Terminals Galileo GLONASS v4.x
User's manual

Firmware 0206

Quality

Reliability

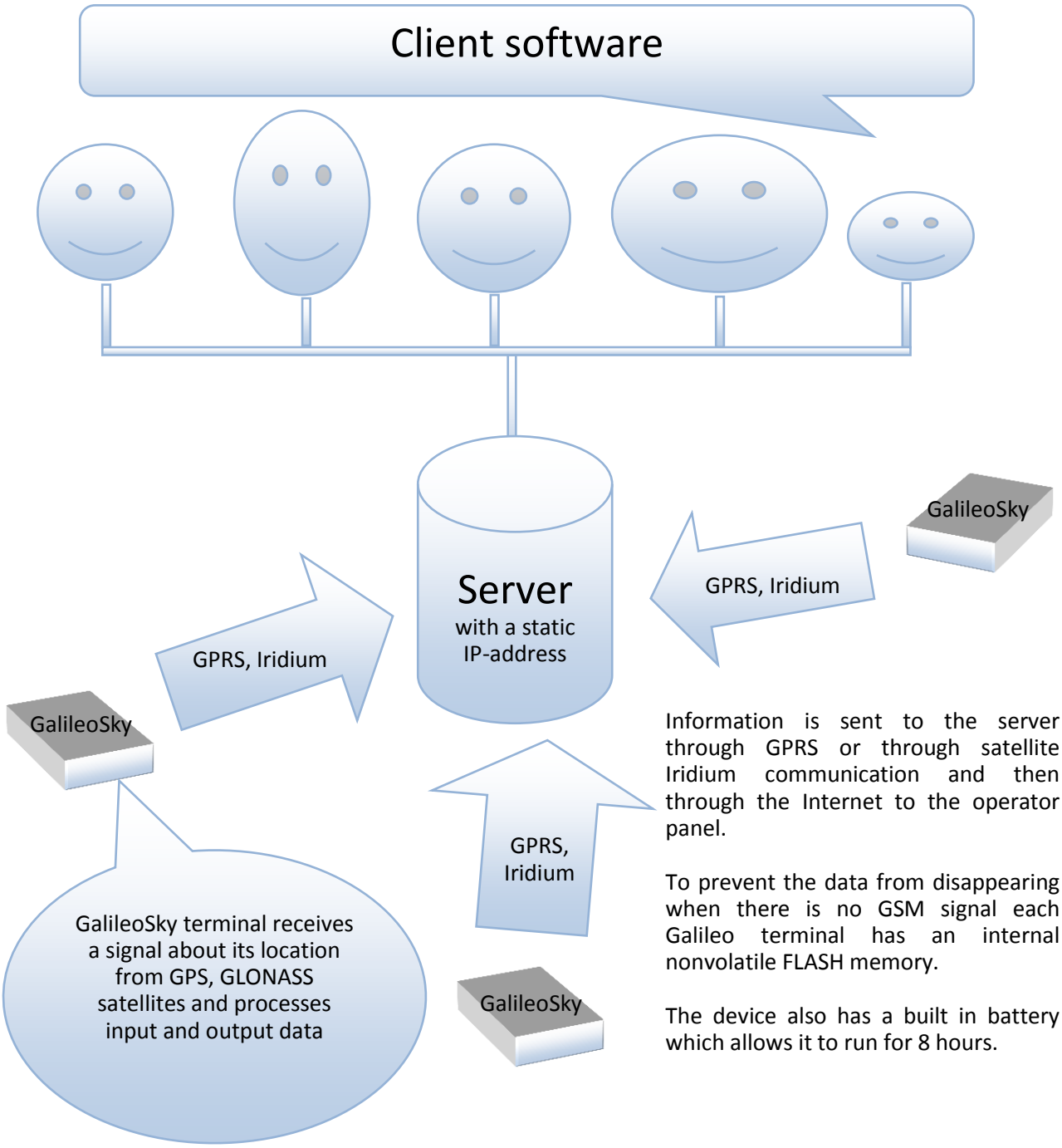
Simplicity

GALILEOSKY

Contents

Introduction

Galileo sky Ltd. produces GalileoSky devices for GPS and GLONASS monitoring of vehicles in real time. The 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, the device state, the state of digital interfaces.
The terminals can be used in any vehicle.



Information is sent to the server through GPRS or through satellite Iridium communication and then through the Internet to the operator panel.

To prevent the data from disappearing when there is no GSM signal each Galileo terminal has an internal nonvolatile FLASH memory.

The device also has a built in battery which allows it to run for 8 hours.

The 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 enabled remote software update;
- ✓ Continuous troubleshooting of the device through the USB port;
- ✓ Car alarm and a remote engine start;
- ✓ Protection of stationary objects;
- ✓ Automatic stops announcement;
- ✓ Adjusting the device through SMS, GPRS, USB;
- ✓ And others (see Terminal unit's performance and Connecting external peripherals).

The information sent by the terminal includes:

- ✓ The exact Greenwich time and date;
- ✓ Vehicle coordinates: latitude, longitude, height;
- ✓ Vehicles speed and direction;
- ✓ Vehicle speeding-up;
- ✓ Temperature inside of the device;
- ✓ Input (buttons) and analog sensors state;
- ✓ External digital sensors state (fuel, temperature sensors etc.);
- ✓ Discrete outputs state;
- ✓ And others (see details of transmitted data in GalileoSky protocol)

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

Before starting the work study the instruction carefully.

Package

The standard package includes the Galileo terminal (hereinafter referred to as the terminal) and a pin connector with contacts. Everything extra should be bought separately.

The exterior of the Terminal:

1. **GSM antenna connector**
2. **GLONASS antenna connector**
3. **Iridium antenna connector**
4. **microSD**
5. **USB slot**
6. **SIM 0**
7. **SIM 1**

The terminal has 4LED indicators which show its current status: red (external power supply), yellow (microcontroller), green (GLONASS receiver), and blue (GSM modem). **See LED indicators.**

You will also need:

1. USB-cable	1
2. GLONASS antenna	1
3. GSM antenna	1
4. Power supply unit	9V-39V (15W)1

Technical specifications

Parameter	***Description***
Discrete, analog, pulse and frequency inputs	*6 pcs.;* *voltage range – 0-33 V;* *Maximum measured frequency – 2 kHz;* *Input resistance of every input is 14 kOhm to the ground.*
Transistor outputs (output 0/1)	*4*
Battery type	*Li-Ion battery; 600mAh;*
Average power consumption	*1,6 W*
ADC resolution in bits	*10*
Archive capacity	*At the installed 2Mb flash memory - up to 58 000 points;* *At the installed 16Mb flash memory - up to 450 000 points;* *When using micro-SD card up to 2500000 points* per GB.
1-Wire	*yes*
CANBUS	*J1939, FMS, J1979, OBD II, 29- bit and 11-bit identifiers*
RS485	*1*
USB 2.0	*Terminal troubleshooting, adjusting, reflashing*
microSD	*support of cards of up to 32GB*
Speakerphone	*Yes, only through GSM*
Speaker (Autoinformer)	*built in*
The number of Geofences for voice prompts	*Limited by the micro-SD card capacity*
Speaker output type	*analogue (linear output) 250mW*

The size of a data packet sent by the device	*GalileoSky protocol: variable-length protocol, tag format*
Accelerometer	*built in*
Type of GLONASS receiver	*MGGS2217*
Positional accuracy, 95% of time	*5 m*
GSM modem	GSM 900/1800, GPRS *class* 10
2 SIM-cards support	*yes*
Voice menu	*yes (when using SD-card)*
Satellite modem	*Iridium*
Dampproofness	*no*

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*
Continuous work form battery	*depends on the settings, 8hrs on average*
External power supply	*9-39 V,* is protected against voltage jumps in the vehicle power supply
Dimensions	157,0 mm x 72,0 mm x 28,0 mm
Weight	*within 300g*
Body material	*Metal*

Warranty	*1 year since the purchase date;*
Average service life	*10 years*
Internal Li-Ion battery life	*500 charge/discharge cycles, two years maximum*

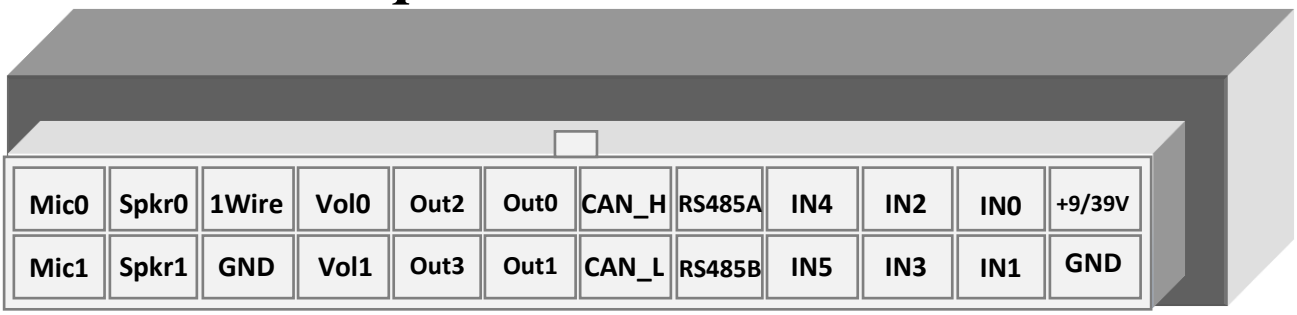
Safe operating rules

Before using the terminal study the instructions of GSM, GPRS, Iridium devices safe operating.
Make sure the polarity is correct when connecting to the power supply.
It is necessary to feed the device from the vehicle's accumulator, but not from the vehicle's network.
Caution! To avoid failure:

- **Connect the contacts correctly!**
- **Unused contacts must be will insulated!**

The ground is connected to the device body. In order not to damage the terminal or the vehicle's electronics it is necessary to separate the device body and the vehicle.

Contacts description



Contact	Description
+9/+39V	Positive supply voltage
GND	Negative supply voltage
IN0	Zero analog discrete input
IN1	First analog discrete input
IN2	Second analog discrete input
IN3	Third analog discrete input
IN4	4th analog discrete input
IN5	5th analog discrete input
RS485A	RS485 A signal
RS485B	RS485 B signal
CAN_H	CAN interface CAN_H contact
CAN_L	CAN interface CAN_L contact
Out0	Zero transistor output (output 0/1)
Out1	First transistor output (output0/1)
Out2	Second transistor output (output0/1)
Out3	Third transistor output (output0/1)
Vol0	Zero contact to connect an external speaker for autoinformer function
Vol1	First contact to connect an external speaker for autoinformer function
1-Wire	1-Wire interface
GND	Ground to connect interfaces which need the ground contact
Spkr0	Zero contact to connect dynamics of external headset
Spkr1	First contact to connect dynamics of external headset
Mic0	Zero contact to connect external headset microphone
Mic1	First contact to connect external headset microphone

Connecting

GLONASS antenna installation

Carefully screw the antenna to the terminal, the upper side above. To have a better view of the sky it is recommended that the antenna should be mounted on the vehicle roof, windscreen or under the dashboard.

If GLONASS antenna is mounted correctly, your coordinates will be found in 1.5 minutes. To be sure see that the green LED indicator is on. (***See' LED indicators')***.

GSM antenna installation

Carefully screw the antenna to the terminal.
The antenna should be mounted in such a way so as to prevent the GSM signal from fading because of the vehicle body, for example, under the dashboard or outside the vehicle.

To make sure the GPRS modem is sending data see that the blue LED indicator is on. (***see' LED indicators')***.

Iridium antenna installation

Carefully screw the antenna to the terminal, the upper side above. To have a better view of the sky it is recommended that the antenna should be mounted on the vehicle roof, windscreen or under the dashboard.

Inserting SIM-card

Use a card with activated GPRS and SMS services.
Insert the card carefully ***without applying excessive force***.

1. To eject the SIM holder press the indicated place with something sharp (needle, toothpick);
2. Insert the card so that it is completely hidden in the holder cover.

The second SIM-card should be inserted in a similar way.

Connecting power supply to the device

Positive supply voltage should be connected to contact +9/+39V, negative supply voltage should be connected to GND. (See ***Contacts description).*** If the connection is correct, the red LED will be on.

LED indicators

- **Red LED**

Is on when the power unit is connected to the terminal.

- **Yellow LED**

Is on when the microcontroller is running (blinks with the frequency of 1 Hz).
It is also used to indicate the bootloader mode. (see LED indicators during device flashing)

- **Green LED**

Shows the GLONASS unit status.

Blinking frequency, times	Description
3	GLONASS unit is not found and is at the initialization stage
2	GLONASS unit is found but there are no correct coordinates
1	GLONASS unit works properly, coordinates found and updated once a second

- **Blue LED**

Shows the GSM unit status.

Blinking frequency, times	Description
4	Stels mode (GSM unit is off and is set to be on according to schedule)
3	GSM unit is not found or is at the initialization stage
2	GSM unit is found but there is no server connection
1	GSM unit works properly, server is connected

Terminal units functional description

Discrete analogue inputs (DAI)

To attach external sensors the terminal has 6 discrete analogue inputs which are pulse-frequency at the same time. Each input's function is set in terminal settings (see Discrete analogue inputs setting and Inputs/outputs). In Contacts description inputs are designated as IN0, IN1, IN2, IN3, IN4, IN5.
Each input saves its values to the nonvolatile memory, i.e. in case 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	33mV
Maximum transmitted signal frequency	2 kHz (synchronous measuring at 2 inputs) 1,5 kHz (measuring at 3 inputs) 1 kHz (measuring at 6 inputs)

DAI has the following settings:

Parameter	***Explanation***
Filter type (input function)	0 - arithmetical average (also discrete input state is generated); 1 - pulse count; 2 - frequency input; 3 - pulse count from two synchronous connected sensors.
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 terminal counts extra pulses, the filter length should be increased by one and accuracy estimated.
Ranges for response/ nonresponse areas (logical 1 and 0)	To process discrete signals, discrete signal response/nonresponse range should be set where signals equal to one and zero. Discrete input statuses should be seen in the field Status Of Inputs, but not in the Input voltage. (Table 2. GalileoSky protocol tags). 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) While counting pulses from 2 synchronous connected sensors, response zone limits must be the same and equal to half of pulse value at response of one of the sensors. Non-response zone limits are equal to half of pulse value at two sensors simultaneous response.

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 Status of Inputs field, and a point will be recorded. If there is no another pulse for 30 seconds, the bit returns to 0.

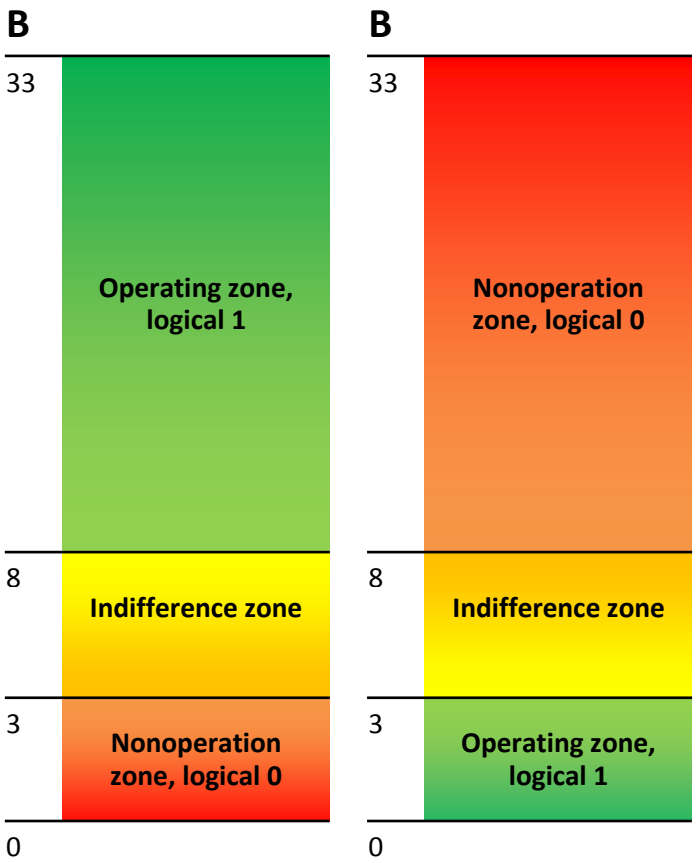
Mean value and discrete event generation

Let us consider the example with the following zero input setting (see the left-hand figure**):**

Filter type: 0;
Filter length: 5;
Logical one zone range is 8-33V;
Logical zero zone range is 0-3V.

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 the Status of Inputs field and a point will be recorded.
At 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 changes its state only in the logical one/logical zero zone

Example2.
In contrast to example 1 (see the right-hand figure) the logical one zone and the logical zero zone have changed places.



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.

Frequency count from two synchronously connected sensors

The Terminal allows connection of 2 pulse sensors on one input, in this case pulse fronts number is count, i.e. for each sensor response counter value increases by 2. Connection circuit details are given in section Connection of passengers flow registration gauge Ш2.

Determination of strike and incline

All devices can determine the terminal strike and incline.
Accelerometer axis directions:

To determine strike:

1. Install the terminal so as one of the accelerometer axis looks vertically, it will exclude false detections on road unevens;
2. Turn on strike and incline determination by SHOCK command (see Track parameters setting). For example, if Z axis is vertical: SHOCK 3,90,5,1200.

A strike is an acceleration increase of 4g in horizontal plane; the correspondent bit is put in the device state field (Table 3. Explanation of device state field) and strike coordinates are recorded.

To determine incline:

1. Install the terminal in 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 terminal homing position change in a vehicle, SHOCK command should be given to adopt the terminal to a new position.

Data archiving to the external SD card

To create a backup on the external micro-SD card it is necessary to insert it into the device. If the need arises, it can be ejected from the terminal and the data can be read in a file manager or explorer with a card reader. It is also possible to send the archive to the server. CSV-files can be opened both with a text editor and Microsoft Excel. The saved data will be ordered in the following way:

MSD:\Track\
    20100201.csv
    20100202.csv
    ...
    20100331.csv

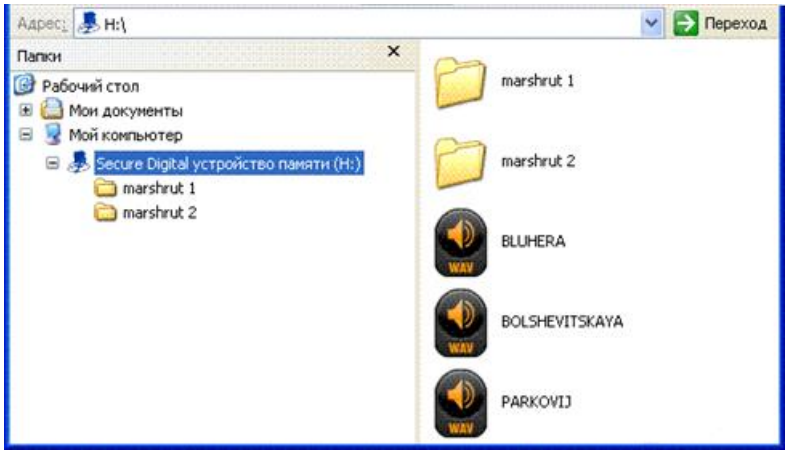
If there isn't enough space on the micro-SD card, (less than 12 MB) the device will delete the oldest files from the Track directory.

Autoinformer function

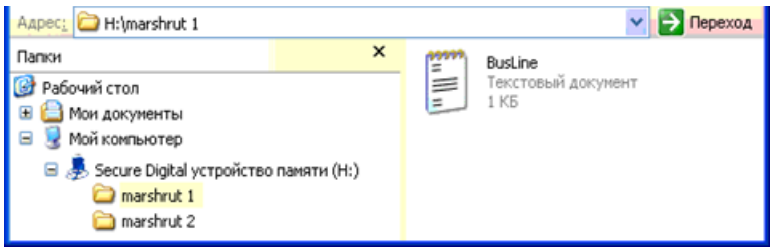
Autoinformer function may be used for automatic (without participation of the driver) public transport stops announcement with the use of a satellite navigation system.
The main difference from analogous systems is taking into account vehicle movement direction, thereby excluding false operation at other stops located in the same geographic area.

To use autoinformer it is necessary:

1. Attach the speaker to the terminal (see Connecting autoinformer speaker).
2. Set the micro-SD 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, at 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.



