Galileosky

User Guide
Galileosky 7x

firmware 22.7

Contents

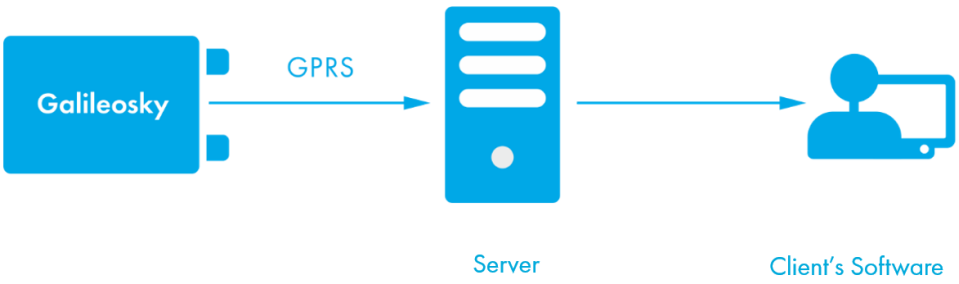
Introduction

1

RSA "Galileosky", LLC produces satellite monitoring equipment (hereinafter – tracking devices, trackers) for GPS and GLONASS real time vehicles monitoring. The tracking devices determine the mobile object location recording the time and route as points with geographical coordinates and send the data to the server to be further processed and sent to the traffic controller panel.

In addition, a number of other vehicle parameters are recorded: the state of analog and discrete inputs of the tracker and the state of digital interfaces.

The tracking devices can be used in any vehicle.



Galileosky Tracker receives a signal about its location from GPS, GLONASS satellites and processes input and output data

Information is sent to the server by means of GPRS and then via the Internet to the operator panel.

To prevent the data from disappearing when there is no GSM/3G signal, each Galileosky tracker has an internal nonvolatile FLASH memory. The tracker also has a built-in battery, which allows operation for 8 hours.

The tracking device provides the following opportunities:

- ✓ vehicles monitoring in real time;
- ✓ a detailed turn by turn track (without any extra points in a straight track);
- ✓ voice communication with the traffic dispatcher;
- ✓ GSM/3G enabled remote software update;
- ✓ continuous troubleshooting of the tracking device through the USB port;
- ✓ car alarm and a remote engine start;
- ✓ securing facilities against intrusion;
- ✓ automatic stops announcement;
- ✓ adjusting the tracker through SMS, GPRS, USB;
- ✓ and others (see sections Operation of the Tracker's Units and Connecting external peripheral).

The information sent by the tracker includes:
- ✓ the exact Greenwich time and date;
- ✓ vehicle coordinates: latitude, longitude, height;
- ✓ vehicles speed and direction;
- ✓ vehicle acceleration;
- ✓ inside temperature;
- ✓ inputs (buttons) and analog sensors state;
- ✓ external digital sensors state (fuel, temperature sensors and etc.);
- ✓ discrete outputs state;
- ✓ and others (see details of transmitted data in section Galileosky protocol data)

In addition, the company provides warranty service and technical support on its site.

Before starting the work, study the instruction carefully.

Package

The standard package includes Galileosky tracker, SIM-holder, a connector with contacts, a fuse and a fuse holder, passport, a certificate of equipping a vehicle with GLONASS/GPS devices. Everything extra should be purchased separately.

You will also need:

USB-cable	1
Power supply unit 9-39V (15 W)	1

Technical Specifications

Parameter	Modification*		
	Galileosky 7x C	Galileosky 7x	Galileosky 7x Plus
Analog-discrete and pulse frequency inputs	10	6	8
	voltage range -0-33V; maximum measured frequency-4 kHz custom pulling-up to +2,7 V		
Transistor outputs (output 0/1)	4 pcs.; maximum voltage 30V; maximum current 200mA		
Battery type	Li-Ion battery; 560mAh		
Average power consumption	0,48 W	0,54 W	0,54 W
ADC resolution in bits	12		
CANBUS	J1939, FMS, J1979, OBD II, 29-bit and 11-bit identifiers		
CAN inputs	1	2	1
RS485	1		
USB 2.0	mini-USB, tracker setting, troubleshooting and updating; charging enough for setting but not for GSM module; inner battery charging.		
Accelerometer	built-in		
GLONASS/GPS receiver	sensitivity – 164 dBm cold start 2+ s.; hot start 1 s.		
Coordinates determination accuracy, 95% of time, not worse	2,5 m (hitting not less than 50% of measurements in 24-hour period under condition of input signal level higher than 130 dBm and not less than 6 satellites)		
SIM-card type, pcs	nano-SIM, 2		
Opportunity of installing SIM-microchip	yes, instead of the second SIM-card		
GSM modem	GSM 850/900/1800/1900, GPRS class 12 UMTS Cat M1/NB1 deployed bands: 2, 3, 4, 5, 8, 12, 13, 20, 26*, 28* (* roaming bands)		
Wi-Fi	no		no
Bluetooth	no	Yes, 5.0	
Archive capacity	up to 170 000 points	up to 170 000 points; up to 2 500 000 points for each GB	

Parameter	Modification*		
	Galileosky 7x C	Galileosky 7x	Galileosky 7x Plus
		by using micro SD card.	
1-Wire	yes		
RS232	no	1	1
microSD	no	maximum memory of supported card is 32 GB (FAT32), 512GB (exFAT). It is recommended to use the following micro SD cards: Diamond MLC 4Gb (MT-DMM004GTME7-10W-10) Transcend SLC 2Gb (TS2GUSD230I)	
Speakerphone	no	a push-to-talk or a speaker and a microphone	
Speaker (Autoinformer)	no	built- in	
The number of geofences for voice prompts	no	limited by microSD capacity	
Speaker output type	no	Analog (linear output), 250mW	
Increase of functional capabilities	yes, by means of algorithms stored and run at the device with no interference into manufactory firmware code (Easy Logic)		
Protocol of data transmission	1. Galileosky: variable length-tag. 2. EGTS (GOST R 54619-2011, Order № 285 of the RF Ministry of Transportation) 3. EGTS (GOST 33472-2015)		
GLONASS/GPS aerial	external SMA/internal		
GSM	external SMA/internal		
Wi-Fi aerial	internal		

For 3G or LTE versions, we add 3G or LTE to the modification name, e.g. Galileosky 3G C or Galileosky LTE. If external aerials are used, there is an “ext” adding in the modification name, e.g. Galileosky 7x C ext or Galileosky 7x ext

Physical Specifications

Operating temperature range	-40...+85 °C
Storage temperature	-40...+85 °C
Relative humidity	0...90% (0...35 °C); 0...70% (35...55 °C)
Performance (height above the sea level)	0-2000 m
Storage	0-10000 m
Operating power supply	9-39V; is protected against voltage jumps in the vehicle power supply
Allowable voltage, continuously applied to power input, that does not let the tracker fail	-900...+200 V
Dimension	97,0 mm x 68,0 mm x 22,0 mm (without aerial connectors) 102,0 mm x 68,0 mm x 24,0 mm (with aerial connectors)
Weight	within 150 g
Body material	plastic
Average service life	10 years
Internal Li-Ion battery life	500 charge/discharge cycles, three years maximum

Safe Operating Rules

Before using the tracker, study the instructions of GSM/3G/GPRS devices safe operating.

Make sure the polarity is correct when connecting to the power supply.

The device should be connected straight to the vehicle battery, not to the vehicle power supply.

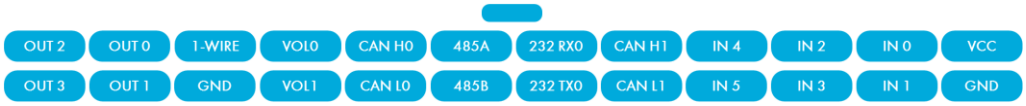
ATTENTION! To avoid failure:

- Make sure the contacts are connected correctly!
- Unused contacts must be well insulated!

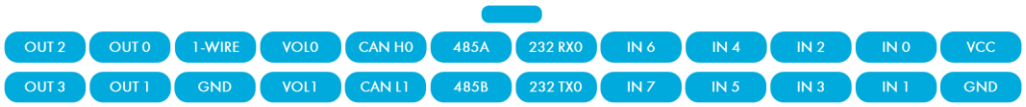
Contacts Description

Galileosky 7x C

Galileosky 7x



Galileosky 7x Plus



Contact	Description
VCC	Positive supply voltage
GND	Negative supply voltage
IN 0	Zero analog-discrete input
IN 1	1st analog-discrete input
IN 2	2nd analog-discrete input
IN 3	3rd analog-discrete input
IN 4	4th analog-discrete input
IN 5	5th analog-discrete input
IN 6	6th analog-discrete input
IN 7	7th analog-discrete input
IN 8	8th analog-discrete input
IN 9	9th analog-discrete input
RS485A	A signal of RS485 channel
RS485B	B signal of RS485 channel
232 RXD0	RXD signal of zero RS232 port
232 TXD0	TXD signal of zero RS232 port
CAN_H0	CAN_H contact of CAN_0 interface
CAN_L0	CAN_L contact of CAN_0 interface
CAN_H1	CAN_H contact of CAN_1 interface
CAN_L1	CAN_L contact of CAN_1 interface

Contact	Description
VOL 0	Zero contact to connect an external speaker for Autoinformer (all modifications, except C)
VOL 1	First contact to connect an external speaker for Autoinformer (all modifications, except C)
1-Wire	1-Wire interface (all modifications, except Lite)
GND	Ground to connect interfaces which need the ground contact
OUT 0	Zero transistor output (output 0/1)
OUT 1	First transistor output (output 0/1)
OUT 2	Second transistor output (output 0/1)
OUT 3	Third transistor output (output 0/1)

Connection

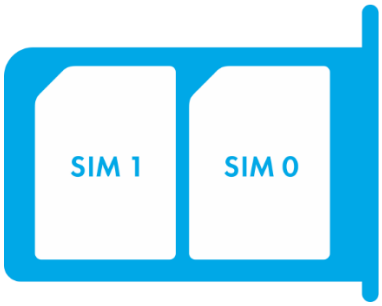
2

Inserting a SIM-card

Use a SIM-card with activated GPRS and SMS services.

Insert the card carefully without applying excessive force.

To eject a SIM-card press the hole in a SIM-holder with a needle, the holder will be ejected automatically.

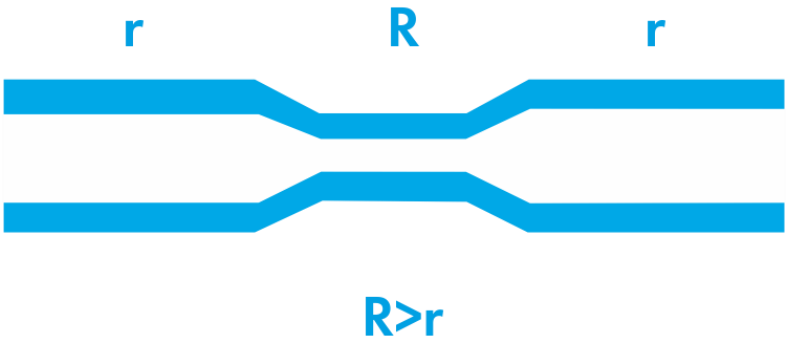


SIM-card location in the holder.

Connecting Power Supply to the Device

Positive supply voltage should be connected to VCC contact, negative supply voltage should be connected to GND (section Contacts description). If the connection is correct, the red LED will be on.

ATTENTION! Power-supply source needs to provide efficient current rate (more than 1,5 A) and handle impulse load as GSM-module in maximum load may require up to 2 A. Wires, used for power the tracking unit, must have regular diameter throughout the length, not less than 0,5 mm^2. They should not have any deformations like the one presented in the picture below.



When using such poor-quality wires, the strain parts R have higher resistance and consequently high heat release which can lead to having too low power supply resistance of the tracking unit and short circuit if the protecting wire cover melts.

LED indicators

For state indication one multi-coloured LED is used.

Red LED blinks when the power unit is connected to the tracking device.
Green LED shows the GLONASS/GPS unit status.

Blinking frequency, times	Description
3	GLONASS/GPS unit is not found or is at the initialization stage
2	GLONASS/GPS unit is found but correct coordinates are absent
1	GLONASS/GPS unit works properly, coordinates are found and updated once a second

Blue LED shows the state of GSM-module.

Blinking frequency, times	Description
4	GSM-unit is on
3	GSM-unit is initializing
2	GSM-unit is found, GPRS-session is activated
1	GSM-unit works properly, server is connected

Purple LED indicates Wi-Fi state.

Blinking frequency, times	Description
2	Wi-Fi-unit is not connected to the access point or there is no client's connection.
1	Wi-Fi-unit is connected to the access point and there is client's connection

Yellow LED indicates Bluetooth state.

Blinking frequency, times	Description
3	Waiting for IMEI
2	Bluetooth-unit is initializing
1	Bluetooth-unit works properly

Connecting to PC

In order to connect the tracking device to PC, use USB A – Mini-USB B.

3

Operation of the Tracker's Units

Serial Ports

To connect external sensors, the tracking device has 6 discrete-analog inputs which are pulse-frequency at the same time. Each input's function is set in the tracker's settings (sections Discrete-analog inputs setting and Inputs/outputs). In section Contacts description inputs are designated as IN0, IN1, IN2, IN3, IN4, IN5, IN6, IN7, IN8, IN9.

Each input saves its values to the nonvolatile memory, i.e. in case if the channel is set to be a pulse one, the pulse number value will be restored after resetting the device.

Feature	Value
Maximum measured voltage	33 V
Analog inputs resolution	1 mV
Maximum transmitted signal frequency	4 kHz

DAI have the following settings:

Parameter	Explanation
Filter type (input function)	0 - arithmetical average (discrete input state is also generated); 1 - pulse count; 2 - frequency input; 3 - Wiegand 26 Data0 (Data1)
Filter length to calculate the mean value	The greater this parameter, the more slowly the device responds to the input signal change. With filter length equal to 1 - averaging does not happen. Set this parameter to 1 for frequency inputs. It is necessary to set this parameter to 1 for pulse inputs. If the tracking device counts extra pulses, the filter length

Parameter	Explanation
	should be increased by one and accuracy should be estimated.
Ranges for response/ nonresponse areas (logical 1 and 0)	To process discrete signals, response/nonresponse range should be set where signals are equal to one and zero. Discrete inputs states should be seen in the Status of Inputs field, but not in the Input voltage field. While counting pulses or frequency, it is necessary to put the value equal to half the pulse value into all the fields of the given group (example: the pulses' amplitude is 5000 mV, so all the fields must take the value 2500 mV)
Null pulses after point record	When it isn`t ticked, pulses are counted cumulatively up to 65535 with subsequent nulling. The presence of a tick resets the pulse counter to 0 after recording each point according to the settings for writing points to the tracker`s memory and the parameters for track drawing.
Pulling-up to +	When it is ticked, the signal can be pulled up to +2,7 V, which is needed for some sensors.

Pulse Count

In case of a renewable counter the maximum pulse number can be 65535, after that the number is reset to zero.

If there is pulse at input, the correspondent bit will be set in the “Status of Inputs” field, and a point will be recorded. If there is no other pulse for 30 seconds, the bit returns to 0.

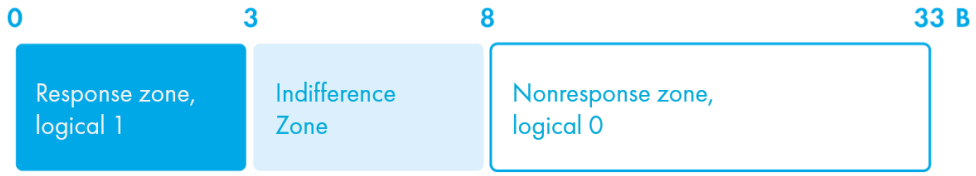
Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction DAI. Connection of pulse flow meter

Mean Value and Discrete Event Generation

Example 1.

Let us consider the example with the following zero input setting (see the left-hand figure:
Filter type is 0;
Filter length is 5;
Logical one zone range is 8-33V;
Logical zero zone range is 0-3V.

Example1



Example 2

The mean value is calculated continuously and is put into the corresponding field IN0.

At the same time, it is continuously checked whether the calculated value belongs to the given range.

If it is in the range 8-33V, the corresponding bit will find itself in the Status of Inputs field and a point will be recorded.

By value coming into the indifference zone (3V-8V), the former bit value will be saved to the Status of Inputs field.

If the value is in the logical zero zone (0V-3V), the corresponding bit in the Status of Inputs field is reset.

Thus, we can see that the given bit can change its state only in the logical one/logical zero zones.

Example 2.

In contrast to example 1 (see the right-hand figure) the logical one zone and the logical zero zone have changed their places.

In the same way it is possible to set zones of response and non-response for the tracker`s power supply voltage (POWINCFG command, section Analog-discrete inputs setting). The state of discrete signal for input can be identified according to the 8th bit of the "Status of the device" field.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction DAI. Discrete analog sensors connection

Frequency Count

To measure frequency in some sensors it is necessary to connect the sensor frequency output to the sensor positive power supply via a 1kOhm resistor. Otherwise, frequency count is impossible. It is done by ticking the parameter "Pulling-up to '+'"

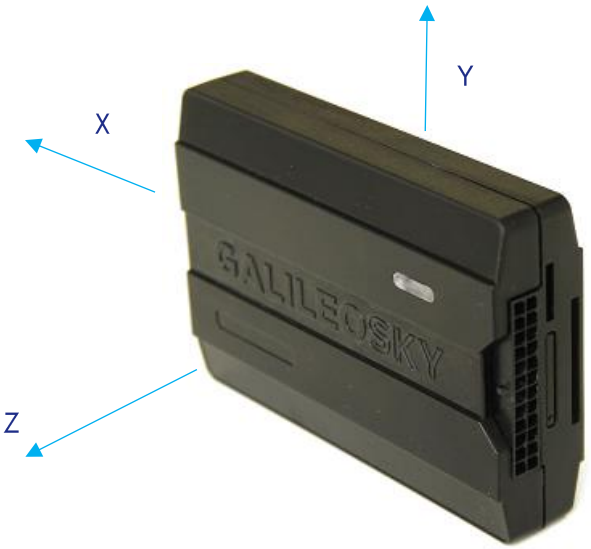
Wiegand26

This filter is used when connection up to two RFID-card readers to inputs IN 0 and IN 1, IN 2 and IN 3 using Wiegand protocol with the output pulse length not less than 250 microseconds.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction DAI. Connection of a keyboard with RFID-reader through Wiegand-26

Determination of Strike and Incline

All devices can determine the tracker`s strike and incline.



Accelerometer axis directions:

To determine strike:

- Install the tracker so as one of the accelerometer axis looks vertically, it will exclude false detections on road bumps;
- Turn on strike and incline determination by SHOCK command (section Track parameters setting). For example, if Z axis is vertical: SHOCK 3,90,5,1200, where: 3 – strike detection is activated, axis Z is vertical; 90 – maximum allowable incline angle; 5 – allowed time of incline angle excess 5 sec; 1200 – allowed acceleration

of strike (conditional units, where 600 = acceleration 1g); 2 – allowed duration of strike (standart units – approximately 1/400 second).

A strike is an acceleration increase in horizontal plane; the correspondent bit is put in the Device state field and strike coordinates are recorded.

To determine incline:

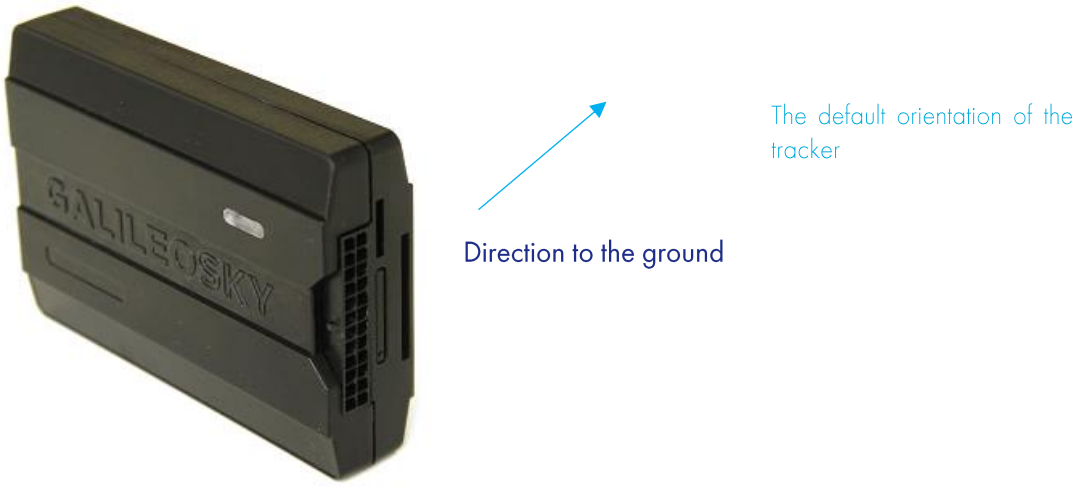
1. Install the tracking device in a vehicle;
2. By SHOCK command set maximum allowable incline angle and allowable time of this angle exceeding. For example, a maximum angle is 20º, allowable exceed time is 5 seconds; SHOCK 3,20,5,1200.

On the tracker homing position change in a vehicle, SHOCK command should be given to adopt the tracker to a new position.

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction
Accelerometer. Determination of strike and incline

"EcoDriving" Function and Determination of the Driving Style

The tracking device can detect rapid acceleration, braking, harsh turns and strikes on bumps. For correct operation of this function, the tracker must detect its orientation in space with respect to the vehicle (the direction to the ground).



If the tracking device cannot be installed as illustrated in the picture, user-defined installation may be performed with the subsequent calibration of orientation.

To determine the position of the tracking device in respect to the vehicle, perform the following steps:

1. install the tracker to ensure its rigid link with the vehicle's body;
2. ensure the horizontal position of the vehicle;
3. execute the shock 0 command, which determines the direction of the tracker to the ground.

4. To calibrate the accelerometer, the vehicle needs to move for some time. The time of calibration depends on the intensity of accelerations/braking/turnings. The fact of calibration ending is the appearance of acceleration values on the server and on Troubleshooting tab in "Configurator".

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction Service. Determination of Driving Style

Data Archiving to the MicroSD Card

Can be applied only for modifications equipped with a microSD slot.

To create a backup on the microSD card it is necessary to insert it into the tracking device. The saved data will be ordered in the following way:

X:\Arc\
1.gsa
2.gsa
...
10.gsa
...
, where X is a disc name of microSD.

If there isn't enough space on the microSD card, the tracking device tries to delete the oldest files from the Arc folder. If it is needed, data for a specified time period can be uploaded from the archive to the server by command EFS DDMMYY, DDMMYY2. See detailed information in EFS command description.

Autoinformer Function

Can be applied only for modifications equipped with a microSD slot and autoinformer output.

