Inertial Labs
Attitude is Everything

Single & Dual antenna GPS-Aided Inertial Navigation Systems INS

Interface Control Document

Models	INS-B	INS-P	INS-D	INS-DL	INS-B-OEM	INS-D-OEM	INS-DL-OEM
Revision	2.38						

CHANGE STATUS LOG			
Document: Inertial Labs™ GPS-Aided INS Interface Control Document			
Revision	**Date**	**Affected Paragraphs**	**Remarks**
1.0	Jul. 14, 2015	All	Released version.
1.1	Jul. 17, 2015	6.2	Minor changes.
1.2	Sep.03, 2015	6, 6.8 6.2 6.2.5 6.3.2	**1**. Implemented auto start option with choice of desirable variant of output data format after device power on. Since INS firmware version 1.0.2.0. **2**. Corrected mistake in tables: ms_pos is replaced by ms_gps **3**. Updated description of the «INS NMEA Output» data format (timestamp is added). **4**. Renamed command DataOnRequest to SetOnRequestMode to exclude misunderstanding of this command action.
1.3	Nov.02, 2015	5 5 6.2.6 6.3.1 6.10 6.4	**1**. Added diagrams Fig.5.4, Fig.5.5 of electric connection of INS with two and three COM ports. **2**. Changed connector pinout to include two more COM ports, Table 5.1. **3**. Added new “INS Sensors NMEA Output” data format. **4**. Updated “Table 6.16. INS maximum data rate at different output data formats”. **5**. Added section “6.10. Post processing of the INS data”. **6**. Added sections “6.4.1. GNSS receiver parameters” and “6.4.2. Control of GNSS receiver model”.
1.4	Dec.01, 2015	5	Added color of wires in cable with mating connector in Table 5.1.
1.5	Feb.02, 2016	6.2.1, 6.2.2, 6.2.4, 6.2.6 6.4.1 5	**1**. Corrected values of maximum data rate for different data formats. **2**. Added description of new GNSS_com2_bps parameter. **3**. Added description of PPS in section “5.2. PPS description”.
1.6	Feb.18, 2016	 6.2.1 6.2.2 6	For INS with firmware version since 2.1.2.0: **1**. Corrected INS message payload at the “INS OPVT” data format in the Table 6.4. **2**. Implemented new “INS QPVT” output data format. **3**. Byte #3 in the INS output data is used for identification of command which the INS answers on (see Table 6.2).
1.7	Apr.21, 2016	1 5 5 6.2.3 6.3.1 6.3.2 6	**1**. Added section “1.3. True and magnetic heading”. **2**. Added description of 19 pin connector of the INS with RS-422 interface. **3**. Added section “5.2. Connection of the Inertial Labs™ INS with RS-422 interface to the host computer for tests”. **4**. Added magnetic declination field to “INS Full Output Data” format instead of reserved field (see Table 6.8) – since firmware version 2.2.0.2. **5**. Changed byte structure of the block of initial alignment data – added Table 6.18. **6**. INS_SensorsData command is not supported in the “On Request” operating mode since INS firmware version 2.1.1.0. **7**. Added sections “6.11. Change of the main COM port baud

		Appendix D	rate" and "6.12. Limitation of the INS maximum output data rate". **8**. Added Appendix D. Forms of the Inertial Labs™ INS orientation presentation.
1.8	Jul.29, 2016	 5.3 5.4 6.2.1, 6.2.2, 6.2.3, 6.2.5 6.2.1 6.4.1 6.10.1 6.11 6	For INS with firmware version since 2.2.1.0: **1**. Changed section "5.3. PPS description". **2**. Added section "5.4. GPIO description". **3**. Changed GNSS information in output data formats INS OPVT; INS QPVT; INS Full Output Data; INS Minimal Data. See notes in these sections. **4**. Changed GNSS information in Table 6.5, Table 6.6. **5**. Added new parameters to section "6.4.1. GNSS receiver parameters" including PPS control and input marks control. **6**. Added section "6.10.1. Raw GNSS receiver data". **7**. Added section "6.11. Synchronization of the INS data with LiDAR and other devices". **8**. COM3 has two functions: to receive data for GNSS differential corrections or to output $GPRMC messages
2.0	Aug.09, 2016	1.1 1.4, 1.5 4.3 5 5 6.2, 6.3, Appendix C 6.5	**1**. Presented new line of Inertial Labs INS: INS-B, INS-P, INS-D. **2**. Added sections "1.4. Ground track angle vs heading" and "1.5. Using GNSS heading in INS-D". **3**. Added section "4.3 Installation of two GNSS antennas for INS-D operation". **4**. Shown connectors position on back side of INS-B, INS-P, INS-D units (Fig.5.1, Fig.5.2). **5**. Added electrical specifications for INS-B, INS-P, INS-D units (Table 5.3). **6**. Added two output data formats – OPVT2A, OPVT2AHR and appropriate commands INS_OPVT2Adata, INS_OPVT2AHRdata. **7**. Heave calculation is supported in INS-D but not in INS-B and INS-P units.
2.1	Sep.05, 2016	6.2 6.9	**1**. Added ±450°/s gyro range for KG values (see notes to Tables 6.4, 6.7, 6.8). **2**. Added indication of GNSS receiver failure in the Unit Status Word (since INS firmware version 2.5.0.2).
2.2	Sep.16, 2016	6.2, 6.3, Appendix C	Added output data format – OPVT2AW, and appropriate command
2.3	Oct.02, 2016	6.2 6.3.6	**1**. Corrected KA scale factor for ±8g accelerometer range and scale factor for supply voltage (see Tables 6.4, 6.7, 6.8, 6.9, 6.10 and notes to them). **2**. Added type of pressure sensor in INS devices information Table 6.25.
2.4	Dec.06, 2016	6.2.12 6.2.13 6.4.1, 6.7.5, Appendix A, Appendix C 6.9	**1**. Added GNSS receiver NMEA data set. **2**. Changed GPRMC format description. **3**. Added new parameters COM2_data, NMEA_set. **4**. Added description of VG3D calibration, and appropriate command StartVG3DClb. **5**. Changed description of the bits #7, 15 of USW

2.5	Jan.06, 2017	6.7.12, Appendix A, Appendix C 6.9	**1**. Added description of on-the- fly VG3D calibration, and appropriate commands StartVG3Dclb_flight, StopVG3Dclb_flight. **2**. Bits #7, 15 of USW are used for indication of stages of on-the- fly VG3D calibration.
2.6	Sep.25, 2017	1.1 5 5 6 6.2.11 6.4.1 6.6 6.2.1, 6.2.2, 6.2.6 6.2.3, 6.2.5 6.2.6 6.5 6.9 6.10 All All	**1**. Added Fig.1.4. Position of the accelerometer mass-center in Inertial Labs™ INS unit. **2**. Added one more function to COM3 port – to output of the raw GNSS receiver data (since INS firmware version 2.8.0.6). **3**. Added description of the combined RS-232 / RS-422 interface (24 pin connector). **4**. Deleted “INS Full Output Data” format. **5**. Added description of GPHDT log. **6**. Updated the description of GNSS control parameters. **7**. Added description of external sensors data input. **8**. Latency field is split into two fields: Latency ms_pos and Latency ms_vel in the following output formats: INS Sensors, INS OPVT, INS QPVT, INS Minimal **9**. Changed fields Pitch GNSS, Heading STD and Pitch STD to Latency ms_head, Latency ms_pos, Latency ms_vel respectively in the following output formats: INS OPVT2A, OPVT2AHR. **10**. Added INS OPVTAD data format and its description. **11**. Changed Baro_enabled to Baro_altimeter parameter and its description. **12**. Added information about GNSS receiver cold start to sequence of INS operations after power on at INS automatic start. **13**. Changed USW bit #1 indication to “IMU data correctness”. **14**. Removed the information about Heave. **15**. Removed TSS1 data format
2.7	Jan.23, 2018	1.1 4.3 5 5.4, 6.6.1 5.7 6.2 6 6.2, 6.3, Appendix C	**1**. Implemented calculation of INS position and velocity for any measuring point set by its position relative to INS unit (since firmware version 3.2.2.6). **2**. Added feature for INS-D secondary antenna installation in arbitrary (but known) position relative to the INS-D unit and the primary antenna (since firmware version 2.9.1.7). **3**. Corrected picture of the connector pinout of the Inertial Labs™ INS with combined RS-232 / RS-422 interface (Fig.5.5). **4**. Added description of COM4 port functions (to receive external data from a device with RS232 interface or to output GPRMC messages). **5**. Added section “5.7. Connection of the Inertial Labs™ INS with Ethernet interface to the host computer”. **6**. Added **g** value to notes to Tables 6.4, 6.7 – 6.11 with description of output data formats. **7**. Since firmware version 3.2.0.0 only INS data rates that are factors of 200 Hz are available. **8**. Added two output data formats “INS OPVT & Raw IMU Data”, “SPAN rawimu” and appropriate commands (since

		6.6 6.6.3, Appendix C 6.15	firmware version 3.2.1.8). **9** Added sections “6.6.2. Odometer data input from encoder (wheel speed sensor)” and “6.6.3. Calibration of encoder-based odometer (wheel speed sensor)”. **10**. Added commands Start_Odom_Clb and Stop_Odom_Clb to start and stop odometer calibration. **11**. Added simple “INS solution status” (good / poor) to “Angles position type” value in appropriate output data formats (since firmware version 3.2.2.7).
2.8	Mar.14, 2018	6 6.2.10	**1**. Corrected Table 6.2. Byte structure for all commands and messages to / from the INS. **2**. Changed “SPAN rawimu” data format (scale factors for raw accelerometers and gyros data, IMU status).
2.9	May 10, 2018	5.5	Added the information about availability of the 5V TTL level of the PPS.
2.10	Jun.07, 2018	5.8 6.2.17 6.2.1 6.2.1 6.2, 6.3, Appendix C 6.2.6, 6.6.4 6.5 6.2.12, 6.15 6.3.10 6.9 6.16	**1**. Added section “5.8. Connection of the Inertial Labs™ INS with CAN interface to the host computer”. **2**. Added section “6.2.17. The CAN messages (transmitted through CAN port)”. **3**. Added Table 6.5. Values of KG, KA factors for gyro and accelerometer scaled data. **4**. Added Table 6.8. New_GPS indicator of new update of GNSS data **5**. Added two output data formats “INS OPVT GNSSext”, “User Defined Data” and appropriate commands (since firmware version 3.2.4.3). **6**. Changed type of “Air Speed” field in aiding data from word to sword (signed short), see Table 6.13, Table 6.80. **7**. Implemented two variants of altitude output – above mean sea level or WGS84 ellipsoid. **8**. Implemented extended “INS solution status” as data type in “User Defined Data” output format. **9**. Description of the block of the initial alignment data is transferred from section 6.3.1 to new section 6.3.10. Description of the extended block of the initial alignment data is added. **10**. Corrected description of INS auto start. **11**. Added section “6.16. Time stamps in INS messages”.
2.11	Aug.01, 2018	1.1 6.2.3, 6.2.5 6.2.17 6.2.12, 6.6.4 6.2.12	**1**. Corrected position of the accelerometer mass-center in Inertial Labs™ INS unit (see Fig.1.4). **2**. V_latency parameter is 2 byte word in payload of “INS OPVT2A”, “INS OPVT2AHR” data (see Table 6.10, Table 6.12). **New features in INS with firmware version since 3.2.5.8:** **3**. Added description of “Angular rates”, “Accelerations”, “Magnetic field” CAN messages. **4**. Added “Heading” measurement type to aiding data (see Table 6.84) and appropriate data type to “User Defined Data” output format (see Table 6.29, Table 6.30). **5**. Added “Dilution of precision” data type to “User Defined Data” output format (see Table 6.29, Table 6.30).

		6.11.1 6.2.18	**6.** Described minimal variant of raw GNSS data logs for post processing (see Table 6.94). **7**. Added description of "IMU TGA" output data transmitted through COM4 port.
2.12	Aug.30, 2018	6.2.12	For INS with firmware version since 3.2.6.0: Added "UTC" and "GNSS Position and Speed accuracy" data types to "User Defined Data" output format (see Table 6.29, Table 6.30).
2.13	Oct.03, 2018	1.1, 4, 5.2 4.1 4.6 5, Appendix E 6.2.11 6.2.12 6.2.12 6.2.12 6.3.11, Appendix C 5.6, 6.4.1 6.11.1, 6.12.3, 5.8, Appendix F	**1.** Added Inertial Labs™ INS-B/P-OEM and INS-D/DL-OEM units (see Fig.1.3, Fig.1.4), their position of accelerometer mass-center (see Fig.1.7), outline drawings (see Fig.4.3), electrical interface. **2.** Section "4.1. Mechanically mounting the Inertial Labs™ INS" is divided into two subsections ": 4.1.1. Mounting of the housed INS unit" and "4.1.2. Mounting of the INS-OEM unit". **3.** Added section "4.6. The installation peculiarity for the Inertial Labs™ INS-P unit". **4.** Description for electrical interface of the housed INS units with 12 and 19 pin connector is moved from section 5 to Appendix E. **5.** Changed measurement units of accelerations AccX, AccY, AccZ from m/s^2 to 'g' in "INS OPVT GNSSext" data format. **6.** Added "GDOP, PDOP" and "GNSS Track over ground" data types to "User Defined Data" output format (see Table 6.29, Table 6.30). **7.** Renamed "UTC" data types in "User Defined Data" output format (see Table 6.30). **8.** Added explanation of "UTC Decimal Seconds" in "User Defined Data" output format (see Note 4 to Table 6.30). **9.** Added DevSelfTest command to get result of the device self-test. **10.** "GPIO" line of electrical interface is renamed to "MARK IN". **11.** Description of Inertial Labs™ CAN2.0-to-RS232 adapter is moved from section 5.8 to Appendix F.
2.14	Oct.26, 2018	6.2.6, 6.2.12, 6.6.4 6.2.17 1.2, 4.3, 5.1	**1.** Added Doppler Velocity Log "DVL data" measurement type to aiding data (see Table 6.14, Table 6.88) and appropriate data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **2.** Added description of "INS solution accuracy", "GNSS info" CAN messages. **3.** Terms "master" and "rover" antennas are replaced with "primary" and "secondary" antennas.
2.15	Jan.04, 2019	6.1 6.2 6.3.8 6.3.10, 6.3.11	**1.** Added description of the Firmware update operating mode. **2.** Corrected notes to all output data formats with orientation, position, velocity data, and sensors output data. **3.** Corrected notes to Table 6.60. Payload of the INS answer on the User_Def_Data_config command with list of data types. **4.** Added description of CAN_message_set_config and Get_CAN_message_set_struct commands.

		Appendix E	**5.** Added Appendix E. Instant and averaged output data of the Inertial Labs™ INS.
2.16	Jan.22, 2019	6.2.6, 6.2.12, 6.6.4 6.2.12 5.8, Appendix G	**1.** Added "Sensors bias external" and "Pitch and Roll external" measurement types to aiding data (see Table 6.14, Table 6.91) and appropriate data types to "User Defined Data" output format (see Table 6.29, Table 6.30). **2.** Added "Sensors bias" data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **3.** Removed Appendix G. Using CAN2.0-to-RS232 adapter for evaluation.
2.17	Feb.04, 2019	6.2, Appendix C 5.9	**1**. Added "Cobham UAV 200 Satcom" output data format and appropriate command Cobham_UAV200_Satcom (since firmware version 3.2.7.9). **2.** Added section "5.9. Connection of OS3D-FG Stand Alone Magnetic Compass to INS".
2.18	Feb.28, 2019	6.2 6.2.1 6.2.18 6.2.20 6.3.13 6.2.6, 6.2.12, 6.6.4 6.2.12	**1.** Made more detailed description of Latency ms_head, Latency ms_pos and Latency ms_vel fields in output data formats. **2.** Removed obsolete ranges of gyros and accelerometers in Table 6.5. Values of KG, KA factors for gyro and accelerometer scaled data. **3.** Added support of CAN 2.0B messages in addition to CAN 2.0A. **4.** Added description of "HEHDT" output data transmitted through COM4 port`. **5.** Added bits #7 to 9 to INS answer on DevSelfTest command (see Table 6.72). **6.** Changed name of external aiding data type "DVL data" to "External velocity" (see Table 6.14, Table 6.29, Table 6.30, Table 6.93). Added note about two variants of reference frames of external velocity data. **7**. Added "GNSS receiver clock info" data type to "User Defined Data" output format (see Table 6.29, Table 6.30, Table 6.31).
2.19	Apr.12, 2019	5.2.2	**1.** Added description of 14-pin electrical interface for INS-OEM units.
2.20	May.13, 2019	6.2.6, 6.2.12, 6.6.4	**1.** Changed name of external aiding data type "External velocity" to "DVL data" (see Table 6.14, Table 6.29, Table 6.30, Table 6.94).
2.21	May.24, 2019	6.2.12 6.6 6.6.4	**1.** Added "KF position covariance HR" data to "INS service data" group of "User Defined Data" output format. **2.** Section "6.6. Using external sensors data" was renamed to "6.6.INS operation in GNSS-denied environment". **3.** Added section "6.6.4. Velocity constraints method".
2.22	Jun.20, 2019	5.4 6.3.14 6.6.4	**1.** Added important note about connection of encoder-based odometer to INS. **2.** Added section "6.3.14. Configuration of INS parameters". **3.** Section "6.6.4. Velocity constraints method" was renamed to "6.6.4. Velocity constraints and ZUPT option".
2.23	Jul.02, 2019	5.1	**1.** Corrected wire color codes in Table 5.1.

2.24	Jul.03, 2019	6.2.1 6.4.3	**1.** Added new position types in GNSS_info1 data (see Table 6.6). **2.** Added new section "6.4.3. Quicker start of the NovAtel GNSS receiver at known position".
2.25	Jul.24, 2019	6.4.3 6.11.1 All	**1.** Section "6.4.3. Quicker start of the NovAtel GNSS receiver at known position" was renamed to "6.4.3. Accelerated start of the NovAtel GNSS receiver with TerraStar service at known position" and was removed PPPSEED_AUTO command (see section 6.4.3). **2.** Replaced BDSEPHEMA and GALEPHEMA logs with BD2EPHEM and GALEPHEMERIS logs respectively (see Table 6.92). **3.** Corrected information related to INS initial alignment process throughout the whole document.
2.26	Jul.30, 2019	1.1, 6.2	**1.** Added note about the possibility to use internal datalogger.
2.27	Aug.09, 2019	6.3.14, 6.4.1 6.6.3 6.6.3	**1.** Added TerraStarType parameter (code 0x1B) to Table 6.79 which specifies type of TerraStar correction for the GNSS receiver. **2.** Added note 4 to Table 6.81 that reference point for odometer, air speed or DVL data can be set by Odom_offset parameter. **3.** Noted that currently INS can use all types of aiding data listed in Table 6.83 except "Doppler shift from locator".
2.28	Sep.24, 2019	4.7 5.9 6.2.6 6.2.16 6.2.21, 6.3.1, Appendix C 6.3.14 6.4.1 6.6.3 6.6.4, 6.3.14 6.10 6.15 6.17	**1.** Added section "4.7. Installation of OS3D-FG Stand Alone Magnetic Compass". **2.** Removed notes in section "5.9. Connection of OS3D-FG Stand Alone Magnetic Compass to INS", appropriate information is transferred to sections 4.7, 6.8. **3.** Corrected message payload at the "INS OPVTAD" data format in Table 6.13. **4.** Corrected GPRMC message structure (Table 6.36). **5.** Added "NAV440" output data format and appropriate command INS_NAV440. **6.** Added ZUPT_threshold parameter (code 0x2A) to Table 6.81 which is velocity threshold for stop detection at operation with encoder. **7.** Corrected description to the GNSS correction "AUTO" type. **8.** Corrected description to total packet length calculation in Table 6.84, Table 6.85, Table 6.86. **9.** "Velocity constraints" method is updated to "Tunnel Guide" feature. **10.** Changed USW bit #1 indication to "Software status". **11.** Added values to "INS solution status": applied ZUPT, INS invalid solution. **12.** Added section "6.17. Using external GNSS receiver"
2.29	Oct.11, 2019	6.2.12 6.11.1 6.3.14, 6.6.3	**1.** Added "GNSS diff age" data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **2.** Added new asynchronous logs to raw GNSS data (see Table 6.93). **3.** Added configuration parameters for CAN BUS based odometer (codes 0x46 to 0x49) to Table 6.81, and description

			to section 6.6.3.
2.30	Nov.06, 2019	6.2.12 6.3.14 6.6.3 6.6.5	**1.** Added "GNSS ECEF Velocity STD" data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **2.** Added "CAN based odometer" variant to configuration parameter "Odometer_type" (code 0x29) in Table 6.81. **3.** Added formulas for using Factor and Offset parameters at conversion of CAN data for CAN BUS based odometer. **4.** Added configuration parameters for ZUPT option (codes from 0x60 to 0x63) to Table 6.81, and description to section 6.6.5.
2.31	Dec.02, 2019	6.2.12 6.3.14, 6.6.3	**1.** Added "Accelerometer data in PV measuring point" data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **2.** Changed configuration parameters for signal 3 of CAN BUS based odometer (code 0x49) in Table 6.81, and changed appropriate description in section 6.6.3.
2.32	Dec.20, 2019	5.2.2	**1.** Updated Table 5.4 and Table 5.5.
2.33	Jan.09, 2020	5.7 6.2.12 6.3.14	**1.** Added note about the inability to use simultaneously the internal dataloger and Ethernet data/command transfer. **2.** Added "PPPSEED status" data type to "User Defined Data" output format (see Table 6.29, Table 6.30). **3.** Added Save_last_pos parameter (code 0x0D) to Table 6.81 which allows saving the last position to INS flash memory.
2.34	Jan.23, 2020	6.3.14 6.8.13 6.8.14	**1.** Increased length of PPS_pulse_width parameter from two to four bytes (see Table 6.80) to increase maximum PPS pulse width that user can set. **2.** Added description of automatic 3D calibration of magnetometers. **3.** Added section "6.8.14. Choice of set of INS calibration parameters".
2.35	Feb.03, 2020	5.9 6.3.14	**1.** Added section "5.9. Connection of the Inertial Labs™ INS with second CAN interface to the host computer". **2.** Added configuration parameters for CAN2 port (codes 0x4A to 0x4B) to Table 6.80.
2.36	Feb.10, 2020	6.8.13	**1.** Added 2D variant of automatic calibration of magnetometers.
2.37	Mar.05, 2020	6.2.12 6.6.6 6.15	**1.** Added "KF velocity covariance" data to "INS service data" group of "User Defined Data" output format. **2.** Added commands to switch off/on the GNSS receiver data input to the INS algorithm (for evaluation purposes). **3.** Corrected conditions for setting of "INS solution status" values (see Table 6.97).
2.37	Mar.27, 2020	6.3.14, 6.6.5 6.4.1, 6.3.14	**1.** Added new parameters Gyro_3_zupt, Freeze_PV (code 0x64) to configure INS ZUPT mode. **2.** Added Enable_PDPFILTER parameter (code 0x1C) to enable PDP filter in the NovAtel receiver to smooth a jumpy GNSS position and to bridge outages in satellite coverage.

TABLE OF CONTENTS

List of Tables

1. Introduction

1.1. Description of the System

The **Inertial Labs^{TM} GPS-Aided Inertial Navigation System, INS** is high-performance GPS-aided strapdown system that calculates absolute orientation (heading, pitch and roll) and position (latitude, longitude, altitude) for any device on which it is mounted. Orientation and position are determined with high accuracy for both motionless and dynamic applications.

The Inertial Labs^{TM} INS utilizes 3-axes each of precision accelerometers, magnetometers and gyroscopes to provide accurate heading, pitch and roll of the device under measure. Integration of gyroscopes' output provides high frequency, real-time measurement of the device rotation about all three rotational axes. Accelerometers and Fluxgate magnetometer measure absolute Pitch, Roll and magnetic Azimuth at INS initial alignment as well as providing ongoing corrections to gyroscopes during operation.

The Inertial Labs^{TM} INS has an onboard high-grade Global Navigation Satellite System (GNSS) receiver which provide high accurate position using the next GNSS systems:

- GPS L1, L2, L2C;
- GLONASS L1, L2;
- Galileo E1;
- BeiDou B1;
- Compass3;
- SBAS;
- QZSS.

Inertial Labs^{TM} provides the next models of INS products (see Fig. 1.1 to Fig. 1.4):

- INS-B and INS-B-OEM (Basic model) – use high grade IMU and high grade single antenna GNSS receiver;
- INS-P (Professional model) – use high-grade Fluxgate magnetometers, high grade IMU and high grade single antenna GNSS receiver;
- INS-D, INS-DL and INS-D-OEM - (Dual antenna model) – use high grade IMU, dual-antenna GNSS receiver and measures static and dynamic Heading, independent on magnetic field disturbance.

Notes:

1. In this document all information related to the INS-D is correct for the INS-DL, unless otherwise specified;

2. In this document all information related to the INS-B, INS-P, INS-D are correct for the INS-B-OEM, INS-D-OEM respectively, unless otherwise specified;

3. Any model of INS can be factory configured to contain an internal datalogger to record data, but only upon request by a customer (see INS GUI User Manual, rev.2.22 and newer, Appendix I. Using the Internal Datalogger).

Fig. 1.1. Inertial Labs™ INS-B and INS-P

Fig. 1.2. Inertial Labs™ INS-D and INS-DL

Fig. 1.3. Inertial Labs™ INS-B-OEM

Fig. 1.4. Inertial Labs™ INS-D-OEM and INS-DL-OEM

Fig. 1.5 shows the INS own coordinate system $Ox_oy_oz_o$. This coordinate system is body-fixed and defined as the calibrated sensors coordinate system. Non-orthogonality between axes of the body-fixed coordinate system $Ox_oy_oz_o$ is an order of 0.01°.

Measured angles are the standard Euler angles of rotation from the Earth-level frame (East-North-Up) to the body frame, heading first, then pitch, and then roll.

Orientation angles, measured by the Inertial Labs™ INS, are not limited and are within common ranges:

- Heading 0…360°;
- Pitch ±90°;
- Roll ±180°.

Also the Inertial Labs™ INS provides orientation calculation in quaternion form. See “Appendix D. Forms of the Inertial Labs™ INS orientation presentation”.

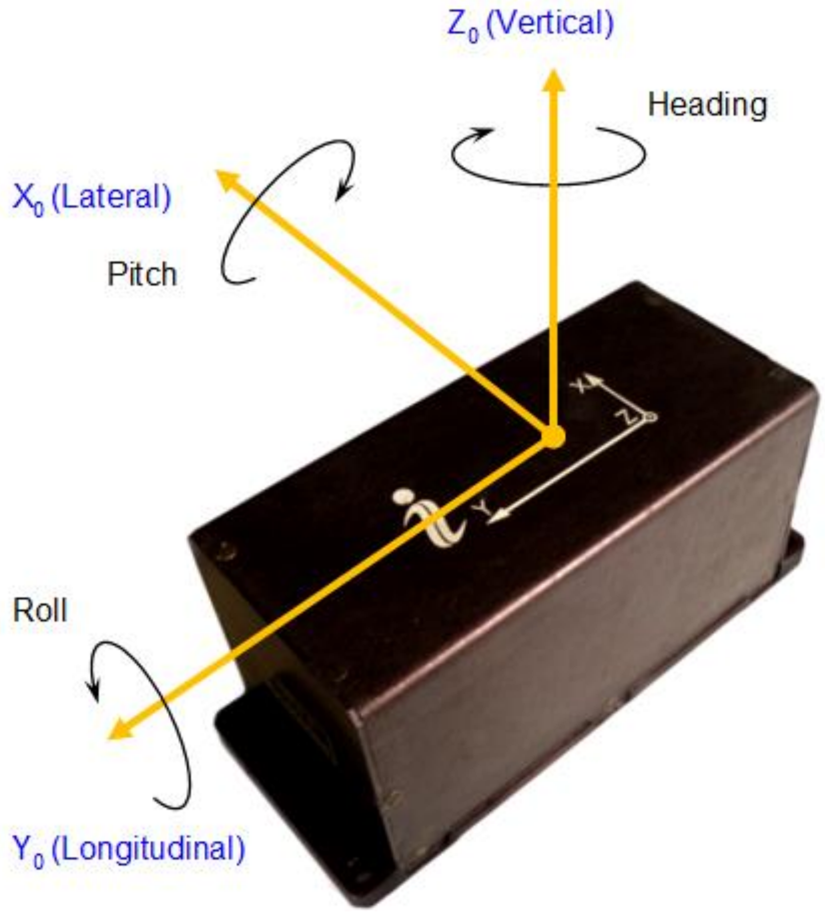
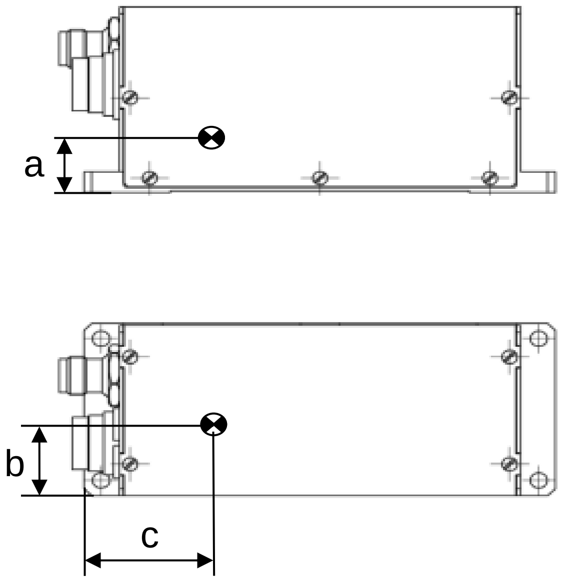


Fig. 1.5. Coordinate system of the Inertial Labs™ INS

At its operation the Inertial Labs™ INS calculates position and linear velocity using its gyros and accelerometers data with correction from the onboard GNSS receiver. For altitude calculation the INS also uses correction from the onboard pressure sensor.

The Inertial Labs™ INS calculates position and velocity for the accelerometer mass-center of the INS unit (see Fig. 1.6 and Fig. 1.7).



Dimensions, mm	a	b	c
Variant 1	10	20	29
Variant 2	15	27	58

Note:
Variant 1: Includes INS-P, INS-B units with serial numbers up to F1710189 and INS-D units with serial numbers up to F1691036; **Variant2:** Includes all other INS units.

Fig. 1.6. Position of the accelerometer mass-center in housed Inertial Labs™ INS unit

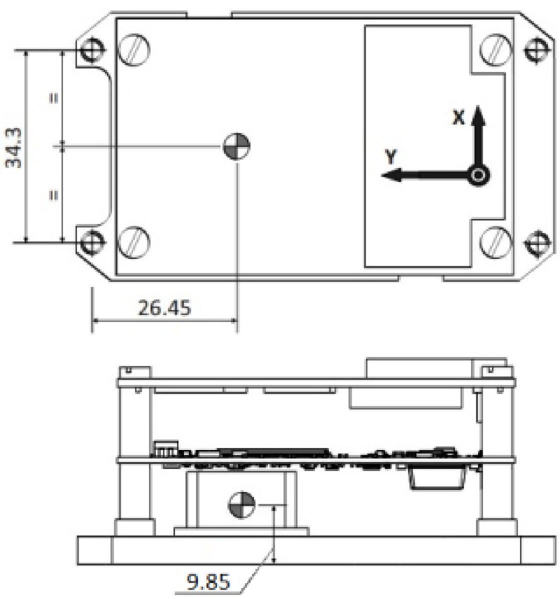


Fig. 1.7. Position of the accelerometer mass-center in Inertial Labs™ INS-OEM unit (in millimeters)

Since INS firmware version 3.2.2.6 INS can calculate position and velocity for any measuring point set by its position relative to the accelerometer mass-center of the INS unit. There are two ways to configure the measuring point, via INS GUI since version 2.0.45.248 from 2017-12-29 (see INS GUI User's Manual, section "4.2.1. "IMU" tab of "Devices options" window for details), and via the command 0x09 PV_meas_point (see Table 6.80) since INS firmware version 3.3.0.1.

1.2. Principles of the Inertial Labs™ INS Operation

Fig. 1.8 shows the operational diagram of the Inertial Labs™ INS. The INS uses gyros to measure absolute angular rate of the carrier object, accelerometers to measure the specific force (apparent acceleration of the object), magnetometers to measure components of the Earth magnetic field.

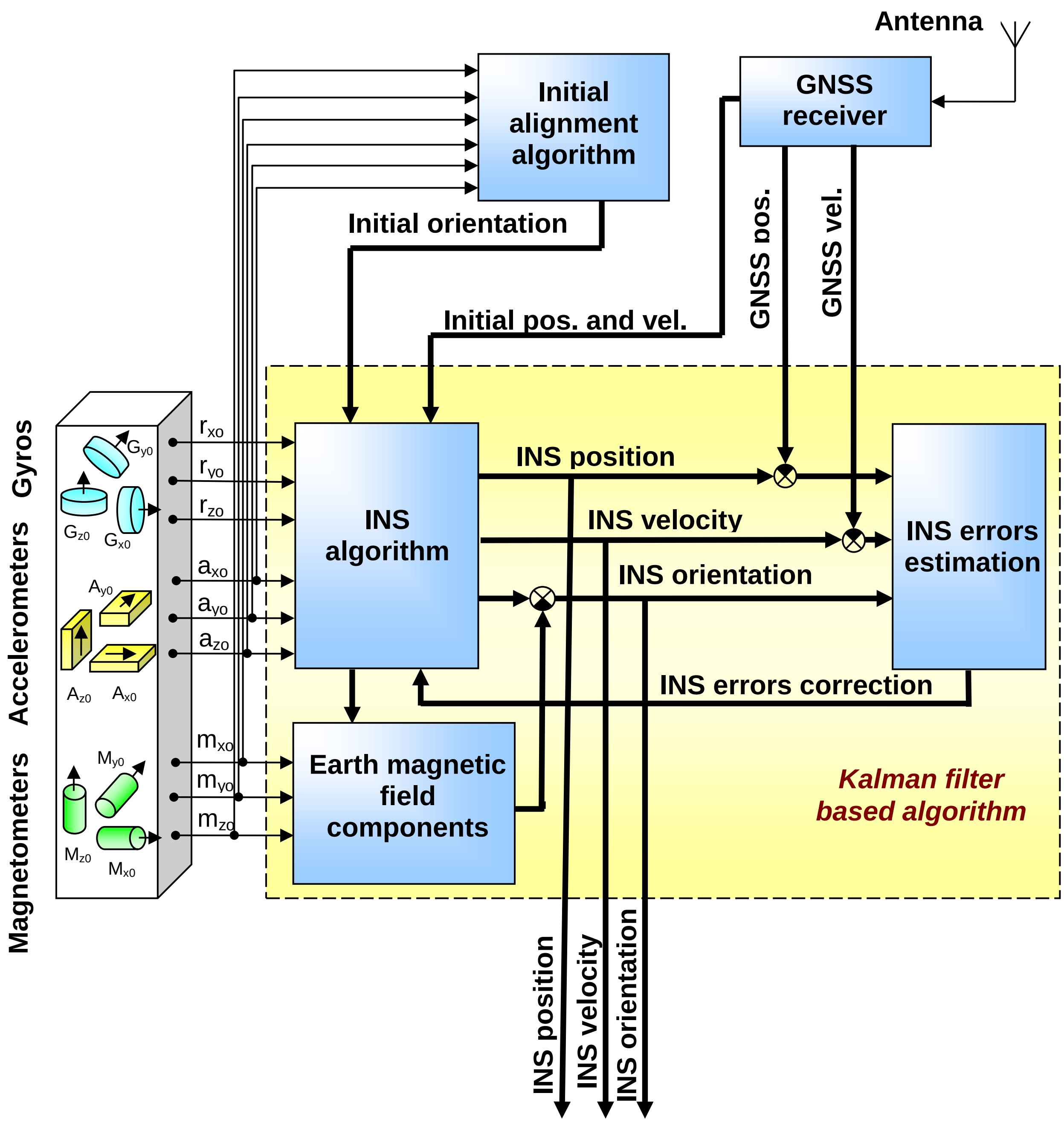


Fig. 1.8. Operational Diagram of the Inertial Labs™ INS

Orientation angles (heading, pitch and roll) are obtained by using special integration of gyros outputs with correction from GNSS position and velocity data. INS-D also utilises gyros correction by heading calculated as direction from the primary GNSS antenna to the secondary one.

Position (latitude, longitude and altitude) are calculated using special integration of accelerometers and known orientation. To avoid accumulation of the INS error they are estimated and compensated using Global Navigation Satellite System (GNSS) data provided by onboard receiver and pressure sensor data.

Also accelerometers are used to determine initial attitude of the INS, and. magnetometers are used to determine initial heading. In INS-D unit initial heading is calculated as direction from the primary GNSS antenna to the secondary one if GNSS data are available and RTK solution for heading is made by on-board GNSS receiver.

The base of the INS algorithm is robust Kalman filter which is used for estimation of the INS errors in orientation, position, velocity calculation and also gyros and accelerometers biases. For this purpose, the Kalman filter uses aiding information from GNSS about position and velocity, and also barometric altitude calculated fro, pressure sensor data.

As result of integration of all above data, the INS provides accurate calculation of stabilized heading, pitch and roll angles, latitude, longitude and altitude, east, north and vertical velocity. The Kalman filter automatically adjusts for changing of dynamic conditions.

Note the initial position and velocity are provided by the GNSS receiver if it has solution. If GNSS data are not available, then the initial position is taken from the INS nonvolatile memory. There the initial position can be changed using the LoadINSpar command (see Table 6.59, bytes #8-19) or using the INS GUI (that is easier).

After the Inertial Labs™ INS power on an initialisation of the onboard GNSS receiver starts that takes about 25 seconds. Then the INS is ready to receive commands from the host computer and to start required operation.

Once the Inertial Labs™ INS starts, the initial alignment process determines initial orientation angles as initial conditions for integration of gyros outputs. Also gyros bias is estimated using Kalman filter for next compensation.