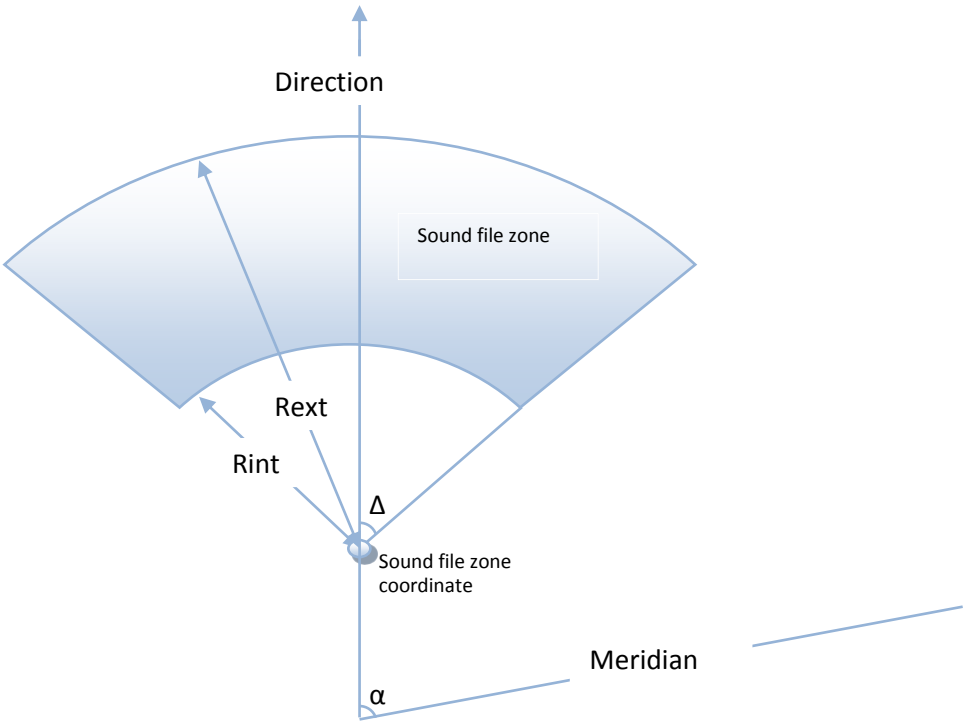
The format of one zone:

- Latitude;
- Longitude;
- Direction angle α (the angle between the meridian and a vehicle direction);
- Spread for the direction angle Δ (see the diagram below);
- Response (activation) zone outer radius Rext;
- Response (activation) zone inner radius Rint;
- Sound file name corresponding to this zone.

It is convenient to fill in the information for zones from Device tab of Configurator while going along the route. At 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 through the point "." (For example: 57.9842) where the value after point – is degree fractions. To transfer minutes into degree fractions (Xderg.Ymin.) use the following expression Xdegr. = Ymin./60. For example: 57 derg. 55.4513min = 57.924188 degr.

Explanatory diagram



3. Activate the autoinformer function with the Autoinformer command. (see Autoinformer setting).
4. Insert a micro-SD card into the terminal and reset it with the Reset command. After the terminal resetting the function will be activated.

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

To test sound files:

1) Unscrew GLONASS aerial from the terminal;
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 micro-SD card root. This file will be played again and again after the terminal resetting.

Signaling function

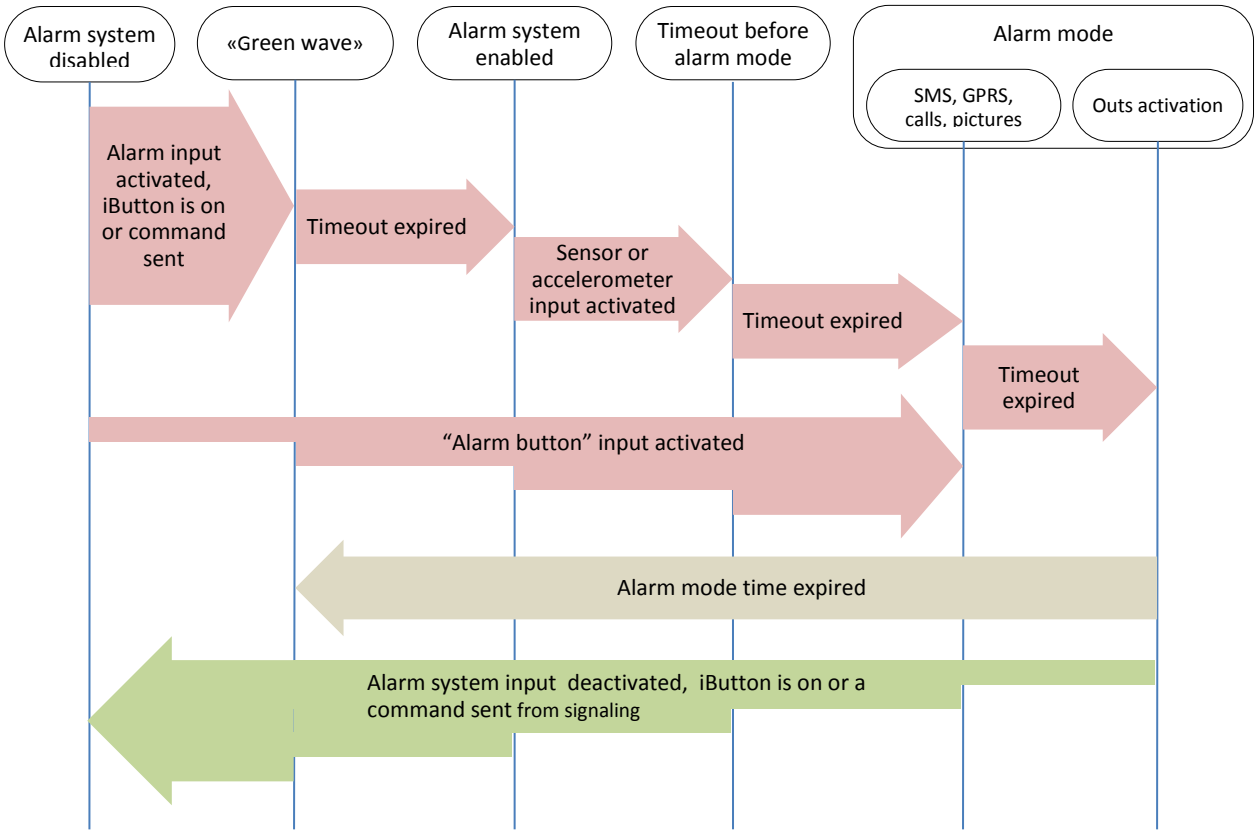
Signaling function allows assigning the response to:

1. analog input status change;
2. strikes and inclines (accelerometer data);
3. location change;
4. speeding;
5. iButton or RFID CARD connection.

The terminal can react by inverting input status, sending an output pulse, sending SMS, making a preset number(s) telephone call, taking a picture or recording the point.

The settings that users can change are as follows (section Signaling mode setting):

1. the time when input signals are not processed ("green wave"), with signaling enabled;
2. the maximum time of alert mode after which the Terminal will automatically change into an alarm system mode;
3. the time between activation and change to the alert mode individual for each input;
4. an SMS text message when changing to the alert mode individual for each input;
5. the time between enabling the alert mode and status change individual for each input.



Alarm mode states change diagram

Alarm system activation and deactivation can be made by input, SMS or server message, using the iButton key previously programmed in the Terminal (see Digital inputs setting). Commands prevail over inputs states. Input activation depends on the settings given by the InCfg command (see Discrete analogue inputs setting), the level outputs are inverted with respect to is set by the Out command (see Transistor outputs setting).

Monitoring data transmission via GSM

The terminal allows data transmission to the main and backup monitoring server via GSM. If there only transfer to the main server is set, continuous connection will be maintained. If there transfer to both servers is set, the terminal is connected to the main server and then after the set period of time it breaks the communication and connects to the backup server etc. The terminal accounts transmitted data separately for each server, thus both will receive full archive with the track.
Transmitted data may be coded; XTEA3 algorithm (http://tomstdenis.tripod.com/xtea.pdf). 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 micro-SD card and to adjust archive transmission from it (Archive command, section Service command).

Monitoring data transmission via satellite Iridium modem

The terminal is equipped with Iridium satellite modem that allows the transfer of monitoring data outside of the coverage zone of GSM networks. Satellite Iridium communication allows you to transfer data from any point on the surface of the Earth. The Terminal allows you to configure the absence of GSM communication after which the Iridium modem will be automatically switched on (Sputnic command, section **Ошибка! Источник ссылки не найден.**). After switching on the satellite modem the Terminal will try to send current coordinates and the sensors readings within 10 minutes. The terminal then waits again for a predetermined time interval, if the GSM coverage is still missing, the terminal sends the packet again via the satellite modem. Data in the packet are formed in accordance with the settings of the main packet of GalileoSky protocol (MAINPACK command, section Server exchange protocol settings).

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 Terminal stores the data from all the inputs and interfaces, even when they have no connected sensors, in the internal flash memory archive. If it is unnecessary to store all the data, the dynamic archive can be used (FLASHARCHIVE command, see Service commands). In this case only the data selected in configuration of the head and main packets will be saved (HEADPACK and MAINPACK commands, see Server exchange protocol settings).
Any change of configuration of the head and main packets when the dynamic archive is on can cause flash memory formatting and data loss.
The use of dynamic archive can considerably increase the maximum number of kept pixels up to 58000.
By using the internal flash memory it is possible to choose the order in which pixels 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 (section).After changing the direction of memorizing data the flash memory will be formatted and the data will be lost.
By using a micro-SD card the data are always stored in chronological order. Note, that only current data are used for the first packet.

Operation with two SIM-cards

The Terminal has 2 slots for installation of SIM-cards. Only one SIM-card can be active and support registration in GSM-network at the same time. Each SIM-card has its own APN. If a PIN code is used, it must be the same for both SIM-cards. The Terminal supports the following algorithm of SIM-card operation:

1. Only one SIM0 card is always active.
2. Automatic switching to the other card, if the data cannot be sent to the server within 9 minutes. Switching occurs in cycles, i.e. first SIM0 is used, then SIM1, and after this SIM0 again.

The second algorithm is always used when updating firmware remotely; the Terminal attempts to get the connection to the server with firmware through SIM 0, and if it fails - through SIM 1.

GPRS traffic costs optimization

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

1. Turn off the transmission of unused data, for example temperature, acceleration, analog and digital inputs values which have no connected sensors. It can be made in the Configurator tab Settings/Protocol or by MainPack and HeadPack commands (see Server exchange protocol settings).
2. Increase points record period. It can be made in the Configurator tab Settings/Track or by WrPeriod command (see 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 tab Settings/Tracks or by Turning command (see Track parameters settings).
4. Find out the time of disconnection because of the terminal inactiveness from the server software developers. This parameter should be taken into account at points' record period setting otherwise the traffic will increase by reason of costs for restoring connection to the server. Example: points' record period at a stop is 1200 seconds (20 minutes), the server disconnection by reason of the terminal inactiveness is 180 seconds (3 minutes). The terminal 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 terminal. The terminal tries to reconnect the server at once. It happens 6 times and only in 20 minutes the terminal 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 terminal can correctly chose points' record period. The terminal 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).

If continuous online monitoring is not necessary it is possible to set packet data transmission (see Stels mode and package transmission). In this case the device will periodically contact, send the data from the blackbox and disconnect from the server. Savings are due to decrease of costs for one data packet transmission as when sending data from the archive a packet size may be up to 1000 byte, and at online monitoring usually one point is sent (a few tens of bytes). At the same time the terminal operation from the battery increases as during server disconnection periods the device switches GSM-modules off.

Operation in international roaming

The terminal allows setting special parameters of data transmission in the international roaming (Roaming command, section Data transmission settings). After registration in GSM-networks the terminal receives code of the country and code of the operator from base station and compares them with the set once, if they do not match the terminal is in roaming. You may specify only code of the country (international roaming) or code of the country and code of the operator (national roaming) Being in roaming the terminal constantly supports registration in GSM-network but initializes GPRS-session only according to the schedule, thus it is always possible to make a call to the terminal or send SMS with a command and decrease GPRS-traffic costs. For GPRS-session the maximum volume of transmitted data in bytes is determined. Each cell operator has minimum tariffing interval in roaming, it is recommended to set maximum data volume equal to half of this interval (the second half is for official traffic TCP/IP which volume depends on connection quality). At archive transmission from internal flash-memory, the terminal always unloads the data in accordance with the archive settings (FLASHARCHIVE command, section Service commands). At archive transmitting from SD-card it is recommended to set coordinates transmission and sensors indications in the first packet, thus the terminal sends one point with a current vehicle coordinate and the oldest unloaded archive part. The data from the SD-card are unloaded in chronological order.

Stels mode and packet transmission

In this mode the Terminal switches GSM unit off and contacts only according to a strict schedule, which allows decreasing the Internet traffic and power consumption.
Stels mode settings command: «*stels pday,phours,minGSMon*», where

- *pday* – device contact is enabled every *p days* since the beginning of the month, in other words on *pday*- multiple days;
- *phours* – device contact is enabled every *p hours* since midnight GMT, in other words at phours - multiple time.
- *minGSMon* – GSM unit is enabled for *minGSMon minutes since the beginning of the hour*.

Packet transmission can also be set in the Configurator on tab Settings/Data transmission.
To disable these modes use the «stels0,0,0» command.
Setting examples:

1) – contact once a day at 14.00 GMT;
- staying in network for 15 minutes.
Setting command: stels 1,14,15
To enable contact once a day phours must be greater than 11, i.e. it can be enabled at 11 and at 22 o'clock. If it is set to contact every 12 hours, the contact will be enabled at 12.00 and the next must be at 24.00, but this is another day, i.e. it is not realized.

2) – contact every day with 2 hours interval;
- staying in network for 15 minutes.
Setting command: stels 1,2,15

3) – contact once in three days at 23.00 GMT;
- staying in network for 15 minutes
Setting command: stels 3,23,15

Note:

- contact at 0 o'clock GMT cannot be enabled whatever the settings;
- if the device is in the stels mode, remote commands will work only when the radio silence mode is disabled, i.e. GSM unit is on;
- do not set the contact time less than five minutes, otherwise the device will not have enough time to establish a link with the server and to tell its location

Geofences

The terminal allows setting areas where coordinates are not updated, the GSM unit is switched off. It is also possible to set periodical camera shooting (PhotoCfg command, section Photo camera settings). Each area

is described by the coordinates of the center and by the radius. Geofences' setting commands are given in section Track parameters setting.

Power saving

To reduce power consumption of the Terminal in the operating mode, perform the following steps:

1. For unused RS232ports, execute RS2320 0 or RS2321 0 command, or specify periphery type as "nothing" in the Configurator.
2. Turn off the integrated CAN-controller if the Terminal is not connected to a CAN-bus. This can be performed by using CANREGIME command with the first parameter set to 0, or by specifying "CAN disabled" as the "Filter type" in the Configurator.
3. Turn off the Autoinformer when it is not in use. This can be done by sending AUTOINFORMER command with the first parameter set to 0, or by unticking the "Autoinformer" section of the Configurator.
4. Reduce the degree of track details. The lower this degree, the less the power consumption.

To reduce power consumption of the Terminal at a stop, perform the following steps:

1. Set up the shutdown of the GPS/GLONASS module at a stop, this can be performed by using SLEEPMODE command (Service commands section) or in the "Power saving" tab in the Configurator.

Enable the "deep sleep" mode at a stop. The "deep sleep" mode is turned on at the end of a pre-specified time period at a stop. In this mode the Terminal disables the specified modules (GPRS, CAN, RS232, micro-SD), reduces the ADC sampling rate, does not sample 1Wire sensors and does not charge the battery. The behaviour in the "deep sleep" mode can be configured by using SLEEPMODE command or in the "Power saving" tab of the Configurator.

Connecting external peripherals

CAN-interface

The terminal allows to extract information from the CAN bus.
The following protocols are supported:

– J1939 (FMS). According to this protocol the Terminal is not a device transmitting to CAN bus, the device does not change vehicle operation, it also doesn't send confirmations to vehicle units packets and there are no electrical noise in the CAN bus. In some cases at 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 Terminal (see CAN settings).
– J1979 (OBD II). This protocol works according to the question-answer mode, consequently the Terminal transmits the data to the CAN bus.

Available performance modes:

J1939_SCANER – the bus scanner sending bus reports to the configurator.
FMS – a standard FMS protocol filter. (see www.bus-fms-standard.com).
J1939_USER_29bit – a configurable user filter. Identifier length is 29 bits.
J1939_USER_11bit – a configurable user filter. Identifier length is 11 bits.
J19379_SCANER – *the* bus scanner defining bus speed and identifier capacity.
J1979 _29bit – a standard J1979 protocol filter for 29 bits identifiers.
J1979 _11bit – a standard J1979 protocol filter for 11 bits identifiers.

J1939_SCANER mode

This mode is intended to study CAN bus reports, according to J1939 protocol.
Bit rates from 10000 bit/s up to 500000 bit/s (typical values: 62500, 12500, 250000, 500000) are supported.
11 and 29 bit identifiers are supported.
The scanning mode works as follows:

1. The «**CAN. Start scan.»** message is displayed;
2. The CAN bus reports are displayed with a delay indicated by the CAN Regime command. (see CAN settings).

29bit identifiers are displayed in the following format:
ID=00000009 (8) 06 07 08 09 00 CC DD EE
where
ID – is a 29bit message identifier;
(8) – is the number of received bus bytes.
06 07 08 09 00 CC DD EE – is an 8byte message. (The lower byte on the left, the high byte on the right),

11bit identifiers are displayed as:
ID=009 (8) 06 07 08 09 00 CC DD EE
where
ID – is an 11bit message identifier;
(8) – is the number of received bus bytes;
06 07 08 09 00 CC DD EE – is an 8byte message. (The lower byte on the left, the high byte on the right).

3. After all the identifiers have been displayed you can see the **«CAN. End scan.»** message in the diagnostic window.

To enable this mode:

1) Connect the terminal to the vehicle CAN interface;
2) In the Configurator tab Settings/CAN select bus rate and delay time (time of message waiting time);
3) Press "Start Scanning J1939". Received data are displayed in the right panel.

FMS mode

This mode is included in all terminals by default; it allows retrieving and decoding messages relevant to FMS protocol:

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

Attention! Many car manufacturers support FMS protocol partially or do not support it at all.
To enable this mode:

1) attach the terminal to the vehicle's CAN interface;
2) give the «CanRegime 2,250000,2000» command (see CAN settings) or select FMS 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 right data transmission to the server using the MainPack command (see Server exchange protocol settings) or in the Configurator "Settings/Protocol" tab.

J1939_USER_29bit mode

This mode enables us to receive 29bit identifiers (ID) messages from the vehicle CAN-bus, according to J1939 protocol.

To enable this mode:

1) connect the terminal to the vehicle's CAN interface;
2) select Custom filter (29bit identifiers)type in the Configurator "Settings/CAN" tab, set the bus rate and delay time or give CanRegime command with necessary parameters (see CAN settings);
3) set filers for CAN bus messages.
4) set sending of received data to the server with the help of MainPack command (see 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 terminal (Table 2. GalileoSky protocol tags) 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 terminal. The hexadecimal notation is used for convenience only.
   We can choose the bytes to fill the tag from the information obtained under this ID by means of shifting.

Let us see an example:

The CAN message identifier is ID=0x18F00300.
We need only the first byte of all the sent content with this ID.
As we need only one byte we shall choose the tag CAN_R0 as an example.
The command to set the tag is as follows: CAN8BITR0 ID,Shift.

1) The tag number ID=0x18FEEE00 will look as 419360256 in the decimal system.
2) The byte we need is shifted by one byte that is the second parameter is equal to 1.

So we have the following command to set the filter «CAN8BITR0 419360256,1».
Now when the message in question 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 correspondent tag;
4) Visually indicate the shift using a mouse. The number transmitted to the server will be displayed in the Value Column.

J1939_USER_11bit mode is set similarly.

J1979_SCANER mode

This mode is used to define data transfer rate and Identifier length according to J1979 protocol. If the parameters of transfer are known, it is recommended to use the J1979_29bit and J1979_11bit modes, having specified necessary rate of the bus.
The rate of 250000 bits per second and 500000 bits per second and 11 and 29 bit identifiers are supported.
To enable this mode:

1) connect the terminal to the vehicle's CAN interface;
2) press "Test OBD II". Received data are displayed in the right panel.
3) If scanning finished successfully, data transfer rate and Identifier length will be set automatically.

Attention! Scanning on J1979 protocol can cause failures in on-board equipment operation. GalileoSky Ltd bears no responsibility for any failures after CAN bus scanning.

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 ;
- error codes.

Attention! Many car manufacturers support J1979 partially or do not support it at all.
To enable this mode:

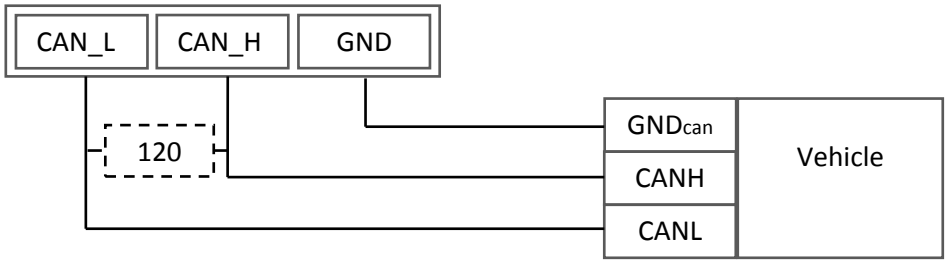
1) attach the terminal to the vehicle's CAN interface;
2) give the CanRegime command (see CAN settings) or select "OBD II 29bit" filter type in the Configurator on "Settings/CAN" tab;
3) make sure the device receives bus data and sends them to "Device" tab in the Configurator;
4) set the right data transmission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

J1979_ 11bit mode is set in a similar way.

Attention! If your vehicle doesn't support J1939 protocol, J1979_USER-29bit and J1979_USER-11bit modes operation can cause failures of board equipment operation. GalileoSky Ltd bears no responsibility for failures after activation of these modes.

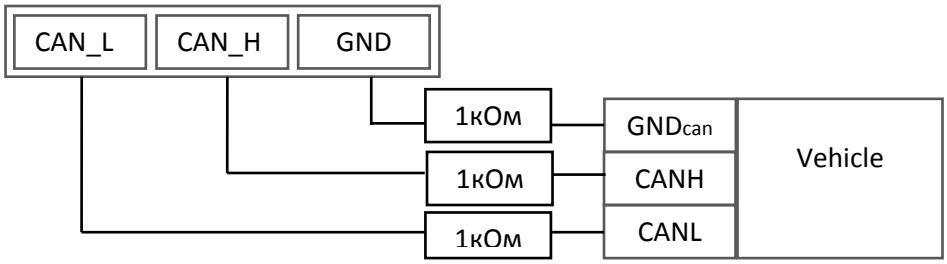
Different variants of connection to the CAN-bus

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.

