ]	3D sensor Z axis acceleration	not applicable	8-bit signed (ASCII text)
Total_number_of_brake _applies (CAN Data)	Quantity of brake applies for Eco driving	total	32-bit unsigned (ASCII text)
Pedal_braking_factor (CAN Data)	Pedal braking factor for Eco driving	not applicable	32-bit unsigned (ASCII text)
Engine_braking_factor (CAN Data)	Engine braking factor for Eco driving	not applicable	32-bit unsigned (ASCII text)
Total_number_of_accel erator_kick_downs (CAN Data)	Total number of accelerator kick downs for Eco driving	total	32-bit unsigned (ASCII text)
Total_driving_time_with _accelerator_kick_down (CAN Data)	Total driving time with accelerator kick down for Eco driving	total	32-bit unsigned (ASCII text)
Total_driving_time_whe n_cruise_control_is_on (CAN Data)	Total driving time when cruise control is on for Eco driving	total	32-bit unsigned (ASCII text)
ecospeed_1 (CAN Data)	1 st range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)

ecospeed_2 (CAN Data)	2 nd range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecospeed_3 (CAN Data)	3 rd range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecospeed_4 (CAN Data)	4 th range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecospeed_5 (CAN Data)	5 th range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecospeed_6 (CAN Data)	6 th range for speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecoRPM_1 (CAN Data)	1 st range for engine speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecoRPM_2 (CAN Data)	2 nd range for engine speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecoRPM_3 (CAN Data)	3 rd range for engine speed parameter for Eco driving unified CAN unit:	differential	8-bit unsigned (ASCII text)

	1sec/bit		
ecoRPM_4 (CAN Data)	4 th range for engine speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecoRPM_5 (CAN Data)	5 th range for engine speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
ecoRPM_6 (CAN Data)	6 th range for engine speed parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_acc_1 (CAN Data)	1 st range for accelerator parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_acc_2 (CAN Data)	2 nd range for accelerator parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_acc_3 (CAN Data)	3 rd range for accelerator parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_acc_4 (CAN Data)	4 th range for accelerator parameter for Eco driving unified CAN unit:	differential	8-bit unsigned (ASCII text)

	1sec/bit		
eco_acc_5 (CAN Data)	5 th range for accelerator parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_acc_6 (CAN Data)	6 th range for accelerator parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_braking_1 (CAN Data)	1 st range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_braking_2 (CAN Data)	2 nd range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_braking_3 (CAN Data)	3 rd range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_braking_4 (CAN Data)	4 th range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_braking_5 (CAN Data)	5 th range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)

eco_braking_6 (CAN Data)	6 th range for braking parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_1 (CAN Data)	1 st range for pedalpress parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_2 (CAN Data)	2 nd range for pedalpress parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_3 (CAN Data)	3 rd range for pedalpress parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_4 (CAN Data)	4 th range for pedalpress parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_5 (CAN Data)	5 th range for pedalpress parameter for Eco driving unified CAN unit: 1sec/bit	differential	8-bit unsigned (ASCII text)
eco_pedalpress_6 (CAN Data)	6 th range for pedalpress parameter for Eco driving unified CAN unit:	differential	8-bit unsigned (ASCII text)

	1sec/bit		
adblue (CAN Data)	Catalyst fluid volume (Adblue or similar), unit is the same like in case of fuel level	not applicable	8-bit unsigned (ASCII text)
engine_oil_level (CAN Data)	Engine oil level, unit is the same like in case of fuel level	not applicable	8-bit unsigned (ASCII text)
alternative_gasoline (CAN Data)	Alternative fuel level, unit is the same like in case of fuel level	not applicable	8-bit unsigned (ASCII text)
axle_weight5 (CAN Data)	Current weight of 5 th axle unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
engine_torque (CAN Data)	Engine percent torque unified CAN unit: 1%/bit	not applicable	8-bit unsigned (ASCII text)
distance_to_service (CAN Data)	Total mileage to vehicle service from CAN bus unified CAN unit: 1m/bit	not applicable	32-bit unsigned (ASCII text)
axle_brutto (CAN Data)	Vehicle gross weight unified CAN unit: 1kg/bit	not applicable	16-bit unsigned (ASCII text)
coasting (CAN Data)	Driving time when engine is on, pressure on the gas pedal is 0% and vehicle speed is above 0 km/h	differential	16-bit unsigned (ASCII text)
pto_duration (CAN Data)	PTO usage time	differential	16-bit unsigned (ASCII text)
pto_fuel_consumption (CAN Data)	PTO fuel consumption	differential	16-bit unsigned (ASCII text)

clutch_presses (CAN Data)	Quantity of clutch presses	differential	16-bit unsigned (ASCII text)
retarder_usage (CAN Data)	Current retarder usage	not applicable	16-bit unsigned (ASCII text)
R1	reserved for future use	-	32-bit unsigned (ASCII text)
R2	reserved for future use	-	32-bit unsigned (ASCII text)
R3	reserved for future use	-	16-bit unsigned (ASCII text)
R4	reserved for future use	-	16-bit unsigned (ASCII text)
R5	reserved for future use	-	16-bit unsigned (ASCII text)
R6	reserved for future use	-	16-bit unsigned (ASCII text)
R7	reserved for future use	-	16-bit unsigned (ASCII text)

