

FM Pro, FM Eco & FM Tco SMS Commands List

Table of Contents

1 SMS.....	6
1.1 Commands.....	6
1.1.1 Coords – current coordinates.....	6
1.1.2 Version – FM device version.....	7
1.1.3 Gsminfo – GSM/GPRS information.....	7
1.1.4 Imei.....	9
1.1.5 Reset.....	9
1.1.6 Connect – custom connection.....	9
1.1.7 Econnect – emergency custom connection.....	9
1.1.8 Getapn – get APN parameters.....	10
1.1.9 Setconnection – change connection configuration.....	10
1.1.10 Switchip – switch primary IP and port.....	11
1.1.11 Setio – set outputs.....	11
1.1.12 Getio – read inputs/outputs states.....	11
1.1.13 Delrecords – delete all records.....	12
1.1.14 Modrev – modem revision.....	12
1.1.15 Caninfo – can configuration info.....	12
1.1.16 Cansinfo – dual can configuration info.....	13
1.1.17 Getsd – SD card info.....	13
1.1.18 Clear obd – clear OBD values.....	14
1.1.19 Tacho – tachograph status.....	14
1.1.20 webcoords – Google maps hyperlink with coordinates	14
1.1.21 setiotime – set output for temporary period.....	15
1.1.22 Banned – temporary banned operators.....	17
1.1.23 accinfo.....	17
1.1.24 accreset.....	18
1.1.25 lastchange.....	18
1.1.26 SMS during critical process	19
1.1.27 setcfg.....	19
1.1.28 getcfg.....	20
1.1.29 setioparam.....	21
1.1.30 getioparam.....	22
1.1.31 setvalue – set specific IO values.....	23
1.1.32 set3g.....	24
1.1.33 get3g.....	25
1.1.34 dfota – update modem firmware over the FTP or HTTP server.....	25
1.1.35 enginevolt – Automatic engine detection.....	27
1.1.36 lcvselect – LCV autoselect.....	27

- 1.1.37 setlcv – set LCV group, subgroup and automatically enable possible IOs.....29
- 1.1.38 getlog/stoplog – get device log remotely.....31
- 1.1.39 info – get some specific information from the FM device via SMS32
- 1.1.40 USSD command34
- 1.1.41 clear dtc.....34
- 1.1.42 doutreset.....35
- 1.1.43 immobilizer.....35
- 1.1.44 setdinmode.....36
- 1.1.45 getdinmode.....36
- 1.1.46 SMS forward.....37
- 1.1.47 snapshot.....37
- 1.1.48 ssl status.....37
- 1.1.49 optiver.....38
- 1.1.50 iqfinfo.....38
- 1.1.51 iqfset.....39
- 1.1.52 setlock.....39
- 1.1.53 Supported SMS commands table.....41
- 1.2 Informational messages, alerts.....42
 - 1.2.1 SMS alerts with date & time.....42

Change log

Date	Version	Description
2015-08-20	1.0	Initial draft;
2015-09-08	1.1	Added: SMS command: <i>imei</i> – obtain the IMEI number;
2015-11-12	1.2	Added: Table of supported SMS commands; Added: Description of the <i>setcfg</i> , <i>getcfg</i> , <i>setioparam</i> and <i>getioparam</i> commands;
2015-11-25	1.3	Added: Description of the <i>setvalue</i> command;
2016-03-10	1.4	Added: Description of the <i>set3g</i> , <i>get3g</i> and <i>dfota</i> commands;
2016-05-30	1.5	Added: SMS command: <i>enginevolt</i> - Automatic engine detection;
2016-06-30	1.6	Added: SMS command: <i>lcselect</i> - LCV auto select; Added: SMS command: <i>getlog/stoplog</i> - get device log remotely;
2016-08-01	1.7	Updated: Description of the <i>accinfo</i> command;
2016-09-02	1.8	Change log transferred to the document;
2016-10-13	1.9	Corrections: Removed a duplicate command in the table of supported SMS commands; Description of the <i>getlog</i> SMS command; Added: Example of the <i>cansinfo</i> SMS command;
2016-11-18	1.10	Added: SMS command: <i>info</i> - customizable SMS command; Corrections: Improved the description of the <i>switchip</i> SMS command; Corrections: Improved the description of the <i>gsminfo</i> SMS command; Corrections: Added response examples after unsuccessful operation for <i>connect</i> and <i>econnect</i> SMS commands;
2016-11-22	1.11	Added: Description of the feature “Ignore empty space before the SMS command”;
2016-12-05	1.12	Added: SMS commands supported by the FM-Basic to the table of supported SMS commands;
2016-12-12	1.13	Removed: SMS command: <i>fastsleep</i> ; Minor corrections and description improvements;
2016-12-20	1.14	Improved the description of the <i>info</i> SMS command; Improved the description of the <i>getlog/stoplog</i> SMS command;
2017-01-05	1.15	Updated: Table of supported SMS commands;
2017-01-30	1.16	Updated: Description of the <i>lcselect</i> SMS command;
2017-02-14	1.17	Updated: Table of supported SMS commands; Updated: Description of the <i>accinfo</i> SMS command; Added: Description of the <i>USSD</i> SMS command;
2017-02-22	1.18	Updated: Table of supported SMS commands;
2017-03-09	1.19	Updated: Description of the <i>setio</i> SMS command; Updated: Example of the <i>setcfg</i> SMS command;
2017-03-29	1.20	Added: Another response example of the <i>getlog</i> SMS command; Updated: Table of supported SMS commands; Added: Description of the <i>clear dtc</i> SMS command;
2017-05-18	1.21	Updated: Additional description of the <i>setiotime</i> SMS command regarding jammed DOUTs;
2017-06-07	1.22	Updated: The <i>lcselect</i> SMS command with a new <i>enableIO</i> flag;
2017-06-26	1.23	Updated: Description of SMS alerts for FM4 device.
2017-06-29	1.24	Added: New SMS commands – <i>doutreset</i> , <i>immobilizer</i> , <i>resetimmob</i> ;

2017-07-25	1.25	Updated: Custom SMS text function;
2017-08-25	1.26	Added: New SMS commands – <i>setdinmode</i> , <i>getdinmode</i> ;
2017-09-05	1.27	Added: New SMS command – <i>setlcv</i> ;
2018-02-12	1.28	Added: New SMS command – <i>forward</i> ;
2018-03-08	1.29	Updated: SMS command – <i>accreset</i> ;
2018-03-20	1.30	Updated: SMS command – <i>setdinmode</i> ; Updated: SMS command – <i>getdinmode</i> ; Updated: SMS command table.
2018-08-01	1.31	Added: SMS command <i>snapshot</i> ; Updated: Supported SMS commands table.
2018-10-08	1.32	Updated: SMS commands <i>accinfo</i> and <i>accreset</i> .
2018-10-29	1.33	Added: SMS command <i>ssl status</i> – SSL authentication status; Updated: Supported SMS commands table.
2018-12-11	1.34	Removed: All commands unique to FM3 devices; Added: SMS command <i>optiver</i> – Optitemp version status; Updated: Supported SMS commands table. Updated: SMS command password description. Various formatting corrections.
2018-12-21	1.35	Added: SMS command <i>iqfinfo</i> – iQFreeze parameter status; Added: SMS command <i>iqfset</i> – iQFreeze configuration; Updated: Supported SMS commands table.
2019-02-15	1.36	Updated: SMS command <i>snapshot</i> .
2019-03-08	1.37	Updated: Supported SMS commands table.
2019-05-14	1.38	Added: IDs in SMS commands; Updated: Supported SMS commands table. Updated: SMS command <i>getapn</i> .
2019-08-23	1.39	Added: SMS command <i>setlock</i> .

1 SMS

There is a possibility to send SMS commands to the FM device. The FM device answers with an SMS message as well. This is an easy way to quickly interact with the FM device. SMS commands are used:

- to get specific information from the FM device;
- to reconfigure some parameters of the FM device;
- to influence how the FM device functions.

SMS command structure: "**(password) (command) (command text)**"

The password, command and command text are separated by a space symbol. Command text parameters are usually separated by a comma (,) symbol. The password can contain up to 16 symbols. The following symbols are permitted:

- All digits (0-9);
- Letters A-Z (both lowercase and uppercase);
- "_", "@", "#", "/", "%", "-".

Note

The password cannot match any names of SMS commands. Please refer to the Supported SMS commands table for a full list of SMS commands.

Note

SMS commands that are sent over GPRS do not require a password.

It is possible to include IDs in SMS commands, so that the server knows to which SMS command the device responds. To do so, the **Use ID in SMS commands** functionality must be enabled in the configuration. If the functionality is enabled, the ID must be added to all SMS commands, otherwise the commands will not be processed.

SMS command structure with ID: *<password> <command> <command text> ID<ID string>*
<ID string> - a 16-32 hexadecimal symbol tag. It must be entered right after ID (no spaces).

Example: *password reset ID575D6468E934C867*

Response: *Resetting device ID575D6468E934C867*

Note

The ID must be unique. If the device receives several SMS commands with the same ID, only the first one will be processed.

Note

SMS commands *gsminfo*, *getapn* and *accreset -r* will NOT return an ID even if it is included in these SMS commands.