There are two ways to configure features of INS algorithm, via INS GUI since version 2.0.47.308 from 2019-02-20 (see INS GUI User's Manual, section "4.3. Correction options" for details), and via appropriate commands since INS firmware version 3.3.0.1 (see ”2. Correction options” portion in Table 6.80).

1.3. True and magnetic heading calculation in INS-P

If the Inertial Labs™ INS uses magnetic sensors for heading reference, then it directly determines just magnetic heading. Then INS calculates true North heading using the current magnetic declination. Declination, also called magnetic variation, is the difference between true and magnetic North, relative to a point on the Earth. Declination angle vary throughout the world, and changes slowly over time. Magnetic declination angle can be entered directly to the Inertial Labs™ INS memory using special command (see Table 6.59, bytes #4-7) or the Inertial Labs™ INS GUI. Also, the magnetic declination can be calculated by INS itself based on calculated latitude, longitude, altitude and date.

Both INS unit on-board and INS GUI calculate the magnetic declination using the World Magnetic Model WMM2020 produced by the U.S. National Geophysical Data Center and the British Geological Survey. The World Magnetic Model is the standard model of the US Department of Defence, the UK Ministry of Defence, the North Atlantic Treaty Organization (NATO), and the World Hydrographic Office (WHO) navigation and attitude/heading referencing systems.

Since INS firmware version 2.2.0.2 the INS unit can calculate the magnetic declination continuously on-board. There are two ways to configure “Auto” mode, via INS GUI (see INS GUI User’s Manual, section ““IMU” tab of the “Devices Options” window” for details), and via the command 0x06 Mdec_AUTO (see Table 6.80) since INS firmware version 3.3.0.1.

Note using magnetometers for INS heading correction requires necessity of magnetometers calibration after INS unit installed on carrier object to compensate hard and soft iron effects of the carrier object on the INS heading determination accuracy (see section “6.8. Calibration of the Inertial Labs™ INS on hard and soft iron” for details).

1.4. Ground track angle vs heading

Ground track angle or the course over ground is determined using the position delta between two position computed. Track angle shows direction of vehicle motion in horizon plane.

For some carrier objects it is possible to use GNSS track angle instead of magnetic heading for INS correction. In such case magnetometers can be switched off, and INS does not require any calibration of magnetometers.

Ground track angle can be used as heading for ground vehicles where the direction of travel is coincident with the forward axis of the vehicle. But replacement of heading by the ground track angle may not be suitable for some marine or airborne applications, where the direction of travel may be different from the forward axis of the vehicle because of factors like a crab angle.

Also the ground track angle has no sense when the vehicle is stationary. But integration of INS with GNSS data allows to use GNSS track angle instead of magnetic heading for INS correction even at vehicle stops. Only initial vehicle movement is required to perform calculation of initial heading in INS.

1.5. Using GNSS heading in INS-D

More accurate INS heading correction than use of magnetometers or GNSS track angle can be provided in INS-D with two antennas installed along forward axis of carrier object. In INS-D magnetometers also can be switched off, and INS does not require any calibration of magnetometers in such case.

In contrast to using GNSS track angle, heading calculated on base of two antennas position does not require vehicle movement strictly in direction of the forward axis of the vehicle, moreover, vehicle can be stationary.

2. Scope and applicability

This Interface Control Document (ICD) provides details on mechanically mounting, the electrical connections, powering and software interface between the Inertial Labs™ INS and host computer. This document is intended for all parties requiring such information, including engineers and researchers responsible for implementing the interface.

3. Specifications

See separate document, Inertial Labs INS Datasheet.

4. Mechanical interface

Fig. 4.1 to Fig. 4.3 shows the outline drawings of the housed Inertial Labs™ INS and INS-OEM units. All dimensions are in millimetres.

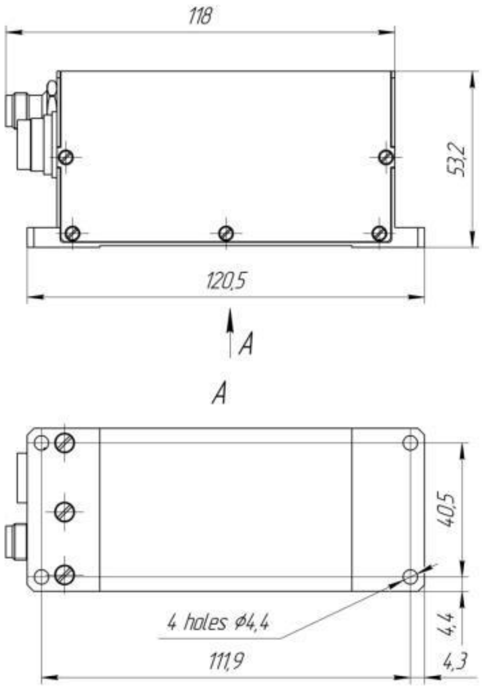


Fig. 4.1. The Inertial Labs™ INS-B/P outline drawing

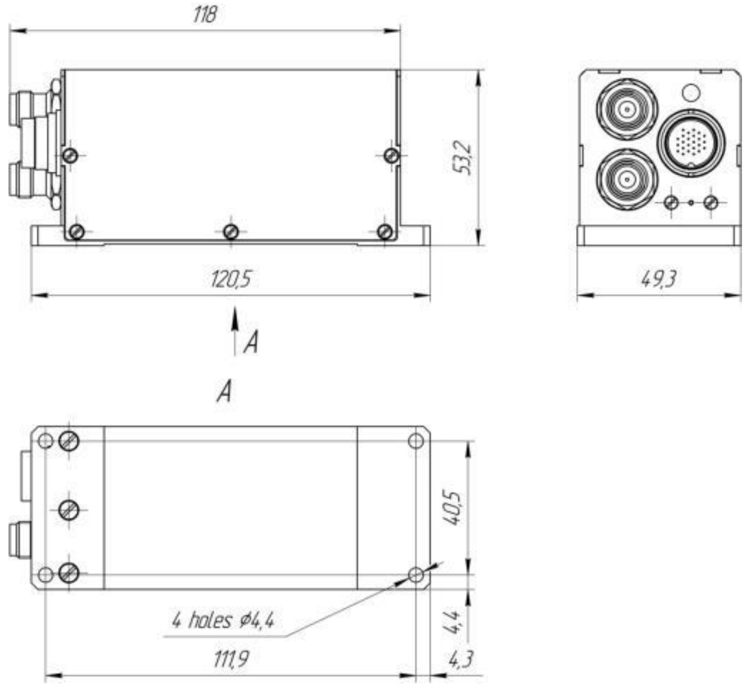


Fig. 4.2. The Inertial Labs™ INS-D/DL outline drawing

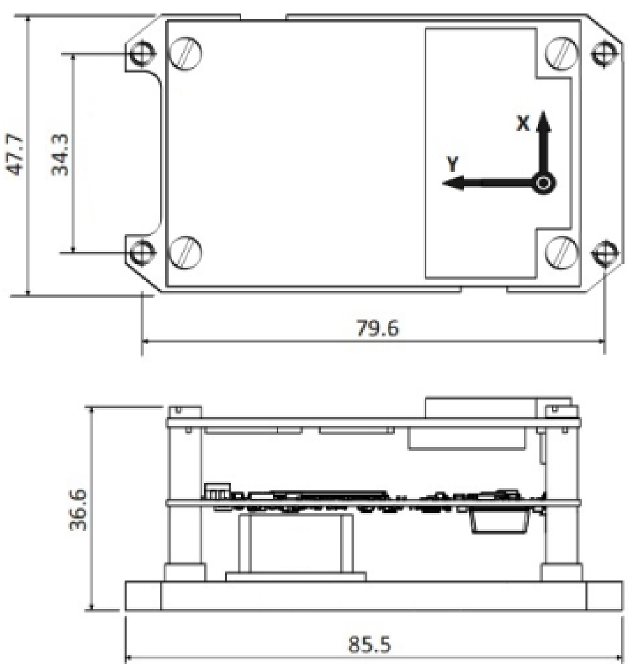


Fig. 4.3. The Inertial Labs™ INS-OEM outline drawing

4.1. Mechanically mounting the Inertial Labs™ INS

4.1.1. Mounting of the housed INS unit

The Inertial Labs™ INS housing has two base surfaces A and B (see Fig. 4.4) that are designed for the INS mounting during its run and testing.

Salient bottom base surface A has 4 holes ∅4.4 mm (see Fig. 4.4, positions 1) which are designed for the INS mounting. Lateral base surface B is designed for the INS alignment during mounting.

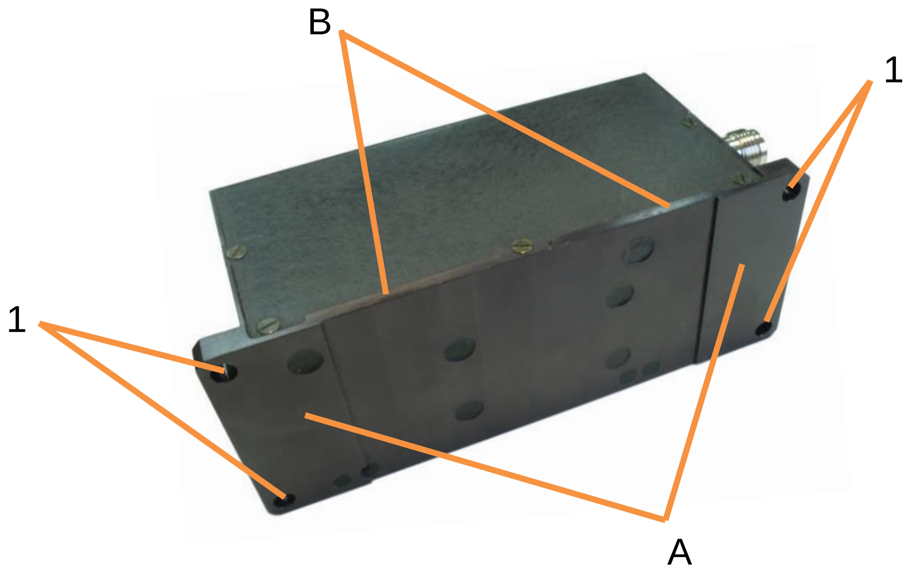


Fig. 4.4. INS mounting surfaces A, B and mounting holes 1

The Inertial Labs™ INS is factory calibrated with respect to the base surfaces A and B, thus it must be aligned within the host system (carrier object) with respect to these mounting surface, not the device edges.

When mounting Inertial Labs™ INS on your system, please pay attention to orientation of input axes X", "Y", "Z" marked on the cover of the INS (see Fig. 1.5). During the ordinary operation on the carrier object the INS is set on the surface A with the axis Y directed to the nose of the object.

Also the Inertial Labs™ INS can be mounted on the object in any known position (up to upside-down, upright etc.) relative to the object axes. Such mounting doesn't change right determination of the object orientation if angles of the INS mounting are correctly stored in the INS nonvolatile memory. See Appendix B. Variants of the Inertial Labs™ INS mounting relative to carrier object axes.

To obtain accurate attitude and heading, please remember that mounting is very important and mounting error can cause attitude and heading errors. When Inertial Labs^{TM} INS mounting please align it on two base surfaces A, B relative your system axes.

The Inertial Labs^{TM} INS is mounting on your system by using 4 holes ∅4.4 mm (see Fig. 4.4, positions 1).

Requirements to the mounting surface of the carrier object: flatness tolerance is 0.03 mm; undulation is Ra=1.25.

4.1.2. Mounting of the INS-OEM unit

The Inertial Labs^{TM} INS-OEM has two base surfaces A and B (see Fig. 4.5) that are designed for the INS-OEM mounting during its run and testing.

Salient bottom base surface A has 4 holes ∅4 mm (see Fig. 4.5, positions 1) which are designed for the INS mounting. Lateral base surface B is designed for the INS alignment during mounting.

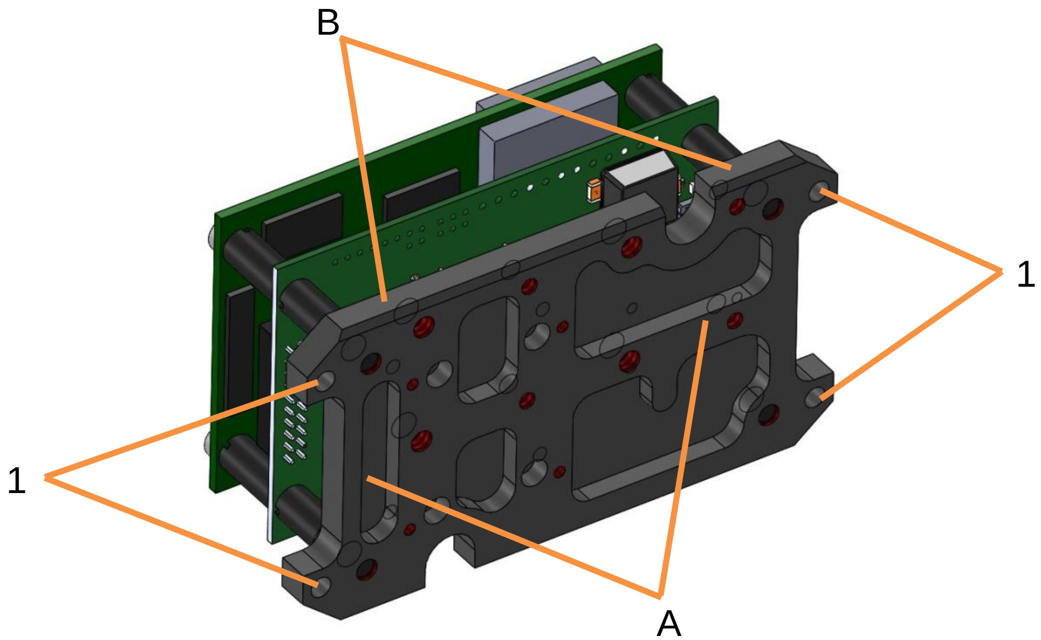


Fig. 4.5. INS mounting surfaces A, B and mounting holes 1

The Inertial Labs^{TM} INS-OEM is factory calibrated with respect to the base surfaces A and B, thus it must be aligned within the host system (carrier object) with respect to these mounting surface, not the device edges.

When mounting Inertial Labs™ INS-OEM on your system, please pay attention to orientation of input axes “X", "Y", "Z" marked on the cover of the INS (see Fig. 1.5). During the ordinary operation on the carrier object the INS is set on the surface A with the axis Y directed to the nose of the object.

Also the Inertial Labs™ INS-OEM can be mounted on the object in any known position (up to upside-down, upright etc.) relative to the object axes. Such mounting doesn’t change right determination of the object orientation if angles of the INS mounting are correctly stored in the INS nonvolatile memory. See Appendix B. Variants of the Inertial Labs™ INS mounting relative to carrier object axes.

To obtain accurate attitude and heading, please remember that mounting is very important and mounting error can cause attitude and heading errors. When Inertial Labs™ INS-OEM is mounting please align it on two base surfaces A, B relative your system axes.

The Inertial Labs™ INS-OEM is mounting on your system by using 4 holes ∅4 mm (see Fig. 4.5, positions 1).

Requirements to the mounting surface of the carrier object: flatness tolerance is 0.03 mm; undulation is Ra=1.25.

4.2. Installation of single GNSS antenna

Usually the INS unit and GNSS antenna are installed in different places of the carrier object. Moreover, placement of the antenna close to the INS unit is undesirable because of the antenna impact on the INS magnetometers.

While the best place for the INS unit is center of gravity of the carrier object, the GNSS antenna must of course be placed with a clear view of the sky with a sufficient ground plane.

After the INS unit and GNSS antenna installation on the carrier object <u>it is necessary to measure the antenna position</u> relative to the accelerometer mass-center of the INS unit (see Fig. 1.6 and Fig. 1.7), in the object axes – on the right, forward and up. Then it is necessary t<u>o store these coordinates</u> to the INS nonvolatile memory using the LoadINSPar command (see Table 6.59, bytes #35-40) or using the INS GUI (that is easier).

Fig. 4.6 shows positive right, forward and up directions of the antenna position relative to the INS unit.

Important notes:
1. If after the INS mounting its axes (see Fig. 1.5) are parallel to the carrier object axes, then the antenna coordinates should be measured in the directions of X, Y and Z axes.
2. On the other hand, the INS unit can be mounted on the object in any known position (up to upside-down, upright etc., see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes). In that case please set the GNSS antenna coordinates measures just in the object axes (on the right, forward and up directions), but not in the INS axes.

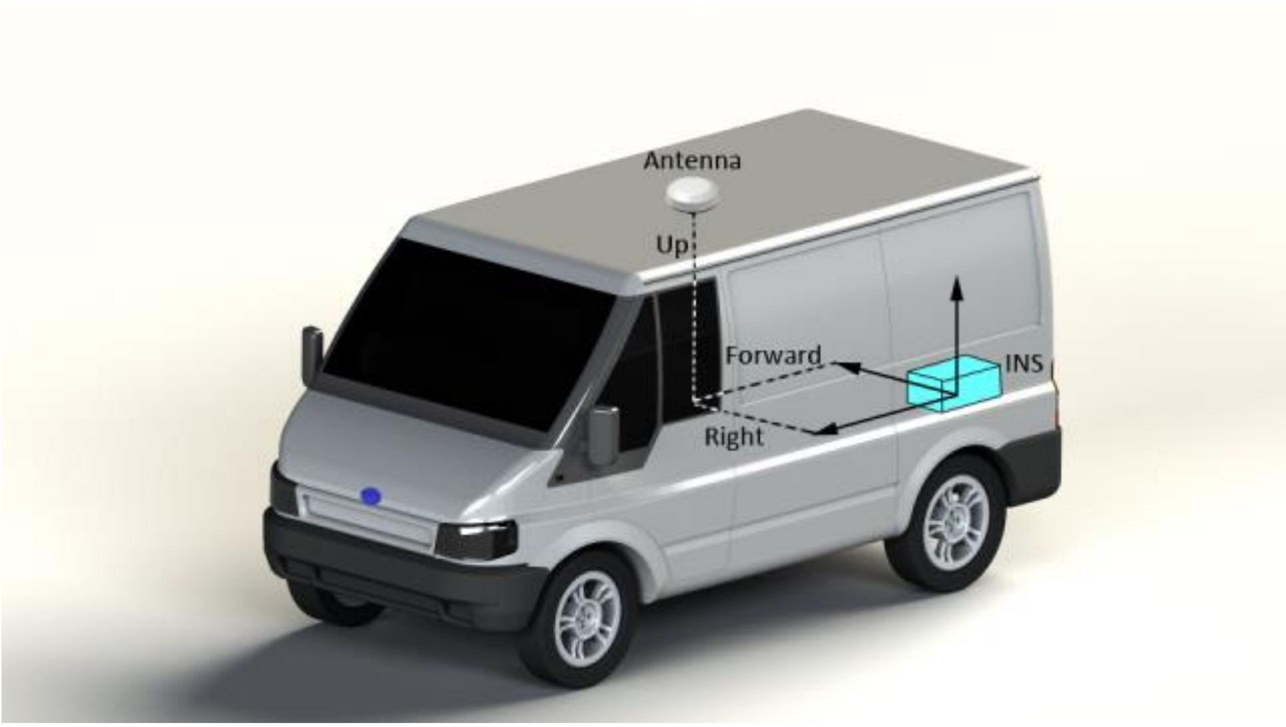


Fig. 4.6. Determination of the GNSS antenna position relative to the INS unit (positive directions)

4.3. Installation of two GNSS antennas for INS-D operation

The Inertial Labs™ INS-D uses heading calculated by dual-antenna GNSS receiver for the INS correction. Two antennas should be installed in parallel to the longitudinal axis of the carrier object to allow GNSS receiver to measure object heading accurately. At this the secondary antenna is installed ahead the primary antenna, so direction from the primary to the secondary antenna is forward for the carrier object, see Fig. 4.7.

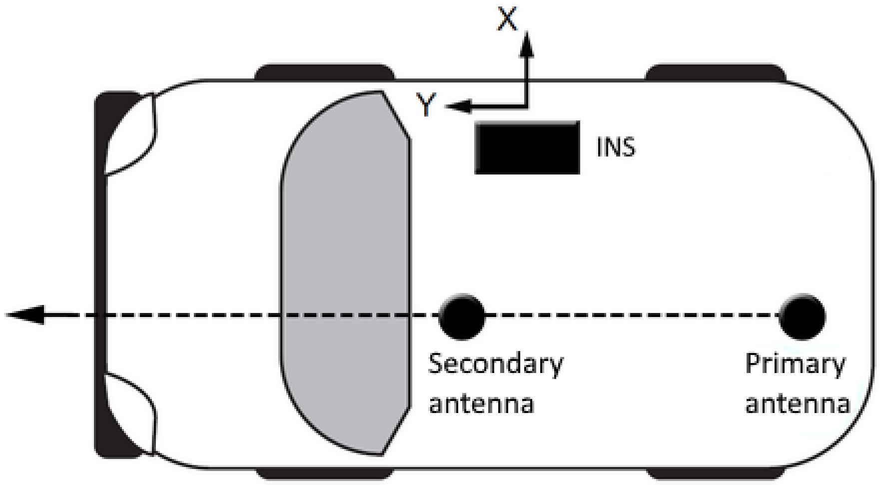


Fig. 4.7. Installation of the primary and secondary GNSS antennas on carrier object

Requirements for the primary antenna installation are the same as described in section "4.2. Installation of single GNSS antenna". Position of the primary antenna relative to the accelerometer mass-center of the INS unit must be measured and stored to the INS nonvolatile memory.

Since the firmware version 2.9.1.7 the INS-D algorithm has the feature of the secondary antenna installation in arbitrary (but known) position relative to the INS-D unit and the primary antenna. It is possible to set secondary antenna location in two ways: by specifying its position in meters relative to the accelerometer mass-center of the INS or by specifying the antennas baseline orientation in degrees. There are two ways to configure these parameters, via INS GUI since version 2.0.40.196 from 2017-06-23 (see INS GUI User's Manual, section "4.2.1 "IMU" tab of the "Devices Options" window" for details), and via the commands 0x0B Ant2_pos, 0x0C Ant2_pos_angl (see Table 6.80) since INS firmware version 3.3.0.1.

Fig. 4.8 shows Alpha and Beta angles of the antennas baseline orientation relative to the INS unit. Alpha angle is measured in horizon plane of the object, clockwise direction is positive. Beta angle is measured in vertical plane, positive is up. Note that on the Fig. 4.8 there are positive Alpha and negative Beta angles.

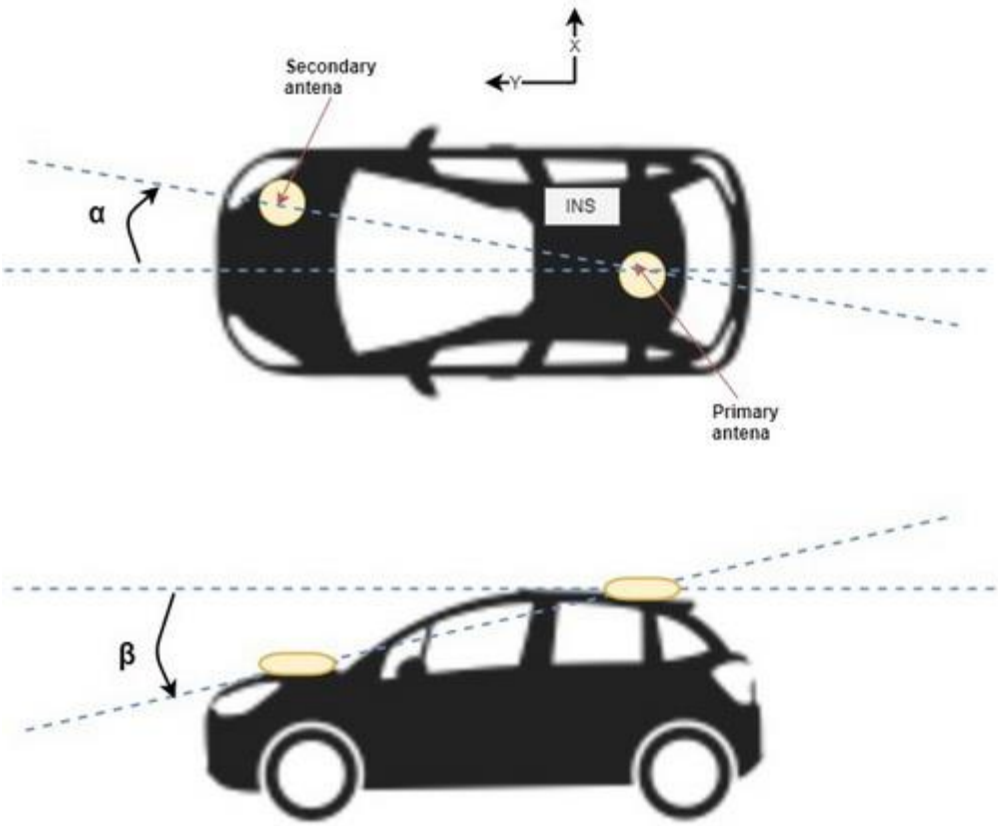


Fig. 4.8. Angles of baseline of two GNSS antennas installation on carrier object

4.4. Where to install the Inertial Labs™ INS and its antenna for tests

For optimal performance, you should mount the Inertial Labs™ INS with the following considerations in mind.

- ***The Inertial Labs™ INS should be mounted in a physically stable location***

Choose a location that is isolated from excessive shock, oscillation, and vibration. Special rotary table must be used for the Inertial Labs™ INS accuracy testing, that mounted on a special testing basement which is free from the laboratory oscillations and vibrations.

Tests on vibrations and shocks are fulfilled separately from the main accuracy tests.

- ***Install the Inertial Labs™ INS and GNSS antenna on the same base***

For test of the INS position and linear velocity calculation, it is necessary to connect the active GNSS antenna(s) to the INS. Both INS unit and the antenna(s) should be installed immovable each to other. Position of the antenna(s) relative to the INS unit should be measured and stored to the INS nonvolatile memory (see sections “4.2. Installation of single GNSS antenna” and “4.3. Installation of two GNSS antennas for INS-D operation”, for details).

4.5. Where to install the Inertial Labs™ INS on the object

At the Inertial Labs™ INS installation on a carrier object it is necessary to follow the recommendations listed in the section 4.4 whenever it is possible.

- ***It is preferable to locate the Inertial Labs™ INS as close to the center of gravity of the object as possible***

4.6. The installation peculiarity for the Inertial Labs™ INS-P unit

The Inertial Labs™ INS-P has magnetometers with wide dynamic range and their sophisticated calibration algorithms allow them to operate in different environments. For optimal performance it is necessary to follow the items listed in the 4.4 and 4.5 sections with additional recommendations.

- ***Inertial Labs™ INS should be installed on an object as far as possible from large ferromagnetic masses of the object and powerful sources of magnetic, electrical and electro-magnetic fields***

Inertial Labs™ INS software allows compensation of hard and soft iron effects of the carrier object on the heading measurement accuracy. For this purpose, field calibration of the INS magnetometers is provided. This calibration does not require any additional equipment, but it requires turns of the carrier object, on which the INS is mounted.

Note that the above field calibration is correct until the residual magnetic field of the object surrounding the INS is changed. If this field is changed due to displacement of ferromagnetic masses of the object or magnetic field sources, the INS should be re-calibrated.

Field calibration procedure of the Inertial Labs™ INS can be performed by two means:

- by INS itself using special commands described in the section 6.8;
- using the Inertial Labs™ INS GUI.

The INS GUI provides more variants of the field calibration and is more convenient for use, but it requires connection of the INS to PC. Calibration of the INS itself is performed without its disconnection from the host system on the carrier object.

More detailed description of the field calibration procedure is given in the section “6.8. Calibration of the Inertial Labs™ INS on hard and soft iron”.

For INS-P heading accuracy test

- ***Locate the Inertial Labs™ INS away from local sources of magnetic fields***

The place for testing must not have ferromagnetic (magneto-susceptible) materials and the lab room itself must have the level of intrinsic magnetic and electro-magnetic fields suitable for the magnetic heading system testing:

- inside and near the lab room there must be no powerful source of magnetic, electrical and electro-magnetic fields. The magnetic field intensity must not be different from the Earth magnetic field intensity at the test site more than 0.01%;
- small ferromagnetic objects must be as far as 3 meters from the test table. Large size ferromagnetic objects such as cars and trucks must be as far as 15 m from the table;
- it is necessary to conduct a regular check-up of the magnetic field uniformity inside the lab room.

It is highly recommended to degauss the INS before heading test to remove permanent magnetization of some components in the INS (if you accidentally expose the unit to a large magnetic field). You can use a hand-held degausser (tape eraser) to demagnetize the INS. Most audio and video degaussing units can be used. Follow the instructions for your demagnetizer.

4.7. Installation of OS3D-FG Stand Alone Magnetic Compass

The Inertial Labs™ INS can be factory configured to use the external OS3D-FG Stand Alone Magnetic Compass (SAMC) instead of on-board magnetometers to measure components of the Earth magnetic field.

The SAMC should be firmly mounted to the carrier object, so SAMC location and orientation relative to the INS unit should not change under any circumstances.

The SAMC can be installed with any orientation relative to INS unit. But then it is necessary to align the SAMC axes to INS axes (see INS GUI User’s Manual, rev.2.17 and newer, section “10.9. INS and OS3D-FG SAMC axes alignment”).

The SAMC should be installed on carrier object as far as possible from large ferromagnetic masses of the object and powerful sources of magnetic, electrical and electro-magnetic fields. After the SAMC installation it is highly recommended to perform the hard/soft iron calibration of the SAMC magnetometers with the same procedure as for INS internal magnetometers (see section 6.8).

Using external magnetic compass option can be enabled or disabled, via INS GUI (see INS GUI User Manual, section “4.2.4.3 Using external magnetic compass”), or via the command 0x33 Ext_MC (see Table 6.80) since INS firmware version 3.3.0.1.

Please contact Inertial Labs to receive INS unit with OS3D-FG SAMC input support.

5. Electrical Interface

5.1. Electrical interface of the housed INS unit

All connectors and LED indicator are placed on the back side of the Inertial Labs™ INS, see Fig. 5.1 and Fig. 5.2.

Fig. 5.1. Back side of the Inertial Labs™ INS-B and INS-P

1 – interface connector; 2 – GNSS antenna TNC connector; 3 – LED indicator; 4 – pressure sensor

Fig. 5.2. Back side of the Inertial Labs™ INS-D

1 – interface connector; 2 – primary GNSS antenna TNC connector; 3 – secondary GNSS antenna TNC connector;
4 – LED indicator; 5 – pressure sensor

The Inertial Labs™ INS has TNC female connectors for the GNSS antenna connection.

Also the Inertial Labs™ INS has the Binder Series 723 male 24 pin connector, part # 09 0497 90 24, for electrical connection to the host system.

The host system should have a cable with appropriate mating connector – the Binder Series female 24 pin cordset, part #99 5896 15 24.

Note: Inertial Labs™ INS units with 12 pin (RS-232) and 19 pin (RS-422) connectors are not recommended for the new design. Electrical interface of such INS units is described in Appendix F.

Fig. 5.3 shows connector pinout of the Inertial Labs™ INS with 24 pin interface. Table 5.1 contains pin diagram of this connector and appropriate color of wires in 24-pin multiport cable part number #99 5896 15 24.

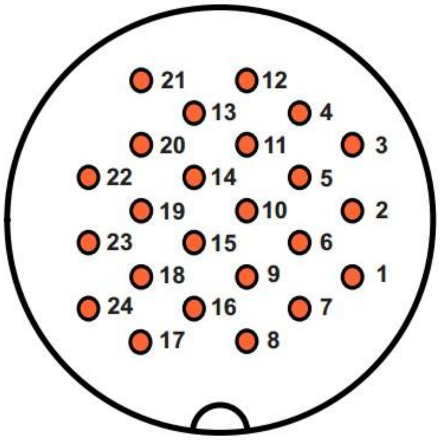


Fig. 5.3. Connector pinout of the Inertial Labs™ INS with 24 pin interface (mating side of the connector)

Table 5.1. Pin diagram of the connector of the Inertial Labs™ INS with 24 pin interface

Signal	Pin	Wire color	Signal	Pin	Wire color
POWER	1	ORNG	RS422-B1	13	VIOLT
GROUND	2	ORNG	RS422-A1	14	VIOLT
GROUND	3	RED	RS232-RX2	15	DK.GR
TRIGGER	4	RED	RS232-TX2	16	DK.GR
PPS	5	BRWN	RS232-RX1	17	LT.GR
GROUND	6	BRWN	RS232-TX1	18	LT.GR
GROUND	7	DK.BL	RS232-RX3	19	YELL
GROUND	8	DK.BL	RS232-TX3	20	YELL
RS232-RX4	9	WHITE	ETHX+	21	PINK
RS232-TX4	10	WHITE	ETHX-	22	PINK
RS422-Y1	11	GRAY	ERHX+	23	LT.BL
RS422-Z1	12	GRAY	ERHX-	24	LT.BL

Notes:
1. The names of the signals are given relative to device, i.e. RS232-RX lines are for data coming to device, while RS232-TX lines are for data coming from the device.
2. The functions of the signals of RS422 interface: A and B lines are for data coming to device, while Y and Z lines are for data coming from device.

The Inertial Labs™ INS has four COM-ports with RS-232 interface on default:

- COM1 is the main. It is used for commands and data transfer between the Inertial Labs™ INS and the host computer.
- COM2 is used for output of the raw GNSS receiver data (see section 6.11. Post-processing of the INS and GNSS data) or NMEA data set.
- COM3 has three functions:
  - to receive data for differential corrections of GNSS (DGPS mode);
  - or to output of the raw GNSS receiver data (see section 6.11.1. Raw GNSS receiver data) – since INS firmware version 2.8.0.6;
  - or to output GPRMC messages (since INS firmware version 2.2.0.3).
- COM4 has the next functions:
  - to receive external data from a device with RS232 interface (like odometer data – see section 6.6.1);
  - to output IMU data with their maximum frequency (see section 6.2.19);
  - to output GPRMC messages (since INS firmware version 3.0.1.6);
  - to output HEHDT messages (since INS firmware version 3.0.1.6);
  - to output through port COM4 the same data that are transmitted via main port COM1 (duplication).

Also, the Inertial Labs™ INS has CAN1-port (see section 5.8), and can be equipped with Ethernet port (see section 5.7) or added additional CAN2-port (see section 5.9).

There are two ways to configure INS ports, via INS GUI (see INS GUI User's Manual, section "4.2. "Devices options"" for details), and via appropriate commands since INS firmware version 3.3.0.1 (see "1. Devices options" portion in Table 6.80).

Table 5.2. Electrical specifications

Parameter	Conditions	Min	Typical	Max	Units
Input Supply		+9	+12V	+36V	Volts DC
INS-B Current	V_{DD} = +12V	200	220	250	mA
INS-P Current	V_{DD} = +12V	225	245	275	mA
INS-D Current	V_{DD} = +12V	325	345	355	mA

At the Inertial Labs™ INS operations, it should be connected to the host system that provides command interface described in the section 6 and the INS powering.