Inputs field description

Bit	Description	Bit value	Pin out S8
0 (LSB)	Ignition input	1 – High state on input 0 – GND on input	JP1-3 <i>(same as S6)</i>
1	No.0 input with pull-down to GND	1 – High state on input 0 – GND on input	JP1-5 <i>(same as S6)</i>
2	No.1 input with pull-down to GND	1 – High state on input 0 – GND on input	JP2-8 <i>(same as S6)</i>
3	Not used in S8 series	Always 0	Not used
4	No.0 input with pull-up to power supply	0 – High state on input 1 – GND on input	JP1-4 <i>(same as S6)</i>
5	No.1 input with pull-up to power supply	0 – High state on input 1 – GND on input	JP2-1 <i>(same as S6)</i>
6	CAN Module ALARM state (if installed)	1 – Alarm activated 0 – Alarm deactivated	Internal connection <i>(same as S6)</i>
7	Not used in S8 series	Always 0	Not used
8 (MSB)	CAN1939	1- Device is reading CAN1939 data 0- Device is not reading CAN1939 data	Software option <i>(new feature in S8)</i>
9	GPS antenna	1 – Antenna GPS connected 0 – Alarm! Antenna disconnected	Software option(<i>new feature in S8</i>)
10	CAN module	1 - connected 0 - disconnected	Software option(<i>new feature in S8</i>)
11	Internal battery	1-Connected 0-Disconnected	Software option(<i>new feature in S8</i>)
12	Roaming status	1- Device is in Roaming network 0-Device is in Home network	Software option(<i>new</i>)

			<i>feature in S8)</i>
13	Garmin	1 - Garmin connected(FMI) 0 - Garmin disconnected(FMI)	Software option(<i>new feature in S8)</i>
14	3D sensor calibration state	1 - Calibrated 0 – Not calibrated	Software option(<i>new feature in S8)</i>
15	J1708 data	1 - CAN is reading J1708 0 - CAN is not reading J1708	Software option(<i>new feature in S8)</i>

Outputs field description

Bit	Description	Bit value	Pin out S8
0 (LSB)	No.1 open collector output	1 – Output sinking current 0 – Output in Hi-Z state	JP2-2
1	No.2 open collector output	1 – Output sinking current 0 – Output in Hi-Z state	JP2-1
2	No.3 open collector output	1 – Output sinking current 0 – Output in Hi-Z state	JP2-7
3 (MSB)	Not used in S8 series	N/A	

CAN indicators description

Bit	Description	Bit value
0 (LSB)	Fuel reserve indicator	1 – Indicator ON 0 – Indicator OFF or no information
1	Seat belts indicator	1 – Indicator ON 0 – Indicator OFF or no information
2	Reserved for future use	Always 0
3	Reserved for future use	Always 0
4	Brake pedal	1 – Indicator ON 0 – Indicator OFF or no information
5	Clutch pedal	1 – Indicator ON 0 – Indicator OFF or no information
6	Handbrake indicator	1 – Indicator ON 0 – Indicator OFF or no information
7	Central lock	1 – Locked 0 – Unlocked or no information
8	Reverse gear indicator	1 – Indicator ON 0 – Indicator OFF or no information
9	Running lights	1 – Indicator ON 0 – Indicator OFF or no information
10	Low beams	1 – Indicator ON 0 – Indicator OFF or no information
11	High beams	1 – Indicator ON 0 – Indicator OFF or no information
12	Rear fog lights	1 – Indicator ON 0 – Indicator OFF or no information
13	Front fog lights	1 – Indicator ON 0 – Indicator OFF or no information
14	Doors open	1 – any door open 0 – all doors closed or no information

15 (MSB)	Trunk open	1 – trunk open 0 – trunk closed or no information
----------	------------	--

CAN indicators "C" description

Bit	Description	Bit value
0 (LSB)	Fuel reserve indicator	1 – Indicator ON 0 – Indicator OFF or no information
1	Seat belts indicator	1 – Indicator ON 0 – Indicator OFF or no information
2	Hazard lights indicator	1 – Indicator ON 0 – Indicator OFF or no information
3	Cruise control	1 – Cruise control ON 0 – Cruise control OFF or no information
4	Factory alarm triggered	1 – Factory alarm triggered 0 – Factory alarm not triggered or no information
5	Car locked by remote control	1 – Car locked by remote control 0 – Car unlocked or no information
6	Handbrake indicator	1 – Indicator ON 0 – Indicator OFF or no information
7	Central lock	1 – Locked 0 – Unlocked or no information
8	Reverse gear indicator	1 – Indicator ON 0 – Indicator OFF or no information
9	Air conditioning indicator	1 – Indicator ON 0 – Indicator OFF or no information
10	Running lights OR low beams	1 – Running lights ON or low beams ON 0 – Both lights OFF or no information
11	High beams	1 – Indicator ON 0 – Indicator OFF or no information
12	Rear fog lights	1 – Indicator ON 0 – Indicator OFF or no information
13	Front fog lights	1 – Indicator ON 0 – Indicator OFF or no information
14	Doors or trunk open	1 – any door open or trunk open