1.1 Commands

1.1.1 Coords – current coordinates

SMS message is used to get current GPS status. Response has 8 parameters.

Parameter	Description
Time	Current GMT date & time.

lat.	Current latitude.
long.	Current longitude.
alt.	Current altitude (meters).
sat.	Currently visible satellites.
dir.	Current angle.
hdop	Current HDOP level.
state	Current GPS/GLONASS state: 1-off, 2-on no fix, 3-on got fix, 4-not responding, 5-sleep, 6-disabled.

Example: *pass coords*

Response example: *2013-04-24 07:01, lat. 46.1443183, long. 11.881766, alt. 217.5, sat. 8, dir. 198.10, hdop 100, state 3*

1.1.2 Version – FM device version

SMS message is used to get current FM device version. Response has 5 parameters.

Parameter	Description
1	Bootloader version.
2	Firmware version.
3	Hardware version.
4	GSM signal level.
5	Voltage status: 0 – lower than 8 Volts (bad), 1- higher than 8 Volts (OK). Exception for Eco4 S: 0 – lower then 6 Volts (bad), 1- higher then 6 Volts (OK).

Example: *pass version*

Response example: *5402,00.02.15,1089,5,1*

1.1.3 Gsminfo – GSM/GPRS information

SMS message is used to get GSM and GPRS information. Response has 16 parameters.

Parameter	Title	Description
-----------	-------	-------------

ST	Start Time	Date & time (GMT) from the last FM device reset/power ON.
GSM network		
OP	Operator	GSM operator number
lvl	Level	GSM signal level.
LAC	Location Area Code	16 bit number thereby allowing 65536 location areas within one GSM PLMN.
CID	Cell ID	A GSM Cell ID (CID) is a generally unique number used to identify each Base Transceiver Station (BTS) or sector of a BTS
""	Network mode	Current network mode. There are two possible values: GSM and 3G.
FM device modem parameters (M)		
I	Initialization	The number of times when FM device has tried to initialize modem since Start Time (ST).
R	Reset	The number of resets of modem since Start Time (ST).
SP	Status Pin	The number of times when modem was turned ON and turned OFF unsuccessfully.
GPRS service		
GPRS	General Packet Radio Service	Status of GPRS. There are two possible values: 0 – no GPRS / 1 – attached to GPRS.
O	Opened	The number of opened GPRS sessions.
C	Closed	The number of closed GPRS sessions.
E	Error	The number of GPRS errors.
Link with server (LK)		
O	Opened	The number of opened links.
C	Closed	The number of closed links.
E	Error	The number of link errors,
TMO	Timeout	The number of server response timeout.
Reset		
RS	Reset	Device last reset source. Possible causes: 04 - reset pin; 08 - power on/down reset; 10 - software reset; 20 - independent watchdog; 40 - window watchdog; 80 - Low power reset. 14 means 10 (software reset) + 04 (reset pin)

		In other words FM device resets himself, for example after FW update.
P	Protocol	GPRS protocol version: 0 – standard protocol; 1 – extended protocol.

Example: *pass gsminfo*

Response example: *ST:2013.04.20 23:26:33; OP 22210,lvl 15,LAC 20030, CID: 28289, GSM; M:I 126, R 125, SP: 0; GPRS 0:O 64,C 0, E 248; LK:O 575, E 1, TMO 126; RS: 04; P 0*

1.1.4 Imei

SMS message is used to get device IMEI number.

Example: *pass imei*

Response example: *IMEI: 863071016796615*

1.1.5 Reset

SMS message is used to reset FM device. Device will reboot, its configuration parameters will not be lost.

Example: *pass reset*

Response example: *Resetting device*

1.1.6 Connect – custom connection

SMS message is used to force FM device to connect (for one time) to server with custom IP, port and protocol settings. FM device creates dummy record just with header part. Triggered event ID = 0 (zero). SMS has three parameters. SMS format: *pass connect IP,Port,Protocol*

Parameter	Description
IP	32-bit number, commonly known as an Internet Protocol address (xxx.xxx.xxx.xxx).
Port	16-bit number, commonly known as the port number (xxxxx).
Protocol	The principal of communication. There are two available protocols: TCP and UDP.

Example: *pass connect 192.168.0.1,7011,TCP*

Response example: *connection data ok*

After unsuccessful operation device responds: *connection data incorrect*

1.1.7 Econnect – emergency custom connection

SMS message is used to force FM device to connect (for one time) to server with custom APN, user, pass, IP, port and protocol settings. FM device creates dummy record just with header part. Triggered event ID = 0 (zero). SMS has five parameters. SMS format: *pass econnect apn,apnLogin,apnPassword,IP,Port,Protocol*

Parameter	Description
-----------	-------------

APN	An Access Point Name (APN) is the name of a gateway between a GPRS (or 3G, etc.) mobile network and another computer network, frequently the public Internet.
User	User name for APN settings.
Pass	Password for APN settings.
IP	32-bit number, commonly known as an Internet Protocol address (xxx.xxx.xxx.xxx).
Port	16-bit number, commonly known as the port number (xxxxx).
Protocol	The principal of communication. There are two available protocols: TCP and UDP.

Example: *pass econnect apn,apnlogin,apnpass,192.168.0.1,7011,TCP*

Response example: *Emergency connection data ok*

After unsuccessful operation device responds: *Emergency connection data incorrect*

1.1.8 Getapn – get APN parameters

SMS message is used to get APN (APN), username (USER), password (PSW), IPs (IP1, IP2), ports (Port1, Port2) and protocol (TCP/UDP) settings from FM device.

Example: *pass getapn*

Response example: *APN: banga User: PSW: IP1: 92.62.134.38 Port1: 9021 IP2: 195.14.173.3 Port2: 9000 TCP/UDP: 0*

*TCP/UDP: 0 – TCP, 1 – UDP.

Note

If the *getapn* response text is very long (for example, the APN, username and password are 32 symbols long each and two servers are used), it might not fit into a single message and be cropped.

1.1.9 Setconnection – change connection configuration

SMS message is used to permanently change FM device configuration settings: APN, APN username, APN password, protocol, IP1, PORT1, IP2, and PORT2 (parameter description: 3.1.6).

SMS format: *pass setconnection apn,apnlogin,apnpassword,Protocol,IP1,Port1,IP2,Port2*

Example: *pass setconnection apn,apnlogin,apnpass,TCP,111.111.111.111,1111,222.222.222.222,2222*

Response example: *set connection data ok*

If one of the parameters should be preserved, then the specific location for the parameter should be filled with **old**. For example, ip1 and port1 should be preserved (old value should remain):

Example: *pass setconnection apn,apnlogin,apnpass,TCP,*old*,*old*,222.222.222.222,2222*

Response example: *set connection data ok*

If the message is not ending with port2, then those parameters, which are not mentioned in the message should not be changed. For example ip2 and port2 were not in the message:

Example: *pass setconnection apn,apnlogin,apnpass,TCP,111.111.111.111,1111*

Response example: *Set connection data ok*

If configuration failed to set, when FM device sends response: *Set connection data incorrect*

1.1.10 Switchip – switch primary IP and port

SMS message is used to change current primary IP and port (IP1, Port1 or IP2, Port2). This change is temporary and applies only until device reset. After the reset IP and port configuration will be restored to the previous state.

SMS format: *pass switchip X*

X – which IP and port should be primary

IP1 primary set example: *pass switchip 1*

Response example: *Setting primary IP OK*

IP2 primary set example: *pass switchip 2*

Response example: *Setting primary IP OK*

If the operation is unsuccessful, then the answer is: *Setting primary IP FAIL*

1.1.11 Setio – set outputs

SMS message is used to set Dout1 and Dout2 output level. Values: 0 – low, 1 – high, 2 – do not change. Remember: Douts have to be connected to electric circuit correctly.

SMS format: *pass setio X1,X2*

X1 – state of Dout1

X2 – state of Dout2

Example: *pass setio 0,1*

Response example: *SETIO configuration data ok*

If configuration SMS is incorrect, device will response: *SETIO configuration data incorrect*

Note

“Setio” SMS command will not work with several FM device configurations. If a DOUT is configured to “Blocking” and driver registration is enabled and registration function is set to blocking / or “DOUT activation under conditions” enabled and DOUT is set as an output channel, then “setio” SMS command will have no effect.

1.1.12 Getio – read inputs/outputs states

SMS message is used to get status about Dout1, Dout2, Din1, Din2, Din3, Din4, Ain1 and Ain2. Values: 1 – high, 0 – low. Analog inputs – millivolts.

SMS format: *pass getio*

Answer SMS format: *DIN1=X,DIN2=X,DIN3=X,DIN4=X,DOOUT1=X,DOOUT2=X,AIN1=Y,AIN2=Y*

X – Digital value: 1 – high, 0 – low.

Y – analog value in millivolts.

Example: *pass getio*

Response example: *DIN1=0,DIN2=1,DIN3=1,DIN4=1,DOOUT1=0,DOOUT2=0,AIN1=4210,AIN2=8600*

1.1.13 Delrecords – delete all records

SMS message is used to delete all records from internal flash memory FM device memory.