2. Connection with current-limiting resistors

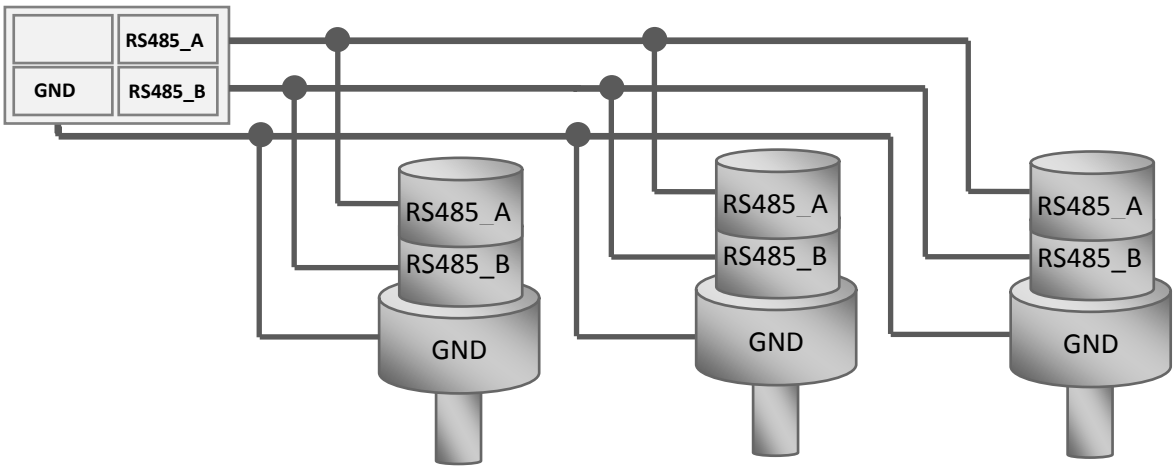


Use the first variant of connection to plug the device into the diagnostic outlet.
The second variant is recommended to connect the device directly to the CAN bus of a vehicle.

Connecting digital fuel sensors using RS485 interface

The order of connection:

1. Connect RS485_A, RS485_B, GND sensor contacts to terminal RS485_A, RS485_B, GND contacts (see Contacts description).
   The sensor power supply is provided separately.

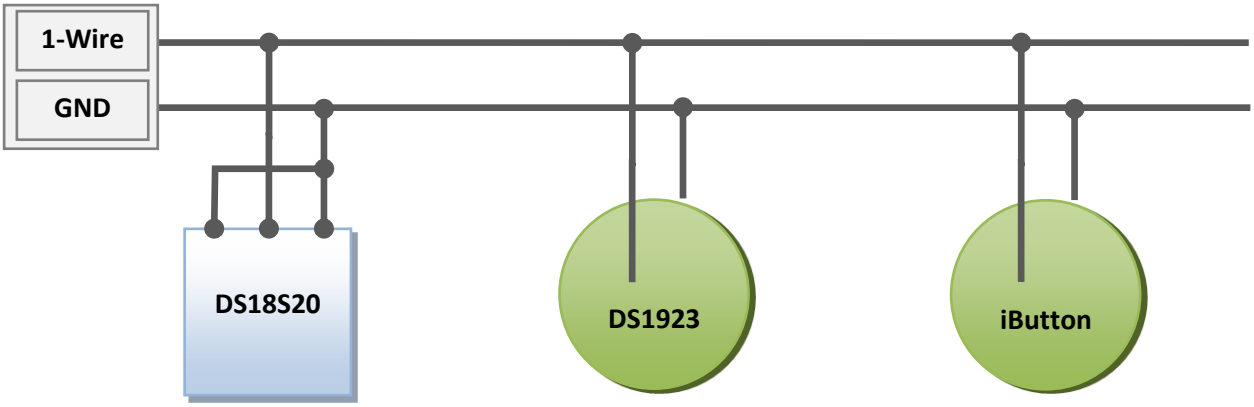


2. To set transfer of received data to the server by MAINPACK command (see Server exchange protocol settings) or in the Configurator on "Settings/Protocol" tab. These data transmission is on by default.
3. Make sure that the terminal receives data from the sensor. It can be done in the Configurator,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

The terminal can support up to 3 sensors at a time. The sensors should have addresses 0, 1, 2 correspondingly. If the terminal receives no messages from the sensor for 18 seconds, the RS485 field value will be cleared. In this way it is possible to detect sensor disconnection or failure.

Connecting 1Wire sensors

It is possible to connect different sensors working through 1-Wire interface, and they can operate simultaneously.



Connecting iButton (DS1990, DS1982) identification key

There are several identification key (IK) applications:

- driver identification;
- trailer turning off identification;
- doors opening identification.

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

The terminal can support up to 8 identification keys with certain identifiers or two identification keys with any identifier. When using a micro-SD card up to 1000 IK with certain identifier are supported.
At 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. At key disconnection the number turns to zero, the point is recorded and message is sent to the server.
There can be set up to eight key identifiers using iButtons command (see Digital inputs settings) or in the Configurator on "Settings\Digital inputs" tab. You should enter the 4 Lower bytes of iButton key number without checksum, in hexadecimal system.
For example, full hexadecimal key number:

09 00 00 00 91 02 0C 5C, where
09 – type of device (in this case it is DS1982, for DS1990 is 01),
00 00 00 91 02 0C – unique number,
5C – checksum.

In this case one should enter 00 91 02 0C.
At 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.
When using a micro-SD card, you may edit a list of trusted keys, with the help of the AddKey, DelKey commands (see Digital inputs 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. (Table 3. Explanation of the device status field).

Connecting DS18S20 (DS1820, DS18B20) thermometers and DS1923 temperature and humidity sensors.

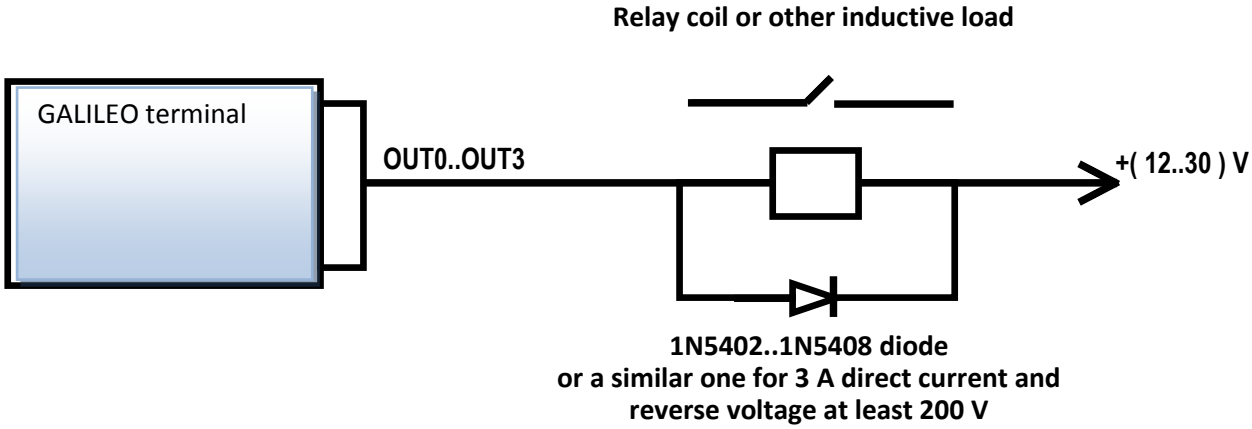
It is possible to connect up to 8 DS18S20 thermometers and 8 DS1923 humidity sensors. To use the sensors connect them to 1-Wire and GND contacts (Contacts description) and activate the corresponding protocol items (Server exchange protocol settings, Table 2, GalileoSky protocol tags). There is no binding between a thermometer or humidity sensor and a certain tag cell in the protocol. All data are stored in memory cells in a definite order: from a lower tag to a high tag. If the number of cells exceeds the number of thermometers, the extra high cells will contain the data which correspond to disconnected sensor state.
At temperature sensor tripping the thermometer field shows disconnection (-128ºC).
At humidity sensor tripping the thermometer field shows disconnection (0%).

Connecting Autoinformer speaker

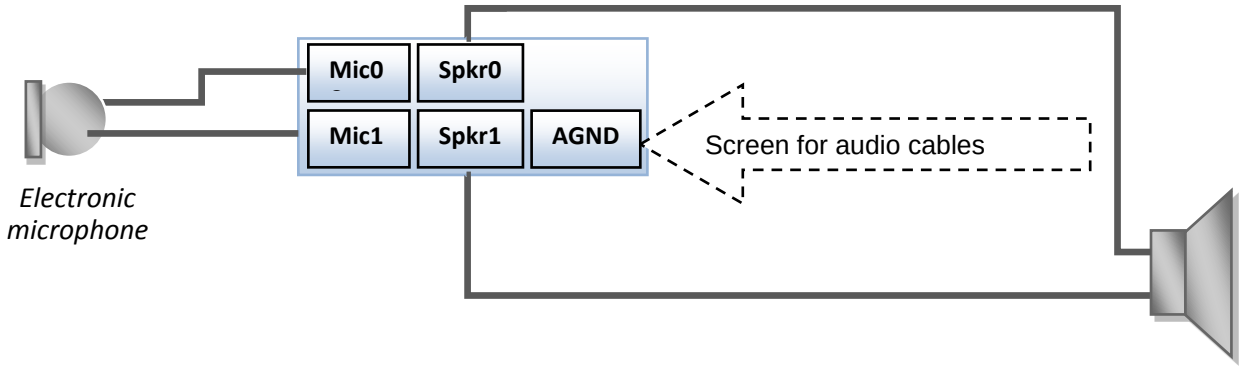
Speaker
Not less than 8 Ohm

Transistor outputs (0/1)

To operate external devices there are 4 discrete «open collector» outputs (see Contacts description) in the Terminal. The maximum output voltage is -+30V, each output current is no more than 80mA.
The output values are stored in the nonvolatile memory, so the device sets these stored values even after being reset.
To operate outputs use Out command (see Transistor output settings) or the Settings/Input/Output tab in the Configurator.
OUT0…OUT3 outputs relay connection circuit



Connecting audio equipment and Tg V1.x push-to talk



Microphone specifications

Parameter	Min. value	Mean value	Max. value
Operating voltage, V		1.60	2.2
Operating current, microampere	70		300
Load resistance, kOhm	1.2	2.2	

Speaker specifications

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

It is also possible to connect Tg V1.x push-to talk. PTT is of GalileoSky LLC production and has a special adapter to connect to the Terminal. For connection it is necessary:

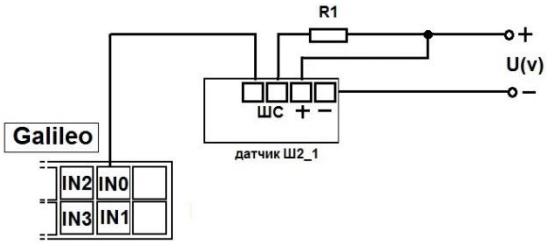
1. Set the 0 input (IN0) to measure the middle value. It can be done via the Configurator in the "Settings" /"Ins/Outs" tab or using the InCfg0 command (section **Ошибка! Источник ссылки не найден.**).
2. Disable tube autolifting. It can be done in the Configurator in the "Settings"/"Sound" tab or using the Autoanswer command (section Voice communication settings).
3. Enable support for push-to-talk. . It can be done in the Configurator in the "Settings"/"Sound" tab or using the Tangenta command (section Voice communication settings).

At incoming call the earpiece of the PTT will issue a beep. The tube can be raised by clicking once on the PTT button. Further, switching between speaker and PTT microphone is made by pressing the button. If the button is pressed - the microphone is on, differently – the speaker is on.

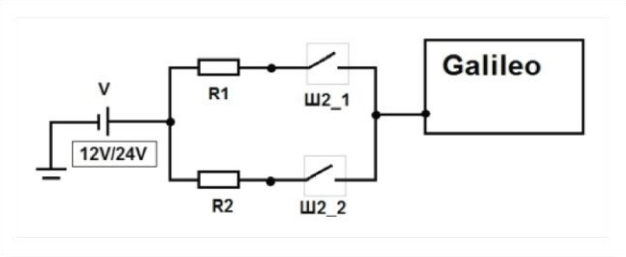
Connecting passenger flow registration sensors Ш2

The terminal supports connection up to 16 Ш2 sensors through 8 discrete analogue-inputs (DAI) IN0-IN7 (Contacts description).

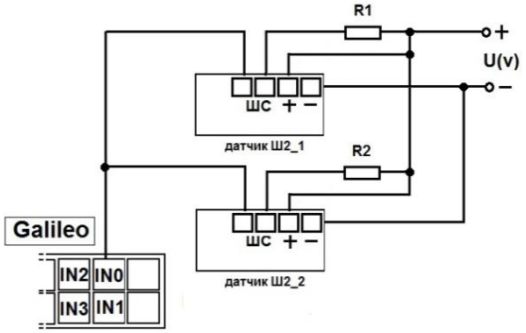
Connection order of one Ш2 sensor through resistor to one of DAI of the terminal.



To connect 2 Ш2 sensors to one of DAI use divisor on two resistors. Calculation principle is realized on voltage level change at sensors triggering.



V – Power supply (battery/ vehicle power supply);
R1, R2 – resistors;
Ш2_1, Ш2_2 – Ш2 passenger flow registration sensors.



Connection order of 2 Ш2 sensors through resistors to one of DAI of the terminal.

To set an input to count pulses from two sensors one may in the Configurator or by **incfg0 3,2,X,X,Y,Y** command (where Y – one sensor triggered; X – two sensors triggered).

Parameter **X** and **Y** depending on supply voltage and **R1, R2** resistors resistance assumes different values, for example:

U(v)=12, R1=10k, R2=10k, тогда X=3500, Y=7921
U(v)=12, R1=14k, R2=14k, тогда X=3000, Y=7000
U(v)=24, R1=10k, R2=10k, тогда X=7000, Y=15842
U(v)=24, R1=14k, R2=14k, тогда X=6000, Y=14000

It is calculated according to the formula:

$$X = \left(\frac{7 * U}{14 + R1 * 0.001}\right) * 1000; \qquad Y = \left(\frac{14 * U}{28 + R1 * 0.001} + \frac{7}{14 + R1 * 0.001}\right) * 1000;$$

Attention! To avoid false operation at sensors connection and further operation of sensors use stable voltage power supply.

Terminal operation result will be pulse fronts count from each sensor, i.e. when one person passes through one door total pulse number increases by 2. Correspondingly, to count passengers number passed through the sensors divide pulse count result by 2.

Configurator

Configurator is a PC program which allows:

- to configure the device via graphic interface and with the help of commands;
- to troubleshoot the device saving the results in a log-file;
- to see the device units state in real time mode;
- to download monitoring data from the internal memory and a SD card filing them;
- to send the downloaded data to the server;
- to specify autoinformer's areas.

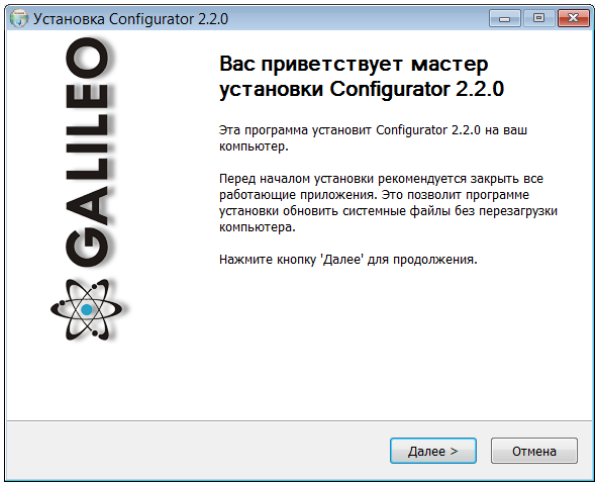
32 and 64 bit OS are supported: Windows 2000, Windows XP, Windows Vista, Windows 7.

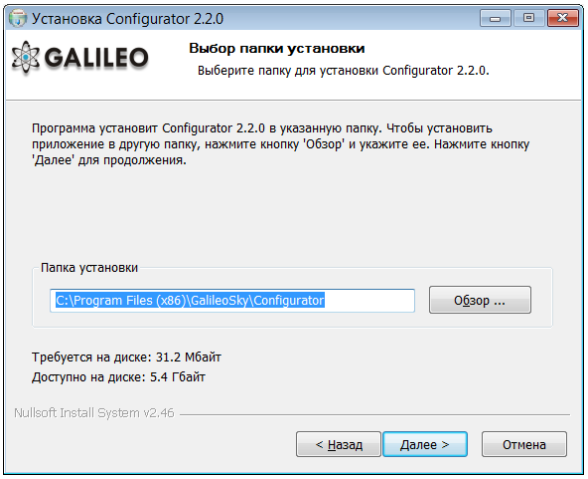
Program installation and running

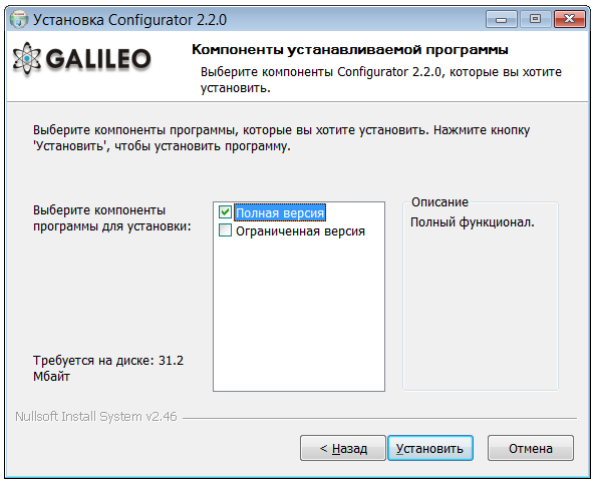
Download the Configurator program from the site and launch it.

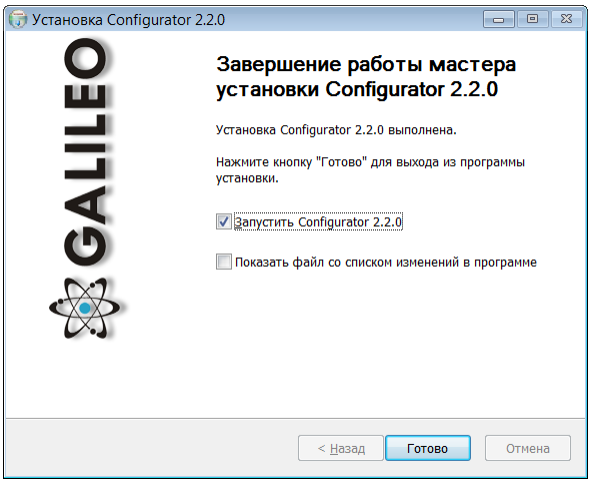
Attention! Program installation may require changes of crucial OS elements. Do not let your antivirus program block the installer operation.

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









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

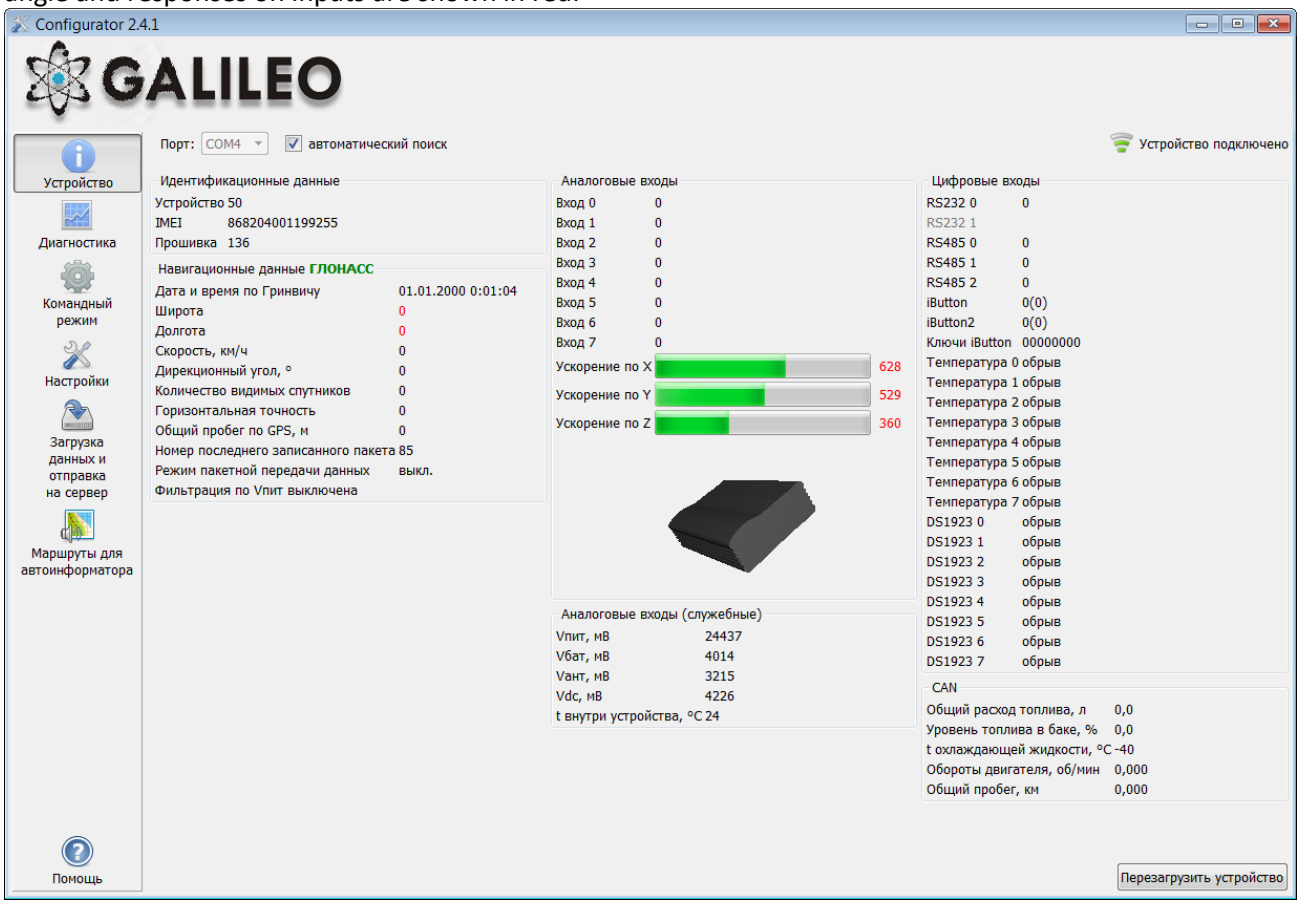
Start the Configurator program (Start menu\Programs\GalileoSky\Configurator). Turn the power of the Terminal on and connect the device to the computer via a USB cable.