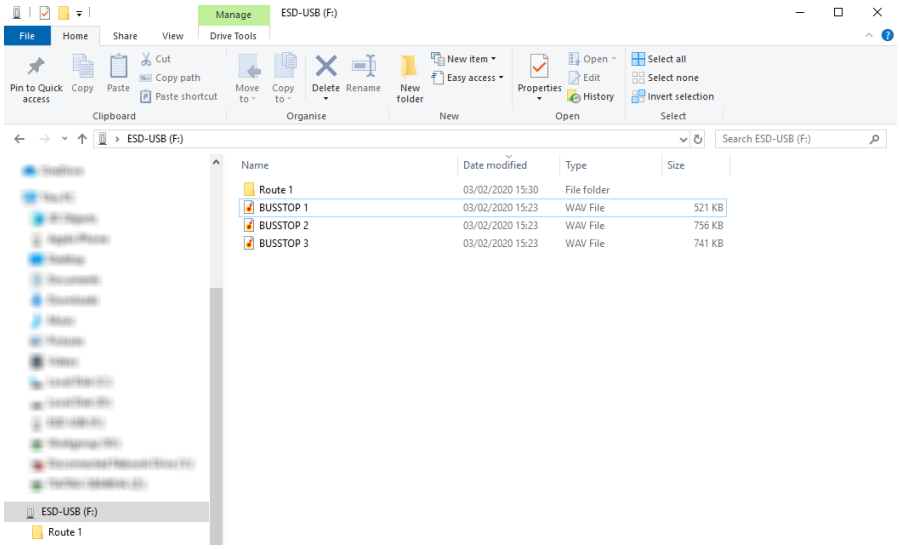
Autoinformer function may be used for automatic (without participation of the driver) public transport stops announcement with the use of a satellite navigation system.

The main difference from analogous systems is taking into account vehicle movement direction, thereby excluding false operation at other stops located in the same geographic area.

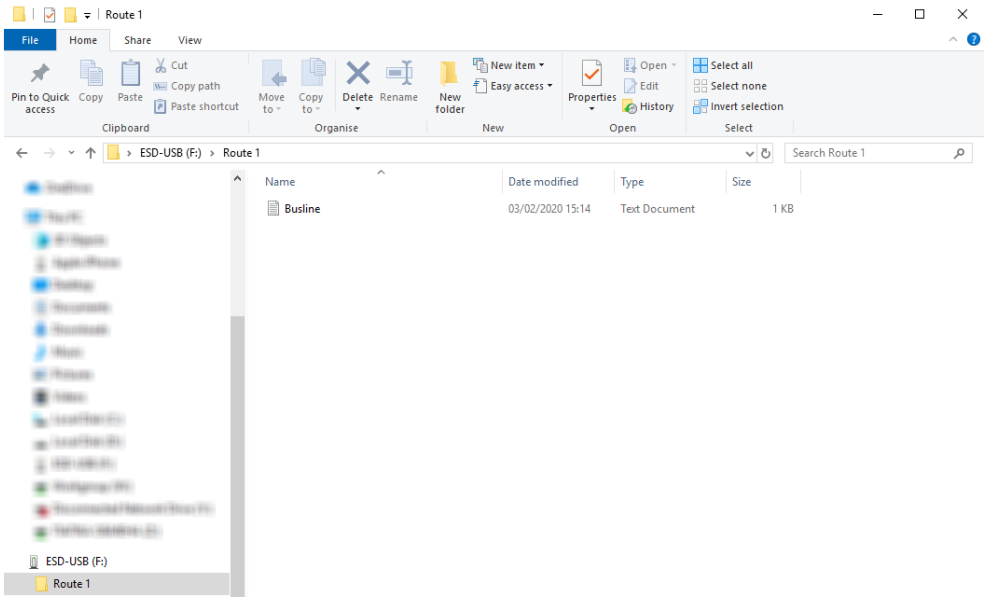
To use autoinformer:

1. Attach the speaker to the tracker (section Connecting autoinformer speaker).
2. Set the microSD card:

   a. Place sound files in format: wav, 16 kHz, mono, 16 bit to the card root folder. The file name must not exceed 20 symbols, including the extension, for example, PARKOVIJ.wav. The record length is recommended within 4 minutes (in case of exceeding, by the following file reproduction crackle may appear);
   b. Create folders with routes names in the card root folder. The smallest number of routes is 1.



   c. It is necessary to place the BusLine.txt file to the route folders, where response areas and areas linkage to the sound files are stored.



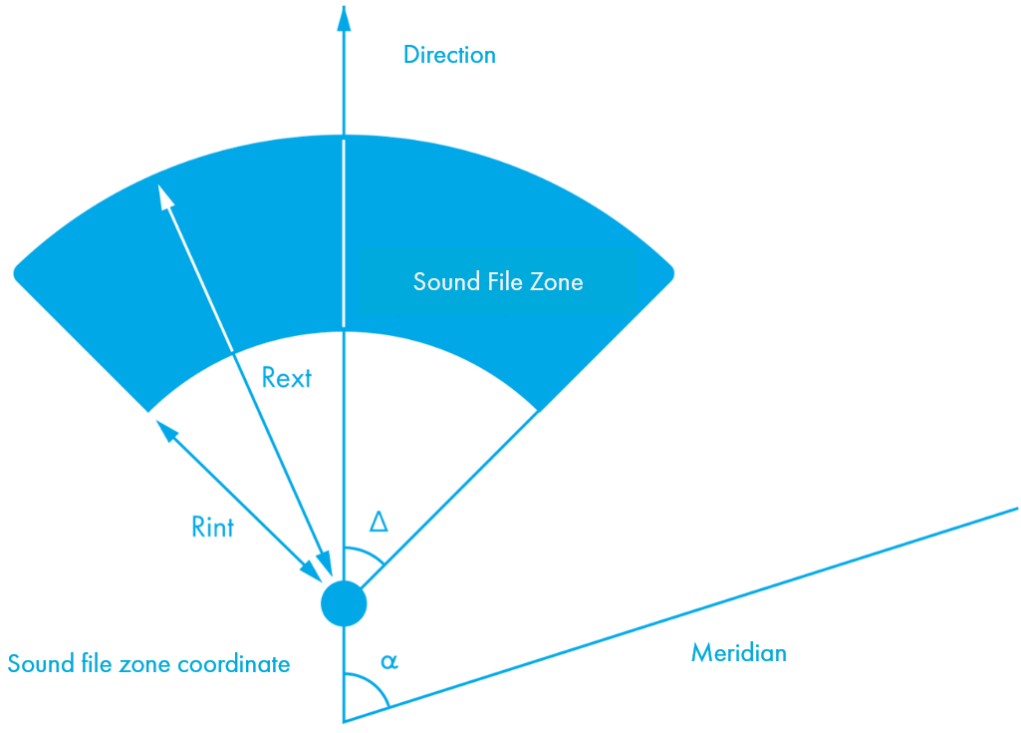
The format of each zone is the following:

- Latitude;
- Longitude;
- Direction angle α(the angle between the meridian and a vehicle direction);
- Spread for the direction angle Δ (see the diagram below);
- Outer radius of Rext response zone;
- Inner radius of Rint response zone;
- Sound file name corresponding to this zone;
- Bus stop name corresponding to this zone.

It is convenient to fill in the information for zones from the Device tab of the Configurator while going along the route.

By route forming it is necessary to indicate separate zones for stops in both directions even if the stops are opposite each other.

Latitude and longitude values are entered by means of the point "." (For example: 57.9842) where the value after the point – is degree fractions. To transfer minutes into degree fractions (Xdegr.Ymin.) use the following expression Xdegr. = Ymin./60. For example: 57 degr. 55.4513min = 57.924188 degr.



Explanatory diagram

3. Activate the Autoinformer function with the Autoinformer command (section Autoinformer setting).
4. Insert a microSD card into the tracking device and reset it with the Reset command. After the tracker's resetting, the function will be activated.

In the process of sound files playback there is a 5 second pause between adjacent files.

To test sound files:

1) Unscrew GLONASS aerial from the tracking device;
2) Enter into file BusLine.txt the following lines:

[the following format: LAT,LON,ANGL,DELTA,RAD_EXT,RAD_INT,STRING_STATION]
0.0;0.0;12.0;180;500;0;TEST.wav

3) Create *TEST.wav*. file in the microSD card root. This file will be played again and again after the tracker's resetting.

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction Audio. Autoinformer function setting

Monitoring Data Transmission

The tracking device allows specifying the list of preferred GSM networks. The main priority is given to the networks from the beginning of the list. Each network is specified with country's code and network operator's code. Tracking device supports up to 30 networks (OPS0 command, section Data transmission settings). If it is impossible to connect to one of the preferred networks, the tracker connects to any network but does not establish connection to the server, thus, voice communication and SMS will be available according to a service plan of the installed SIM-card.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction General Information. Data transmission settings for Galileosky 7.0 and Base Block tracking devices

The tracker allows data transmission to the main and backup monitoring servers. The tracking device accounts transmitted data separately for each server, thus, both will receive full archive with the track.

Data can be transmitted according to the Galileosky or EGTS protocol (Protocol command, section Data transmission settings).

By using the EGTS protocol the tracker`s number parameter (ID command, section Data transmission settings) specifies the object number by the authentification.

By using the Galileosky protocol transmitted data can be coded; XTEA3 algorithm (Service commands) is used for coding. Commands, responses and photos are not coded.

The data are archived in the internal flash-memory by default. During long periods without connection the oldest records of the internal flash-memory may be erased by the new ones. In this case, it is recommended to insert a microSD card for HARD 3G/HARD WI_FI modifications and to adjust archive transmission from it (Archive command, section Service commands).

Internal Archive Structure

The data archive can be stored on the internal flash memory or a micro-SD card. The internal flash memory card is used by default.

The tracker stores the data from all the inputs and interfaces into the archive on the internal flash memory which were marked in the Main pack on the Protocol tab in 'Configurator'.

By using the internal flash memory, it is possible to choose the order in which points are sent to the server. By default, the data are saved in the depth of the data store, i.e. current data are saved before older data. Transfer in chronological order can be set by FLASHARCHIVE command. After changing the direction of memorizing data, the flash memory will be formatted, and all previously stored data will be lost.

By using a micro-SD card, the data are always sent in chronological order. Note that only current data are used for the first packet.

Note that current data is taken for the main pack.

Operation with SIM-microchip

The tracking device has a slot for a SIM-card and a place for SIM-microchip sealing. Only SIM-card or SIM-microchip can be active and support the registration in GSM/3G-network at the same time. When the SIM-chip is installed, the second slot for SIM-cards (SIM1) becomes inactive. APN can be set for SIM-card and SIM-microchip. The tracker supports the following algorithm of SIM-card operation:

1. Only SIM0 card is always active.
2. Automatic switching to SIM-microchip or SIM1, if the data cannot be sent to the server within 9 minutes. Switching occurs in cycles, i.e. first SIM0 is used, then SIM1 or SIM-microchip and after that SIM0 again.
3. Switching between the SIM-cards and SIM-microchip according to the list of preferred GSM/3G-networks. If the tracking device detects the availability of one of the specified GSM/3G-networks, it switches to the corresponding SIM-card or SIM-microchip. If both networks specified for a SIM-card and SIM-microchip are available at the same time, the preference will be given to SIM0.
4. Always only SIM1 or SIM-microchip is activated.
5. Periodic switching (according to set value from 120 seconds and more) between SIM-cards and SIM-microchip for sending data to different servers.

The second algorithm is always used for remote firmware updating; the tracker attempts to get the connection to the server with firmware via SIM 0, and if it fails – via SIM1 or SIM-microchip.

GPRS Traffic Costs Optimization

GPRS-traffic costs decrease by online monitoring may be reached by following these steps:

1. Turn off the transmission of unused data, for example, temperature, acceleration, analog and serial ports values which have no connected sensors. It can be made in the Configurator on Settings/Protocol tab or by MainPack and HeadPack commands (section Server exchange protocol settings).
2. Increase points record period. It can be made in the Configurator on Settings/Track tab or by WrPeriod command (section Track parameters settings).
3. Increase turning angle at which the device records a point, and distance at exceed of which the point is recorded. It can be made in the Configurator on Settings/Tracks tab or by Turning command (section Track parameters settings).
4. Find out the time of disconnection because of the tracker inactiveness from the server software developers. This parameter should be taken into account by points' record period setting otherwise the traffic will increase because of costs for restoring connection to the server. Example: points' record period at a stop is 1200 seconds (20 minutes), the server disconnection because of the tracker inactiveness is 180 seconds (3 minutes). The tracking device determines that a vehicle has stopped and switches on a timer for the next point record in 20 minutes, in 3 minutes the server

disconnects as it hasn't received the data from the tracking device. The tracking device tries to reconnect the server at once. It happens 6 times, and only in 20 minutes the tracker sends the next point. As a result, traffic costs considerably exceed savings from points record interval increase.

5. Set filtering of coordinates at a stop so as the tracking device can correctly choose points' record period. The tracker can determine a stop according to several elements:

- accelerometer data (AccSens command, section Track parameters setting);
- external supply voltage (MHours command, section Track parameters setting);
- ignition sensor indications (Ignition command, section Track parameters setting).

Remote Configuration

Remote configuration can be performed through several data transfer channels:

1. SMS. The tracking device has a list of 4 authorized phone numbers, the messages from which are treated as configuration commands. The available commands are described in the section Settings for SMS control. A phone number can be added to the list of authorized numbers either through the Configurator, or by sending a message with AddPhone command (section Settings for SMS control).
2. GPRS. Commands can be sent from the monitoring data processing server. The format of the commands is described in the section Galileosky protocol data.
3. GPRS. Commands can be sent via the Configurator and the remote configuration server of RSA "Galileosky", LLC. In this case, the tracking device supports two parallel connections: the first – with the monitoring data processing server, and the second – with the remote configuration server. Remote configuration can be enabled using RemoteConfig 1 command (section Service commands). It is possible to send commands to the tracking device, to receive current information from the sensors connected and to receive diagnostic messages, when working with the remote configuration server. Using the Configurator, it is possible to create a command pack to configure the tracker and to save it on the server. These commands will be sent to the tracker when it establishes the connection to the server.

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction Service. Remote configuration of Galileosky tracking devices

Function of Signaling System

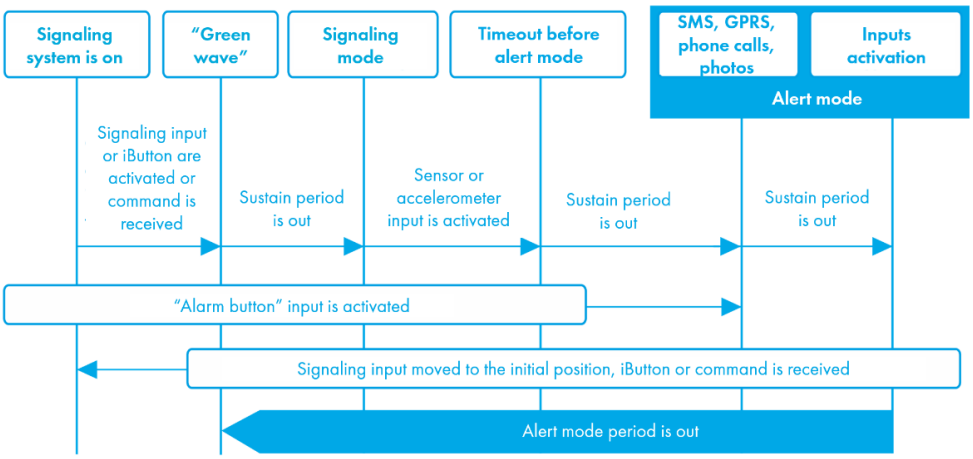
Function of signaling system allows to set reaction to:

1. Change of analog inputs state;
2. Strikes and inclines (accelerometer data);
3. Position change;
4. Overspeeding;
5. Connection of iButton or RFID-card.

Tracking units can react by inverting inputs state, impulsing outputs, SMS sending, making phone calls to the set phone numbers, photo making and coordinate recording.

Settings that can be changed by a user:

1. Time period after signaling system is launched during which no input signal will be processed ("Green wave");
2. Maximum time in alert mode after which the tracking unit will automatically move to signaling mode;
3. Individual for each input time period between activation and moving to alert mode;
4. Individual for each input text message which is sent via SMS when moving to alert mode;
5. Individual for each input time period between moving to alert mode and state changing.
6. Activation and deactivation of signaling system can be carried out by inputs, SMS or server command or iButton key, which should be set beforehand in the Configurator software (command iButton, paragraph Serial ports settings). Commands have more priority than inputs state. Activation on inputs depends on the settings made by InCfg command, inverting level of inputs (see Analog-discrete inputs setting), which is set by Out command (see Transistor outputs setting)



Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instructions Signaling. Signaling settings to operate with analog-discrete sensors, Signaling. Signaling setting to operate with iButton keys or RFID-cards, Signaling. Speeding and (or) exceeding set radius, Signaling. Connection of Emergency button

4

Connecting External Peripheral

CAN-interface

The tracker allows extracting information from the CAN-bus.

The following protocols are supported:

– J1939 (FMS). According to this protocol, the tracking device is not a device transmitting to CAN-bus, the device does not change vehicle operation, it also does not send confirmations to vehicle units packets and there is no electrical noise in the CAN-bus. In some cases, by connection to the troubleshooting socket for correct reading of information from the bus it is necessary to send confirmations to vehicle units packets, for this give ActiveCAN 1 command to the tracker (section CAN settings).

– J1979 (OBD II). This protocol works according to the question-answer mode, consequently, the tracker issues requests to the CAN-bus.

Available performance modes:

SCANNER – the bus scanner sending bus reports to the configurator.

FMS – a standard FMS protocol filter. (see www.bus-fms-standard.com).

USER – programmable custom filter.

J1939_ 29bit – a configurable user filter. Identifier length is 29 bits.

J1939_11bit – a configurable user filter. Identifier length is 11 bits.

Easy Logic – setting the working modes with CAN bus using Easy Logic algorithms.

J1979_CHECK_PIDS – search of 11 and 29-bit identifiers J1979 protocol responded to requests.

SCANNER Mode

This mode is intended to study CAN messages, sent in CAN bus.

Speeds from 500000 bit/s up to 1000000 bit/s (typical values: 50000, 83333, 125000, 2500000, 5000000, 1000000) are supported.

11 and 29 bit identifiers are supported.

Scanner mode begins with preliminary speed setting and pushing the button 'Start receiving'

CAN message start to appear in the order of ascending values with a delay shown in 'Timeout' parameter.

29bit identifiers are displayed in the following format:
ID= 00000009 06 07 08 09 00 CC DD EE
Where:
ID - is a 29bit message identifier;
06 07 08 09 00 CC DD FF - is an 8byte message. (The lower byte is on the left, the higher byte is on the right; the bytes are numbered from 1 to 8),

11bit identifiers are displayed as

ID=009 06 07 08 09 00 CC DD EE
Where:
ID - is an 11bit message identifier;
06 07 08 09 00 CC DD EE is an 8byte message. (The lower byte on the left, the higher byte on the right; the bytes are numbered from 1 to 8).

To enable this mode, you need:

1)connect the tracker to the vehicle CAN-interface;
2) in the Configurator on Settings/CAN tab select bus rate and delay time (timeout);
3) click Start Scanning. Received data are displayed below.

FMS Mode

This mode is activated in all tracking devices by default; it allows retrieving and parsing messages relevant to FMS protocol:

- total fuel consumption: the amount of fuel the vehicle had used since it was made;
- tank fuel level: measured in percent. 0%-empty, 100%- full;
- coolant temperature;
- engine speed;
- total mileage;
- operating hours;
- axis load.

ATTENTION! Many car manufacturers support FMS protocol partially or do not support it at all.

To enable this mode:

1) connect the tracker to the vehicle's CAN-interface;
2) give the CanRegime 2,250000,2000 command (section CAN settings) or select FMS filter type in the Configurator on Settings/CAN tab;
3) make sure that the device receives bus data and sends them to Device tab in the Configurator;
4) set data transmission to the server using the MainPack command (section Server exchange protocol settings) or in the Configurator on Settings/Protocol tab.

USER Mode

This mode enables us to receive messages from the vehicle CAN bus, according to J1939 protocol.

To enable this mode:

1) connect the tracking device to the vehicle's CAN-interface;
2) select Custom filter (29bit identifiers) type in the Configurator on Settings/CAN tab, set the bus rate and delay time or give CanRegime command with necessary parameters (section CAN settings);
3) set filters for CAN-bus.
4) set sending of received data to the server with the help of MainPack command (section Server exchange protocol settings) or in the Configurator on Settings/Protocol tab.

Notes:

1) In protocol of the first and the main packet of the tracker there are 1-byte, 2-bytes and 4-bytes tags for this mode operation, i.e. if the necessary ID needs only one byte from all data, better choose 1-byte tag.
2) Any of these tags can correspond to the right CAN message ID

ATTENTION! The data should be recorded in the decimal system in the tracking device. The hexadecimal notation is used for convenience only. From the useful data obtained by this ID, using the mouse mark, you can select exactly that part of the bytes that should be filled in the contents of the tag.

Let us consider an example:

The CAN message identifier is ID=0x0CF00400.

We need only the 4th and 5th bytes of all the sent content with this ID.

As we need only two bytes, we shall choose the tag CAN16BITR0 as an example.

The command to set the tag is as follows: CAN8BITR0 ID,Shift, Format.

1) The tag number ID=Ox18FEEE00 will look as 217056256 in the decimal system.
2) The byte we need is shifted by three bytes, i.e. the second parameter is equal 3.
3) The third parameter determines how the value will be determined in decimal when converting from hexadecimal - in the order of reading bytes from right to left (little endian format) or from left to right (big endian format)

So, we have the following command for filter settings: CAN16BITR0 217056256,3,0.

Now when this message is passing through the bus, the first effective load byte will automatically be placed to the tag R0 and sent to the server.

These settings are easier to make in the Configurator:

1. Scan the bus;
2. Indicate identifier in the first column;
3. Select the correspondent tag;
4. Visually indicate the shift using a mouse. The number, transmitted to the server, will be displayed in the Value column.

J1979_29bit Mode

This mode allows extracting and decoding the messages with 29 bit identifiers, transferred according to J1979 protocol automatically:

- tank fuel level: measured in percent. 0%-empty, 100%- full;
- coolant temperature;
- engine speed;
- errors codes.

ATTENTION! Many car manufacturers support J1979 partially or do not support it at all.

To enable this mode:

1) connect the tracker to the vehicle's CAN-interface;
2) give the CanRegime command (section CAN settings) or select OBD II 29bit filter type in the Configurator on Settings/CAN tab;
3) make sure the device receives bus data and sends them to Device tab in the Configurator;
4) set the received data transmission to the server using the MainPack command (section Server exchange protocol settings) or in the Configurator on Settings/Protocol tab.

J1979_11bit mode is set in a similar way.

ATTENTION! If your vehicle doesn't support J1939 protocol, J1979_29bit and J1979_11bit modes operation can cause failures of board equipment operation. RSA "Galileosky", LLC bears no responsibility for failures after activation of these modes.

Easy Logic Mode

This mode allows to receive messages from CAN bus of the vehicle using Easy Logic algorithms, in which the processing of all the required data from the bus is configured.