Example: *pass delrecords*

Response example: *all records deleted*

1.1.14 Modrev – modem revision

SMS message is used to get modem revision information. Answer SMS format: *Modem revision: <info>*

Example: *pass modrev*

Response examples:

- M95 modem: *Modem revision: M95FAR02A06*
- UG96 modem: *Modem revision: UG96LNAR02A04E1G*
- MC60 modem: *Modem revision: MC60CAR01A07*

1.1.15 Caninfo – can configuration info

SMS message is used to get information about the settings of the CAN1 interface. These settings are used to see exact CAN interface setup in configuration file (it doesn't necessarily reflect actual mode of operation). Answer SMS format: *CAN enable: X Manufacturer Y Type Z Active A*

Parameter	Description																		
CAN enable	0 – CAN is disabled / 1 – CAN is enabled, FMS standard mode / 2 – CAN is enabled, LCV mode / 3 – CAN is enabled, OBD mode / 4 – CAN is enabled, Tachograph mode.																		
Manufacturer*	<p>Manufacture group of Light Commercial Vehicles (number value).</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>VAG</td> </tr> <tr> <td>2</td> <td>Mercedes</td> </tr> <tr> <td>3</td> <td>Citroen</td> </tr> <tr> <td>4</td> <td>Ford</td> </tr> <tr> <td>5</td> <td>Fiat</td> </tr> <tr> <td>6</td> <td>Opel</td> </tr> <tr> <td>7</td> <td>Renault</td> </tr> <tr> <td>8</td> <td>Toyota</td> </tr> </tbody> </table>	Value	Name	1	VAG	2	Mercedes	3	Citroen	4	Ford	5	Fiat	6	Opel	7	Renault	8	Toyota
Value	Name																		
1	VAG																		
2	Mercedes																		
3	Citroen																		
4	Ford																		
5	Fiat																		
6	Opel																		
7	Renault																		
8	Toyota																		

		9	FMS Tractor
Type*	Type of Light Commercial Vehicle (number value).		
Active	0 – CAN mode is silent mode, 1 – CAN mode is active mode		

Example: *pass caninfo*

Response example: *CAN enable: 2 Manufacturer 2 Type 1 Active 0*

* CAN mode is LCV, vehicle manufacture group Mercedes, type is Mercedes1, CAN is in silent mode.

1.1.16 Cansinfo – dual can configuration info

SMS message is used to get information about CAN1 and CAN2 interface settings of FM device. These settings are used to see the exact dual CAN interface setup in configuration file (it doesn't necessarily reflect actual mode of operation).

Answer SMS format: *CAN1 enable: X Manufacturer Y Type Z Active A; CAN2 enable: X Manufacturer Y Type Z Active A* (see *caninfo* message for fields description)

Example: *pass cansinfo*

Response example: *CAN1 enable: 2 Manufacturer 2 Type 1 Active 0; CAN2 enable: 2 Manufacturer 2 Type 1 Active 0*

1.1.17 Getsd – SD card info

SMS message is used to get information about SD card inserted into FM Tco or FM Pro device. There are 4 available answers.

SD card is inserted and used for records: *Using SD Card for Records. Size: sector_count x sector_size B, H: SDrecordHead, T: SDrecordTail*

SD card is inserted and used for log: *Using SD Card for Log and working OK. Size: sector_count x sector_size B, H: SDrecordHead, T: SDrecordTail*

SD card is inserted and used for log, but some errors occurred: *Using SD Card for Log but ERROR. Size: sector_count x sector_size B, H: SDrecordHead, T: SDrecordTail*

Parameter	Description
-----------	-------------

sector_count	the number of sectors in SD card.
sector_size	the size (in bytes) of one sector in SD card.
SDrecordHead	SD card address of the last of record's end.
SdrecordTail	SD card address of beginning of the first record

SD card is not inserted: *Using Internal Flash for Records*

Example: *pass getsd*

Response example: *Using SD Card. Size: 3911680 x 512 B, H: 6008, T: 5993*

1.1.18 Clear obd – clear OBD values

SMS message is used to clear all OBD (On-board diagnostic) related data in device memory (not configuration). It can be assumed as OBD values reset command.

Example: *pass clear obd*

Response example: *OBD parameters and DTC cleared*

1.1.19 Tacho – tachograph status

SMS message is used get tachograph status information. Answer SMS format: *TACHO status:X*

Status (X)	Description
0	Not available.
1	Everything is OK.
2	Tacho parameters not configured.
3	Extender not responding.
4	Tacho task is in progress.
5	Physical communication OK, logical is not OK.

Example: *pass tacho*

Response example: *TACHO status:1*

1.1.20 webcoords – Google maps hyperlink with coordinates

SMS message is used to get hyperlink to Google maps with coordinates, current vehicle speed and ignition status.

Parameter	Description
Time	GMT date & time of coordinates
Hyperlink	Hyperlink to Google maps with coordinates: latitude and longitude
Speed	Current vehicle speed, km/h
Ignition	Ignition status

Syntax: *pass webcoords*

Can be 3 different SMS message responses.

1. When GPS data is available.

Response example: *2015-05-11 13:01, https://www.google.com/maps/?q=54.7404933,25.2222366, speed: 94, ignition: ON*

2. When GPS not available.

Response example: *GPS data not available. Ignition: ON*

3. When GPS not available but was available then device can return last known coordinates and the time shows when the coordinates were taken.

Response example: *No GPS. Last entry: 2015-05-16 17:32, https://www.google.com/maps/?q=54.7404933,25.2222366, speed: 65, ignition: ON, current ignition: OFF*

1.1.21 setiotime – set output for temporary period

Using this feature, FM is able to switch the DOUTs for temporary period of time. User must provide pulse lengths for logical '1' & '0' values.

SMS format example: *pass setiotime 1 500 0 500,0 200 1 300*

Description:

Parameter	Description
pass	SMS password
setiotime	Command Identifier
1	DOUT1 first logical state (1/0)
500	DOUT1 first logical state length (ms)
0	DOUT1 second logical state (1/0)
500	DOUT1 second logical state length (ms)
,	DOUT separator
0	DOUT2 first logical state (1/0)
200	DOUT2 first logical state length (ms)
1	DOUT2 second logical state (1/0)
300	DOUT2 second logical state length (ms)

Impulse resolution is 10 ms. Minimum impulse duration is 10 ms. If Eco-panel is connected, minimum resolution and duration is 50 ms. When the FM device receives the *setiotime* command, it stores the status of DOUT's and after the sequence restores the previous status. If one *setiotime* command interrupts another, the state is **UNDEFINED** ('1' or '0')

after both sequences are finished. Interrupt in this case means that several consecutive *setiotime* commands were sent, one overlapping the other. Sending several *setiotime* commands in a fast succession will make the the DOUT stuck, thus requiring a device reboot. It is highly recommended to at least wait until a response is received for the first sent *setiotime* command, before sending the next.

'0' means **GND** and '1' means **no GND**.

If one of the DOUT is configured as LED, Buzzer blocking or Jamming block, *setiotime* is not possible to use.

Additional feature. It is possible to set a repeat amount of a sequence.

SMS format: *pass setiotime 1 500 0 500 n=10,0 200 1 300 n=20*

Description:

Parameter	Description
pass	SMS password
setiotime	Command Identifier
1	DOUT1 first logical state (1/0)
500	DOUT1 first logical state length (ms)
0	DOUT1 second logical state (1/0)
500	DOUT1 second logical state length (ms)
n	Repetition identifier
=	Setter symbol
10	Amount of repetitions
,	DOUT separator
0	DOUT2 first logical state (1/0)
200	DOUT2 first logical state length (ms)
1	DOUT2 second logical state (1/0)
300	DOUT2 second logical state length (ms)
n	Repetition identifier
=	Setter symbol
20	Amount of repetitions

Each single DOUT has 10 slots for impulses levels (high / low). No more than 10 can be defined for one DOUT.

Longest time possible is 999 999 999 ms. Max number of repeats is 9999.

Max impulse count – as many as you can fit into 160 symbols of SMS.

It is possible to interrupt a sequence with “pass setiotime 0 10,0 10” followed by setio SMS command “pass setio 1,1”.

The sequence would be interrupted and the states of DOUT’s is defined that’s it.

It is possible to set only one of the DOUT.

SMS format example for set DOUT1: *pass setiotime 1 500, 0 500*

SMS format example for set DOUT2: *pass setiotime 0 200, 1 300*

Response example: *setiotime set OK*

If configuration SMS is incorrect, device will response: *setiotime syntax error in DOUTX settings: error text*

Where „DOUTX“ is DOUT1 or DOUT2. „Error text“ is described below in the table. It's minimal diagnostic when something wrong with impulse set in SMS.

Error text	Error description
other process controls output	Means that other functionality is configured on DOUT's. For example: LED or buzzer
no comma symbol	No comma separator. Comma is used to separate DOUT's configuration
wrong format	When not enough symbols were received
wrong level	Level can be only logical '1' or '0'. Other values is unacceptable
no space symbol	After level symbols must be space symbol
wrong ms number	Wrong time number
wrong repeat number	Wrong repeat count number
exceeded max slot	For one output is possible to set maximum 10 slots. If this count is exceeded this error is displayed
pulse is shorter than 10ms	Pulse length less than 10 milliseconds was set. It's not acceptable
other reason	Includes other reason not described in this table

1.1.22 Banned – temporary banned operators

SMS message is used to get information about temporary banned operators. Answer SMS format: *Already banned:X, Newly banned:Y, ops:ZZZ*.