		0 – all doors and trunk closed or no information
15	Webasto	1 – Webasto ON 0 – Webasto OFF or no information
16	Electric car charging	1- car charging 0 – Indicator OFF or no information
17	ABS indicator	1 – Indicator ON 0 – Indicator OFF or no information
18	ESP indicator	1 – Indicator ON 0 – Indicator OFF or no information
19	Low level of Engine cooling fluid	1 – Low level of Engine cooling fluid 0 – Normal level of Engine cooling fluid or no information
20	Engine temperature indicator	1 – Indicator ON 0 – Indicator OFF or no information
21	Low level of brake fluid	1 – Low level of brake fluid 0 – Normal level of brake fluid or no information
22	Brake system fault indicator	1 – Indicator ON 0 – Indicator OFF or no information
23	Airbags indicator	1 – Indicator ON 0 – Indicator OFF or no information
24	Check engine indicator	1 – Indicator ON 0 – Indicator OFF or no information
25	Oil pressure indicator	1 – Indicator ON 0 – Indicator OFF or no information
26	Oil level indicator	1 – Indicator ON 0 – Indicator OFF or no information
27	Service indicator	1 – Indicator ON 0 – Indicator OFF or no information
28	Car battery indicator	1 – Indicator ON 0 – Indicator OFF or no information
29	Reserved for future use	Always 0

30	Impulse counting (only if ZVALUE 25 is set)	1 – Indicator ON 0 – Indicator OFF or no information
31 (MSB)	ECOROLL mode	1 – Indicator ON 0 – Indicator OFF or no information

Jamming/CREG/CME/CMS description

Jamming/CREG/CME/CMS value is available in 1WireTemp_6 field consist from 4 hexadecimal characters representing 16bit signed value. First 11 MSB represent last CME/CMS error value, next 3 MSB represent last CREG state and last 1 MSB represent jamming state.

For example: Last data frame have hexadecimal value in 1WireTemp_6 field – 100A. This value have to be converted to binary – 1000000001010.

<i>GSM Jamming</i>	<i>Network registration</i>			<i>GPS Jamming*</i>	<i>CME/CMS error value</i>										
GSMJ	NR2	NR1	NR0	GPSJ	EV10	EV9	EV8	EV7	EV6	EV5	EV4	EV3	EV2	EV1	EV0

GSMJ – GSM Jamming

GSMJ = 0 – jamming not detected

GSMJ = 1 – jamming detected

NR2:NR0 – Network registration

b'000' – not registered, MT is not currently searching a new operator to register to

b'001' – registered, home network

b'010' – not registered, but MT is currently searching a new operator to register to

b'011' – registration denied

b'100' – unknown

b'101' – registered, roaming

b'110' – unknown

b'111' – unknown

GPSJ – GPS Jamming*

GPSJ = 0 – jamming not detected

GPSJ = 1 – jamming detected

EV10:EV0 – CME/CMS error value

To read error value please convert binary to decimal and refer to official CME/CMS error documentation.

* GPS Jamming can be recognized only if you have L76 GPS version module on the board.

Dallas temperature description

Dallas Temperature fields consist from 4 hexadecimal characters representing 16bit signed value. First 8 MSB bits represent absolute value and last 8 LSB bits represent fraction. Table below shows some data calculation examples.

TEMPERATURE (°C)	Data (BIN)	DATA (ASCII HEX)
+125	0000 0111 1101 0000	07D0
+85	0000 0101 0101 0000	0550
+25.0625	0000 0001 1001 0001	0191
+10.125	0000 0000 1010 0010	00A2
+0.5	0000 0000 0000 1000	0008
0	0000 0000 0000 0000	0000
-0.5	1111 1111 1111 1000	FFF8
-10.125	1111 1111 0101 1110	FF5E
-25.0625	1111 1110 0110 1111	FE6F
-55	1111 1100 1001 0000	FC90

Tachograph field description

Tachograph field consists of 2 bytes representing 2 independent sets of data about both drivers. Least significant byte contain information about the first driver, while the other byte about the second. Table below describes meaning of specific bits in every driver byte. Value 0xFF of the whole byte means no information or invalid driver data.

Bit	Description	Bit value
0-2 (LSB)	Driver working time	b'000' – below limits b'001' – 15 minutes below 4,5 hours of drive b'010' – above 4,5 hours of drive b'011' – 15 minutes below 9 hours of drive b'100' – above 9 hours of drive b'101' – 15 minutes below 16 hours of drive b'110' – above 16 hours of drive b'111' – other limit
3	Driver card status	1 – driver card in slot 0 – no driver card
4-5	Driver work type	b'00' – resting b'01' – short break b'10' – working (loading/unloading/office work) b'11' – driving
6-7 (MSB)	Valid data indicator	b'00' – data in bits 0-5 is valid b'11' – invalid data – whole byte should be 0xFF