J1979_CHECK_PIDS Mode

The mode allows to search for 11 bit and 29bit identifiers of J1979 protocol automatically. The tracking device sends requests to identifiers in the range 1...255, first to 11bit, then to 29bit in the same range. Identifiers that respond, will be displayed in diagnostics "CAN detailed". After scanning, CAN will be transferred to "CAN disabled" mode.

The order of procedures in this mode:

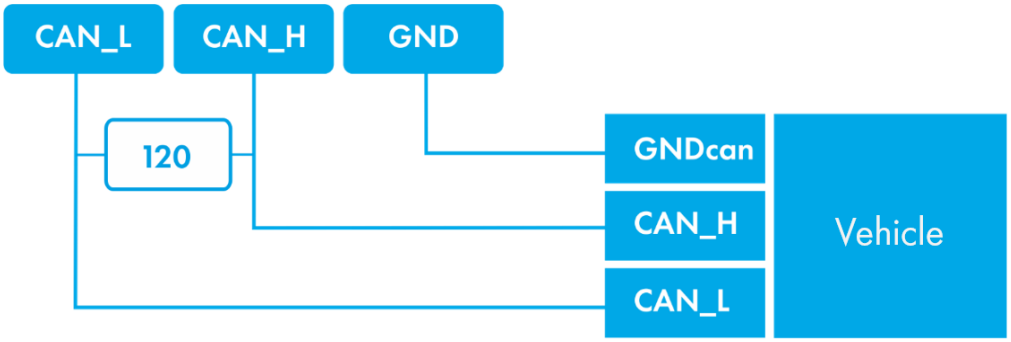
1) Connect the tracker to CAN-interface of a vehicle.
2) Send CanRegime command (CAN settings) with parameter Mode=8. In 5 seconds, the tracking device will start searching for identifiers.
3) Go to tab "Troubleshooting" and tick "CAN detailed" parameter to display the process of searching. In case the parameter is not ticked, or you go to another tab while scanning, data will not be saved.
4) «CAN mode: J1979. Searching PIDs...» message means that searching has started.
5) «CAN mode: J1979. Checking 11bit PIDs ...» message displays the results of search for 11bit identifiers. The tracking device will report on the current scanning process. In case of response by one of the identifiers, the message will be displayed.
6) «CAN mode: J1979. Checking 29bit PIDs ...» message displays the results of search for 29bit identifiers. The process is similar to the previous point.
7) «CAN mode: J1979. Search is finished» message means that the search is completed. CAN will automatically be transferred to the mode "CAN disabled".

ATTENTION! Many manufacturers of vehicles partially support J1979 protocol or do not support it at all.

ATTENTION! Activation of the mode in the vehicles with no support of J1979 protocol may lead to a failure in operation of onboard equipment. RSA "Galileosky", LLC bears no responsibility for failures after activation of these modes.

Different options of connection to CAN-bus

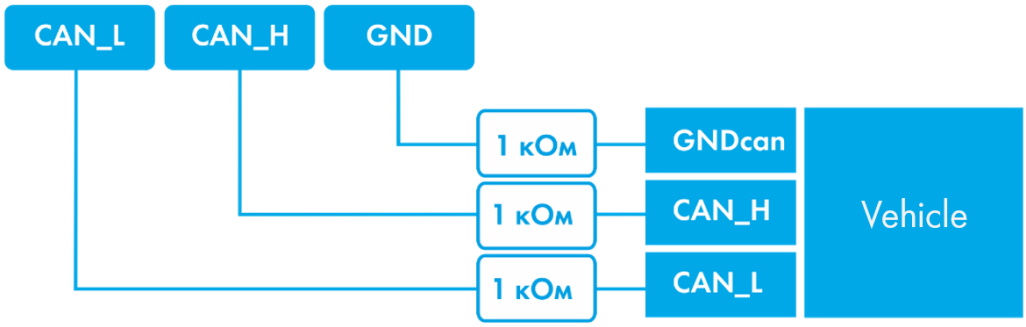
1. Direct connection.



ATTENTION! If the terminating resistor (is shown with a dotted line in the diagram) is not installed on the vehicle side, it should be installed. Its presence can be checked with the help of a multimeter: it is necessary to measure the resistance between CAN_H and CAN_L when the vehicle electronics is off. If the resistance is about 60 Ohm, there is no need for a terminating resistor. If the resistance is 120 Ohm, it is necessary to connect a standard 120 Ohm resistor between the CAN_H and CAN_L wires.

To plug the tracker into the troubleshooting socket it is necessary to use the first connection variant. The second variant is recommended only for use to connect the tracker directly to the vehicle's CAN-bus.

2. Connection with current-limiting resistors



ATTENTIONT! There are some versions of the trackers, where two CAN interface ports are used. The above connection options can be used when connecting to any of the ports.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instructions CAN-bus. Connection to the CAN-bus, CAN. User Manual for work with CAN scanner, User Manual for work with CAN-logger.

Connecting digital fuel sensors using RS232 protocol

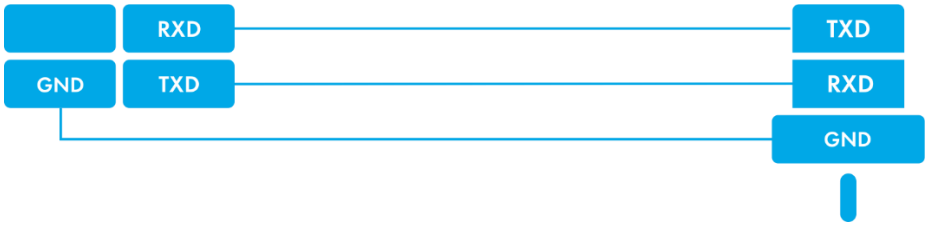
Can be applied only for modifications equipped with RS232 ports.

The order of connection:

1. Connect RXD, TXD and GND sensor contacts to tracker TXD, RTD and GND contacts (section Contacts description).

ATTENTION! The tracker and sensor grounds should be connected!

The sensor power supply is provided separately.



2. Set an option to receive relative fuel level or frequency from the sensor for the RS232 tracker channel. It can be done by sending RS2320 command (section Serial ports settings) or through the Configurator on Settings/Serial ports tab. RS232 tracker channel is set to receive relative fuel level by default.
3. If you need to filter bursts of fuel, adjust filter length by DFILTER command (section Serial ports settings) or in the Configurator on Settings/Serial ports tab. Temperature is saved only by archive dynamic structure activation (FlashArchive command, section Service commands).
4. Set transfer of received data to the server by MAINPACK command (section Server exchange protocol settings) or in the Configurator on Settings/Protocol tab. These data transmission is on by default.
5. Make sure that the tracking device receives data from the sensor. It may be done in the Configurator on Device tab.

If the tracker receives no messages from the sensor for 18 seconds, the RS232 field value will come to zero. In this way, it is possible to detect sensor disconnection or failure.

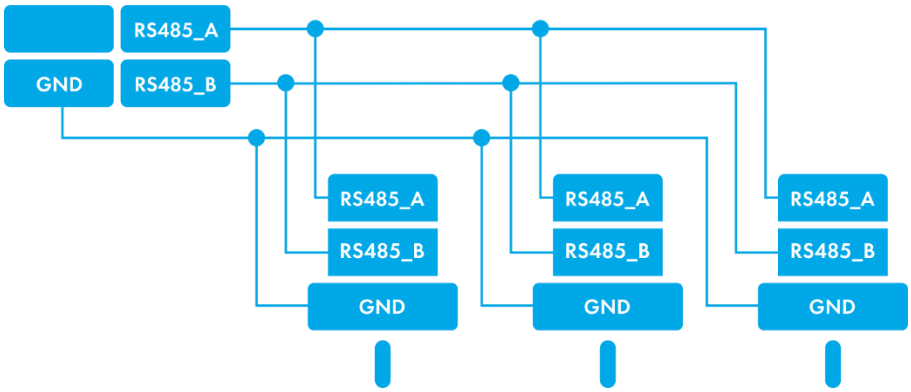
Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232/RS485. Connection and operation of digital fuel level sensors

Connecting digital fuel sensors using RS485 protocol

The order of connection:

1. Connect RS485_A; RS485_B, GND sensor contacts to tracking device RS485_A; RS485_B, and GND contacts (section Contacts description).

The sensor power supply is provided separately.



2. Set transfer of received data to the server by MAINPACK command (section Server exchange protocol settings) or in the Configurator on Settings/Protocol tab. These data transmission is on by default.
3. Make sure that the tracking device receives data from the sensor. It can be done in the Configurator on Device tab.

The tracker can support up to 16 sensors at a time. The sensors should have addresses from 0,1, 2, ..., 15 correspondingly. If the tracker receives no messages from the sensor for 18 seconds, the tracking device will not receive messages from the sensor, the RS485 field value will be set to zero. In this way, it is possible to detect sensor disconnection or failure.

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction
RS232/RS485. Connection and operation of digital fuel level sensors

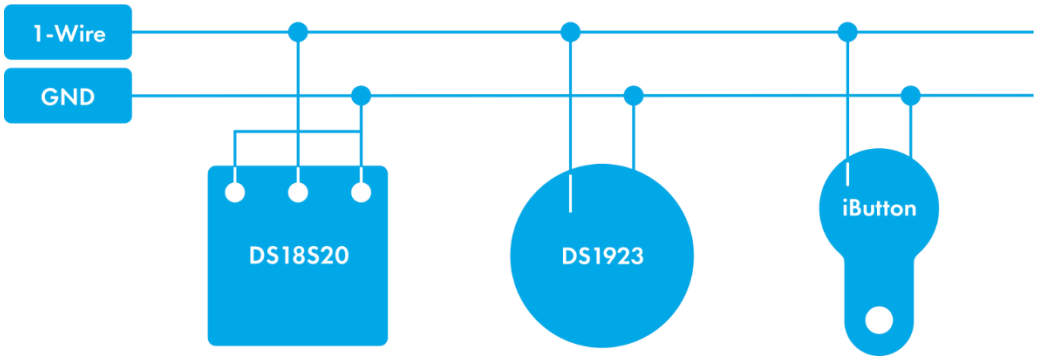
Connecting a photo camera

It`s possible to connect Galileosky and other manufacturers` cameras.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instructions RS232/RS485. Connection of Galileosky photo camera, RS232. Connection and setting of photo camera JC029F-Y01, RS232,RS485. Integration with Camera ZM-CAM30

Connecting 1Wire sensors

Can be applied only for modifications equipped with 1-Wire interface.
It is possible to connect different sensors working via 1-Wire interface, and they can operate simultaneously.



Connecting iButton (DS1990, DS1982) identification key

There are several identification key (IK) applications:

- driver identification;
- trailer detachment identification;
- doors opening identification.

In the same way, it is possible to connect devices emulating iButton, for example, RFID-codes readers.

The tracking device can support up to 8 identification keys with certain identifiers or two identification keys with any identifier. When using a microSD card up to 1000 IK with certain identifier are supported.

By identification key applying to 1-Wire and GND contacts (Contacts description) the key number is entered into the memory, the point is recorded, and four lower bytes are sent to the server without checksum. By key disconnection, the number turns to zero, the point is recorded, and message is sent to the server. Keys having code less than 100000 are stored in iButton[1] field. By iButton detachment from the tracking device, key value is set to zero in the memory cell in 4 seconds, message with the zero code is transmitted to the server. For keys with the code less than 100000, timeout from detachment to setting to zero of the memory cell is 25 minutes. Timeout for setting to zero can be set by IBCFG command (section Serial ports setting) or in the Configurator on «Settings»\«Serial ports» tab.

Eight key identifiers can be set using iButtons command (section Serial ports setting) or in the Configurator on Settings\Serial ports tab. You should enter 4 lower bytes of iButton key number without checksum, in hexadecimal system.

For example, full key hexadecimal number:
09 91 02 0C 00 00 00 5C, where
09 – type of device (in this case, it is DS1982, for DS1990 – 01),
91 02 0C 00 00 00– unique number,
5C – the checksum.
In this case, 91 02 0C 00 must be entered.

By identification key applying with one of the certain identifiers, a correspondent bit will be set in iButton connection status field. You may control it on the Device tab in the Configurator.

By using a microSD card, you may edit a list of trusted keys with the help of the AddKey, DelKey commands (section Serial ports settings). You can also connect the card to the computer and edit the list in the Configurator on the Trusted iButton keys tab. In case of connection of one of the enlisted IK, a correspondent bit will be set in the Device Status field.

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction 1-Wire. Connection and operation with iButton keys and RFID-cards

Connecting DS18S20 (DS1820, DS18B20) thermometers

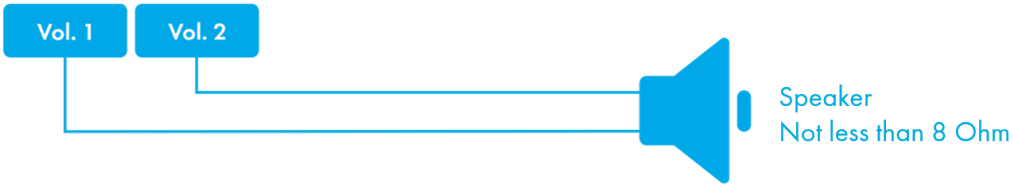
It is possible to connect up to 8 DS18S20 thermometers. To use the sensors connect them to 1-Wire and GND contacts (Contacts description) and activate the corresponding protocol items (Server exchange protocol settings). There is no linkage between a thermometer and a certain tag cell in the protocol. All data is stored in memory cells in a definite order: from a lower tag to a higher tag. If the number of cells exceeds the number of thermometers of one type, the extra higher cells will contain the data which correspond to disconnected sensor state.

By temperature sensor turning off the thermometer field shows disconnection (-128°C).

Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction 1-Wire. Connection and operation of temperature and humidity sensors

Connecting Autoinformer speaker

Can be applied only for modifications equipped with autoinformer contacts.

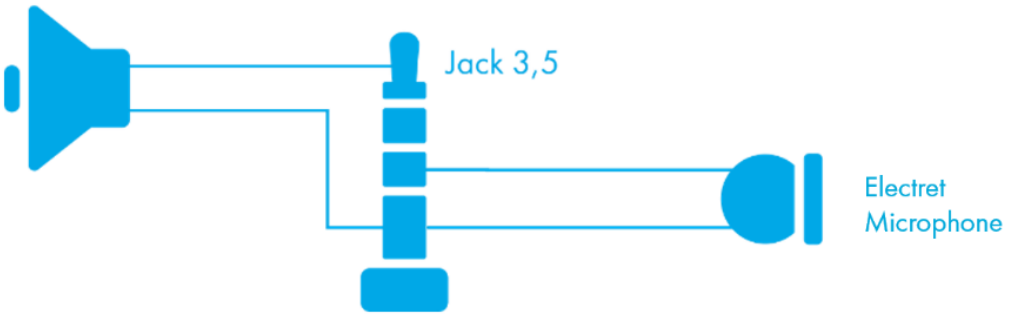


Connecting audio equipment

Can be applied only for modifications equipped with voice communication connector.

A push-to-talk can be connected directly to the tracking device from radio handheld transceivers VX-1R/VX-5R/VX-7R. For working with push-to-talk, it is needed to run command "Tangenta 1" or to set this parameter in Configurator. If you are using push-to-talk at the moment of incoming call, you may answer by pressing a push-to-talk button shortly.

Jack 3,5 Stereo or Jack 3,5 Mono connector can be used to connect the speaker separately. A speaker and a microphone can be connected simultaneously through a splitter. Telephone headset with corresponding contacts location can be connected.



Microphone specifications:

Parameter	Minimum value	Mean value	Maximum value
Operating voltage, V		1.60	2.2
Operating current, uA	70		300
Load resistance, kOhm	1.2	2.2	

Speaker specifications

Parameter	Minimum value	Mean value	Maximum value
Connected speaker resistance, Ohm	8		
Operating current, mA			+-250
Power with a 32Ohm speaker, mW		250	

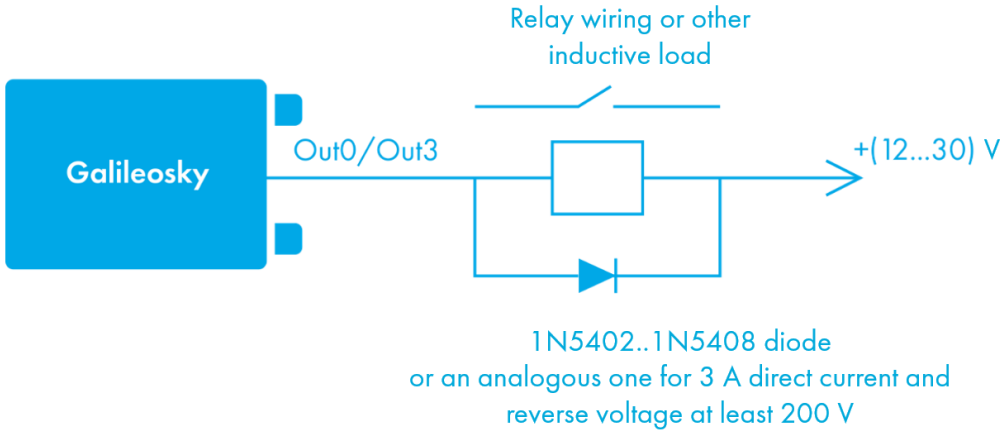
Transistor outputs (0/1)

To operate external devices there are 4 discrete «open collector» outputs (section Contacts description) in the tracking device. The maximum output voltage is -+30V, each output current is no more than 200mA.

The tracker outputs values are stored in the nonvolatile memory, so the device sets these stored values even after being reset.

To operate outputs, use Out command (section Transistor outputs settings) or the Settings/Inputs/Outputs tab in the Configurator.

OUT0...OUT3 outputs relay connection circuit



Connecting Garmin navigators supporting FMI protocol

Can be applied only for modifications equipped with RS232 interface

The tracking device can serve as a gateway for Garmin FMI protocol packets transmission to the server and back. Garmin FMI protocol enables:

- drivers identification by code numbers (driver enters his/her number into a navigator, the number is transmitted to the server);
- driver status transmission to the server (driver selects his/her status on a navigator display);
- text messages exchange between Garmin navigator and the server;
- transmission of route end points from the server to the navigator (in this case, the navigator calculates optimal path to the set points and displays it on the map);
- control for speeding and informing the server about it;
- geofences transmission from the server to the navigator and displaying them.

A description of protocol abilities can be found on http://www8.garmin.com/solutions/pnd .

The tracker only transmits the packets from the navigator to the server and back, so parsing the packets should be done at the server.

To connect Garmin navigator:

1. Purchase an interface Garmin FMI cable (for example, Garmin FMI 10), cable is to be chosen according to the navigator model. The cable has a mini-USB socket to connect to the navigator at one end, and supply and RS232 interface contacts at another end.
2. Connect RS232 interface contacts to the tracker RS232 zero port; connect RX contact of the cable to the TXD contact of the tracker; TX contact of the cable to the RXD contact of the tracking device; be sure to connect GND RS232 contact of the cable to the GND contact of the tracking device.
3. Connect cable power supply.
4. Connect the cable to the navigator.
5. Configure RS232 port in the tracker to work with the navigator. It can be done by RS2320 5 command or in the Configurator on Settings\Serial ports tab.

To make sure that the tracking device and the navigator connection works properly select messages from necessary RS232 port in troubleshooting. If the connection is correct, the following messages will be displayed regularly:

RS2320. Garmin FMI. Enable FMI.

RS2320. Garmin FMI. ACK.

If the connection is right, the navigator interface will change, the left-hand icon will take the form of a truck, it will give an access to messages sending and receiving functions.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connection of Garmin navigator

Connecting REP-500 electrical meter

Can be applied only for modifications equipped with RS232 interface

The REP-500 is an electrical meter, which can be connected to the tracking device via RS232 interface. To connect the REP-500, perform the following steps:

1. Connect RXD, GND contacts of the tracker to TX, Ground contacts of the REP-500 correspondingly.
2. Configure the RS232 port in the tracking device for operation with the meter. This can be performed using the RS2320 10 command or in the Configurator on the Settings\Serial ports tab.
3. Set the meter readings transfer to the server with the help of MainPackBit 171,1 command.

Connecting CAN-LOG

Can be applied only for modifications equipped with RS232 interface.

The CAN-LOG device is designed for CAN-bus data reading. It allows to get the data on:

- total time of engine operation;
- total mileage of a vehicle;
- whole fuel consumption of a vehicle since the date of the vehicle`s creation;
- fuel level in % or liters;
- engine speed;
- coolant temperature;
- vehicle speed;
- axle load.

The tracker supports CAN-LOG with string (P-series) and binary (B-series) protocols.

The following prefixes are deciphered automatically: A, B, C, D, E, F, G, R, H, I, J, K, L, M, N, O, S, P, WA.

If the tracking device detected one of known prefixes, the following decipher stops. Thus, its value will not be in the list of user prefixes.

The CAN-LOG support was added to secure compatibility with the installed board equipment. By initial installation it is recommended to use the device built-in functions for CAN-bus operation (section CAN-interface). Advantages of using built-in functions of the device:

- no extra costs;
- a possibility to scan the CAN-bus;
- a possibility to read all CAN-bus data, and not just those listed for the CAN-LOG.

To connect the CAN-LOG with string (P-series) protocol support you have to:

1. Connect RXD, TXD, GND contacts of the tracking device and RS232 TX, RS232 RX, CAN-LOG's mass correspondingly.
2. Configure RS232 port in the tracker for operation with CAN-LOG. It can be done by RS2320 6 command or in the Configurator on Settings\Serial ports tab.
3. Disconnect the tracker built-in functions for operation with CAN-bus. It can be done by CANREGIME 0,250000,2000 command or in the Configurator on Settings\CAN tab.
4. Specify the parameters transferred to the server. HEADPACK or MAINPACK commands can also be used (section Server exchange protocol settings) or in the Configurator on Settings/Track tab.

To connect CAN-LOG with binary (series B) protocol support, a special script must be loaded into the tracker.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instructions RS232/RS485. CAN-LOG connection and operation, RS485/RS232 Connection and Setting of CAN-LOG (since January, 2018)

Connecting CUB5B indicator

Can be applied only for modifications equipped with RS232 interface.

The CUB5B indicator is a segment 8-bit indicator, which can be connected to the tracking device using RS232 interface. It can display current sensors data, CAN-bus data, mileage and so on. Information updates every second.

Connection:

1. If the indicator was set to operate with other equipment, it is necessary to reset to factory settings. If the indicator is new, it is unnecessary.
2. Connect RXD, TXD, GND tracker contacts and TX, RX and Ground contacts of the CUB5B correspondingly.
3. Configure RS232 port in the tracking device for operation with indicator, it can be done by RS2320 7 command or in the Configurator on Settings\Serial ports tab.
4. Choose the displayed parameter, it can be done by CUB5 command or in the Configurator on Settings\Serial ports tab.

Connecting CI5010A weight indicator

Can be applied only for modifications equipped with RS232 interface.