Parameter	Description
X	Number of times when FM tried to ban operator which has already been in the banned list.
Y	Number of times when FM added operator in the banned list.
ZZZ	List of currently banned operators (which are still in the list).

Example: *pass banned*

Response example: *Already:1, Newly banned:1, ops:24602,*

1.1.23 accinfo

This SMS message is used to check if the accelerometer is calibrated for the Eco-Drive functionality. Only state information should be interpreted.

Parameter	Description
State	2 – accelerometer calibration has been initiated

	<p>3 – accelerometer calibration is in progress</p> <p>11 – accelerometer calibrated</p>
<i>XYZo, Xc, Yc, Zc</i>	Parameters used for accelerometer event calculations.

Example: *pass accinfo*

Response example: *AXL state:11 XYZo:120 -39 -945 Xc:131 -92 -94 -7 Yc:-138 97 -96 16 Zc:953 120 -39 -945*

1.1.24 accreset

This SMS message is used to reset the existing accelerometer (used for the Eco-Drive functionality) calibration and initiate a new automatic calibration process. For automatic calibration to be successful, the data source must be set to *ACC, CAN+OBD* or *OBD+ACC*.

Example: *pass accreset*

Response example: *Acc reset OK*

An additional parameter can be included in the main SMS command which will reset the existing calibration, initiate a new automatic calibration process and inform the user of the ongoing calibration status:

Example: *pass accreset r*

Response example if the conditions are correct to start calibration: *Acc calibration initiated. Drive normally to calibrate.;*

Response example if the conditions are incorrect: *Configuration error;*

Afterwards the device will send SMS messages based on the calibration status:

Up direction found. Drive normally to finish calibration. – the calibration is in progress.

Calibration successful – the calibration process was finished successfully.

1.1.25 lastchange

SMS message is used to know when was the last time device configuration or primary server IP changed.

Example: *pass lastchange*

If time is not synchronized, or changes occur during that time, SMS content will be:

Response example: *lastchange ip: no time available; cfg: no time available*

If time is available:

Response example: *lastchange ip: 2015-11-09, 09:02; cfg: 2015-11-09, 10:02*

Configuration change date and time will be updated during any type of configuration modification.

When device is connecting to GPRS and tries to open link (with different IP settings), IP change date and time will be updated. When device reconnects to server with same IP but different Port, IP change date and time will not be updated.

1.1.26 SMS during critical process

There are 4 critical processes:

- Firmware update
- Configuration update
- Tacho read
- Smart Card read

During these processes following SMS commands will be ignored:

- reset
- connect
- econnect
- switchip
- setconnection
- delrecords
- setcfg
- getcfg
- setioparam
- getioparam
- clear obd
- tacho

Response example: *The device is busy with critical process. Please try again later.*

1.1.27 setcfg

This SMS message is dedicated to change FM parameters. After SMS password, user inputs parameter ID and parameter value:

password **setcfg** ParamID1 Value1, ParamID2 Value2, ParamID3 Value3, ...

Examples:

password setcfg 101 wave, 102 pioneer, 103 , 100 1

Sets APN name: wave, APN user: pioneer, no APN password, Protocol: UDP

password setcfg 4362 6, 4202 1, 4242 12, 4282 5, 4322 10

Sets IO in 4 profile, slot 3. IO modem temperature: enabled, level is 12, delta is 5, average is 10, event on – hysteresis on both

Only parameters provided with SMS are changed, others remain the same.

One SMS message can be 160 symbols long, so the maximum simultaneously transmitted parameters count in SMS message are limited to one message length.

Setcfg command, allows user to change all parameters of the configuration except timetable (because it will not fit into one SMS message), passwords (configuration and SMS) and “enable SMS configuration”.

Following responses are provided for the number, which sent configuration by SMS:

- When successfully setting parameters for FM device: „Configuration parameter(s) was set!“
- Incorrect parameter setting: „Configuration parameter(s) was NOT set! Parameter No. 1 is incorrect“.
- Unsuccessful set-up, when parameters are locked: „Configuration parameter(s) was NOT set! Parameter No. 1 is locked“
- Unsuccessful set-up, when setting IO parameters, but not every setting was provided by SMS: „Configuration parameter(s) was NOT set! IO slot No. 9 is not fully set“.
- Unsuccessful set-up, when device is busy: „Configuration parameter(s) was NOT set! Device is busy try again later“
- If SMS configuration is disabled in configuration tool: „You do not have permission to change the settings“.

SMS configuration feature is enabled/disabled in configuration tool, in Authorized numbers section, by putting a checkbox by the „Enable SMS configuration“.

1.1.28 getcfg

Structure of getcfg SMS:

password **getcfg** id

This command is only used for getting the current status of the parameter selected. Maximum simultaneously received parameters in one SMS message is limited by SMS message length (160 symbols).

When parameter ID is correct, FM-device answer: „ID: XXX,value:XXX;“

Example: ‘ID: 96,value:1000;’

Value can be string or number.

Multiple parameter values can be requested in one SMS message and when response does not fit in one message then in the end of the message there will be a phrase: “other values not fit”.

Example:

Request: “password getcfg 100,101,102,110,120,111,121,130”

Response:

“ID:100,value:1; ID:101,value:aerospace; ID:102,value:laguna; ID:110,value:101.16.17.245; ID:120,value:23451; ID:111,value:m2m.member.com; other values not fit”

Parameters configuration password and SMS password are not reachable with this SMS. In this case, the device answers: „ERROR: parameter(s) read is forbidden“

If parameter ID is invalid, answer would be: „ID:XXX,value:requested ID not found;“

Example: “ID:556,value:requested ID not found;“

Other possible replies when something is wrong:

“ERROR: request is empty”

“ERROR: wrong request syntax”

“ERROR: allowed numbers and commas only”

“ERROR: requested parameter ID too big”

“ERROR: requested parameter ID is low”

If SMS configuration is disabled in configuration tool or configuration has password:

„You do not have permission to read the settings“

1.1.29 setioparam

Structure of setioparam SMS:

```
password setioparam  
id=id,profile=profile,enable=enable,level=level,delta=delta,average=average,eventon=eventon,include=include,prior  
ity=priority,switch=switch,edge=edge
```

Example:

```
01234567890123456 setioparam  
id=256,profile=4,enable=1,level=3500,delta=250,average=2000,eventon=2,include=1,priority=1,switch=1,edge=3
```

This SMS can be used in any way, but it must at least contain *id*, *profile* and *enable* parameters. Shortest possible message should be written this way:

```
password setioparam id=id,profile=profile,enable=enable
```

Example:

```
01234567890123456 setioparam id=256,profile=4,enable=1
```

If the short version is used the other parameters are set with old values. *id*, *profile* and *enable* parameters are mandatory to make a valid SMS command.

Note that only two spaces between *password*[space]*setioparam*[space]*id* are required. All other commands are separated by comma.

SMS can be written in lower case and upper case letters. Therefore, **setioparam** SMS is not case sensitive.

If SMS command doesn't define new values to **level**, **delta**, **average**, **eventOn**, **include**, **priority**, **switch** or **edge** parameters, it will not be changed. Previous value will remain.

When the FM-device receives the SMS where the parameter **enable** is set to 1, it searches in the current configuration if this parameter with the same ID is already enabled.

If parameter with the same ID was not enabled, FM-device will set this IO parameter to the first free slot that is available.

If IO parameter with the same ID was enabled, FM-device will overwrite the IO parameter with new values.

If FM-device finds more than one IO parameters with the same ID it will send an error message.

If newly created IO event is not set to specific parameters it will be set to default. Default values are listed below:

Level	Delta	Average	EventOn	Include	Priority	Switch	Edge
0	0	1000	2	0	0	0	0

For set enable, eventOn, include, priority, switch on edge it is necessary to indicate the state with a number:

Enable	EventOn	Include	Priority	Switch	Edge
0 – disable 1 – enable	0 – Hysteresis 1 – Change 2 – Monitoring	0 – not include data 1 – include data	0 – Low 1 – High	0 – no switch 1 – 1 st profile 2 – 2 nd profile 3 – 3 rd profile 4 – 4 th profile	0 – On Both 1 – On rising 2 – On falling

Following responses are provided for IO configuration by SMS:

- If IO parameter was set correctly, the FM-device would answer:
„setioparam OK, slot: XX“

Where XX is the slot number where the FM-device set parameter.

- If IO parameter was not set correctly, FM-device would answer:
„setioparam ERROR, <explanation>“
- If all slots are full, FM-device would answer:
„setioparam ERROR, no free slots for set the I/O“
- If FM-device finds more than one IO parameters with the same ID it would answer:
“setioparam ERROR, more than one I/O with same ID”
- If user sent a SMS to disable a parameter (enable field = 0), and the parameter is not found as enabled, FM-device would answer:
„setioparam ERROR, parameter is already disable“
- If SMS configuration is disabled in configuration tool:
„You do not have permission to change the settings“

1.1.30 getioparam

Structure of getioparam SMS:

password **getioparam** id,profile

Example: *password getioparam 28,3*

This command is only used for get the current status of the IO parameter selected.