5.2. Electrical interface of the INS-OEM units

5.2.1. 26-pin connector electrical interface

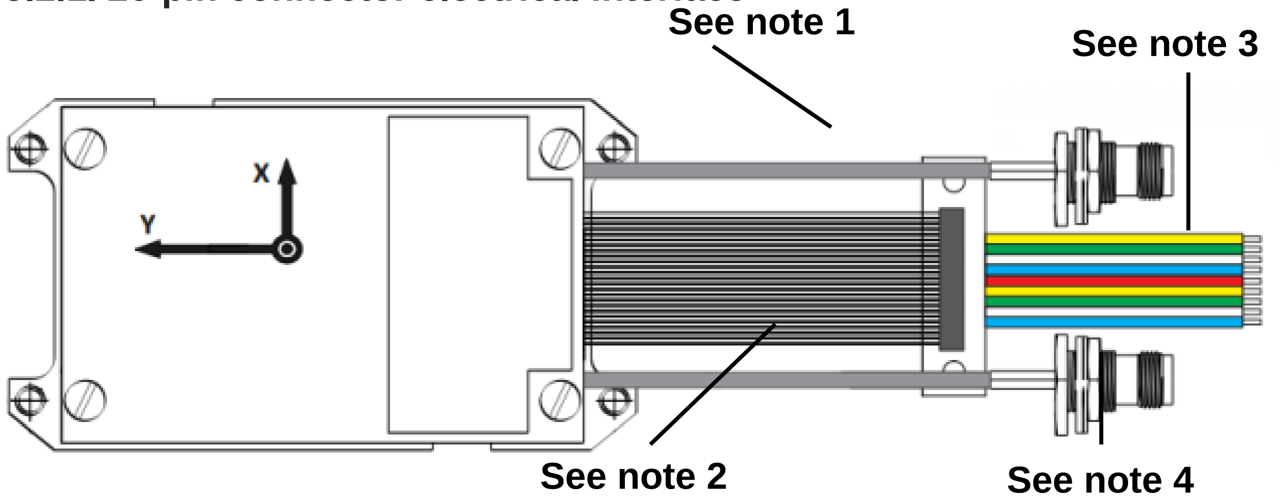


Fig. 5.4. INS-OEM assembly drawing

Notes:

1. TNC Female Bulkhead to MMCX Plug Right Angle Cables 150mm length RG174 Coax
2. 26 lines ribbon cable (can be converted to open end wires)
3. Optional connection - 26 AWG stranded wires by Alpha Wire (250mm)
4. Present only in INS-D-OEM

Table 5.3. 26-pin diagram of the connector of the Inertial Labs™ INS with Interface harness

Interface harness

1,2	POWER IN	RED
3,4	GROUND	BLACK
16	Rs232 – Rx1	BLUE
18	Rs232 – Tx1	ORANGE
12	Rs232 – Rx2	WHITE
14	Rs232 – Tx2	GREEN
15	Rs232 – Rx3	YELLOW
17	Rs232 – Tx3	BROWN
10	PPS	PURPLE
8	MARK IN	WT/RED

RS422 version

21	RS422-A	BLUE
22	RS422-B	ORANGE
19	RS422-Y	WT/GREEN
20	RS422-Z	YL/GREEN

4 ports version

11	RS232-RX4	WT/GREEN
13	RS232-TX4	YL/GREEN

Ethernet version

23	ETHTX+	BLUE
24	ETHTX-	ORANGE
25	ETHRX+	WT/GREEN
26	ETHRX-	YL/GREEN

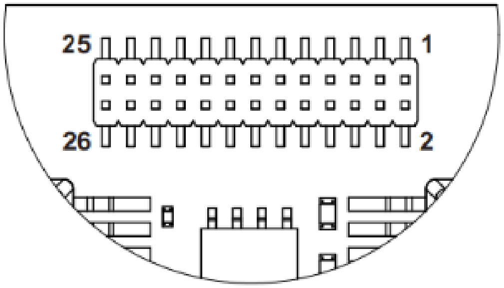


Fig. 5.5. Ribbon cable connector 26-pin header 20021 121-00026T4LF by Amphenol

5.2.2. 14-pin connector electrical interface

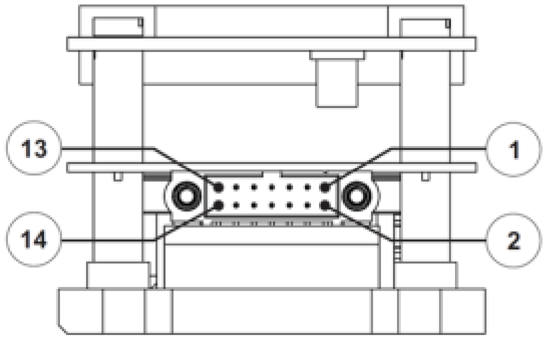


Fig. 5.6 Device side: 14-pin connector M80-5401442 by Harwin (Ethernet is not available in this configuration)

Fig. 5.7 Cable side: M80-4601405 by Harwin

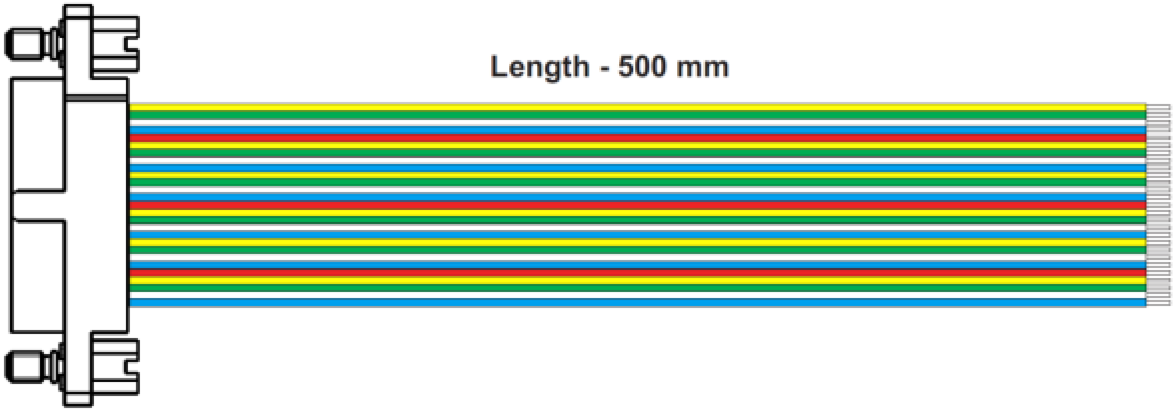


Fig. 5.8.

Notes:

1. The names of the signals are given relative to the device.
   I.e. the Rx pin is the input pin of the INS, Tx is the output one.
   RS422-A and RS422-B pair is used by device for data receive
   RS422-Y and RS422-Z pair is used by device for data transmit
2. Wires are taken from a kit 24STRPTFEKIT10COLOR100 by Remington Industries.

Table 5.4. Main harness (RS232 option, V2.0)

1	POWER IN		RED	
2	GROUND		BLACK	
3	GNSS EV2	TWISTED	PURPLE	
4	GNSS PPS		GREY	
5	RS232-RX2		GREY	
6	RS232-TX2		BROWN	
7	RS232-RX4 (CAN_L)		YELLOW	
8	RS232-TX4 (CAN_H)		WHITE	
9	RS232-RX3		PURPLE	
10	RS232-TX3		GREEN	
11	NC	TWISTED	WHITE	
14	NC		ORANGE	
12	RS232-TX1	TWISTED	WHITE	
13	RS232-RX1		BLUE	

Notes:

1. PPS and EV2 conductors are twisted;
2. RS232-RX1 and NC conductors are twisted;
3. RS232-TX1 and NC conductors are twisted.

Table 5.5. Main harness (RS422 option, V2.1)

1	POWER IN		RED	
2	GROUND		BLACK	
3	GNSS EV2	TWISTED	PURPLE	
4	GNSS PPS		GREY	
5	RS232-RX2		GREY	
6	RS232-TX2		BROWN	
7	RS232-RX4 (CAN_L)		YELLOW	
8	RS232-TX4 (CAN_H)		WHITE	
9	RS232-RX3		PURPLE	
10	RS232-TX3		GREEN	
11	RS422-TX+	TWISTED	WHITE	
14	RS422-TX-		ORANGE	
12	RS422-Rx+	TWISTED	WHITE	
13	RS422-Rx-		BLUE	

Notes:

1. PPS and EV2 conductors are twisted;
2. RS422-A and RS422-B conductors are twisted;
3. RS422-Y and RS422-Z conductors are twisted.

5.3. Connection of the housed Inertial Labs™ INS to the host computer for tests

The delivery set for the INS electrical connection to PC is provided by the Inertial Labs and includes:

- 24-pin multiport development kit (see Fig. 5.9) with the cable for housed units.
- USB-to-Serial converters for the INS COM ports with RS-232 and RS-422 interface connection to the USB ports of PC;
- AC/DC adapter.

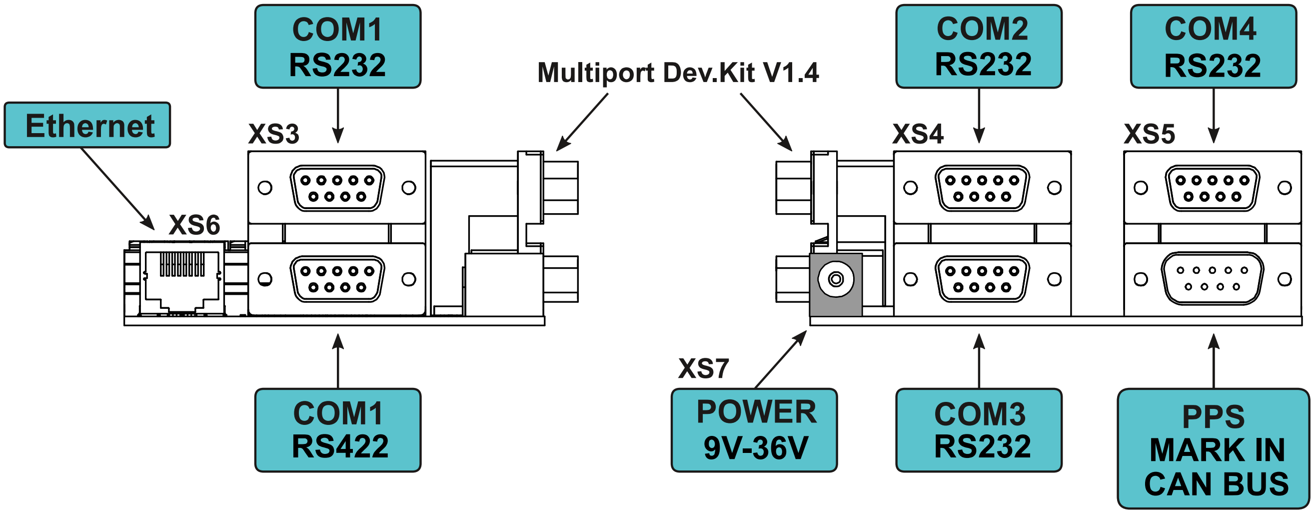


Fig. 5.9. The Inertial Labs™ 24-pin multiport cable development kit

Fig. 5.10 shows the diagram of the interface cable for the Inertial Labs™ INS connection to the COM-ports of host computer and to the DC power source.

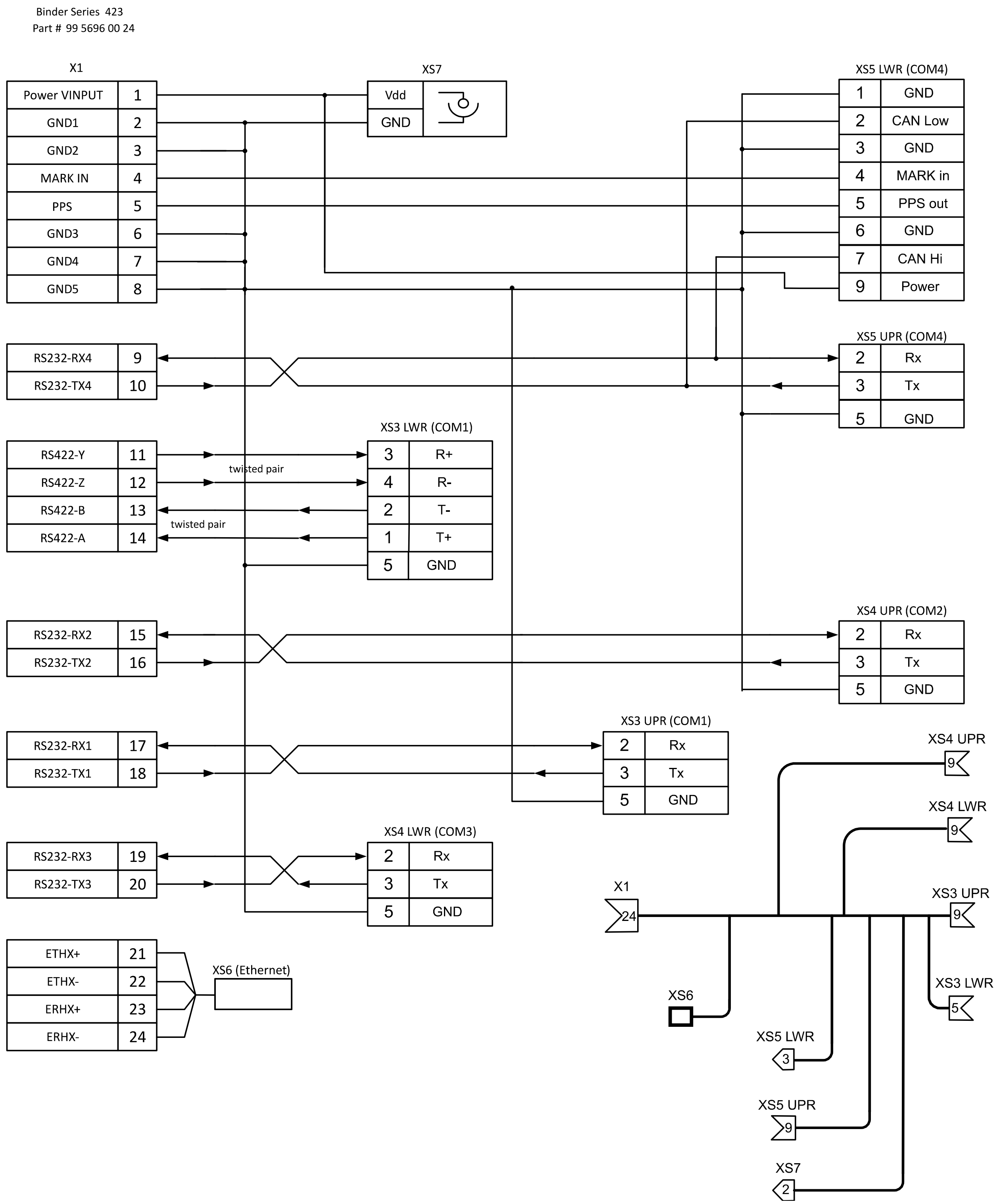


Fig. 5.10. The diagram of the interface cable for the Inertial Labs™ INS connection to COM-ports of the host computer and to the AC/DC adapter

5.4. Connection of encoder-based odometer to INS

The Inertial Labs™ INS can be factory configured to receive pulse/bi-phase signals from encoder. In such case pins assigned to COM4 port (see Table 5.1) are used to connect to encoder lines as Table 5.6 shows.

Table 5.6. Diagram of bi-phase encoder connection to COM4 lines of the Inertial Labs™ INS

Incremental encoder, contact function	COM4 INS connector, pin number	24-pin INS connector, pin number
OUT A	2	10
OUT B	3	9
GND	5	8

Important note: Please contact Inertial Labs for details regarding the connection, since pinout of it differs based on partnumber of a particular device.

5.5. PPS description

The Inertial Labs™ INS outputs the pulse per second (PPS) signal generated by GNSS receiver. Appropriate pin of the INS main connector provides the PPS signal (see Table 5.1).

The leading edge of the PPS pulse is always the trigger / reference:

- Negative – generates a normally high, active low pulse with the falling edge as the reference;
- Positive – generates a normally low, active high pulse with the rising edge as the reference.

GNSS receiver produces a TTL (Transistor-transistor logic) level pulse. Either 3.3V or 5V TTL levels are available. Note that 3.3V level is available by default and 5V level is optional (please contact Inertial Labs for this option).

PPS pulse is shown on the Fig. 5.11 and Fig. 5.12.

Since the INS firmware version 2.2.0.3 the pulse polarity, period and pulse width are user-adjustable and can be set using two ways, via INS GUI since version 2.0.22.84 from 2016-04-22 (see INS GUI User's Manual, section "13.1. Control of PPS output signal"), and via the command 0x51 PPS_switch, PPS_polarity, PPS_period, PPS_pulse_width (see Table 6.80) since INS firmware version 3.3.0.1.

By default, GNSS receiver generates a normally high, active low (negative polarity) pulse with the falling edge as the reference. Default PPS period is 1 second, pulse width is 1000 microseconds.

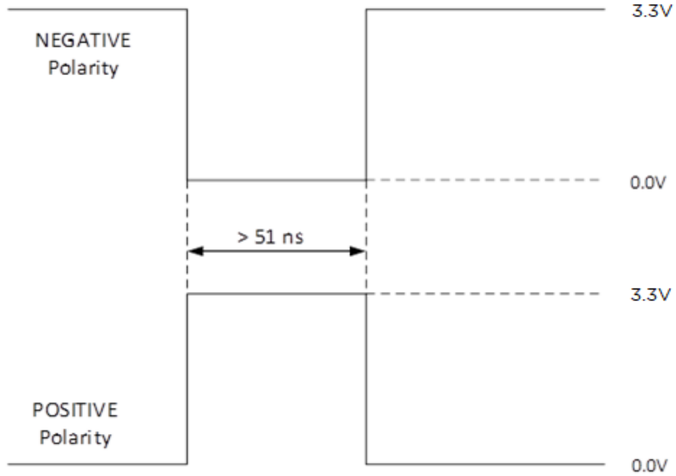


Fig. 5.11. PPS pulse; 3.3V TTL

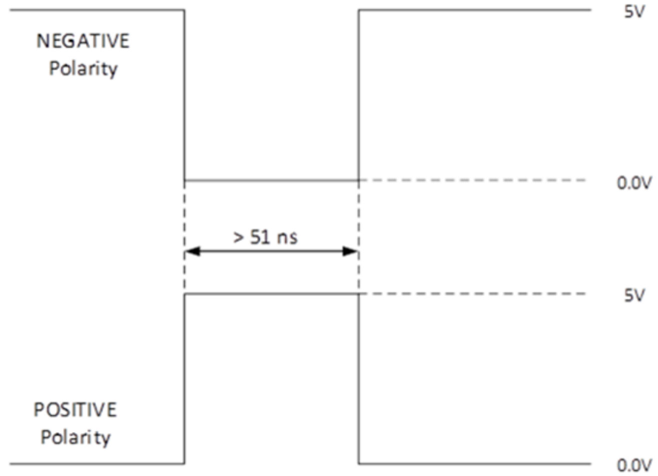


Fig. 5.12. PPS pulse; 5V TTL (optional)

<u>Note</u>: Cable set provided with the Inertial Labs INS is designed for INS connection to PC and therefore it does not transfer PPS signal. To receive PPS signal it is necessary to make another cable with PPS wire. Please contact Inertial Labs to purchase cable with PPS signal transferring.

5.6. Mark input description

The Inertial Labs™ INS provides MARK IN pin in the main connector (see pin diagram of the Inertial Labs™ INS connector in Table 5.1).

Since the INS firmware version 2.2.0.3 the mark inputs are used to trigger specific GNSS raw receiver data. TTL mark pulse configuration is the same as Fig. 5.11 shows. There are two ways to configure mark input signal, via INS GUI since version 2.0.22.84 from 04/22/2016 (see INS GUI User's Manual, section "13.2. Processing of mark input signal"), and via the command 0x50 MARK_switch, MARK_polarity, MARK_timebias, MARK_timeguard (see Table 6.80) since INS firmware version 3.3.0.1.

To allow mark inputs the MARK_switch should be set to 1 (see section 6.4.1).

When a pulse is detected at mark input then the GNSS receiver generates asynchronous MARK2POS and MARK2TIME logs which are added to the raw GNSS data.

5.7. Connection of the Inertial Labs™ INS with Ethernet interface to the host computer

Inertial Labs™ INS unit, which have built-in Ethernet port, is equipped with USR-TCP232-ED2 module made by USR-IOT. This document refers to the website of USR-IOT as to the location of required software tools and datasheets, http://www.usriot.com/p/modbus-tcp-ethernet-ip-modules/

Such Inertial Labs™ INS is delivered to a customer with the multi-conductor cable. One side of the cable should be connected to the connector on the rear wall of the device. The other side of the cable is attached to the breakout board with numerous connectors on it (see Fig. 5.13). The connectors needed for Ethernet communication are XS6 (used for a data) and XS7 (used for a power). The INS with Ethernet retains the communication via a serial port as well. So XS3 can also be used as RS232 port if necessary.

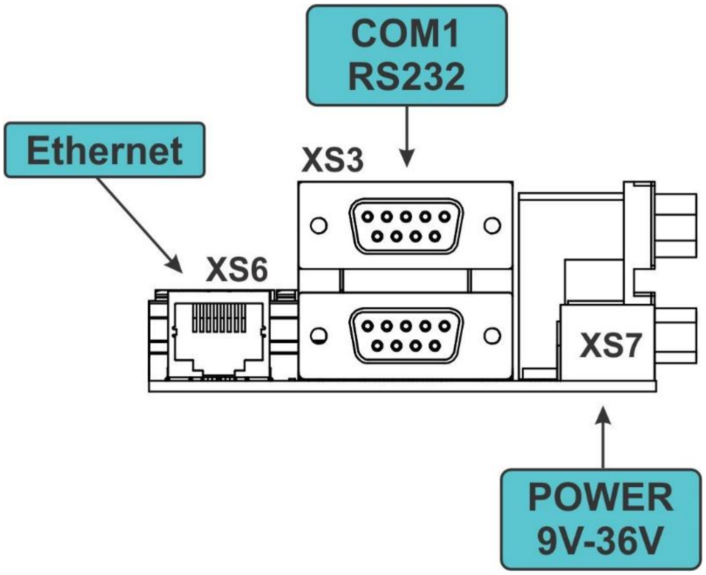


Fig. 5.13.

Note: If INS unit is equipped by internal datalogger, then it automatically excludes utilizing Ethernet for data/command transfer. In such case, Ethernet can be used for datalogger configuration only.

See detailed description of Ethernet settings in the INS GUI User's Manual, Appendix H. "Using Ethernet port for communication with the Inertial Labs™ INS".

5.8. Connection of the Inertial Labs™ INS with CAN interface to the host computer

Inertial Labs™ INS unit has built-in CAN port. Since firmware version 3.2.4.0 INS can output CAN2.0A messages. Since firmware version 3.2.8.1 CAN2.0B messages are provided, too, and possibility of CAN messages choice is implemented. There are two ways to configure CAN settings, via INS GUI since version 2.0.47.308 from 2019-02-20 (see INS GUI User's Manual, section "10.8. Operations with CAN data" for details), and via appropriate commands since INS firmware version 3.3.0.1 (see "1.5 CAN/COM4" portion in Table 6.80).

If the CAN data output is enabled, then pins originally assigned to COM4 port (see Fig. 5.3 and Table 5.1) become connected to CAN lines as Table 5.7 shows.

In case if 24-pin multiport cable development kit (see Fig. 5.9) is being used then CAN data appears on connector "PPS / MARK IN / CAN BUS" according to the pin diagram Table 5.8.

Table 5.7. Diagram of CAN lines of the 24-pin connector of Inertial Labs™ INS

Signal	24-pin INS connector, pin number
CAN_H	10
CAN_L	9

Table 5.8. Diagram of CAN lines connection to "PPS / MARK IN / CAN BUS" connector of 24-pin multiport cable development kit

Signal	DB-9M male connector, pin number
Trigger	4
PPS	5
Ground	1
Ground	3
Ground	6
CAN_H	7
CAN_L	2
Power	9

Notes:

1. Ordinary communication with INS unit through COM1, COM2, COM3 ports is still available at INS data output through CAN port.

2. Once user has access to the vehicle's CAN BUS and knows which CAN messages indicate the velocity of the car, then it is possible to receive these messages through INS CAN port and to use them as odometer data (see section 6.6.3).

See section 6.2.18 to know which massages can be transmitted through CAN port.

5.9. Connection of the Inertial Labs™ INS with second CAN interface to the host computer

Since firmware version 3.4.3.2 INS can be factory configured to have onboard second CAN BUS (this option is only available upon request). It allows outputting CAN messages through first and second CAN port simultaneously.

There are two ways to configure second CAN port settings, via INS GUI since version 2.0.50.360 from 2020-01-23 (see INS GUI User's Manual, section “10.8. Operations with CAN data” for details), and via appropriate commands (see “1.5 CAN/COM4” portion in Table 6.80).

If the CAN data output is enabled through second CAN port, then pins originally assigned to Ethernet port (see Fig. 5.3 and Table 5.1) become connected to second CAN lines as Table 5.9 shows.

In case if 24-pin multiport cable development kit (see Fig. 5.9) is being used then CAN data appears on connector “Ethernet” according to the pin diagram Table 5.10.

Table 5.9. Diagram of CAN lines of the 24-pin connector of Inertial Labs™ INS

Signal	24-pin INS connector, pin number
GROUND	21
GROUND	22
CAN2_L	23
CAN2_H	24

Table 5.10. Diagram of second CAN lines connection to “Ethernet” connector of 24-pin multiport cable development kit

Signal	RJ45 socket, pin number
GROUND	1
GROUND	2
CAN2_L	3
DNC	4
DNC	5
CAN2_H	6
DNC	7
DNC	8

Notes:

1. Ordinary communication with INS unit through COM1, COM2, COM3 ports is still available at INS data output trough second CAN port.

2. In current version second CAN port outputs data only, without receiving of outer data.

See section 6.2.18 to know which massages can be transmitted through CAN port.

5.10. Connection of OS3D-FG Stand Alone Magnetic Compass to INS

The Inertial Labs™ INS can be factory configured to use the external OS3D-FG Stand Alone Magnetic Compass (SAMC) instead of on-board magnetometers to measure components of the Earth magnetic field. In such case pins assigned to Ethernet lines (see Table 5.1) are used to connect the OS3D-FG SAMC lines as Table 5.11 shows.

Table 5.11. Diagram of SAMC connection to the Inertial Labs™ INS

4-pin OS3D-FG connector		24-pin INS connector, pin number
Pin number	Description	
1	Supply voltage	21
2	Ground	22
3	A-RS-485	23
4	B-RS-485	24

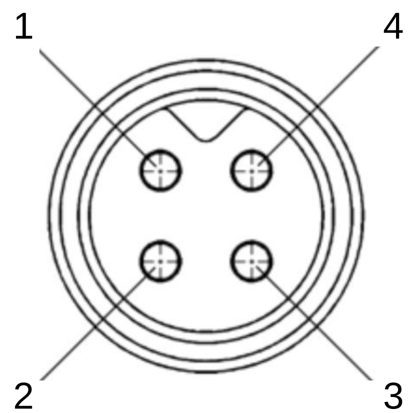


Fig. 5.14. Connector pinout of the Inertial Labs™ OS3D-FG Stand Alone Magnetic Compass with 4 pin interface (mating side of the connector)

Notes:

1. In case of using external OS3D-FG SAMC, the hard/soft iron calibration is the same as for INS magnetometers (see section 6.8).
2. Before using of external OS3D-FG SAMC, it is necessary to align the SAMC axes to INS axes (see INS GUI User's Manual, section "10.9. INS and OS3D-FG SAMC axes alignment").
3. Any change of "Use external magnetic compass" setting gets applied only after unit power off/on.

Using external magnetic compass option can be enabled or disabled, via INS GUI (see INS GUI User Manual, section "4.2.4.3 Using external magnetic compass"), and via the command 0x33 Ext_MC (see Table 6.80) since INS firmware version 3.3.0.1.

Please contact Inertial Labs to receive INS unit with OS3D-FG SAMC input support.

6. Software interface

After the Inertial Labs™ INS power on an initialisation of the onboard GNSS receiver starts that takes about 25 seconds. During this initialization the INS' LED indicator (see Fig. 5.1) lights yellow. After the initialization completed the INS' indicator lights red, and the INS works in the mode of commands waiting.

If the auto start option is enabled the INS starts operation automatically after power on (see section 6.9 for more details). The INS indicator lights green.

The commands are transmitted through the COM1 serial port according to the protocol RS232 with baud rate 115200 bps (default settings).

Table 6.1. COM-port parameters

COM-port parameters	
Baud rate	115200
Data bits	8
Parity	none
Stop bits	1

Notes:
1. Other baud rate than 115200 bps can be set for INS with firmware version since 2.2.0.0. See section "6.13. Change of the main COM port baud rate" for details.
2. The Inertial Labs™ INS with RS-422, RS-485, CAN 2.0 interfaces are also available.

All commands and messages to / from the Inertial Labs™ INS have the byte structure shown in the Table 6.2. Exception is done for the INS output in the NMEA and other text formats (see section 6.2).

Table 6.2. Byte structure for all commands and messages to/from INS

Byte number	**0**	**1**	**2**	**3**	**4, 5**	**6 .. (n+5)**	**n+6, n+7**
Parameter	Header 0	Header 1	Message type	INS data identifier	Message length	Payload	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	n bytes (variable)	1 word
Note	0xAA	0x55		In INS messages	n+6		

In the Table 6.2 and in all other tables there is denoted:

word = unsigned 2 byte integer;
sword = signed 2 byte integer.

Message type is equal to:

0 – for commands;
1 – for transferred data.

All the INS outputs are data, therefore they have Message type = 1.

INS data identifier is used in INS output data only. This byte is equal to code of the command from the host system which requested this INS message. See all commands code in “Appendix C. Full list of the Inertial Labs™ INS commands”.

Note byte #3 in the block of the initial alignment data is equal to set output data rate (see Table 6.69 and Table 6.71 in section 6.3.12). In all other messages and commands byte #3 in the Table 6.2 is zero.

Note: in INS with firmware version before 2.1.2.0 this byte #3 is zero in all messages.

The Message length is the number of bytes in the message without header. It is equal to the payload length (n) + 6.

The Check sum is the arithmetical sum of bytes 2…(n+5) (all bytes without header). In the check sum the low byte is transmitted first (see Table 6.3).

Table 6.3. Format of the check sum transmitting

byte0	byte1
low byte	high byte

Important note: The low byte is transmitted by first in all data denoted as word, sword, float.

6.1. Operational modes of the Inertial Labs™ INS

The Inertial Labs™ INS can operate in the four modes:

1. **Idle** mode. All sensors and electronics are powered. The INS microprocessor waits any command from the host computer to start operate in one of the next modes. In the idle mode the indicator of INS lights red.

2. **Continuous** operating mode. In this mode the INS operates in the endless loop, providing the continuous output of calculated position, orientation and other data according to chosen output data format (see section 6.2). Data rate is set by user from 1 Hz to 200 Hz. In the Continuous operating mode indicator of the INS lights green.

3. “**On Request**” (pooling) operating mode. It is close to the Continuous operating mode, but the INS sends only one data block after each Request command issued from host computer. In this mode indicator of the INS lights green.

4. **Calibration** operating mode. In this mode the embedded calibration procedure is performed for compensation of hard and soft iron effects of the carrier object. See section 6.8 for more details.

5. **Firmware update** operating mode. This mode allows updating the firmware of the INS. To enter the firmware update mode set the “break” condition on COM1 before the power is applied to the INS. After the device was powered on, keep the “break” condition for at least 250ms and then clear. The device is ready for the firmware update. In this mode, the LED indicator is off. Use the software provided by Inertial Labs for entering the firmware update mode and updating the firmware. For details, please see INS-MRU Firmware Update Guide.

6.2. Output Data Formats of the Inertial Labs™ INS in the Operating Modes

The next output data formats are available in the “Continuous” and “On Request” operating modes:

- INS OPVT;
- INS QPVT;
- INS OPVT2A;
- INS OPVT2AW;
- INS OPVT2AHR;
- INS OPVTAD;
- INS Sensors Data;
- INS Minimal Data;
- INS OPVT & Raw IMU Data;
- SPAN rawimu;
- INS OPVT GNSSext;
- User Defined Data;
- INS NMEA Output;
- INS Sensors NMEA Output;
- Cobham UAV 200_Satcom;
- GNSS receiver NMEA data (through COM2 port);
- GNSS receiver GPRMC data (through COM3 port);
- CAN messages (only through special CAN port);
- IMU TGA Data (only through COM4 port, continuous output);
- HEHDT Data (only through COM4 port, continuous output);
- NAV440.

Note output data formats INS OPVT2A, INS OPVT2AW, INS OPVT2AHR, INS OPVTAD are created for INS-D to output orientation data calculated by dual antenna GNSS receiver. But all these data formats can be used for any INS model – INS-B, INS-P, INS-D, absent data will be replaced by zeros.

Note: Any model of INS can be factory configured to contain an internal datalogger to record data formats listed above within operation, but only upon request by a customer (see INS GUI User Manual, rev.2.22 and newer, Appendix I. Using the Internal Datalogger).

6.2.1. The “INS OPVT” (Orientation, Position, Velocity, Time) data format

This is default data format. It provides the INS output in the form of:

- 3 orientation angles (heading, pitch and roll);
- calibrated outputs of the 9 sensors (gyros, accelerometers, magnetometers) that give information about current angular rate, linear acceleration of the INS and components of outer magnetic field;
- IMU service data;
- position – latitude, longitude, altitude;
- east, north and vertical velocity;
- GNSS position and velocity data;
- GPS reference time;
- GPS service data;
- calibrated data from the pressure sensor – pressure and barometric altitude.

Structure of the INS data blocks at the “INS OPVT” data format corresponds to the Table 6.2 with payload shown in the Table 6.4.

Maximum data rate for the INS output at the “INS OPVT” data format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.4. The INS message payload at the “INS OPVT” data format

Byte number	0 – 1	2 – 3	4 – 5	6 – 11	12 – 17	18 – 23	24 – 25	26 – 27	28 – 29
Parameter	Heading	Pitch	Roll	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	2 byte word	2 byte sword	2 byte sword	3× 2 byte sword	3× 2 byte sword	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Angular rates, deg/s *KG	Accele-rations g*KA	Magne-tic fields, nT/10		Supply voltage, VDC* 100	Temperature, ºC*10

Table 6.4 (continued)

Byte number	30 – 33	34 – 37	38 – 41	42 – 45	46 – 49	50 – 53
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	m/s*100	m/s*100

Table 6.4 (continued)

Byte number	54 – 57	58 – 61	62 – 65	66 – 69	70 – 71	72 – 75
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	deg*100	m/s*100

Table 6.4 (continued)

Byte number	76-79	80	81	82	83	84	85-86	87-90	91
Parameter	ms_gps	GNSS_info1	GNSS_info2	#solnSVs	Latency ms_pos	Latency ms_vel	P_bar	H_bar	New GPS
Length	4 byte	1 byte	1 byte	1 byte	1 byte	1 byte	2 byte word	4 byte integer	1 byte
Note	ms				ms	ms	Pa/2	m*100	

Notes:
1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Values of KG, KA are scale factors depending on gyro and accelerometer range (see Table 6.5).
3. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
4. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
5. g = 9.8106 m/s^2.
6. USW is unit status word (see section 6.10 for details).
7. Vinp is input voltage of the INS.
8. Temper is averaged temperature in 3 gyros.
9. ms_gps are milliseconds from the beginning of the GPS reference week;
10. GNSS_info1, GNSS_info2 contain info about GNSS data (Table 6.6, Table 6.7).
11. #SolnSVs is number of satellites used in navigation solution.
12. Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver's position and velocity logs relative to INS time (ms_gps), in milliseconds.
13. P_bar, H_bar – pressure and barometric height.
14. New_GPS is indicator of new update of GPS data (see Table 6.8).
15. The low byte is transmitted by first.

Table 6.5. Values of KG, KA factors for gyro and accelerometer scaled data

Gyro range, deg/sec	450	950
KG	50	20

Accelerometer range, g	8	15	40
KA	4000	2000	500

Table 6.6. GNSS_info1 – information about GNSS data

Bit	Value and Description
0 – 3	Position type: 0 – Single point position; 1 – DGPS (pseudorange differential solution); 2 – Solution calculated using corrections from SBAS; 3 – PPP solution; 4 – RTK (other) solution; 5 – RTK (narrow-int) solution; 6 – Other; 7 – Converging TerraStar-C, TerraStar-C PRO or TerraStar-X solution; 8 – Converged TerraStar-C, TerraStar-C PRO or TerraStar-X solution; 9 – Converging TerraStar-L solution; 10 – Converged TerraStar-L solution.
4 – 7	Pseudorange iono correction: 0 – unknown or default Klobuchar model; 1 – Klobuchar Broadcast; 2 – SBAS Broadcast; 3 – Multi-frequency Computed; 4 – DGPS (pseudorange differential correction); 5 – NovAtel Blended Iono Value.

Table 6.7. GNSS_info2 – information about GNSS data

Bit	Value and Description
0 – 1	Solution status: 0 – GNSS solution is computed; 1 – insufficient observations; 2 – not yet converged from cold start; 3 – other reason of absent solution.
2 – 3	GPS reference time status: 0 – time validity is unknown; 1 – time is coarse set and is being steered; 2 – position is lost and the range bias cannot be calculated; 3 – time is fine set and is being steered.
4	1 – GPS GNSS signal is used
5	1 – GLONASS GNSS signal is used
6	1 – Galileo GNSS signal is used
7	1 – BeidDou GNSS signal is used

Table 6.8. New_GPS indicator of new update of GNSS data

Bit	Value and Description
0	1 – GNSS position data update
1	1 – GNSS velocity data update
2	1 – GNSS heading data update
3	1 – valid 1PPS signal received
4	1 – GNSS BESTXYZ log update
5	1 – GNSS PSRDOP log update

Note: INS onboard GNSS receiver generates 1PPS signal all the time, but the New_GPS indicator sets bit #3 to 1 only if GPS reference time status is fine (see also Table 6.7).

6.2.2. The “INS QPVT” (Quaternion of orientation, Position, Velocity, Time) data format

This data format is near the same as the “INS OPVT” format but provides the quaternion of orientation instead of orientation angles. See “Appendix D. Forms of the Inertial Labs™ INS orientation presentation” for correct relationship between orientation angles and quaternion.

The “INS QPVT” format provides output in the form of:

- Quaternion of orientation;
- calibrated outputs of the 9 sensors (gyros, accelerometers, magnetometers) that give information about current angular rate, linear acceleration of the INS and components of outer magnetic field;
- IMU service data;
- position – latitude, longitude, altitude;
- east, north and vertical velocity;
- GNSS position and velocity data;
- GPS reference time;
- GPS service data;
- calibrated data from the pressure sensor – pressure and barometric altitude.

Structure of the INS data blocks at the “INS QPVT” data format corresponds to the Table 6.2 with payload shown in the Table 6.9.

Maximum data rate for the INS output at the “INS QPVT” data format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.9. The INS message payload at the “INS QPVT” data format

Byte number	0-7	8 – 13	14 – 19	20 – 25	26 – 27	28 – 29	30-31
Parameter	q0,q1,q2,q3	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	4× 2 byte sword	3× 2 byte sword	3× 2 byte sword	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Quaternion of orientation *10000	Angular rates, deg/s*KG	Accele-rations g*KA	Magnetic fields, nT/10		Supply voltage, VDC*100	Temperature, °C*10

Table 6.9 (continued)

Byte number	32 – 35	36 – 39	40 – 43	44 – 47	48 – 51	52 – 55
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	m/s*100	m/s*100

Table 6.9 (continued)

Byte number	56 – 59	60 – 63	64 – 67	68 – 71	72 – 73	74 – 77
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	deg*100	m/s*100

Table 6.9 (continued)

Byte number	78-81	82	83	84	85	86	87-88	89-92	93
Parameter	ms_gps	GNSS info1	GNSS info2	#SolnSVs	Latency ms_pos	Latency ms_vel	P_bar	H_bar	New GPS
Length	4 byte	1 byte	1 byte	1 byte	1 byte	1 byte	2 byte word	4 byte integer	1 byte
Note	ms				ms	ms	Pa/2	m*100	

Notes:

1. The “INS QPVT” data format is implemented in INS with firmware since version 2.1.2.0.
2. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
3.Values of KG, KA are scale factors depending on gyro and accelerometer range (see Table 6.5).
4. g = 9.8106 m/s^2.
5. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
6. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
7. USW is unit status word (see section 6.10 for details).
8. Vinp is input voltage of the INS.
9. Temper is averaged temperature in 3 gyros.
10. ms_gps are milliseconds from the beginning of the GPS reference week;
11. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7);
12. Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver’s position and velocity logs relative to INS time (ms_gps), in milliseconds.
13. P_bar, H_bar – pressure and barometric height.