The CI-5010A weight indicator is designed to measure, control and indicate electrical signals produced by weight measuring strain gages sensors. It can be connected to the tracking device via RS232 interface.

The order of CI-5010A connection:

1. Adjust the indicator to output the measured value constantly.
2. Connect RXD, TXD, GND contacts of the tracker and TX, RX, Ground of CI-5010A correspondingly.
3. Configure the RS232 port in the tracker to operate with the indicator. It can be done by the RS2320 12 command or in the Configurator on Settings\Serial ports tab.
4. Set «RS232» tag for sending to the server;
5. If possible, measured weight exceeds the interval [0, 65535], turn on the dynamic archive and set «Extended RS232 data» tag sending.

The data from the indicator are rounded up to a whole number. Measured weight is transmitted in two tags: «RS232» and «Extended RS232 data». If measured weight is within the interval from 0 to 65535 kg, it is transmitted only in «RS232» tag, weight is calculated in a different manner according to a formula «RS232»+65535x«Extended RS232 data». Signed number is transmitted in «Extended RS232 data» tag, thus, it is possible to receive both positive and negative weight.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connection of Weight Indicators CI-5010A, Tenso-M 003_05D, AWT640, WIN scale

Connecting Tenzo-M weight indicator

Can be applied only for modifications equipped with RS232 interface.

The Tenzo-M weight indicator is designed to measure, control and indicate electrical signals produced by weight measuring strain gages sensors. It can be connected to the tracking device via RS232 interface.

The order of Tenzo-M connection:

1. Adjust the indicator to output the measured value constantly. Port speed is 2400 bit/s.
2. Connect RXD, TXD, GND contacts of the tracking device and TX, RX, Ground of Tenzo-M correspondingly.
3. Configure the RS232 port in the tracking device to operate with the indicator. It can be done by the RS2320 16 command or in the Configurator on Settings\Serial ports tab.
4. Set «RS232» tag for sending to the server;

5. If possible, measured weight exceeds the interval [0, 65535], turn on the dynamic archive and set «Extended RS232 data» tag sending.

The data from the indicator are rounded up to a whole number. Measured weight is transmitted in two tags: «RS232» and «Extended RS232 data». If measured weight is within the interval from 0 to 65535 kg, it is transmitted only in «RS232» tag, weight is calculated in a different manner according to a formula «RS232»+65535x«Extended RS232 data». Signed number is transmitted in «Extended RS232 data» tag, thus, it is possible to receive both positive and negative weight.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connection of Weight Indicators CI-5010A, Tenso-M 003_05D, AWT640, WIN scale

Connecting AWT 640 weight indicator

Can be applied only for modifications equipped with RS232 interface.

The AWT 640 weight indicator is designed to measure, control and indicate electrical signals produced by weight measuring strain gages sensors. It can be connected to the tracker via RS232 interface.

The order of AWT 640 connection:

1. Adjust the indicator to output the measured value constantly. Port speed is 9600 bit/s.
2. Connect RXD, TXD, GND contacts of the tracking device and TX, RX, Ground of AWT 640 correspondingly.
3. Configure the RS232 port in the tracker to operate with the indicator. It can be done by the RS2320 18 command or in the Configurator on Settings\Serial ports tab.
4. Set «RS232» tag for sending to the server;
5. If possible, measured weight exceeds the interval [0, 65535], turn on the dynamic archive and set «Extended RS232 data» tag sending.

The data from the indicator are rounded up to a whole number. By receiving the message about the changed weight from the indicator, the tracking device records a point and sends it to the server. Measured weight is transmitted in two tags: «RS232» and «Extended RS232 data». If measured weight is within the interval from 0 to 65535 kg, it is transmitted only in «RS232» tag, weight is calculated in a different manner according to a formula «RS232»+65535×«Extended RS232 data». Signed number is transmitted in «Extended RS232 data» tag, thus, it is possible to receive both positive and negative weight.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connection of Weight Indicators CI-5010A, Tenso-M 003_05D, AWT640, WIN scale

Connecting WIN SCALE weight indicator

Can be applied only for modifications equipped with RS232 interface.

The Dinamica Generale S.pA. WIN SCALE weight indicator is designed to measure, control and indicate electrical signals produced by weight measuring strain gages sensors. It can be connected to the tracking device via RS232 interface.

The order of WIN SCALE connection:

1. Adjust the indicator to output the measured value constantly. Port speed is 9600 bit/s.
2. Connect RXD1, TXD1, GND contacts of the tracking device and TX, RX, Ground of WIN SCALE correspondingly.
3. Configure the RS232 port in the tracking device to operate with the indicator. It can be done by the RS2320 19 command or in the Configurator on Settings\Serial ports tab.

The data from the indicator are rounded up to a whole number. By receiving the message about the changed weight from the indicator, the tracking device records a point and sends it to the server. If measured weight is within the interval from 0 to 65535 kg, it is transmitted only in «RS232» tag, weight is calculated in a different manner according to a formula «RS232»+65535×«Extended RS232 data». Signed number is transmitted in «Extended RS232 data» tag, thus, it is possible to receive both positive and negative weight.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connection of Weight Indicators CI-5010A, Tenso-M 003_05D, AWT640, WIN scale

Connecting devices operating via WIEGAND26 protocol

To connect the devices operating via WIEGAND26 protocol:

1. Connect In0, In1 (or In2, In3), GND contacts of the tracker with Data0, Data1, GND of the device correspondingly.
2. Set analog inputs of the tracking device for operation with device via WIEGAND26 protocol. It can be set by commands InCfg0 4, InCfg1 5 or in Configurator program, tab “Settings”/”Serial ports”.

The values received from the device will be stored in iButton[1]. Every 4 seconds the values will reset.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction DAI. Connection of a keyboard with RFID-reader through Wiegand-26

Connecting Matrix 5 RFID-codes reader via RS485 interface

The order of Matrix5 connection:

1. Connect the RS485A, RS485B, GND contacts of the tracker and the RS485A, RS485B, GND of the Matrix5 correspondingly.
2. Configure the RS485 port in the tracking device to operate with the reader. It can be done by the RS485FN 1 command or in the Configurator on Settings\Serial ports tab.
3. Reset the tracker.
4. The values received from the device will be stored in iButton[1].

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS485. Connection and operation of Matrix5 reader

Connecting DBG-S11D dosimeter via RS485 interface

DBG-S11D dosimeter is intended for continuous measuring ambient equivalent dose rate (ADER). The tracking device enables to receive the current ADER readings in 3V/h and data on dosimeter state. Then this information will be sent to the server. One dosimeter operation is supported. The dosimeter can be connected simultaneously with fuel sensors and Galileosky photo camera.

The order of DBG-S11D dosimeter connection:

1. Connect the RS485A, RS485B, GND contacts of the tracker and the RS485A (contact 1), RS485B (contacts 2), GND (contact 6) of the dosimeter correspondingly. Dosimeter has separate power supply.
2. Configure the RS485 port in the tracker to operate with fuel level sensors, photo cameras and dosimeter. It can be done by the RS485FN 2 command or in the Configurator on Settings\Serial ports tab.
3. Specify dosimeter data in the transmission protocol to the server. If dosimeter data sending is off, the tracking device will not sample DBG-S11D.

Connecting PressurePro tires pressure monitoring system

Can be applied only for modifications equipped with RS232 interface.

PressurePro tires pressure monitoring system enables to carry out continuous control of pressure in tires and tire temperature and warns about critical situations. The system

consists of wireless pressure sensors, installed on tires, and monitor, receiving data from sensors. The tracking device connects to PressurePro monitor via RS232 interface and receives data about sensors state and measured parameters.

The tracker supports several Ttire-pressure monitoring systems: PressurePro, PressurePro Pulse, TPMS 6-13, TM-508 (AIRGUARD).

Connecting PressurePro:

1. Connect RXD, TXD, GND contacts of the tracker and TXD (white), RXD (green), GND (black) contacts of monitor correspondingly. Monitor has separate power supply.
2. Configure the RS232 port in the tracking device to operate with PressurePro. It can be performed by the RS2320 13 command or in the Configurator on Settings\Serial ports tab.
3. Specify PressurePro data in the transmission protocol to the server.

By default, PressurePro sensors transmit data once in 5 minutes or by emergency arising (tripping sensors, pressure reduction and etc.). The tracking device transmits PressurePro data to the server by analogy: once in 5 minutes or by emergency arising, data are not transmitted at passing points to save GPRS-traffic. Operation of up to 34 sensors is supported.

To connect PressurePro Pulse, TPMS 6-13, TM-508 (AIRGUARD) a special script must be uploaded to the tracker.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instructions RS232. PressurePro connection, RS232. PressurePro Pulse connection, RS232. Connection of TPMS 6-13, RS232. Connecting TPMS TM-508 (AIRGUARD)

Connecting CarrierDataCOLD500, ThermoKing, Euroscan refrigeration units temperature recorders

Can be applied only for modifications equipped with RS232 interface.

Temperature recorders allow transmitting the information about the refrigerator state to the tracking device, which, in its turn, sends to the server the following transformed data:

1. operation mode of refrigeration unit;
2. cargo temperature (for one- and multi temperature units);
3. alarm signals;
4. total time of engine operation.

Connecting a temperature recorder:

1. Connect RXD, TXD, GND contacts of the tracking device and TXD, RXD, GND contacts of temperature recorder correspondingly. Temperature recorder has separate power supply.
2. Configure the RS232 port in the tracker to operate with one of temperature recorder types. It can be performed by the RS2320 command (DATACOLD500: RS2320 11, ThermoKing: RS2320 14, EuroScan: RS2320 15) or in the Configurator on Settings\Serial ports tab.
3. Specify temperature recorder data in the transmission protocol to the server.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction RS232. Connecting refrigerators equipment

Configurator

5

Configurator is a PC program, which allows:

- configuring the tracker via graphic interface and with the help of commands;
- troubleshooting the tracking device saving the results in a log-file;
- seeing the tracker units state in real time mode;
- downloading monitoring data from the internal memory and a SD card;
- sending the downloaded data to the server;
- specifying areas for autoinformer.

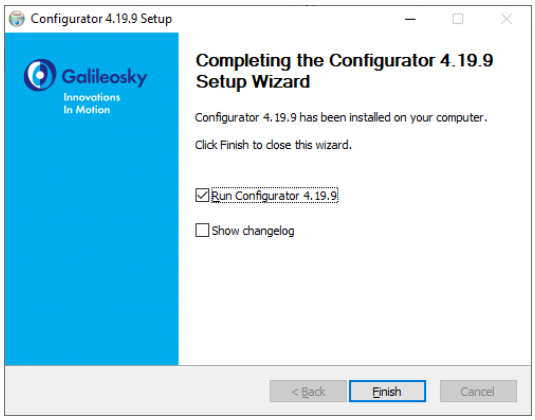
32 and 64 bit OS are supported: Windows 7, Windows 8, Windows 10.

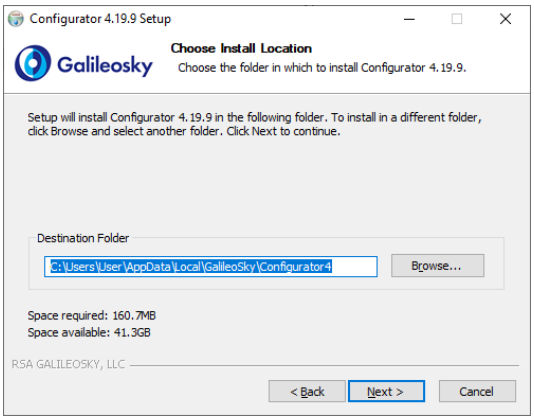
Launching the Software

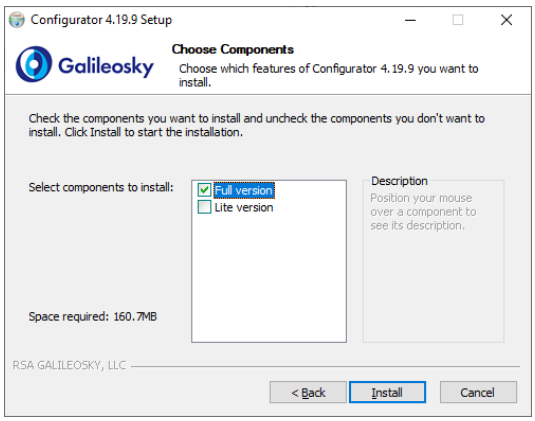
Download the Configurator program from the site and launch it.

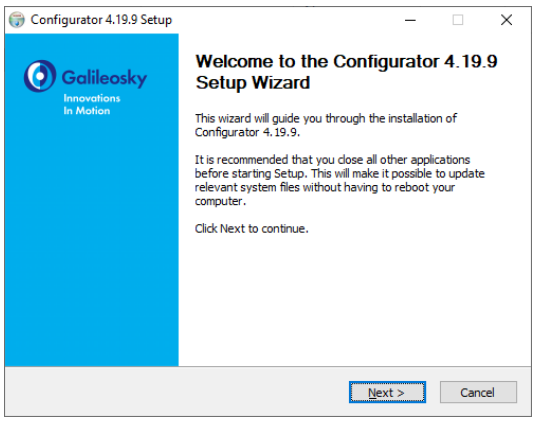
In case of a security system warning, confirm launching the program.

ATTENTION! Program installation may require changes of crucial OS elements. Do not let your antivirus program block the installer operation.









During the installation of the Configurator old drivers will be deleted and new ones will be installed. It is possible to install the major version of the Configurator or a limited one. The latter one allows uploading archive and receiving the current parameters of sensors, but not changing the settings.

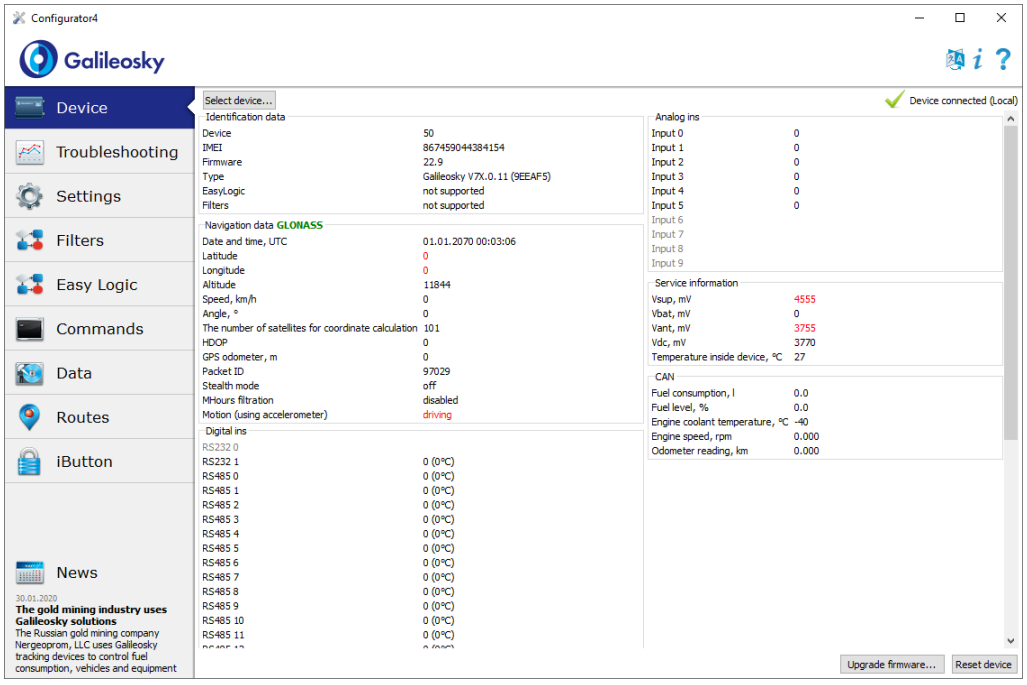
Start the Configurator program (from Start menu\Programs\Galileosky\Configurator4).

Turn on the power of the tracker and connect it to the computer via a USB-cable.

After the tracker connection the program loads all the device`s settings parameters automatically. If the program identifies the tracker, all the buttons on the vertical left-hand panel will be active.

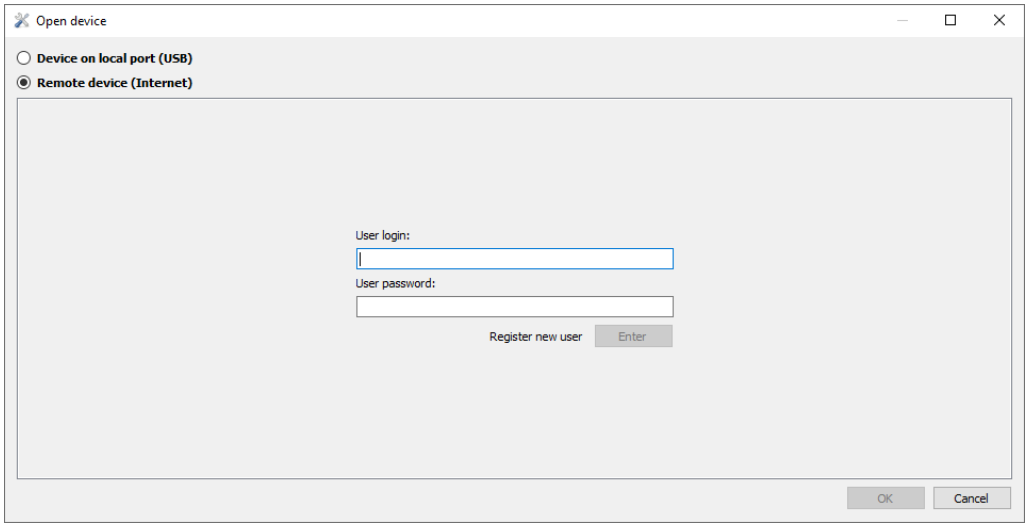
Device tab

The tab displays the information about the tracker state and allows its resetting. This tab contains the tracker model, oriented in space according to accelerometer indications. The model can be rotated by mouse. Parameter values, which are beyond the limits, wrong coordinates, exceeding of maximum incline angle and responses on inputs are shown in red.

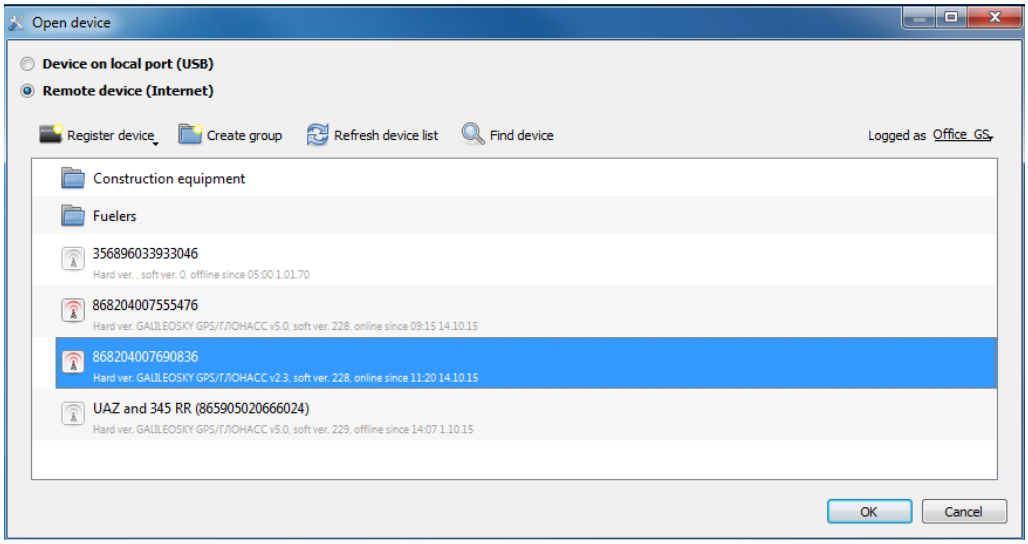


If there is a PIN code in the tracker, the program will request it to access the settings. By wrong code entering the tracking device will disconnect from the computer, reset, connect to the Configurator again and wait for the right code enter.

For remote configuration and troubleshooting of the tracker, click Select device... button. In the window appeared, enter your login and password to get the access to the remote configuration server. You can get the login and the password in RSA "Galileosky", LLC. department of technical support or by clicking the Register new user button.



After successful authorization on the server, the form of devices list management will become available. When connecting for the first time, the list of the controlled tracking devices will be empty. To add a tracker to the list, click "Register device" button. During registration the Configurator will request a password for a particular tracker, by-default password corresponds to IMEI of the tracking device; this can be later changed in the Configurator by the user. Trackers may be grouped.



After selecting a particular tracking device, it can be controlled via the Configurator, the same way as it occurs with the USB-connection.
Further information is provided on the page
https://galileosky.com/podderzhka/dokumentacziya.html in the instruction Service. Remote configuration of Galileosky tracking devices

Troubleshooting tab

This tab allows us to see the current device state through the troubleshooting reports.

The troubleshooting mode has the following buttons:

1) **Start /Stop**
2) The time scale displays the information about the server connection, packet recording, updating coordinates etc. and with a 10 sec interval.
3) **Clear troubleshooting window**
4) **Save** the tracker`s troubleshooting results as a log-file which can be opened by any text editor.
5) **Search** in the troubleshooting history file.

GSM/3G-unit debug info

Command mode tab

This tab is intended to message a single command or a set of commands to the tracking device.

The command mode has the following buttons:

1) Run commands;
2) Run single command;
3) Open from file;
4) Save to file.

The commands will be identified whether you use capital or lower-case letters or both in turn.

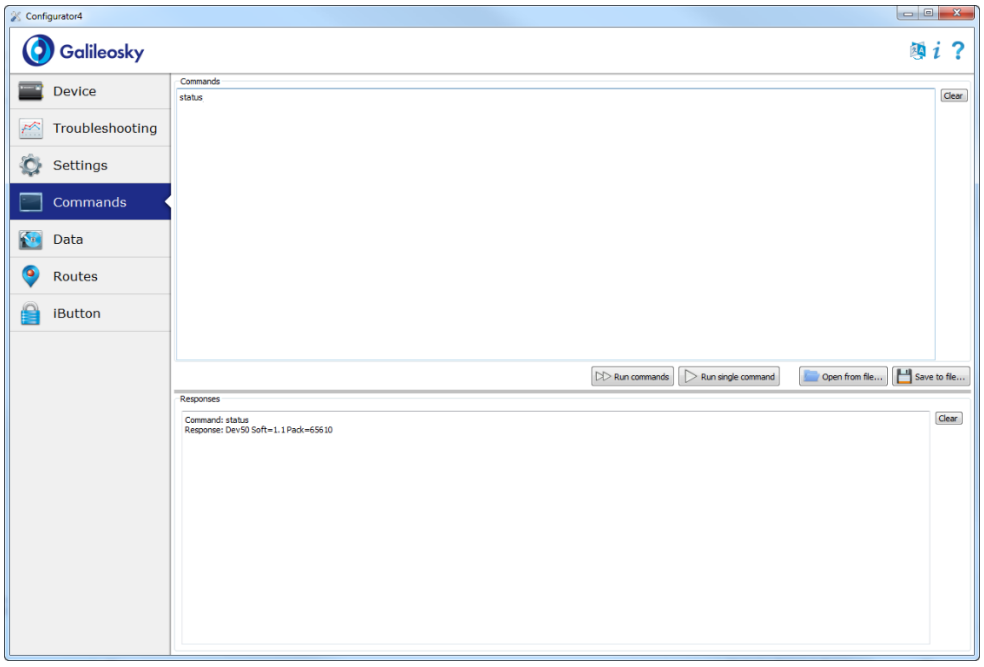
ATTENTION!