- If IO parameter was enabled, the FM-device would answer:

Example: *“id=28,profile=3,enable=1,level=0,delta=0,average=1000,eventon=2,include=0,priority=0,switch=0,edge=0”*

- If IO parameter was not enabled, the FM-device would answer:

„I/O ID XXX is NOT enabled“

- If IO parameter ID is invalid, the FM-device would answer:

„ERROR: I/O ID XXX does not exist“

- Other possible replies when is something wrong:

“ERROR: I/O ID XXX read is forbidden”

“ERROR: more than one I/O parameter with same ID was found”

“ERROR: wrong request syntax”

Where XXX – IO ID

- If SMS configuration is disabled in configuration tool or configuration has password:

„You do not have permission to read the settings“

1.1.31 setvalue – set specific IO values

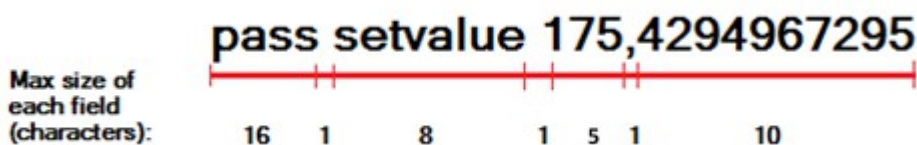
Purpose is to set specific IO values.

Parameter	Pro4 family	Eco4 family	Plug4
Virtual odometer (ID:65)	•	•	•
Absolute idling time (ID:175)	•	•	•
CAN Bus Distance (ID:114)*	•		
DIN1 hours accumulated (ID:577)	•	•	
DIN2 hours accumulated (ID:578)	•	•	
DIN3 hours accumulated (ID:579)	•	•	
DIN4 hours accumulated (ID:580)	•	•	

**Value will be overwritten by valid message from CAN-Bus (if available in specific vehicle).*

In general SMS structure is: *“(password) (command) (command text)”*

In “command text” field there will be ID and its value separated by comma (,).



The response format depends on these options:

1. If IO value was changed successfully, then format is: ID,value
2. If FM device failed to change IO value, then format is: ID,fail

3. If the value change for specified IO is not supported, then format is: ID,unsupp
4. If the FM device is even failed to parse the ID or data is incorrect: Set IO value data incorrect
Example: *pass setvalue 65,0* Response example: *65,0*
Example: *pass setvalue 175,0* Response example: *175,0*
Example: *pass setvalue 155,15* Response example: *155,unsupp*

1.1.32 set3g

The structure of the command would be:

password set3g 1/0/2

The field enable/disable could have three values: '1' means enabled, '0' means disabled and '2' means automatic. Only current profile would be changed.

Example: *password set3g 1*

In this example, 3G would be enabled.

Response from FM device:

Following responses are provided for the number, which sent the set3g by SMS.

- Response format:
3G automatic/disabled/enabled mode set OK/FAIL (1/2/3/4-profile)
- If device enabled 3G. For all examples current profile is 1:
"3G enabled mode set OK (1-profile)"
- If device could not enabled 3G:
"3G enabled mode set FAIL (1-profile)"
- If device disabled 3G:
"3G disabled mode set OK (1-profile)"
- If device could not disable 3G:
"3G disabled mode set FAIL (1-profile)"
- If device change to automatic 3G:
"3G automatic mode set OK (1-profile)"
- If device could not change to automatic 3G:
"3G automatic mode set FAIL (1-profile)"
- If field enabled/disabled is out of range:
"Set3G ERROR: Field is out of range"
- If HW version does not support 3G:

"Set3G: modem does not support 3G"

1.1.33 get3g

The structure of the command would be:

password get3g

No fields are required to fill into the SMS.

Example: *password get3g*

Following responses are provided for the number, which sent the get3g by SMS (response from FM device).

Response format:

automatic/disabled/enabled 3G mode, connected to 3G/GSM (1/2/3/4-profile)

- If device has enabled 3G and status is OK. For all examples current profile is 3:
"enabled 3G mode, connected to 3G (3-profile)"
- If device has enabled 3G but it is not connected to the 3G network:
"enabled 3G mode, connected to GSM (3-profile)"
- If device has disabled 3G:
"disabled 3G mode, connected to GSM (3-profile)"
- If device has automatic selection 3G and it is working in 3G:
"automatic 3G mode, connected to 3G (3-profile)"
- If device has automatic selection 3G and it is working in GSM:
"automatic 3G mode, connected to GSM (3-profile)"
- If HW version does not support 3G:
"Get3G: modem does not support 3G"

1.1.34 dfota – update modem firmware over the FTP or HTTP server

This SMS is used to update modem firmware of a device with 3G GSM modem. Firmwares of 3G modems are in diff form so it only sends differences between new and current firmware. Because of this, current modem firmware has to be known before the update. Binary file of Modem firmware must be placed in FTP or HTTP server before update.

Modem firmware over the FTP server

The structure of the command would be:

password dfota ftp://<user_name>:<password>@<server_URL>:<port>/<file_path>

<user_name> - String type, the user name for authentication. The maximum size of the parameter is 50 bytes

<password> - String type, the password for authentication. The maximum size of the parameter is 50 bytes

<server_URL> - String type, the IP address or domain name of the FTP server. The maximum size of the parameter is 50 bytes

<port> - Integer type, the port of the FTP server. The default value is 21. The range is 1 – 65535

<file_path> - String type, the file name in FTP server. The maximum size of the parameter is 50 bytes

Example of FTP update command:

```
password dfota ftp://test:test@demo.com:21/update/delta.bin
```

Modem firmware over the HTTP server

The structure of the command would be:

```
password dfota http://<http_server_URL>:<http_port>/<http_file_path>
```

<http_server_URL> - String type, the IP address or domain name of the HTTP server

<http_port> - Integer type, the port of the HTTP server. The default value is 80. The range is 1 – 65535

<http_file_path> - String type, the file name in HTTP server

Example of HTTP update command:

```
password dfota http://www.example.com:80/delta.bin
```

Note: total SMS message size with password, command name and link to file can be maximum 160 symbols long!

Responses to sent commands

After device receives dfota command it responds:

When modem update was started successfully:

- *Modem FW update process started. Check modem FW version with "modrev" command later*

When modem firmware update does not start because of bad SMS input:

- *ERROR: DFOTA FW update NOT supported on this modem* – when trying to update modem that is not 3G
- *ERROR: link string too short* – when minimum link requirements are not met
- *ERROR: wrong command syntax* – something wrong with command syntax
- *ERROR: DFOTA FW update process already started* – when update process was started previously
- *ERROR: other reason* – when there is another error not described above

After modem FW update is finished FM device responds (does not work on SMS via GPRS):

Modem firmware updated successfully:

- *Modem FW update process successful. Current modem version: xx*

Modem firmware update failed:

- *Modem FW update process FAILED. Error code: xx*

(error codes are specified in modem documentation)

1.1.35 enginevolt – Automatic engine detection

This SMS message is used to automatically measure engine power voltage upper level, when engine is on and lower level, when engine is off. With these voltage levels the device will automatically detect, when the engine is on and when it is off. Measured voltage levels are used in all profiles. Before sending this SMS command the engine must be on.

The structure of the command would be:

password enginevolt

<password> - String type, the password for authentication. The maximum size of the parameter is 50 bytes

<enginevolt> - Automatic engine detection command.

If message structure is wrong, FM-device answers "*Unknown command*". If message structure is correct, then after FM-device measured upper voltage level, device answers "*Turn off engine*". After this response, engine must be turned off. When after 30 s, device measures lower voltage level. If engine detection is done, device answers "*Engine detection done*". If engine detection is not successful, response from the device is "*Engine detection error*".

1.1.36 lcvselect – LCV autoselect

SMS message is used to automatically select a correct LCV group and LCV sub group combination. It reads and analyses CANbus line data. After that it searches for a match within its LCV profile database and selects the best possible configuration.

With the SMS message user can also provide "distance" and/or "RPM" values. These values are used by the auto selection process to make a more accurate selection. Additionally the user can add a optional parameter to enable all possible IOs for the selected LCV group on the active profile after the process of LCV group selection is complete.

Note

Enable IO parameters function requires the use of Protocol v1.1, without it enabled, this function will not work. Additionally enough free IO slots must be available to accommodate the newly enable IO parameters.

Configuration compactible with "LCV auto select" feature

The LCV auto selection functionality will work only in CAN "silent" mode. This functionality will not work with Toyota1, Toyota2 groups, because CAN "active" mode for these groups is needed. Also, Tractor1, Tractor2, Truck1, Truck2, Claas1 and Komatsu1 groups will not work.

LCV selection can be performed, if vehicle engine is started, vehicle is not moving and if one of these four statements is true:

1. In both CAN1 and CAN2 interfaces LCV modes are selected.
2. In one CAN interface LCV mode is selected and another CAN interface is not enabled.
3. In one CAN interface LCV mode is selected and another CAN interface Tacho read, MobileEye or Trailers are selected.
4. Both CAN interfaces are not enabled.

SMS command

The structure of the command is:

password lcvselect distance,RPM,enableIO

<password> - String type, the password for authentication. The maximum size of the parameter is 50 symbols

<lcvselect> - Automatic select of correct LCV group and sub group combination command name.