14. New_GPS is indicator of new update of GPS data (see Table 6.8).
15. The low byte is transmitted by first.

6.2.3. The "INS OPVT2A" (Orientation, Position, Velocity, Time, Dual-antenna receiver data) format

The "INS OPVT2A" data format is implemented in INS with firmware since version 2.2.1.7. This data format is based on the "INS OPVT" format but provides additional data calculated by dual antenna GNSS receiver:

- 3 orientation angles (heading, pitch and roll) calculated by INS;
- calibrated outputs of the 9 sensors (gyros, accelerometers, magnetometers) that give information about current angular rate, linear acceleration of the INS and components of outer magnetic field;
- IMU service data;
- position – latitude, longitude, altitude;
- east, north and vertical velocity;
- GNSS position and velocity data;
- GPS reference time;
- GPS orientation data;
- GPS service data;
- calibrated data from the pressure sensor – pressure and barometric altitude.

Structure of the INS data blocks at the "INS OPVT2A" data format corresponds to the Table 6.2 with payload shown in the Table 6.10.

<u>Maximum data rate</u> for the INS output at the "INS OPVT2A" data format is limited to 90 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.10. The INS message payload at the "INS OPVT2A" data format

Byte number	0 – 1	2 – 3	4 – 5	6 – 11	12 – 17	18 – 23	24 – 25	26 – 27	28 – 29
Parameter	Heading	Pitch	Roll	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	2 byte word	2 byte sword	2 byte sword	3× 2 byte sword	3× 2 byte sword	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Angular rates, deg/s *KG	Accele-rations g*KA	Magnetic fields, nT/10		Supply voltage, VDC* 100	Temperature, ºC*10

Table 6.10 (continued)

Byte number	30 – 33	34 – 37	38 – 41	42 – 45	46 – 49	50 – 53
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	m/s*100	m/s*100

Table 6.10 (continued)

Byte number	54 – 57	58 – 61	62 – 65	66 – 69	70 – 71	72 – 75
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	deg*100	m/s*100

Table 6.10 (continued)

Byte number	76-79	80	81	82	83-84	85	86-87
Parameter	ms_gps	GNSS_info1	GNSS_info2	#solnSVs	V_latency	Angles position type	Heading GNSS
Length	4 byte	1 byte	1 byte	1 byte	2 byte word	1 byte	2 byte word
Note	ms				s*1000		deg*100

Table 6.10 (continued)

Byte number	88-89	90-91	92-93	94-95	96-99	100
Parameter	Latency ms_head	Latency ms_pos	Latency ms_vel	P_bar	H_bar	New GPS
Length	2 byte sword	2 byte sword	2 byte sword	2 byte word	4 byte integer	1 byte
Note	ms			Pa/2	M*100	

Notes:
1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Values of KG, KA are scale factors depending on gyro and accelerometer range (see Table 6.5):
3. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
4. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
5. g = 9.8106 m/s^2.
6. USW is unit status word (see section 6.10 for details).
7. Vinp is input voltage of the INS.
8. Temper is averaged temperature in 3 gyros.

9. ms_gps are milliseconds from the beginning of the GPS reference week;
10. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7);
11. #SolnSVs is number of satellites used in navigation solution;
12. V_latency is latency in the velocity time tag in milliseconds;
13. Latency ms_head, Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver's heading, position and velocity logs relative to INS time (ms_gps), in milliseconds.
14. Angles position type is GNSS position type at orientation calculation using dual GNSS antennas. If this value is less than 100 then see its description in the Table 6.17. Otherwise subtract 100 from the "Angles position type" and refer again to Table 6.17. See section 6.15 for explanation.
15. P_bar, H_bar – pressure and barometric height;
16. New_GPS is indicator of new update of GPS data (see Table 6.8)
17. The low byte is transmitted by first.

6.2.4. The "INS OPVT2AW" (Orientation, Position, Velocity, Time, Dual-antenna receiver data, GPS Week) format

The "INS OPVT2AW" data format is implemented in INS with firmware since version 2.5.0.5. This data format is based on the "INS OPVT2A" format but also provides the GPS Week number.

Structure of the INS data blocks at the "INS OPVT2AW" data format corresponds to the Table 6.2 with payload shown in the Table 6.11.

Maximum data rate for the INS output at the "INS OPVT2AW" data format is limited to 90 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.11. The INS message payload at the "INS OPVT2AW" data format

Byte number	0 – 1	2 – 3	4 – 5	6 – 11	12 – 17	18 – 23	24 – 25	26 – 27	28 – 29
Parameter	Heading	Pitch	Roll	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	2 byte word	2 byte sword	2 byte sword	3× 2 byte sword	3× 2 byte sword	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Angular rates, deg/s *KG	Accele-rations g*KA	Magnetic fields, nT/10		Supply voltage, VDC* 100	Temperature, ºC*10

Table 6.11 (continued)

Byte number	30 – 33	34 – 37	38 – 41	42 – 45	46 – 49	50 – 53
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	m/s*100	m/s*100

Table 6.11 (continued)

Byte number	54 – 57	58 – 61	62 – 65	66 – 69	70 – 71	72 – 75
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m/s*100	deg*100	m/s*100

Table 6.11 (continued)

Byte number	76-79	80-81	82	83	84	85	86	87
Parameter	ms_gps	GPS week	GNSS info1	GNSS info2	#solnSVs	Latency ms_pos	Latency ms_vel	Angles position type
Length	4 byte	2 byte word	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte
Note	ms					ms	ms	

Table 6.11 (continued)

Byte number	88-89	90-91	92-93	94-95	96-97	98-101	102
Parameter	Heading GNSS	Pitch GNSS	Heading STD GNSS	Pitch STD GNSS	P_bar	H_bar	New GPS
Length	2 byte word	2 byte sword	2 byte word	2 byte word	2 byte word	4 byte integer	1 byte
Note	Orientation angles, deg*100		STD, deg*100		Pa/2	m*100	

<u>Notes</u>:

1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
3. Values of KG, KA are scale factors depending on gyro and accelerometer range (see Table 6.5).
4. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
5. g = 9.8106 m/s^2.
6. USW is unit status word (see section 6.10 for details).
7. Vinp is input voltage of the INS.

8. Temper is averaged temperature in 3 gyros.
9. ms_gps are milliseconds from the beginning of the GPS reference week;
10. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7);
11. #SolnSVs is number of satellites used in navigation solution;
12. Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver's position and velocity logs relative to INS time (ms_gps), in milliseconds;
13. Angles position type is GNSS position type at orientation calculation using dual GNSS antennas. If this value is less than 100 then see its description in the Table 6.17. Otherwise subtract 100 from the "Angles position type" and refer again to Table 6.17. See section 6.15 for explanation;
14. P_bar, H_bar – pressure and barometric height.
15. New_GPS is indicator of new update of GPS data (see Table 6.8).
16. The low byte is transmitted by first.

6.2.5. The "INS OPVT2AHR" (Orientation, Position, Velocity, Time, Dual-antenna receiver data, with high resolution) format

The "INS OPVT2AHR" data format is implemented in INS with firmware since version 2.3.0.5. This data format provides the same data as the "INS OPVT2A" format, but sensors and position data are presented with higher resolution:

- 3 orientation angles (heading, pitch and roll) calculated by INS;
- calibrated outputs of 3 gyros and 3 accelerometers with high resolution;
- calibrated outputs of 3 magnetometers;
- IMU service data;
- position – latitude, longitude, altitude, with high resolution;
- east, north and vertical velocity;
- GNSS position (with high resolution) and velocity data;
- GPS reference time;
- GPS orientation data;
- GPS service data;
- calibrated data from the pressure sensor – pressure and barometric altitude.

Structure of the INS data blocks at the "INS OPVT2AHR" data format corresponds to the Table 6.2 with payload shown in the Table 6.12.

Maximum data rate for the INS output at the "INS OPVT2AHR" data format is limited to 70 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.12. The INS message payload at the "INS OPVT2AHR" data format

Byte number	0 – 1	2 – 3	4 – 5	6 – 17	18 – 29	30 – 35	37 – 37	38 – 39	40 – 41
Parameter	Heading	Pitch	Roll	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	2 byte word	2 byte sword	2 byte sword	3× 4 byte integer	3× 4 byte integer	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Angular rates, deg/s *1.0e5	Accele-rations, g*1.0e6	Magnetic fields, nT/10		Supply voltage, VDC* 100	Temperature, ºC*10

Table 6.12 (continued)

Byte number	42 – 49	50 – 57	58 – 61	62 – 65	66 – 69	70 – 73
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	8 byte integer	8 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg*1.0e9	deg*1.0e9	m*1000	m/s*100	m/s*100	m/s*100

Table 6.12 (continued)

Byte number	74 – 81	82 – 89	90 – 93	94 – 97	98 – 99	100 – 103
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	8 byte integer	8 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg*1.0e9	deg*1.0e9	m*1000	m/s*100	deg*100	m/s*100

Table 6.12 (continued)

Byte number	104-107	108	109	110	111-112	113	114-115
Parameter	ms_gps	GNSS_info1	GNSS_info2	#solnSVs	V_latency	Angles position type	Heading GNSS
Length	4 byte	1 byte	1 byte	1 byte	2 byte word	1 byte	2 byte word
Note	ms				s*1000		deg*100

Table 6.12 (continued)

Byte number	116-117	118-119	120-121	122-123	124-127	128
Parameter	Latency ms_head	Latency ms_pos	Latency ms_vel	P_bar	H_bar	New GPS
Length	2 byte sword	2 byte sword	2 byte sword	2 byte word	4 byte integer	1 byte
Note	ms			Pa/2	m*100	

Notes:

1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).

2. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
3. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
4. g = 9.8106 m/s^2.
5. USW is unit status word (see section 6.10 for details).
6. Vinp is input voltage of the INS.
7. Temper is averaged temperature in 3 gyros.
8. ms_gps are milliseconds from the beginning of the GPS reference week;
9. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7);
10. #SolnSVs is number of satellites used in navigation solution;
11. V_latency is latency in the velocity time tag in milliseconds;
12. Latency ms_head, Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver's heading, position and velocity logs relative to INS time (ms_gps), in milliseconds;
13. Angles position type is GNSS position type at orientation calculation using dual GNSS antennas. If this value is less than 100 then see its description in the Table 6.17. Otherwise subtract 100 from the "Angles position type" and refer again to Table 6.17. See section 6.15 for explanation;
14. P_bar, H_bar – pressure and barometric height;
15. New_GPS is indicator of new update of GPS data (see Table 6.8);
16. The low byte is transmitted by first.

6.2.6. The "INS OPVTAD" output data format with external aiding data

To control receiving of the external aiding data the special output data format "INS OPVTAD" is implemented since INS firmware version 2.8.2.0. The "INS OPVTAD" data format provides the same data as the "INS OPVT2AHR" format, but includes data from the external sensors.

External aiding data structure is described in section "6.6.4. Aiding data input through the main COM port".

Structure of the INS data blocks at the "INS OPVTAD" data format corresponds to the Table 6.2 with payload shown in the Table 6.13.

Maximum data rate for the INS output at the "INS OPVTAD" data format is limited to 50 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.13. The INS message payload at the "INS OPVTAD" data format

Byte number	0 – 1	2 – 3	4 – 5	6-17	18 – 29	30 – 35	36-37	38-39	40-41
Parameter	Heading	Pitch	Roll	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	USW	Vinp	Temper
Length	2 byte word	2 byte sword	2 byte sword	3× 4 byte integer	3× 4 byte integer	3× 2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Angular rates, deg/s *1.0e5	Accele-rations g*1.0e6	Magnetic fields, nT/10		Supply voltage, VDC* 1000	Temperature, ºC*10

Table 6.13 (continued)

Byte number	42-49	50-57	58-61	62-65	66-69	70-73
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	8 byte integer	8 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e9	deg *1.0e9	m*1000	m/s*100	m/s*100	m/s*100

Table 6.13 (continued)

Byte number	74 – 81	82 – 89	90 – 93	94 – 97	98 – 99	100 – 103
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	8 byte integer	8 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e9	deg *1.0e9	m*1000	m/s*100	deg*100	m/s*100

Table 6.13 (continued)

Byte number	104 – 107	108	109	110	111	112	113
Parameter	ms_gps	GNSS_info 1	GNSS_info 2	#solnSVs	Latency ms_pos	Latency ms_vel	Angles position type
Length	4 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte
Note	ms				ms	Ms	

Table 6.13 (continued)

Byte number	114 – 115	116 – 117	118 – 119	120 – 123	124
Parameter	Heading GNSS	Latency ms_head	P_bar	H_bar	New GPS
Length	2 byte word	2 byte sword	2 byte word	4 byte integer	1 byte
Note	Orientation angles, deg*100	ms	Pa/2	m*100	

Table 6.13 (continued)

Byte number	125-128	129-130	131-132	133-134	135-136	137-138
Parameter	Odometer	Air speed	North wind	East wind	North wind STD	East wind STD
Length	4 byte integer	2 byte sword	2 byte sword	2 byte sword	2 byte word	2 byte word
Note	Distance, m*1000	Speed, kt*100	North wind, kt*100	East wind, kt*100	STD, kt*100	

Table 6.13 (continued)

Byte number	139-142	143-146	147-150	151-152	153-154	155-156	157-158
Parameter	Latitude external	Longitude external	Altitude external	Latitude external STD	Longitude external STD	Altitude external STD	External position latency
Length	4 byte integer	4 byte integer	4 byte integer	2 byte word	2 byte word	2 byte word	2 byte word
Note	deg *1.0e7		m*1000	m *100			sec*1000

Table 6.13 (continued)

Byte number	159-162	163-166	167-170	171-172	173-174	175-176
Parameter	Locator latitude	Locator longitude	Locator altitude	Doppler shift	Doppler shift STD	New aiding data
Length	4 byte integer	4 byte integer	4 byte integer	2 byte sword	2 byte word	2 byte word
Note	deg *1.0e7	deg *1.0e7	m*1000	(m/s) *100		

Notes:

1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
3. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis).
4. g = 9.8106 m/s^2.
5. USW is unit status word.
6. Vinp is input voltage of the INS.
7. Temper is averaged temperature in 3 gyros.
8. ms_gps are milliseconds from the beginning of the GPS reference week;
9. GNSS_info1, GNSS_info2 contain information about GNSS data;
10. #SolnSVs is number of satellites used in navigation solution;
11. Latency ms_head, Latency ms_pos and Latency ms_vel are latencies of time stamps in GNSS receiver's heading, position and velocity logs relative to INS time (ms_gps), in milliseconds;
12. Angles position type is GNSS position type at orientation calculation using dual GNSS antennas. If this value is less than 100 then see its description in the Table 6.17.

Otherwise subtract 100 from the "Angles position type" and refer again to Table 6.17. See section 6.15 for explanation;
13. P_bar, H_bar – pressure and barometric height;
14. New_GPS is an indicator of new update of GPS data (see Table 6.8);
15. New aiding data is an indicator of update of external sensors data (see Table 6.14). For more details about aiding data see section "6.6.4. Aiding data input through the main COM port";
16. The low byte is transmitted by first.

Table 6.14. New aiding data indicator

	Bit	Parameter	Description
Low byte	0	Odometer	0 – data absent 1 – data updated
	1	Air speed	0 – data absent 1 – data updated
	2	Wind data	0 – data absent 1 – data updated
	3	External position	0 – data absent 1 – data updated
	4	Doppler shift from locator	0 – data absent 1 – data updated
	5	Heading external	0 – data absent 1 – data updated
	6	DVL data	0 – data absent 1 – data updated
	7	Sensors bias external	0 – data absent 1 – data updated
High byte	8	Pitch and Roll external	0 – data absent 1 – data updated
	9 to 15	Reserved	

6.2.7. The "INS Sensors Data" format

This data format contains data from the devices inside INS:

AHRS (IMU) data:

- 3 orientation angles (heading, pitch and roll);
- raw data from the 9 sensors (gyros, accelerometers, magnetometers) in original ADC codes;
- IMU service data;

GNSS receiver data:

- position – latitude, longitude, height;
- standard deviations of latitude, longitude and height;
- horizontal and vertical speed;
- direction of motion (track over ground);

- GPS reference time;
- GPS service data;

Pressure sensor data:

- temperature and pressure raw data in ADC codes.

Structure of the INS data blocks at the "INS Sensors Data" format corresponds to the Table 6.2 with payload shown in the Table 6.15.

Maximum data rate for the INS output at the "INS Sensors Data" format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.15. The message payload at the "INS Sensors Data" format

Byte number	0 – 1	2 – 3	4 – 5	6 – 23	24 – 25	26 – 27	28 – 29	30 – 31	32 – 33
Parameter	Heading (AHRS)	Pitch (AHRS)	Roll (AHRS)	Ugyro, Uacc, Umag	Reser-ved	Reser-ved	USW	Vinp/Vdd	Utermo
Length	2 byte word	2 byte sword	2 byte sword	9×2 byte sword	2 byte sword	2 byte sword	2 byte word	2 byte word	2 byte sword
Note	Orientation angles, deg*100			Raw sensor data (gyros, accelero-meters, magne-tometers)				Combined voltage	Temperature in each sensor

Table 6.15 (continued)

Byte number	34 – 37	38 – 41	42 – 45	46-47	48 – 49	50 – 51	52-55	56-57	58-61
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Latitude STD	Longitude STD	Altitude STD	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Length	4 byte integer	4 byte integer	4 byte integer	2 byte word	2 byte word	2 byte word	4 byte integer	2 byte word	4 byte integer
Note	deg *1.0e7	deg *1.0e7	m*100	m*1000	m*1000	m*1000	m/s*100	deg*100	m/s*100

Table 6.15 (continued)

Byte number	62 – 65	66	67	68	69	70	71	72
Parameter	ms_gps	TS_gps	sol_stat	pos_type	#SVs	#SolnSVs	#SolnL1SVs	#SolnMultiSVs
Length	4 byte integer	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte
Note	ms							

Table 6.15 (continued)

Byte number	73	74	75	76	77	78-79	80-81	82	83
Parameter	ext_sol_stat	Galileo and BeiDou	GPS and GLONASS	Latency ms_pos	Latency ms_vel	UP	UT	New GPS	Reserved
Length	1 byte	1 byte	1 byte	1 byte	1 byte	2 byte word	2 byte word	1 byte	1 byte
Note				ms	ms				

Notes:

1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
3. USW is unit status word (see section 6.10 for details).
4. The following data are recorded in the field «Vinp/Vdd» sequentially:
   – the INS input voltage, Vinp, VDC*100;
   – stabilized voltage supplied to the INS sensors, Vdd, VDC*1000;
5. In the «Utermo» field ADC codes are recorded sequentially from 7 temperature sensors inside gyros, accelerometers and magnetometers.
6. ms_gps are milliseconds from the beginning of the GPS reference week;
7. TS_gps is time status which indicates the quality of the GPS reference time (see Table 6.7);
8. sol_stat is GNSS solution status (see Table 6.16);
9. pos_type is GNSS position type (see Table 6.17);
10. #SVs is number of satellites tracked;
11. #SolnSVs is number of satellites used in navigation solution;
12. #SolnL1SVs is number of satellites with L1/E1/B1 signals used in solution;
13. #SolnMultiSVs is number of satellites with multi-frequency signals used in solution;
14. ext_sol_stat is GNSS extended solution status (see Table 6.18);
15. GPS and GLONASS is GPS and GLONASS signal-used mask (see Table 6.19);
16. Galileo and BeiDou is Galileo and BeiDou signal-used mask (see Table 6.20);
17. Latency_ms_pos and Latency_ms_vel are latencies of time stamps in GNSS receiver's position and velocity logs relative to INS time (ms_gps), in milliseconds;
18. UP and UT are raw data from the pressure sensor – pressure and temperature;
19. New_GPS is indicator of new update of GPS data (see Table 6.8);
20. The low byte is transmitted by first.

Table 6.16. sol_stat – GNSS solution status

Value	Description
0	Solution computed
1	Insufficient observations
2	No convergence
3	Singularity at parameters matrix
4	Covariance trace exceeds maximum (trace > 1000 m)
5	Test distance exceeded (maximum of 3 rejections if distance >10 km)
6	Not yet converged from cold start
7	Height or velocity limits exceeded (in accordance with export licensing restrictions)
8	Variance exceeds limits
9	Residuals are too large
13	Large residuals make position unreliable
18	When a FIX POSITION command is entered, the receiver computes its own position and determines if the fixed position is valid
19	The fixed position, entered using the FIX POSITION command, is not valid
20	Position type is unauthorized - HP or XP on a receiver not authorized

Table 6.17. pos_type – GNSS position or velocity type

Value	Description
0	No solution
8	Velocity computed using instantaneous Doppler
16	Single point position
17	Pseudorange differential solution
18	Solution calculated using corrections from an WAAS
19	Propagated by a Kalman filter without new observations
20	OmniSTAR VBS position (1)
32	Floating L1 ambiguity solution
33	Floating ionospheric-free ambiguity solution
34	Floating narrow-lane ambiguity solution
48	Integer L1 ambiguity solution
50	Integer narrow-lane ambiguity solution
64	OmniSTAR HP position (1)
65	OmniSTAR XP or G2 position (1)
68	Converging TerraStar-C, TerraStar-C PRO or TerraStar-X solution (2)
69	Converged TerraStar-C, TerraStar-C PRO or TerraStar-X solution (2)
77	Converging TerraStar-L solution (2)
78	Converged TerraStar-L solution (2)

Notes:

1. A subscription for OmniSTAR or use of a DGPS service is required. It is not realized in the Inertial Labs™ INS firmware yet.

2. PPP solution requires access to a suitable correction stream, delivered either through L-Band or the internet. For L-Band delivered TerraStar or Veripos service, appropriate receiver software model is required, along with a subscription to the desired service.

Table 6.18. ext_sol_stat – GNSS extended solution status

Bit	Mask	Description
0	0x01	If an RTK solution: NovAtel CORRECT solution has been verified If a PDP solution: solution is GLIDE Otherwise: Reserved
1-3	0x0E	Pseudorange Iono Correction 0 = Unknown or default Klobuchar model 1 = Klobuchar Broadcast 2 = SBAS Broadcast 3 = Multi-frequency Computed 4 = PSRDiff Correction 5 = NovAtel Blended Iono Value
4	0x10	Reserved
5	0x20	0 = No antenna warning 1 = Antenna information is missing
6-7	0xC0	Reserved

Table 6.19. GPS and GLONASS signal-used mask

Bit	Mask	Description
0	0x01	GPS L1 used in solution
1	0x02	GPS L2 used in solution
2	0x04	GPS L5 used in solution
3	0x08	Reserved
4	0x10	GLONASS L1 used in solution
5	0x20	GLONASS L2 used in solution
6-7	0x40-0x80	Reserved

Table 6.20. Galileo and BeiDou signal-used mask

Bit	Mask	Description
0	0x01	Galileo E1 used in solution
1-3	0x02-0x08	Reserved
4	0x10	BeiDou B1 used in solution
5	0x20	BeiDou B2 used in solution
6-7	0x40-0x80	Reserved

6.2.8. The “INS Minimal Data” format

This data format specifies the minimum of the INS data that can be transferred with larger data rate:

- 3 orientation angles (heading, pitch and roll);
- IMU service data;
- position – latitude, longitude, altitude;
- east, north and vertical velocity;
- GPS reference time;
- GPS service data.

Structure of the INS data blocks at the “INS Minimal Data” format corresponds to the Table 6.2 with payload shown in the Table 6.21.

Maximum data rate for the INS output at the “INS Minimal Data” format is 200 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.21. The message payload at the “INS Minimal Data” format

Byte number	0 – 1	2 – 3	4 – 5	6 – 7	8 – 9	10 – 11	12 – 15	16 – 19	20 – 23
Parameter	Heading	Pitch	Roll	USW	Vinp	Temper	Latitude	Longitude	Altitude
Length	2 byte word	2 byte sword	2 byte sword	2 byte word	2 byte word	2 byte sword	4 byte integer	4 byte integer	4 byte integer
Note	Orientation angles, deg*100				Supply voltage, VDC* 100	Temperature, ºC*10	deg *1.0e7	deg *1.0e7	m*100

Table 6.21 (continued)

Byte number	24 – 27	28 – 31	32 – 35	36-39	40	41
Parameter	East speed	North speed	Vertical speed	ms_gps	GNSS_info1	#SolnSVs
Length	4 byte integer	4 byte integer	4 byte integer	4 byte integer	1 byte	1 byte
Note	m/s*100	m/s*100	m/s*100			

Notes:
1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. USW is unit status word (see section 6.10 for details);
3. Vinp is input voltage of the INS;
4. Temper is averaged temperature in 3 gyros;
5. ms_gps are milliseconds from the beginning of the GPS reference week;
6. GNSS_info1 contains information about GNSS data (see Table 6.6);
7. #SolnSVs is number of satellites used in navigation solution;
8. The low byte is transmitted by first.

6.2.9. The “INS OPVT & Raw IMU Data” (Orientation, Position, Velocity, Time and raw IMU) format

The “INS OPVT & Raw IMU Data” format provides the most accurate information about the raw IMU data (measurements from the gyros and accelerometers) including their precision time stamps, and is intended for post-processing of INS data, and other tasks where accurate information from gyros and accelerometers is necessary. Also this data format contains the main INS data – orientation, position and velocity.

This data format is implemented in INS firmware since version 3.2.1.2.

Structure of the INS data blocks at the “INS OPVT & Raw IMU Data” format corresponds to the Table 6.2 with payload shown in the Table 6.22.

Maximum data rate for the INS output at the “INS OPVT & Raw IMU Data” format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.22. The message payload at the “INS OPVT & Raw IMU Data” format

Byte number	0-7	8-15	16-27	28 – 39	40-41	42-45	46-49	50-53
Parameter	GPS INS Time	GPS IMU Time	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	USW	Heading	Pitch	Roll
Length	8 byte unsigned integer	8 byte unsigned integer	3× 4 byte integer	3× 4 byte integer	2 byte word	4 byte unsigned integer	4 byte integer	4 byte integer
Note	s*1.0e9	s*1.0e9	Angular rates, deg/s ∗10000	Accele-rations, m/s^2 ∗10000		Orientation angles, deg*1000		

Table 6.22 (continued)

Byte number	54-61	62-69	70-73	74-77	78-81	82-85
Parameter	Latitude	Longitude	Altitude	East speed	North speed	Vertical speed
Length	8 byte integer	8 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg *1.0e9	deg *1.0e9	m*1000	m/s*100	m/s*100	m/s*100

Table 6.22 (continued)

Byte number	86	87	88	89
Parameter	GNSS_info 1	GNSS_info 2	#solnSVs	New GPS
Length	1 byte	1 byte	1 byte	1 byte
Note				

Notes:

1. GPS INS Time is time of INS navigation solution (orientation, position, velocity data), in nanoseconds from the beginning of the GPS reference week;

2. GPS IMU Time is time of the last IMU data package received (measurements from the gyros and accelerometers), in nanoseconds from the beginning of the GPS reference week;

3. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).

4. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).

5. USW is unit status word (see section 6.10 for details);

6. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7);

7. #SolnSVs is number of satellites used in navigation solution;

8. New_GPS is indicator of new update of GPS data (see Table 6.8);
9. The low byte is transmitted by first.

6.2.10. The “SPAN rawimu” data format

This is copy of the NovAtel SPAN rawimub data log that is widely used to output the raw IMU data (measurements from the gyros and accelerometers) and their precision time stamps, see document
https://www.novatel.com/assets/Documents/Manuals/OM-20000144UM.pdf
In contrast to the “INS OPVT & Raw IMU Data” format, the “SPAN rawimu” data do not contain information about orientation, position, velocity. The “SPAN rawimu” data format is implemented in INS firmware since version 3.2.1.8.

Note: scale factors for raw gyros and accelerometers data, IMU status description were changed in INS firmware since version 3.2.3.8.

The Table 6.23 shows full structure of the INS data blocks at the “SPAN rawimu” Note this data format has the NovAtel binary data structure and does not corresponds to the Table 6.2.

Maximum data rate for the INS output at the “SPAN rawimu” data format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.23. The “SPAN rawimu” message structure

Byte number	0	1	2	3	4-5	6	7	8-9
Parameter	Sync1	Sync2	Sync3	Header Length	Message ID	Message Type	Port Address	Message Length
Length	1 byte	1 byte	1 byte	1 byte	2 byte word	1 byte	1 byte	2 byte word
Note	0xAA	0x44	0x12	28	268	0x00		40

Table 6.23 (continued)

Byte number	10-11	12	13	14-15	16-19	20-23	24-25	26-27
Parameter	Sequence	Idle Time	Time Status	Week	GPS Time	Receiver Status	Reserved	Receiver S/W Version
Length	2 byte word	1 byte	1 byte	2 byte word	4 byte unsigned integer	4 byte unsigned integer	2 byte word	2 byte word
Note	0x00				ms			

Table 6.23 (continued)

Byte number	28-31	32-39	40-43	44-55	56-67	68-71
Parameter	Week	GPS IMU Time	IMU Status	AccZ, -Accy, AccX	GyroZ, -GyroY, GyroX	32-bit CRC
Length	4 byte unsigned integer	8 byte double	4 byte integer	3× 4 byte integer	3× 4 byte integer	4 byte hex
Note		s		$m/s*2^{31}/200$	$deg*2^{31}/720$	

Notes:

1. Bytes 0 to 27 are header of the “SPAN rawimu” message;
2. Time Status indicates the quality of the GPS reference time (see Table 6.24);
3. Week is the GPS week number;
4. GPS Time are milliseconds from the beginning of the GPS reference week;
5. Receiver Status is 32-bits representing the status of various hardware and software components of the GNSS receiver (see Table 6.25);
6. Receiver S/W Version is a value (0 - 65535) representing the receiver software build number;
7. GPS IMU Time is time of the last IMU data package received (measurements from the gyros and accelerometers), in seconds from the beginning of the GPS reference week;
8. IMU Status – see Table 6.26;
9. AccZ, -AccY, AccX – are accelerometers output along Z, Y, X axes in form of linear velocity increments in m/s. Note -AccY has opposite sign for velocity Y increment;
10. GyroZ, -GyroY, GyroX – are gyros output around Z, Y, X axes in form of angle increments in degrees. Note -GyroY has opposite sign for angle Y increment.
11. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).

Table 6.24. GPS Reference Time Status

Value	Description
20	Time validity is unknown
60	Time is set approximately
80	Time is approaching coarse precision
100	Time is valid to coarse precision
120	Time is coarse set and is being steered
130	Position is lost and the range bias cannot be calculated
140	Time is adjusting to fine precision
160	Time has fine precision
170	Time is fine set and is being steered by the backup system
180	Time is fine set and is being steered

Table 6.25. GNSS Receiver Status

Nibble	Bit	Mask	Description	Bit = 0	Bit = 1
N0	0	0x00000001	Error flag	No error	Error
	1	0x00000002	Temperature status	Within specifications	Warning
	2	0x00000004	Voltage supply status	OK	Warning
	3	0x00000008	Antenna power status	Powered	Not powered
N1	4	0x00000010	LNA Failure	OK	Failure
	5	0x00000020	Antenna open flag	OK	Open
	6	0x00000040	Antenna shorted flag	OK	Shorted
	7	0x00000080	CPU overload flag	No overload	Overload
N2	8	0x00000100	COM1 buffer overrun flag	No overrun	Overrun
	9	0x00000200	COM2 buffer overrun flag	No overrun	Overrun
	10	0x00000400	COM3 buffer overrun flag	No overrun	Overrun
	11	0x00000800	Link overrun flag	No overrun	Overrun
N3	12	0x00001000	Reserved		
	13	0x00002000	Aux transmit overrun flag	No overrun	Overrun
	14	0x00004000	AGC out of range	OK	Out of range
	15	0x00008000	Reserved		
N4	16	0x00010000	INS Reset	No Reset	INS reset
	17	0x00020000	Reserved		
	18	0x00040000	Almanac flag/UTC known	Valid	Invalid
	19	0x00080000	Position solution flag	Valid	Invalid
N5	20	0x00100000	Position fixed flag	Not fixed	Fixed
	21	0x00200000	Clock steering status	Enabled	Disabled
	22	0x00400000	Clock model flag	Valid	Invalid
	23	0x00800000	External oscillator locked flag	Unlocked	Locked
N6	24	0x01000000	Software resource	OK	Warning
	25	0x02000000	Reserved		
	26	0x04000000	Reserved		
	27	0x08000000	Reserved		
N7	28	0x10000000	Reserved		
	29	0x20000000	Auxiliary 3 status event flag	No event	Event
	30	0x40000000	Auxiliary 2 status event flag	No event	Event
	31	0x80000000	Auxiliary 1 status event flag	No event	Event

Table 6.26. IMU Status

Nibble	Bit	Mask	Description	Range Value
N0	0	0x00000001	Alarm Status Flag	
	1	0x00000002	Reserved	
	2	0x00000004	Reserved	
	3	0x00000008	SPI Communication Error	0 = Passed, 1 = Failed
N1	4	0x00000010	Sensor Over-Range	0 = Passed, 1 = One or more sensors over-ranged
	5	0x00000020	Initial Self Test Failure	0 = Passed, 1 = Failed
	6	0x00000040	Flash Memory Failure	0 = Passed, 1 = Failed
	7	0x00000080	Processing Overrun	0 = Passed, 1 = Failed
N2	8	0x00000100	Self Test Failure – X-axis gyro	0 = Passed, 1 = Failed
	9	0x00000200	Self Test Failure – Y-axis gyro	0 = Passed, 1 = Failed
	10	0x00000400	Self Test Failure – Z-axis gyro	0 = Passed, 1 = Failed
	11	0x00000800	Self Test Failure – X-axis accelerometer	0 = Passed, 1 = Failed
N3	12	0x00001000	Self Test Failure – Y-axis accelerometer	0 = Passed, 1 = Failed
	13	0x00002000	Self Test Failure – Z-axis accelerometer	0 = Passed, 1 = Failed
	14	0x00004000	Reserved	
	15	0x00008000	Reserved	
N4	16	0x00010000	IMU temperature reading as follows: Signed 2-byte value (SHORT) 25ºC = 0x0000 1 LSB = 0.00565ºC	
	17	0x00020000		
	18	0x00040000		
	19	0x00080000		
N5	20	0x00100000		
	21	0x00200000		
	22	0x00400000		
	23	0x00800000		
N6	24	0x01000000		
	25	0x02000000		
	26	0x04000000		
	27	0x08000000		
N7	28	0x10000000		
	29	0x20000000		
	30	0x40000000		
	31	0x80000000		

6.2.11. The "INS OPVT GNSSext" output data format with extended GNSS data

The "INS OPVT GNSSext" data format is implemented in INS with firmware since version 3.2.3.5. This data format contains high resolution orientation, position and velocity data like other INS output data formats but also it included extended information about time and the GNSS data:

- 3 orientation angles (heading, pitch and roll) calculated by INS, with high resolution;
- calibrated outputs of 3 gyros and 3 accelerometers with high resolution;
- calibrated outputs of 3 magnetometers (if they are installed in INS unit);
- IMU service data (temperature, USW);
- position – latitude, longitude and altitude calculated by INS, with high resolution;
- east, north and vertical velocity calculated by INS, with high resolution;
- GNSS position as latitude, longitude, altitude and in ECEF coordinates (all with high resolution) and their accuracy estimates;
- GNSS velocity data as horizontal and vertical speed and ECEF velocities and their accuracy estimates;
- GNSS heading (for dual-antenna GNSS receiver);
- accurate GPS time information in three forms: time of GPS week, UTC, and time represented as year, month, day, hours, minutes, seconds, decimal seconds;
- GPS service data including GDOP, PDOP, HDOP, VDOP, TDOP;
- calibrated data from the pressure sensor.

Structure of the INS data blocks at the "INS OPVT GNSSext" data format corresponds to the Table 6.2 with payload shown in the Table 6.27.

Maximum data rate for the INS output at the "INS OPVT GNSSext" data format is limited to 50 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.27. The INS message payload at the "INS OPVT GNSSext" data format

Byte number	0 – 3	4 – 7	8 – 11	12 – 15	16 – 23	24 – 31	32 – 35
Parameter	TOW	Heading	Pitch	Roll	Latitude	Longitude	Altitude
Data format	4 byte unsigned integer	4 byte unsigned integer	byte intege	4 byte integer	8 byte integer	8 byte integer	4 byte integer
Note	sec*10^3	deg*10^6			deg*10^9	deg*10^9	m*1000

Table 6.27 (continued)

Byte number	36-39	40-43	44-47	48 – 59	60 – 71	72 – 77	78-79	80-81	82-83
Parameter	East speed	North speed	Vertical speed	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	MagX, MagY, MagZ	P_bar	Temper	USW
Data format	4 byte integer	4 byte integer	4 byte integer	3× 4 byte integer	3× 4 byte integer	3× 2 byte sword	2 byte word	2 byte sword	2 byte word
Note	m/s*10^6			deg/s*10^6	g*10^6	nT/10	Pa/2	ºC*10^2	

Table 6.27 (continued)

Byte number	84	85 – 88	89 – 92	93 – 96	97- 98	99 – 102	103–106	107–110	111-112
Parameter	Pos_type	GNSS ECEF X	GNSS ECEF Y	GNSS ECEF Z	GNSS_PACC	GNSS ECEF VX	GNSS ECEF VY	GNSS ECEF VZ	GNSS_SACC
Data format	1 byte	4 byte integer	4 byte integer	4 byte integer	2 byte word	4 byte integer	4 byte integer	4 byte integer	2 byte word
Note		m*10^2			m*10^2	m/s*10^6			m/s*10^2

Table 6.27 (continued)

Byte number	113-120	121-128	129-132	133-136	137-140	141-144
Parameter	Latitude GNSS	Longitude GNSS	Altitude GNSS	Horizontal speed GNSS	Track over ground GNSS	Vertical speed GNSS
Data format	8 byte integer	8 byte integer	4 byte integer	4 byte integer	4 byte integer	4 byte integer
Note	deg*10^9	deg*10^9	m*1000	m/s*10^6	deg*10^6	m/s*10^6

Table 6.27 (continued)

Byte number	145	146-147	148	149	150	151-152	153-154	155-156	157-158
Parameter	Angles_ pos_type	GNSS Heading	#solnSVs	GNSS_info1	GNSS_info2	GDOP	PDOP	HDOP	VDOP
Data format	1 byte	2 byte word	1 byte	1 byte	1 byte	2 byte word	2 byte word	2 byte word	2 byte word
Note		deg*10^2				*10^3	*10^3	*10^3	*10^3

Table 6.27 (continued)

Byte number	159-160	161-162	163	164	165	166-167	168	169	170-171
Parameter	TDOP	Diff_age	GPS Hours	GPS Minutes	GPS Seconds	GPS Decimal Seconds	Month	Day	Year
Data format	2 byte word	2 byte word	1 byte	1 byte	1 byte	2 byte word	1 byte	1 byte	2 byte word
Note	*10^3	sec*10							

Table 6.27 (continued)

Byte number	172-179	180	181-182	183	184	185
Parameter	UTC	Latency ECEF_time	V_latency	Latency pos_time	Latency vel_time	New GPS
Data format	8 byte integer	1 byte	2 byte sword	1 byte	1 byte	1 byte
Note		msec	msec	msec	msec	

Notes:

1. TOW (time of week) are seconds from the beginning of the GPS reference week.
2. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
3. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
4. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
5. P_bar is pressure;
6. Temper is averaged temperature in 3 gyros.
7. USW is unit status word (see section 6.10 for details).
8. pos_type is GNSS position type (see Table 6.17);
9. GNSS_PACC is 3D position accuracy estimate (1 sigma).
10. GNSS_SACC is speed accuracy estimate (1 sigma).
11. #solnSVs is number of satellites used in navigation solution.
12. GNSS_info1, GNSS_info2 contain information about GNSS data (see Table 6.6, Table 6.7).
13. Diff_age is age of differential correction.
14. UTC is the absolute number of seconds since midnight, January 1, 1970, including leap seconds. A value of zero indicates that UTC is not available.
15. "GPS Decimal Seconds" are milliseconds part of the GPS time.
16. Latency ECEF_time, Latency pos_time, Latency vel_time – are latencies of time stamps in received BESTXYZ, BESTPOS, BESTVEL logs relative to INS time (TOW) synchronized to the receiver PPS.
17. V_latency – is the latency in the velocity time tag.
18. New_GPS is indicator of new update of GPS data (see Table 6.8).
19. The low byte is transmitted by first.