There are no spaces in command name!

Spaces between parameters are not allowed!

Commands and parameters are separated by space.

Commands are separated by Enter.



Single command performance example

An example of a command with a parameter:

Enter APN internet.beeline.ru,beeline,beeline as shown in the figure above and press **Run single command** button. The command and a response will be displayed in the Responses window.

To access the parameters in the device memory, you should use a command without parameters!

Command: APN internet.beeline.ru,beeline,beeline

Response: GPRS: APN=INTERNET.BEELINE.RU, user=BEELINE, pass=BEELINE

An example of a command without a parameter:

APN command	Request: APN Response: GPRS:APN=INTERNET.BEELINE.RU,user=BEELINE,pass=BEELINE

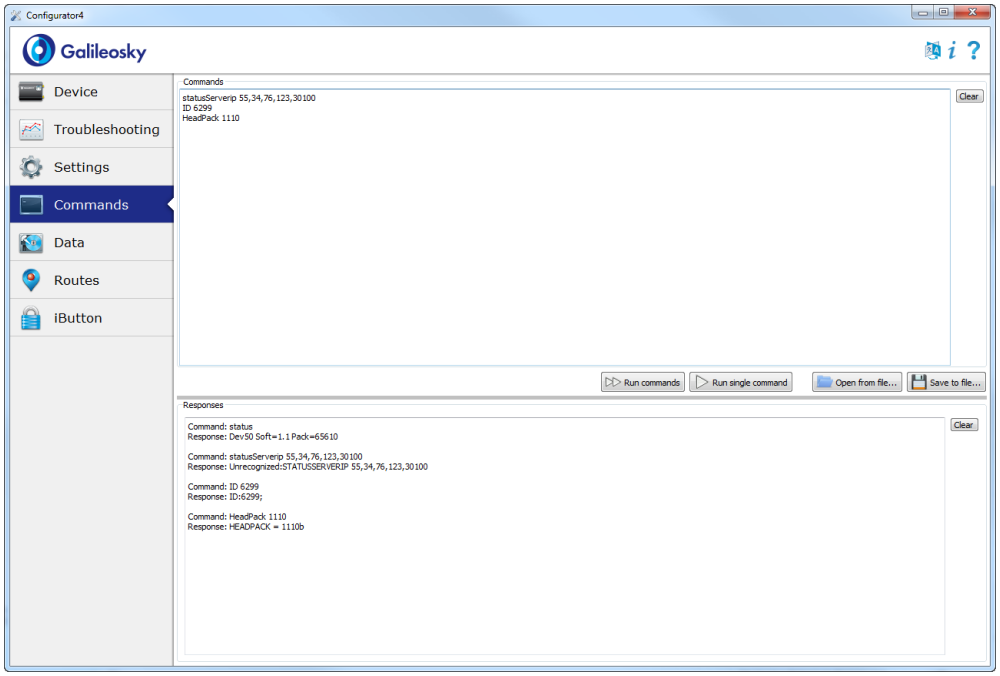
Set of commands example

Enter the necessary commands in Commands window, each beginning a new line, as shown in the figure below and press the **Run commands** button.

Example: Serverip 55,34,76,123,30100
ID 6299
HeadPack 1110

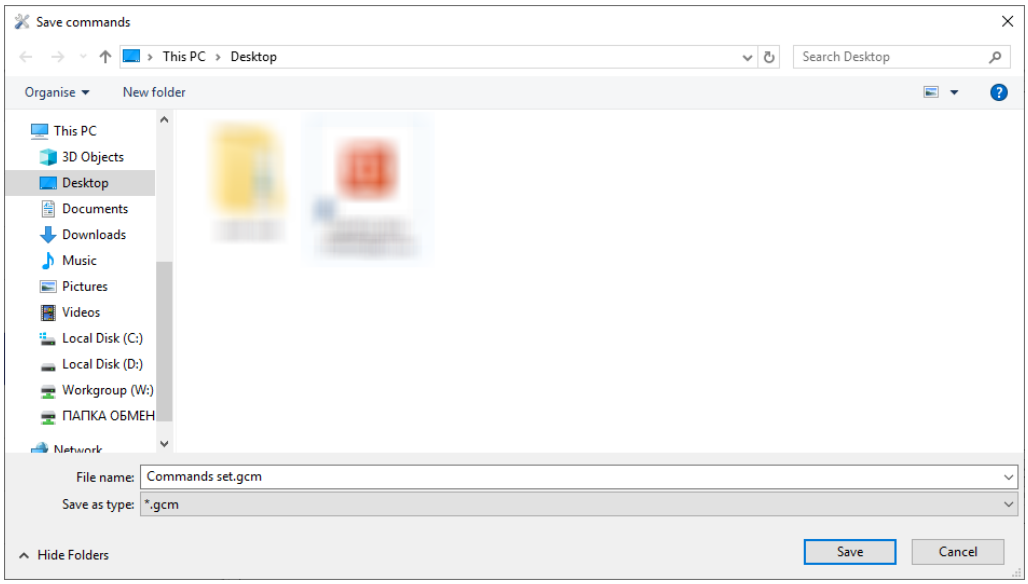
The given commands and results will be displayed in the Responses window.

Command: Serverip 55,34,76,123,30100
Response: ServerIp=55.34.76.123:30100
Command: ID 6299
Response: ID: 6299
Command: HeadPack 1110
Response: HeadPack =
00000000000000000000000000000000000000000000000000000000000001
110b

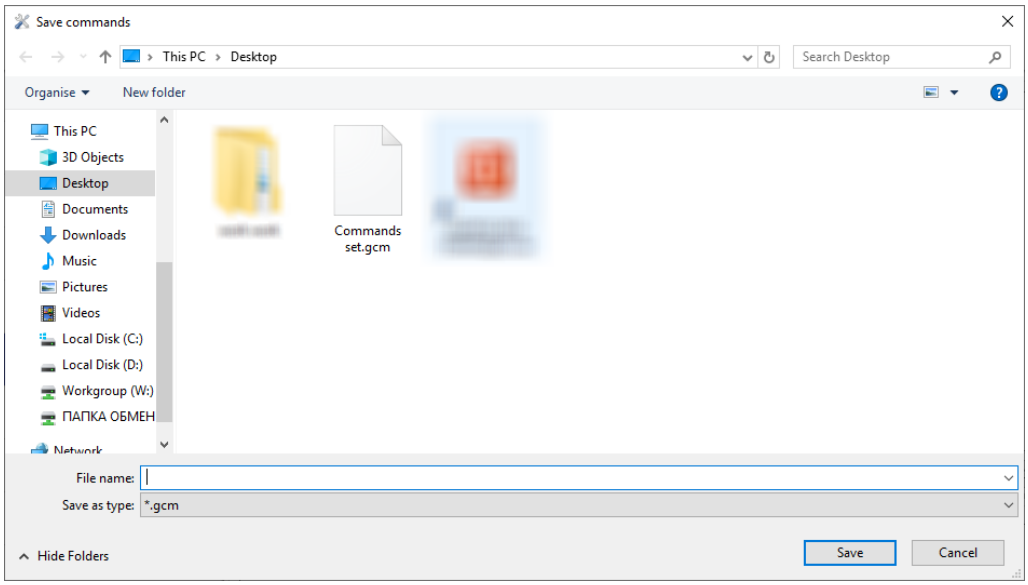


Example of saving and downloading parameters set

For quick configuration of several tracking devices with the same set of commands it is recommended to run the commands from a pre-saved file. To do this, enter a list of commands in the Command window. Make sure that they are typed correctly by pressing the **Run commands** button and then press **Save to file**.



The file will be saved in log configurator folder. Then press **Open from file...** button.



To run several commands at the same time press **Run commands** button.

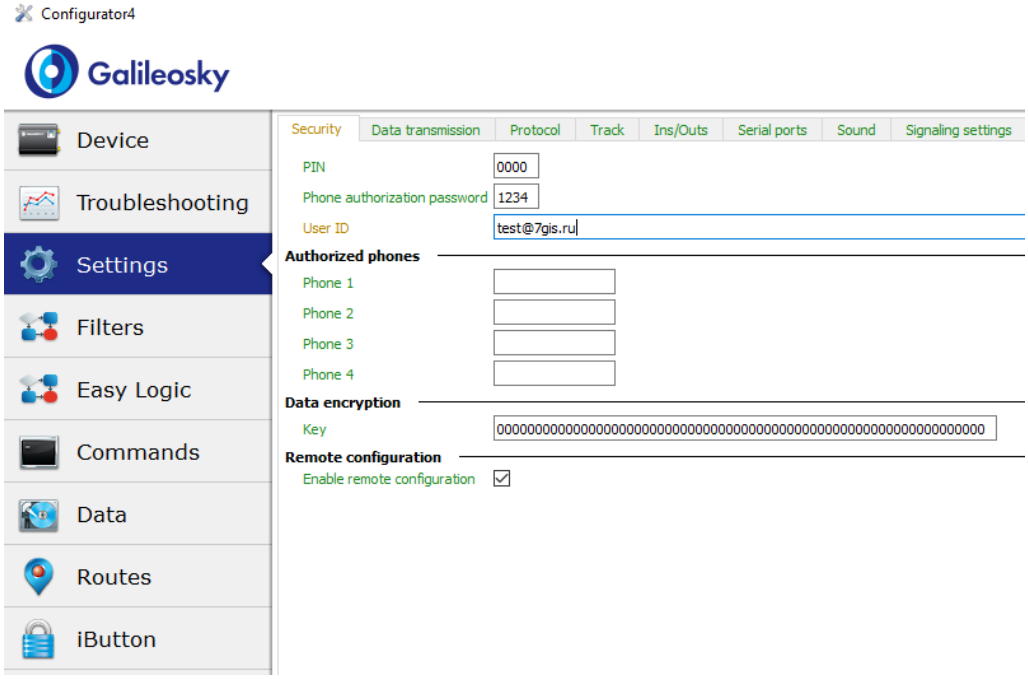
To run only one command, it is necessary to go to it in Commands window and press **Run single command** button.

Graphic interface settings

All main settings of the tracker are placed on tabs in the program upper part.

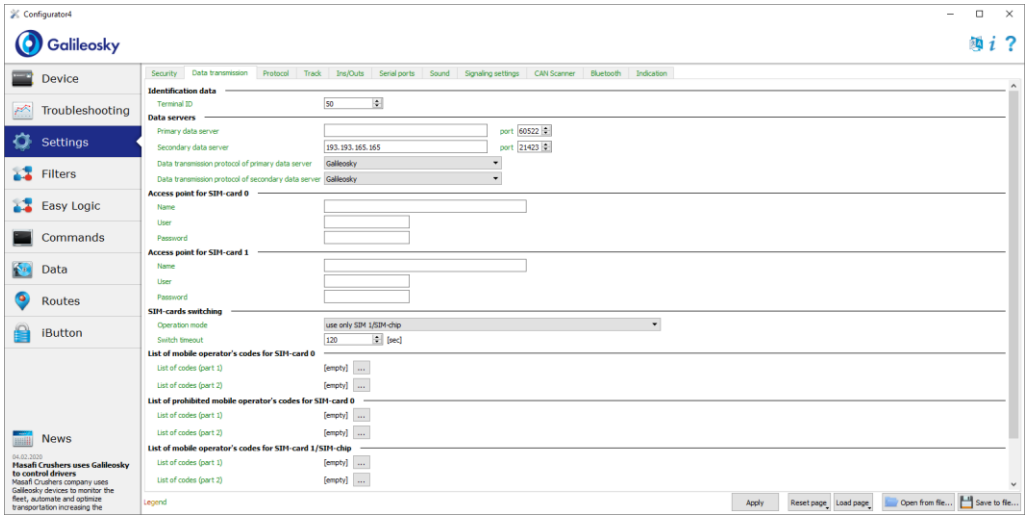
Security

This tab allows setting SIM-cards PIN code, phone authorization password, list of authorized phone numbers and encryption key for data transfer to the server.



Data Transmission

This tab allows setting SIM-card PIN code, APN for the Internet access, monitoring data processing servers, Wi-Fi access point.



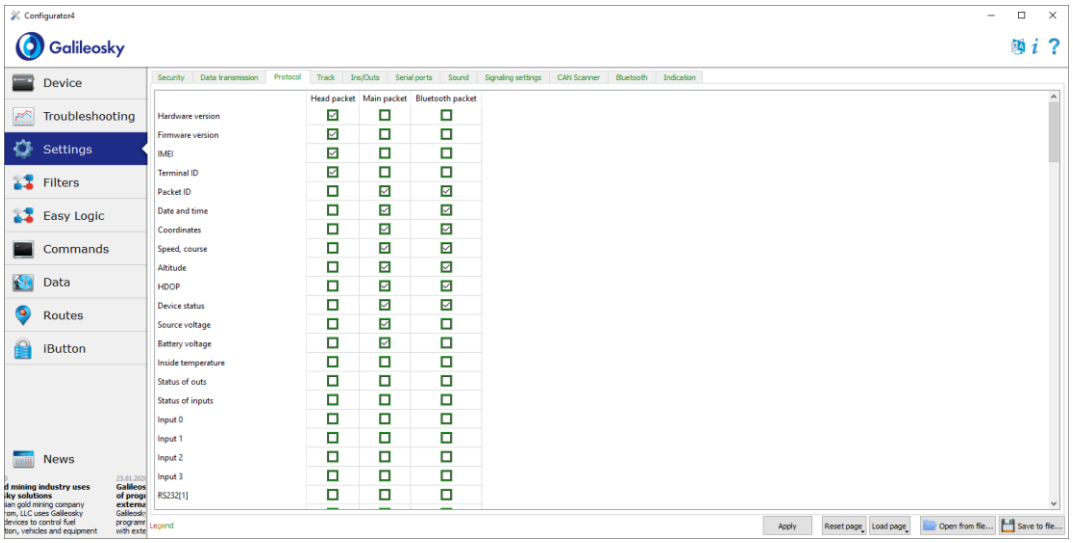
Protocol

The tracking device has its own data transmission protocol developed by RSA "Galileosky", LLC.

During device operating and data sending to the server, the following stages are possible:

1) Initialization of TCP/IP connection (does not need any additional settings);
2) Sending of initialization data described in the Head packet column (the data to be sent to the server are ticked in the first column);
3) If the tracking device has passed the first two stages, it starts sending accumulated packets according to the format described in the Main packet column.

To send the data the modem establishes a server connection and keeps it active even after sending the packet. It is done to save server connection traffic used to establish the connection to the server.



Track

This tab allows to set archive storage place and recording periods of coordinates at stops and in motion, details of track and false coordinates filtering.

The device filters coordinates by speed, acceleration, travelled distance, horizontal accuracy, number of satellites.

In addition, the tracking device allows filtering of coordinates crowding during stops by supply voltage at vehicles battery (Mhours command).

Parameters:

- supply voltage at stopped engine;
- supply voltage at started engine;

The first parameter is selected in the following way:

1) stop the engine for 5 minutes;
2) save the Vsupply voltage parameter from Device tab.

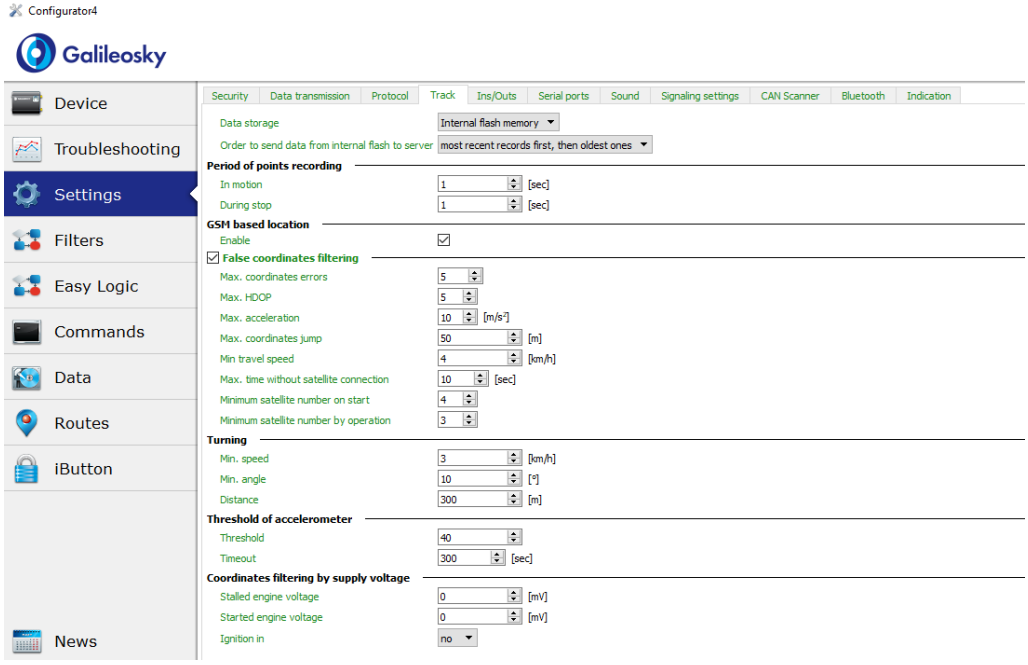
The second parameter is selected in the following way:

1) start the engine;
2) save the Vsupply parameter;
3) parameters of the mhours command are filled in and sent to the tracking device.

When the engine is started, the 9th bit will be set in the device status.

Each tracker is equipped with an accelerometer which allows filtering coordinates crowding during stops. It is based on vehicles vibration.

Parameters:
Sensitivity – a standard unit, where the sensitivity of the 600 units corresponds to the acceleration of 1g (gravitational acceleration)
Time parameter. The tracker switches on this filter when there is no vibration within a certain time period. The filter operates until the necessary amplitude acceleration is reached.



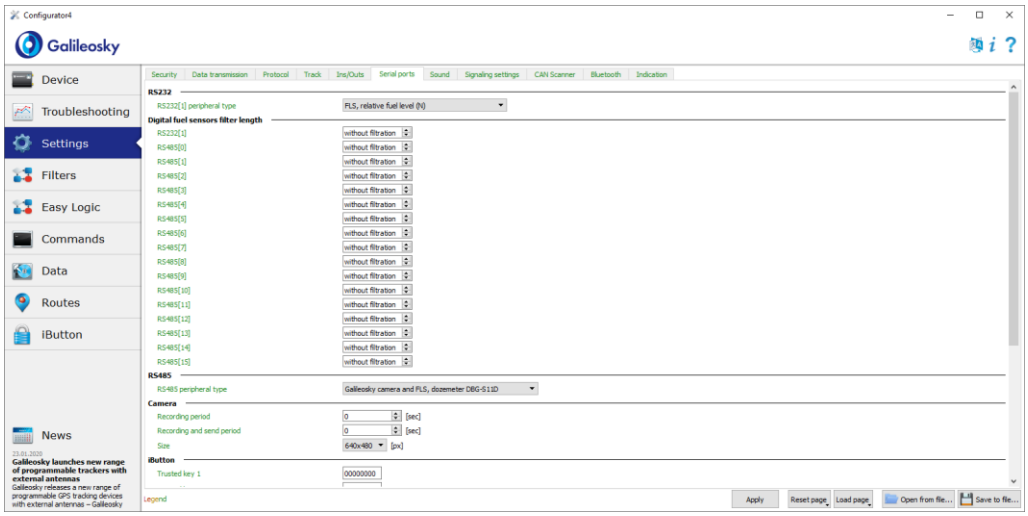
Inputs/Outputs

For inputs operating principles see section Discrete-analog inputs (DAI).

For discrete outputs description see section Transistor outputs (0/1).

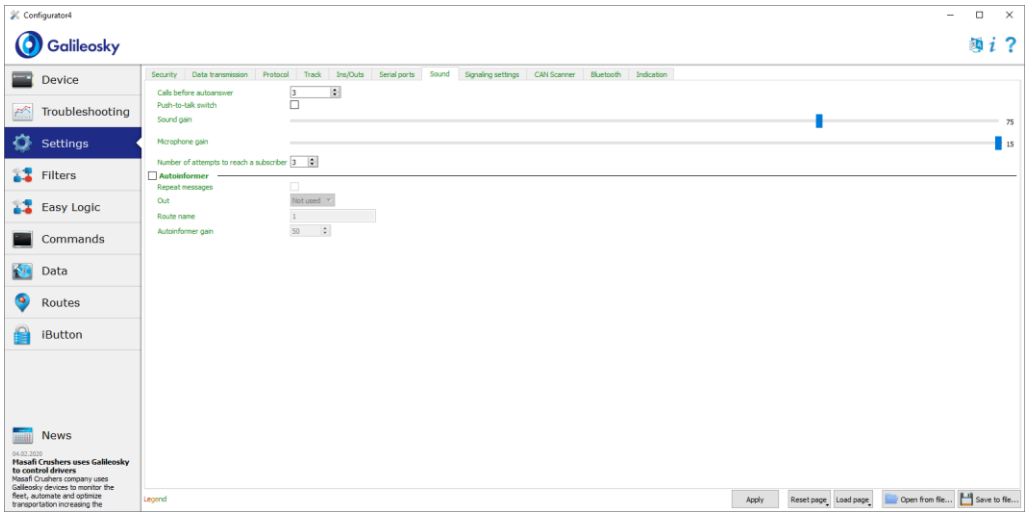
Serial ports

This option allows choosing the type of peripheral connected to RS232 inputs, setting iButton keys, setting up periodical camera shooting.



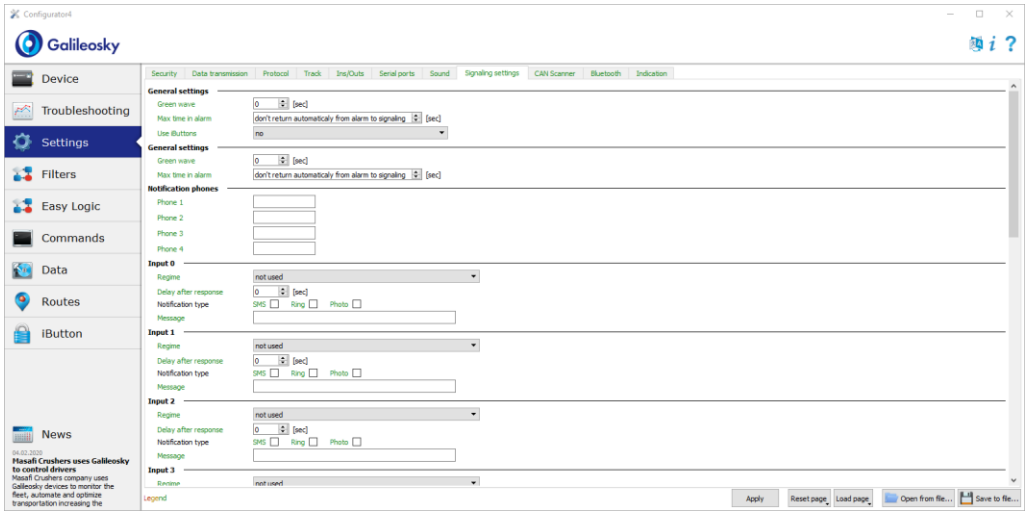
Sound

This option allows setting a microphone and sound gain via a GSM channel, the number of calls before autoanswer, as well as parameters of the Autoinformer.