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

Vertical menu items

1. Device tab

The tab displays the information about the device state and allows resetting the device. This tab contains the terminal 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 Terminal, the program will request it to access the settings. At wrong code entering the terminal will disconnect from the computer, reset, connect to the Configurator again and wait for the right code enter.

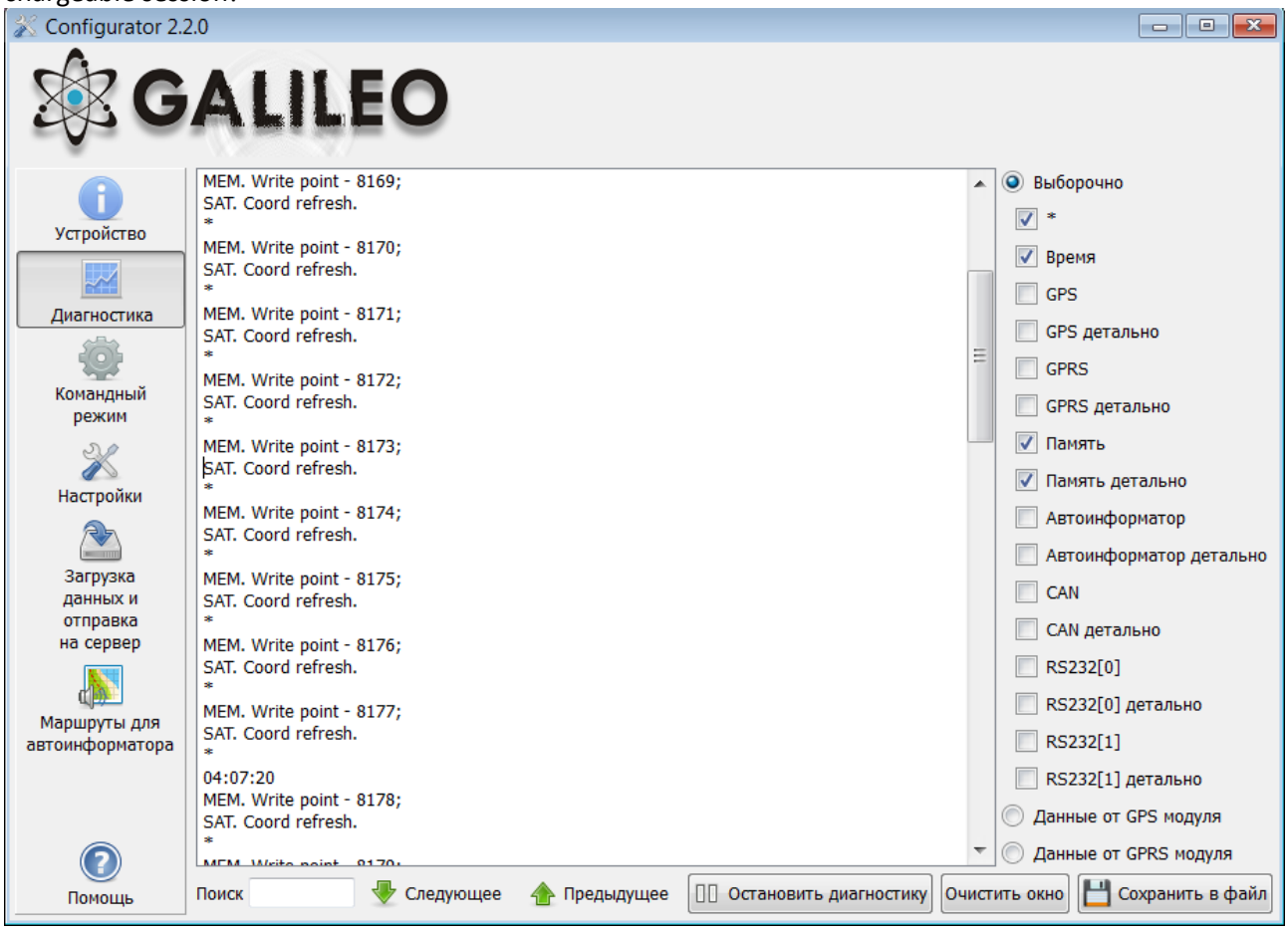
2. Troubleshooting tab

This tab allows us to see the current device state through the device troubleshooting reports.
The troubleshooting mode has the following buttons:

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

2.1. GSM unit debug info

Attention!
If the service has already been registered by the Terminal, another GPRS connection is only possible through switching off the GSM modem. It means that no money will be lost due to the minimum chargeable session!



Troubleshooting messages	Description	Possible causes
GSM. Success turn on.	GSM unit powered. Turning on successful.	
GSM. Not success turn on!	GSM unit powered. Turning on denied by the unit.	
GSM. Success init.	GSM module initialization successfully performed.	
GSM. Not success init!	GSM module initialization failed.	
GPRS. Activated.	Инициализация GPRS-услуги успешно произведена.	
GPRS. Not activate.	GPRS initialization failed.	GPRS is not activated on this SIM

		card. Not enough money on the account. GSM network overloaded.
GPRS. Success connect to server.	Device server connection successful	
GPRS. Not success connect to server.	Device server connection failed	The Server access denied or wrong server settings for the device.
GPRS. Reconnect Number=№	Number of server reconnections. № - reconnection number.	
GPRS. Firstpack OK.	First packet has been sent to the server.	
GPRS. Firstpack False.[0]	The device has sent the first packet, but there is no confirmation at TCP/IP level.	GSM network overloaded. The packet has been blocked by device firewall.
GPRS. Firstpack False.[1]	The device has sent the first packet, but there is no confirmation at the application level.	GSM network overloaded. The server is not handling the first packet.

2.2. SMS debug info

Troubleshooting message	Description
SMS. RX SMS.	A new SMS message received
SMS. TelNum: +79112299922	received from a given phone number
Command: ID	ID command received
SMS. TX OK.	Message successfully sent
SMS delfromslot 1	handled SMS deleting (from the first SIM card slot)
Not reply SIM. Slot 1	no SIM card reply (from the first SIM card slot)
GSM. No SIM-card	no SIM card reply (the card is probably not inserted)

2.3. Internal Flash memory debug info (track memory)

Troubleshooting message	Description
MEM. Inp-s	Point record reason is the change of inputs state;
MEM. Turn,dist	Point record reason is the change of distance between previous and new place or angle of driving direction;
MEM. Time	Record reason is time;
MEM. Write point – 200	Point with sequence number 200 is recorded.

2.4. GPS unit debug info

Troubleshooting message	Description	Possible causes
SAT. Coord refresh.	Coordinates for current record have been updated by GPS unit. The vehicle is considered to be moving, packet has not been filtered off.	
SAT. Coord not refresh.	Coordinates for current record have not been updated. Filtering at stops is activated.	
SAT. Temper is low than -40	Device temperature is lower than -40ºC. Operation at lower temperatures is impossible.	
SAT. Temper is high than 65	Device temperature is higher than +65ºC. Operation at higher temperatures is impossible.	
SAT. Time out. Restart MCU.	No GPS data for 60 seconds. Device reset.	GPS unit out of order. GPS unit failure.
GLONASS. Message received. Len = 401	Device received information from GLONASS unit. 401 byte received.	
GPS. Message received. Len = 172	Device received information from GPS unit. 172 byte received.	
GPS. Change baud rate = 1	Attempt to set GPS unit rate. Attempt № 1.	
SAT. Fix = 1	Current position fixed (0 – not fixed);	
SAT. SatInUse = 7	7 satellites are used for navigation;	
SAT. Valid = 1	Coordinates are right (they can be used for	

	location determination). This *Valid* is not related to *valid* in packet and status.	
Galileo uses GLONASS	Terminal uses GLONASS system.	
Galileo uses GPS	Terminal uses GPS system.	
SAT. Incorrect data from GLNS/GPS module	Wrong data received from the unit in use (probably because of processor overload)	
SAT. Time out. Restart MCU	Device gets no data from receivers (GLNS/GPS)	
SAT. High Speed = 200	Navigation speed data filter turned on (this data will be skipped by the unit).	
SAT. HDOP is high = 6	Navigation HDOP data filter turned on (this data will be skipped by the unit).	
SAT. Jump = 5000	Navigation coordinate data filter turned on (large distance jump occurred).	
SAT. First start OK. Sat count >= MIN	At the terminal turning on the unit must get more than MIN satellites (only in this case the data is reliable).	

Other troubleshooting messages are not described, but they have intuitive names. If there are any questions, you will find the answer at our forum.

3. Command mode tab

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

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.

3.4. Single command example

An example of a command with a parameter:

In the "Commands" window 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.

Command**:** APN internet.beeline.ru,beeline,beeline
Response**:** GPRS:APN=INTERNET.BEELINE.RU, user=BEELINE, pass=BEELINE;

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

An example of a command without a parameter:

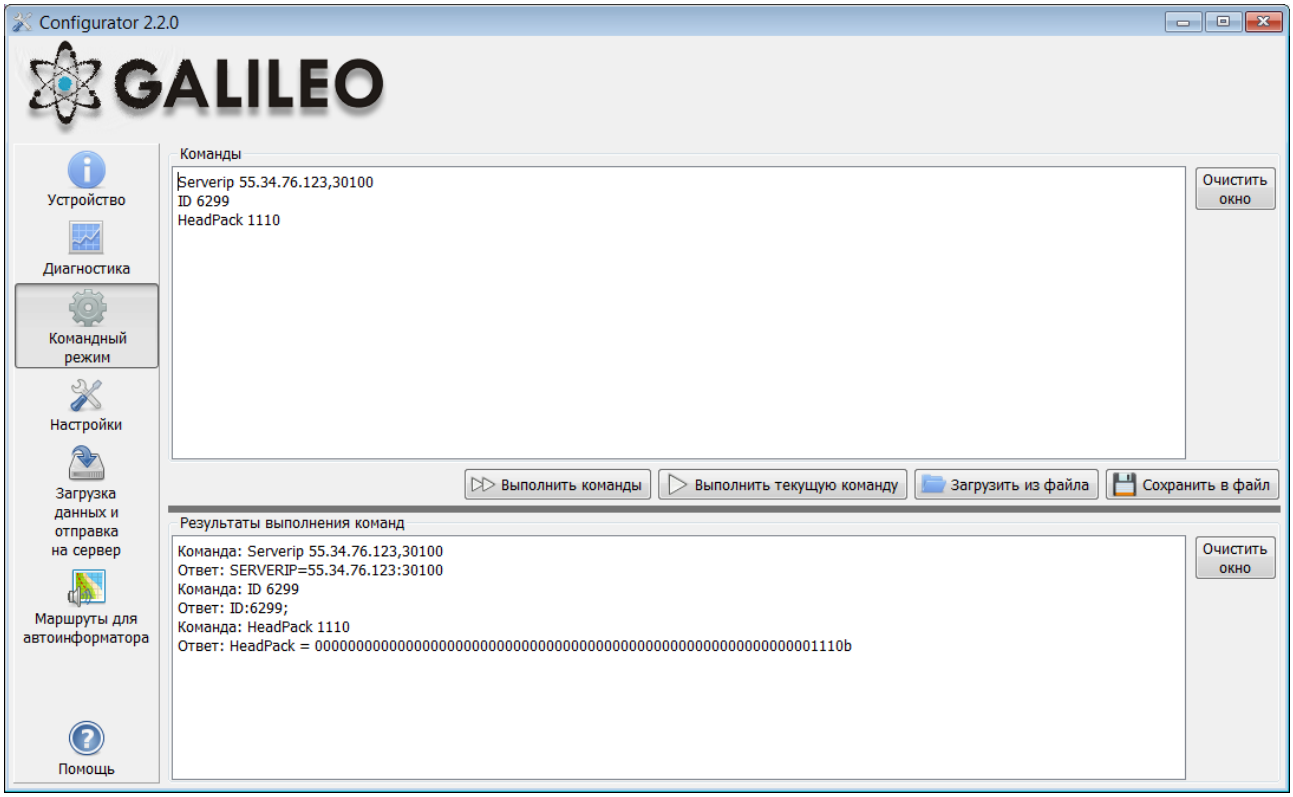
«APN» command	Request: **APN** Response: **GPRS:APN=INTERNET.BEELINE.RU, user=BEELINE, pass=BEELINE;**

3.5. Set of commands example

Enter the necessary commands in Command 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 = 00000000000000000000000000000000000000000000000000000000000001110b

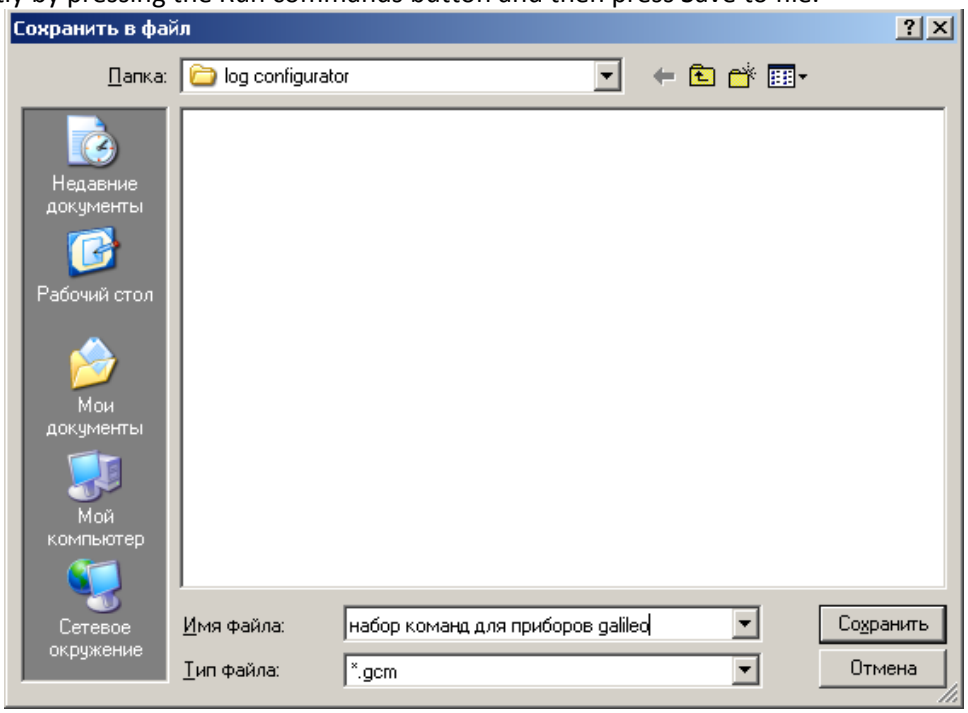


3.6. Example of saving and downloading parameter set

For quick configuration of several 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.

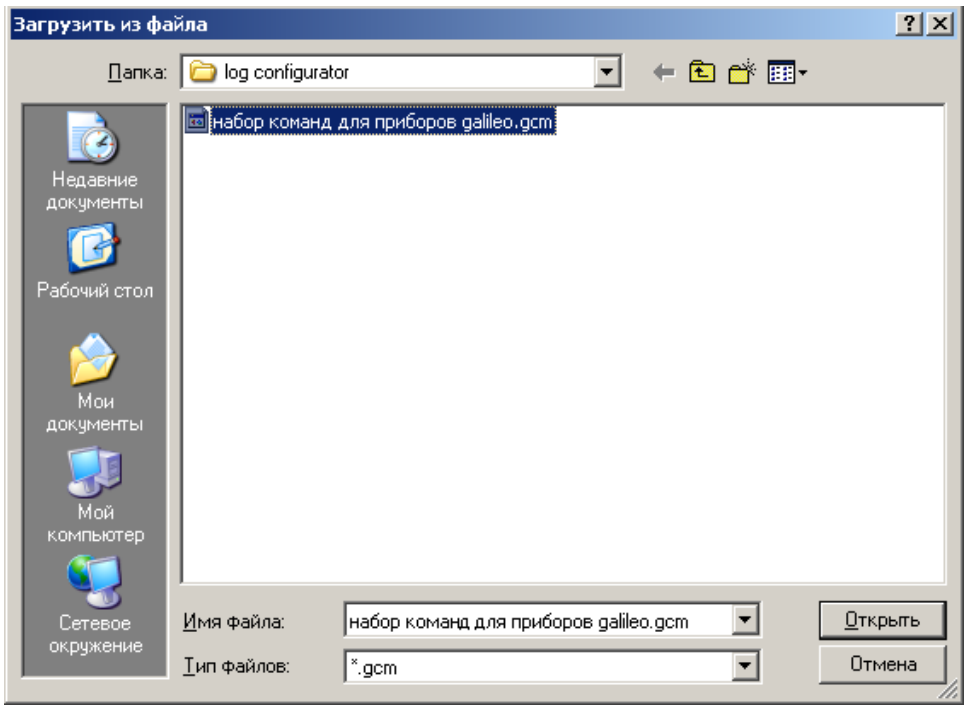
In the opened window, you will be offered to save the file in log configurator folder.

Type the file name and press Save button as shown in the figure on the right



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

Select the necessary file and press Open button, as shown in the figure on the right.



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.

3.7. Commands list

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

3.7.1. Settings for SMS control

Command format	**AddPhone xxxx[,n]**
Parameters	**xxxx** - is a 4 digit password, 1234 by default **n** – slot number (0-3) where a telephone number will be saved.
Explanation	When you configure the Terminal with 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)=89010123456 (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'

Формат команды	**Phones P1,P2,P3,P4**
Parameters	**P1,P2,P3,P4** – authorized phone numbers written in international format
Explanation	Getting and setting the list of authorized phones
Example	Request: Phones +7901012345,,, Reply: Phones (0)=+79010123456 (1)= (2)= (3)=

3.7.2. Data transmission settings

Command format	**APN a,u,p**
Parameters	**a** – access point name **u** – user **p** – password
Explanation	Access point settings for 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 SIM1
Example	Request: APN2 internet.beeline.ru,beeline,beeline Reply: GPRS2:APN=internet.beeline.ru, user=beeline, pass=beeline

Command format	**SIMSwitch mode**
Parameters	**mode** – SIM switching algorithm: 0 – only SIM 0 is used 1 – cyclic switching between the SIM-cards, if you cannot send the data for 9 minutes
Explanation	Setting of an algorithm of switching between the SIM-cards.
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. Old command syntax is also supported for indication of the IP-address: **Serverip ip1,ip2,ip3,ip4,port** **ip1, ip2, ip3, ip4** - IP server address.
Explanation	Main server parameters where the monitoring data will be transmitted.
Example	Starting from firmware 92 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 На всех прошивках 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. Old command syntax is also supported for indication of the IP-address: **Serverip2 ip1,ip2,ip3,ip4.port** **ip1, ip2, ip3, ip4** - IP server address.
Explanation	Additional server parameters.
Example	Request: Serverip2 m.7gis.ru,60521 Reply: Serverip2= m.7gis.ru: 60521

Command format	**ServersCfg t**
Parameters	**t** – Time of connection with one server, [sec]. At the value being equal to 0 the data will only be transmitted to the main server.
Explanation	Sets the time of server connection session.
Example	Request: ServersCfg 120 Reply SERVERSCFG:SeansTime=120;

Command format	**ID n**
Parameters	**n** - terminal number
Explanation	Changes device number.
Example	Request: ID 123 Reply: ID=123

Command format	**Roaming MCC_MNC,Size,Interval**
Parameters	**MCC_MNC** – a mobile code of the country where the data can be transmitted without any limitations (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 or it may be a combination of a country and a mobile codes. Zero means that there are no special roaming settings; **Size** – maximum number of bytes which can be transmitted during one connection session in roaming. At the value being equal to 0 only the first packet is transmitted; **Interval** – connections interval in hours.
Explanation	Settings of data transmission in international roaming.
Example	Request: Roaming 25099,10000,24 Reply: ROAMING:Home=25099,MaxBytes=10000,Interval=24;

Command format	**Sputnic t**
Parameters	**t** – time of connection with the help of satellite modem, [sec]. At the value being equal to 0 the, the satellite modem is not used.
Explanation	Sets the time of connection with the help of satellite modem.
Example	Request: Sputnic 1200 Reply: SPUTNIC: 1200;

3.7.3. Server exchange protocol settings

Command format	**HeadPack bbbbbbbbbbbbbb**
Parameters	**bbbbbbbbbbbbbb** - tag set 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 Table 2. GalileoSky protocol tags.
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	**HeadPackBit index,value**
Parameters	**index** –a tag number, which will be on or off for transmission to the server value – 1 if the tag should be transmitted to the server 0 if the tag should not be transmitted to the server Tag numeration order is given in Table 2. GalileoSky protocol tags.
Explanation	Head packet configuring.
Example	*Initially the second tag is off:* *HeadPack= 1100b* *Switch on this tag.* *Request: HeadPackBit 1,1* *Reply: HeadPack= 1110b*

Command format	**MainPack bbbbbbbbbbbbbb**
Parameters	**bbbbbbbbbbbbbb** – tag set 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 Table 2. GalileoSky protocol tags.
Explanation	Main packet configuring and configuring if the packet with data which is transmitted through satellite modem.
Example	*Request: MainPack 11111111111111111111110000* *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	**MainPackBit index,value**
Parameters	**index** –tag number, which will be 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 Table 2. GalileoSky protocol tags.
Explanation	Main packet configuring and configuring if the packet with data which is transmitted through satellite modem.
Example	*Initially the second tag is off:* *MainPack= 1100b* *Switch on this tag.* *Request: MainPackBit 1,1* *Reply: MainPack= 1110b*

Command format	**DataKey key**
Parameters	**Key** – data encryption key in hexadecimal form, if equal to 0 data are not encoded.
Explanation	Specifies the key that will encrypt the data transmitted.