EVENTID [Report ID] codes description

0	Answer to GETPOSITION query
1	Telemetry data from archive
2	Tracking telemetry
3	Timer generated telemetry
9	Towed report
11	Input 0 (ignition) changed
12	Input 1 changed
13	Input 2 changed
14	Input 3 changed
15	Input 4 changed
16	Input 5 / Output 2 changed
17	Input 6 changed
18	Input 7 changed
19	Input 8 changed
22	iButton read
23	iButton detached
24	No ACC authorization (Event generated according only to ACCAUTH command)
25	Main power ON (Main power voltage is higher than 5V)
26	Delayed ACC authorization (Event generated according only to ACCAUTH command)
27	Authorization expire (Only for DALLAS authorization - end of

	authorization)
28	Not Authorized (Only for DALLAS authorization - ignition on and during 30 second there was no authorization)
29	Main power off (Main power voltage is lower than 5V)
32	Car towing! Device verifies sum of overloads when ignition is off (3D sensor)
33	Accident! Device verifies sum of overloads when ignition is on by use advanced algorithm (3D sensor)
34	ALARM! Antenna GPS disconnected
35	Frame generated in SLEEPMODE mode according to defined time
36	GSM Jamming detected
37	GPS Jamming detected (only if L76 GPS version module is on the board)
38	Frame generated in SLEEPMODE mode according to 3D overload
47	Alarm from VOLTDETECT feature
49	One of the EXPANDER input (-) changed its own status
50	Rapid acceleration – acceleration is verify above threshold when ignition is on (3D sensor)
51	Rapid braking – braking is verify above threshold when ignition is on (3D sensor)
52	Rapid turning – turning is verify above threshold when ignition is on (3D sensor)

53	Changed state of one of the Bluetooth TAG
63	ECOROLL start
64	ECOROLL stop
191	Barcode reader
195	Impulse counting start
196	Impulse counting stop
200	Output 1 turned on
203	Output 4 turned on
204	Output 5 turned on
205	Output 1 turned off
208	Output 4 turned off
209	Output 5 turned off
ASYNCHRONOUS EVENTS FROM CAN BUS	
65	Ignition on
66	Ignition off
67	Engine on
68	Engine off
69	Reverse gear on
70	Reverse gear off
71	Tachograph status changed (Driver card, work state, time related states)
72	All doors closed (Including trunk)
73	Any door opened (First door or trunk is opened)
74	Any door changed

	(Any door or trunk is opened or closed)
75	Trunk opened
76	Trunk closed
77	Central lock locked
78	Central lock unlocked
79	Driver seat-belt belt fastened (Indicator off)
80	Driver seat-belt unfastened (Indicator on)
81	Handbrake pulled-up
82	Handbrake released
83	Factory alarm status changed (Arming, disarming, triggering etc.)
84	Car locked with remote
85	Car unlocked with remote
86	Driver card inserted into Tachograph
90	Hazard lights on
91	Hazard lights off
92	Speed limit exceeded
93	Engine speed limit exceeded
94	Rapid acceleration occurred
95	Rapid braking occurred
96	Drive START (According to ZVALUE 22 – event will be generated when drive speed will be greater or equal than defined in ZVALUE)
97	Drive STOP

	(According to ZVALUE 22 – event will be generated when drive speed will be lower than defined in ZVALUE)
98	Exceeded accelerator pressure limit (According to ZVALUE 21)
120	TMR: Insert/eject drivers card
121	TMR: Tachograph status changed

1-Wire temperature calculation:

The value of temperature is sent in appropriate field in data frames. Format of the value contains 3-4 ASCIIHEX numbers. The table below shows examples of temperature values.

3(MSP)	2	1	0(LSP)	Decimal
F	F	F	8	-0,5°C
0	1	9	1	25,0625 °C

The value splits into a fraction and an integer. Four least significant bits are fraction part of the value. If the temperature is a positive value, four most significant bits can be omitted in the temperature field.

Examples of temperature calculation:

Temperature (h)	Integer	Fraction	Result (°C)
FFF8	$0xFFFF = -1_{U2}$	$0x08 = 0b1000 = \frac{1}{2} \times 1 + \frac{1}{4} \times 0 + \frac{1}{8} \times 0 + \frac{1}{16} \times 0 = 0,5$	$-1+0,5 = -0,5$
0191 (191)	$0x19 = 25_{U2}$	$0x01 = 0b0001 = \frac{1}{2} \times 0 + \frac{1}{4} \times 0 + \frac{1}{8} \times 0 + \frac{1}{16} \times 1 = 0,0625$	$25+0,0625 = 25,0625$
FF5E	$0xFF5 = -11_{U2}$	$0x0E = 0b1110 = \frac{1}{2} \times 1 + \frac{1}{4} \times 1 + \frac{1}{8} \times 1 + \frac{1}{16} \times 0 = 0,875$	$-11+0,875 = -10,125$

Software update procedure

It is recommended to check GSM signal strength before performing update procedure. Performing update procedure in weak GSM signal conditions may cause errors. Unit will not block update in case of weak GSM signal conditions.

Send SOFTUPDATE command. Unit will replay witch command acknowledgment.

After performing update unit will reboot its self and erase all data stored in telemetry archive. After 6 minutes unit will generate asynchronous software update report (see SOFTUPDATE command). If any part of the system was not successfully updated it is necessary to repeat update procedure. Keep in mind that time from sending software update command to receiving asynchronous software update status strictly depends on network conditions that may affect download speed. Typically in good network conditions unit generates update status report after 6-8 minutes from receiving software update command.