Signaling settings

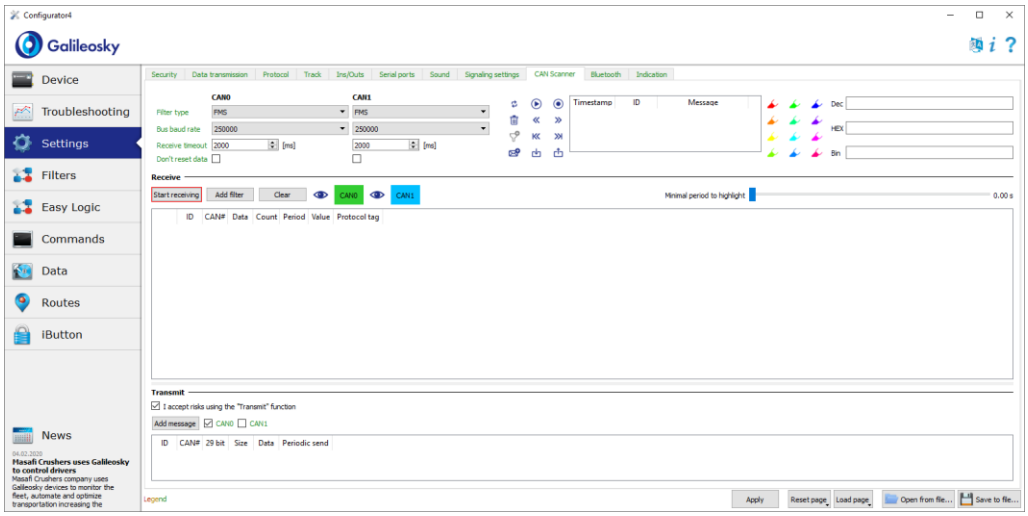
This tab allows you to configure the tracker reactions to external events when signals appear on serial ports, apply trusted or any iButton keys or RFID cards, detect strike and incline by inverting the status of the ports or applying pulses to them, taking a picture or sending SMS.



CAN

This option allows setting a CAN-filter and scanning the CAN bus for message identifiers being used.

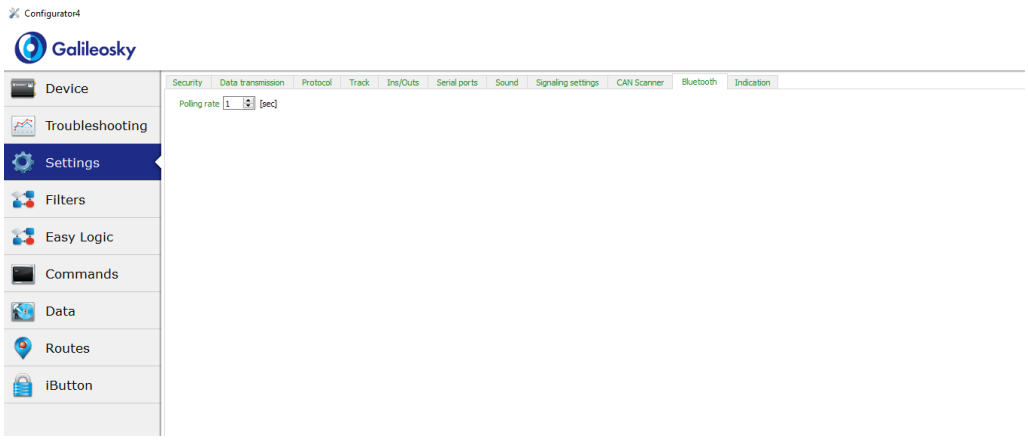
After clicking 'Start receiving' the CAN scanner will be activated and received messages will be displayed in the right-hand panel. When scanning is completed, it is possible to set tags in the protocol, in which the bus data will be sent. To do this: choose CAN-identifier and tag and point transmitted part of the message with a mouse. To delete the filter the corresponding message identifier should be selected and deleted.



Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction User Manual for work with CAN-logger

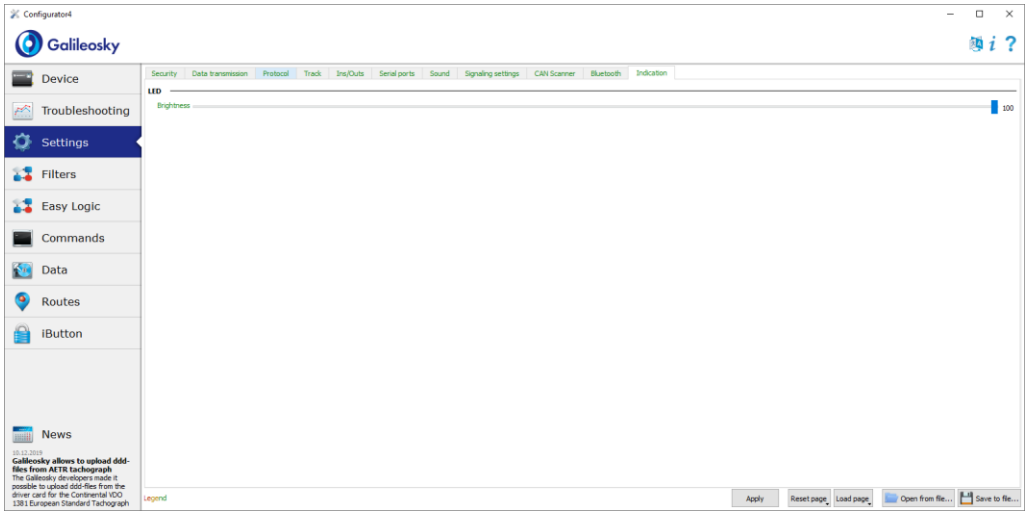
Bluetooth

This tab allows to set the frequency of sending data via Bluetooth.



Indication

This tab allows to change the brightness of the LED in the tracker.

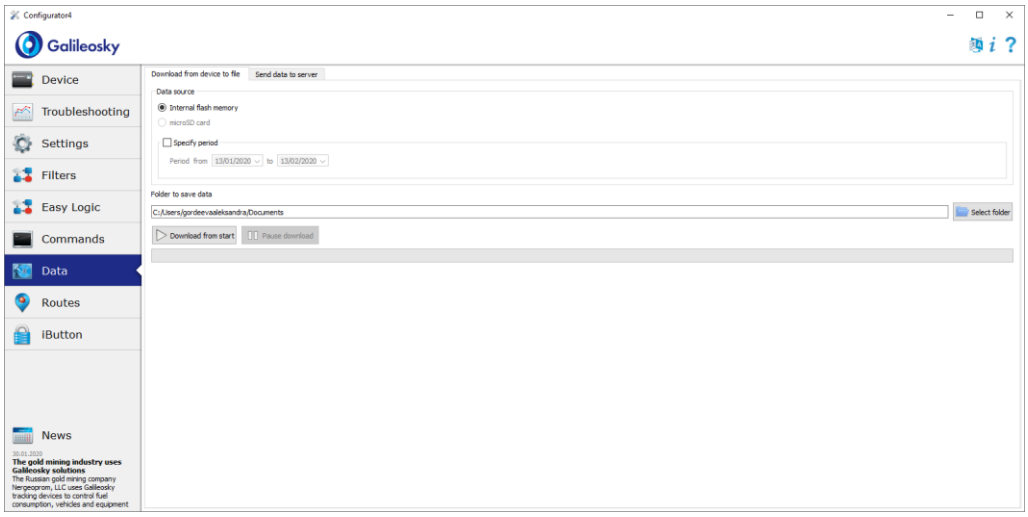


Data loading and sending to server

Data loading from the Tracker to file

This option allows transferring the data from the internal memory to a computer file via a USB cable.

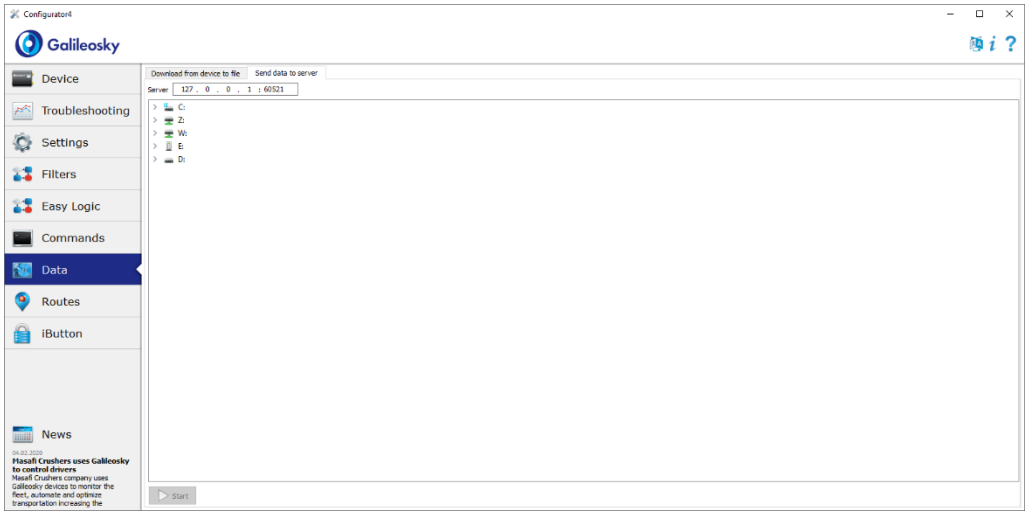
When data download from the internal memory occurs, only one file InternalFlash.csv will be created.



The Data transfer from the internal memory can be stopped and resumed.

Sending data to server

This option allows sending the data previously transferred from the tracking device to any server emulating the Galileosky protocol. To send it you should specify the IP-address and the server port and choose a file or a catalogue to be sent. If a catalogue is chosen, the program will send all its data files. The process can be stopped and resumed.



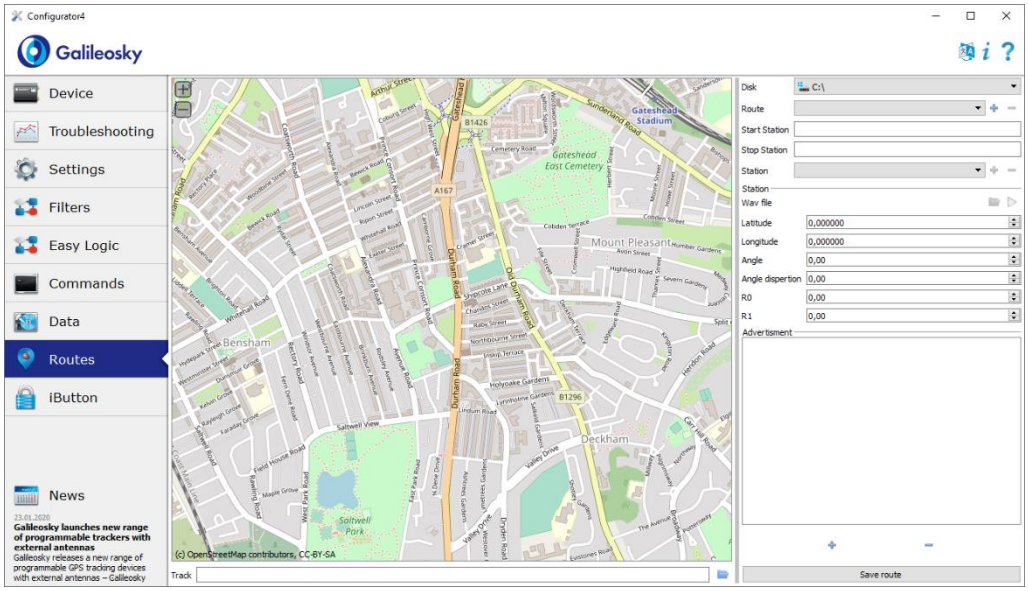
Routes for Autoinformer

This tab contains a graphical Autoinformer zones editor (section Autoinformer function).

The Internet access is required to display the map.

To edit the zones:

1) Insert a micro-SD card into the computer, choose the disk in the drop-down list. The route list and the track list will be loaded automatically.
2) Choose an edited route in the drop-down list or create a new one by clicking "+" on the right side of the route list. All stops will be loaded automatically, the map will move to the first stop.
3) Edit parameters of the zones. It can be done on the right panel as well as visually by moving red points with a mouse.
4) Create new zones. Click "+" on the right side of the stops list. A new zone will appear in the middle of the map, and then it can be edited.
5) Choose an audio file for the zone. It can be done in the Audio file drop-down list or you can load a new file by clicking "+" on the right side of the stops list. The following formats are supported: wav,mp3, flac, ogg, raw, gsm. The Configurator will automatically convert the file in appropriate format for the tracking device and copy it to the micro-SD card.
6) Press Save route to save changes to the micro-SD card.



To make sure all zones are correct, download the track developed after a trip. The Configurator supports the tracks uploaded from the internal memory of the tracking device or stored on the micro-SD card in the csv format.

Further information is provided on the page https://galileosky.com/podderzhka/dokumentacziya.html in the instruction Audio. Autoinformer function setting

Commands List

6

To request current settings, you need to issue a command without any parameters.

Settings for SMS control

Command format

AddPhone xxxx[,n]

Parameters

xxxx - is a four-digit password, 1234 by default
n – slot number (0-3) where a telephone number will be saved.

Explanation

When you configure the tracker from a cell phone, you should first authorize it by using the command. Up to 4 telephone numbers can be authorized.

Example

Request: AddPhone 1234
Reply: Phones (0) = 890101243456 (1) = (2) = (3) =

Command format

ChangePass aaaa

Parameters

aaaa - is a numeric four-digit password;

Explanation

Changing and viewing the current password.

Example

Request: ChangePass 5678
Reply: Password changed to '5678'

Command format
Phones P1,P2,P3,P4

Parameters
P1,P2,P3,P4 – authorized phone numbers in international format.

Explanation
Getting and setting the list of authorized phones

Example
Request: Phones +7901012345,,,
Reply: Phones (0)=+790101243456 (1)= (2)= (3)=

Data transmission settings

Command format
APN a,u,p

Parameters
a – access point name
u – user
p – password

Explanation
Access point settings for a SIM-card (SIM0)

Example
Request: APN internet.beeline.ru,beeline,beeline
Reply: GPRS:APN=internet.beeline.ru, user=beeline, pass=beeline

Command format
APN2 a,u,p

Parameters
a – access point name
u – user
p – password

Explanation
Access point settings for a SIM-card (SIM1) or a SIM-microchip

Example
Request: APN2 internet.beeline.ru,beeline,beeline
Reply: GPRS2:APN=internet.beeline.ru, user=beeline, pass=beeline

Command format
USER name

Parameters
name – user ID.

Explanation
ID setting for working with Exigner

Example
Request: USER test@7gis.ru
Reply: USER:test@7gis.ru;

Command format
OPS0 n1,n2,n3,n4,n5,n6,n7,n8,n9,n10,n11,n12,n13,n14,n15

Parameters
n1-n15 – preferred GSM-networks

Explanation
A list of preferred GSM- networks for a SIM-card (SIM0). The network is defined by a mobile country code and a mobile operator code (the list of codes is given in http://www.itu.int/dms_pub/itu-t/opb/sp/T-SP-E.212A-2010-PDF-E.pdf), for example, the Russian Federation code is 250.

Example
Request: OPS0 25001,25099
Reply: OPS0:25001,25099,,,,,,,,,,,,;

Command format
OPS02n16,n17,n18,n19,n20,n21,n22,n23,n24,n25,n26,n27,n28,n29,n30

Parameters
n16-n30 – preferred GSM-networks

Explanation
Additional list of preferred GSM-networks for a SIM-card (SIM0).

Example
Request: OPS02 25001,25099
Reply: OPS02:25001,25099,,,,,,,,,,,,;

Command format
OPS1 n1,n2,n3,n4,n5,n6,n7,n8,n9,n10,n11,n12,n13,n14,n15

Parameters
n1-n15 – preferred GSM-networks

Explanation
A list of preferred GSM-networks for a SIM-card (SIM1) or a SIM-microchip.

Example
Request: OPS1 25001,25099
Reply: OPS1:25001,25099,,,,,,,,,,,,;

Command format
OPS12 n16,n17,n18,n19,n20,n21,n22,n23,n24,n25,n26,n27,n28,n29,n30

Parameters
n16-n30 – preferred GSM-networks

Explanation
Additional list of preferred GSM-networks for a SIM-card (SIM1) or a SIM-microchip.

Example
Request: OPS12 25001,25099
Reply: OPS12:25001,25099,,,,,,,,,,,,;

Command format
BOPS0 n1,n2,n3,n4,n5,n6,n7,n8,n9,n10,n11,n12,n13,n14,n15

Parameters
n1-n15 – GSM-networks, in which data transfer is forbidden, GSM-module registration is allowed only if lack of preferred networks.

Explanation
The list of forbidden GSM-networks for a SIM-card (SIM 0). The network is defined by a mobile country code and a mobile operator code (the list of codes is given in http://www.itu.int/dms_pub/itu-t/opb/sp/T-SP-E.212A-2010-PDF-E.pdf), for example, Poland code is 260.

Example
Request: BOPS0 26003
Reply: BOPS0:26003,,,,,,,,,,,,;

Command format
BOPS02n16,n17,n18,n19,n20,n21,n22,n23,n24,n25,n26,n27,n28,n29,n30

Parameters
n16-n30 – GSM-networks, in which data transfer is forbidden, GSM-module registration is allowed only if lack of preferred networks.

Explanation
Additional list of forbidden GSM-networks for a SIM-card (SIM 0).

Example
Request: BOPS02 26003
Reply: BOPS02:26003,,,,,,,,,,,,,;

Command format
BOPS1 n1,n2,n3,n4,n5,n6,n7,n8,n9,n10,n11,n12,n13,n14,n15

Parameters
n1-n15 – GSM-networks, in which data transfer is forbidden, GSM-module registration is allowed only if lack of preferred networks.

Explanation
The list of forbidden GSM-networks for a SIM-card (SIM 1) or SIM chip.

Example
Request: BOPS1 26003
Reply: BOPS1:26003,,,,,,,,,,,,,;

Command format
BOPS12n16,n17,n18,n19,n20,n21,n22,n23,n24,n25,n26,n27,n28,n29,n30

Parameters
GSM-networks, in which data transfer is forbidden, GSM-module registration is allowed only if lack of preferred networks.

Explanation
Additional list of forbidden GSM-networks for a SIM-card (SIM 1) or SIM chip.

Example
Request: BOPS12 26003
Reply: BOPS12:26003,,,,,,,,,,,,,;

Command format

SIMSwitch mode

Parameters

mode – SIM-cards switching algorithm:
0 – only SIM0 is used;
1 – cyclic switching between the SIM-cards, if you cannot send the data for 9 minutes;
2 – switching according to the preferred GSM-networks list;
3 – only SIM1 or SIM-microchip is used;
5 – cyclic switching between the SIM-cards, if you cannot send the data for 9 minutes, starting with SIM1 or SIM chip;
6 – switching according to the preferred GSM-networks list, starting with SIM1 or SIM chip;
7 – repeating switching between the SIM-cards, the primary server is connected to SIM0, the secondary server is connected to SIM1 or SIM chip;
period – the periodicity of switching in mode 7

Explanation

Setting of an algorithm of switching between a SIM-card and a SIM-microchip.

Example

Request: SIMSwitch 1
Reply: SIMSwitch:1;

Command format

Serverip host,port

Parameters

host – domain name of a server or its IP-address;
port – server port.

Explanation

Main server parameters where the monitoring data will be transmitted to.

Example

Request: Serverip m.7gis.ru,60521
Reply: SERVERIP=m.7gis.ru:60521
Request: Serverip 46.146.233.216,60521
Reply: SERVERIP=46.146.233.216:60521

Command format
Serverip2 ip1,ip2,ip3,ip4,port

Parameters
host –domain name of a server or its IP-address;
port – server port.

Explanation
Additional server parameters.

Example
Request: Serverip2 m.7gis.ru,60521
Reply: Serverip2=m.7gis.ru:60521

Command format
ID n

Parameters
n - tracker number.

Explanation
Changes device number. The same number is used as the device identifier in the EGTS protocol.

Example
Request: ID 123
Reply: ID=123

Command format

Protocol s, s2

Parameters

s1 – version of data transmission protocol for the main server:
0 – Galileosky protocol;
3 – EGTS GOST R 56360-2015 is not used since 2018;
4 – Galileosky protocol with compression;
5 - EGTS GOST 33472-2015 is used since 2018.
s2 – version of data transmission protocol for the additional server:
0 – Galileosky protocol;
3 – EGTS GOST R 56360-2015 is not used since 2018;
4 – Galileosky protocol with compression;
5 - EGTS GOST 33472-2015 is used since 2018.

Explanation

Choosing a protocol of monitoring data transmission to the server

Example

Request: Protocol 0,0
Reply: PROTOCOL:0,0;

Command format

BLEPOLLINGRATE n

Parameters

n – periodicity of data sending via Bluetooth.

Explanation

Setting periodicity of data sending via Bluetooth, sec.

Example

Request: BLEPOLLINGRATE 5
Reply: BLEPOLLINGRATE:5;

Command format

WIFI SSID,Type,Key,Mode,RSSI

Parameters

SSID – Wi-Fi network ID;
Type – authentication:
0 – no authentication;
1 – WEP;
2 – WPA/WPA2;
Key – password for Wi-Fi network;
Mode – working mode of the module:
0 - client;
1 – access point;
RSSI - received signal level, -100 dBm - low received signal level, 0 dBm - indicator of an excellent received signal level); if the signal level is lower than the setting, the tracker does not connect to such a point, or, if it was connected, switches to GPRS

Explanation

Determining the parameters of connection to Wi-Fi.

Example

Request: Wifi Redmi,2,1234,0,-100
Reply: WIFI:SSID=Redmi,Type=2,Key=1234,Mode=0,RSSI=-100;

Command format

GSMLOCATION n

Parameters

n – working mode.
0 – determining the coordinates using only GPS/GLONASS-module
1 – determining the coordinates with the help of cellular stations while receiving no data from GPS/GLONASS-module

Explanation

Setting the coordinates determination mode

Example

Request: GSMLOCATION 1
Reply: GSMLOCATION:1;

Galileosky server exchange protocol settings

Command format
HeadPack bbbbbbbbbbbbb

Parameters
bbbbbbbbbbbbb - tag set from 1 to 128.
If b is replaced by 1, the tag is on.
If b is replaced by 0, the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation
Head packet configuring.