3.7.4. Track parameters setting

Command format	**Turning V,A,D,S,dS**
Parameters	**V** – minimum speed that enables drawing of the track on the corners, [km/h]; **A** – minimum turn angle for device to record a track point, [º]; **D** – the distance above which another packet will be saved to the device memory, [m]; **S** –the speed above which for dS-multiple amount another 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 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/s²]; **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, is 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 device is on; **MinSatWork**– minimum number of satellites during operation of the Terminal. 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 will be updated, [sec].
Explanation	This function allows you to avoid 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; 7 – input 6 is used as ignition sensor; 8 – input 7 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 you to avoid outliers after the vehicle stops. Triggering on input is determined by the limits set by InCfg command (see Discrete analogue 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 incline determination off; **Timeout** – maximum allowable time when incline angle exceeded, [sec.]. **ShockSens** – maximum acceleration at exceed of which a strike is detected. 600 points – gravitational acceleration.
Explanation	Enabling strike and incline determination mode.
Example	*Request: Shock 3,30,5* *Reply: Shock: Mode=3,MaxAngle=30,RT=5;*

Command format	**Mhours LoLevel,HiLevel**
Parameters	**LoLevel** – input voltage + supply voltage at stopped engine, [mV]; **HiLevel** – input voltage +supply voltage at started engine, [mV];
Explanation	Allows filtering false coordinates after the vehicle stops.
Example	*Request: mHours 12000,14500* *Reply: Mclock: lolevel=12000,hilevel=14500;*

3.7.5. Geofences settings

Command format	**Dzone Mode**
Parameters	**Mode** – operation mode inside a geofence: 0 – geofence processing is switched off; 1 – coordinates updating prohibition inside a geofence; 2 – GSM unit disabling inside a geofence, 3 – updating prohibition and GSM unit disabling at the same time.
Explanation	Allows setting of terminal behavior inside a geofence.
Example	Request: Dzone 3 Reply: Dzone:3;

Command format	**DzoneAdd Lat,Lon,R**
Parameters	**Lat** – geofence centre latitude; **Lon** – geofence centre longitude; **R** – geofence radius in meters.
Explanation	Allows you to add a geofence. Each geofence is a circle with the determined center and radius.
Example	Request: DzoneAdd 55.9999,66.123456,100 Reply: DzoneAdd:lat=55.9999,lon=66.123456,rad=100;

Command format	**DzoneDel Lat,Lon**
Parameters	**Lat** – geofence center latitude; **Lon** – geofence center longitude.
Explanation	Allows you to delete the geofence, set by its center coordinates.
Example	Request: DzoneDel55.9999,66.123456 Reply: DzoneDel:lat-55.9999,lon=66.123456;

Command format	**DzoneClear**
Parameters	This parameter deletes all geofences.
Example	Request: DzoneClear Reply: Dead zones are cleared

Command format	**DzoneCount**
Parameters	This parameter allows you to get the number of all geofences set in the Terminal.
Example	Request: DzoneCount Reply: DZONECOUNT:2;

Command format	**DzoneInfo N**
Parameters	**N** – Ordinal number of geofence beginning with 0.
Explanation	This parameter gives an access to the geofence's data.
Example	Request: DzoneInfo 0 Reply: DZONEINFO:Lat=10.000000,Lon=20.000000,Rad=30;

3.7.6. Information commands

Command format	**Status**
Parameters	Allows finding device status at the moment of sending a command Dev – this device number; Soft – current firmware version number; Pack – Last recorded serial packet number; TmDt – Current time and date; Per – Current packet saving period (different when the vehicle is moving or stops); Nav – Coordinates accuracy. 0 – coordinates 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**
Parameters	Allows us to obtain a unique GSM unit identifier,15byte
Example	Request: IMEI Reply: IMEI 123456789012345

Command format	**imsi**
Parameters	Allows us to obtain a unique IMSI identifier of the SIM-card
Example	Request: IMSI Reply: IMSI 123456789012345

Command format	**inall**
Parameters	Allows analog input values in0..in5 to be obtained as well as accelerometer values with respect to three axes (10bit for each axis starting with the zero bit).
Example	Request: inall Reply: INALL:in0=0,in1=0,in2=0,in3=0,in4=0,in5=0,Acc=332943891;

Command format	**Insys**
Parameters	Allows us to obtain external source voltage, internal battery voltage, GPS aerial voltage, the main power bus voltage and also the temperature inside the device.
Example	Request: insys Reply: INSYS: Pow=12438,Vbat=4196,Vant=2921,Vdc=4115,Temper=37

Command format	**RS485**
Parameters	Allows you to receive a value of the digital fuel level sensor connected on the RS485 interface. The first parameter corresponds to fuel level sensor with address №-0, the second – with address №1, the third – with address №2.
Example	Request: RS485 Reply: RS485 100,25,0;

Command format	**Temex0**
Parameters	Allows finding temperature from the first four external thermometers DS18S20. Format: the lower byte is the thermometer identifier; the high 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**
Parameters	Allows finding temperature of the second four external thermometers DS18S20. Format: the lower byte is the thermometer identifier; the high 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	**Hum0**
Parameters	Allows you to find the temperature of the first four DS1923 humidity sensors. ID – sensor identifier; T – temperature in Celsius degrees H – Humidity in %.
Example	Request: Hum0 Reply: Hum0:ID0=1,T0=20,H0=20.0,ID1=2,T1=30,H1=30.0,ID2=3,T2=25,H2=40.0,ID3=5, T3=15,H3=50.0;

Command format	**Hum1**
Parameters	Allows you to find the temperature of the second four DS1923 humidity sensors. ID – sensor identifier; T – temperature in Celsius degrees H – Humidity in %.
Example	Request: Hum1 Reply: Hum1:ID4=1,T4=20,H4=20.0,ID5=2,T5=30,H5=30.0,ID6=3,T6=25,H6=40.0,ID7=5, T7=15,H7=50.0;

Command format	**Canibut**
Parameters	Allows current CAN bus state (Table 2. GalileoSky protocol tags) and iButton decimal value to be obtained.
Example	Request: canib Reply: CAN_Ib: CANA0=0,CANA1=0,CANB0=0,CANB1=0,iBut=0

Command format	**statall**
Parameters	Allows device, inputs, outputs decimal status to be obtained (Table 3. Device status field explanation) and mileage according to GPS/GLONASS data.
Example	Request:statall Reply: StatAll: Dev=1,Ins=2,Outs=7,Mileage=152;

Command format	**AccType**
Parameters	Allows obtaining accelerometer type. Returns "analog" for analogue type and "digital" for digital type.
Example	Request: AccType Reply: AccType: digital

3.7.7. Service commands

Command format	**PIN N**
Parameters	**N** – four-digit PIN-code of a SIM card.
Explanation	SIM card PIN-code and password setting for access settings in the Configurator. The default PIN-code is 0. If you enter the wrong code, the terminal will be blocked for 25 seconds, and then restart. PIN-code is identical for both SIM-cards.
Example	Request: PIN 1234 Reply: PIN:1234;

Command format	**Vibro OnOff**
Parameters	**OnOff** – 0 – vibration analysis is off; 1 – vibration analysis is on.
Explanation	Enabling the mode of vibration analysis. After configuration, you must restart the Terminal.
Example	Request: VIBRO 0 Reply: VIBRO:0;

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 restart the terminal. Before you select a microSD card, delete the archive created by earlier firmwares (EraseTrackSD or delete the file from Track catalogue through the Card-Reader).
Example	Request: ARCHIVE 0 Reply: ARCHIVE:0;

Command format	**FLASHARCHIVE Dynamic,SendOrder**
Parameters	**Dynamic** – whether the dynamic archive structure is off or on: 0 – the dynamic archive structure is off, all possible data are saved in archive; 1 – the dynamic archive structure is on, only the data selected to be transmitted to the server are saved in archive. **SendOrder** – order of data transmission: 0 – the data are sent deep into the archives; the most actual data are sent first 1 – the data are sent in chronological order
Explanation	Archive structure setting and the setting of the data transmission order to the server
Example	Request: FLASHARCHIVE 1,1 Reply: FLASHARCHIVE:Dynamic=1,StraightSendOrder=1;

Command format	**EraseCfg**
Explanation	Restoring 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	**GlonassColdStart**
Explanation	GLONASS unit cold start on Galileo GLONASS terminals.
Example	Request: GlonassColdStart Reply: GLONASS cold start

Command format	**GPSColdStart**
Explanation	GPS unit cold start on Galileo GPS terminals.
Example	Request: GpsColdStart Reply: GPS cold start

Command format	**Reset**
Explanation	Allows you to reset the device remotely.
Example	Request: Reset Reply: Reset of device. Please wait 15 seconds…

Command format	**Upgrade**
Explanation	See Bootloader section.
Example	Request: Upgrade 47 Reply: UPGRADE 47

Command format	**SleepMode OffGNSSOnStop,DSTime,GNSS,GPRS,ADC,CAN,0,RS485,SD**
Parameters	**OffGNSSOnStop** – 0 – do not turn GPS\GLONASS unit off at a stop; 1 – turn GPS\GLONASS unit off at a stop. **DSTime** – Time spent at a stop, after which the Terminal will switch to the deep sleep mode; **GNSS** – turn GPS\GLONASS unit off in the deep sleep mode; **GPRS** –turn GSM unit off in the deep sleep mode; **ADC** – reduce ADC sampling rate in the deep sleep mode; the maximum frequency that can be measured at inputs is reduced by 2 and the minimum impulse period that can be registered increases 2-fold; **CAN** – turn off the CAN in the deep sleep mode; **RS485** – turn off RS485 in the deep sleep mode; **microSD** – turn off the microSD card in the deep sleep mode, the reading of trusted iButton keys is supported;
Explanation	Power saving mode control. In the deep sleep mode no 1Wire sensors sampling and no battery charging are performed.
Example	Request: SLEEPMODE 1,60,1,1,1,1,0,1,1 Reply: SLEEPMODE:OffGNSSOnStop=1,DSTimeout=60, GNSS=1,GPRS=1,ADC=1,CAN=1,RS2320=0,RS232_1/RS485=1,SD=1;

3.7.8. 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 to be 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 disabled.
Explanation	An incoming call results in a device automatic answer.
Example	Request: AutoAnswer 1 Reply: AUTOANSWER=1

Command format	**Calls N**
Parameters	**N** – the number of call attempts
Example	Request: Calls 3 Reply: CALLS:3;

Command format	**RingTo N**
Parameters	**N** – a telephone number
Explanation	This command results in the device dialing the given number.
Example	Request: RingTo 89119988899 Reply: RINGTO=89119988899

Command format	**SendSMS Tel,Msg**
Parameters	**Tel** – a telephone number to which SMS will be sent **Msg** – SMS template. It may contain parameters to insert current data: %IMEI – device's IMEI, %LAT – latitude, %LON – longitude.
Explanation	This command results in the device sending SMS to the given number.
Example	Request: SendSMS 89119988899,Test Reply: SMS sheduled

3.7.9. Discrete-analog input 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 3 – pulse count from two synchronous connected sensors. **fl** – filter length [1÷50]. 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 operation.
Explanation	Allows one of 8 analog/discrete inputs to be 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	**AccVal**
Explanation	Obtaining filtered accelerometer mean-square value by three axes. Accelerometer sensitivity: min = 555mV/g; average = 600мВ/g; max = 645мВ/g; Where g is gravitational acceleration (g≈9.8м/c^2).
Example	Request: AccVal Reply: ACCVAL = 625 AccVal = 0.625В. As you can see, the accelerometer is affected only by the gravity force.

3.7.10. Transistor output setting

Command format	**Out v,s**
Parameters	**v** – output ordinal number (starting with the zero); **s** – Desired state (0 – on-state transistor output; 1 –off-state transistor output).
Explanation	Transistor output control. With one output is being controlled, the others outputs' state remains unchanged. Transistor outputs are off by default.
Example	Request: Out 1,1 Reply: OUT(3..0) = 0010 As you can see all outputs except 1 are on.

3.7.11. Autoinformer setting

Command format	**Autoinformer OnOff,Repeat,Out,FileName**
Parameters	**OnOff** – enable/disable Autoinformer function: 1 – the function is enabled, 0 – the black box function is enabled (the navigation data is duplicated and saved to the external micro-SD 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. **FileName** – a route name. The route is understood as the number of zones to be announced.
Explanation	For more information see Autoinformer section.
Example	Request: Autoinformer 1,0,0,Marshrut 1 Reply: AUTOINFORMER:OnOff=1,Repeat=0,Out=0,Rout=Marshrut 1;.

3.7.12. Digital inputs settings

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 00 00 00 91 02 0C 5C, where 09 – type of device (in this case DS1982, for DS1990 - 01), 00 00 00 91 02 0C – unique number, 5C – the checksum. In this case 00 91 02 2C must be entered.
Explanation	List of iButton identifiers which state will be monitored by terminals.
Example	Request: iButtons 0091022C,0,0,0,0,0,0,0 Reply: IBUTTONS:0091022C,0,0,0,0,0,0,0;

Command format	**AddKey key**
Parameters	**key** – the lower 4 bytes of the iButton identification number excluding the checksum in hexadecimal form. For example, the full key number in hexadecimal form is: 09 00 00 00 91 02 0C 5C, where 09 – the type of device (in this case it is DS1982, for DS1990 - 01), 00 00 00 91 02 0C – the unique number, 5C – the checksum. In this case, the number to be entered is 00 91 02 2C.
Explanation	To add a key to the list of trusted iButton identifiers on a micro-SD-card.
Example	Request: AddKey 0091022C Reply: Key 0091022C added

Command format	**DelKey key**
Parameters	**key** – the lower 4 bytes of the iButton identification number excluding the checksum in hexadecimal form. For example, the full key number in hexadecimal form is: 09 00 00 00 91 02 0C 5C, where 09 – the type of device (in this case it is DS1982, for DS1990 - 01), 00 00 00 91 02 0C – the unique number, 5C – the checksum. In this case, the number to be entered is 00 91 02 2C.
Explanation	To remove a key from the list of trusted iButton identifiers on a micro-SD-card.
Example	Request: DelKey 0091022C Reply: Key 0091022C deleted

Command format	**KeyCount**
Explanation	The number of trusted iButton keys on a micro-SD card.
Example	Request: KeyCount Reply: KEYCOUNT:12;

Command format	**ShowKey N**
Parameters	**N** – The order number of an iButton key in the list on a micro-SD-card. Numbering starting from 1.
Explanation	To show an identifier of a trusted iButton key on a micro-SD-card.
Example	Request: ShowKey 1 Reply: SHOWKEY 1:9503276 (0x0091022C)

3.7.13. Signaling mode setting

Command format	**SIGN GWTime,DropAlarmTimeout, UseIB**
Parameters	**GWTime** – duration of the "green wave" which is the time after the signalling starts during which no sampling of sensors takes place, [sec]; **DropAlarmTimeout –** time in the alarm mode, after which an automatic switching to the alarm mode will be performed. At zero, the Terminal will stay in the signalling mode until the command is sent or until it is disabled using the input, [sec]; **UseIB –** should iButton keys be used for arming and disarming: 0 – no; 1 – arming and disarming through the short-time hold of one of the trusted iButton keys; 2 – arming only having one of the trusted iButton keys, if no key is put, the Terminal is disarmed; 3 – arming only having any of the iButton keys, if no key is put, the Terminal is disarmed. 4 – disarming only having any of the iButton keys, if no key is put, the Terminal is armed;
Explanation	General configurations of signalling.
Example	Request: SIGN 40,60,0 Reply: SIGN:GWTime=40,DropAlarmTimeout=60 ,UseIB=0;

Command format	**S**
Explanation	Arming.
Example	Request: S Reply: Signaling is enabling

Command format	**DS**
Explanation	Disarming.
Example	Request: DS Reply: Signaling is disabling

Command format	**ST**
Explanation	Signaling status. Possible Statuses: Signaling is disabled – signaling is off, Signaling is enabled – signaling is on, Alarm – alarm mode.
Example	Request: ST Reply: Signaling is disabled

Command format	**AddSigPhone phone[,n]**
Parameters	**phone** – a phone number **n** – an optional parameter, added phone number index.
Explanation	Notification phone setting
Example	Request: AddSigPhone 123456789 Reply: SignPhones 123456789;;;;

Command format	**SIN0 type,delay,sms,ring,photo,msg**
Parameters	**type** – an input mode: • 0 – is not used for signaling; • 1 – input activation results in signaling mode on; • 2 – input activation results in alert mode on if signaling mode is on; • 3 – input activation results in alert mode on even if signaling mode is off. **delay** – post-activation delay before alert mode, [sec]. **sms** – enable SMS notification: 1 – yes, 0 – no **ring** – enable phone call notification: 1 – yes, 0 – no **photo** – take photo: 1 – yes, 0 – no **msg** – alert mode message. The message may contain the parameters which will be replaced by the current data: %IMEI – IMEI of the terminal, %LAT – latitude, %LON – longitude.
Explanation	Setting the behavior of an input in signaling mode.
Example	Request: SIN0 3,0,1,1,0,Alarm %IMEI Reply: SIN0:SignType=3,Adelay=0, SMS=1,Ring=1,Photo=0,Msg=Alarm %IMEI;

sin1, sin2, sin3, sin4, sin5, sin6, sin7 – similar to sin0 commands

Command format	**SGPS type,speed,r,t,sms,ring**
Parameters	**type** – operating mode • 0 – is not used for signaling; • 1 – alert mode when the speed is exceeded; • 2 – alert mode if the vehicle is beyond the radius longer than a predetermined time interval; • 3 – alert mode when the speed is exceeded or if the vehicle is beyond the radius longer than a predetermined time interval. **speed** – maximum speed, [km/h]. **r** – maximum radius, [m]. **t** – maximum time interval out of the radius, [s]. **sms** – enable SMS notification: 1 – yes, 0 – no **ring** – enable phone call notification: 1 – yes, 0 – no
Explanation	Setting of the use of GPS data in signaling mode
Example	Request: sgps 1,10,1,10,1,1 Reply: SGPS:SignType=1,Speed=10,R=1,T=10,SMS=0,Ring=0;

Command format	**SACC type,sms,ring,photo,msg**
Parameters	**type** – operation mode: • 0 – is not used for signaling; • 1 – an incline more than the given angle results in Alarm in Signaling mode; • 2 –acceleration exceeding (strike) results in Alarm in Signaling mode; • 3 – both an incline and a strike results in Alarm in Signaling mode. **sms** – enable SMS notification: 1 – yes, 0 – no **ring** – enable phone call notification: 1 – yes, 0 – no **photo** – take photo: 1 – yes, 0 – no **msg** – alert mode message. The message may contain the parameters which will be replaced by the current data: %IMEI – IMEI of the terminal, %LAT – latitude, %LON – longitude.
Explanation	Using accelerator data in signaling mode. Operation thresholds are set by SHOCK command (see Determination of strike and incline)
Example	Request: SACC 2,1,1,0,Удар Reply: SACC:SignType=2,SMS=1,Ring=1,Photo=0,Msg=Удар;

Command format	**SOUT0 EMode,EImpT,EImpC,DMode,DImpT,DImpC,AMode,AImpT,AImpC,ADelay**
Parameters	**EMode** – output mode at arming: • 0 – no reaction, • 1 – output is inverted, • 2 – output generates impulses, **EImpT** – impulse time during Arming, msec. **EImpC** – number of impulses during Arming. **DMode** – output mode at Disarming: • 0 – no reaction; • 1 – output is inverted; • 2 – output generates impulses. **DImpT** – impulse time during Disarming, msec. **DImpC** – number of impulses during Disarming **AMode** – output mode at Alarm: • 0 – no reaction; • 1 – output is inverted; • 2 – output generates impulses. **AImpT** – impulse time during Alarm, msec **AImpC** – number of impulses during Alarm. **ADelay –** activation delay after Alarm mode is on, in sec. The Device rounds impulse duration to 0.1 sec
Explanation	Setting the behavior of an output in signaling mode.
Example	Request: SOUT0 2,1,1,2,2,2,1,0,0,20 Reply: SOUT0:EMode=2,EImpT=1,EImpC=1,DMode=2,DImpT=2,DImpC=2,AMode=1, AImpT=0,AImpC=0, ADelay=20;

sout1, sout2, sout3 – similar to sout0 commands

3.7.14. CAN settings

Command format	**CanRegime Mode,BaudRate,TimeOut**
Parameters	**Mode** – operating mode: • 0 – CAN interface is off and is not used; • 1 – CAN bus scanner; • 2 – standard FMS filter; • 3 – user filter29 bit; • 4 – user filter 11 bit **BaudRate** – bus rate. It must be the same as the vehicle bus rate. It can have the following values: from 10000 up to 500000. Typical valuations: 62500, 125000, 250000, 500000. **TimeOut** – measured in msec. For CAN_SCANER mode it is response latency. If it is too small, not all bus messages will be received. The recommended time for CAN_SCANER 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.
Explanation	General CAN bus control.
Example	Example: Enable 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;

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. Confirmation sending may be necessary at connection to the troubleshooting socket if the data cannot be read in passive mode.
Example	Request: ActiveCAN 1 Reply: ACTIVECAN:1;

Command format	**CAN8BitR0 ID,Shift**
Parameters	**ID** – captured bus identifier: **Shift** – useful data shift in the received packet
Explanation	Single CAN-tag content control.
Example	Request: Can8BitR0 419360256,1 Reply: CAN8BITR0:ID=419360256,Shift=1;

Commands: **CAN8BitR1, …, CAN8BitR7, CAN16BitR0, …, CAN16BitR4, CAN32BitR0, …, CAN32BitR4 –** are similar to CAN8BitR0 command.