<distance> - Distance travelled by a vehicle, which is currently displayed by the vehicle's odometer. Make sure to enter the same value, it has to be a natural number. Error cannot exceed ± 1 km. Maximum value, which could be sent via SMS is 21055406 km.

<RPM> - RPM value currently displayed in the vehicle's dashboard. Make sure to enter the same value, it has to be a natural number. Error cannot exceed ± 100 rev/min. Maximum value, which could be sent via SMS is 8031 rev/min.

<enableIO> - Command to automatically enable IOs for the automatically selected LCV group. If command is not included or a "0" is written, then the function will not automatically enable IO parameters. If "1" is selected, IOs will be enabled if the condition are met.

Examples of the command:

pass lcvselect 10200,900,1

Other examples:

- If you do not want to send distance and RPM values via SMS, the SMS command should look as follows:
"pass lcvselect ,"
- If you want to send only RPM value, the SMS command should look as follows:
"pass lcvselect ,RPM"
- If you want to send only vehicle distance value, the SMS command should look as follows:
"pass lcvselect distance,"
- If you do not want to send distance and RPM values via SMS, but enable automatic IO selection, the SMS command should look as follows:
"pass lcvselect ,,1"

Responses

If SMS structure is correct and LCV auto select process started successfully, then the FM-device responds:

"LCV auto selection is in progress"

If SMS something went wrong and LCV auto select process cannot be started, the device responds with an error message. One of these error messages can be received:

- *"Unknown command"* - LCV auto select SMS structure is incorrect.
- *"Wrong CAN configuration"* - Current FM device configuration is incompatible with LCV auto select feature. Correct configuration options are described below in the "Configuration compatible with "LCV auto select" feature" section.
- *"LCV autoselection not started, try again"* - This message signals that "LCV auto select" should be restarted.
- *"Waiting for LCV autoselection start"* - Currently a critical process (i.e. CFG task) is running in the FM device. Within 2 minutes period device constantly checks process status. If it has concludes and there are no other

critical processes, then the auto select will start. User will be notified with the *“LCV auto selection is in progress”* message. If critical processes are still running, then the auto select will not be started. User will get the *“LCV autoselection not started, try again”* message.

- *“Parameters are out of ranges”* - Distance and/or RPM value received via SMS are out of accepted range.
- *“No CAN data were received”* - No CAN data packets were received.
- *“No valid matches were found”* - LCV auto select task concluded, but it failed to find matches in our database.
- *“Configured group: [Group]/[Subgroup], can1 ch:[number], can2 ch:[number], ERR: Not enough I/O slots”* - group is selected, but there are not enough free IO slots to enable all the auto-select IOs, IOs are not enabled.
- *“Configured group: [Group]/[Subgroup], Error: old I/O protocol is used”* - protocol v1.1 was not enabled prior to the SMS command.

Selection logic

“LCV auto select” feature is designed to choose the best group / sub-group combination for your specific vehicle. In order to do so, it employs a rating system, where each CANbus parameter has its own weight coefficient. Coefficient values within each sub-group are added together. This allows to compare different group / sub-group combinations and select only the best ones.

CANbus parameters themselves also undergo a verification procedure. In order to add weight to a specific sub-group, they have to have valid values. Valid value could mean that it has to be equal to something or it has to behave in a certain fashion, or it has to be in a valid range, etc. Rules for validation are defined internally.

Currently, “Fuel”, “Vehicle distance”, “RPM” and “Wheel based speed” parameters are assigned different coefficient values. Remaining CANbus parameters are rated lower.

After completing the selection process, the FM device will send a SMS message with the information about selected group and sub-group. For example:

“Configured group: Ford/Ford2”

LCV group and subgroup will be configured automatically.

“Configured group: [Group]/[Subgroup], can1 ch:[number], can2 ch:[number], I/O elements enabled ”

LCV group, subgroup and IO parameters will be configured automatically

Notes

- LCV Auto select functionality does not interfere with IO list configuration unless *enableIO* command is used.
- LCV Auto select does not guarantee that all CAN IO parameters supported for currently selected LCV Group and Sub-group will be available for this vehicle.
- *EnableIO* parameters are enabled in the active profile.

1.1.37 setlcv – set LCV group, subgroup and automatically enable possible IOs

SetLCV SMS command is used to set a specific LCV group, subgroup and additionally enable IOs which are assigned to this group, thus reducing the amount of work needed for configuring the FM device.

The structure of the command is:

pass setlcv <Group_Name>,<Sub_Group_number>,<CAN_interface_number>,<Silent/Active>,

<Channel_X>,<Additional_CAN_interface>

Excluding the command itself, only the <Group_name> ,<Sub_Group_number> and <CAN_interface_number> fields are mandatory. All other fields are optional. Default values for the optional parameters:

- <Silent/Active> = Silent;
- <Channel_X> = 1;
- <Additional_CAN_interface> - not active.

Notes

The SMS command is not case sensitive, so uppercase letters won't have influence when inputting the Group name. All LCV Group names and subgroup numbers are identical to those in the configurator. The only exception is the Caterpillar subgroup, which does not have a number in the configurator. In this case the <Group_name> would be "tractor" and <Sub_Group_number> - "2". (Example: pass setlcv tractor,2,1)

<pass> - the password for authentication. The maximum size of the parameter is 50 bytes.

<setlcv> - command to set lcv group, subgroup and other parameters to the ones specified.

<Group_name> - LCV group name, which the user wants to enabled (Example: vag).

<Sub_Group_number> - LCV subgroup number, which the user wants to enable. Note that only the **number** is used, not the whole subgroup name like in the configurator (Example: name in configurator is VAG17, then <Sub_Group_name> = 17)

<CAN_interface_number> - CAN interface on which the selected parameters will be configured. CAN1 = 1; CAN2 = 2.

<Silent/Active> - Set the enabled CAN interface to either Active or Silent. Active = 1, Silent = 0.

<Channel_X> - set the Channel for the CAN interfaces. The Channel setting is used when two separate physical connections must be made to receive all parameters from a LCV group/subgroup. Channel1 = 1, Channel 2 = 2.

<Additional_CAN_interface> - The user can specify whether to enable the other CAN interface. Activate = 1, not activate = 0.

Notes

Enable IO parameters function requires the use of Protocol v1.1, without it enabled, this function will not work. Additionally enough free IO slots must be available to accommodate the newly enabled IO parameters. The LCV group, and IOs will be set only on the currently active profile, overwriting any previous CAN parameters on that profile.

Examples of the command:

pass setlcv vag,17,1,0,2, 1 – VAG17 subgroup will be enabled on CAN1 interface, silent mode on CAN1 and CAN2, CAN1 will be enabled on Channel 2, CAN2 will also be enabled on Channel 1.

pass setlcv VAG,17,2 – VAG17 subgroup will be enabled on CAN2 interface, all other parameters are set to default settings.

Responses

If SMS structure is correct, Protocol v1.1. was enabled prior to sending the "setlcv" command, LCV group and subgroup is enabled, the response will be:

"Configured group: VAG17, I/O elements enabled"

If SMS something went wrong , the device responds with an error message. One of these error messages can be received:

- *"Configured group: VAG17, Error: Not enough I/O slots"* - the LCV subgroup was set, but there are not enough

free IO slots to enable all the IO parameters for this subgroup.

- “Configure group: VAG17, Error: old I/O protocol is used” - the LCV subgroup was set, but Protocol v1.1 is not enabled, thus some parameters cannot be enabled, so no parameters are enabled at all.
- “Error: unknown parameter” - Group name is misspelled, Subgroup number incorrect or invalid value was provided for one of the parameters.
- “Error: LCV auto select still in progress” - The device is currently in the progress of LCV auto select, during which the user cannot set a LCV group or enable IO parameters.
- “Wrong CAN configuration” - an incompatible LCV mode is configured on a CAN interface.
- “Configured group: Tractor2, Error: Not enough I/O slots” - If a LCV group contains more than 80 IO elements, it will not be enabled .
-

1.1.38 getlog/stoplog – get device log remotely

Parameter	Description
IP	32-bit number, commonly known as an Internet Protocol address (xxx.xxx.xxx.xxx).
Port	16-bit number, commonly known as the port number (xxxxx).
Mode	1 – send full log to server. All other values reserved for future functionalities.
Time	Time period for log sending. Defined in minutes. Range from 1 to 60 minutes.

SMS structure: *pass getlog <ip>,<port>,<mode>,<Time>*

Example: *password getlog 100.101.102.103,1111,1,15*

When device gets this command, it should open a link with defined server and send log according to defined mode. All other FM device processes (e.g. records sending, peripheral device communication) should keep working. Only device restart can stop log data sending process.

If data sending is disabled by configuration, log will not be sent and the device's response will be: "Data sending is disabled."

Log will be sent as ASCII. Log starts from device type [HCV/LCV/PRO], selected mode, defined time, bootloader version, firmware version, hardware version, GSM signal level and Voltage status: 0 –lower than 8 Volts (bad), 1 –higher than 8 Volts (OK) and device IMEI.

Getlog

This command initiates logging process.