Example
Request: HeadPack 1110
Reply: HeadPack=
00000000000000000000000000000000000000000000001110b,
This means that tags from the second to the fourth inclusive are on and the first and the rest tags inclusive are off.

Command format
HeadPack2 bbbbbbbbbbbbb

Parameters
bbbbbbbbbbbbb - tag set from 129 to 256.
If b is replaced by 1, the tag is on.
If b is replaced by 0, the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation
Head packet configuring.

Example
Request: HeadPack2 110000
Reply: HeadPack= 110000b,
This means that tags 129, 130,131,132 are off and tags133 and 134 are on. All the following tags are off.

Command format
HeadPackBit index,value

Parameters
index – is tag number, which is on or off for transmission to the server
value – 1 if the tag should be transmitted to the server
0 if tag should not be transmitted to the server
Tag numeration order is given in section Galileosky protocol data.

Explanation
Head packet configuring.

Example
Initially the second tag is off:
HeadPack=1100b
Switch on this tag.
Request: HeadPackBit 2,1
Reply: HeadPack=1100b

Command format
MainPack bbbbbbbbbbbbb

Parameters
bbbbbbbbbbbbb – tag set from 1 to 128.
If b is replaced by 1, the tag is on.
If b is replaced by 0 the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation
Main packet configuring.

Example
Request: MainPack 1111111111111111111110000
Reply: MainPack= 00000000000000000000011111111111111111111110000b
This means that tags 1, 2, 3, 4 are off and tags 5-26 inclusive are on. All the following tags are off.

Command format
MainPack2 bbbbbbbbbbbbb

Parameters
bbbbbbbbbbbbb – tag set from 129 to 256.
If b is replaced by 1, the tag is on.
If b is replaced by 0 the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation
Main packet configuring.

Example
Request: MainPack2 110000
Reply: MainPack2=110000b
This means that tags 129, 130, 131, 132 are off and tags 133 and 134 are on. All the following tags are off.

Command format
MainPackBit index,value

Parameters
index –tag number, which is on or off for transmission to the server
value – 1 if this tag should be transmitted to the server
0 if this tag should not be transmitted to the server
Tag numeration order is given in section Galileosky protocol data.

Explanation
Main packet configuring.

Example
Initially the second tag is off:
HeadPack=1100b
Switch on this tag.
Request: HeadPackBit 2,1
Reply: HeadPack=1100b

Command format

BlePack bbbbbbbbbbbbb

Parameters

bbbbbbbbbbbbb – tag set from 1 to 128.
If there is 1 instead of b, the tag is on.
If there is 0 instead of b, the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation

Configuring Bluetooth-packet.

Example

Request: BlePack 1111111111111111111110000
Reply: BlePack=00000000000000000000011111111111111111111110000b
It means that 1, 2, 3, 4 tags are off, from 5 to 26 are on. All the following tags are off.

Command format

BlePack2 bbbbbbbbbbbbb

Parameters

bbbbbbbbbbbbb – tag set from 129 to 256.
If there is 1 instead of b, the tag is on.
If there is 0 instead of b, the tag is off.
Tag numeration order is given in section Galileosky protocol data.

Explanation

Configuring Bluetooth-packet.

Example

Request: BlePack2 110000
Reply: BlePack2=110000b
It means that 129, 130, 131, 132 tags are off, 133 and 134 tags are on. All the following tags are off.

Command format

BlePackBit index,value

Parameters

index – number of the tag, which will be on or off for the server sending
value – 1, if the tag is to be sent to the server,
0, , if the tag isn`t to be sent to the server.
Tag numeration order is given in section Galileosky protocol data.

Explanation

Configuring Bluetooth-packet.

Example

Initially the second tag is off:
BlePack= 1100b
Let`s make it on.
Request: BlePackBit 2,1
Reply: BlePack= 1110b

Command format

DataKey key

Parameters

Key – data encryption key in hexadecimal form, if it is equal to 0, data are not encoded.

Explanation

Specifies the key that encrypts the transmitted data.

Track parameters setting

Command format

Turning V,A,D,S,dS

Parameters

V – minimum speed that enables drawing of the track at turnings, [km/h];
A – minimum turn angle for the tracking device to record a track point, [º];
D – the distance above which the next packet will be saved to the device memory, [m];
S –the speed above which for dS-multiple value track point will be recorded, [km/h];
dS – speeding interval, [km/h].

Explanation

Configures track detail representation.

Example

Request: Turning 3,10,300,60,20
Reply: TURNING:Speed=3,Angle=10,Distance=300,SpeedEx=60,SpeedDelta=20

Command format

WrPeriod x,y

Parameters

x – Period of packet recording in memory in motion, [sec.];
y – Period of packet recording in memory when the vehicle stops, [sec.].

Explanation

Period of packets recording when the vehicle is moving or when it stops.

Example

Request: WrPeriod 60,180
Reply: WRPERIOD move=60 parking=180

Command format

GPS.Correct OnOff,MaxWrong,HDOP,Spd,Acc,Jump,TravelSpeed

Parameters

OnOff –coordinates filtering is on (1) or off (0);
MaxWrong – the number of wrong coordinates to be filtered (the recommended number is 5). This parameter accounts errors of acceleration exceed and jump, for other parameters the coordinates are always filtered;
HDOP – Maximum HDOP above which the coordinates are not updated;
Spd – Maximum speed. When it is exceeded, coordinates are considered false and are not updated, [km/h];
Acc –GPS or GLONASS data-based acceleration [m/s^2];
Jump – Maximum coordinate jump in the nearest 2 seconds, [m];
TravelSpeed – Minimum speed for coordinates to be updated, [km/h]. This function is not suitable for low speed vehicles (tractors, asphalt placing machines)

Explanation

Allows filtering false coordinates: jumps when the vehicle stops, in or out of tunnels, near high-rise buildings

Example

Request: GPS.CORRECT 1,5,2,150,3,50,3
Reply: GPS.correct: OnOff=1, MaxWrong=5, MaxHDOP=2, MaxSpd=150, MaxAcc=3, MaxJump=50, MaxTravelSpeed=3;

Command format

GPS.Correct2 MaxNoSatTime,MinSatStart,MinSatWork

Parameters

MaxNoSatTime – maximum time without satellite connection when no disconnection is registered, [sec.];
MinSatStart – minimum number of satellites to be connected to when the tracking device is on;
MinSatWork– minimum number of satellites during operation of the tracker. If the number is smaller, a disconnection will be registered.

Explanation

These settings affect coordinates updating, if filtering is on by **GPS.** Correct command.

Example

Request: GPS.CORRECT2 10,5,4
Reply: GPS.correct2:MaxNoSatTime=10,MinSatStart=4,MinSatWork=3;

Command format

AccSens Sens,TO

Parameters

Sens – accelerometer sensitivity.
TO – the time after the vehicle stops, during which coordinates are updated, [sec].

Explanation

This function allows avoiding unnecessary outliers after the vehicle stops.
Default value is 40,300.
Sens value equal to 600 is 1g (g –gravitational acceleration)

Example

Request: AccSens 40,300
Reply: Accelerometer sensitive: sens = 40, time out=300

Command format

Ignition N

Parameters

N – an input used as an ignition sensor:
0 – ignition sensor is not used;
1 – input 0 is used as ignition sensor;
2 – input 1 is used as ignition sensor;
3 – input 2 is used as ignition sensor;
4 – input 3 is used as ignition sensor;
5 – input 4 is used as ignition sensor;
6 – input 5 is used as ignition sensor;

Explanation

If there is no response for a given input, vehicle is considered to be not started, and coordinates are not updated. It allows avoiding outliers after the vehicle stops. Triggering on input is determined by the limits set by InCfg command (section Analog- discrete inputs settings).

Example

Request: Ignition1
Reply: IGNITION:1;

Command format

Shock Mode,Angle,Timeout,ShockSens

Parameters

Mode – strike determination mode:
0 – strike determination is switched off;
1 – strike determination is switched on; X axis is in vertical position;
2 – strike determination is switched on; Y axis is in vertical position;
3 – strike determination is switched on; Z axis is in vertical position;
Angle – maximum incline angle [0º-180º], value equal to 180 switches off incline determination;
Timeout – maximum allowable time when incline angle is exceeded, [sec.].
ShockSens – maximum acceleration by exceed of which a strike is detected. 600 points – gravitational acceleration.
ShockDuration – duration of incline, in case of its excess strike it detected. Default values – 2. It is measured in conditional units.

Explanation

Switching on strike and incline determination mode.

Example

Request: Shock 3,30,5,600,3
Reply: SHOCK:Mode=3,MaxAngle=30,RT=5,Sens=600,Duration=3;

Command format

Mhours LoLevel,HiLevel

Parameters

LoLevel – input voltage + supply voltage by stopped engine, [mV];
HiLevel – input voltage +supply voltage by started engine, [mV];

Explanation

Allows filtering false coordinates after the vehicle stops

Example

Request: mHours 12000,14500
Reply: Mclock: lolevel=12000, hilevel=14500;

Information commands

Command format

Status

Explanation

Allows receiving device status at the moment of sending a command.
Dev – this device number;
Soft – current firmware version;
Pack – last recorded packet serial number;
TmDt – Current time and date;
Per – Current packet saving period (different when the vehicle is moving and stops);
Nav – Coordinates determination accuracy. 0 – coordinates are found.
Lat – Latitude;
Lon – Longitude;
Speed – Linear speed (vehicle speed);
HDOP – Horizontal accuracy (The closer to 1, the better);
SatCnt – Number of available satellites;
A – movement directional angle

Example

Request: Status
Reply:Dev50 Soft=91 Pack=17230 TmDt=10:58:6 20.6.9 Per=60 Nav=0 Lat=60.4007 Lon=31.0070 Speed=0.0194 HDOP=0.88 SatCnt=10 A=27.55

Command format

imei

Explanation

Allows obtaining a unique GSM/3G-unit identifier,15byte and CCID of SIM-microchip, if there is one.

Example

Request: IMEI
Reply: IMEI 123456789012345, 12345678901234567890

Command format

imsi

Explanation

Allows obtaining a unique IMSI identifier of a SIM-card

Example

Request: IMSI
Reply: IMSI: 123456789012345;

Command format
inall

Explanation
Allows analog input values in0..in3 as well as digital fuel level sensor values and temperature, and accelerometer values with respect to three axes (10 bit for each axis starting with the zero bit) being obtained.

Example
Request: inall
Reply: INALL:in0=0,in1=0,in2=0,in3=0, RS232=10,1,Acc=332943891;

Command format
Insys

Explanation
Allows obtaining external source voltage, internal battery voltage, GPS aerial voltage, the main power bus voltage of the tracking device and also the temperature inside the tracking device.

Example
Request: insys
Reply: INSYS: Pow=12438, Vbat=4196,Vant=2921,Vdc=4115,Temper=37

Command format
RS485

Explanation
Allows receiving a value of the digital fuel level sensor connected via the RS485 interface. For each sensor the command prints fuel level and temperature values

Example
Request: RS485
Reply:
RS485 100,0;100,1;100,2;100,3;100,4;100,5;100,6;100,7;100,8;100,9;100,10;
100,11;100,12;100,13;100,14;100,15;

Command format
Temex0

Explanation
Allows finding temperature from the first four external DS18S20 thermometers. Format: the lower byte is the thermometer identifier; the higher byte is the temperature itself. To calculate the temperature the obtained value must be divided by 256 and rounded off to the whole number, with the fractional part deleted.

Example
Request: temex0
Reply: TemEx0: DS0=0,DS1=0,DS2=0,DS3=0

Command format
Temex1

Explanation
Allows finding temperature of the second four external DS18S20 thermometers. Format: the lower byte is the thermometer identifier; the higher byte is the temperature itself. To calculate the temperature the obtained value must be divided by 256 and rounded off to the whole number, with the fractional part deleted.

Example
Request: temex1
Reply: TemEx1: DS4=0,DS5=0,DS6=0,DS7=0

Command format
Canibut

Explanation
Allows obtaining current CAN-bus state and iButton decimal value

Example
Request: canibut
Reply: CAN_Ib: CANA0=0,CANA1=0,CANB0=0,CANB1=0,iBut=0

Command format
statall

Explanation
Allows obtaining device, inputs, outputs decimal status and mileage according to GPS/GLONASS data.

Example
Request: statall
Reply: StatAll: Dev=1,Ins=2,Outs=7,Mileage=152;

Service commands

Command format
PIN N

Parameters
N – four-digit PIN-code of SIM-cards.

Explanation
SIM-card PIN-code and password setting for settings access in the Configurator. The default PIN-code is 0. If you enter the wrong code, the tracking device will be blocked for 25 seconds, and then reset. PIN-code is identical for both SIM-cards.

Example
Request: PIN 1234
Reply: PIN:1234;

Command format
Archive type

Parameters
type – data source for sending to the server:
0 – archive from the internal flash-memory;
1 – archive from the microSD card.

Explanation
Selection of data source for sending to the server. After command execution it is necessary to reset the tracker. Before you select a microSD card, delete the archive, created by earlier firmwares, (EraseTrackSD or delete files from Track catalogue through the Card-Reader).

Example
Request: ARCHIVE 0
Reply: ARCHIVE:0;

Command format
FlashArchive SendOrder

Parameters
SendOrder – order of data transmission from archive to server:
0 – the data are sent deep into the archives; the most current data are sent first, then the oldest ones.
1 – the data are sent in chronological order.

Explanation
Setting of the data transmission order to the server
After changing the order of sending, it is necessary to restart the tracker.

Example
Request: FLASHARCHIVE 1
Reply: FLASHARCHIVE: StraightSendOrder=1;

Command format
Efs begin,end

Parameters
Begin – start date of uploaded period in the format DDMMYY[HH[MM]],
End – end date of uploaded period in the format DDMMYY[HH[MM]],
Where DD is a day, MM – a month, YY – a year, HH – hours, MM – minutes.
If hours and/or minutes are not specified, the following time will be used: 00 hours 00 minutes for the start date, 23 hours 59 minutes for the end date.

Explanation
Upload data from the SD memory to the server for a specified period

Example
Request: EFS 010117,01011712
Reply: EFS: **Uploading of archive has been scheduled**

Command format
EraseCfg

Explanation
Setting default configuration.

Example
Request: EraseCfg
Reply: ERASECFG

Command format
EraseTrack

Explanation
Deleting all tracks from the memory.

Example
Request: EraseTrack
Reply: ERASETRACK

Command format
EraseTrackSD

Explanation
Deleting all tracks from the SD memory.

Example
Request: EraseTrackSD
Reply: ERASETRACKSD

Command format
ColdStart

Explanation
GLONASS unit cold start.

Example
Request: ColdStart
Reply: GLONASS cold start

Command format
LED LED

Parameters
LED – coefficient of diode intensity of device operation modes: 0 – full absence of brightness, 100 – maximum brightness

Explanation
Installation of light intensity of light-emitting diode of device operation modes

Example
Request: LED 60
Reply: LED:LED=60

Command format
Reset

Explanation
Allows resetting the device remotely.

Example
Request: Reset
Reply: Reset of device. Please wait 15 seconds...

Command format
Upgrade N

Parameters
Firmware upgrading up to the specified one. If 0 is specified, the tracking device will upgrade firmware up to the last stable one.

Explanation
Firmware upgrading up to the specified one.

Example
Request: Upgrade 47
Reply: UPGRADE SOFT= 47

Command format
UpgradeStatus

Explanation
Showing the process of upgrading.

Example
Request: UPGRADESTATUS
Reply: UPGRADESTATUS: Processing: 42987/397615 (10 %)

Command format
Rollback

Explanation
Restoring the previous version of the firmware (if it is present in the tracker`s memory). After the command is performed, the tracker resets.

Example
Request: rollback
Reply: ROLLBACK: No more stable firmware to rollback

Command format
RemoteConfig OnOff

Parameters
OnOff – turning on the remote configuration function:
0 – remote configuration is off;
1 – remote configuration is on.

Explanation
Turns on and off the remote configuration (section Remote configuration).

Example
Request: RemoteConfig 1
Reply: REMOTECONFIG:1;

Command format
FormatSD key

Parameters
key – the code to format microSD-card

Explanation
Formats microSD-card. It is performed in two stages: at first FormatSD is sent without any parameter. The reply is the confirmation code. After that, the command is sent one more time with the code.

Example
Request: formatsd
Reply: FORMATSD: format key: 6309
Request: formatsd 6309
Reply: FORMATSD: scheduled

Command format
Fslist path,page

Parameters
path – the route to view the content of microSD-card. (/ - the view of the root catalogue)
page – page number to view the parts if there are many files.

Explanation
Remote viewing of microSD-card content.

Example
Request: fslist /
Reply: FSLIST: Content of /
D 061119 153012 0 logs
F 061119 153600 1 route_name
D 301119 120726 0 Arc
Part 1/1

Request: fslist /arc/collector/F4B85E022926
Reply: FSLIST: Content of /arc/collector/F4B85E022926
F 301119 120726 28 1.EAR
F 301119 123506 127696 2.EAR
F 301119 123542 130531 3.EAR
F 301119 123608 131020 4.EAR
F 301119 123634 129187 5.EAR
F 301119 123710 131032 6.EAR
Part 1/1

Command format
Setmileage N

Parameters
N – mileage in meters

Explanation
Setting the initial value of the mileage for counting with the help of GPS/GLONASS

Example
Request: setmileage 123450
Reply: SETMILEAGE 123450

Voice communication settings

Command format
GSMVolume k,m

Parameters
k – GSM-channel sound gain [1÷100].
m – GSM-channel microphone gain [1÷15].
The greater the parameter, the greater the gain.

Explanation
Allows speakerphone sound gain parameters being customized.

Example
Request: GSMVolume 75,15
Reply: GSMVOLUME=75,15

Command format
AutoAnswer n

Parameters
n – the number of calls before autoanswer. [0÷10]
If the parameter is equal to 0, the function is off.

Explanation
Incoming call results in the tracking device automatic answer.

Example
Request: AutoAnswer 1
Reply:: AUTOANSWER=1

Command format
Calls N

Parameters
N – the number of call attempts

Example
Request: Call 3
Reply: CALL:3;

Command format
RingTo N

Parameters
N –a telephone number.

Explanation
Making a call from the tracker to the given telephone number.

Example
Request: RingTo 89119988899
Reply: RINGTO=89119988899

Command format
SendSMS Tel,Msg

Parameters
Tel – a telephone number to which SMS is sent
Msg – SMS template. It may contain parameters to insert current data:
%IMEI – device IMEI, %LAT – latitude, %LON – longitude.

Explanation
Sending SMS to the given telephone number

Example
Request: SendSMS 89119988899, Test
Reply: SMS sheduled

Command format
Tangenta OnOff

Parameters
OnOff – use a push-to-talk;
0 – a speaker and a microphone are connected;
1 – a push-to-talk KMC-25 is connected.

Explanation
Voice communication setting

Example
Request: Tangenta 1
Reply: TANGENTA:1;

Analog-discrete inputs setting

Command format

InCfg_num_in ft,fl,up_low,up_hi,down_low,down_hi,imp_null

Parameters

num_in – an input number, beginning from 0;
ft – filter type
- 0 – mean value computation;
- 1 – pulse count;
- 2 – frequency count
- 4 – Wiegand26 Data0
- 5 – Wiegand26 Data1

fl – filter length. It is used for average and discrete signal function;
up_low – lower limit of a discrete signal triggering, [mV];
up_hi – upper limit of a discrete signal triggering, [mV];
down_low – lower limit of a discrete signal failure, [mV];
down_hi – upper limit of a discrete signal failure, [mV];
imp_null – pulses counter behavior: 1 – counter is set to zero, 0 – counter continues increasing.

Explanation

Allows one of 4 analog/discrete inputs being configured.

Example

Request: InCfg0 0,10,8000,15000,0,3000,0
Reply:
INCFG0:FiltType=0,FiltLen=10,UpLow=8000,UpHi=15000,DownLow=0,DownHi=3000,ImpNull=0;

Command format

PowInCfg fl,up_low,up_hi,down_low,down_hi

Parameters

fl – average length of the filter [1÷50];
up_low – lower limit of a discrete signal triggering, [mV];
up_hi – upper limit of a discrete signal triggering, [mV];
down_low – lower limit of a discrete signal failure, [mV];
down_hi – upper limit of a discrete signal failure, [mV];.

Explanation

Allows configuring the operating limits for the external power input

Example

Request:
PowInCfg 10,8000,15000,0,3000
Reply: POWINCFG:FiltLen=10,UpLow=8000,UpHi=15000,DownLow=0,DownHi=3000

Transistor outputs setting

Command format

Out v,s

Parameters

v – output ordinal number (starting with the zero output);
s – desired state (0 – on-state transistor output;
1 –off-state transistor output).

Explanation

Transistor output control.
With one output being controlled, the others outputs' state remains unchanged.
Transistor outputs are off by default.

Example

Request: Out 1,1
Reply: OUT(1..0) = 10
As you can see, all outputs except 1 are on.

Autoinformer setting

Command format

Autoinformer OnOff,Repeat,Out,FileName,Gain

Parameters

OnOff – enable/disable Autoinformer function: 1 – the function is enabled, 0 – the black box function is enabled (the navigation data are duplicated and saved to the external microSD card).
Repeat – determines whether the file should be replayed when the device is in the playback zone. If the value is 0, the file is played only once on entering the zone.
Out – output index, which is inverted during the audio file playing. 0 – no output is inverted, 1 – OUT0, 2 – OUT1, 3 – OUT2, 4 – OUT3, 5 – OUT4.
FileName – a route name. The route is understood as the number of zones to be announced.
Gain- sound gain

Explanation

For more information, see section Autoinformer.

Example

Request: Autoinformer 1,0,0,Marshrut 1,50
Reply: AUTOINFORMER:OnOff=1,Repeat=0,Route=Marshrut 1,Gain=50;.

Serial ports settings

Command format

RS2320 nf

Parameters

nf – function number

- 0 – no function on RS232;
- 1 – digital fuel level sensor (relative level N);
- 2 - digital fuel level sensor (frequency F);
- 3 – external GLONASS unit;
- 4 – Galileo photo camera
- 5 – Garmin navigator
- 6 – CAN-LOG
- 7 – CUB5 indicator
- 9 – RS232-RS485 adapter;
- 10 – REP-500 energy meter;
- 11 – Carrier DataCOLD500 temperature recorder;
- 12 – CI-5010A weight indicator;
- 13 – PressurePro;
- 14 – ThermoKing temperature recorder;
- 15 – EuroScan temperature recorder;
- 16 – Tenzo-M weight indicator;
- 17 – test of port performance;
- 18 – AWT 640 weight indicator.

Explanation

RS232 port function setting.