3.7.15. Packet transmission, energy saving, Stels mode settings

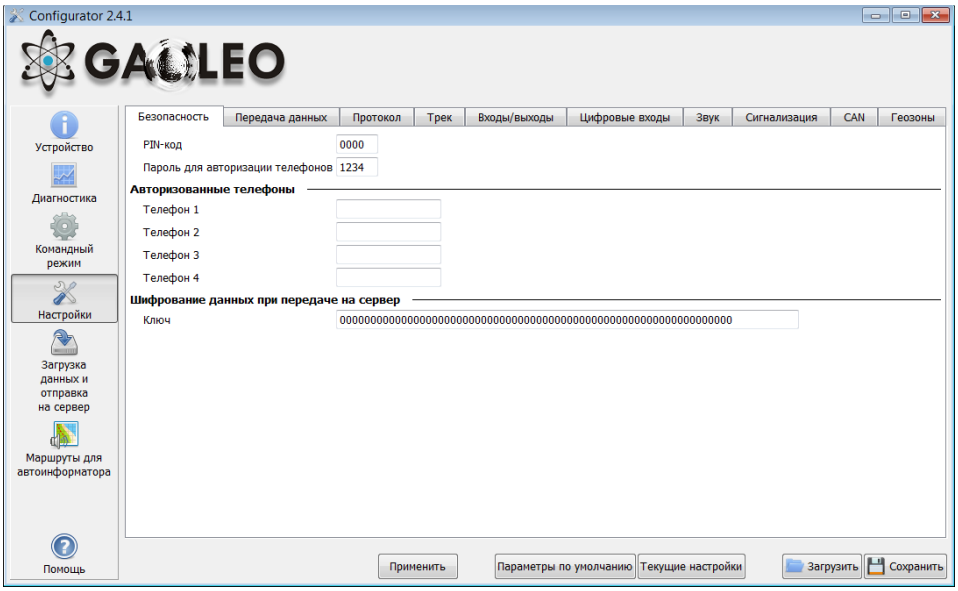
Command format: **Stels pday, phours, minutesGSMOn**
See **Stels mode and packet transmission** section.

4. Graphic interface settings

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

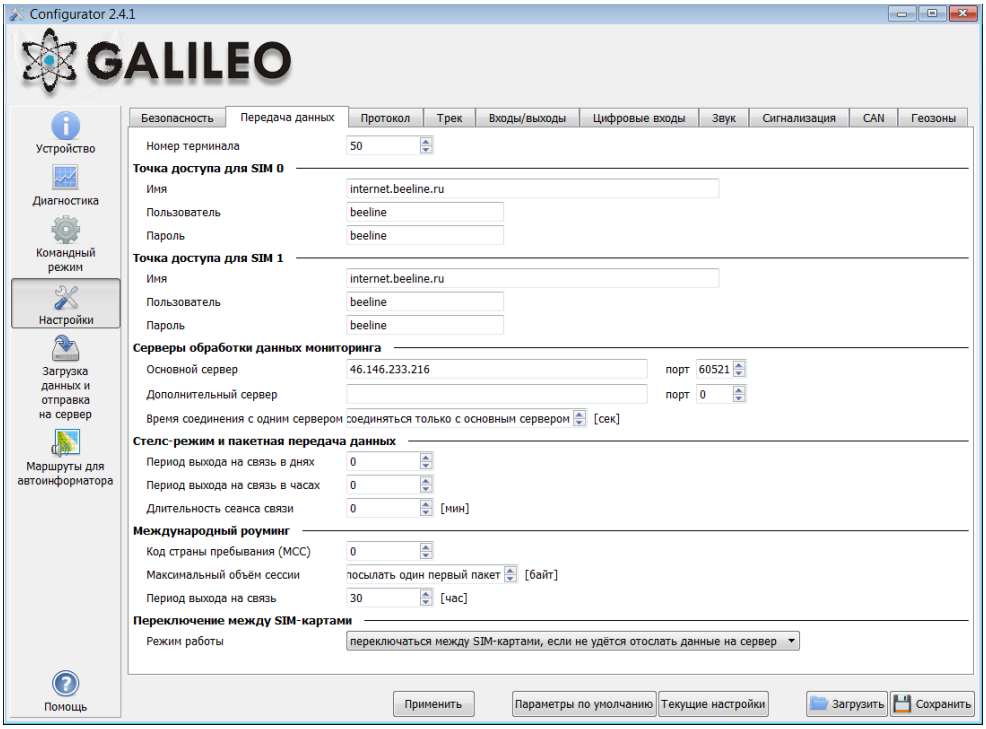
4.1. Security

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



4.2. Data transmission

This tab allows to set SIM-card PIN code, APN for Internet access, monitoring data processing servers, batch data transmission and international roaming.



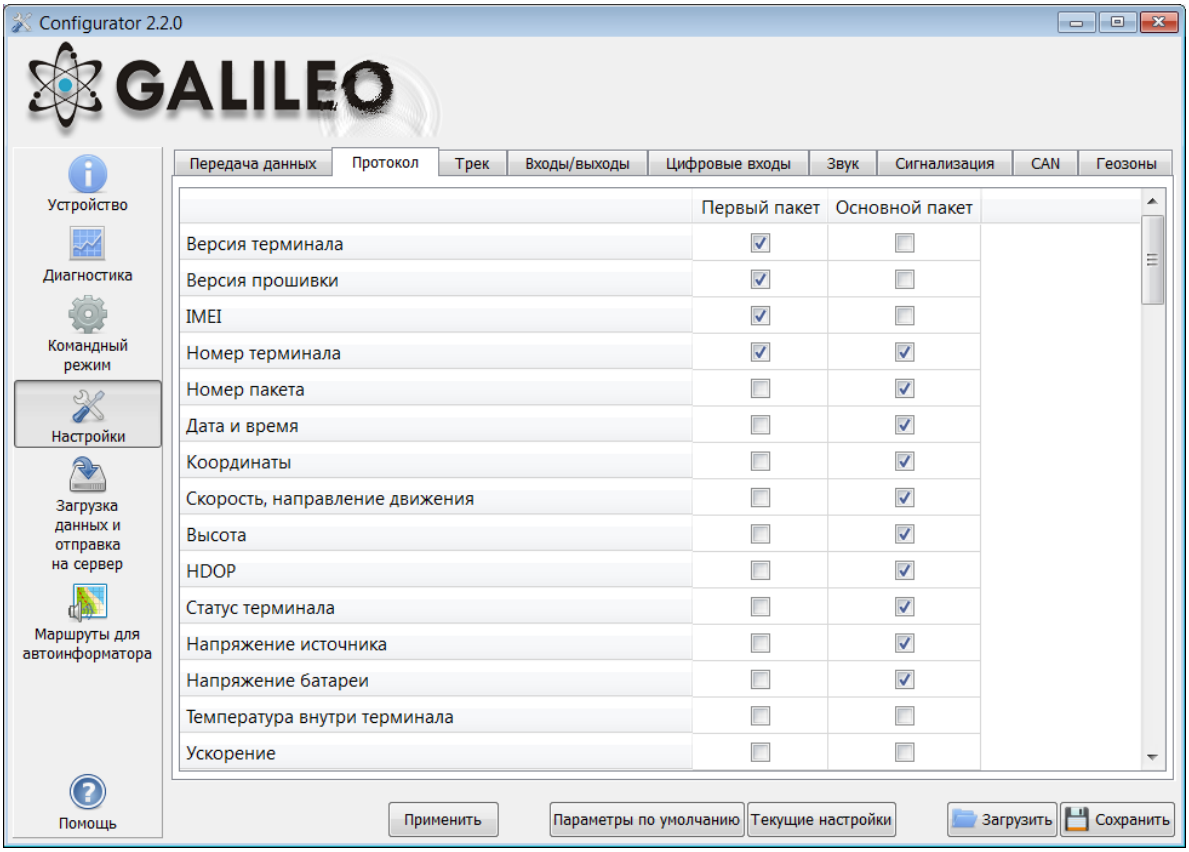
4.3. Protocol

The device has its own data transmission protocol developed by GalileoSky Ltd.

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

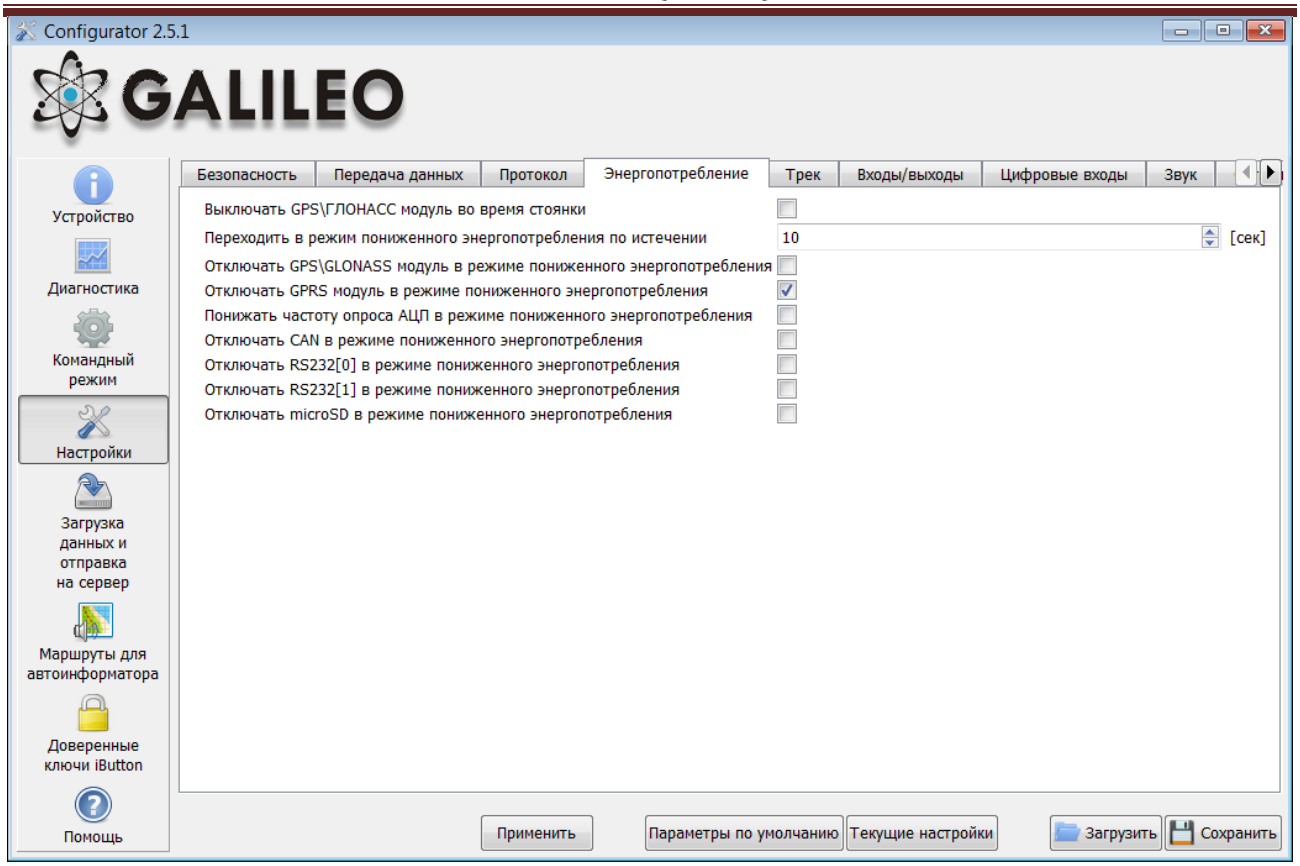
1) Initialization of TCP/IP connection (does not need any additional settings);
2) Sending of initialization data described in the Head packet column (the data to be sent to the server are marked in the first column);
3) If the device has passed the first two stages, accumulated packets begin to send 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.



4.4. Power saving

This tab allows setting shutdown options of the Terminal units at stop, to reduce power consumption.



4.5. Track

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

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

In addition the terminal 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 Vpit voltage parameter in "Device" tab.

The second parameter is selected in the following way:

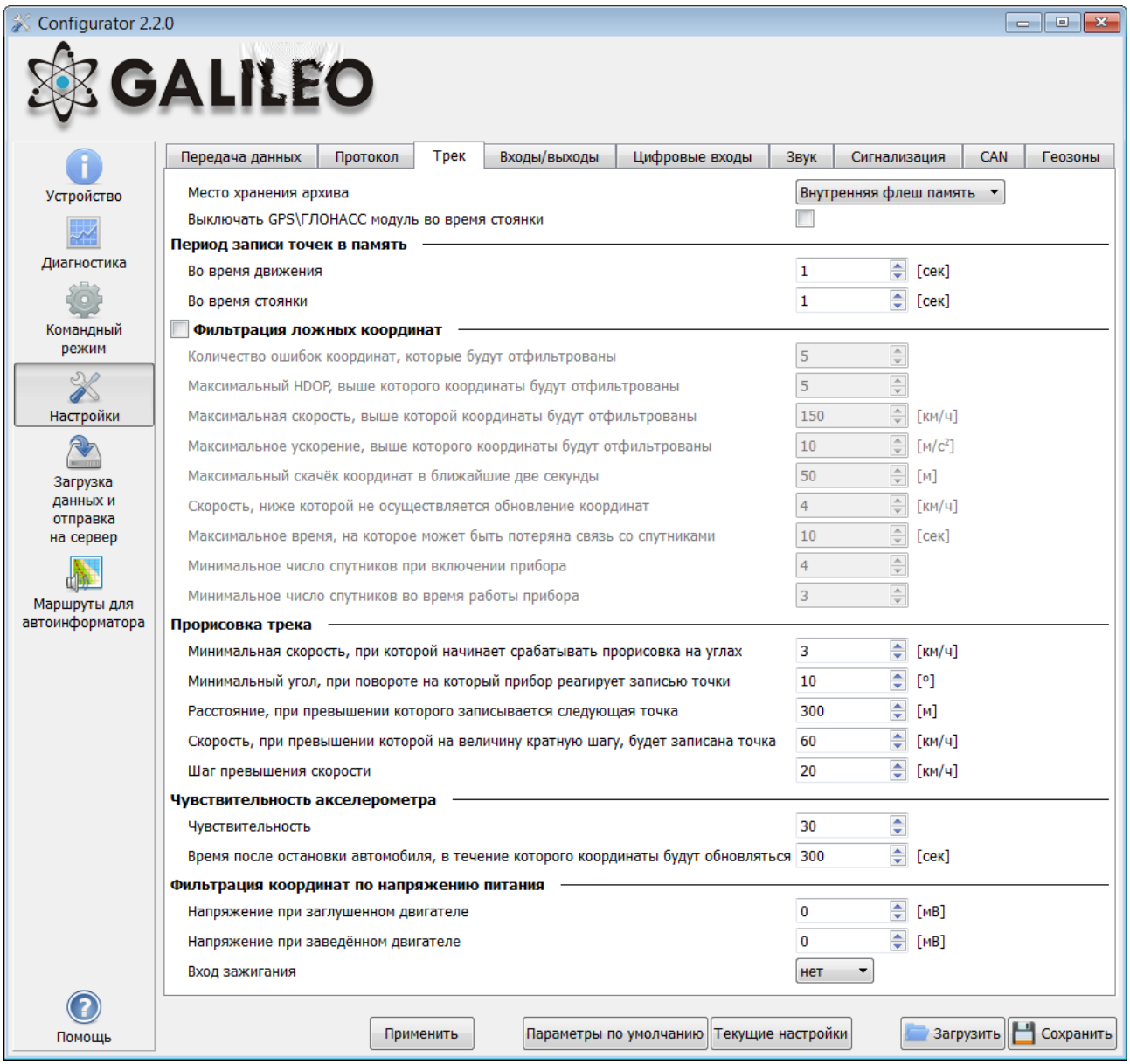
1) start the engine;
2) save the Vpit parameter;
3) parameters of the mhours command are filled in and sent to the terminal.

When the engine is started, the 9th bit will be set in the device status (Table 3. Device status field explanation).

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

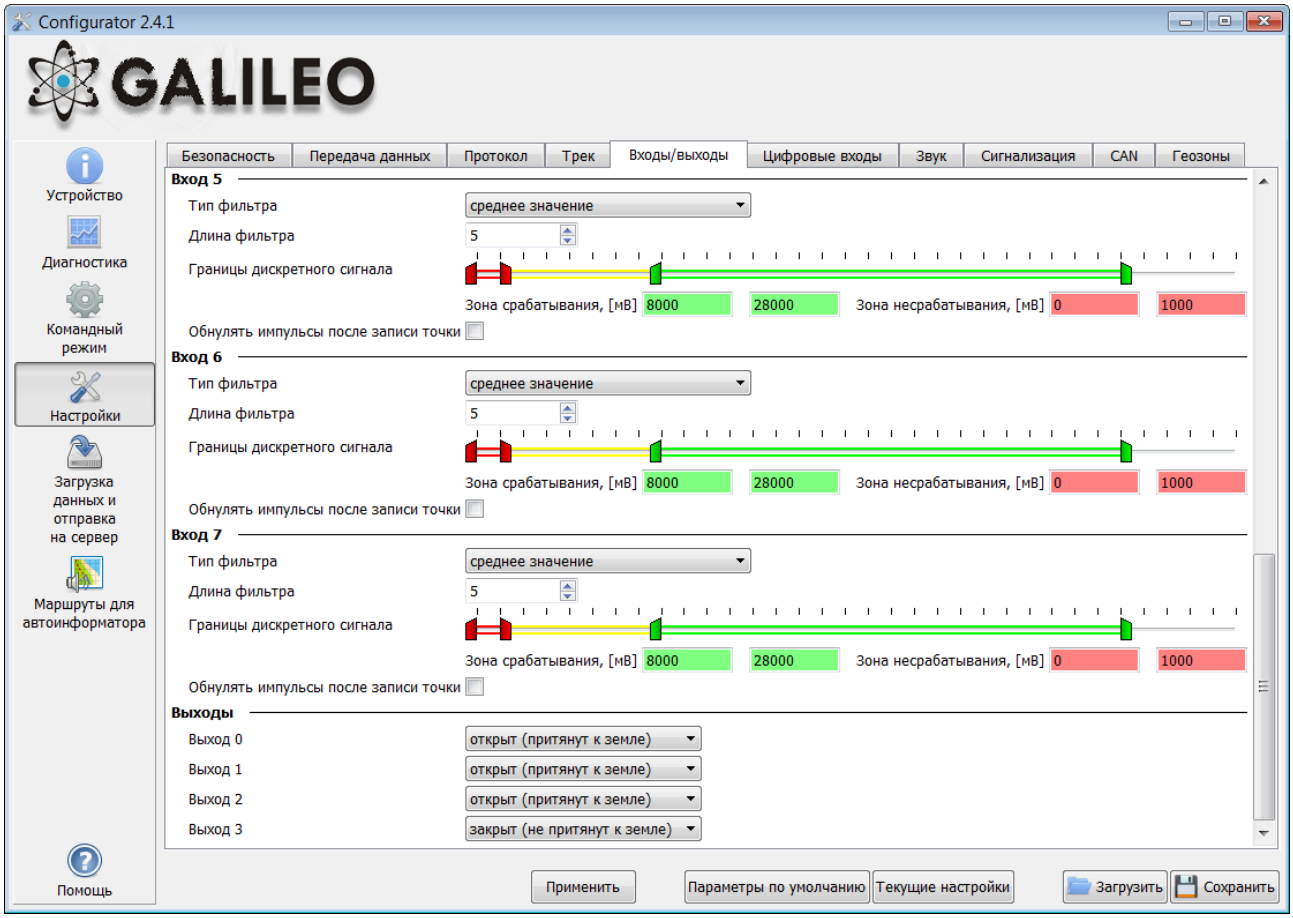
Parameters:

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


Configurator 2.2.0
GALILEO
Устройство
Диагностика
Командный режим
Настройки
Загрузка данных и отправка на сервер
Маршруты для автоинформатора
Помощь
Передача данных
Протокол
Трек
Входы/выходы
Цифровые входы
Звук
Сигнализация
CAN
Геозоны
Место хранения архива
Внутренняя флеш память
Выключать GPS\ГЛОНАСС модуль во время стоянки
Период записи точек в память
Во время движения
1
[сек]
Во время стоянки
1
[сек]
Фильтрация ложных координат
Количество ошибок координат, которые будут отфильтрованы
5
Максимальный HDOP, выше которого координаты будут отфильтрованы
5
Максимальная скорость, выше которой координаты будут отфильтрованы
150
[км/ч]
Максимальное ускорение, выше которого координаты будут отфильтрованы
10
[м/с²]
Максимальный скачёк координат в ближайшие две секунды
50
[м]
Скорость, ниже которой не осуществляется обновление координат
4
[км/ч]
Максимальное время, на которое может быть потеряна связь со спутниками
10
[сек]
Минимальное число спутников при включении прибора
4
Минимальное число спутников во время работы прибора
3
Прорисовка трека
Минимальная скорость, при которой начинает срабатывать прорисовка на углах
3
[км/ч]
Минимальный угол, при повороте на который прибор реагирует записью точки
10
[°]
Расстояние, при превышении которого записывается следующая точка
300
[м]
Скорость, при превышении которой на величину кратную шагу, будет записана точка
60
[км/ч]
Шаг превышения скорости
20
[км/ч]
Чувствительность акселерометра
Чувствительность
30
Время после остановки автомобиля, в течение которого координаты будут обновляться
300
[сек]
Фильтрация координат по напряжению питания
Напряжение при заглушенном двигателе
0
[мВ]
Напряжение при заведённом двигателе
0
[мВ]
Вход зажигания
нет
Применить
Параметры по умолчанию
Текущие настройки
Загрузить
Сохранить

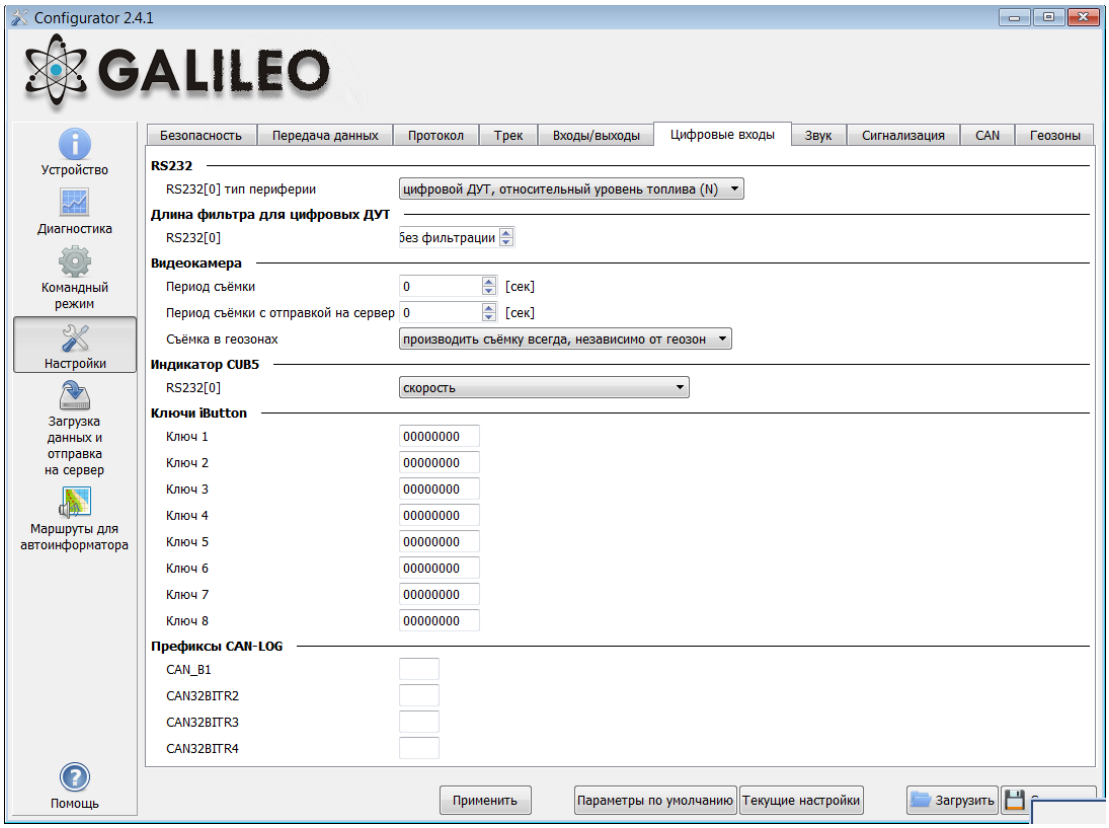
4.6. Inputs/ Outputs

For inputs operating principles see Discrete analog input (DAI) operation.
For discrete inputs description see Transistor outputs (0/1).



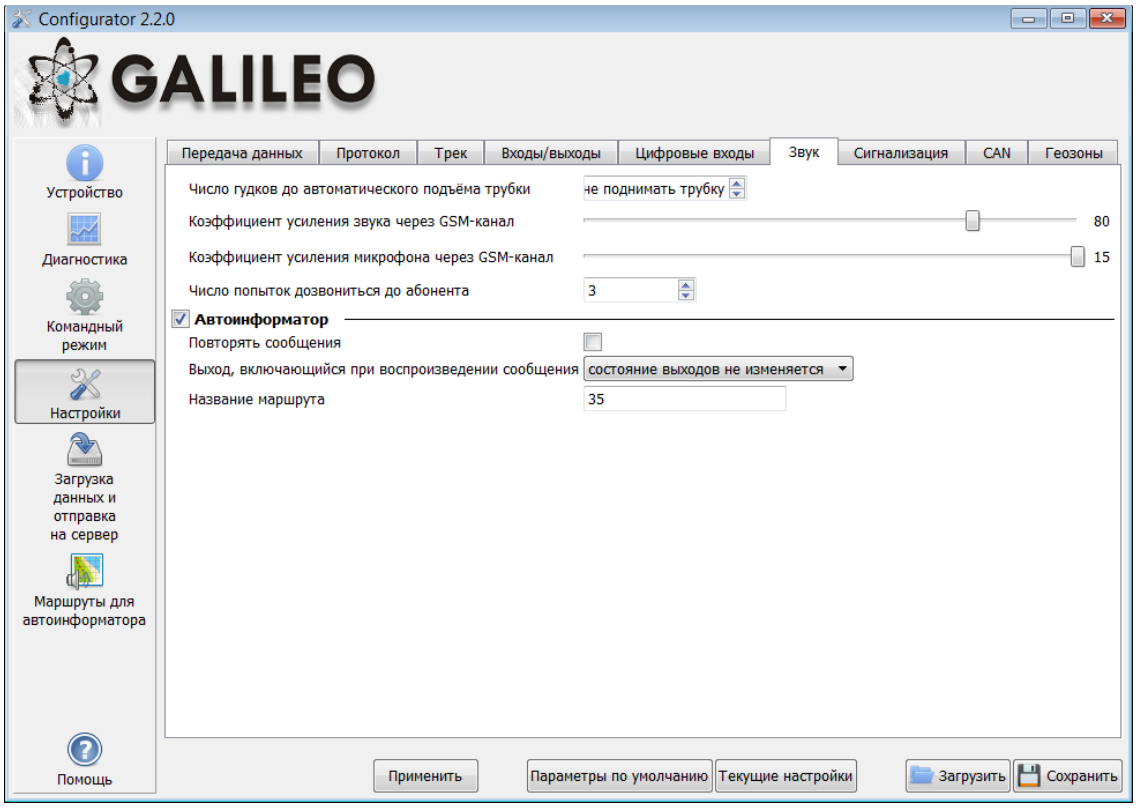
4.7. Digital inputs

This option allows setting iButton keys, to setup periodical photo camera shooting.



4.8. Sound

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

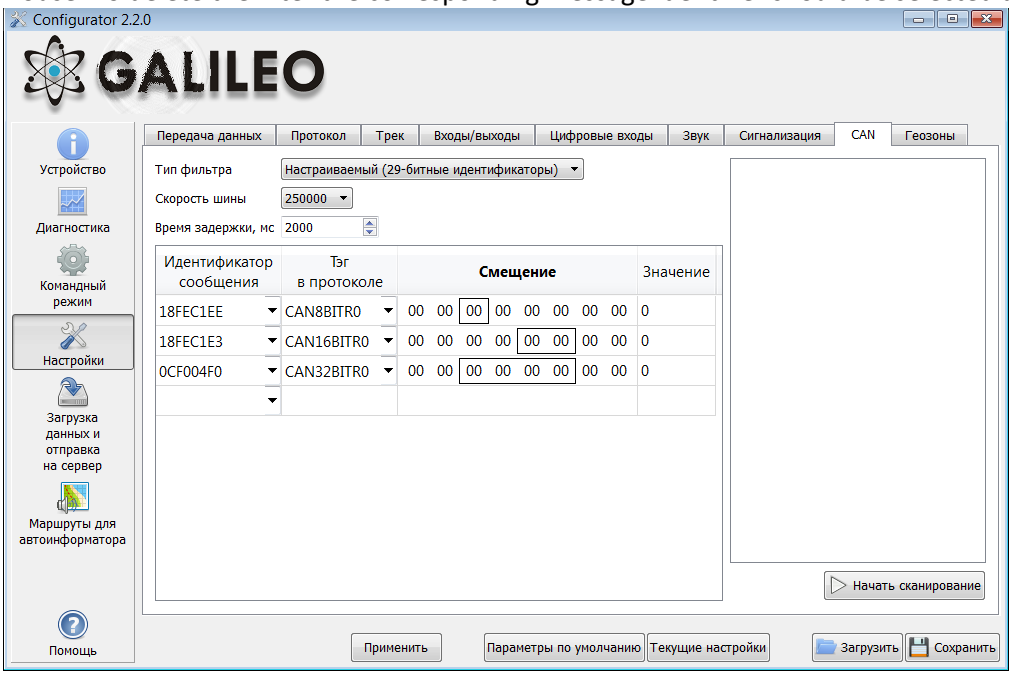


4.9. Signaling

This option allows you to set the response of the Terminal to inputs state change, speed and coordinate change. You can set telephone numbers for SMS or call in case of signaling actuation. Strike and incline determination may also be set here.

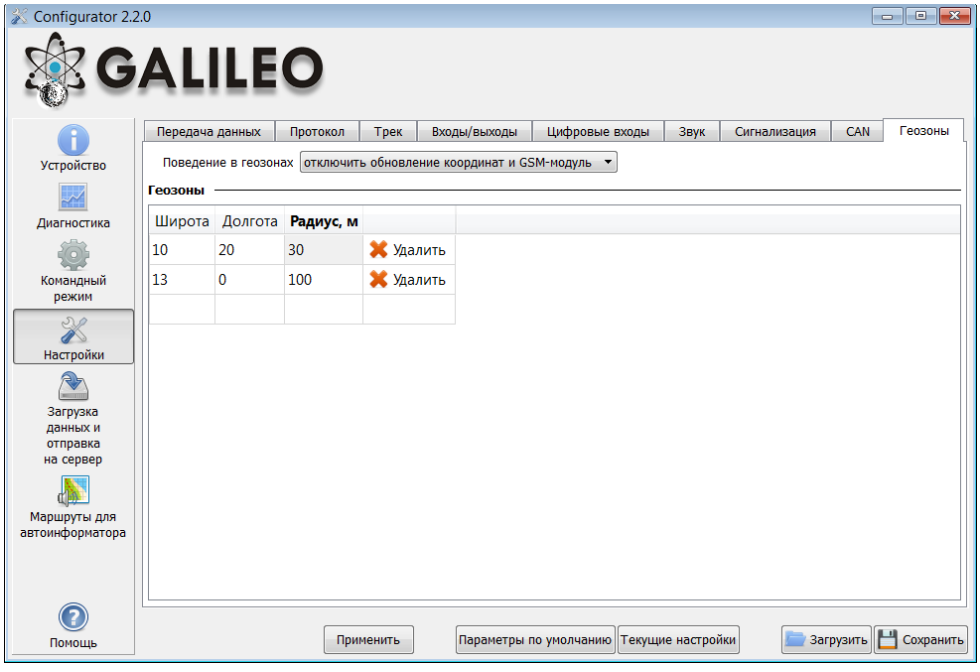
4.10. CAN

This option allows you to set a CAN filter and to scan the CAN bus for identifiers being used.
After clicking "Listen scan" 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 where 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.



4.9. Geofences

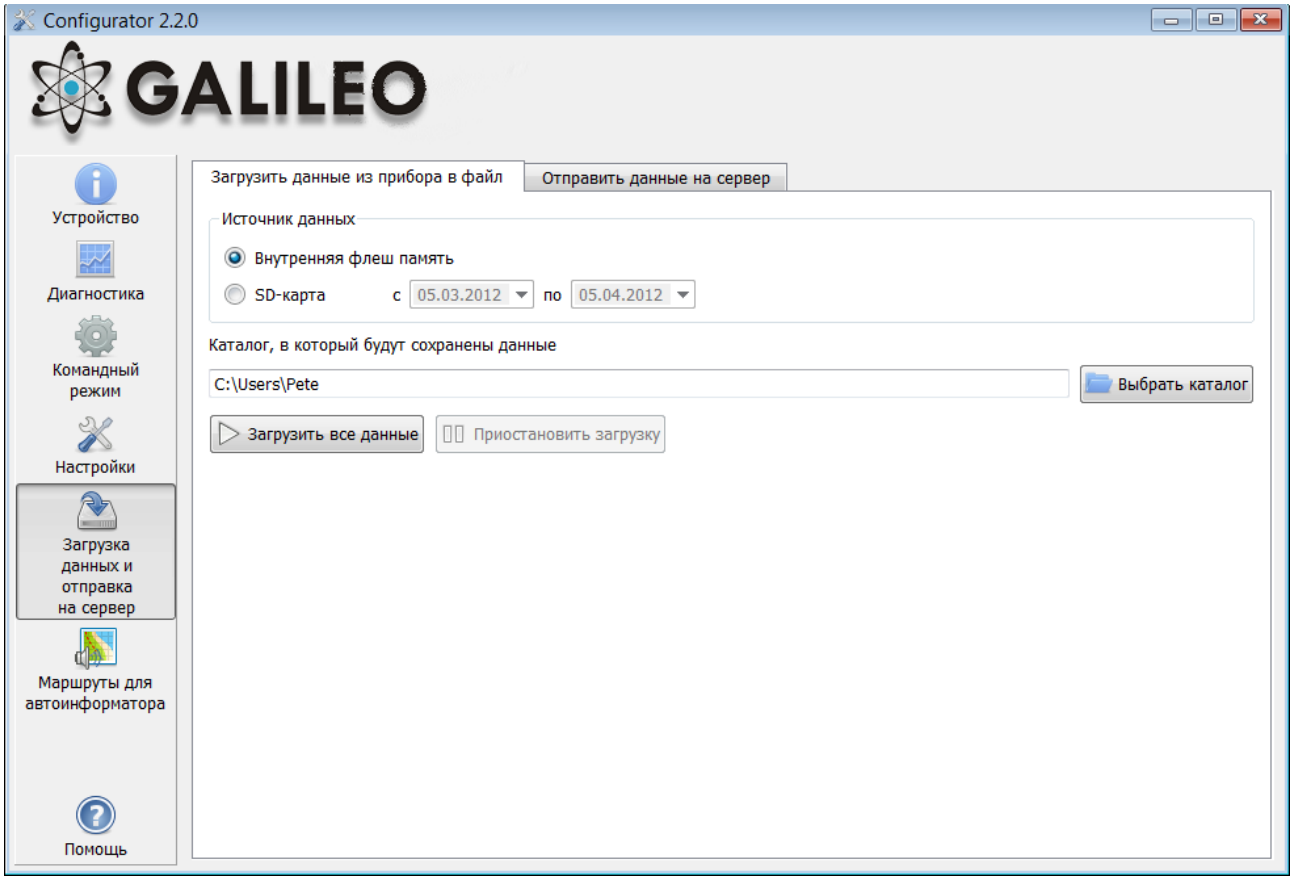
On this tab it is possible to set the list of Geofences and behavior of the Terminal inside and outside them.



5. Loading and sending data

5.1. Data loading from device to file

This option allows you to transfer the data from the internal memory or a SD-card to computer files via a USB cable. At data loading from the internal memory one InternalFlash.csv, at data loading from the SD-card there will be several files created sorted by the dates in the same way as they are stored on the card.



The Data transfer from the internal memory can be stopped and resumed; the data transfer from the SD card can be stopped only if you disconnect the USB cable.

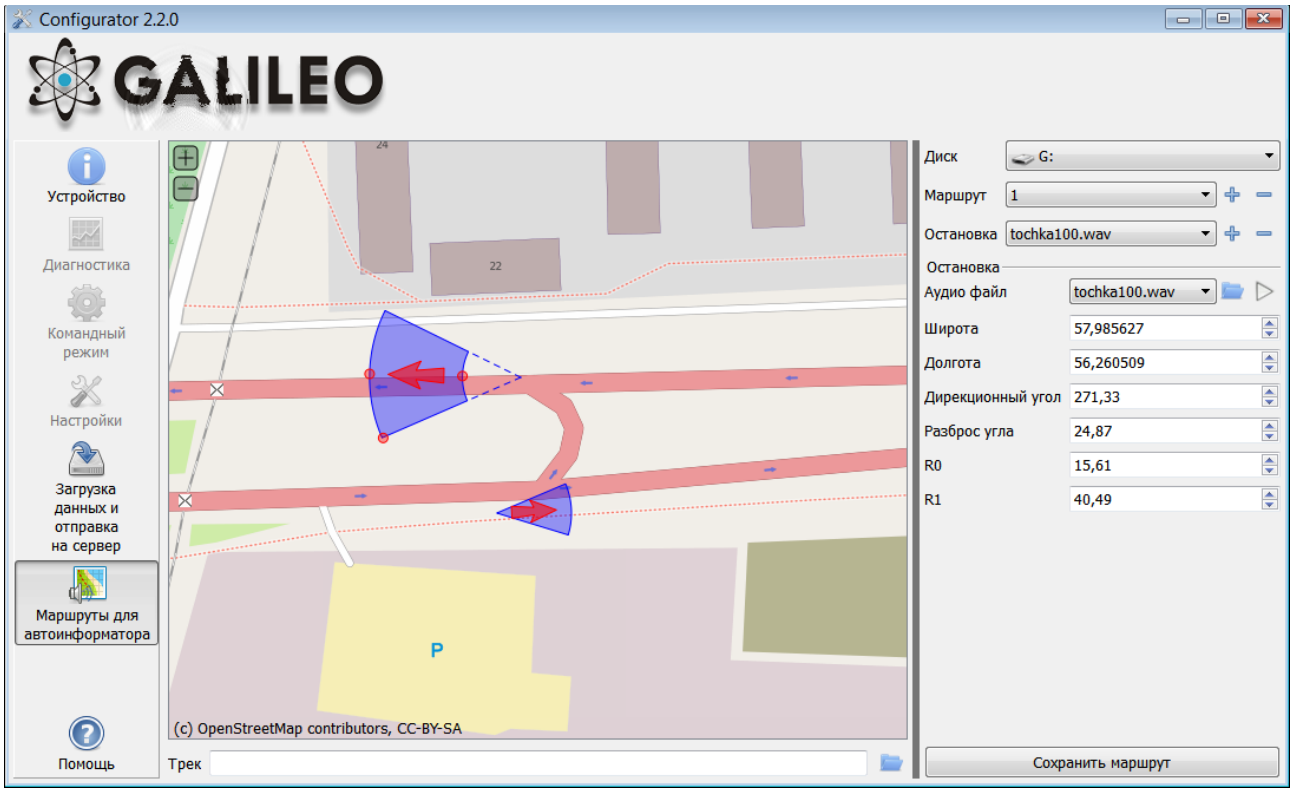
5.2. Sending data to server

This option allows sending the data previously transferred from the 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.