Example: *FMTco4 HCV, 100.101.102.103:1111, mode=1, time=3min; version:472C,00.02.25.21,431,26,1; imei:868324027387005*

Responses:

- If getlog SMS format is correct, the device will respond: *getlog ok*
- If getlog SMS format is incorrect, the device will respond: *getlog data incorrect*
- If logging proces is already running, the device will respond: *getlog already running*

Stoplog

This SMS command stops log sending to the server.

Example: *pass stoplog*

Responses:

- When device gets this command and log sending is in progress, it should stop sending log to server and reply with SMS: *stoplog ok*
- If device gets this command and the log is not running, then the reply should be: *getlog already stopped*

Disclaimer: *In some cases remote debug might have to log data about many different events and processes. This would force remote debug to generate large internal data streams. Device might not be able to sent it all via GPRS. This might lead to data loss.*

1.1.39 info – get some specific information from the FM device via SMS

This customizable SMS command with configurable set of parameters can be used to get some specific information from the FM device via SMS.

Parameter	Description
time	GNSS date and time
coords	Current latitude and longitude separated by comma (,). Device will reply “GNSS data not available” if there will be no GNSS fix
alt	Current altitude
sat	Currently visible satellites
dir	Current angle
speed	Current speed, km/h
hdop	Current HDOP level
gps	Current GPS/GLONASS state: 1-off, 2-on no fix, 3-on got fix, 4-not responding, 5-sleep, 6-disabled.
eng	Current engine state: 0 – OFF, 1 – ON.
blver	Bootloader version
fwver	Firmware version
hwver	Hardware version
st	Date & time (GMT) from the last FM device rest/power ON.
gsmlvl	GSM signal level
op	GSM operator number
lac	Location Area Code
cid	GSM Cell ID
modi	The number of times when FM device has tried to initialize modem since start time.
modr	The number of modem resets since start time.
modsp	The number of times when modem was turned ON and turned OFF unsuccessfully.
gprs	Status of GPRS. There are two possible values: 0-no GPRS, 1-attached to GPRS.
gprso	The number of opened GPRS sessions.
gprsc	The number of closed GPRS sessions.
gprse	The number of GPRS errors.
lko	The number of opened links.
lke	The number of link errors.
lktmo	The number of server response timeout.
rs	Device last reset source. Possible causes: 04 - reset pin; 08 - power on/down reset; 10 - software reset; 20 - independent watchdog; 40 - window watchdog; 80 - Low power reset.

Parameter	Description
	14 means 10 (software reset) + 04 (reset pin) In other words FM device resets himself, for example after FW update.
p	GPRS protocol version: 0 – standard protocol; 1 – extended protocol.
imei	Device IMEI number
apn	APN name
apnu	APN username
apnp	APN password
ip1	IP1 address and port number
ip2	IP2 address and port number
prot	Currently set protocol: 0 - TCP or 1 - UDP.
modrev	Used to get modem revision information.
tacho	Current tachograph status information: 0 – Not available; 1 – everything is OK; 2 – tacho parameters not configured; 3 – extender not responding; 4 – tacho task in progress; 5 – physical connection OK, logical is not OK.
get3g	Answer according to “get3g” SMS command description
gmap	Hyperlink to Google maps with coordinates: latitude and longitude.
dout1	Is used to get status about Dout1, Dout2, Din1, Din2, Din3, Din4, Ain1 and Ain2. Values: 1 – high, 0 – low. Analog inputs – millivolts. Response will contain only values of requested parameters.
dout2	
din1	
din2	
din3	
din4	
ain1	
ain2	
pwr	
bat	Returns battery voltage in millivolts.
io[xxxxx]	Used to get specific IO current value. [xxxxx] contains IO ID number. Only activated in configuration IO data can be retrieved. For multi-part IO elements, first IO ID should be requested. IO parameters that are being reset with record sending should not be available, response would be NA.

Password, command and command text are separated by space symbol. If there is no password, then you need to write just space symbol before the command. Command text parameters are separated by semicolon (;) symbol. Response command text parameters are separated by semicolon (;) symbol. All content of SMS command is not case sensitive.

SMS structure: *pass info param1;param2;param3;[...];paramN*

Example 1: *pass info bat;alt;IO29;IO123;coords*

Response example: *4075;213.9;11659;0x4C54553132333435363738393031323334000000000000;54.7412916,25.2227750*

Example 2: *pass info time;imei;eng;IO29*

Answer: *2016-12-13 12:40:30;865789021340700;0;11691*

Maximum length of a response is 160 symbols. If response is longer, then the device will send maximum 157 characters of response plus “#” symbol and two digits number that shows, which tag was last in the reply counting from the beginning.

Example: *pass info gmap;coords;IO123;pwr;st;time*

Answer: *https://www.google.com/maps/?q=54.7413016,25.2227550;54.7413016,25.2227550;0x4C54553132333435363738393031323334000000000000;11659;2016-12-13 11:20:02;#05*

If some requested IO is not enabled in configuration or some tag is unknown, then the device will reply: *NA*

Example: *pass info time;imei;eng;IO29;lktma*

Answer: *2016-12-13 12:41:57;865789021340700;0;NA;NA*

1.1.40 USSD command

GSM network operators are providing various services via USSD communication. Support of the USSD commands allows users to send various USSD request to the GSM operator over the FM device and receive answers. In this case the device operates in the transparent channel principle.

Operation principles

- USSD command is received through the SMS and then passed to the modem;
- USSD command is not affected or ignored by ongoing firmware, configuration updates or tachometer and smart card reading.
- After getting the response, the device transmits the answer to the sender of the SMS;
- Timeout of 120 seconds is used when waiting for the response from the GSM operator.

SMS command

Request SMS structure: *pass ussd *command#*

Where command is a set of digits and asterisk symbols (*). Command always begins with an asterisk symbol * and is terminated by #. Inside, the command can have several asterisk symbols.

Correct command examples:

**485#*

**52*452*4#*

Incorrect command examples:

52# - missing *

**45* - missing #

**5 65#* - blank space in the command

Responses

- When the operator responds to the ussd command within 120 seconds, the answer is sent in the following format: *ussd response*
- If * or # symbols are missing or command is blank in the request message: *ussd bad format*
- If the response is not received from the operator within 120 seconds: *ussd timeout*
- If there is an error with the GSM network operator: *ussd modem error*

1.1.41 clear dtc

This SMS command allows to clear J1939 and J1708 DTC codes from the memory of the FM device.

SMS command

SMS command structure: *pass clear DTC*

Responses

- If command was performed successfully, response will be: *DTC codes were cleared*
- If DTC reading is not configured, response will be: *DTC reading is not configured*

1.1.42 doutreset

This SMS command allows to reset the state of activated DOUTs. This command works in correlation with the “DOUT activation with conditions” functionality, and requires it to be enabled and configured to work.

SMS command

SMS command structure: *pass doutreset*

Responses

- If command was performed successfully, response will be: *DOUT activation mechanism was reset*
- If DOUTs are not active at the time, response will be : *DOUT activation mechanism is inactive*
- If the “DOUT activation with conditions” functionality is not configured, response will be : *DOUT activation is not configured*

1.1.43 immobilizer

This SMS command allows to immobilize the vehicle. This functionality works in correlation with the “Immobilizer” functionality, described in the “DOUT activation with conditions” document. If the “Immobilizer” mode is not configured, the command will not work.

SMS command

SMS command structure: *pass immobilizer <speed>*

<speed> - describes the vehicle speed at which the progressive DOUTs must be toggled.

Responses

- If command was performed successfully, response will be: Immobilizer OK
- If no custom DOUTs are configured, the device will respond with the following message: *immobilizer is not configured*.
- If *<speed>* parameter was inputted incorrectly, only numbers from 1 to 250 can be sent, the response will be: *Invalid speed*.

SMS command to reset functionality

SMS command structure: *pass resetimmob*

Responses

- If the immobilizer is successfully reset, the response will be: *immobilizer reset*

1.1.44 setdinmode

This SMS command allows to set the DIN mode (Positive mode or Negative mode) for configurable DINs (DIN3 or DIN4)

SMS command

SMS command structure: *pass setdinmode <din#> <mode>*

<din#> - can have the following values:

- 3 – selects DIN3
- 4 - selects DIN4

<mode> - can have the following values:

- 0 – positive trigger inputs (positive mode);
- 1 – ground trigger inputs (negative mode);

SMS command example

pass setdinmode 3 1 – this SMS command will set DIN 3 to ground (negative) mode

Responses

- If DIN# was set to ground (negative) mode, response will be: *DIN mode set: DIN# - positive mode;*
- If DIN# was set to positive mode, response will be: *DIN mode set: DIN# - positive mode;*

If the device does not support configuration of DINs or wrong DIN mode is specified, the following responses can be received:

- *DIN mode set: configuration of DIN# is not supported;*
- *DIN mode set: incorrect DIN specified;*
- *DIN mode set: incorrect DIN mode specified;*
- *Configurable DIN functionality is not available for your HW version.*

1.1.45 getdinmode

This SMS command allows to receive information on the configuration of the specified DIN. This applies only to configurable DINs.

SMS command

SMS command structure: *pass getdinmode <DIN#>*

<DIN#> - can have the following values:

- 3 – information for DIN3
- 4 – information for DIN4

SMS command example

pass getdinmode 3 – This SMS command will provide information on the DIN3 configured mode.