6.2.12. The "User Defined Data" output format

Since firmware version 3.2.4.3 user can choose desirable INS data for output using "User Defined Data" output format. Table 6.29 shows full list of data that user can choose, these data are combined in several groups. Table 6.30 gives detailed description of each data type.

Structure of the INS data blocks at the "User Defined Data" output format corresponds to the Table 6.2 with payload shown in the Table 6.28.

Table 6.28. Payload of the "User Defined Data"

Field	Offset in payload, bytes	Size, bytes	Value
Pckg Number	0	1	Number of data packages present in the payload (N)
Data List	1	N	List of data types, one per byte, refer to Table 6.29 for values
Data #1	N+1	Variable, depends on data type, refer to Table 6.30 for sizes	Data according to structure shown in Table 6.30
Data #2	Variable, depends on Data #1 size	Variable, depends on data type, refer to Table 6.30 for sizes	Data according to structure shown in Table 6.30
…	…	…	…
Data #N	Variable, depends on preceding data	Variable, depends on data type, refer to Table 6.30 for sizes	Data according to structure shown in Table 6.30

Table 6.29. Groups of "User Defined Data"

Data group	Data type	Description
Time data	0x01	GPS INS Time (round)
	0x02	GPS INS Time
	0x03	GPS IMU Time
	0x04	UTC
	0x05 – 0x06	Reserved
Orientation data	0x07	Orientation angles
	0x08	Orientation angles HR
	0x09	Quaternion of orientation
	0x0A – 0x0F	Reserved
Navigation data	0x10	Position
	0x11	Position HR
	0x12	Velocities
	0x13 – 0x19	Data destined for other device
	0x1A – 0x1F	Reserved
Sensors data	0x20	Gyro data
	0x21	Gyro data HR
	0x22	Accelerometer data
	0x23	Accelerometer data HR
	0x24	Magnetometer data
	0x25	Barometer data
	0x26	Sensors bias
	0x27	Accelerometer data in PV measuring point
	0x28 –0x2F	Reserved
GNSS data	0x30	GNSS Position
	0x31	GNSS Position HR
	0x32	GNSS Velocity, Track over ground
	0x33	GNSS Heading, GNSS Pitch
	0x34	GNSS position STD
	0x35	GNSS Heading STD,GNSS Pitch STD
	0x36	GNSS info short

	0x37	Full satellites info
	0x38	GNSS Solution status
	0x39	GNSS Position or Velocity type
	0x3A	GNSS Angles position type
	0x3B	Number of satellites used in solution
	0x3C	GPS week
	0x3D	GNSS Velocity Latency
	0x3E	GNSS Position timestamp
	0x3F	GNSS Velocity timestamp
	0x40	GNSS Heading timestamp
	0x41	New GPS
	0x42	Dilution of precision
	0x43	GNSS Position and Speed accuracy
	0x44	GDOP, PDOP
	0x45	GNSS Track over ground
	0x46	Reserved
	0x47	GNSS diff age
	0x48	GNSS ECEF Velocity STD
	0x49	PPPSEED status
	0x4A – 0x4F	Reserved
INS service data	0x50	Supply voltage
	0x51	Stabilized voltage
	0x52	Temperature
	0x53	Unit status word (USW)
	0x54	INS solution status (see section 6.15)
	0x55	KF position covariance
	0x56	KF heading covariance
	0x57	KF position covariance HR
	0x58	KF velocity covariance
	0x59 – 0x5F	Reserved
External aiding data	0x60	Odometer
	0x61	Air speed
	0x62	Wind data
	0x63	External position
	0x64	Doppler shift from locator
	0x65	New aiding data
	0x66	Heading external
	0x67	DVL (Doppler Velocity Log) data
	0x68	Sensors bias external
	0x69	Pitch and Roll external
	0x6A – 0x6F	Reserved

Table 6.30. Detailed description of “User Defined Data” structure

Data type	Description	Size, bytes	Structure	*Note*
0x00	Reserved	TBD	TBD	
0x01	GPS INS Time (round)	4	unsigned integer	Time of INS solution, milliseconds from the beginning of the GPS reference week, rounded to 1000/(output data rate)
0x02	GPS INS Time	8	unsigned integer	Time of INS solution, seconds*1.0e9 from the beginning of the GPS reference week
0x03	GPS IMU Time	8	unsigned integer	Time of INS IMU data, seconds*1.0e9 from the beginning of the GPS reference week
0x04	UTC	9	byte	UTC Hours
			byte	UTC Minutes
			byte	UTC Seconds
			word	UTC Decimal Seconds
			byte	UTC Month
			byte	UTC Day
			word	UTC Year
0x05 – 0x06	Reserved	TBD	TBD	
0x07	Orientation angles	6	word	Heading, deg*100
			sword	Pitch, deg*100
			sword	Roll, deg*100
0x08	Orientation angles HR	12	unsigned integer	Heading, deg*1000
			integer	Pitch, deg*1000
			integer	Roll, deg*1000
0x09	Quaternion of orientation	8	sword	q0*10000
			sword	q1*10000
			sword	q2*10000
			sword	q3*10000
0x0A – 0x0F	Reserved	TBD	TBD	
0x10	Position	12	integer	Latitude, deg*1.0e7
			integer	Longitude, deg*1.0e7
			integer	Altitude, m*100
0x11	Position HR	20	integer (8 byte)	Latitude, deg*1.0e9
			integer (8 byte)	Longitude, deg*1.0e9
			integer	Altitude, m*1000
0x12	Velocities	12	integer	East speed, m/s*100
			integer	North speed, m/s*100
			integer	Vertical speed, m/s*100
0x13 – 0x19	Reserved			Data destined for other device
0x1A – 0x1F	Reserved	TBD	TBD	
0x20	Gyro data	6	sword	Gyro X, deg/s*KG
			sword	Gyro Y, deg/s*KG
			sword	Gyro Z, deg/s*KG
0x21	Gyro data HR	12	integer	Gyro X, deg/s*1.0e5

			integer	Gyro Y, deg/s*1.0e5
			integer	Gyro Z, deg/s*1.0e5
0x22	Accelerometer data	6	sword	Accelerometer X, g*KA
			sword	Accelerometer Y, g*KA
			sword	Accelerometer Z, g*KA
0x23	Accelerometer data HR	12	integer	Accelerometer X, g*1.0e6
			integer	Accelerometer Y, g*1.0e6
			integer	Accelerometer Z, g*1.0e6
0x24	Magnetometer data	6	sword	Magnetometer X, nT/10
			sword	Magnetometer Y, nT/10
			sword	Magnetometer Z, nT/10
0x25	Barometer data	6	word	Pressure, Pa/2
			integer	Baro altitude, m*100
0x26	Sensors bias	7	signed byte	Gyro bias X, deg/s*0.5*10^4
			signed byte	Gyro bias Y, deg/s*0.5*10^4
			signed byte	Gyro bias Z, deg/s*0.5*10^4
			signed byte	Accel bias X, g*0.5*10^5
			signed byte	Accel bias Y, g*0.5*10^5
			signed byte	Accel bias Z, g*0.5*10^5
			byte	Reserved
0x27	Accelerometer data in PV measuring point (see note #14)	12	integer	Accelerometer X, m/s2*1.0e5
			integer	Accelerometer Y, m/s2*1.0e5
			integer	Accelerometer Z, m/s2*1.0e5
0x28 –0x2F	Reserved	TBD	TBD	
0x30	GNSS Position	12	integer	GNSS Latitude, deg*1.0e7
			integer	GNSS Longitude, deg*1.0e7
			integer	GNSS Altitude, m*100
0x31	GNSS Position HR	20	integer (8 byte)	GNSS Latitude, deg*1.0e9
			integer (8 byte)	GNSS Longitude, deg*1.0e9
			integer	GNSS Altitude, m*1000
0x32	GNSS Velocity, Track over ground	10	integer	GNSS Horizontal speed, m/s*100
			word	GNSS Track over ground, deg*100
			integer	GNSS Vertical speed, m/s*100
0x33	GNSS Heading, GNSS Pitch	4	word	GNSS Heading, deg*100
			sword	GNSS Pitch, deg*100
0x34	GNSS position STD	6	word	GNSS Latitude STD, m*1000
			word	GNSS Longitude STD, m*1000
			word	GNSS Height STD, m*1000
0x35	GNSS Heading STD, GNSS Pitch STD	4	word	GNSS Heading STD, deg*100
			word	GNSS Pitch STD, deg*100
0x36	GNSS info short	2	byte	GNSS_info1 (Position type, pseudorange iono correction – see Table 6.6)

			byte	GNSS_info2 (solution status, time status, GNSS constellations in use – see Table 6.7)
0x37	Full satellites info	8	byte	#SVs (Number of satellites tracked)
			byte	#SolnSVs (Number of satellites used in solution)
			byte	#SolnL1SVs (Number of satellites with L1/E1/B1 signals used in solution)
			byte	#SolnMultiSVs (Number of satellites with multi-frequency signals used in solution)
			byte	Galileo and BeiDou signal-used mask (see Table 6.20)
			byte	GPS and GLONASS signal-used mask (see Table 6.19)
			byte	GPS time status (see Table 6.24)
			byte	Extended solution status (Table 6.18)
0x38	GNSS Solution status	1	byte	See Table 6.16
0x39	GNSS Position or Velocity type	1	byte	See Table 6.17
0x3A	GNSS Angles position type	1	byte	Angles position type at calculation of GNSS angles (see Table 6.17)
0x3B	Number of satellites used in solution	1	byte	
0x3C	GPS week	2	word	
0x3D	GNSS Velocity Latency	2	word	ms
0x3E	GNSS Position timestamp	4	unsigned integer	ms
0x3F	GNSS Velocity timestamp	4	unsigned integer	ms
0x40	GNSS Heading timestamp	4	unsigned integer	ms
0x41	New GPS	1	byte	Indicator of new update of GPS data (see Table 6.8)
0x42	Dilution of precision	10	word	GDOP (Geometric dilution of precision), $*10^3$
			word	PDOP (Position dilution of precision), $*10^3$
			word	HDOP (Horizontal dilution of precision), $*10^3$
			word	VDOP (Vertical dilution of precision), $*10^3$
			word	TDOP (Time dilution of precision), $*10^3$
0x43	GNSS Position and	4	word	GNSS_PACC, m*100

	Speed accuracy		word	GNSS_SACC, m/s*100
0x44	GDOP, PDOP	4	word	GDOP (Geometric dilution of precision), *10^3
			word	PDOP (Position dilution of precision), *10^3
0x45	GNSS Track over ground	2	word	GNSS Track over ground, deg*100
0x46	Reserved			
0x47	GNSS diff age	2	word	Age of differential correction, s*10
0x48	GNSS ECEF Velocity STD	6	word	Standard deviation of GNSS ECEF velocity X, m/s*1000
			word	Standard deviation of GNSS ECEF velocity Y, m/s*1000
			word	Standard deviation of GNSS ECEF velocity Z, m/s*1000
0x49	PPPSEED status (see note #15)	2	byte	APPLICATION status
			byte	STORE status
0x4A – 0x4F	Reserved	TBD	TBD	
0x50	Supply voltage	2	word	VDC*100
0x51	Stabilized voltage	2	word	VDC*1000
0x52	Temperature	2	sword	°C*10
0x53	Unit status word	2	word	USW (see section 6.10)
0x54	INS solution status	1	byte	See section 6.15
0x55	KF position covariance	3	byte	Latitude STD, cm
			byte	Longitude STD, cm
			byte	Altitude STD, cm
0x56	KF heading covariance	1	byte	Heading STD, deg*100
0x57	KF position covariance HR	6	word	Latitude STD, mm
			word	Longitude STD, mm
			word	Altitude STD, mm
0x58	KF velocity covariance	3	byte	East speed STD, mm/s
			byte	North speed STD, mm/s
			byte	Vertical speed STD, mm/s
0x59 – 0x5F	Reserved	TBD	TBD	
0x60	Odometer	4	integer	Odometer distance, m*1000
0x61	Air speed	2	sword	Air speed, kt*100
0x62	Wind data	8	sword	North wind, kt*100
			sword	East wind, kt*100
			word	North wind STD, kt*100
			word	East wind STD, kt*100
0x63	External position	20	integer	Latitude external, deg*1.0e7
			integer	Longitude external, deg*1.0e7
			integer	Altitude external, m*1000
			word	Latitude external STD, m*100
			word	Longitude external STD, m*100

			word	Altitude external STD, m*100
			word	External position latency, sec*1000
0x64	Doppler shift from locator	16	integer	Locator latitude, deg*1.0e7
			integer	Locator longitude, deg*1.0e7
			integer	Locator altitude, m*1000
			sword	Doppler shift, m/s*100
			word	Doppler shift STD, m/s*100
0x65	New aiding data	2	word	Indicator of update of external sensors data (see Table 6.14)
0x66	Heading external	6	word	Heading external, deg*100
			word	Heading external STD, deg*100
			word	Heading external latency, sec*1000
0x67	DVL data (depends on reference frame, see note #13)	24	integer	Lateral velocity, 1000*m/sec
			integer	Forward velocity, 1000*m/sec
			integer	Vertical velocity, 1000*m/sec
			word	Lateral velocity STD, 1000*m/sec
			word	Forward velocity STD, 1000*m/sec
			word	Vertical velocity STD, 1000*m/s
			word	Velocity latency, msec
			integer	Reserved
0x68	Sensors bias external	7	signed byte	Gyro bias X, $deg/s*0.5*10^4$
			signed byte	Gyro bias Y, $deg/s*0.5*10^4$
			signed byte	Gyro bias Z, $deg/s*0.5*10^4$
			signed byte	Accel bias X, $g*0.5*10^5$
			signed byte	Accel bias Y, $g*0.5*10^5$
			signed byte	Accel bias Z, $g*0.5*10^5$
			byte	Reserved
0x69	Pitch and Roll external	4	sword	Pitch external, deg*100
			sword	Roll external, deg*100
0x6A – 0x6F	Reserved	TBD	TBD	

Notes:

1. Unit status word (USW) is highly recommended for choice to control INS unit state (see section 6.10 for details).
2. If even one data type from “GNSS data” group is chosen, then “New_GPS” data is highly recommended for choice too. This is indicator of new update of GNSS data (see Table 6.8 for details).
3. If even one data type from “External aiding data” group is chosen, then “New aiding data” is highly recommended for choice too. This is indicator of new update of aiding data (see Table 6.14 for details).
4. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
5. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).
6. "UTC Decimal Seconds" are milliseconds part of the UTC time.
7. Values of KG, KA are scale factors depending on gyro and accelerometer range (see Table 6.5).

8. g = 9.8106 m/s^2.
9. Angular rates, linear accelerations and magnetic fields are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
10. Temper is averaged temperature in 3 gyros.
11. GNSS_PACC is 3D position accuracy estimate (1 sigma).
12. GNSS_SACC is speed accuracy estimate (1 sigma).
13. DVL (Doppler Velocity Log) data can be presented in the vehicle axes (in the right, forward, vertical directions) or in navigational frame (East, North, Up) depending on "DVL reference frame" value which can be set using INS GUI since version 2.0.48.320 from 2019-04-10 (see User's Manual rev.2.16 and higher, section "4.2.4.2. Using the main COM1 port to receive external aiding data" for details), or can be set via the command 0x34 DVL_reference_frame (see Table 6.80) since INS firmware version 3.3.0.1.
14. INS can calculate accelerations for any measuring point set by its position relative to the accelerometer mass-center of the INS unit. There are two ways to configure the measuring point, via INS GUI (see INS GUI User's Manual, section "4.2.1. "IMU" tab of "Devices options" window for details), and via the command 0x09 PV_meas_point (see Table 6.80).
15. PPPSEED status logs are present only if TerraStar correction is chosen for the NovAtel GNSS receiver otherwise appropriate fields will be denoted by 0. Adjustment of the GNSS receiver`s type of correction can be done via INS GUI (see INS GUI User's Manual, section "10.2.1. GNSS correction"), and via the command 0x17 GNSS_corr_type (see Table 6.80) since INS firmware version 3.3.0.1.
16. The low byte is transmitted by first.

Use **User_Def_Data_config** command (see section 6.3.8) to choose data types for output.

Because the package length of the "User Defined Data" is variable and depends on user choice of data, then the <u>maximum data rate</u> for the INS output is also variable. It can be estimated using formula (6.2) in section 6.14, taking into account that there are available only data rates that are factors of 200: (1, 2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 200) Hz.

INS unit controls set data rate. If it exceeds the maximum value at current COM port baud rate and length of chosen data, then INS decreases set data rate to allowed value and informs user about this change in answer on the User_Def_Data_config command (see section 6.3.8).

Send **Get_User_Def_Data_struct** command to check the "User Defined Data" structure set by user last time (see section 6.3.9).

6.2.13. The “INS NMEA Output” data format

At the “INS NMEA Output” the INS data are transmitted in the form of sentences with printable ASCII characters like the NMEA 0183 format. Each sentence starts with a "$" sign and ends with <CR><LF> (carriage return 0xD and line feed 0xA symbols). All data fields are separated by commas. The general form of the “INS NMEA Output” sentence is the next

$PAPR,LLmm.mmmm,n,YYYmm.mmmm,x,AAAA.aa,B,RRRR.rr,PPP.pp,HHH.hh, tttttttt,TTT.t,VV.v,SSSS*CC<CR><LF>

where PAPR is identifier and other fields are listed below:

- **LLmm.mmmm** is unsigned latitude, where LL are degrees, mm.mmmm are minutes;
- **n** is N or S (North or South);
- **YYYmm.mmmm** is unsigned longitude, where YYY are degrees, mm.mmmm are minutes;
- **x** is E or W (East or West);
- **AAAA.aa** is altitude in meters;
- **B** denotes kind of height data
- ‘a’ – altitude.
- **RRRR.rr** is roll in degrees;
- **PPP.pp** is pitch in degrees;
- **HHH.hh** is heading in degrees;
- **tttttttt** is timestamp (milliseconds from the beginning of the GPS reference week);
- **TTT.t** is temperature inside INS (averaged value for 3 gyros);
- **VV.v** is input voltage of the INS;
- **SSSS** is unit status word, USW (see section 6.10 for details). It is hex written with ASCII;
- **CC** is check sum that consists of a "*" and two hex digits representing XOR of all characters between, but not including "$" and "*".

Note: Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).

Maximum data rate for the INS output at the “INS NMEA Output” data format is limited to 100 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

6.2.14. The “INS Sensors NMEA Output” data format

The “INS Sensors NMEA output” data have structure close to the “INS NMEA”, with addition of gyros and accelerometers data. So, at the “INS Sensors NMEA output” the INS data are transmitted in the form of sentences with printable ASCII characters like the NMEA 0183 format. Each sentence starts with a "$" sign and ends with <CR><LF> (carriage return 0xD and line feed 0xA symbols). All data fields are separated by commas. The general form of the “INS Sensors NMEA output” sentence is the next

$PAPS,LLmm.mmmm,n,YYYmm.mmmm,x,AAAA.aa,B,RRRR.rr,PPP.pp,HHH.hh, GGGG.xx,GGGG.yy,GGGG.zz,AA.xxxx,AA.yyyy,AA.zzzz,ttttttttt,TTT.t,VV.v,SSS S*CC<CR><LF>

where PAPS is identifier and other fields are listed below:

- **LLmm.mmmm** is unsigned latitude, where LL are degrees, mm.mmmm are minutes;
- **n** is N or S (North or South);
- **YYYmm.mmmm** is unsigned longitude, where YYY are degrees, mm.mmmm are minutes;
- **x** is E or W (East or West);
- **AAAA.aa** is altitude in meters;
- **B** denotes kind of height data:
- ‘a’ – altitude.
- **RRRR.rr** is roll in degrees;
- **PPP.pp** is pitch in degrees;
- **HHH.hh** is heading in degrees;
- **GGGG.xx** is gyro X data in degrees/s;
- **GGGG.yy** is gyro Y data in degrees/s;
- **GGGG.zz** is gyro Z data in degrees/s;
- **AA.xxxx** is accelerometer X data in g (g = 9.8106 m/s^2);
- **AA.yyyy** is accelerometer Y data in g;
- **AA.zzzz** is accelerometer Z data in g;
- **ttttttttt** is timestamp (milliseconds from the beginning of the GPS reference week);
- **TTT.t** is temperature inside INS in °C (averaged value for 3 gyros);
- **VV.v** is input voltage of the INS, in Volts;
- **SSSS** is unit status word, USW (see 6.10 for details). It is hex written with ASCII;
- **CC** is check sum that consists of a "*" and two hex digits representing XOR of all characters between, but not including "$" and "*".

Notes:
1. Orientation, position and velocity data can be presented in form of instant or average values (see Appendix E.1).
2. Gyros, accelerometers and magnetometers data can be presented in form of instant, average or incremental values (see Appendix E.2).

Maximum data rate for the INS output at the “INS Sensors NMEA Output” data format is limited to 70 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

6.2.15. The “Cobham UAV 200 Satcom” data format

At the “Cobham UAV 200 Satcom” the INS data are transmitted as AT-commands in the form of sentences with printable ASCII characters. Each sentence starts with a "AT_ITINS=" text and ends with <CR><LF> (carriage return 0xD and line feed 0xA symbols). All data fields are separated by commas. The general form of the “Cobham UAV 200 Satcom” sentence is the next:

AT_ITINS=<lat>,<lon>,<alt>,<height>,<utc>,<ns_vel>,<ew_vel>, <gnd_spd>,<track_ angle>,<roll>,<pitch>,<heading>,<mag_heading>, <roll_rate>,<pitch_rate>,<heading_rate><CR><LF>

Table 6.31. Structure of the “Cobham UAV 200 Satcom” sentence

Parameter	Description	Units/Format	Positive Sense	Range min	Range max
AT_ITINS=	AT-Command	–	-	-	-
<lat>	Latitude	decimal degrees	North	-90	90
<long>	Longitude	decimal degrees	East	-180	180
<alt>	Altitude (pressure)	meters	Up	-500	25000
<height>	Height	meters	Up	-500	25000
<utc>	Time	<year>.<month>.<day>-<hour>:<minute>:<second>		13-12-1901	19-01-2038
<ns_vel>	North-South velocity	m/s	North	-500	500
<ew_vel/>	East-West velocity	m/s	East	-500	500
<gnd_spd>	Ground speed	m/s		0	500
<track_angle>	Track angle - true	degrees	Clockwise from North	-180	360
<roll>	Roll angle	degrees	Right	-180	360
<pitch>	Pitch angle	degrees	Nose up	-180	360
<heading>	True Heading	degrees	Clockwise from North	-180	360
<mag_heading>	Magnetic Heading	degrees	Clockwise from North	-180	360

<roll_rate>	Roll rate	degrees/second	Right	-90	90
<pitch_rate>	Pitch rate	degrees/second	Nose up	-90	90
<heading_rate>	Yaw rate	degrees/second	Clockwise	-90	90
<CR>	Carriage return				
<LF>	Line feed				

Maximum data rate for the INS output at the "Cobham UAV 200 Satcom" data format is limited to 80 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

6.2.16. The GNSS receiver NMEA data formats (through COM2 port)

The Inertial Labs™ INS can use the second COM2 port for output the set of GNSS receiver data in NMEA format. The INS starts output of these data after power on and completing of the receiver initialization (when the INS LED indicator switches from yellow to red).

Available NMEA data sets:

- GPGGA,
- GPGSA,
- GPRMC,
- GPVTG,
- GPZDA,

and one asynchronous log:

- GPHDT.

There are two ways to choose NMEA set, via INS GUI (see INS GUI User's Manual, section "4.2.2. "GNSS receiver" tab of "Devices options…" window" for details), and via the command 0x15 NMEA_set_COM2, NMEA_COM2_data_rate (see Table 6.80) since INS firmware version 3.3.0.1.

The data for synchronous logs are generated with frequency set for each log individually (see section 6.4. Control of the GNSS receiver).

Data are transmitted in the form of sentences with printable ASCII characters like the NMEA 0183 format. Each sentence starts with a "$" sign and ends with <CR><LF> (carriage return 0xD and line feed 0xA symbols). All data fields are separated by commas. GPGGA log contains time, 3D position and fix related data of the GNSS receiver. The structure of the GPGGA log is shown in Table 6.32.

Table 6.32. The GPGGA log structure

Message component	Description
$GPGGA	Log header
utc	UTC time of position (hours/minutes/seconds/decimal seconds, hhmmss.ss)
lat	Latitude (DDmm.mm)
lat dir	Latitude direction (N = North, S = South)
lon	Longitude (DDDmm.mm)
lon dir	Longitude direction (E = East, W = West)
quality	GPS Quality Indicators (see Table 6.33)
# sats	Number of satellites in use. May be different to the number in view
hdop	Horizontal dilution of precision
alt	Antenna altitude above/below mean sea level
a-units	Units of antenna altitude (M = meters)
undulation	Undulation - the relationship between the geoid and the WGS84 ellipsoid
u-units	Units of undulation (M = meters)
age	Age of correction data (in seconds)
stn ID	Differential base station ID
***xx**	Checksum
[CR][LF]	Sentence terminator

Table 6.33. GPS Quality Indicators

Indicator	Description
0	Fix not available or invalid
1	Single point
2	Pseudorange differential
	Unconverged OmniSTAR
	HP/XP/G2/VBS converging PPP
4	RTK fixed ambiguity solution (RT2)
	Operational
5	RTK floating ambiguity solution (RT20)
	Converged OmniSTAR HP/XP/G2
	Converged PPP
6	Dead reckoning mode
7	Manual input mode (fixed position)
8	Simulator mode
9	WAAS (SBAS)

GPGSA log contains GNSS receiver operating mode, satellites used for navigation and DOP values. The structure of the GPGSA log is shown in Table 6.34.

Table 6.34. The GPGSA log structure

Message component	Description
$ GPGSA	Log header
mode MA	A = Automatic 2D/3D M = Manual, forced to operate in 2D or 3D
mode 123	Mode: 1 = Fix not available; 2 = 2D; 3 = 3D
prn (fields 4-15)	PRN numbers of satellites used in solution (null for unused fields), total of 12 fields GPS = 1 to 32 SBAS = 33 to 64 (add 87 for PRN number) GLO = 65 to 96
pdop	Position dilution of precision
hdop	Horizontal dilution of precision
vdop	Vertical dilution of precision
***xx**	Checksum
[CR][LF]	Sentence terminator

GPRMC log contains time, 2D position, track made good and speed data provided by the GNSS navigation. The structure of the GPRMC log is shown in Table 6.35.

Table 6.35. The GPRMC log structure

Message component	Description
$ GPRMC	Log header
utc	UTC time of position (hours/minutes/seconds/decimal seconds, hhmmss.ss)
pos status	Position status: A = data valid, V = data invalid
lat	Latitude (DDmm.mm)
lat dir	Latitude direction (N = North, S = South)
lon	Longitude (DDDmm.mm)
lon dir	Longitude direction (E = East, W = West)
speed Kn	Speed over ground, knots
track true	Track made good, degrees True
date	Date: dd/mm/yy
mag var	Magnetic variation, degrees
var dir	Magnetic variation direction E/W
mode ind	Positioning system mode indicator (see Table 6.37)
***xx**	Checksum
[CR][LF]	Sentence terminator

GPVTG log contains the track made good and speed relative to the ground. The structure of the GPVTG log is shown in Table 6.36.

Table 6.36. The GPVTG log structure

Message component	Description
$ GPVTG	Log header
track true	Track made good, degrees True
T	True track indicator
track mag	Track made good, degrees Magnetic; Track mag = Track true + (MAGVAR correction)
M	Magnetic track indicator
speed Kn	Speed over ground, knots
N	Nautical speed indicator (N = Knots)
speed Km	Speed, kilometers/hour
K	Speed indicator (K = km/hr)
mode ind	Positioning system mode indicator (see Table 6.37)
***xx**	Checksum
[CR][LF]	Sentence terminator

Table 6.37. NMEA Positioning System Mode Indicator

Mode	Indicator
A	Autonomous
D	Differential
E	Estimated (dead reckoning) mode
M	Manual input
N	Data not valid

GPZDA log outputs the UTC date and time. The structure of the GPZDA log is shown in Table 6.38.

Table 6.38. The GPZDA log structure

Message component	Description
$ GPZDA	Log header
utc	UTC time
day	Day, 01 to 31
month	Month, 01 to 12
year	Year
null	Local zone description—not available
7null	Local zone minutes description—not available
***xx**	Checksum
[CR][LF]	Sentence terminator

GPHDT log outputs actual carrier object heading in degrees True (from True North). The structure of the GPHDT log is shown in Table 6.39.

Table 6.39. The GPHDT log structure

Message component	Description
$ GPHDT	Log header
heading	Heading in degrees
True	Degrees True
***xx**	Checksum
[CR][LF]	Sentence terminator

6.2.17. The GNSS receiver GPRMC data format (through COM3 port)

The Inertial Labs™ INS can use the third COM3 port for output the GNSS receiver log GPRMC. There are two ways to set GPRMC message, via INS GUI (see INS GUI User's Manual, section "4.2.2 GNSS receiver tab of "Device options" window" for details), and via the command 0x16 GNSS_COM3_data_set, GNSS_COM3_data_rate, GNSS_COM3_bps (see Table 6.80) since INS firmware version 3.3.0.1.

The INS starts output of these data after power on and completing of the receiver initialization (when the INS LED indicator switches from yellow to red). The data for synchronous logs are generated with set frequency (see section 6.4. Control of the GNSS receiver).

Data are transmitted in the form of sentences with printable ASCII characters like the NMEA 0183 format. Each sentence starts with a "$" sign and ends with <CR><LF> (carriage return 0xD and line feed 0xA symbols). All data fields are separated by commas. GPRMC log contains time, position and fix related data of the GNSS receiver. See the structure of the GPRMC log in the Table 6.35.

6.2.18. The CAN messages (transmitted through CAN ports)

Since firmware version 3.2.4.0 the Inertial Labs INS can output CAN2.0A messages through CAN ports. Since firmware version 3.2.8.1 CAN2.0B messages are provided, too, and possibility of CAN messages choice is implemented. There are two ways to configure two CAN ports, via INS GUI since version 2.0.50.360 from 2020-01-23 (see INS GUI User's Manual, section "10.8. Operations with CAN data" for details), and via appropriate commands since INS firmware version 3.3.0.1 (see "1.5 CAN/COM4" portion in Table 6.80).

Default baud rate of CAN messages is 500K bits per second. It is possible to choose one of the next baud rates: 10K; 20K; 50K; 100K; 125K; 250K; 500K; 1M bits per second.

On default data rate of CAN messages is equal to rate of data output through the main COM1 port. Though low CAN output baud rate may not allow transfer high frequency INS data. The maximum data rate (Hz) can be calculated using the baud rate and data package length:

$$max\_CAN\_data\_rate = \frac{CAN\_baud\_rate}{bits\_CAN\_mes * nmb\_CAN\_mes}, \quad (6.1)$$

where CAN_baud_rate is CAN output baud rate (bits/s); bits_CAN_mes = 128 bits – is length of one CAN message; nmb_CAN_mes is number of chosen CAN messages in one package.

Note**:** before INS firmware version 3.2.5.8 the number of CAN messages in one package nmb_CAN_mes = 8.

INS unit controls correctness of the CAN data rate setting. If user sets data rate which exceeds maximum value (6.1) then data rate is decreased and CAN messages resample down. User can select type of CAN messages for output between CAN2.0A and CAN2.0B. CAN2.0A message has 11-bit identifier (see Fig. 6.1).

10	9	8	7	6	5	4	3	2	1	0
Device identifier (7 bits)							Message offset (4 bits)			

Fig. 6.1. CAN2.0A message identifier

The Device Identifier can be changed in range 0x00 to 0x7F.

If the term “Base Address” is used instead of “Device Identifier” in the Windows configuration program, then “Base Address” simply equals to the “Device Identifier” multiplied by 16. User is able to change the Device Identifier from 0x00 to 0x7F (hexadecimal). “Message offset” is unique ID of specific CAN message (see Table 6.40). CAN2.0B message has 29 bit extended identifier with structure according to SAE J1939 standard (see Fig. 6.2).

28	27	26	25	24	23	...	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Priority, EDP, DP (5 bits)					PDU format (8 bits)			Message offset (8 bits)								Device identifier (8 bits)							

Fig. 6.2. CAN2.0B message identifier

User is able to change the next parts of CAN2.0B identifier:

- **Priority_EDP_DP** – is composed part that includes Priority (3 bits), Extended Data Page (1 bit) and Data Page (1 bit). Priority_EDP_DP parameter can be changed in range 0x00 to 0x1F.
- **PDU_Format** – is PDU format that can be changed in range 0x00 to 0xFF.
- **Device identifier** – can be changed in range 0x00 to 0xFF.

In both CAN2.0A and CAN2.0B identifiers the “Message offset” is unique ID of specific CAN message according to Table 6.40.

Table 6.40. List of supported CAN messages

Message number	Message Offset	Data description
0	0x0	Angular rates
1	0x1	Accelerations
2	0x2	Magnetic field
3	0x3	Orientation
4	0x4	East velocity
5	0x5	North velocity
6	0x6	Vertical velocity
7	0x7	Longitude
8	0x8	Latitude
9	0x9	Altitude
A	0xA	Time information
B	0xB	INS solution accuracy
C	0xC	GNSS info

Note: CAN messages number 0 to 2 are provided by INS since firmware version 3.2.5.8.

User can choose desirable CAN messages for output. Use **CAN1_message_set_config** command (see section 6.3.10) to choose CAN data messages for output through CAN1. Send **Get_CAN1_message_set_struct** command to check the “CAN messages set” structure ordered by user the last time for CAN1 port (see section 6.3.11). **CAN2_message_set_config** and **Get_CAN2_message_set_struct** applied to CAN2 port for the same reasons.

Data field of each CAN message has length 8 byes. All INS data in CAN messages are scaled integers. Each CAN message contains time stamp which is GPS reference time at the time of the measurement. GPS reference time is time from the beginning of the GPS reference week, in seconds with one millisecond resolution. The message #A “Time information” contains full value of GPS reference time including seconds and milliseconds. All other messages contain only number of milliseconds in full GPS reference time.

Message #0. Angular rates

There are Gyro X, Gyro Y, Gyro Z data presented as scaled 2-bytes signed integers, in (deg/s)*KG, where scale factor KG depends on gyro range, see Table 6.5. So angular rate resolution is 1/KG deg/s.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.41. Angular rates (Message offset: 0x0)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Angular rate X		Angular rate Y		Angular rate Z		Time stamp	
MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB

Message #1. Accelerations

There are Accelerometer X, Accelerometer Y, Accelerometer Z data presented as scaled 2-bytes signed integers, in g*KA, where scale factor KA depends on accelerometer range, see Table 6.5. So acceleration resolution is 1/KA g.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.42. Accelerations (Message offset: 0x1)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Acceleration X		Acceleration Y		Acceleration Z		Time stamp	
MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB

Message #2. Magnetic field

There are Magnetometer X, Magnetometer Y, Magnetometer Z data presented as scaled 2-bytes signed integers, in nT/10. So magnetic field resolution is 10 nT.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.43. Magnetic field (Message offset: 0x2)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Magn. field X		Magn. field Y		Magn. field Z		Time stamp	
MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB

Message #3. Orientation data

Heading data is scaled 2-bytes unsigned integer, in degrees*100. So heading data resolution is 0.01 deg.

Pitch and roll data are scaled 2-bytes signed integers, in degrees*100. So pitch and roll data resolution is 0.01 deg.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.44. Orientation Data (Message offset: 0x3)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Heading		Pitch		Roll		Time stamp	
MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB

Message #4. East velocity

East velocity data is scaled 4-bytes signed integer, in (m/s)*100. So east velocity data resolution is 0.01 m/s.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.45. East velocity data (Message offset: 0x4)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
East velocity				Time stamp		–	–
MSB			LSB	MSB	LSB	–	–

Message #5. North velocity

North velocity data is scaled 4-bytes signed integer, in (m/s)*100. So north velocity data resolution is 0.01 m/s.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.46. North velocity data (Message offset: 0x5)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
North velocity				Time stamp		–	–
MSB			LSB	MSB	LSB	–	–

Message #6. Vertical velocity

Vertical velocity data is scaled 4-bytes signed integer, in (m/s)*100. So vertical velocity data resolution is 0.01 m/s.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.47. Vertical velocity data (Message offset: 0x6)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Vertical velocity				Time stamp		–	–
MSB			LSB	MSB	LSB	–	–

Message #7. Longitude

Longitude data is scaled 6-bytes signed integer, in degrees*10^9. So longitude data resolution is 10^{-9} degrees.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.48. Longitude data (Message offset: 0x7)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Longitude						Time stamp	
MSB					LSB	MSB	LSB

Message #8. Latitude

Latitude data is scaled 6-bytes signed integer, in degrees*10^9. So latitude data resolution is 10^{-9} degrees.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.49. Latitude data (Message offset: 0x8)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Latitude						Time stamp	
MSB					LSB	MSB	LSB

Message #9. Altitude

Altitude data is scaled 4-bytes signed integer, in meters*1000. So Altitude data resolution is 0.001 meter.

Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.50. Altitude data (Message offset: 0x9)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Altitude				Time stamp		–	–
MSB			LSB	MSB	LSB	–	–

Message #A. Time information

This message contains GPS reference time at the time of the measurement. GPS reference time is time from the beginning of the GPS reference week, in seconds with one millisecond resolution.

The first part of this message contains integer part of GPS reference time in seconds. The second part contains fractional part of GPS reference time as number of millisecond. So full GPS reference time = “GPS time (integer part)” + “GPS time (milliseconds)” / 1000

Table 6.51. Time information (Message offset: 0xA)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
GPS time (integer part)				GPS time (milliseconds)		–	–
MSB			LSB	MSB	LSB	–	–

Message #B. INS solution accuracy

This message contains next data:

- INS position covariance (3D position accuracy estimate) is scaled 2-bytes <u>unsigned</u> integer, in m*1000. So data resolution of INS position covariance is 0.001 m;
- INS velocity covariance (velocity accuracy estimate) is scaled 2-bytes <u>unsigned</u> integer, in m/s*1000. So data resolution of INS velocity covariance is 0.001 m/s;
- INS heading covariance (heading accuracy estimate) is scaled 1-byte integer, in deg*100. So data resolution of INS heading covariance is 0.01 deg;
- INS solution status (see Table 6.97);

- Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.52. INS solution accuracy data (Message offset: 0xB)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
INS position covariance		INS velocity covariance		INS heading covariance	INS solution status	Time stamp	
MSB	LSB	MSB	LSB			MSB	LSB

Message #C. GNSS info

This message contains next data:

- #SolnSVs - number of satellites used in solution;
- GNSS position type (see Table 6.17);
- GPS time status (see Table 6.24);
- GNSS solution status (see Table 6.16);
- Time stamp is number of milliseconds in GPS reference time at which the measurements were done.

Table 6.53. GNSS info data (Message offset: 0xC)

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
#SolnSVs	GNSS position type	GPS time status	GNSS solution status	Time stamp		-	-
				MSB	LSB	-	-

6.2.19. The “IMU TGA” output data (transmitted through COM4 port)

Since firmware version 3.2.6.1 the Inertial Labs INS can output IMU data with their maximum frequency through COM4 port. There are two ways to configure COM4 port, via INS GUI since version 2.0.46.279 from 2018-07-26 (see INS GUI User’s Manual, section “4.2.5. “CAN / COM4” tab of “Devices options” window” for details), and via appropriate commands since INS firmware version 3.3.0.1 (see “1.5 CAN/COM4” portion in Table 6.80).

Note the baud rate for COM4 port should be set to not less than the next values to avoid skips of IMU TGA data:

- 460800 or 921600 bps – for 800 Hz IMU data rate;
- 2000000 bps – for 2000 Hz IMU data rate.

Structure of the INS data blocks at the “IMU TGA” (IMU Time, Gyros, Accelerometers) output data format corresponds to the Table 6.2 with payload shown in the Table 6.54.

Table 6.54. The INS message payload at the “IMU TGA” output data format

Byte number	0 – 11	12 – 23	24 –31
Parameter	GyroX, GyroY, GyroZ	AccX, AccY, AccZ	GPS IMU Time
Length	3× 4 byte integer	3× 4 byte integer	8 byte unsigned integer
Note	Angular rates, deg/s*1.0e5	Accelerations, g*1.0e6	GPS seconds of the week, s*1.0e9

Notes:

1. Angular rates and linear accelerations are in the carrier object axes (X is lateral axis, Y is longitudinal axis, Z is vertical axis). The INS orientation relative to the carrier object axes is set by alignment angles (see Appendix B. Variants of the Inertial Labs™ INS mounting relative to the object axes).
2. Angular rates and linear accelerations are instantaneous, not filtered IMU data.
3. g = 9.8106 m/s^2.
4. GPS IMU Time are nanoseconds from the beginning of the GPS reference week (see section 6.16 for details);
5. The low byte is transmitted by first.

6.2.20. The “HEHDT” data format (transmitted through COM4 port)

Since firmware version 3.2.6.1 the Inertial Labs INS can output messages in NMEA HEHDT format through COM4 port. There are two ways to configure COM4 port, via INS GUI since version 2.0.47.311 from 2019-02-27 (see INS GUI User’s Manual, section “4.2.5. “CAN / COM4” tab of “Devices options” window” for details), and via appropriate commands since INS firmware version 3.3.0.1 (see “1.5 CAN/COM4” portion in Table 6.80).

The general forms of the “HEHDT” sentences are the next:

$HEHDT,XXX.XX,T*CC<CR><LF>

Table 6.55. The NMEA HEHDT message in “HEHDT” format

Message component	Description
$	Start character
HEHDT	identifier
XXX.XX	heading in degrees
T	true heading indicator
CC	check sum that consists of a "*" and two hex digits representing XOR of all characters between, but not including "$" and "*"
<CR>	Carriage return
<LF>	Line feed

6.2.21. The “NAV440” data format

Since firmware version 3.4.0.9 the Inertial Labs INS can output the next data in “NAV440” data format:

- Roll angle;
- Pitch angle;
- Yaw angle (true north);
- X angular rate corrected;
- Y angular rate corrected;
- Z angular rate corrected;
- North velocity;
- East velocity;
- Down velocity;
- GPS Longitude;
- GPS Latitude;
- GPS altitude;
- GPS ITOW;
- Master BIT and Status.

The “NAV440” format consists of a fixed-length binary message with structure shown in the Table 6.23. Note this data format has specific data structure and does not corresponds to the Table 6.2.

Maximum data rate for the output is 200 Hz at standard COM-port baud rate 115200 bps. See Table 6.96 for maximum data rate at other baud rates.

Table 6.56. The “NAV440” message structure

Byte number	0-1	2-3	4	5-6	7-8	9-10
Parameter	Preamble	Packet Type	Length	rollAngle	pitchAngle	yawAngleTrue
Length	2 byte	2 byte	1 byte	2 byte sword	2 byte sword	2 byte sword
Note	0x5555	0x4E30	0x20	2*pi/2^16, radians [360°/2^16, deg]	2*pi/2^16, Radians [360°/2^16, deg]	2*pi/2^16, Radians [360°/2^16, deg]

Table 6.56 (continued)

Byte number	11-12	13-14	15-16	17-18	19-20	21-22
Parameter	xRateCorrected	yRateCorrected	zRateCorrected	nVel	eVel	dVel
Length	2 byte sword	2 byte sword	2 byte sword	2 byte sword	2 byte sword	2 byte sword
Note	7*pi/2^16, rad/s [1260°/2^16, deg/s]	7*pi/2^16, rad/s [1260°/2^16, deg/s]	7*pi/2^16, rad/s [1260°/2^16, deg/s]	512/2^16, m/s	512/2^16, m/s	512/2^16, m/s

Table 6.56 (continued)

Byte number	23-26	27-30	31-32	33-34	35-36	37-38
Parameter	longitudeGPS	latitudeGPS	altitudeGPS	GPSITOW	BITstatus	Check sum
Length	4 byte integer	4 byte integer	2 byte sword	2 byte word	2 byte word	2 byte word
Note	2*pi/2^32, Radians [360°/2^32, deg]	2*pi/2^32, Radians [360°/2^32, deg]	(2^14/2^16) – 100, m	lower 2 bytes of GPS TOW, ms		CRC-16 check sum

Notes:

1. rollAngle, pitchAngle, yawAngleTrue – are orientation angles: roll, pitch, yaw (true north), in radians. Angles: scaled to a range of [-pi, +pi] or [-180 deg to +180 deg].
2. xRateCorrected, yRateCorrected, zRateCorrected – are corrected angular rate around X, Y, Z axes, in rad/s;
3. nVel, eVel, dVel – are North, East and Down velocity respectively, in m/s;
4. longitudeGPS, latitudeGPS, altitudeGPS – are GPS Longitude in radians, GPS Latitude in radians and GPS altitude in meters (range is from -100 to 16284 meters);
5. GPSITOW – GPS Integer Time of Week (lower 2 bytes of GPS Time), in milliseconds;
6. BITstatus – Master BIT and Status, see Table 6.57;
7. All multiple byte values are transmitted Big Endian (Most Significant Byte First);
8. All communication packets end with a single word CRC (2 bytes). CRC‟s are calculated on all packet bytes excluding the preamble and CRC itself. Input packets with incorrect CRC’s will be ignored. This 16-bit CRC-CCITT standard is maintained by the International Telecommunication Union (ITU). The highlights are:
   - Width = 16 bits;
   - Polynomial 0x1021;
   - Initial value = 0xFFFF;
   - No XOR performed on the final value.

Table 6.57. NAV440 Default BIT Status Definitions

BITstatus Field	Bits	Meaning	Category
masterFail (1)	0	0 = normal, 1 = fatal error has occurred	BIT
HardwareError (2)	1	0 = normal, 1= internal hardware error	BIT
comError	2	0 = normal, 1 = communication error	BIT
softwareError (3)	3	0 = normal, 1 = internal software error	BIT
Reserved	4:7	N/A	
masterStatus (4)	8	0 = nominal, 1 = one or more status alert	Status
hardwareStatus	9	0 = nominal, 1 = Internal GPS unlocked or 1PPS invalid	Status
comStatus	10	Disabled	Status
softwareStatus	11	0 = nominal, 1 = algorithm Initialization	Status
sensorStatus (5)	12	0 = nominal, 1 = sensor over range	Status
Reserved	13:15	N/A	

Notes:

1. masterFail equals 1 when there is an error with one of the bits in BIT Category, otherwise value equals 0;
2. HardwareError equals 1 in case when there is an error with: Software status, Gyroscope Unit, Accelerometer Unit, Magnetometer Unit, Electronics, GNSS receiver (see Table 6.93), otherwise value equals 0 if there are no errors;
3. softwareError equals 1 in case when Initial Alignment failed, otherwise value equals 0 when there are no errors;
4. masterStatus equals 1 when there is an error with one of the bits in Status Category, otherwise value equals 0;
5. sensorStatus equals 1 in case when there is an error with: Incorrect Power Supply, Angular Rate Exceeding Detect, Large Magnetic Field Detect, Environmental Temperature, otherwise value equals 0 when there are no errors.

6.3. Control of the Inertial Labs™ INS

After power connection an initialisation of the onboard GNSS receiver starts that takes about 15 seconds. During this initialization the INS' LED indicator (see Fig. 5.1) lights yellow. After the initialization completed the LED indicator switches to red, and the INS goes to the idle mode in which it is ready to receive commands from the host computer.

When the INS switches from idle to any operation mode, the light indicator changes its color from red to green.

The next commands are used to control the INS unit:

- INS_OPVTdata;
- INS_QPVTdata;
- INS_OPVT2Adata;
- INS_OPVT2AHRdata;
- INS_OPVT2AWdata;
- INS_OPVTADdata
- INS_SensorsData;
- INS_minData;
- INS_OPVT_rawIMUdata;
- SPAN_rawimu;
- INS_OPVT_GNSSextdata;
- User_Def_Data;
- INS_NMEA;
- INS_Sensors_NMEA;
- Cobham_UAV200_Satcom;
- INS_NAV440;
- SetOnRequestMode;
- Stop;
- ReadINSpar;
- LoadINSpar;
- GetDevInfo;
- GetBIT;
- DevSelfTest.

All these commands have the byte structure shown in the Table 6.2. Payload for all commands has length 1 byte and contains code of the command. See Appendix C for exact structure of these commands.

6.3.1. Commands for INS start in continuous operating mode

Commands, listed in Table 6.58, are used to start the Inertial Labs™ INS in the "Continuous" operating mode with appropriate variant of output data format.

Table 6.58. INS control command and appropriate output data format

Command	Code	Output data format	
		Name	Structure
INS_SensorsData	0x50	INS Sensors Data	Inertial Labs binary structure according to Table 6.2
INS_OPVTdata	0x52	INS OPVT	
INS_QPVTdata	0x56	INS QPVT	
INS_OPVT2Adata	0x57	INS OPVT2A	
INS_OPVT2AHRdata	0x58	INS OPVT2AHR	
INS_OPVT2AWdata	0x59	INS OPVT2AW	
INS_OPVTADdata	0x61	INS OPVTAD	
INS_minData	0x53	INS Minimal Data	
INS_OPVT_rawIMUdata	0x66	INS OPVT & Raw IMU	
INS_OPVT_GNSSextdata	0x67	INS OPVT GNSSext	
User_Def_Data	0x95	User Defined Data	
INS_NMEA	0x54	INS NMEA Output	ASCII sentences
INS_Sensors_NMEA	0x55	INS and Sensors NMEA Output	
Cobham_UAV200_Satcom	0x46	Cobham UAV 200 Satcom	
SPAN_rawimu	0x68	SPAN rawimu	Unique binary structure
INS_NAV440	0x6A	NAV440	

All these commands have the byte structure shown in the Table 6.2. Payload for all commands has length 1 byte and contains code of the command listed in the Table 6.58.

INS acknowledges receiving any of these commands by responding back immediately. The INS calculates the check sum of the message (without its header and check sum) and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word). This check sum should be equal to the check sum in the message that was sent to the INS.

After acknowledgement, the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don't move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out message with block of the initial alignment data (see Section 6.3.12) and goes to the “Continuous” operating mode.

In the “Continuous” operating mode the program in the INS operates in the endless loop, providing calculation and output of position and orientation. Structure of output data depends on chosen output data format, refer to section 6.2 for their detailed description.

The update rate of data blocks is set by the user in range (1...200) Hz, but maximum data rate depends on chosen output data format and COM port baud rate (see Table 6.96).

6.3.2. SetOnRequestMode command – getting INS data on request (on demand)

The command SetOnRequestMode is used to start the Inertial Labs™ INS operation in the “On Request” (on demand) operating mode. This command has the byte structure shown in the Table 6.2 where payload is one byte equal to 0xC1.

INS acknowledges receiving any of these commands by responding back immediately. The INS calculates the check sum of the message (without its header and check sum) and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word). This check sum should be equal to the check sum in the message that was sent to the INS.

After acknowledgement, the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don’t move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out message with block of the initial data (see the Section 6.3.12) and goes to the “On Request” operating mode.

In the “On Request” operating mode the INS sends only one data block after each request. To get this data block send one of commands listed in Table 6.58. Note INS_SensorsData command is not supported in the “On Request” operating mode since the INS firmware version 2.1.1.0. Structure of output data block depends on chosen output data format, refer to section 6.2 for their detailed description.

6.3.3. Stop command

At receiving the Stop command (code 0xFE in the “Payload” field) the INS stops work in an operating mode and goes to the idle mode. At that the INS LED indicator changes its color to red. The INS is ready to receive any command from the host computer.

Important Note: Before using all other commands please send the **Stop** command to the INS to switch device into the idle mode. Be sure that the INS’s light indicator is red before sending of any other commands.

6.3.4. LoadINSpar command

The LoadINSpar command (code 0x40 in the “Payload” field) is used to load the block of the INS parameters (which are available for changing by user) into the INS nonvolatile memory. After sending the LoadINSpar command, the block of the INS parameters must be send to the INS in the message shown in the Table 6.2 with payload shown in the Table 6.59. This message should be sent without pause after sending the LoadINSpar command.

The INS calculates the check sum of received parameters and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (2 bytes).

Table 6.59. Payload of the message following after the LoadINSpar command (block of parameters for loading to the INS)

Byte	Parameter	Format	Length	*Note*
0-1	Data rate	**word**	**2**	Hz
2-3	Initial alignment time	**word**	**2**	seconds
4-7	Magnetic declination, Mdec	**longint**	**4**	degrees*100, if Mdec > 360 then INS calculates it
8-11	Latitude	**longint**	**4**	degrees*1e7
12-15	Longitude	**longint**	**4**	degrees*1e7
16-19	Altitude	**longint**	**4**	meters*100
20	Date (Year from 2000)	**byte**	**1**	0 to 255
21	Date (Month)	**byte**	**1**	1 to 12
22	Date (Day)	**byte**	**1**	1 to 31
23-24	Alignment angle A1	**sword**	**2**	Angles of INS mounting on the carrier object, degrees*100 (see Appendix B)
25-26	Alignment angle A2	**sword**	**2**	
27-28	Alignment angle A3	**sword**	**2**	

29-30	INS mount, right	**sword**	**2**	INS mounting lever relative to the object center of gravity, m*100 (see section 6.7)
31-32	INS mount, forward	**sword**	**2**	
33-34	INS mount, up	**sword**	**2**	
35-36	Antenna pos., right	**sword**	**2**	GNSS antenna mounting lever relative to the INS, meters*100
37-38	Antenna pos., forward	**sword**	**2**	
39-40	Antenna pos., up	**sword**	**2**	
41	Altitude	**byte**	**1**	1 = Altitude
42-49	Reserved	**byte**	**8**	
50-57	INS device name	**char**	**8**	**only read**, change is ignored
58	Baro_altimeter	**byte**	**1**	0 = disabled; 1 = primary altitude sensor, 2 =secondary altitude sensor
59	Reserved	**byte**	**1**	

Notes:

1. Since firmware version 3.3.0.1 user can change each INS parameter separately by sending LoadINSpar_RAM command (see section 6.3.14) or using the Inertial Labs™ INS GUI.

2. Before using **LoadINSpar** command it is necessary to use **ReadINSpar** command (see below) to read parameters from the INS at first. After that user can change some parameters listed in the Table 6.59, and to send back all block of parameters to the Inertial Labs™ INS.

3. Since firmware version 3.2.0.0, available are only data rates that are factors of 200: (1, 2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 200) Hz.

4. By default, the initial alignment time is set to 30 seconds. If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

5. It is necessary to set current latitude, longitude and altitude for setting the initial position in case of the GNSS data may be not available at the INS start.

6. It is necessary to set current latitude, longitude, altitude, year, month, day before hard/soft iron calibration of the INS magnetometers (see section 6.8).

7. Baro_altimeter switch enables or disables using of the pressure sensors data for the INS altitude correction. On default it is set to 0 (altitude sensor is disabled). See section 6.5 for details.

6.3.5. ReadINSpar command

The ReadINSpar command (code 0x41 in the “Payload” field, see the Table 6.2) is used to read block of the Inertial Labs™ INS parameters (60 bytes) from the INS nonvolatile memory.

After receiving ReadINSpar command, the INS sends out the message with structure according to Table 6.2 and payload shown in the Table 6.60.

Table 6.60. Payload of the INS answer on the ReadINSpar command (block of parameters read from the INS)

Byte	Parameter	Format	Length	*Note*
0-1	Data rate	**word**	**2**	(1 … 200) Hz
2-3	Initial alignment time	**word**	**2**	seconds
4-7	Magnetic declination, Mdec	**longint**	**4**	degrees*100
8-11	Latitude	**longint**	**4**	degrees*1e7
12-15	Longitude	**longint**	**4**	degrees*1e7
16-19	Altitude	**longint**	**4**	meters*100
20	Date (Year from 2000)	**byte**	**1**	0 to 255
21	Date (Month)	**byte**	**1**	1 to 12
22	Date (Day)	**byte**	**1**	1 to 31
23-24	Alignment angle A1	**sword**	**2**	Angles of INS mounting on the carrier object, degrees*100 (see Appendix B)
25-26	Alignment angle A2	**sword**	**2**	
27-28	Alignment angle A3	**sword**	**2**	
29-30	INS mount, right	**sword**	**2**	INS mounting lever relative to the object center of gravity, m*100 (see section 6.7)
31-32	INS mount, forward	**sword**	**2**	
33-34	INS mount, up	**sword**	**2**	
35-36	Antenna pos., right	**sword**	**2**	GNSS antenna mounting lever relative to the INS, meters*100
37-38	Antenna pos., forward	**sword**	**2**	
39-40	Antenna pos., up	**sword**	**2**	
41	Altitude	**byte**	**1**	1 = Altitude
42-49	Reserved	**byte**	**8**	
50-57	INS device name	**char**	**8**	
58	Baro_altimeter	**byte**	**1**	0 = disabled; 1 = primary altitude sensor, 2 =secondary altitude sensor
59	Reserved	**byte**	**1**	

See Notes to the section “6.3.4. LoadINSpar command”.

6.3.6. GetDevInfo command

The GetDevInfo command (code 0x12 in the “Payload” field) is used to get detailed information about devices installed in the INS:

1) INS processor;
2) IMU;
3) GNSS receiver;
4) Pressure sensor.

As answer the INS sends out the message with structure according to the Table 6.2 and payload shown in the Table 6.61.

Table 6.61. Payload of the INS answer on the GetDevInfo command

Byte	Parameter	Format	Length	*Note*
0-7	ID_sn	char	8	Integrated device (INS) s/n
8-47	ID_fw	char	40	INS firmware version
48	Press_Sens	byte	1	Pressure sensor: 0 = absent, 1= Type1, 2= Type2
49	IMU_type	byte	1	IMU type
50-57	IMU_sn	char	8	IMU s/n
58-97	IMU_fw	char	40	IMU firmware version
98-113	GNSS_model	char	16	GNSS receiver model
114-129	GNSS_sn	char	16	GNSS receiver product s/n
130-145	GNSS_hw	char	16	GNSS receiver hardware version
146-161	GNSS_fw	char	16	GNSS receiver firmware version
162-163	GPS_week	word	2	GPS reference week number
164	GNSS_data_rate	byte	1	GNSS receiver max data rate, Hz
165	Reserved	byte	1	Reserved

6.3.7. GetBIT command

The Inertial Labs™ INS has continuous built-in monitoring of its health. In both “Continuous” and “On Request” operation modes the INS sends out the Unit Status Word (USW) in each data block (see Tables 6.4, 6.9, 6.10, 6.11, 6.12, 6.13, 6.15, 6.21, 6.22, 6.27, 6.28). The USW is described in the section 6.10.

The USW can be got in any time if the INS is in Idle or “On Request” operation mode (after SetOnRequestMode command). For this the **GetBIT** command (code 0x1A in the “Payload” field) is used. In answer the INS sends out the message with data according to the Table 6.62.

Table 6.62. Payload of the INS answer on the GetBIT command

Byte number	0 – 1	2 – 3
Parameter	Utermo100	USW
Length	2 byte word	2 byte word

In the Table 6.62 Utermo100 is the INS temperature in 1/100 °C increments.

6.3.8. User_Def_Data_config command

To set necessary data types for the "User Defined Data" output the User_Def_Data_config command (code 0x96 in the "Payload" field) should be sent followed by message with block of parameters listed in the Table 6.63. This message has the byte structure shown in the Table 6.2, and should be sent without pause after sending the User_Def_Data_config command.

Table 6.63. Payload of the message following after the User_Def_Data_config command and answer on the Get_User_Def_Data_struct command

Byte	Parameter	Format	Length	Note
0	Pckg Number	byte	1	Number of data packages for output (N)
1 to N	Data List	byte	N	List of data types, one per byte, refer to Table 6.29 for values

The INS calculates the check sum of received message with parameters and returns it for a checking. Also the INS checks correctness of the list of data types. Byte structure of the INS answer corresponds to the Table 6.2 where payload is shown in Table 6.64.

Table 6.64. Payload of the INS answer on the User_Def_Data_config command with list of data types

Byte number	0 – 1	2	3 – 4	5	6	7
Parameter	Check sum of received message	Data list error	Data types correctness	Maximum data rate	Reserved	Reserved
Length	1 word	1 byte	1 word	1 byte	1 byte	1 byte

Notes:

1. Calculated check sum in bytes 0 – 1 should be equal to the check sum of the message with payload Table 6.63 that was sent to the INS. Otherwise the User_Def_Data_config command is ignored.

2. "Data list error" can have values listed in the Table 6.65.

3. If even one of bits #0 to #2 of the "Data list error" is not zero then the User_Def_Data_config command is ignored. In such case the check sum of the message after this command is not calculated and bytes #0, #1 in INS answer Table 6.64 are zero.

4. If bit #3 of the "Data list error" is not zero then User_Def_Data_config command is still accepted. The same as for other data formats, after INS start it checks compliance of data package size with set data rate and COM port baud rate. In case of no compliance the INS data rate will be corrected to allowed (lower) value.

5. "Data types correctness" indicates correctness of each ordered data type for up to 16 first types. If i-th data type is correct then i-th bit of the "Data list correctness" is zero. Otherwise i-th bit is set to 1. If "Data list correctness" is not zero, then the User_Def_Data_config command is ignored.

6. “Maximum data rate” shows the maximum data rate (Hz) for user defined data at current baud rate.

Table 6.65. Codes of the Data list error

Bit	Value and description
0	1 – incorrect structure of the message following after the User_Def_Data_config command, with payload listed in Table 6.63; 0 – no errors in the message structure.
1	1 – incorrect data in the payload (Table 6.63) of the message following after the User_Def_Data_config command; 0 – no errors in the payload.
2	1 – wrong data types are ordered; 0 – no errors in data types. Wrong data types are shown in the “Data list correctness” field of the INS answer (Table 6.64).
3	1 – ordered list of data is too large for current output data rate and COM port baud rate; 0 – size of INS output data meets to current data rate and COM port baud rate.

6.3.9. Get_User_Def_Data_struct command

The Get_User_Def_Data_struct command (code 0x97 in the “Payload” field) can be send from the host computer to check the “User Defined Data” structure set by user last time. As answer on this command the INS sends out the message with information shown in the Table 6.63. This message have the byte structure shown in the Table 6.2.

6.3.10. CAN1_message_set_config and CAN2_message_set_config commands

It is possible to choose necessary CAN messages for output through CAN1 and CAN2 ports. For this purpose the CAN1_message_set_config command (code 0x98 in the “Payload” field) and CAN2_message_set_config command (code 0xB9 in the “Payload” field) should be sent followed by message with block of parameters listed in the Table 6.66. This message has the byte structure shown in the Table 6.2, and should be sent without pause after sending the CAN1_message_set_config or CAN2_message_set_config commands.

Table 6.66. Payload of the message following after the CAN1_message_set_config and CAN2_message_set_config commands as well as the answer on the Get_CAN1_message_set_struct and Get_CAN2_message_set_struct commands

Byte	Parameter	Format	Length	*Note*
0	Pckg Number	byte	1	Number of CAN data messages for output (N)
1 to N	Data List	byte	N	List of CAN data messages, one per byte, refer to Table 6.40 for values

The INS calculates the check sum of received message with parameters and returns it for a checking. Also the INS checks correctness of the list of data types. Byte structure of the INS answer corresponds to the Table 6.2 where payload is shown in Table 6.67.

Table 6.67. Payload of the INS answer on the CAN1_message_set_config and CAN2_message_set_config commands with list of data types

Byte number	0 – 1	2	3 – 4	5	6	7
Parameter	Check sum of received message	CAN data list error	CAN data types correctness	Maximum data rate	Reserved	Reserved
Length	1 word	1 byte	1 word	1 byte	1 byte	1 byte

Notes:

1. Calculated check sum in bytes 0 – 1 should be equal to the check sum of the message with payload Table 6.66 that was sent to the INS. Otherwise, the CAN1_message_set_config or CAN2_message_set_config command is ignored.
2. “CAN data list error” can have values listed in the Table 6.68.
3. If even one of bits #0 to #2 of the “CAN data list error” is not zero then the CAN1_message_set_config or CAN2_message_set_config command is ignored. In such case the check sum of the message after this command is not calculated and bytes #0, #1 in INS answer Table 6.67 are zero.
4. If bit #3 of the “CAN data list error” is not zero then CAN1_message_set_config or CAN2_message_set_config command is still accepted. The same as for other data formats, after INS start it checks compliance of data package size with set data rate and COM port baud rate. In case of no compliance the INS data rate will be corrected to allowed (lower) value.
5. “CAN data types correctness” indicates correctness of each ordered data type for up to 16 first types. If i-th data type is correct then i-th bit of the “Data list correctness” is zero. Otherwise i-th bit is set to 1. If “Data list correctness” is not zero, then the CAN1_message_set_config or CAN2_message_set_config command is ignored.
6. “Maximum data rate” shows the maximum data rate (Hz) for user defined data at current baud rate.

Table 6.68. Codes of the CAN data list error

Bit	Value and description
0	1 – incorrect structure of the message following after the CAN1_message_set_config or CAN2_message_set_config command, with payload listed in Table 6.66; 0 – no errors in the message structure.
1	1 – incorrect data in the payload (Table 6.66) of the message following after the CAN1_message_set_config or CAN2_message_set_config command; 0 – no errors in the payload.
2	1 – wrong data types are ordered; 0 – no errors in data types. Wrong data types are shown in the “Data list correctness” field of the INS answer (Table 6.67).
3	1 – ordered list of data is too large for current output data rate and COM port baud rate; 0 – size of INS output data meets to current data rate and COM port baud rate.

6.3.11. Get_CAN1_message_set_struct and Get_CAN2_message_set_struct commands

The Get_CAN1_message_set_struct command (code 0x99 in the "Payload" field) and Get_CAN2_message_set_struct command (code 0xBA in the "Payload" field) can be sent from the host computer to check the "CAN message set" structure ordered by user the last time for the CAN1 and CAN2 ports. As answer on any of these commands the INS sends out the message with information shown in Table 6.66. This message has the byte structure shown in Table 6.2.

6.3.12. Block of the initial alignment data

After receiving of any command to start INS run it begins the procedure of initial alignment. After completing of this procedure the INS gives out message with block of the initial alignment data.

Since INS firmware version 2.6.2.0 user can set short or extended block of the initial alignment data, via INS GUI (see INS GUI User's Manual, section ""IMU" tab of the "Devices Options" window" for details), and via the command 0x04 Init_align_block_type (see Table 6.80) since INS firmware version 3.3.0.1. It is recommended to choose the extended block of the initial alignment data which contains more information about results of initial alignment procedure. Short block is kept more for compatibility with older versions of INS firmware and INS GUI.

Byte structure of the block of the initial alignment data corresponds to the Table 6.2. Table 6.69 and Table 6.70 show detailed byte structure of the short block of the initial alignment data. Table 6.71 and Table 6.72 show detailed byte structure of the extended block of the initial alignment data.

Table 6.69. Byte structure of the short block of initial alignment data

Byte number	**0**	**1**	**2**	**3**	**4, 5**	**6..55**	**56, 57**
Parameter	Header 0	Header 1	Message type	Output data rate (Hz)	Message length	Payload	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	**50 bytes**	1 word
Note	0xAA	0x55	0x01	hexadecimal value	**0x38 0x00**	see Table 6.70	

Table 6.70. Structure of the payload of the short block of initial alignment data

Byte	**Parameter**	**Format**	**Length**	***Note***
0-11	Gyros bias	**float**	**3*4**	3 numbers in ADC codes
12-23	Average acceleration	**float**	**3*4**	3 numbers in ADC codes
24-35	Average magn. field	**float**	**3*4**	3 numbers in ADC codes
36-39	Initial Heading	**float**	**4**	degrees
40-43	Initial Roll	**float**	**4**	degrees
44-47	Initial Pitch	**float**	**4**	degrees

48-49	USW (see section 6.10)	**word**	**2**	0 – successful initial alignment ≠0 – unsuccessful

Table 6.71. Byte structure of the extended block of initial alignment data

Byte number	0	1	2	3	4, 5	6..133	134, 135
Parameter	Header 0	Header 1	Message type	Output data rate (Hz)	Message length	Payload	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	**128 bytes**	1 word
Note	0xAA	0x55	0x01	hexadecimal value	**0x86 0x00**	see Table 6.72	

Table 6.72. Structure of the payload of the extended block of initial alignment data

Byte	Parameter	Format	Length	*Note*
0-11	Gyros bias	**float**	**3*4**	3 numbers in ADC codes
12-23	Average acceleration	**float**	**3*4**	3 numbers in ADC codes
24-35	Average magn. field	**float**	**3*4**	3 numbers in ADC codes
36-39	Initial Heading	**float**	**4**	degrees
40-43	Initial Roll	**float**	**4**	degrees
44-47	Initial Pitch	**float**	**4**	degrees
48-49	USW (see section 6.10)	**word**	**2**	0 – successful initial alignment ≠0 – unsuccessful
50-53	Pressure sensor temperature, UT_sr	**long**	**4**	in ADC codes
54-57	Pressure data, UP_sr	**long**	**4**	in ADC codes
58-75	Temperatures in 3 gyros, 3 accelerometers, 3 magnetometers	**sword**	**9*2**	in ADC codes
76-83	Latitude	**double**	**8**	degrees
84-91	Longitude	**double**	**8**	degrees
92-99	Altitude	**double**	**8**	m
100-103	Velocity V_east	**float**	**4**	m/s
104-107	Velocity V_north	**float**	**4**	m/s
108-111	Velocity V_up	**float**	**4**	m/s
112-119	Gravity, G_true	**double**	**8**	m/s^2
120-123	Reserved1	**float**	**4**	
124-127	Reserved2	**float**	**4**	

6.3.13. DevSelfTest command

The DevSelfTest command (code 0x13 in the “Payload” field) is used to get result of the device self-test during INS initialization after power on.

As answer the INS sends out the message with structure according to the Table 6.2 and 4 bytes payload shown in the Table 6.73. If this payload is zero, then INS unit passed self-test successfully.

Table 6.73. Structure of the payload of INS answer on the DevSelfTest command

Bit	Value and description
0	1 – INS parameters are absent; 0 – INS parameters are present
1	1 – Undefined IMU baud rate; 0 – IMU baud rate is detected correctly
2	1 – Undefined IMU model; 0 – IMU model is read
3	1 – Undefined IMU firmware version; 0 – IMU firmware version is read
4	1 – Undefined IMU calibration date; 0 – IMU calibration date is read
5	1 – Undefined IMU serial number; 0 – IMU serial number is read
6	1 – GNSS receiver initialization error; 0 – receiver initialization is successful
7	1 – Error of IMU ID read; 0 – IMU ID is read successfully
8	1 – Error of IMU Status Buffer read; 0 – IMU Status Buffer is read successfully
9	1 – Undefined GNSS receiver baud rate; 0 – GNSS receiver baud rate is correct
10 to 31	Reserved

Note: The low byte is transmitted by first in this payload – the same as in all other multi-byte integer data.

6.3.14. Configuration of INS parameters

The LoadINSpar command can be used to load the fixed block of the INS parameters into the INS nonvolatile memory (see section “6.3.4. LoadINSpar command”), and the ReadINSpar command to read this block of INS parameters.

Since firmware version 3.3.0.1 user can change each INS parameter separately by sending LoadINSpar_RAM command. Table 6.74 shows the structure of LoadINSpar_RAM command. The command ReadINSpar_RAM can be used to read value of specified INS parameters. Structure of the ReadINSpar_RAM command is shown in Table 6.77.

Note LoadINSpar_RAM and ReadINSpar_RAM commands operate with parameter value stored in INS RAM only. To save the present configuration of INS parameters from RAM to flash memory, use SaveINSpar_FLS command (see Table 6.79).

Full list of parameters that user can change, with description, is shown in Table 6.80. Note in all multi-byte data the low byte is transmitted by first.

Table 6.74. The structure of LoadINSpar_RAM command to load a new parameter value to RAM

Byte number	0	1	2	3	4, 5	6	7	8…(n+7)	n+8, n+9
Parameter	Header 0	Header 1	Message type	INS data identifier	Message length	Command code	Parameter code	Parameter value	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	1 byte	1 byte	n bytes (variable)	1 word
Note	0xAA	0x55	0x00	0x00	n+8	0xB0			

Message length - the number of bytes in the message without header. It is equal to the “Parameter(s) value” length (n) + 8.

Parameter code – the code of the parameter which is changed, see Table 6.80.

Table 6.75. The INS answer on the LoadINSpar_RAM command

Byte number	0	1, 2
Parameter	Data list error (see Table 6.76)	Reserved
Length	1 byte	2 byte

Table 6.76. Codes of the Data list error

Bit	Value and description
0	1 – wrong parameter code; 0 – no errors in parameter code.
1	1 – incorrect data length; 0 – no errors in the payload.
2	1 – invalid value of parameter(s); 0 – parameter(s) is valid.

Table 6.77. The structure of ReadINSpar_RAM command to read parameter from RAM

Byte number	0	1	2	3	4, 5	6	7	8, 9
Parameter	Header 0	Header 1	Message type	INS data identifier	Message length	Command code	Parameter code	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	1 byte	1 byte	1 word
Note	0xAA	0x55	0x00	0x00	0x08	0xB1		

Table 6.78. The INS answer on the ReadINSpar_RAM command

Byte number	0	1	2	3	4, 5	6	7…(n+6)	n+7, n+8
Parameter	Header 0	Header 1	Message type	INS data identifier	Message length	Parameter code	Parameter value	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	1 byte	n bytes (variable)	1 word
Note	0xAA	0x55	0x01	0x00	n+7			

Table 6.79. The structure of SaveINSpar_FLS command to save all parameters to flash memory

Byte number	0	1	2	3	4, 5	6	7, 8
Parameter	Header 0	Header 1	Message type	INS data identifier	Message length	Command code	Check sum
Length	1 byte	1 byte	1 byte	1 byte	1 word	1 byte	1 word
Note	0xAA	0x55	0x00	0x00	0x07 0x00	0xB2	0xB9 0x00

For example, INS unit has to be configured so COM2 port outputs raw GNSS data with 10Hz frequency at baud rate 460800 bps, and this setting must be saved to Nonvolatile memory of the device. There has to be two commands executed to accomplish that. First LoadINSpar_RAM (to put the configuration into RAM of the unit), and then SaveINSpar_FLS (to right the configuration from RAM to nonvolatile memory).