In case when unit does not generate software update within 10 minutes from executing command, user should **reboot** unit and repeat update procedure.

Error Codes

0	Unknown command
1	Invalid unit password
2	Invalid command parameters
3	Not supported parameter
6	CRC error in CAN update file
7	Wrong device configuration
8	CAN module disconnected
9	Update procedure error
10	No GPRS/TCP connection
11	File name error or FTP server unavailable
12	File download error or download timeout
13	Internal Error

Tachograph Memory Reading (TMR) Device (S8.5 device)

Tachograph Memory Reading (TMR) device is upgradable add on module similar to CAN module, which supports reading of tachograph memory. Three types of memory files are possible to read: main tachograph memory form specified range of time, driver card in slot 1 and driver card in slot 2. Every combination of these memories can be readed in one session resulting in up to 3 files uploaded to the server. Also D8/K-Line data reading is fully available. All dedicated command, protocol are described in dedicated documents.

Please request that!

Important additional information

- From 1 April 2016 it is available to authorize download process via local software that needs to be run in customer computer. All details are available in partnership website.
- TMR protocol are described in different document - **Please request that**
- TachoCARD authorization software is free to use - **Please request that**
- If you need to check download procedure in working tachograph – You can request access to tachograph installation in Our Company – with Company card and reader.

Please let us know if you are interested in TMR technology support

Advanced Ecodriving Solution

GPRS terminals are based on advanced modules which enables generating a number of ECODRIVING data. Adhering to the principles of ECODRIVING driver lowers the combustion in the vehicle, extends its life and improves your safety while driving. The manufacturer supplies a number of parameters calculated on the basis of:

CAN module – is now a most accurate tool on the basis of which there is very wide range of data showing the process of using the vehicle by the driver. Most of the parameters was created in a cooperation with a professional team involved in the subject of ecodriving. The data refreshed in real time in CAN bus guarantee the effectiveness of display solutions.

3D Sensor – High-quality 3D sensor (accelerometer) based on the proprietary calibration algorithm and capture of events is an ideal solution to determine the driving style. Improves safety by detecting accidents and protects the vehicle by detecting towing's as well as unauthorized movement of the vehicle. Innovative attitude to visualize the operation of the solution (a free Android application) clearly presents all the advantages of this solution. Very accurately reproduces the mechanism of the calibration of the sensor 3D, configuring events and event generation target. Obtained configuration on the phone can be sent to the target GPRS S8 / FM Terminal!

There are almost fifty ecodriving fields that are available in F12 frame format.

Please let us know if you are interested in Advanced Ecodriving Solution

Technical information about FM11

Information about way to read battery level

Battery level is available in VoltageAnalog1 field. This value should be interpreted according to below table:

Battery level	Voltage level
From 100% to 75%	From 4.20V to 3.85V
From 75% to 50%	From 3.85V to 3.71V
From 50% to 25%	From 3.71V to 3.60V
From 25% to 0%	From 3.60V to 3.50V

Command responsible for FM11 sleeping while main power is disconnected:

SLEEPMODE

Command	SLEEPMODE	
Description	Device can go to deep sleep mode, where power consumption is minimum. Device will wake up after time set in command to send one actual data frame (with refreshed GPS position, CAN data...).	
Command Syntax	\$AL+SLEEPMODE+[TAG]=[Password],[enable],[time],[sleep_cause],[3D_sens or_wakeup_threshold]	
Field Description	TAG	Command specific TAG number. If present unit answer must contain the same TAG number. (Max. 5 characters)
	Password	Unit current password. (Max. 4 characters)
	mode	0 – Disabled 1 – Device will generate data frames only according to 3D overloads and in case of sabotage (sabotage means that JP1-5 input is active and data frames will be generated every 30 seconds while this) 2 – Device will generate data frames only at defined time 3 – Device will generate data frames only according to 3D overloads and in case of change state of one of inputs

	time	Time of sleep duration. Value in minutes. Minimum time 10 minutes.
	sleep_cause	0 – Device will go to sleep when the main power will turn off 1 – Device will go to sleep when the ignition will turn off
	3D_sensor_wakeup_threshold	3D sensor threshold to generate asynchronous data frame when threshold will be exceeded. One of values 3DX, 3DY, 3DZ have to be exceeded at least by value that is inserted into 3D_sensor_wakeup_threshold.
Example Query	Query: \$AL+SLEEPMODE=0000,? Example answer: \$QR:SLEEPMODE=1,10,1,3	

Attention!

Second mode of SLEEPMODE is required while transporting GPRS Terminal FM11. After making installation preferred mode is third. The inputs works according to inputs field description.

Technical information about EXPANDER

Data from EXPANDER are available in the fields wirecode4, wirecode5, wirecode6.