Responses

- *DIN# - positive mode;*
- *DIN# - negative mode;*
- *Configurable DIN functionality is not available for your HW version.*

1.1.46 SMS forward

This SMS command allows to forward a SMS via the FM device.

SMS command

SMS command structure: *pass forward <phone_number> <text>*

<phone_number> - must not contain "0" or "+" in the beginning of the number (Not +370 or 00370, but 370);

<text> - Supports only GSM 03.38 characters.

Total length of whole command with text may not exceed 160 symbols.

Responses

- *SMS-Forward OK*
- *SMS-Forward Fail*

1.1.47 snapshot

This SMS command allows taking pictures on demand via connected peripherals with cameras.

SMS command

SMS command structure: *pass snapshot <peripheral number>*

<peripheral number> can have the following values:

- *0* – the command applies to all configured peripherals with cameras;
- *1* – the command applies to the RS232 camera configured on PortA;
- *2* – the command applies to the RS232 camera configured on PortB;
- *3* – the command applies to the fatigue sensor.

Response structure: *snapshot <peripheral number> <answer>* , *<other peripheral number> <answer>*

<answer> can be one of the following:

- *ok* – indicates that the peripheral took a picture successfully;
- *busy* – indicates that the peripheral is busy with a previously taken picture at the moment;
- *n/a* – indicates that the peripheral is either not configured on the given port, not connected or not responding;
- *full* – indicates that the memory of the SD card is full;
- *fullsd* – indicates that the memory of the SD card is full and the picture will be discarded, as there are no pictures to be overwritten;
- *err* – indicates that there was an error.

Example 1: *pass snapshot 1*

Response: *snapshot 1 ok*

Example 2: *pass snapshot 0*

Response: *snapshot 1 ok, 2 n/a*

1.1.48 ssl status

This SMS command allows receiving information about current SSL authentication status.

SMS command

SMS command structure: *pass ssl status*

Response structure: *SSL status server1 <status>, server2 <status>*

<status> can have the following values:

- 0 – SSL authentication is disabled on this server;
- 1 – SSL authentication is enabled on this server.

If OCSP validation is enabled, *<status>* can have additional values:

- 2 – the certificate is valid;
- 3 – OCSP lookup failed;
- 4 – the certificate is revoked;
- 5 – OCSP server URL not found;
- 6 – unknown certificate;
- 7 – validation request timeout;
- 8 – modem firmware does not support OCSP validation.

1.1.49 optiver

This SMS command is used to request the version of the OptiTemp device.

SMS command

SMS command structure: *pass optiver*

Response examples:

- *Version: TMS X2 V2.100*
- *Device does not respond* – the version could not be retrieved.

1.1.50 iqfinfo

This SMS command is used to obtain the values of various iQFreeze parameters.

SMS command

SMS command structure: *pass iqfinfo*

The response includes the following parameters:

- Refrigerator serial number
- Trailer serial number
- Device serial number
- Device FW version
- Device BT name
- Current time and date (UTC)
- Vehicle registration number

If an error occurred, the FM device will send the following response: *Communication error*

1.1.51 iqfset

This SMS command is used to manage refrigerator set-point temperature values and configuration.

SMS command

SMS command structure: *pass iqfset <setpoint>,<cycle mode>*

<setpoint> value range: from -99.9 to 99.9. A dot must be used as the delimiter symbol.

<cycle mode> can have the following values:

- 1 – the refrigerator is in continuous mode;
- 2 – the refrigerator is in start/stop mode.

If an incorrect symbol or value is included in the command, or no value is included, the FM device will send the following response: *Syntax error*

Any of the parameters can be omitted, only the provided parameters will be changed.

Examples:

pass iqfset ,2 – only the cycle mode will be changed;

pass iqfset 10.5, – only the set-point will be changed.

1.1.52 setlock

This SMS command is used for locking the vehicle. This SMS command uses the following structure:

pass setlock <lock> <ATH>

<lock> can have the following values:

- *lock* – locks the vehicle if the doors are closed
- *unlock* – unlocks the vehicle
- *mlock* – locks the vehicle regardless of the door state
- *skip* – does not modify the current lock status

<ATH> can have the following values:

- *ath-on* – blocks the engine
- *ath-off* – unblocks the engine
- *ath-skip* – does not modify the current engine blocking status

Response structure:

- *setlock was successful: <lock>-OK <ATH>-OK*
- *setlock was unsuccessful: <lock>-FAIL:<parameters> <ATH>-FAIL*

<parameters> can have the following values:

- *IGN* – ignition
- *ODW* – open door/window error
- *DFD* – door front driver
- *DFP* – door front passenger

- *DRD – door rear driver*
- *DRP – door rear passenger*
- *DHT – door hatch*

<ATH>-FAIL may occur when:

- the ignition is on
- the ignition state is unknown

Response examples:

- *setlock was successful: LOCK-OK ATH-ON-OK*
- *setlock was unsuccessful: LOCK-FAIL:IGN,DFD,DFP,DRD,DRP,DHT ATH-ON-FAIL*

1.1.53 Supported SMS commands table

	Eco4	Eco4S	Pro4	Tco4 LCV	Tco4 HCV	Plug4
accinfo	•	•	•	•	•	•
accreset	•	•	•	•	•	•
banned	•	•	•	•	•	•
caninfo			•	•	•	
cansinfo			•	•	•	
clear obd				•	•	•
connect	•	•	•	•	•	•
coords	•	•	•	•	•	•
delrecords	•	•	•	•	•	•
econnect	•	•	•	•	•	•
getapn	•	•	•	•	•	•
getcfg	•	•	•	•	•	•
getecu				•	•	•
getio	•	•	•	•	•	
getioparam	•	•	•	•	•	•
getsd			•	•	•	
gsminfo	•	•	•	•	•	•
imei	•	•	•	•	•	•
lastchange	•	•	•	•	•	•
modrev	•	•	•	•	•	•
optiver					•	
reset	•	•	•	•	•	•
setcfg	•	•	•	•	•	•
setconnection	•	•	•	•	•	•
setio	•	•	•	•	•	
setioparam	•	•	•	•	•	•
setiotime	•	•	•	•	•	
setvalue			•	•	•	•
switchip	•	•	•	•	•	•
tacho					•	
version	•	•	•	•	•	•
webcoords	•	•	•	•	•	•
set3g	•	•	•	•	•	
get3g	•	•	•	•	•	
dfota	•	•	•	•	•	
enginevolt						•
lcvselect				•		
setlcv				•	•	
getlog/stoplog	•	•	•	•	•	
info	•	•	•	•	•	•
USSD	•	•	•	•	•	
clear dtc					•	
doutreset	•	•	•	•	•	
immobilizer	•	•	•	•	•	
resetimmob	•	•	•	•	•	
setdinmode		•	•	•	•	
getdinmode		•	•	•	•	
forward	•	•	•	•	•	•
snapshot			•	•	•	
ssl status			•	•	•	
iqfinfo			•			
iqfset			•			
setlock				•		

1.2 Informational messages, alerts

1.2.1 SMS alerts with date & time

SMS message can be sent if one of the following SMS alerts is configured: **overspeeding, power supply disconnecting, DIN1, DIN2, DIN3, DIN4 (ignition)**. Purpose of this message is to inform when these events were triggered. Date & time is GMT. To receive SMS alerts, "SMS Alert Number" must be configured.

"SMS Alert Number" can be configured via FM configurator. In the main configurator window, locate the "Authorized numbers" section and click "Options. This will open the "Numbers window". The second input field in this window is the "SMS alert number", the user must enter a number which will receive the SMS alerts.

Notes

When inputting the number, the "+" symbol should be replaced with "00".

IO parameters must be properly configured to be able to send SMS alerts, this means one of the previously mentioned IO parameters must be enabled as shown:

- DIN1, DIN2, DIN3 or Ignition (DIN4) – "Event on" set to **On Change**;
- Power supply voltage - "Event on" set to **Hysteresis** and hysteresis must be set to **On Falling** ;
- GPS speed - "Event on" set to **Hysteresis** and hysteresis must be set to **On Rising**.

After the "Event on" parameter is properly set, a new checkbox will appear at the top of the IO properties section called "SMS alert". Put a tick in this checkbox to receive SMS alerts for the selected IO parameter.

Afterwards, when a IO is triggered, the device will send an SMS alert to the number previously specified in the "Authorized number".

Notes

If the device was out of the GSM coverage, the FM device will accumulate events that will be sent as soon as GSM connectivity is restored. The FM device can store up to 16 events. If the 16 event threshold is reached, the device will start to overwrite oldest events. The device will send a report of all events in **ONE** message (as many, that can fit in 160 characters). Multiple events in one SMS message will be delimited by a semicolon symbol ";".

Message format:

date1&time1-event1; date2&time2-event2; ...

Examples:

2015.05.01 16:24:01-device disconnected;

2015.06.02 22:05:16-ignition: ON; 2015.06.02 22:05:35-DIN1: OFF;

2015.06.14 08:30:45-overspeed;

Additionally the SMS alert function now supports custom SMS text. Custom text fields appear after the "SMS alert" checkbox is clicked. The maximum custom text length is 50 symbols, all previous rules apply.