First command is LoadINSpar_RAM – to output raw GNSS data with 10Hz frequency at baud rate 460800 bps through COM2 port:

AA 55 00 00 0B 00 B0 14 01 05 02 D7 00

Part of LoadINSpar_RAM command	AA 55 00 00
Message length	0B 00
code for LoadINSpar_RAM Command	B0
Parameter code (first column in Table 6.80)	14
GNSS_COM2_data_set = “Raw GNSS”	01
GNSS_COM2_data_rate = 10 Hz	05
GNSS_COM2_bps = 460800 bps	02
Check sum	D7 00

Second command is SaveINSpar_FLS (it has constant structure) – to save all current configurations from RAM memory to nonvolatile memory:

AA 55 00 00 07 00 B2 B9 00

Note: these are hexadecimal numbers, not ASCII text symbols.

For full list of available parameters and values which can be applied to a device, please see Table 6.80.

Table 6.80. Full list of parameters to control INS

<table>
<tr><th>Parameter code</th><th>Parameter name</th><th>Data type</th><th>Description</th></tr>
<tr><th colspan="4">1. Devices options</th></tr>
<tr><th colspan="4">1.1. IMU</th></tr>
<tr><td>0x01</td><td>COM1_bps</td><td>byte</td><td>Baud rate of the main INS COM1 port.
<table>
<tr><th>Baud rate</th><th>Value</th></tr>
<tr><td>4800 bps</td><td>0x01</td></tr>
<tr><td>9600 bps</td><td>0x02</td></tr>
<tr><td>14400 bps</td><td>0x03</td></tr>
<tr><td>19200 bps</td><td>0x04</td></tr>
<tr><td>38400 bps</td><td>0x05</td></tr>
<tr><td>57600 bps</td><td>0x06</td></tr>
<tr><td>115200 bps</td><td>0x07</td></tr>
<tr><td>230400 bps</td><td>0x08</td></tr>
<tr><td>460800 bps</td><td>0x09</td></tr>
</table>
Default value is set to 115200 bps.</td></tr>
<tr><td rowspan="2">0x02</td><td>Data_rate</td><td>word</td><td>Output data rate in Hertz. Minimal value of the parameter is 1, maximal value is 200.
Default value is set to 100.</td></tr>
<tr><td>Fixed_data_rate</td><td>byte</td><td>Limits data rates to be factors of 200 Hz: 1, 2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 200 Hz.
<table>
<tr><th>Type</th><th>Value</th></tr>
<tr><td>No fixed</td><td>0x00</td></tr>
<tr><td>Fixed</td><td>0x01</td></tr>
</table></td></tr>
<tr><td>0x03</td><td>Output_data_avrg</td><td>byte</td><td>The output of averaged or instant data at a data rate less than 200 Hz (see Appendix E).
<table>
<tr><th>Type of data</th><th>Value</th></tr>
<tr><td>Instant</td><td>0x00</td></tr>
<tr><td>Averaged</td><td>0x01</td></tr>
</table></td></tr>
<tr><td rowspan="2">0x04</td><td>Init_align_time</td><td>word</td><td>The initial alignment time in seconds.
Default value is set to 0 seconds.</td></tr>
<tr><td>Init_align_block_type</td><td>byte</td><td>Specifies format of block of the initial alignment (see section 6.3.12).
<table>
<tr><th>Block of the initial alignment</th><th>Value</th></tr>
<tr><td>Short block</td><td>0x00</td></tr>
<tr><td>Extended block</td><td>0x01</td></tr>
</table></td></tr>
<tr><td rowspan="4">0x05</td><td rowspan="4">Location_date</td><td>float</td><td>Initial latitude of the INS operating location in deg.</td></tr>
<tr><td>float</td><td>Initial longitude of the INS operating location in deg.</td></tr>
<tr><td>float</td><td>Initial altitude above sea level of the INS operational location in m.</td></tr>
<tr><td>2*byte, word</td><td>Day, month and year when the INS is used.</td></tr>
<tr><td>0x06</td><td>Mdec</td><td>float</td><td>Magnetic declination at the place where the INS operates in deg.</td></tr>
</table>

<table>
<tr><td></td><td>Mdec_AUTO</td><td>byte</td><td>Calculating magnetic declination in the INS continuously using calculated current position and time.<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table></td></tr>
<tr><td>0x07*</td><td>AutoStart</td><td>byte</td><td>Automatic start of the INS and data output after power on without any command from the host computer (see section 6.9).<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Output data format</td><td>0xXX</td></tr></table>0xXX – code of Output data format (see Table 6.58)</td></tr>
<tr><td rowspan="2">0x08</td><td>AlAngles</td><td>3*float</td><td>Angles of INS mounting on the carrier object in deg. (see Appendix B).
Default values are set to 0 degrees.</td></tr>
<tr><td>Sensors_output_axes</td><td>byte</td><td>Sensors output data in the INS or carrier object axes.<table><tr><th>Type of axes</th><th>Value</th></tr><tr><td>Object axes</td><td>0x00</td></tr><tr><td>INS axes</td><td>0x01</td></tr></table>By default, sensors data are output in the carrier object axes.</td></tr>
<tr><td>0x09</td><td>PV_meas_point</td><td>3*sword</td><td>Coordinates of measuring point for Position and Velocity calculation. Set coordinates of the measuring point relative to the accelerometer mass-center of the INS unit (see Fig. 1.6 and Fig. 1.7) in the object axes – on the right, forward and up directions, in m*100.
Also, it is possible to output accelerations for set measuring point through UDD format (see “Accelerometer data in PV measuring point” data type under “Sensors data” group in Table 6.29).
Default values are set to 0 m.</td></tr>
<tr><td>0x0A</td><td>Ant1_pos</td><td>3*sword</td><td>Primary GNSS antenna mounting lever relative to the INS unit, in m*100 (see section 4.2)
Default values are set to 0 m.</td></tr>
<tr><td>0x0B</td><td>Ant2_pos</td><td>3*sword</td><td>Secondary GNSS antenna mounting lever relative to the INS unit, in m*100. This option is supported for INS-D (see section 4.3).
Default values are set to 0 m.</td></tr>
<tr><td>0x0C</td><td>Ant2_pos_angl</td><td>2*sword</td><td>Alternate way to set the secondary antenna position – as orientation of the baseline between two antennas relative to the carrier object axes, in deg*100. This option is supported for INS-D. (see section 4.3)
Default values are set to 0 degrees.</td></tr>
</table>

<table>
<tr><td>0x0D</td><td>Save_last_pos</td><td>byte</td><td>The last calculated position may be written to the INS flash memory after INS stop. Thus, the next operation will be started from its saved point.
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Disable saving</td><td>0x00</td></tr>
<tr><td>Enable saving</td><td>0x01</td></tr>
</table>
By default, saving last position is disabled.</td></tr>
<tr><td colspan="2">0x0E – 0x11</td><td colspan="2">Reserved</td></tr>
<tr><td colspan="4">1.2. GNSS receiver</td></tr>
<tr><td>0x12*</td><td>GNSS_data_rate</td><td>byte</td><td>Rate of the GNSS data update, in Hz.</td></tr>
<tr><td>0x13*</td><td>Altitude_var</td><td>byte</td><td>Variant of altitude output – above mean sea level (MSL) or above WGS84 ellipsoid (see section 6.5).
<table>
<tr><th>Height type</th><th>Value</th></tr>
<tr><td>MSL</td><td>0x00</td></tr>
<tr><td>WGS84</td><td>0x01</td></tr>
</table></td></tr>
<tr><td rowspan="3">0x14*</td><td>GNSS_COM2_data_set</td><td>byte</td><td>GNSS receiver data for output through COM2 port.
<table>
<tr><th>Data set</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Raw GNSS</td><td>0x01</td></tr>
<tr><td>NMEA set</td><td>0x02</td></tr>
<tr><td>Min Raw GNSS</td><td>0x03</td></tr>
</table></td></tr>
<tr><td>GNSS_COM2_data_rate</td><td>byte</td><td>Frequency of COM2 port GNSS data output.
<table>
<tr><th>Frequency</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>1 Hz</td><td>0x01</td></tr>
<tr><td>2 Hz</td><td>0x02</td></tr>
<tr><td>4 Hz</td><td>0x03</td></tr>
<tr><td>5 Hz</td><td>0x04</td></tr>
<tr><td>10 Hz</td><td>0x05</td></tr>
<tr><td>20 Hz</td><td>0x06</td></tr>
</table></td></tr>
<tr><td>GNSS_COM2_bps</td><td>byte</td><td>Baud rate of the COM2 port for output GNSS data.
<table>
<tr><th>Baud rate</th><th>Value</th></tr>
<tr><td>115200 bps</td><td>0x00</td></tr>
<tr><td>230400 bps</td><td>0x01</td></tr>
<tr><td>460800 bps</td><td>0x02</td></tr>
<tr><td>9600 bps</td><td>0x04</td></tr>
</table></td></tr>
</table>

<table>
<tr><td rowspan="2">0x15*</td><td>NMEA_set_COM2</td><td>word</td><td>NMEA messages for output through COM2 port when NMEA Set is chosen in GNSS_COM2_data_set.
<table>
<tr><th>NMEA set</th><th>Mask</th></tr>
<tr><td>GGA</td><td>0x0001</td></tr>
<tr><td>GSA</td><td>0x0002</td></tr>
<tr><td>RMC</td><td>0x0004</td></tr>
<tr><td>VTG</td><td>0x0008</td></tr>
<tr><td>ZDA</td><td>0x0010</td></tr>
<tr><td>HDT</td><td>0x0020</td></tr>
<tr><td>Reserved</td><td>0x0040 – 0x8000</td></tr>
</table>
By default, INS outputs GGA, VTG and ZDA messages if GNSS_COM2_data_set value is set to 0x02.
<u>Note</u>: HDT message is available only in dual antenna receivers.</td></tr>
<tr><td>NMEA_COM2_data_rate</td><td>5*byte</td><td>Frequency for each NMEA message in such order: GGA, GSA, RMC, VTG, ZDA.
<table>
<tr><th>Frequency</th><th>Value</th></tr>
<tr><td>1 Hz</td><td>0x00</td></tr>
<tr><td>2 Hz</td><td>0x01</td></tr>
<tr><td>4 Hz</td><td>0x02</td></tr>
<tr><td>5 Hz</td><td>0x03</td></tr>
<tr><td>10 Hz</td><td>0x04</td></tr>
<tr><td>20 Hz</td><td>0x05</td></tr>
</table>
<u>Note</u>: If no one NMEA message is selected, but “NMEA set” is chosen in “GNSS_COM2_data_set”, the following NMEA messages are issued by default, with the frequency stored in the flash in such order: GGA, VTG, ZDA.</td></tr>
<tr><td rowspan="2">0x16*</td><td>GNSS_COM3_data_set</td><td>byte</td><td>GNSS receiver data for output through COM3 port.
<table>
<tr><th>Type</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Raw GNSS</td><td>0x01</td></tr>
<tr><td>GPRMC</td><td>0x02</td></tr>
<tr><td>Min Raw GNSS</td><td>0x03</td></tr>
</table>
By default, it is disabled.
<u>Note</u>: If GNSS_COM3_data_set parameter is set to nonzero value then COM3 port can’t be used for input of the GNSS corrections.</td></tr>
<tr><td>GNSS_COM3_data_rate</td><td>byte</td><td>Frequency of GNSS receiver data output through COM3 port.
<table>
<tr><th>Frequency</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>1 Hz</td><td>0x01</td></tr>
<tr><td>2 Hz</td><td>0x02</td></tr>
<tr><td>4 Hz</td><td>0x03</td></tr>
<tr><td>5 Hz</td><td>0x04</td></tr>
<tr><td>10 Hz</td><td>0x05</td></tr>
<tr><td>20 Hz</td><td>0x06</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td>GNSS_COM3_bps</td><td>byte</td><td>Baud rate of the COM3 port for output GNSS data.
<table>
<tr><th>Baud rate</th><th>Value</th></tr>
<tr><td>115200 bps</td><td>0x00,
0x05</td></tr>
<tr><td>9600 bps</td><td>0x01</td></tr>
<tr><td>19200 bps</td><td>0x02</td></tr>
<tr><td>38400 bps</td><td>0x03</td></tr>
<tr><td>57600 bps</td><td>0x04</td></tr>
<tr><td>230400 bps</td><td>0x06</td></tr>
<tr><td>460800 bps</td><td>0x07</td></tr>
</table></td></tr>
<tr><td>0x17*</td><td>GNSS_corr_type</td><td>byte</td><td>Correction of GNSS receiver data to improve position accuracy.
<table>
<tr><th>Type of correction</th><th>Value</th></tr>
<tr><td>No Correction</td><td>0x00</td></tr>
<tr><td>Auto</td><td>0x01</td></tr>
<tr><td>SBAS</td><td>0x02</td></tr>
<tr><td>DGPS</td><td>0x03</td></tr>
<tr><td>TerraStar</td><td>0x04</td></tr>
</table>
Default value is “AUTO”.
<u>Note</u>: If the type of the GNSS_COM3_data_rate is "Disabled", then the following correction options are not available: “No Correction”, “SBAS”.</td></tr>
<tr><td>0x18*</td><td>SBAScontrol</td><td>byte</td><td>Specifies type of SBAS correction.
<table>
<tr><th>SBAS correction</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Auto SBAS</td><td>0x01</td></tr>
<tr><td>Any SBAS</td><td>0x02</td></tr>
<tr><td>WAAS</td><td>0x03</td></tr>
<tr><td>EGNOS</td><td>0x04</td></tr>
<tr><td>MSAS</td><td>0x05</td></tr>
<tr><td>GAGAN</td><td>0x06</td></tr>
<tr><td>QZSS</td><td>0x07</td></tr>
</table>
Default value is “Auto SBAS”</td></tr>
<tr><td>0x19*</td><td>GNSS_corr_format</td><td>byte</td><td>Specifies format of differential correction (DGPS) data.
<table>
<tr><th>Type of format</th><th>Value</th></tr>
<tr><td>Auto</td><td>0x00</td></tr>
<tr><td>RTCMv2</td><td>0x01</td></tr>
<tr><td>RTCMv3</td><td>0x02</td></tr>
</table>
Default value is “Auto”.
<u>Note</u>: COM3 port can be used for input of the GNSS corrections <u>only if</u> GNSS_COM3_data_set parameter is disabled.</td></tr>
<tr><td>0x1A*</td><td>COM2_COM3_port_type</td><td>byte</td><td>GNSS receiver’s COM2, COM3 port type:
<table>
<tr><th>Port</th><th>Port type</th><th>Mask</th></tr>
<tr><td>COM2</td><td rowspan="2">0 – COM
1 – Ethernet</td><td>0x01</td></tr>
<tr><td>COM3</td><td>0x02</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>0x1B*</td><td>TerraStarType</td><td>byte</td><td>Specifies type of TerraStar correction.
<table>
<tr><th>TerraStar correction</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Auto</td><td>0x01</td></tr>
<tr><td>TerraStar-L</td><td>0x02</td></tr>
<tr><td>TerraStar-C</td><td>0x03</td></tr>
<tr><td>TerraStar-C-PRO</td><td>0x04</td></tr>
<tr><td>TerraStar-X</td><td>0x05</td></tr>
</table></td></tr>
<tr><td>0x1C*</td><td>Enable_PDPFILTER</td><td>byte</td><td>Allows to enable the Pseudorange/Delta-Phase (PDP) filter in the NovAtel receiver to smooth a jumpy GNSS position and to bridge outages in satellite coverage.
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Disable PDP filter</td><td>0x00</td></tr>
<tr><td>Enable PDP filter</td><td>0x01</td></tr>
</table>
By default, PDP filter is disabled.</td></tr>
<tr><td colspan="2">0x1D - 0x25</td><td colspan="2">Reserved</td></tr>
<tr><th colspan="4">1.3. Pressure sensors</th></tr>
<tr><td rowspan="2">0x26</td><td>Baro_enabled</td><td>byte</td><td>Variant of the pressure sensor using for INS altitude correction.
<table>
<tr><th>Variant</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Primary altitude sensor</td><td>0x01</td></tr>
<tr><td>Secondary altitude sensor</td><td>0x02</td></tr>
</table>
On default is disabled.</td></tr>
<tr><td>STD_h_bar</td><td>byte</td><td>Standard deviation of the pressure sensor noise, in m*100.</td></tr>
<tr><td colspan="2">0x27 - 0x28</td><td colspan="2">Reserved</td></tr>
<tr><th colspan="4">1.4. External sensors</th></tr>
<tr><td>0x29*</td><td>Odometer_type</td><td>byte</td><td>Odometer type (see sections 6.6.1, 6.6.2).
<table>
<tr><th>Odometer type</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>OBDII odometer</td><td>0x01</td></tr>
<tr><td>Encoder-based odometer</td><td>0x02</td></tr>
<tr><td>CAN based odometer</td><td>0x03</td></tr>
</table></td></tr>
<tr><td rowspan="4">0x2A</td><td>Pulse_length</td><td>Float</td><td>Pulse length of encoder – distance between low to high transitions of the encoder signal, in m.</td></tr>
<tr><td>Wheel_circ</td><td>word</td><td>Wheel circumference where encoder is installed, in m*1000.</td></tr>
<tr><td>Ticks_per_rev</td><td>word</td><td>Number of ticks per revolution (for “Encoder-based odometer”).</td></tr>
<tr><td>ZUPT_threshold</td><td>byte</td><td>Velocity threshold for stop detection at operation with encoder, in m/s*100</td></tr>
<tr><td>0x2B</td><td>Odom_offset</td><td>3*sword</td><td>The lever arm measured from the accelerometer mass-center of the INS unit to the point at which the vehicle's tire makes contact with the road in the vehicle co-ordinate frame, in the right, forward and vertical directions, in m*100</td></tr>
</table>

<table>
<tr><td>0x2C</td><td>COM1_Aiding_data</td><td>byte</td><td>Allow aiding data input through COM1 port (see section 6.6.4).<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>By default, aiding data input is disabled.</td></tr>
<tr><td>0x2D</td><td>STD_pos_h_ext</td><td>word</td><td>Noise of external horizontal position aiding data, in m*100.</td></tr>
<tr><td>0x2E</td><td>STD_pos_v_ext</td><td>word</td><td>Noise of external vertical position aiding data, in m*100.</td></tr>
<tr><td>0x2F</td><td>STD_Vh_ext</td><td>byte</td><td>Noise of external horizontal speed data, in m/s*100.</td></tr>
<tr><td>0x30</td><td>STD_Vv_ext</td><td>word</td><td>Noise of external vertical speed data, in m/s*100.</td></tr>
<tr><td>0x31</td><td>k_Sigma_V_ext</td><td>byte</td><td>Kalman filter velocity residuals thresholds.</td></tr>
<tr><td>0x32</td><td>k_Sigma_coord_ext</td><td>word</td><td>Kalman filter coordinate residuals thresholds.</td></tr>
<tr><td>0x33*</td><td>Ext_MC</td><td>byte</td><td>Allow the external Stand Alone Magnetic Compass (see section 5.10).<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>Note: INS should be factory configured to use the external SAMC to measure components of the Earth magnetic field.</td></tr>
<tr><td>0x34</td><td>DVL_reference_frame</td><td>byte</td><td>Specifies the coordinate frame in which DVL data are defined.<table><tr><th>Type of coordinate frame</th><th>Value</th></tr><tr><td>Object</td><td>0x00</td></tr><tr><td>East, North, Up</td><td>0x01</td></tr></table>By default, velocity is in object coordinate frame.</td></tr>
<tr><td colspan="2">0x35 – 0x41</td><td colspan="2">Reserved</td></tr>
<tr><th colspan="4">1.5. CAN/COM4</th></tr>
<tr><td rowspan="2">0x42*</td><td>Use_CAN1_out</td><td>byte</td><td>Type of CAN output messages through CAN1 port (see section 6.2.18).<table><tr><th>Type</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>CAN 2.0A</td><td>0x01</td></tr><tr><td>CAN 2.0B</td><td>0x02</td></tr></table>By default it is disabled.</td></tr>
<tr><td>CAN1_bps</td><td>byte</td><td>CAN1 port baud rate (both for input and output messages). Its value can also be set by command for parameter 0x46 “CAN_bps” change.<table><tr><th>Baud rate</th><th>Value</th></tr><tr><td>10k</td><td>0x00</td></tr><tr><td>20k</td><td>0x01</td></tr><tr><td>50k</td><td>0x02</td></tr><tr><td>100k</td><td>0x03</td></tr><tr><td>125k</td><td>0x04</td></tr><tr><td>250k</td><td>0x05</td></tr><tr><td>500k</td><td>0x06</td></tr><tr><td>1M</td><td>0x07</td></tr></table></td></tr>
</table>

<table>
<tr><td></td><td>CAN1_dev_ID</td><td>byte</td><td>CAN device identifier – part of the CAN message identifier (see section 6.2.18). In case of CAN2.0A messages the device identifier has maximum value 0x7F.</td></tr>
<tr><td></td><td>CAN1_Priority_EDP_DP</td><td>byte</td><td>Composed part of CAN2.0B message identifier (see section 6.2.18) that includes Priority (3 bits), Extended Data Page (1 bit) and Data Page (1 bit).
Parameter can be changed in range 0x00 to 0x1F.</td></tr>
<tr><td></td><td>CAN1_PDU_Format</td><td>byte</td><td>Part of CAN2.0B message identifier (see section 6.2.18), it can be changed in range 0x00 to 0xFF.</td></tr>
<tr><td>0x43*</td><td>CAN1_regular_msg</td><td>byte</td><td>Regular set of CAN output messages through CAN1 port (see section 6.2.18)
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Enabled</td><td>0x01</td></tr>
</table></td></tr>
<tr><td rowspan="2">0x44*</td><td>COM4_data_var</td><td>byte</td><td>Variant of COM4 port using.
<table>
<tr><th>Variant</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>OBDII odometer</td><td>0x01</td></tr>
<tr><td>GPRMC output</td><td>0x02</td></tr>
<tr><td>IMU TGA output</td><td>0x03</td></tr>
<tr><td>HEHDT output</td><td>0x04</td></tr>
<tr><td>COM1 duplicated data</td><td>0x05</td></tr>
</table></td></tr>
<tr><td>COM4_data_rate</td><td>byte</td><td>Frequency of input/output through COM4 port.
<table>
<tr><th>Frequency</th><th>Value</th></tr>
<tr><td>1 Hz</td><td>0x00</td></tr>
<tr><td>2 Hz</td><td>0x01</td></tr>
<tr><td>4 Hz</td><td>0x02</td></tr>
<tr><td>5 Hz</td><td>0x03</td></tr>
<tr><td>10 Hz</td><td>0x04</td></tr>
<tr><td>20 Hz</td><td>0x05</td></tr>
<tr><td>50 Hz</td><td>0x06</td></tr>
<tr><td>Maximum</td><td>0x07</td></tr>
</table>
<u>Notes</u>:
1. Value “Maximum” of Data rate is used only for “IMU TGA output”.
2. Value “50 Hz” of Data rate is used only for “OBDII odometer”.</td></tr>
</table>

<table>
<tr><td></td><td>COM4_bps</td><td>byte</td><td>COM4 port baud rate.
<table>
<tr><th>Baud rate</th><th>Value</th></tr>
<tr><td>4800 bps</td><td>0x01</td></tr>
<tr><td>9600 bps</td><td>0x02</td></tr>
<tr><td>14400 bps</td><td>0x03</td></tr>
<tr><td>19200 bps</td><td>0x04</td></tr>
<tr><td>38400 bps</td><td>0x05</td></tr>
<tr><td>57600 bps</td><td>0x06</td></tr>
<tr><td>115200 bps</td><td>0x07</td></tr>
<tr><td>230400 bps</td><td>0x08</td></tr>
<tr><td>460800 bps</td><td>0x09</td></tr>
<tr><td>921600 bps</td><td>0x0A</td></tr>
<tr><td>2000000 bps</td><td>0x0B</td></tr>
</table>
<u>Note</u>: For “IMU TGA” data the baud rate of COM4 port should be set to not less than the next values to avoid skips:
• 460800 or 921600 bps – for 800 Hz IMU data rate;
• 2000000 bps – for 2000 Hz IMU data rate.</td></tr>
<tr><td>0x46*</td><td>CAN1_bps</td><td>byte</td><td>CAN baud rate (both for input and output messages). Its value can also be set by command for parameter group 0x42 change, “CAN_bps” field.
<table>
<tr><th>Baud rate</th><th>Value</th></tr>
<tr><td>10k</td><td>0x00</td></tr>
<tr><td>20k</td><td>0x01</td></tr>
<tr><td>50k</td><td>0x02</td></tr>
<tr><td>100k</td><td>0x03</td></tr>
<tr><td>125k</td><td>0x04</td></tr>
<tr><td>250k</td><td>0x05</td></tr>
<tr><td>500k</td><td>0x06</td></tr>
<tr><td>1M</td><td>0x07</td></tr>
</table></td></tr>
<tr><td rowspan="5">0x47*</td><td>CAN_Inp_Sig1_MessType</td><td>byte</td><td>CAN BUS velocity data. Type of CAN signal 1.
<table>
<tr><th>Type</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>CAN 2.0A</td><td>0x01</td></tr>
<tr><td>CAN 2.0B</td><td>0x02</td></tr>
</table></td></tr>
<tr><td>CAN_Inp_Sig1_MesID</td><td>4 byte unsigned integer</td><td>CAN message identifier. This number is a positive integer from 0x0 to 0x7FF for CAN 2.0A identifier and from 0x0 to 0x1FFFFFFF for CAN 2.0B identifier.</td></tr>
<tr><td>CAN_Inp_Sig1_DataType</td><td>byte</td><td>Signal 1 data type in CAN message.
<table>
<tr><th>Type</th><th>Value</th></tr>
<tr><td>None</td><td>0x00</td></tr>
<tr><td>Velocity</td><td>0x01</td></tr>
</table></td></tr>
<tr><td>CAN_Inp_Sig1_Startbit</td><td>byte</td><td>The start bit is the least significant bit of signal 1 counted from the start of the message data. The start bit is an integer from 0 to 63.</td></tr>
<tr><td>CAN_Inp_Sig1_Length</td><td>byte</td><td>The number of bits the signal 1 occupies in the message. The length is an integer from 1 through 64.</td></tr>
</table>

	CAN_Inp_Sig1_ByteOrder	byte	Message byte order. **Type** – **Value** little-endian – 0x00 big-endian – 0x01
	CAN_Inp_Sig1_ValueType	byte	Signal 1 value data type. **Type** – **Value** Unsigned – 0x00 Signed – 0x01
	CAN_Inp_Sig1_Factor	float	The factor and offset define the linear conversion rule to convert the signal raw value into the signal's physical value (in km/h) and vice versa, *physical_value = raw_value * factor + offset*
	CAN_Inp_Sig1_Offset	float	
0x48*	CAN_Inp_Sig2_MessType	byte	CAN BUS velocity decimal data. Type of CAN signal 2. **Type** – **Value** Disabled – 0x00 CAN 2.0A – 0x01 CAN 2.0B – 0x02
	CAN_Inp_Sig2_MesID	4 byte unsigned integer	CAN message identifier. This number is a positive integer from 0x0 to 0x7FF for CAN 2.0A identifier and from 0x0 to 0x1FFFFFFF for CAN 2.0B identifier.
	CAN_Inp_Sig2_DataType	byte	Signal 2 data type in CAN message. **Type** – **Value** None – 0x00 Velocity decimal – 0x01
	CAN_Inp_Sig2_Startbit	byte	The start bit is the least significant bit of signal 2 counted from the start of the message data. The start bit is an integer from 0 to 63.
	CAN_Inp_Sig2_Length	byte	The number of bits the signal 2 occupies in the message. The length is an integer from 1 through 64.
	CAN_Inp_Sig2_ByteOrder	byte	Message Byte Order. **Type** – **Value** little-endian – 0x00 big-endian – 0x01
	CAN_Inp_Sig2_ValueType	byte	Signal 2 value data type. **Type** – **Value** Unsigned – 0x00 Signed – 0x01
	CAN_Inp_Sig2_Factor	float	The factor and offset define the linear conversion rule to convert the signal raw value into the signal's physical value (in km/h) and vice versa, *physical_value = raw_value * factor + offset*
	CAN_Inp_Sig2_Offset	float	
0x49*	CAN_Inp_Sig3_MessType	byte	CAN BUS transmission actual gear data. Type of CAN signal 3. **Type** – **Value** Disabled – 0x00 CAN 2.0A – 0x01 CAN 2.0B – 0x02

<table>
<tr><td rowspan="7"></td><td>CAN_Inp_Sig3_MesID</td><td>4 byte unsigned integer</td><td>CAN message identifier. This number is a positive integer from 0x0 to 0x7FF for CAN 2.0A identifier and from 0x0 to 0x1FFFFFFF for CAN 2.0B identifier.</td></tr>
<tr><td>CAN_Inp_Sig3_DataType</td><td>byte</td><td>Signal 3 data type in CAN message.<table><tr><th>Type</th><th>Value</th></tr><tr><td>None</td><td>0x00</td></tr><tr><td>Transmission Actual Gear</td><td>0x01</td></tr></table></td></tr>
<tr><td>CAN_Inp_Sig3_Startbit</td><td>byte</td><td>The start bit is the least significant bit of signal 3 counted from the start of the message data. The start bit is an integer from 0 to 63.</td></tr>
<tr><td>CAN_Inp_Sig3_Length</td><td>byte</td><td>The number of bits the signal 3 occupies in the message. The length is an integer from 1 through 64.</td></tr>
<tr><td>CAN_Inp_Sig3_ByteOrder</td><td>byte</td><td>Message Byte Order.<table><tr><th>Type</th><th>Value</th></tr><tr><td>little-endian</td><td>0x00</td></tr><tr><td>big-endian</td><td>0x01</td></tr></table></td></tr>
<tr><td>CAN_Inp_Sig3_ValueType</td><td>byte</td><td>Signal 3 value data type.<table><tr><th>Type</th><th>Value</th></tr><tr><td>Unsigned</td><td>0x00</td></tr><tr><td>Signed</td><td>0x01</td></tr></table></td></tr>
<tr><td>CAN_Inp_Sig3_ReverseGear</td><td>byte</td><td>Value of the signal that corresponds to the reverse gear (reverse vehicle motion)</td></tr>
<tr><td rowspan="4">0x4A*</td><td>Use_CAN2_out</td><td>byte</td><td>Type of CAN output messages through CAN2 port (see section 6.2.18).<table><tr><th>Type</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>CAN 2.0A</td><td>0x01</td></tr><tr><td>CAN 2.0B</td><td>0x02</td></tr></table>By default it is disabled.</td></tr>
<tr><td>CAN2_bps</td><td>byte</td><td>CAN2 port baud rate for output messages.<table><tr><th>Baud rate</th><th>Value</th></tr><tr><td>10k</td><td>0x00</td></tr><tr><td>20k</td><td>0x01</td></tr><tr><td>50k</td><td>0x02</td></tr><tr><td>100k</td><td>0x03</td></tr><tr><td>125k</td><td>0x04</td></tr><tr><td>250k</td><td>0x05</td></tr><tr><td>500k</td><td>0x06</td></tr><tr><td>1M</td><td>0x07</td></tr></table></td></tr>
<tr><td>CAN2_dev_ID</td><td>byte</td><td>CAN device identifier – part of the CAN message identifier (see section 6.2.18). In case of CAN 2.0A messages the device identifier has maximum value 0x7F.</td></tr>
<tr><td>CAN2_Priority_EDP_DP</td><td>byte</td><td>Composed part of CAN 2.0B message identifier (see section 6.2.18) that includes Priority (3 bits), Extended Data Page (1 bit) and Data Page (1 bit).
Parameter can be changed in range 0x00 to 0x1F.</td></tr>
</table>

<table>
<tr><td></td><td>CAN2_PDU_Format</td><td>byte</td><td>Part of CAN2.0B message identifier (see section 6.2.18), it can be changed in range 0x00 to 0xFF.</td></tr>
<tr><td>0x4B*</td><td>CAN2_regular_msg</td><td>byte</td><td>Regular set of CAN output messages through CAN2 port (see section 6.2.18)
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Enabled</td><td>0x01</td></tr>
</table></td></tr>
<tr><td colspan="2">0x4C – 0x4F</td><td colspan="2">Reserved</td></tr>
<tr><td colspan="4">1.6. Triggers</td></tr>
<tr><td rowspan="4">0x50*</td><td>MARK_switch</td><td>byte</td><td>Using of mark input signal to trigger specific GNSS raw receiver data (see section 5.6).
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Disabled</td><td>0x00</td></tr>
<tr><td>Enabled</td><td>0x01</td></tr>
</table>
By default it is disabled.</td></tr>
<tr><td>MARK_polarity</td><td>byte</td><td>Specifies mark signal polarity.
<table>
<tr><th>Polarity</th><th>Value</th></tr>
<tr><td>Negative</td><td>0x00</td></tr>
<tr><td>Positive</td><td>0x01</td></tr>
</table>
By default it has negative polarity.</td></tr>
<tr><td>MARK_timebias</td><td>word</td><td>An offset, in nanoseconds, to be applied to the time the mark input pulse occurs.
Default value is zero offset.</td></tr>
<tr><td>MARK_timeguard</td><td>word</td><td>A time period, in milliseconds, during which subsequent pulses after an initial pulse are ignored.
Default value is 4 milliseconds.
Note: minimum value is 2 milliseconds.</td></tr>
<tr><td rowspan="2">0x51*</td><td>PPS_switch</td><td>byte,</td><td>Allow the pulse-per-second (PPS) signal generated by GNSS receiver for data synchronization with other devices (see section 5.5).
<table>
<tr><th>State</th><th>Value</th></tr>
<tr><td>Enabled output</td><td>0x00</td></tr>
<tr><td>Disabled</td><td>0x01</td></tr>
</table>
By default PPS is enabled.</td></tr>
<tr><td>PPS_polarity</td><td>byte</td><td>Specifies polarity of the PPS pulse.
<table>
<tr><th>Polarity</th><th>Value</th></tr>
<tr><td>Negative</td><td>0x00</td></tr>
<tr><td>Positive</td><td>0x01</td></tr>
</table>
By default it is negative.</td></tr>
</table>

<table>
<tr><td rowspan="2"></td><td>PPS_period</td><td>byte</td><td>Sets period of the PPS signal in seconds.
<table>
<tr><th>Period</th><th>Value</th></tr>
<tr><td>0.05</td><td>0x01</td></tr>
<tr><td>0.1</td><td>0x02</td></tr>
<tr><td>0.2</td><td>0x03</td></tr>
<tr><td>0.25</td><td>0x04</td></tr>
<tr><td>0.5</td><td>0x05</td></tr>
<tr><td>1</td><td>0x06</td></tr>
<tr><td>2</td><td>0x07</td></tr>
<tr><td>3</td><td>0x08</td></tr>
<tr><td>4</td><td>0x09</td></tr>
<tr><td>5</td><td>0x0A</td></tr>
<tr><td>6</td><td>0x0B</td></tr>
<tr><td>7</td><td>0x0C</td></tr>
<tr><td>8</td><td>0x0D</td></tr>
<tr><td>9</td><td>0x0E</td></tr>
<tr><td>10</td><td>0x0F</td></tr>
<tr><td>11</td><td>0x10</td></tr>
<tr><td>12</td><td>0x11</td></tr>
<tr><td>13</td><td>0x12</td></tr>
<tr><td>14</td><td>0x13</td></tr>
<tr><td>15</td><td>0x14</td></tr>
<tr><td>16</td><td>0x15</td></tr>
<tr><td>17</td><td>0x16</td></tr>
<tr><td>18</td><td>0x17</td></tr>
<tr><td>19</td><td>0x18</td></tr>
<tr><td>20</td><td>0x19</td></tr>
</table>
By Default period is 1 second.</td></tr>
<tr><td>PPS_pulse_width</td><td>4 byte unsigned integer</td><td>Pulse width of the PPS signal in microseconds.
By default it is 1000 microseconds.</td></tr>
<tr><td colspan="2">0x52 – 0x56</td><td colspan="2">Reserved</td></tr>
<tr><th colspan="4">2. Correction options</th></tr>
<tr><th colspan="4">2.1. Settings</th></tr>
<tr><td>0x57</td><td>time_INS_max</td><td>word</td><td>The maximum time of autonomous INS operation at absence of GNSS data (GNSS outage) in seconds.
Default value is set to 120 seconds.</td></tr>
<tr><td>0x58</td><td>H_corr_type</td><td>byte</td><td>Data used for INS heading correction.
<table>
<tr><th>Data type</th><th>Value</th></tr>
<tr><td>Magnetometers</td><td>0x00</td></tr>
<tr><td>AHRS</td><td>0x01</td></tr>
<tr><td>GNSS track</td><td>0x02</td></tr>
<tr><td>Dual GNSS</td><td>0x03</td></tr>
<tr><td>Combined</td><td>0x04</td></tr>
<tr><td>Inertial</td><td>0x05</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>0x59</td><td>Use_mags</td><td>byte</td><td>Enables or disables using of magnetometers for INS heading correction.<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table></td></tr>
<tr><td>0x5A</td><td>Vh_cut</td><td>byte</td><td>Threshold for horizontal speed of carrier object, m/s*10, to allow using of the GNSS track angle for INS correction,
Default value is 1.2 m/s.</td></tr>
<tr><td rowspan="4">0x5B</td><td>ZUPT</td><td>byte</td><td>Allow use “Zero Velocity Update” (ZUPT) option to reduce INS accumulated errors when stop of the carrier object is detected (see section 6.6.5).<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>By default “ZUPT” is disabled.</td></tr>
<tr><td>Vh_zupt</td><td>byte</td><td>Minimum horizontal speed of carrier object below which stop of the carrier object is detected, in m/s*100.
Default value is 0.15 m/s.</td></tr>
<tr><td>Vv_zupt</td><td>byte</td><td>Minimum vertical speed of carrier object below which stop of the carrier object is detected, in m/s*100.
Default value is 0.15 m/s.</td></tr>
<tr><td>T_zupt</td><td>byte</td><td>Time constant of low-pass filter for horizontal and vertical speed used for detection of the carrier object stop, in s*100.
Default value is 0.10 s.</td></tr>
<tr><td>0x5C</td><td>Tunnel_guide</td><td>byte</td><td>Enables “Tunnel guide” feature for INS operation on a car during long-time GNSS outage (see section 6.6.5).<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>Default setting is disabled.</td></tr>
<tr><td>0x5D</td><td>Round_time_stamp</td><td>byte</td><td>Round time stamps (if enabled) or accurate time stamps (if disabled) for all INS output data.<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>Default setting is “Round time stamps”.</td></tr>
<tr><td rowspan="2">0x5E</td><td>Catapult_start</td><td>byte</td><td>Allow the special INS algorithm for the UAV with catapult launching.<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>Enabled</td><td>0x01</td></tr></table>By default it is disabled.</td></tr>
<tr><td>A_catapult_cut</td><td>byte</td><td>The threshold of acceleration, in g*10, at catapult launching. INS uses specific correction from GNSS data during time when acceleration exceeds this threshold.</td></tr>
<tr><td>0x5F</td><td></td><td colspan="2">Reserved</td></tr>
</table>