Example F7 data frame that contains data from EXPANDER:

9926101,20170719083024,F7,20.936676,52.152164,354,8,24,65306,26003,22,2641,1,04.25,23.92,01.41,00.00,0,,398,398,,,,,,,,,,,,,0,,,,,,,,E3022C0000000000,,31303000FFFF0000,1000,800000190C4A4D01,,,,,,,,,800000190C4A4D01,,,,,99,4AA8

wirecode4 field:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
A0	A1	B0	B1	C0	C1

example:

0xE3	0x02	0x2C	0x00	0x00	0x00
------	------	------	------	------	------

A1:A0 (MSB:LSB) – voltage value on the Analog input 1

B1:B0 (MSB:LSB) – voltage value on the Analog input 2

C1:C0 (MSB:LSB) – voltage value on the Analog input 3

Steps to get voltage that is on the Analog input 1:

1. Convert 0x02E3 from hexadecimal to decimal.
2. Please divide decimal value by 1024.
3. Result of divide please multiply by 33.
4. Finally you have voltage on the Analog input 1 – 23,8154296875V.

The same steps are required in case of counting voltage on the other analog inputs.

wirecode5 field:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
I1	I2	I3	O	T0	T1	W	0x00

example:

0x31	0x30	0x30	0x00	0xFF	0xFF	0x00	0x00
------	------	------	------	------	------	------	------

I1/I2/I3 – state of Input 1 (-), Input 2 (-), Input 3 (-)

(MSB)

0	0	1	1	0	0	0	S
---	---	---	---	---	---	---	---

 (LSB)

S – state of input

0 – turned off (higher than 2.5V)

1 – turned on (lower than 1.5V)

O – state of Output 1, Output 2, LED diode

(MSB)

O1	O2	0	0	0	0	0	LD
----	----	---	---	---	---	---	----

 (LSB)

O1 – Output 1

0 – high impedance

1 – ground
O2– Output 1
0 – high impedance
1 – ground
LD – LED diode
0 – ground
1 – 3V on the output
T1:T0 (MSB:LSB) – temperature value (counting according to the description that is available in the 1-Wire temperature calculation section)
W – state of 1-Wire
(MSB)

0	0	0	0	0	W2	W1	W0
---	---	---	---	---	----	----	----

 (LSB)
W0 – capacity of the cable that is connected to 1-Wire bus
0 – capacity in standard
1 – capacity too high
W1 – CRC correctness of the connected device to the 1-Wire bus
0 – correct CRC
1 – wrong CRC
W2 – state of the 1-Wire bus
0 – correct
1 – wrong (the 1-Wire bus is short circuit to the ground)

wirecode6 field:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
DN0	DN1	DN2	DN3	DN4	DN5	DN6	DN7

Example:

0x80	0x00	0x00	0x19	0x0C	0x4A	0x4D	0x01
------	------	------	------	------	------	------	------

DN0:DN7 – number of DALLAS reader or DALLAS thermometer

Change Log:

- SV: Dec 4 2014 01:23:08
 - Added new value (20) to command ZVALUE. This value is used for CAN speed correction calculation ($[\text{new speed}] = [\text{Vehicle_speed}] * [\text{ZVALUE}(20)] / 100$).
 - Added max value for [engine_temp] = 9999.
 - Fixed bug that causes wrong time measurement in TIMER function
- SV: Dec 5 2014 11:10:02
 - Removed max value for [engine_temp] = 9999.
 - Added frame printing rule: [engine_temp] > 9999 => [engine_temp] = null.
- SV: Dec 10 2014 08:39:41
 - Added TRACKING, TRACKROAM mode 8
 - Added SETGPSKM mode 5
 - Fixed bug that causes inserting wrong values in [mileagegps] field after device reset
 - Rebuilt Timer mechanism
- SV: Dec 11 2014 00:41:43
 - Dallas Immobilizer: Only JP2-2 is activated
- SV: Dec 26 2014 11:33:12
 - Added startup event blocking mechanism (40 sec)
 - Added auto TIMER counter synchronization with GPS time.
 - Added rescue modem command for SA region (AT+COPS=?)
 - Added softupdate firmware for S10 Terminals.
- SV: Dec 30 2014 23:13:45
 - Added to detail info "electric car charging"
 - Replaced CAN torque with CAN range value
 - Fixed bug allowing in special condition updating S10 device with wrong firmware
- SV: Jan 5 2015 19:20:10
 - Fixed bug that causes M95 GSM logging with no pin sim card

- M95 COPS rescue function
- SV: Jan 9 2015 01:14:10
 - Waiting for GSM: 5 minutes
 - Added new field to \$AL+SS=0000,? -> MSV: Modem software version
- SV: Feb 7 2015 17:09:05
 - Added ZCONFIG 28 (IMODALLAS_INV)
- SV: Feb 7 2015 17:09:05
 - Tracking 8 bug fixed
 - GSM logic revoke revision 30 Dec.
 - bug fixed (Tr_Present = 1=> axle weight problem)
 - Added new field to \$AL+SS command-> TIMER_DEBUG: (current counter state)| additional info
- SV: Mar 14 2015 01:20:04
 - Range bug fixed
 - CAN FF bug fixed
 - Added ZCONFIG 29 (INP1+> voltageAnalog1 INP2+>voltageAnalog5)
 - M95 CAN UPDATE added functionality
- SV: Apr 1 2015 21:09:11
 - TOW bug fixed
 - added Analogfilter function
- Apr 16 2015 19:42:58
 - DALLASMODO added function
 - CAN TACHO CARD bug fixed
 - CAN D8 BUS detection
 - INPUT bits bug fixed
 - Added sequence number to F0 and F1 frame
- May 11 2015 01:37:16