Example

Request: RS2320 1
Reply: RS232_0: NumFunc=1;

Command format

CUB5 N0

Parameters

N0 – parameter number displayed on the indicator connected to RS232 port.
Parameters:
0 – speed, accurate within 0.1, [km/h];
1 – directional angle accurate within 0.1 [°];
2 – external supply voltage [mV];
3 – internal accumulator voltage [mV];
4 – temperature inside the tracking device [°C];
5 – mileage according to the GPS/GLONASS data, accurate within 0.1, [km]
6 – input IN0;
7 – input IN1;
8 – input IN2;
9 – input IN3;
10 – RS232[0];
12 – temperature sensor 0 [°C];
13 – temperature sensor 1 [°C];
14 – temperature sensor 2 [°C];
15 – temperature sensor 3 [°C];
16 – temperature sensor 4 [°C];
17 – temperature sensor 5 [°C];
18 – temperature sensor 6 [°C];
19 – temperature sensor 7 [°C];
20 – CAN. Total fuel consumption [l]
21 – CAN. Tank fuel level, accurate within 0.1 [%]
22 – CAN. Coolant temperature [°C];
23 – CAN. Engine speed;
24 – CAN. mileage, accurate within 0.1 [km]
25-39 – CAN8BITR0 - CAN8BITR14;
40-44 – CAN16BITR0 – CAN16BITR4;
45-49 – CAN32BITR0 – CAN32BITR4;
50 – RS485[0];
51 – RS485[1];
52 – RS485[2].

Explanation

Settings of parameters display on the CUB5 indicator.

Example

Request: CUB5 1
Reply: CUB5:RS2320=1, RS2321=0;

Command format

DFilter RS2320

Parameters

RS2320 – filter length for a fuel level sensor connected to RS232 and RS485 ports, the number of successive sensor indications, which are averaged. If the value is equal to 1, there is no filtration.

Explanation

Digital fuel sensors indications filtering.

Example

Request: DFILTER 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1

Reply:

DFILTER:RS232_0=1,RS232_1=1,RS485_0=1,RS485_1=1,RS485_2=1,RS485_3=1,RS485_4=1,RS485_5=1,RS485_6=1,RS485_7=1,RS485_8=1,RS485_9=1,RS485_10=1,RS485_11=1,RS485_12=1,RS485_13=1,RS485_14=1,RS485_15=1;

Command format

iButtons ib1,ib2,ib3,ib4,ib5,ib6,ib7,ib8

Parameters

Ib1-ib8 – four lower bytes of iButton identification hexadecimal number without the checksum.

For example, full key hexadecimal number:

09 91 02 0C 00 00 00 5C, where

09 – type of device (in this case, it is DS1982, for DS1990 – 01),

91 02 0C 00 00 00– unique number,

5C – the checksum.

In this case, 91 02 0C 00 must be entered.

Explanation

List of iButton identifiers, the connection state of which is monitored by the tracking device.

Example

Request: iButtons 0091022C,0,0,0,0,0,0,0

Reply: IBUTTONS:0091022C,0,0,0,0,0,0,0

Command format

AddKey key1,...,key25

Parameters

key – the four low-order bytes of iButton ID number excluding checksum in the hexadecimal form
e.g. the full number of the key in the hexadecimal form:
09 91 02 0C 00 00 00 5C, where
09 – type of the device (now it`s DS1982, for DS1990 it will be 01),
91 02 0C 00 00 00 – unique number,
5C – checksum.
In this case we need to enter 91 02 0C 00.

Explanation

Adding the keys to the list of trusted iButton ID on microSD-card.

Example

Request: AddKey 0091022C,0091022D
Reply: Added 2 keys

Command format

DelKey key1,...,key25

Parameters

key – the four low-order bytes of iButton ID number excluding checksum in the hexadecimal form
e.g. the full number of the key in the hexadecimal form:
09 91 02 0C 00 00 00 5C, where
09 – type of the device (now it`s DS1982, for DS1990 it will be 01),
91 02 0C 00 00 00 – unique number,
5C – checksum.
In this case we need to enter 91 02 0C 00.

Explanation

Deleting the keys to the list of trusted iButton ID on microSD-card.

Example

Request: DelKey 0091022C,0091022D
Reply: Deleted 2 keys

Command format
KeyCount

Explanation
The number of trusted keys on microSD-card.

Example
Request: KeyCount
Reply: KEYCOUNT:12;

Command format
ShowKey N

Parameters
N – ordinal number of the iButton key in the list on microSD-card. Numbering from 1.

Explanation
Showing the iButton key ID on microSD-card.

Example
Request: ShowKey 1
Reply: SHOWKEY 1:9503276 (0x0091022C)

Command format
RS485FN nf

Parameters
nf – function number:
- 0 – fuel level sensors, DBG-S11D, PressurePro gateway device;
- 1 – RFID reader MATRIX 5;
- 2 – fuel level sensors, Galileosky camera (current serial production), DBG-S11D, PressurePro gateway device.

Explanation
RS485 port settings.

Example
Request: RS485FN 1
Reply: RS485FN:1;

Command format

IBCFG T1,T2

Parameters

T1 – timeout from iButton detachment to setting to zero of the key code in the memory cell of the tracker, for keys with the code of more or equal to 100000;
T2 – timeout from iButton detachment to setting to zero of the key code in the device`s memory, for keys with the code of less than 100000.

Explanation

Periods settings of iButton keys codes setting to zero.

Example

Request: IBCFG 4,1500
Reply: IBCFG:Timeout=4,Timeout100000=1500;

CAN settings

Command format

CanRegime Mode,BaudRate,TimeOut,DoNotCleanAfterTimeOut

Parameters

Mode – operating mode:

- 0 – CAN-interface is off and is not used;
- 2 – standard FMS filter (FMS mode);
- 3 – J1939 standart filter;
- 5 – J1979 standat filter 29bit (J1979_29bit mode);
- 6 – J1979 standat filter11bit (J1979_11bit mode);
- 10 – J1979 search for responded identifiers (J1979_CHECK_PIDS mode);
- 11 – Easy Logic

BaudRate – data bus rate. It must be the same as the vehicle data bus rate. It can have the following values: from 10000 up to 500000. Typical values: 62500, 125000, 250000, 500000.

TimeOut – measured in msec. For CAN_SCANNER mode it is response latency. If it is too small, not all bus messages will be received. The recommended time for CAN_SCANNER is 2000 msec. For all the rest modes it is time to receive at least one message, otherwise, the value will be set to zero.

DoNotCleanAfterTimeOut – data should not be set to zero by disconnecting CAN-bus.

Explanation

General CAN-bus control for port 0.

Example

Example: switching on scanner for a 250000 b/sec bus with the message (answer) latency, equal to 2 sec.

Request: CanRegime 1,250000,2000

Reply:CANREG:Mode=1,BaudRate=250000,TimeOut=2000;
DoNotCleanAfterTimeOut=0;

Command format

Can1Regime Mode,BaudRate,TimeOut,DoNotCleanAfterTimeOut

Parameters

Mode – operating mode:

- 0 – CAN-interface is off and is not used;
- 2 – standard FMS filter (FMS mode);
- 3 – J1939 standart filter;
- 5 – J1979 standat filter 29bit (J1979_29bit mode);
- 6 – J1979 standat filter11bit (J1979_11bit mode);
- 10 – J1979 search for responded identifiers (J1979_CHECK_PIDS mode);
- 11 – Easy Logic

BaudRate – data bus rate. It must be the same as the vehicle data bus rate. It can have the following values: from 10000 up to 500000. Typical values: 62500, 125000, 250000, 500000.

TimeOut – measured in msec. For CAN_SCANNER mode it is response latency. If it is too small, not all bus messages will be received. The recommended time for CAN_SCANNER is 2000 msec. For all the rest modes it is time to receive at least one message, otherwise, the value will be set to zero.

DoNotCleanAfterTimeOut – data should not be set to zero by disconnecting CAN-bus.

Explanation

General CAN-bus control for port 1.

Example

Example: switching on scanner for a 250000 b/sec bus with the message (answer) latency, equal to 2 sec.

Request: CanRegime 1,250000,2000

Reply:CANREG:Mode=1,BaudRate=250000,TimeOut=2000;
DoNotCleanAfterTimeOut=0;

Command format

ActiveCAN OnOff

Parameters

OnOff – operating mode:
0 – passive mode: packets receiving confirmations are not sent to the CAN-bus. It is a safe mode of operation. It does not interfere with the on-board equipment;
1 – active mode: packets receiving confirmations are sent to the CAN-bus.

Explanation

Control of packets confirmation sending to the CAN-bus for port 0.
Confirmation sending may be necessary by connection to the troubleshooting socket if the data cannot be read in passive mode.

Example

Request: ActiveCAN 1
Reply: ACTIVECAN:1;

Command format

ActiveCAN1 OnOff

Parameters

OnOff – operating mode:
0 – passive mode: packets receiving confirmations are not sent to the CAN-bus. It is a safe mode of operation. It does not interfere with the on-board equipment;
1 – active mode: packets receiving confirmations are sent to the CAN-bus.

Explanation

Control of packets confirmation sending to the CAN-bus for port 1.
Confirmation sending may be necessary by connection to the troubleshooting socket if the data cannot be read in passive mode.

Example

Request: ActiveCAN1 1
Reply: ACTIVECAN1:1;

Command format

CAN8BitR0 ID, Shift, BigEndian

Parameters

ID – captured bus identifier:
Shift – useful data shift in the received packet
BigEndian – byte order, 0 – from the lower byte to the higher, 1 – from the higher to the lower.

Explanation

Single CAN-tag content control.

Example

Request: Can8BitR0 419360256,1,0
Reply: CAN8BITR0:ID=419360256,BigEndian=0;

Commands: **CAN8BitR1, ..., CAN8BitR30, CAN16BitR0, ..., CAN16BitR14, CAN32BitR0, ..., CAN32BitR14** are similar to CAN8BitR0 command.

Photo camera operation settings

Command format

GetPhoto d,t,n

Parameters

d – photo date, format DDMMYY, where DD – day, MM – month, YY – year;
t – photo time, format HHMMSS, where HH – hours, MM – minutes, SS – seconds;
n – port number to which the camera taking photos is connected:
0 – RS232[0],
1 – RS232[1]
2 – RS485.

Explanation

Request to transmit the nearest to the given time and data photo to the server.

Example

Request: GetPhoto 050511,052030,0
Reply: Send of photo is scheduled

Command format

MakePhoto

Explanation

Take a photo and send it to the server

Example

Request: MakePhoto
Reply: Photo ok

Command format

PhotoCfg t1,t2,mode,res,confirm

Parameters

t1 –periodical shooting interval, [sec]. Photos are saved only to the SD-card, 0 – shooting only by event;
t2 – shooting interval [sec]. Photos are saved to the SD-card and sent to the server, 0 – shooting only by event;
mode – periodical shooting in geofences:
0 – photos are taken regardless of geofences;
1 – photos are taken only inside geofences;
2 – photos are taken only outside geofences.
res – picture resolution:
0 – 640x480 points;
1 – 320x240 points.
confirm – waiting for a confirmation of a picture reception from the server:
0 – do not wait;
1 – wait.

Explanation

Settings of a periodical camera shooting, picture format and image transfer protocol..

Example

Request: PhotoCfg 5,150,0,0,0
Reply: PHOTOCFG:WrPeriod=5,SendPeriod=150,Type=0,Size=0,Confirm=0;

Command format

CleanPhotoQueue n

Parameters

n – port number to which the camera taking photos is connected:
0 – RS232[0],
1 – RS232[1]
2 – RS485.

Explanation

Mark all photos as sent.

Example

Request: CleanPhotoQueue 0
Reply: Photo queue is cleaned

Bootloader

7

The processor program (firmware) is a set of algorithms developed by RSA "Galileosky", LLC specialists. Owing to this program, the central processor receives the data from different system units, processes them logically and mathematically and takes decisions for control commands of controller units to be worked out depending on the situation.

Bootloader is a device`s sub-program allowing the main program part being updated.

The main program can be downloaded via the USB or GPRS channel in the tracking device.

USB Channel Download

Install Configurator for updating via USB-channel, after connecting the tracking device, select "Upgrade firmware".

GPRS channel download

1. Connect the tracking device to the external power supply;
2. APN settings must conform to the SIM-card inserted in the tracker, otherwise, the device flashing will not happen, and the tracking device will return to the operating mode;

Give the following command: UPGRADE firmware№ using any of the available channels (SMS, GPRS, USB), where firmware№ is the necessary firmware version. UPGRADE 0 initiates downloading the latest firmware;

3. You may see if the flashing is in progress by LEDs blinking;
4. In 15-25 minutes (depending on connection conditions and GPRS terms of service by operator) upgrade will be completed, and the tracking device will turn into operation mode automatically.

Using analog inputs to enter bootloader mode

After the device power supply is off, energize all analog-discrete inputs (section Contacts description) by applying the voltage of 7.0V ± 0.2V until the tracking device enters the bootloader mode.

This function is used only during an improper device flashing. An improper firmware is the firmware designed for tracking devices with the other functional.

Galileosky Protocol Data

8

Bit number is entered in mainpackbit and headpackbit commands for selecting parameters, transmitted to the server.

Bit №	Description
1	Hardware version
2	Firmware version
3	IMEI
4	Device's identifier
5	Number of an archive record
6	Greenwich date and time
7	Coordinates in degrees, number of satellites, indication of coordinates determination correctness
8	Speed in km/h and direction in degrees
9	Height, m
10	HDOP
11	Status of device
12	Supply voltage, mV
13	Battery voltage, mV
14	Device temperature, ºC
15	Acceleration
16	Status of outputs
17	Status of inputs
18	Input voltage 0, mV Depending on settings: 1. voltage, mV; 2. number of impulses; 3. frequency,Hz.
19	Input voltage 1, mV Depending on settings: 1. voltage, mV; 2. number of impulses; 3. frequency,Hz.

Bit №	Description
20	Input voltage 2, mV Depending on settings: 1. voltage, mV; 2. number of impulses; 3. frequency,Hz.
21	Input voltage 3, mV Depending on settings: 1. voltage, mV; 2. number of impulses; 3. frequency,Hz.
22	RS232 0
23	RS232 1
24	Thermometer 0 identifier and measured temperature, ºC
25	Thermometer 1 identifier and measured temperature, °C
26	Thermometer 2 identifier and measured temperature, °C
27	Thermometer 3 identifier and measured temperature, °C
28	Thermometer 4 identifier and measured temperature, °C
29	Thermometer 5 identifier and measured temperature, °C
30	Thermometer 6 identifier and measured temperature, °C
31	Thermometer 7 identifier and measured temperature, °C
32	iButton first key identification number
33	CAN-bus (CAN_A0) and CAN-LOG data. Fuel used by a vehicle from the date of manufacturing, l.
34	CAN-bus (CAN_A1) and CAN-LOG data; Fuel level, %; Temperature of coolant °C; Engine speed, rotations/min.
35	CAN-bus (CAN_B0) and CAN-LOG data. Vehicle`s mileage, m.
36	CAN_B1
37	CAN8BITR0 Or vehicle speed from CAN-LOG,km/h
38	CAN8BITR1 or the 3rd byte of prefix S CAN-LOG
39	CAN8BITR2 or the 2nd byte of prefix S CAN-LOG
40	CAN8BITR3 or lower byte of prefix S CAN-LOG
41	CAN8BITR4 or the 3rd byte of prefix P CAN-LOG
42	CAN8BITR5 or the 2nd byte of prefix P CAN-LOG

Bit №	Description
43	CAN8BITR6 or the 1st byte of prefix P CAN-LOG
44	CAN8BITR7 or lower byte of prefix P CAN-LOG
45	CAN8BITR8 or higher byte of prefix WA CAN-LOG
46	CAN8BITR9 or 4th byte of prefix WA CAN-LOG
47	CAN8BITR10 or 3rd byte of prefix WA CAN-LOG
48	CAN8BITR11 or 2nd byte of prefix WA CAN-LOG
49	CAN8BITR12
50	CAN8BITR13
51	CAN8BITR14
52	Second iButton key identification number
53	Total mileage according to GPS/GLONASS units, m.
54	State of iButton keys, identifiers of which are set by iButtons command.
55	Depending on settings: 1. CAN16BITR0 2. the 1st vehicle`s axle load, kg 3. failure code OBD II
56	Depending on settings: 1. CAN16BITR1 2. the 2nd vehicle`s axle load, kg 3. failure code OBD II
57	Depending on settings: 1. CAN16BITR2 2. the 3rd vehicle`s axle load, kg 3. failure code OBD II
58	Depending on settings: 1. CAN16BITR3 2. the 4st vehicle`s axle load, kg 3. failure code OBD II
59	Depending on settings: 1. CAN16BITR4 2. the 5th vehicles axle load, kg 3. failure code OBD II
60	Depending on settings: 1. CAN32BITR0 2. total time of engine operation, h.

Bit №	Description
61	Depending on settings: 1. CAN32BITR1 2. CAN-LOG, R prefix, fuel level, l
62	Depending on settings: 1. CAN32BITR2 2. CAN-LOG, user`s prefix
63	Depending on settings: 1. CAN32BITR3 2. CAN-LOG, user`s prefix
64	Depending on settings: 1. CAN32BITR4 2. CAN-LOG, user`s prefix
77	RS485. A value from fuel level sensor with address 0. Relative fuel level.
78	RS485. A value from fuel level sensor with address 1. Relative fuel level.
79	RS485. A value from fuel level sensor with address 2. Relative fuel level.
80	RS485. Fuel level sensor with address 3. Relative fuel level and temperature.
RS485. Fuel level sensor with addresses 4-14, with numbers 81-91. Relative fuel level and temperature.	
92	RS485. FLS with address 15. Relative fuel level and temperature.
93	Extended data RS232[0]. Depending on settings: 1. Temperature from fuel level sensors connected to zero RS232 port, °C. 2. Weight received from weight indicator.
94	Extended data RS232[1]. Depending on settings: 1. Temperature from fuel level sensors connected to zero RS232 port, °C. 2. Weight received from weight indicator.
95	RS485. Temperature from fuel level sensor with address 0, °C.
96	RS485. Temperature from fuel level sensor with address 1, °C.
97	RS485. Temperature from fuel level sensor with address 2, °C.
129	CAN8BITR15
Tags CAN8BITR16 - CAN8BITR29 similar to CAN8BITR16 with numbers 130-143	
144	CAN8BITR30
145	CAN16BITR5
Tags CAN16BITR6 – CAN16BITR13 similar to CAN16BITR5 with numbers 146-153	
154	CAN16BITR14

Bit №	Description
161	CAN32BITR5
Tags CAN32BITR6 – CAN32BITR13 similar to CAN32BITR5 with numbers 162-169	
170	CAN32BITR14
171	REP-500 electricity meter readings
173	Refrigeration unit data
174	EcoDrive and driving style determination
175	PressurePro tires pressure monitoring system, 34 sensors
177	User 0 data
Tags from users data with numbers 178-183	
184	User 7 data
185	User data array

Example1.

The tracking device should be configured so that the head pack (HeadPack) contains information about the device version (HardVersion), firmware version (SoftVersion), unique 15-digit GSM/3G-unit identifier

(IMEI), device user ID (ID device).

Correspondent tag mask: 00000000000000000000000000001111.

To apply the settings, we should use the following command

HeadPack 00000000000000000000000000001111, or omitting zeros, HeadPack 1111

Example2.

It is necessary to configure the main packet (sent in normal mode) so that the tracker`s user ID (ID device), packet number (NumberOfPacket), date and time of packet record (TimeDate), coordinates are sent. Correspondent tag mask: 00000000000000000000000001111000

To apply the settings, we should use the command: MainPack 1111000.

In this case, we have omitted zeros at once.

Additional Information

9

Certifying

The tracking device is certified to comply with GOST R.

Warranty

RSA "Galileosky", LLC hereby guarantees the realization of consumers' rights provided by the local laws throughout Russia and the CIS.

RSA "Galileosky", LLC guarantees the operability of the tracking device subject to compliance with the instructions set out in the above user`s manual.

Warranty conditions

The warranty period is 36 months since the day of purchase.

Note: a defective tracker (with cracks and fissures, dents and impact marks and etc.) due to consumer's fault resulting from inappropriate maintenance, storage and transportation is not liable to warranty.

The above also holds for a device without the body or battery.

In case the guarantee document proving the device sale to the customer does not contain the date of purchase, the name and seller's seal, the warranty period starts since the day of production.

The consumer has the right for free maintenance in the manufacturer's service center if a production or design defect appeared during the warranty period. The consumer has the right for maintenance during the whole period of operation of the device. The consumer has all the other rights provided by the laws of the Russian Federation and the CIS.

If the failure cause cannot be found at the moment of appeal, a technical examination is held, which cannot exceed 30 days since the moment of appeal.

The warranty does not apply in case of:

- Inappropriate transportation, storage or maintenance, described in User's Manual;
- Unauthorised opening the device during the warranty period;
- Repairing controlled by someone or some organization not authorised by Galileosky during the warranty period;
- Signs of electrical and/or other damage due to prohibitive mains parameter changes, misapplication and neglect of the device;
- Physical damage of the device body and board, SIM holder, aerials or wires break;
- Traces of oxidation of outer and inner parts or exposure of the device body to moisture;
- Theft or criminal damage of the external aerial or cable;
- Damages caused by foreign objects, substances, liquids, insects coming into body;
- Damage caused by exposure to high temperature or intense microwave radiation;
- Damage caused by elemental forces, fire, social factors, random external factors and accidents;
- Damage caused by parameters incompatibility or inappropriate attachment of additional devices or sensors to the tracker;
- Operation of the tracking device by the vehicle network voltage deviating from the range mentioned in technical specifications.
- Damages caused by incorrect installation of the tracking device to the vehicle;
- Failure in tracker's operating due to incompatibility of software version and tracker version.
- Connection socket, contacts and SIM-holders are not covered under warranty.
- Warranty period for aerials – 6 (six) calendar months from the moment of realization note in device passport, but not more than 8 (eight) calendar months from the moment of device shipping to the Buyer from the Manufacturer storage included in delivery note.
- Warranty period for a processor, GSM module, GLONASS/GPS module – 34 (thirty-four) calendar months from the moment of realization note in device passport, but not more than 36 (thirty-six) calendar months from the moment of device shipping to the Buyer from the Manufacturer storage included in delivery note.

ATTENTION! The manufacturer shall in no case be liable for claims concerning the damage or loss of the data exceeding the cost of the product, as well as claims for incidental, special or consequential damages (including in each case, without limitation, damages for inability to use the equipment, loss of the data, loss of business, loss of profit, loss of savings, loss of time), arising out of the use or inability to use the equipment within legal limits.

ATTENTION! The Warranty does not affect the statutory rights of the consumer, such as the guarantee of satisfactory quality of work or conformity of the product to the purpose for which analogous products are used under normal conditions and service maintenance and also your rights with regard to the seller of the product resulting from the fact of purchase and contract of sale and purchase.

ATTENTION! Terms of Warranty service, which are in conflict with the current law, have no legal effect and are subject to the current law.

ATTENTION! If the Purchaser fails to comply with the Terms of Warranty, the validity of the Warranty is void.

Contacts of Galileosky Technical Support Team
+7 (342) 270 07 99 add. 2
+7 (499) 704 47 99;
e-mail: support@7gis.ru