6. Routes for Autoinformer

This tab contains a graphical Autoinformer zones editor (see Autoinformer function).
The Internet access is required to display the map.
To edit the zones:

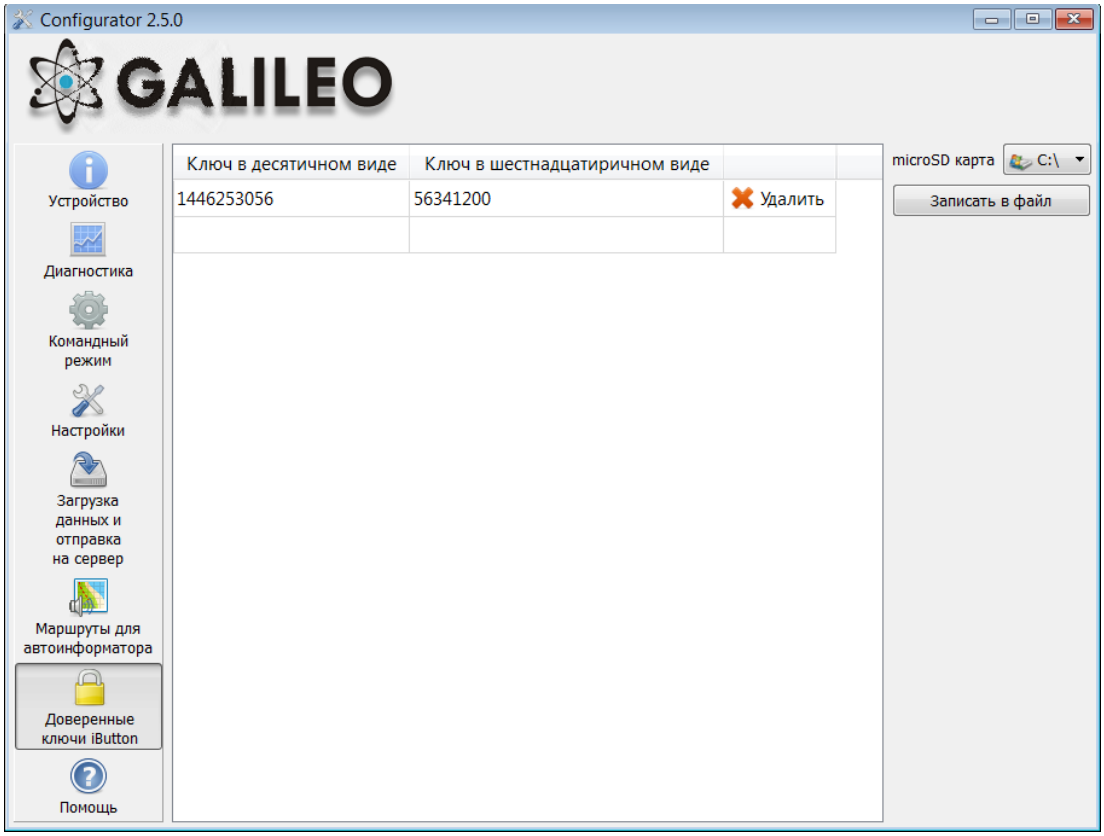
1. Insert a Micro-SD card into the computer, chose the disk in the drop-down list. The route list and the track list will be loaded automatically.
2. Choose an edited route in the drop-down list or create a new one by clicking "+" on the right side of the route list. All stops will be loaded automatically, 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.
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 Terminal and copy it to the micro-SD card.
6. Press "Save route" to save changes to the micro-SD card.



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

7. Trusted iButton keys

This tab contains the editor of a trusted iButton keys list.



The list is stored on a micro-SD card; to edit the list, perform the following:

1. Remove the micro-SD card from the Terminal.
2. Insert the micro-SD card into the computer and choose the disk in the drop-down list; the list of trusted iButton keys will be loaded automatically.
3. Keys may be entered either in decimal or in hexadecimal form; the corresponding value in another column will be converted automatically.
4. Click "Save to disk" to save the changes to the micro-SD card.

The list of keys is stored in the "keys.bin" file; it can be copied and used in micro-SD cards of other Terminals.

Bootloader

The processor program (firmware) is a set of algorithms developed by GalileoSky Ltd 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 sub-programme allowing the main program part to be updated. The firmware can be downloaded from the official site www.7gis.com . The main program can be downloaded via the USB or GPRS channel in the Terminal.

USB channel download

1) Connect the device to the external power supply;
2) Connect the USB cable; the device must be defined on the computer;
3) Launch the Configurator and open the "Command mode" tab;
4) Type upgrade 0 command after which the device will be reset in 15-20 sec;
5) After resetting the terminal will enter the bootloader mode, and the device should be defined as the system storage device (flash memory);
6) Download the right firmware version and extract firmware.bin file from the archive;
7) Copy the downloaded firmware.bin file to the flash memory;
8) After reflashing the device will be reset and enter the operating mode in 15 seconds.

GPRS channel download

1) Connect the device to the external power supply;
2) APN settings must conform to the inserted SIM card, otherwise the device flashing will not happen and it 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 terminal will turn into operation mode automatically.

Using analog inputs to enter bootloader mode

After the device power supply is off energize all discrete-analog inputs (see Contacts description) by applying the voltage of 7.0V ± 0.2V until the device enters the bootloader mode. This function is used only during an improper device flashing. An improper firmware is the firmware designed for terminals with the other functional.

LED operation during reflashing.

Depending on the GSM modem and microcontroller units activation stages the device will go through the following stages:

Yellow LED blinking, times	GSM Modem activation stage
6	GSM unit enabled successfully.
5	GPRS registered successfully.
4	Connection to update server.
3	Downloading mode.
2	Server connection and downloading mode enabled.
1	First request sent successfully.

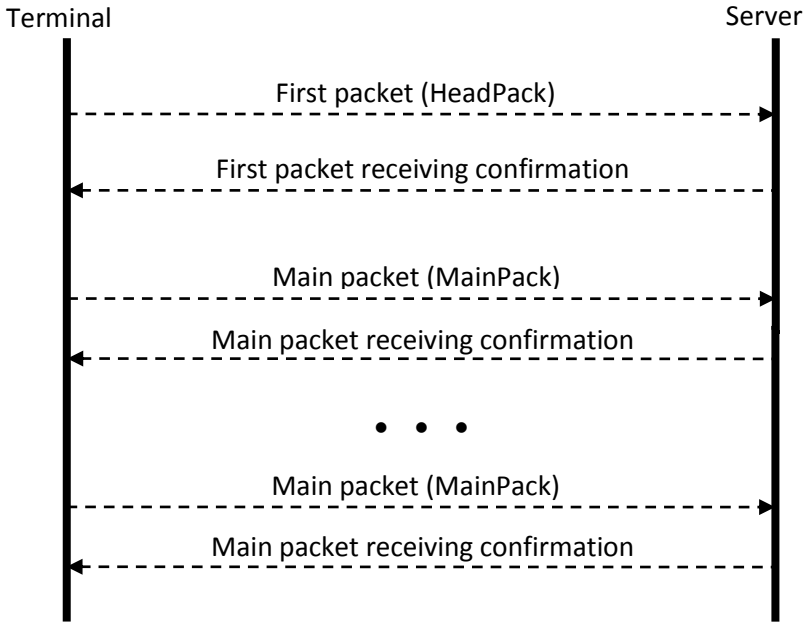
Blue LED blinking: each successfully received and recorded packet is accompanied by a blue LED light change.

Server communication protocol

GalileoSky protocol

This protocol supports bi-directional data exchange between the terminal and the server. The data are transmitted via GPRS channel with the use of TCP/IP protocol. The server must have static address and port for connecting terminals as clients.

Data transmission from the terminal to the server:



After establishing device-server connection the device sends head pack and then main packs with the data. Each pack needs conformation from the server; if confirmation is not received the terminal sends the pack once again.

Head pack structure:

Field	**Size**
Header 0x01	1 byte
Packet length	2 bytes
Tag 1	1 byte
Data corresponding tag 1	depends on the tag type
…	
Tag N	1 byte
Data corresponding tag N	depends on the tag type
Checksum	2 bytes
Field	**Size**

A high-order bit is an indicator of not transferred data in the archive, 15 least significant bits are the number of bytes in the packet.
Transmitted tags are set by HeadPack command. Packet length is calculated from the head tag to checksum beginning. Tags are in ascending order. The data and the checksum are transferred in little-endian format. The Checksum is calculated for the whole packet including the header, length field and indicator of unsent data. The Checksum is calculated by CRC-16 Modbus algorithm, you can find an example of its realization on http://www.modbus.org/docs/Modbus_over_serial_line_V1_02.pdf.

Main pack structure is the same as the structure of the head pack. Transmitted tags are set by MainPack command. Main pack may transmit several records from the archive. First record tags go first, then the second record tag, etc.

The data may be coded; XTEA3 algorithm is used for coding (http://tomstdenis.tripod.com/xtea.pdf) with block length 128 bit, key length 256 bit and 32 rounds.

In this case the header, length and the unsent data indicator stay unchanged, and archives records with the tags are coded. If the data length is not multiple to code block length, missing place is filled with zeros and then coded. The Checksum is calculated for coded data packet.

Field	Size
Header 0x02	1 byte
Received packet checksum	2 byte

Table 1. Confirmation packet structure

Packet will be transmitted again if its checksum does not correspond to the checksum in confirmation packet.

№	Tag	Designation	Parameter	
			Length, byte	
1	0x01	Hardware version	1	Unsigned integer.
2	0x02	Firmware version	1	Unsigned integer.
3	0x03	IMEI	15	ASCII line.
4	0x04	Identifier of a device	2	Unsigned integer.
5	0x10	Number of an archive record	2	Unsigned integer.
6	0x20	Date and time	4	Unsigned integer, seconds since 1970-01-01 00:00:00 GMT.
7	0x30	Coordinates in degrees, number of satellites, indication of coordinates determination correctness	9	4 lower bits: number of satellites. The next 4 bits: coordinates correctness, 0 – coordinates are correct. The next 4 bytes: signed integer, latitude, the value should be divided by 1000000, negative values correspond to southern latitude. Last 4 bytes: signed integer, longitude, the value should be divided by 1000000, negative values correspond to eastern longitude. For example, received: 07 C0 0E 32 03 B8 D7 2D 05. Coordinates correctness: 0 (coordinates are correct). Satellites number: 7 Latitude: 53.612224 Longitude: 86.890424
8	0x33	Speed in km/h and direction in degrees	4	2 Lower bytes: unsigned integer, speed, the value should be divided by 10. 2 Higher bytes: unsigned integer, direction, the value should be divided by 10. For example, received: 5C 00 48 08. Speed: 9.2 km/h. Direction: 212 degrees.
9	0x34	Height, m	2	Signed integer.
10	0x35	HDOP	1	Unsigned integer. The value should be divided by 10.
11	0x40	Status of device	2	Unsigned integer, each bit corresponds to a separate unit state, see explanations below.
12	0x41	Supply voltage, mV	2	Unsigned integer.

№	Tag	Designation	Parameter	
			Length, byte	Length, byte
13	0x42	Battery voltage, mV	2	Unsigned integer.
14	0x43	Terminal temperature, °C	1	Signed integer.
15	0x44	Acceleration	4	10 Lower bits: acceleration by X axis. Next 10 bits: acceleration by Y axis. Next 10 bits: acceleration by Z axis. 600 points correspond to free fall acceleration. Example, received: AF 21 98 15. Acceleration X: 431, Y: 520, Z: 345.
16	0x45	Status of outputs	2	Each bit, beginning with the lower one, indicates the state of a correspondent output.
17	0x46	Status of inputs	2	Each bit, beginning with the lower one, indicates triggering on a correspondent input.
18	0x50	Input voltage 0, mV Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
19	0x51	Input voltage 1, mV Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
20	0x52	Input voltage 2, mV Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
21	0x53	Input voltage 3, mV Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
24	0x70	Thermometer 0 identifier and measured temperature, °C	2	Lower byte: unsigned integer, identifier. Higher byte: signed integer, temperature. Example, received: 01 10 Identifier: 01 Temperature: 16ºC
25	0x71	Thermometer 1 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
26	0x72	Thermometer 2 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
27	0x73	Thermometer 3 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
28	0x74	Thermometer 4 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
29	0x75	Thermometer 5 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
30	0x76	Thermometer 6 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
31	0x77	Thermometer 7 identifier and measured temperature, °C	2	Analogous to temperature sensor 0.
32	0x90	First iButton key identification number	4	

№	Tag	Designation	Parameter	
			Length, byte	Length, byte
33	0xc0	Depending on setting one of the variants: 1. FMS-Standard (CAN_A0). 2. CAN-LOG, prefix E or F. Fuel used by a vehicle from the date of manufacturing, l	4	Unsigned integer, the value should be divided by 2.
34	0xc1	Depending on setting one of the variants: 1. FMS-Standard (CAN_A1). 2. OBDII. 3. CAN-LOG, prefixes G,H,I. Fuel level in tank, %. Coolant temperature, °C. Engine speed, rpm.	4	Lower byte: fuel level, the value should be multiplied by 0.4. Second byte: coolant temperature, the value should be deducted 40. Third and fourth bytes: engine speed, values should be multiplied by 0.125. Example, received: FA 72 50 25. Fuel level: 100%. Temperature 74ºC. Engine speed: 1194 rmp.
35	0xc2	Depending on setting one of the variants: 1. FMS-Standard (CAN_B0). 2. CAN-LOG, prefixes C, D. Mileage, m.	4	Unsigned integer, the value should be multiplied by 5.
36	0xc3	CAN_B1	4	
37	0xc4	CAN8BITR0 Or vehicle speed from CAN-LOG, km/h	1	If speed is transmitted from CAN-LOG – the value is an unsigned integer
38	0xc5	CAN8BITR1 or the 3rd byte of prefix S CAN-LOG	1	
39	0xc6	CAN8BITR2 or the 2rd byte of prefix S CAN-LOG	1	
40	0xc7	CAN8BITR3 or lower byte of prefix S CAN-LOG	1	
41	0xc8	CAN8BITR4 or the 4th byte of prefix P CAN-LOG	1	
42	0xc9	CAN8BITR5 or the 3rd byte of prefix P CAN-LOG	1	
43	0xca	CAN8BITR6 or the 2rd byte of prefix P CAN-LOG	1	
44	0xcb	CAN8BITR7 or the 1st byte of prefix P CAN-LOG	1	
45	0xcc	CAN8BITR8 or lower byte of prefix P CAN-LOG	1	
46	0xcd	CAN8BITR9	1	
47	0xce	CAN8BITR10	1	
48	0xcf	CAN8BITR11	1	
49	0xd0	CAN8BITR12	1	
50	0xd1	CAN8BITR13	1	

№	Tag	Designation	Parameter	
			Length, byte	Length, byte
51	0xd2	CAN8BITR14	1	
52	0xd3	The second iButton key identification number	4	
53	0xd4	Total mileage according to GPS/GLONASS units data, m.	4	Unsigned integer.
54	0xd5	State of iButton keys which identifiers are set by iButton command.	1	Each bit corresponds to one key. Example, received: 05 or 00000101 in binary system. It means that the first and the third keys are connected.
55	0xd6	Depending on settings: 1. CAN16BITR0 2. CAN-LOG, K prefix, the 1st axle load, kg 3. OBD II failure code	2	In case the load is on CAN-LOG axle, the value is an unsigned integer; values should be divided by 2.
56	0xd7	Depending on settings: 1. CAN16BITR1 2. CAN-LOG, L prefix, the 2nd axle load, kg 3. OBD II failure code	2	In case the load is on CAN-LOG axle, the value is an unsigned integer; values should be divided by 2.
57	0xd8	Depending on settings: 1. CAN16BITR2 2. CAN-LOG, M prefix, the 3rd axle load, kg 3. OBD II failure code	2	In case the load is on CAN-LOG axle, the value is an unsigned integer; values should be divided by 2.
58	0xd9	Depending on settings: 1. CAN16BITR3 2. CAN-LOG, N prefix, the 4st axle load, kg 3. OBD II failure code	2	In case the load is on CAN-LOG axle, the value is an unsigned integer; values should be divided by 2.
59	0xda	Depending on settings: 1. CAN16BITR4 2. CAN-LOG, O prefix, the 5th axle load, kg 3. OBD II failure code	2	In case the load is on CAN-LOG axle, the value is an unsigned integer; values should be divided by 2.
60	0xdb	Depending on settings: 1. CAN32BITR0 2. CAN-LOG, A or B prefix, total time of engine operation, h	4	In case the time of engine operation is on CAN-LOG, the value is an unsigned integer; values should be divided by 100.
61	0xdc	Depending on settings: 1. CAN32BITR1 2. CAN-LOG, R prefix, fuel level, l	4	In case the fuel level is on CAN-LOG, the value is an unsigned integer; values should be divided by 10.
62	0xdd	CAN32BITR2	4	
63	0xde	CAN32BITR3	4	
64	0xdf	CAN32BITR4	4	
65	0x54	Input 4 values. Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
66	0x55	Input 5 values. Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.

№	Tag	Designation	Parameter	
			Length, byte	Length, byte
67	0x56	Input 6 values. Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
68	0x57	Input 7 values. Depending on settings: 1.voltage, mV, 2.number of impulses; 3.frequency,Hz.	2	Unsigned integer.
69	0x80	Zero DS1923 sensor Identifier, measured temperature ºC and humidity %.	3	Lower byte: unsigned integer, identifier. The second byte: integer, Higher bytes: humidity, values should be multiplied by 100 and divided by 225. Example, received: 01 10 20. Identifier: 01 Temperature: 10ºC. Humidity: 7.84%
70	0x81	The first DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
71	0x82	The second DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
72	0x83	The third DS1923, sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
73	0x84	The fourth DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
74	0x85	The fifth DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
75	0x86	The sixth DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
76	0x87	The seventh DS1923 sensor Identifier, measured temperature °C and humidity %.	3	The same with zero DS1923 sensor
77	0x60	RS485. fuel level sensor with address 0	2	Unsigned integer.
78	0x61	RS485. fuel level sensor with address 1	2	Unsigned integer.
79	0x62	RS485. fuel level sensor with address 2	2	Unsigned integer.
128	0xA0	CAN8BITR16	1	Accessible only by a dynamic archive structure
Tags CAN8BITR17 - CAN8BITR30 (0xA1-0xAE) similar to CAN8BITR16 with numbers 129-142				
143	0xAF	CAN8BITR31	1	Accessible only in the dynamic archive structure

№	Tag	Designation	Parameter	
			Length, byte	
144	0xB0	CAN16BITR6	2	Accessible only in the dynamic archive structure
Tags CAN16BITR7 – CAN16BITR14 (0xB1-0xB8) similar to CAN16BITR6 with numbers 145-152				
153	0xB9	CAN16BITR15	2	Accessible only in the dynamic archive structure
160	0xF0	CAN32BITR6	4	Accessible only in the dynamic archive structure
Tags CAN32BITR7 – CAN32BITR14 (0xF1-0xF8) similar to CAN32BITR6 with numbers 160-169				
169	0xF9	CAN32BITR15	4	Accessible only in the dynamic archive structure

Table 2. GalileoSky protocol tag

Bit number	Field explanation
0	0 – vibration level corresponds to parking; 1 – to driving (set by AccSens command).
1	0 – incline angle does not exceed the allowable one; 1 – incline level exceeds the allowable one.
2	0 – None of the trusted iButton keys are connected, 1- one of the recorded to the SD-card iButton keys is connected.
3	0 – there is a SIM card, 1 – GSM modem can't determine the SIM-card.
4	0 – terminal is outside the Geofence, 1 - terminal is inside the Geofence.
5	0 – voltage of internal source is normal; 1 – lower than 3.7 V.
6	0 – GPS aerial is connected; 1 – disconnected.
7	0 – voltage of internal terminal bus supply is normal, 1 – declined from normal.
8	0 – external supply voltage is normal, 1 - declined from normal (set by powincfg command).
9	0 – vehicle is stopped; 1 – vehicle is started (set by mhours command).
10	0 – vibration level corresponds to the normal movement, 1 – vibration level corresponds to a strike.
11	0 – GPS operates; 1 – GLONASS unit operates.
12	Signal quality, range: [0-3]. The less value, the worse communication.
13	
14	0 – signaling mode is off; 1 – on.
15	0 – no alarm; 1 – alarm activated.

Table 3. Device status field explanation

Example1.
The terminal should be configured so that the head pack (HeadPack) contains information about the device version (HardVersion), firmware version (SoftVersion), unique 15 digit GSM 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 device user ID (ID device), packet number (NumberOfPacket), date and time of packet record (TimeDate), coordinates would be sent.

Correspondent tag mask: 00000000000000000000000001111000

To apply the settings we should use the command: MainPack 1111000.
In this case we have been omitted the zeros at once.

The server can send commands to the device. After command receiving and its successful execution, the terminal sends a packet with text reply.

Structure of a packet with a command:

Field	**Size**
Header 0x01	1 byte
Payload length	2 bytes
Tag 0x03	1 byte
IMEI	15 bytes
Tag 0x04	1 byte
Device identifier	2 bytes
Tag 0xE0	1 byte
Command number, random number selected by the server	4 bytes
Tag 0xE1	1 byte
Command line length	1 byte
Command text in ASCII	
Checksum. Calculated for the whole packet beginning with the header.	2 bytes

Respond's structure is analogous to the command's packet, but reply text is sent instead of command text.

Structure of a packet with data of vibration analysis:

Field	**Size**
Header 0x05	1 byte
Payload length	2 bytes
Unique identifier of the first characteristic feature	4 bytes
The relative intensity1	2 bytes
Unique identifier of the second characteristic feature	4 bytes
The relative intensity2	2 bytes
…	
Unique identifier of the N characteristic feature	4 bytes
The relative intensityN	2 bytes
Checksum. Calculated for the whole packet beginning with the header.	2 bytes

FAQ

Table. The most common user questions

Question	Answer
When the vehicle is not moving the device records too many packets to the memory	When device, input, output status is changed the device records an extra packet. E.g. When input discrete state changes a packet is recorded when entering logical one zone from logical zero zone. And vice versa, when the state changes from logical one to logical zero a packet is recorded.

Additional information

1. Certifying

The device is certified to comply with GOST R.

2. Warranty

GalileoSky Ltd hereby guarantees the realization of consumers' rights provided by the local laws throughout Russia and the CIS.
GalileoSky Ltd guarantees the operability of the device subject to compliance with the instructions set out in the above manual.

2.1. Warranty conditions

The warranty period is – 12 months since the day of purchase.
Note: a defective device (with cracks and fissures, dents and impact marks 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;
- Independent opening of the device in case of warranty seals and labels.
- Repairing controller 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 terminal;
- Operation of the terminal at the in vehicle network voltage deviating from the range mentioned in technical specifications.

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.