- Added new parameters to ANALOGFILTER
- May 21 2015 10:18:12
 - Fixed accelerator field (limit: 8bit length)
- Jul 13 2015 14:35:59
 - Added frame generation lock (startup time 60s)
 - Added buffering last fuel state
 - Changed interaction between VIRTUALACC and IGNITION function.
 - Added new error codes to TACHPOREQ function in accordance with TACHOREADER documentation ver. 22 April 2015 (status answer: 17+[bits S1.5-S1.7])
 - Added event 120 and event 121 for async changes in TMR D8/KLINE support (drivers card, driver status and so on...)
- Jan 15 2016
 - FM23 support added
 - Wire support bug fixed (27/31)
 - Softupdate/Canupdate procedure optimization
 - Deleted SETGPSKM automatic change after auto sync CAN procedure
 - Full Mobileye support added
 - Tachograph field optimization
 - Basic bug fixed
- Feb 12 2016
 - IP - DNS during Softupdate process problem fixed
 - CAN data bug fixed (total_fuel and total_distance)
 - Inputs refreshing when input event was generated fixed
 - RTC time bug fixed
- May 5 2016
 - Zconfig:28 bug fixed with immodallas inverted
 - WIFI module support and bug fixed

- battery voltage value in F7 fixed
- DALLASMODE fixed
- Internall changes and optimization (SPI/synchronization process)
- Sleepmode added new features and bug fix
- SMS buffer overload bug fixed
- Modem start procedure fixed
- Added Quiet mode in WIFI
- FM10/FM11 support
- CRC in Can module data bug fixed
- Aug 12 2016
 - Added F12 frame format
 - Fixed support of LI2DH sensor
 - Added support of CME/CMS errors
 - Added Jamming detection
 - Added calibration state of 3D sensor to inputs field
- Nov 15 2016
 - Fixed support of downloading DDD files
 - Fixed correctness of available data in the F12 frame format
 - Fixed the troubles with automatic synchronization of CAN module (no answer after \$OK:AUTOSYNC)
 - Fixed sending data frames from archives
 - Fixed SOFTUPDATE and CANUPDATE procedures
 - Fixed error with generating data frames every 1 second if the value in heading field is 359
 - Increased range of 1-Wire sensors (cables of sensor can be longer that earlier – up to 100 meters total)
 - Added GPS Jamming detection (only with GPS modem L76)
 - Added SYNCFRAMEMODE function
- Apr 24 2017

- Fixed downloading DDD files
- Fixed reading of SMID
- Fixed problem with angle (Device was sending a lot of data if angle have value 359)
- Fixed values in [axle_all] and [service_distance] fields
- Fixed SOFTUPDATE/CANUPDATE procedure
- Fixed displaying of GSM signal by led diode
- Fixed Garmin packets
- Fixed generating data frames by TIMER function
- Fixed counter of total engine hours in case of GPRS Terminal FM15
- Fixed SMID request
- Fixed rapid accelerations and braking counters reading
- Added refreshing information from tachograph on stops in case of GPRS Terminal S8.3 D8
- Added new events basis on the CAN-bus - exceeded accelerator pressure limit and car START/STOP (ZVALUE 21, 22)
- Added function that replacing value in [range] field by impulses from JP2-7 (ZCONFIG2 5)
- Added new events that are possible to generate while SLEEPMODE
- Added function that returns error when DALLASID have index bigger than 511
- Added new events basis on DALLAS reader - iButton readed and iButton detached
- Added automatically calibration of 3D sensor
- Added 4th parameter in SLEEPMODE function (for 3D sensor)
- Added dedicated delay values in INPUTDELAY function
- Added dedicated customer feature ZVALUE23
- Added dedicated customer feature POWERMODE3
- Deleted eventid = 17 in case of GPRS Terminal FM10/FM11
- SV: Aug 1 2017 16:05:47
 - Added ZVALUE:26 – time for expiring DALLAS authorization
- SV: Aug 3 2017 13:13:51

- Added ECOROLL feature
- Added events for start/stop impulse counting (for JP2-7)
- Added EXPANDER support
- Added new POWERMODE modes
- Added new SMSREPORT modes
- Fixed reading of accelerator pedal pressure
- Fixed communication between Terminal GPRS and Garmin navigation
- SV: Jan 23 2018 17:00:46
 - Added supporting the Bluetooth 4.0 Low Energy module
 - Added COPS command that checking list of the available GSM networks
 - Added ACTIVEIMMODALLAS feature
 - Added TTL logic for inputs (ZCONFIG2:18)
 - Fixed working of the 1-Wire LED in case of the FM series
 - Fixed L76 GPS support
 - Fixed CANUPDATE process
 - Fixed LED diode working in case of FM23
 - Fixed generating data frames according to TRACKING/TRACKROAM
 - Fixed counting of analog impulses
 - Fixed wrong values of altitude
- SV: Feb 20 2018
 - Fixed transmission of data frames from archives
- SV: Feb 22 2018
 - Fixed spamming events 22 and 23 when iButton was applied to the reader all the time
- SV: Feb 26 2018
 - Fixed generation event 101
- SV: Feb 28 2018
 - ZCONFIG2: 20 – No refreshing GPS position when drivercode field is empty

- SV: Mar 15 2018
 - ZCONFIG2: 21 – Function IMMODALLAS controls output 2
 - ZCONFIG2: 22 – Function IMMODALLAS controls output 3
 - Added dedicated customer features ZVALUE28 and ZVALUE29
- SV: Mar 22 2018 15:57:10
 - Added F16 frame format
 - Fixed generating events 195 and 196
 - Added new parameters of ecodriving
- SV: Mar 26 2018 14:28:59
 - Fixed ZCONFIG2: 20
- SV: Mar 28 2018 14:27:34
 - Login time GSM BG96/UG96 has been extended
- SV: Apr 11 2018 16:51:03
 - Fixed card login in SIM900
 - Fixed fields in frame format F16 – wrong order of fields
- SV: Apr 17 2018 11:34:43
 - Added new parameters to VIRTUALACC – speed, samples to turn ignition on if switched off
 - Fixed supporting 1-Wire thermometers
- SV: Apr 20 2018 12:30:57
 - ZCONFIG:24- Added sending frames on external port when there is not TCP connection with the server
- SV: Apr 24 2018 14:47:59
 - Added ZVALUE:30- You can set baud rate here 0-115200. Default is 0 treated as 9600. You can also set unusual values.
- SV: Apr 27 2018 15:15:26
 - Bit7 has been changed to bit3 in inputs for the function which hides position
- SV: May 9 2018 17:42:26

- Added ZCONFIG2:23- The feature that allows sending and receiving commands via Rx, Tx using UARTDATA command
- Added ZCONFIG2:26- Sending data frames via RS232TTL also with default baudrate 9600 (other baudrate can be set using ZVALUE30)
- SV: May 18 2018 13:38:24
 - Fixed that a device wake up from POWERMODE (modes where GPS sleeps) and send first TRACKING/TRACKROAM data frame with 0 satellites
- SV: May 30 2018 15:54:34
 - Barcode reader support added – ZCONFIG2:27
- SV: Jun 14 2018 14:01:48
 - Increased amount of Dallas iButtons from 1-511 to 1-2047
- SV: Jun 19 2018 16:54:23
 - Added mode 7 in IGNITION
- SV: Jun 22 2018 15:36:56
 - Added mode 5 in SLEEPMODE
- SV: Jun 26 2018 13:08:35
 - Fixed problems with refreshing inputs after device reboot
- SV: Jun 28 2018 16:02:22
 - Fixed synchronizing CAN bus in case of TERMINAL FM15 CAN
- SV: Jul 11 2018 11:43:50
 - Added blocking CAN events above 99
 - Added forcing login before every reading from the tanker
- SV: Jul 17 2018 13:50:07
 - Added CANOBD command
- SV: Jul 23 2018 17:00:20
 - Fixed EVENTID 34 - When POWERMODE cause gps sleep
- SV: Aug 2 2018 15:29:52
 - Added archive compression - increased data frame twice

- SV: Aug 3 2018 16:01:02
 - Fixed TIMER mode 1
- SV: Aug 14 2018 13:36:45
 - Added ZCONFIG2:29 – Dedicated customer feature
- SV: Aug 16 2018 11:29:40
 - Added ZCONFIG2:30 – Inserting in CAN engine speed maximum amount between the frames (RPM)
- SV: Aug 27 2018 16:29:19
 - Added support for SMS number up to 20 characters
 - You can add “@” to o command send by SMS
- SV: Sep 11 2018 16:33:48
 - Fixed ZVALUE:26
- SV: Sep 17 2018 18:00:15
 - Added ZVALUE:31 – Delay for SENT-GEO feature
- SV: Sep 21 2018 15:10:31
 - Added ZCONFIG2:31 – SENT-GEO feature
- SV: Sep 28 2018
 - Fixed ZCONFIG2:20
- SV: Oct 1 2018
 - Added Jamming detection in 3G modems
 - Added SETENGINEHOURS
- SV: Oct 3 2018
 - Fixed Jamming detection in 3G modems
- SV: Oct 4 2018 13:39:47
 - Fixed CME ERROR: 4 in case of 3G modems
- SV: Oct 16 2018 16:44:34
 - Added GEOSPEED command

- Increased SMSREPORT from 253 to 1364
 - Changes GEOSPEEDEVENT
- SV: Oct 18 2018 15:08:39
 - Added mode “2” in VIRTUALACC- function enabled for FM15
- SV: Oct 22 2018 16:54:27
 - Fixed GEOSPEEDEVENT
- SV: Oct 25 2018 16:44:24
 - Support FM10_v9 PCB
- SV: Oct 30 2018 16:48:16
 - Fixed AUTOSYNC
- SV: Nov 5 2018 16:02:41
 - Changed minimum SLEEPMODE time from 10 to 8 minutes
- SV: Nov 9 2018 16:02:41
 - Added mode “2” in NETLISTENABLE - Semi-automatic logging for 2G / 3G modems
- SV: Nov 28 2018 15:12:51
 - Added BLACKBOXENABLE command
 - Added BLACKBOXREAD command