0x60	STD_Vh_zupt	word	Noise standard deviation of horizontal virtual velocity measurements that used in ZUPT mode, in **m/s*1000**. **Default** value is 0.1 m/s.
	STD_Vv_zupt	word	Noise standard deviation of vertical virtual velocity measurements that used in ZUPT mode, in **m/s*1000**. **Default** value is 0.1 m/s.
0x61		Reserved	
0x62	Gyro_1_zupt	float	Thresholds for filtered gyros signal that are used to detect stop of a carrier object, in **deg/s**. **Default** value both of them is 0.05 deg/s.
	Gyro_2_zupt	float	
	Tgyr_1_zupt	float	Time constants of filters for gyros which are used to detect stops of a carrier object, in **seconds**. **Default** value for Tgyr_1_zupt is 0.16 s, for Tgyr_2_zupt is 0.2 s.
	Tgyr_2_zupt	float	
0x63	Accel_zupt	float	Thresholds for filtered accelerometers signal that are used to detect stop of a carrier object, in **m/s^2**. **Default** value is 0.15 m/s^2.
	Tacc_1_zupt	float	Time constants of filters for accelerometers which are used to detect stops of a carrier object, in **seconds**. **Default** value for Tacc_1_zupt is 2.0 s, for Tacc_2_zupt is 0.4 s.
	Tacc_2_zupt	float	
0x64	Gyro_3_zupt	float	Threshold for filtered gyros signal that allows INS to keep the "stop" mode at carrier object jerks or oscillations, in **deg/s**. **Default** value is 0.05 deg/s.
	Freeze_PV	byte	Switch to freeze INS position and to zero INS velocity output at stop detected (ZUPT mode).
0x65 – 0x66		Reserved	
3. Sensors options			
0x77	Sensors_output_var	byte	Type of gyros, accelerometers, and magnetometers output data (see Appendix E). **Data type** / **Value**: Instant angular rate and acceleration – 0x00 Average angular rate and acceleration – 0x01 Incremental Angle and Incremental Velocity – 0x02 By **default** it is set to "Incremental angle and incremental velocity".
0x78 – 0x7A		Reserved	
4. Swaying compensation options			
0x7B	INS_mounting_lever	3*sword	Lever of the INS mounting relative to the center of the object Swaying (usually this is object center of gravity). The lever must be set in the carrier object axes – on the right, forward and up, in **m*100**.
0x7C – 0x7D		Reserved	
5. Magnetometers calibration options			

<table>
<tr><td>0x7E</td><td>InitClbType</td><td>byte</td><td>Specifies set of hard/soft iron calibration parameters the INS starts with (see section 6.8.14).
<table><tr><th>Type of calibration</th><th>Value</th></tr><tr><td>Last INS Clb</td><td>0x00</td></tr><tr><td>Factory Clb</td><td>0x01</td></tr><tr><td>2D-2T, 3D,VG3D, 2D Clb</td><td>0x02</td></tr><tr><td>Previous Auto Clb</td><td>0x04</td></tr></table>Usually, the parameter is set automatically after the last calibration performed.</td></tr>
<tr><td rowspan="3">0x7F</td><td>ClbMagDisp_Threshold</td><td>long</td><td>Threshold for dispersion of magnetometers data to detect INS immobility and to remove appropriate data from calibration procedure. This threshold is set in in nT^2.
Default value is 1000 nT^2.</td></tr>
<tr><td>ClbInclination_Threshold</td><td>word</td><td>Acceptable pitch and roll deviation from their median in the 2D and 2D-2T calibration run. Parameter is in deg*100. INS data over this threshold are not used at calculation of calibration parameters.
Default value is 1.5 degrees.</td></tr>
<tr><td>ClbSuccess_Threshold</td><td>word</td><td>Acceptable value of magnetic field calibration error to have successful result of the 3D calibration, in nT.
Default value is 2500 nT.</td></tr>
<tr><td>0x80</td><td>T_Hfilter</td><td>byte</td><td>Time constant of low-pass filter for measured horizontal component of the Earth magnetic field, in s*10.
Default value is 0.6 seconds.</td></tr>
<tr><td>0x81</td><td>PR_threshold</td><td>byte</td><td>The value, in deg, used for detection of control circuit in the 3D calibration procedure.
Default value is 20 degrees.</td></tr>
<tr><td rowspan="2">0x82</td><td>Matrix Tm_c</td><td>9*float</td><td>Matrix for soft iron correction (3×3) by rows.</td></tr>
<tr><td>Matrix Hm_0</td><td>3*float</td><td>Matrix for hard iron correction (3×1).</td></tr>
<tr><td>0x84</td><td>Mag_auto_clb</td><td>byte</td><td>Allows automatic calibration of magnetometers (see section 6.8.13).
<table><tr><th>State</th><th>Value</th></tr><tr><td>Disabled</td><td>0x00</td></tr><tr><td>2D Clb</td><td>0x01</td></tr><tr><td>3D Clb</td><td>0x02</td></tr></table>By default, automatic calibration is disabled.</td></tr>
<tr><td colspan="2">0x85 – 0x88</td><td colspan="2">Reserved</td></tr>
</table>

Note: If values of the parameters which are marked by "*" sign in Parameter Code column were changed then it is necessary to power off/on the INS unit so changes get applied.

6.4. Control of the GNSS receiver

6.4.1. GNSS receiver parameters

User can get information about the GNSS receiver model, serial number, firmware version and data rate using GetDevInfo command (see section 6.3.6).

There are two ways to configure GNSS receiver parameters, via INS GUI (see INS GUI User's Manual, section "10.2. Control of the GNSS receiver" for details), and via an appropriate command since INS firmware version 3.3.0.1, (see section 6.3.14).

The next parameters of the GNSS receiver can be changed (see Table 6.80, portions "1.2. GNSS receiver" and "1.6. Triggers"):

- **GNSS_data_rate (0x12)** – specifies the maximum measurement rate for the GNSS receiver in Hertz. Default value is determined by the maximum possible measurement rate of the GNSS receiver.
- **Altitude_var (0x13) –** variant of altitude output – above mean sea level (MSL) or above WGS84 ellipsoid.

<u>GNSS port 2</u>

- **GNSS_COM2_data_set (0x14)** – allows to choose GNSS data set for output through COM port 2.
- **GNSS_COM2_data_rate (0x14)** – Frequency of COM2 port GNSS data output.
- **GNSS_COM2_bps (0x14)** – sets baud rate of COM2 which outputs GNSS data.

<u>NMEA set</u>

- **NMEA_set_COM2 (0x15)** parameter allows to set needed NMEA messages for output through COM port 2 with possibility to choose individual frequency by **NMEA_COM2_data_rate** command.

<u>GNSS port 3</u>

- **GNSS_COM3_data_set (0x16)** – allows to choose GNSS data set for output through the COM port 3.
- **GNSS_COM3_data_rate (0x16)** – specifies the GNSS data frequency.
- **GNSS_com3_bps (0x16)** – sets baud rate of COM3 which provides input of the GNSS corrections.

<u>GNSS correction</u>

- **GNSS_corr_type (0x17)** – specifies type of GNSS correction which should be used:
  - **No correction** – no GNSS corrections will be used;

- **AUTO** – the GNSS receiver uses the best available source of corrections: RTK, DGPS, TerraStar or SBAS;
- **SBAS** – correction data from Satellite Based Augmentation Systems (SBAS) will be used;
- **DGPS** – transmitted from a base station Differential GPS (DGPS) correction data will be used;
- **TerraStar** – TerraStar correction service will be used.

- **SBAScontrol (0x18)** – specifies type of SBAS correction.
- **GNSS_corr_format (0x19)** – specifies format of differential correction data.
- **TerraStarType (0x1B)** – specifies type of TerraStar correction.
- **Enable_PDPFILTER (0x1C)** – enables the Pseudorange/Delta-Phase (PDP) filter in the NovAtel receiver to smooth a jumpy GNSS position and to bridge outages in satellite coverage. Note enabled PDP filter can make less accurate the GNSS position.

Mark inputs control

- **MARK_switch (0x50)** – allows to control the processing of the mark input signal through MARK IN pin of the main INS connector (see section "5.6. Mark input description).
- **MARK_polarity (0x50)** – specifies polarity of the pulse at a mark input.
- **MARK_timebias (0x50)** – sets an offset, in nanoseconds, to be applied to the time the mark input pulse occurs.
- **MARK_timeguard (0x50)** – sets a time period, in milliseconds, during which subsequent pulses after an initial pulse are ignored.

PPS control

- **PPS_switch (0x51)** – allows to output PPS signal.
- **PPS_polarity (0x51)** – specifies polarity of the PPS pulse (see section "5.5. PPS description").
- **PPS_period (0x51)** – sets period of the pulse in seconds.
- **PPS_pulse_width (0x51)** – sets pulse width of the PPS signal in microseconds.

Important note: It is necessary to power off / on the INS after changing any of GNSS receiver parameters to restart the GNSS receiver with new settings.

6.4.2. Control of GNSS receiver model

The Inertial Labs™ INS contains the NovAtel GNSS receiver inside. NovAtel uses the term "models" to refer to and control different levels of functionality in the GNSS receiver firmware. For example, user can purchase INS with the base model of the GNSS receiver which has an L1 only capability. At a later time, he can easy upgrade this receiver to a more feature intensive model, like L1/L2 dual-frequency. All that is required to upgrade is an authorization code for the higher model and the INS GUI to enter this code to the receiver. Reloading of the INS or GNSS receiver firmware or returning the INS for service to upgrade the model is not required.

See http://www.novatel.com/assets/Documents/Papers/NovAtelModels.pdf for information about available models for OEM615 NovAtel GNSS receiver.

User can perform next options by using INS GUI:

- add new model to the GNSS receiver
- choose one of saved models
- remove model from the GNSS receiver

See section “10.2.2. Control of GNSS receiver model” in the INS GUI User Manual.

6.4.3. Accelerated start of the NovAtel GNSS receiver with TerraStar service at known position

The **PPPSEED** command controls the seeding of the Precise Point Positioning (PPP) filter in the NovAtel GNSS receiver when TerraStar licensed based corrections are available. Accurate position seeding can accelerate PPP convergence when the antenna does not move between power-down and power-up.

User can send the next commands to INS unit (within idle or operation mode):

- **PPPSEED_STORE** – store the current PPP position in nonvolatile memory for use as a future seed;
- **PPPSEED_RESTORE** – retrieve and apply a seed position that was previously saved in nonvolatile memory via the PPPSEED_STORE command;
- **PPPSEED_CLEAR** – resets the stored seed.

All these commands have the byte structure shown in the Table 6.2. Payload for all commands has length 1 byte and contains code of the command. See Appendix C for exact structure of these commands. Also, it is supported by the INS GUI since version 2.0.49.336 from 2019-07-17 (see INS GUI User Manual, section “10.2.3. Accelerated start of the NovAtel GNSS receiver with TerraStar service at known position”).

6.5. Altitude calculation

At its operation the Inertial Labs^{TM} INS calculates position using its sensors data with correction from the onboard GNSS receiver. Also, for altitude calculation the INS can use correction from the onboard pressure sensor.

In practice the GNSS altitude data are much less accurate than the horizontal position (because of high vertical dilution of precision). Using a static pressure sensor (barometer), as an aiding sensor for the altitude, increases the vertical accuracy. Though the relation between altitude and pressure is dependent on many factors, the most important is the "weather".

The Inertial Labs^{TM} INS allows three variants of the altitude correction that depends on the Baro_altimeter switch:

a) correction by altitude and vertical velocity provided by GNSS data (Baro_altimeter=0);
b) correction by barometric altitude calculated using pressure sensor data and vertical velocity provided by GNSS data (Baro_altimeter=1 – primary altitude sensor);
c) altitude correction is provided by GNSS data if they are valid, otherwise barometric altitude calculated using pressure sensor data is used (Baro_altimeter=2 – secondary altitude sensor).

The default value is Baro_ altimeter=0. User can change this value using the LoadINSpar command (see Table 6.59, byte #58) or using the INS GUI (that is easier).

Important note: To measure barometric altitude the pressure sensor in the INS must have access to the ambient external pressure. Also the pressure sensor must not be exposed to high speed air streams. So if the INS is installed inside a pressurized cabin or outside the high-speed object, please set Baro_altimeter=0 to switch to the GNSS altitude for INS correction.

Note in all variants of the INS altitude correction, the initial altitude is equal to altitude provided by the GNSS receiver if it has solution. If GNSS data are not available, then the initial altitude is equal to its value stored in the INS nonvolatile memory. Initial altitude can be changed there using the LoadINSpar command (see Table 6.59, bytes #16-19) or using the INS GUI (that is easier).

INS can output altitude above mean sea level or above WGS84 ellipsoid. This choice is implemented in INS firmware since version 2.8.3.6 and it is supported by the INS GUI since version 2.0.39.184 from 2017-05-19. See User's Manual rev.2.6

and higher, section «4.2.2. “GNSS receiver” tab of “Devices options…” window» for details.

6.6. INS operation in GNSS-denied environment

GNSS aided INS provides accurate and reliable position, velocity and orientation. At GNSS outage, the INS accuracy degrades over time quickly due to sensors error. It is still possible to keep INS accuracy within acceptable limits by using aiding data from other external sensors.

Inertial Labs INS has different means to improve its operation in GNSS-denied conditions:

1) Using external sensors data input through the main COM1 port (see section "6.6.4 Aiding data input through the main COM port");

2) Using odometer data (see sections "6.6.1. Odometer data input using RS-232 interface", "6.6.2. Odometer data input from encoder (wheel speed sensor)" and “6.6.3. Odometer data input from CAN BUS”;

3) Using “Tunnel Guide” feature and ZUPT (zero velocity update) option if external data are not available (see section"6.6.5. ZUPT option and Tunnel guide feature").

Notes:
1. INS algorithm of operation with odometer includes “Tunnel Guide” feature and ZUPT, so it is not necessary to switch on these options separately.
2. Using external sensors data input through the main COM1 port can be supplemented with “Tunnel Guide” feature and ZUPT option if this makes sense (see section "6.6.5. ZUPT option and Tunnel guide feature").

6.6.1. Odometer data input using RS-232 interface

The Inertial Labs™ INS COM4 port can be used for input data from the odometer for dead reckoning. INS supports the OBDII odometer interface at using the appropriate cable OBDLink S, https://www.scantool.net/obdlink-s/.

There are two ways to configure OBDII odometer, via INS GUI since version 2.0.46.279 from 2018-07-26 (see INS GUI User’s Manual, section “10.7.1. INS operation with OBDII odometer” for details), and via an appropriate command since INS firmware version 3.3.0.1 (see section 6.3.14).

The next parameters should be set to configure OBDII odometer (see “1.5 CAN/COM4” portion in Table 6.80):

- **Odometer_type (0x29)** – set “OBDII odometer” type.

- **COM4_data_var (0x44)** – allows to choose variant of COM4 port use. To use odometer data, it is necessary to set COM4 port to "OBDII odometer" (COM4_data_var = 0x01).
- **COM4_data_rate (0x44)** – specifies the frequency of the odometer data input in Hertz. The default value is set to 20Hz.
- **COM4_bps (0x44)** – sets the baud rate of COM4 which provides input of the odometer data

6.6.2. Odometer data input from encoder (wheel speed sensor)

The Inertial Labs™ INS unit can be factory configured to receive pulse/bi-phase signals from encoder for dead reckoning. Using signal lines originally assigned to COM4 port. See section "5.4. Connection of encoder-based odometer to INS" for details.

There are two ways to configure encoder-based odometer parameters, via INS GUI (see INS GUI User's Manual, section "10.7.2. INS operation with encoder-based odometer (wheel speed sensor)" for details), and via an appropriate command since INS firmware version 3.3.0.1 (see section 6.3.14).

The next parameters should be set to configure encoder-based odometer (see "1.4 External sensors" portion in Table 6.80):

- **Odometer type (0x29)** sets type of odometer. Please choose "Encoder-based" type.
- **Pulse_length (0x2A)** is the distance in meters between low to high transitions of the encoder signal;
- **ZUPT_threshold (0x2A)** is threshold for horizontal speed measured with odometer for stop detection in odometer ZUPT algorithm;
- **Odom_offset (0x2B)** is the lever arm measured from the accelerometer mass-center of the INS unit to the point at which the vehicle's tire makes contact with the road in the vehicle co-ordinate frame, in the right, forward and vertical directions;
- **STD_Vh_ext (0x2F)** is standard deviation of the odometer noise. Usually this value is 0.1 m/s, but it depends on encoder.

The pulse length can be calculated inside INS if in appropriate command (code 0x29, see Table 6.80) it is left zero, but two next parameters are set accurately:

- **Wheel_circ** – wheel circumference where encoder is installed;
- **Ticks_per_rev** – number of ticks per revolution.

Also, the pulse length can be calculated manually using formula:

Pulse length = pi*D/n = C/n

where D – is wheel diameter (in meters); C – is wheel circumference (in meters); n – is number of pulses per revolution.

The pulse length should be set accurately because INS position calculation at GNSS outage is highly depends on pulse length accuracy. On the other hand, the pulse length can be set approximately or even set to zero if the odometer calibration procedure will be performed before ordinary operation of INS with odometer (see section 6.6.2.1).

6.6.2.1. Calibration of encoder-based odometer (wheel speed sensor)

The aim of the odometer calibration is accurate estimation of the pulse length.

Start INS ordinary operation. Start vehicle run. The odometer calibration can be started at any time after INS started calculation of valid position, velocity and orientation.

Send the **Start_Odom_clb** command to start the odometer calibration:

AA 55 00 00 07 00 28 2F 00

Note: these are hexadecimal numbers but not ASCII text symbols.

After receiving of this command INS starts accumulation of odometer data for its calibration. At this the bit #7 is set to 1 in INS status word USW for indication of data accumulation process (see Fig. 6.3 and section 6.10).

It is desirable the vehicle runs approximately straight line during the odometer calibration. Distance of the run should be not less than 1 km. During this calibration run the GNSS outage is allowed, but it should be not more than 10 seconds.

To finish the odometer calibration, send the **Stop_Odom_clb** command:

AA 55 00 00 07 00 29 30 00

Note: these are hexadecimal numbers but not ASCII text symbols.

After receiving of this command INS stops data accumulation for the odometer calibration and sets the USW bit #7 to 0 (see Fig. 6.3).

Then INS starts calculations for the odometer pulse length and sets USW bit #7 to 1 again. In the end of these calculations, if calibration is successful, INS calculates the pulse length. Also INS sets the USW bit #15 to 1 to inform the host system that the odometer calibration is performed and successful (see Fig. 6.3). If this calibration was estimated as unsuccessful then the USW bit #15 is set to 0.

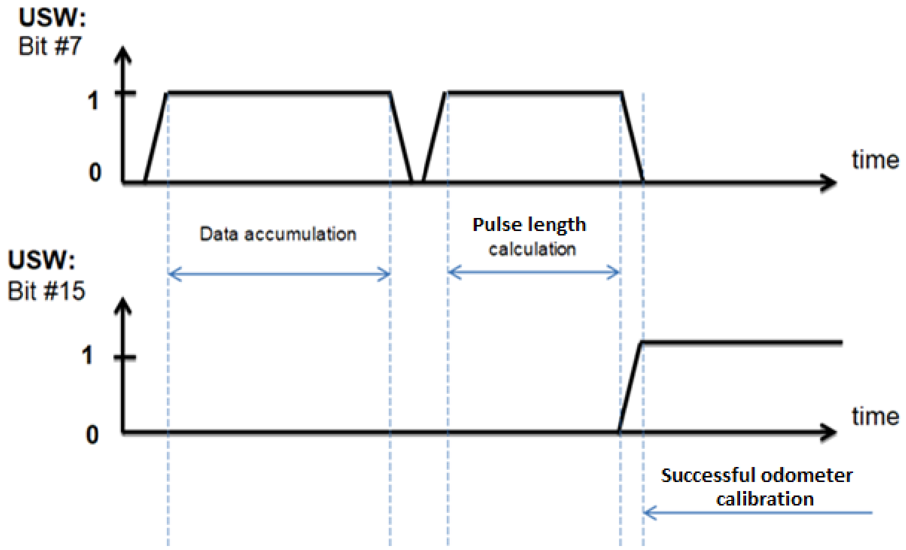


Fig. 6.3. The diagram of indication of the odometer calibration process in the USW (Unit Status Word, see section 6.10)

Calculated pulse length will be used immediately for operation with odometer. But the pulse length will be stored to INS nonvolatile memory only after INS receives the Stop command to finish INS work in operating mode (see section 6.3.3).

During INS run you can perform new odometer calibration several times without INS stop, without vehicle stop. The last calculated pulse length will be written to the INS flash memory after INS stop.

At the next INS run this pulse length will be used for operation with odometer.

Note during all steps of the odometer calibration the INS unit continues calculation of navigation data and their output.

Calculated value of the pulse length can be checked using INS GUI in “Devices option”, “External sensors” tab or using ReadINSpar_RAM command (see section 6.3.14).

6.6.3. Odometer data input from CAN BUS

Once user has access to the vehicle's CAN BUS and knows which CAN messages indicate the velocity of the car, then it is possible to connect the CAN BUS line to the INS unit and to use CAN data as a kind of odometer.

There are two ways to configure CAN BUS based odometer, via INS GUI since version 2.0.50.349 from 2019-10-09 (see INS GUI User Manual, section "10.7.2. INS operation with CAN BUS based odometer" for details), and via an appropriate command since INS firmware version 3.4.1.4 (see section 6.3.14).

The next parameters should be set to configure CAN BUS based odometer (see "1.5 CAN/COM4" portion in Table 6.80):

- **CAN_bps (0x46)** – is CAN baud rate;

Signal 1 describes CAN message which contains velocity data

- **CAN_Inp_Sig1_MessType (0x47)** – is a type of CAN message;
- **CAN_Inp_Sig1_MesID (0x47)**– is CAN message identifier;
- **CAN_Inp_Sig1_DataType (0x47)** – is data type of signal 1 in CAN message;
- **CAN_Inp_Sig1_Startbit (0x47)** – is the least significant bit of signal 1 counted from the start of the message data;
- **CAN_Inp_Sig1_Length (0x47)** – is number of bits which signal 1 occupies in the message;
- **CAN_Inp_Sig1_ByteOrder (0x47)** – is a message byte order;
- **CAN_Inp_Sig1_ValueType (0x47)** – specifies signed or unsigned data type;
- **CAN_Inp_Sig1_Factor (0x47)** and **CAN_Inp_Sig1_Offset (0x47)** define the linear conversion rule to convert the signal raw value into the signal's physical value (in km/h) and vice versa:
  physical_value = raw_value * factor + offset;
  raw_value = (physical_value – offset)/ factor;

Signal 2 can be used to describe CAN message with additional information about velocity (if it exists – like decimal velocity values). It has the same configuration parameters as Signal 1.

- **CAN_Inp_Sig2_MessType (0x48);**
- **CAN_Inp_Sig2_MesID (0x48);**
- **CAN_Inp_Sig2_DataType (0x48);**
- **CAN_Inp_Sig2_Startbit (0x48);**
- **CAN_Inp_Sig2_Length (0x48);**
- **CAN_Inp_Sig2_ByteOrder (0x48);**
- **CAN_Inp_Sig2_ValueType (0x48);**

- **CAN_Inp_Sig2_Factor (0x48);**
- **CAN_Inp_Sig2_Offset (0x48);**

Signal 3 can be used to describe CAN message with information about velocity sign if it is absent in **Signal 1**. Usually the velocity sign can be defined by transmission actual gear. Signal 3 has the same first seven configuration parameters as Signal 1, and the last parameter is new.

- **CAN_Inp_Sig3_MessType (0x49);**
- **CAN_Inp_Sig3_MesID (0x49);**
- **CAN_Inp_Sig3_DataType (0x49);**
- **CAN_Inp_Sig3_Startbit (0x49);**
- **CAN_Inp_Sig3_Length (0x49);**
- **CAN_Inp_Sig3_ByteOrder (0x49);**
- **CAN_Inp_Sig3_ValueType (0x49);**
- **CAN_Inp_Sig3_ReverseGear (0x49)** – value of the signal that corresponds to the reverse gear (reverse vehicle motion).

Note: Each kind of car has unique CAN messages in CAN BUS line. As an example of parameters configuration of CAN BUS based odometer for Hyundai Tucson TL car shown in Table 6.81.

Table 6.81. Example of CAN BUS based odometer configuration for Hyundai Tucson TL

Parameters	Value	Description (see Table 6.80)
	Signal 1	
CAN_Inp_Sig1_MessType	0x01	CAN 2.0A type
CAN_Inp_Sig1_MesID	0x112	CAN message identifier for CAN 2.0A type
CAN_Inp_Sig1_DataType	0x01	Velocity data type
CAN_Inp_Sig1_Startbit	0x10	16 decimal
CAN_Inp_Sig1_Length	0x08	8 decimal
CAN_Inp_Sig1_ByteOrder	0x00	little-endian byte order
CAN_Inp_Sig1_ValueType	0x00	Unsigned value type
CAN_Inp_Sig1_Factor	0x0000803F	1.0 decimal
CAN_Inp_Sig1_Offset	0x00000000	0.0 decimal
	Signal 2	
CAN_Inp_Sig2_MessType	0x01	CAN 2.0A type
CAN_Inp_Sig2_MesID	0x112	CAN message identifier for CAN 2.0A type
CAN_Inp_Sig2_DataType	0x01	Velocity decimal data type
CAN_Inp_Sig2_Startbit	0x38	56 decimal
CAN_Inp_Sig2_Length	0x08	8 decimal
CAN_Inp_Sig2_ByteOrder	0x00	little-endian byte order
CAN_Inp_Sig2_ValueType	0x00	Unsigned value type
CAN_Inp_Sig2_Factor	0x0000003C	0.0078125 decimal
CAN_Inp_Sig2_Offset	0x00000000	0.0 decimal
	Signal 3	
CAN_Inp_Sig3_MessType	0x01	CAN 2.0A type
CAN_Inp_Sig3_MesID	0x111	CAN message identifier for CAN 2.0A type
CAN_Inp_Sig3_DataType	0x01	Transmission
CAN_Inp_Sig3_Startbit	0x08	8 decimal
CAN_Inp_Sig3_Length	0x04	4 decimal
CAN_Inp_Sig3_ByteOrder	0x00	little-endian byte order
CAN_Inp_Sig3_ValueType	0x00	Unsigned value type
CAN_Inp_Sig3_ReverseGear	0x07	7 decimal

6.6.4. Aiding data input through the main COM port

Since INS firmware version 2.8.2.0 it is possible to send external aiding data to INS unit using the main COM1 port during INS ordinary operation when INS outputs data through the same COM1 port.

The Inertial Labs™ INS allows aiding data input when COM1_Aiding_data parameter is set to 1. The default value is COM1_Aiding_data=0 (no aiding data input). User can change this value, via INS GUI (see INS GUI User's Manual, section "4.2.4 "External sensors" tab of "Devices options""), and via the command 0x2C COM1_Aiding_data (see Table 6.80) since INS firmware version 3.3.0.1.

Structure of the external aiding data blocks corresponds to the Table 6.2 with payload shown in the Table 6.82.

Table 6.82. Aiding data payload structure

Field	Offset in payload, bytes	Size, bytes	Value
Meas Num	0	1	Number of measurements present in the payload (M)
Meas List	1	M	List of measurement types, one per byte, refer to Table 6.83 for values
Meas Data 1	M+1	Variable, depends on measurement type, refer to Table 6.83 for sizes	Data according to measurement structure, refer to Table 6.83 for structures
Meas Data 2	Variable, depends on Meas Data 1 size	Variable, depends on measurement type, refer to Table 6.83 for sizes	Data according to measurement structure, refer to Table 6.83 for structures
…	…	…	…
Meas Data M	Variable, depends on preceding data	Variable, depends on measurement type, refer to Table 6.83 for sizes	Data according to measurement structure, refer to Table 6.83 for structures

Table 6.83. Aiding data measurements structure (to be further expanded)

Type	Semantic	Size, bytes	Structure	
0x00	Reserved	TBD	TBD	
0x01	Odometer	4	signed long	Accumulated distance in mm
0x02	Air speed	2	signed short	Air speed in 0.01 kt
0x03	Wind data	8	signed short	North wind component in 0.01 kt
			signed short	East wind component in 0.01 kt
			unsigned short	North wind STD in 0.01 kt
			unsigned short	East wind STD in 0.01 kt
0x04	External position	20	signed long	Latitude external in signed deg*1.0e7
			signed long	Longitude external in signed deg*1.0e7
			signed long	Altitude external above MSL in mm
			unsigned short	Latitude external STD in 0.01 m
			unsigned short	Longitude external STD in 0.01 m
			unsigned short	Altitude external STD in 0.01 m
			unsigned short	External position latency in msec
0x05	Doppler shift from locator	16	signed long	Locator latitude in signed deg*1.0e7
			signed long	Locator longitude in signed deg*1.0e7
			signed long	Locator altitude above MSL in mm
			signed short	Doppler shift in cm/sec
			unsigned short	Doppler shift STD in cm/sec
0x06	Heading external	6	unsigned short	Heading external in 0.01 deg
			unsigned short	Heading external STD in 0.01 deg
			unsigned short	Heading external latency in msec
0x07	DVL data (depends on	24	signed long	Lateral velocity, 1000*m/sec
			signed long	Forward velocity, 1000*m/sec
			signed long	Vertical velocity, 1000*m/sec

	reference frame, see note #3)		unsigned short	Lateral velocity STD, 1000*m/sec
			unsigned short	Forward velocity STD, 1000*m/sec
			unsigned short	Vertical velocity STD, 1000*m/s
			unsigned short	Velocity latency, msec
			unsigned long	Reserveds
0x08	Sensors bias external	7	signed byte	Gyro bias X, $0.5*10^{4}$*deg/s
			signed byte	Gyro bias Y, $0.5*10^{4}$*deg/s
			signed byte	Gyro bias Z, $0.5*10^{4}$*deg/s
			signed byte	Accel bias X, $0.5*10^{5}$*g
			signed byte	Accel bias Y, $0.5*10^{5}$*g
			signed byte	Accel bias Z, $0.5*10^{5}$*g
			byte	Reserved
0x09	Pitch and Roll external	4	signed short	Pitch external in 0.01 deg
			signed short	Roll external in 0.01 deg

Notes:

1. MSL is mean sea level
2. All multi-byte integer values are LSB first.
3. DVL (Doppler Velocity Log) data can be presented in the vehicle axes (in the right, forward, vertical directions) or in navigational frame (East, North, Up) depending on "Ext.Vel. reference frame" value which can be set using INS GUI since version 2.0.48.320 from 2019-04-10 (see User's Manual rev.2.16 and higher, section "4.2.4.2. Using the main COM1 port to receive external aiding data" for details), or can be set via the command 0x34 DVL_reference_frame (see Table 6.80) since INS firmware version 3.3.0.1.
4. Reference point for odometer, air speed or DVL data can be set by Odom_offset parameter (code 0x2B) – see Table 6.80.

Currently INS can use all types of aiding data listed in Table 6.83 except "Doppler shift from locator".

Aiding data example 1 (air speed + wind data). The packet is:

AA 55 01 62 13 00 02 02 03 B9 0B E9 03 0B FE 34 00 2A 00 94 03

Table 6.84 explains data in this packet.

Table 6.84. Example 1

Field	Semantic	Decimal value	Hex value	Bytes (hex)
Header 0		n/a	n/a	AA
Header 1		n/a	n/a	55
Packet type	Data	n/a	n/a	01
Packet ID	Aiding data	n/a	n/a	62
Length	Total packet length (without header 0xAA 0x55)	19	0x0013	13 00
Meas Num	Number of measurements present (2)	2	0x02	02
Meas List	List of types (0x02, 0x03)	n/a	n/a	02 03
Meas 1 Field 1	Air speed of 30.01 kt	3001	0x0BB9	B9 0B
Meas 2 Field 1	North wind component of 10.01 kt	1001	0x03E9	E9 03

Meas 2 Field 2	East wind component of -5.01 kt	-501	0xFE0B	0B FE
Meas 2 Field 3	North wind STD of 0.52 kt	52	0x0034	34 00
Meas 2 Field 4	East wind STD of 0.42 kt	42	0x002A	2A 00
Checksum	Least significant 16 bits of the arithmetic sum of all preceding bytes except header.	916	0x0394	94 03

Aiding data example 2 (air speed + External position + Doppler shift from locator). The packet is:

AA 55 01 62 30 00 03 02 04 05 B9 0B 21 25 CF 17 B5 5E 04 D2 E2 51 02 00 7D 00 32 02 1E 04 2C 01 21 25 CF 17 B5 5E 04 D2 E2 51 02 00 E6 FB 32 00 0C 0D

Table 6.85 explains data in this packet.

Table 6.85. Example 2

Field	Semantic	Decimal value	Hex value	Bytes (hex)
Header 0		n/a	n/a	AA
Header 1		n/a	n/a	55
Packet type	Data	n/a	n/a	01
Packet ID	Aiding data	n/a	n/a	62
Length	Total packet length (without header 0xAA 0x55)	48	0x0030	30 00
Meas Num	Number of measurements present (2)	3	0x03	03
Meas List	List of types (0x02, 0x04, 0x05)	n/a	n/a	02 04 05
Meas 1 Field 1	Air speed of 30.01 kt	3001	0x0BB9	B9 0B
Meas 2 Field 1	Latitude external (39.9451425)	399451425	0x17CF2521	21 25 CF 17
Meas 2 Field 2	Longitude external (-77.1465547)	-771465547	0xD2045EB5	B5 5E 04 D2
Meas 2 Field 3	Altitude external above MSL (152.034 m)	152034	0x000251E2	E2 51 02 00
Meas 2 Field 4	Latitude external STD (1.25 m)	125	0x007D	7D 00
Meas 2 Field 5	Longitude external STD (5.62 m)	562	0x0232	32 02
Meas 2 Field 6	Altitude external STD in (10.54 m)	1054	0x041E	1E 04
Meas 2 Field 7	External position latency (300 msec)	300	0x012C	2C 01
Meas 3 Field 1	Locator latitude (39.9451425)	399451425	0x17CF2521	21 25 CF 17
Meas 3 Field 2	Locator longitude (-77.1465547)	-771465547	0xD2045EB5	B5 5E 04 D2
Meas 3 Field 3	Locator altitude above MSL (152.034 m)	152034	0x000251E2	E2 51 02 00
Meas 3 Field 4	Doppler shift (-10.5 m/sec)	-1050	0xFBE6	E6 FB
Meas 3 Field 5	Doppler shift STD (0.5 m/sec)	50	0x0032	32 00
Checksum	Least significant 16 bits of the arithmetic sum of all preceding bytes except header.	3340	0x0D0C	0C 0D

Aiding data example 3 (DVL data). The packet is:

AA 55 01 62 20 00 01 07 CB 0D 00 00 F1 16 00 00 4F 05 00 00 F6 00 60 01 62 00 FA 00 00 00 00 00 71 05

Table 6.86 explains data in this packet.

Table 6.86. Example 3

Field	Semantic	Decimal value	Hex value	Bytes (hex)
Header 0		n/a	n/a	AA
Header 1		n/a	n/a	55
Packet type	Data	n/a	n/a	01
Packet ID	Aiding data	n/a	n/a	62
Length	Total packet length (without header 0xAA 0x55)	32	0x0020	20 00
Meas Num	Number of measurements present (2)	1	0x01	01
Meas List	List of types (0x07)	n/a	n/a	07
Meas 1 Field 1	Lateral velocity (3.531 m/sec)	3531	0x00000DCB	CB 0D 00 00
Meas 1 Field 2	Forward velocity (5.873 m/sec)	5873	0x000016F1	F1 16 00 00
Meas 1 Field 3	Vertical velocity (1.359 m/sec)	1359	0x0000054F	4F 05 00 00
Meas 1 Field 4	Lateral velocity STD, (0.246 m/sec)	246	0x00F6	F6 00
Meas 1 Field 5	Forward velocity STD, (0.352 m/sec)	352	0x0160	60 01
Meas 1 Field 6	Vertical velocity STD, (0.098 m/sec)	98	0x0062	62 00
Meas 1 Field 7	Velocity latency (250 msec)	250	0x00FA	FA 00
Meas 1 Field 8	Reserved	0	0x00000000	00 00 00 00
Checksum	Least significant 16 bits of the arithmetic sum of all preceding bytes except header.	1542	0x0571	71 05

To control receiving of the external aiding data the special output data format “INS OPVTAD” is implemented since INS firmware version 2.8.2.0 – see section 6.2.6.

6.6.5. ZUPT option and Tunnel guide feature

If aiding data from external sensors are not available, then it is possible to use ZUPT (zero velocity update) option and “Tunnel guide” feature to reduce INS accumulated errors during long-time GNSS outage. This possibility is implemented in INS firmware version since 3.3.0.3.

There are two ways to enable ZUPT option and “Tunnel guide” feature, via INS GUI since version 2.0.49.325 released on 2019-05-23 (see INS GUI User’s Manual, section “4.3. Correction options” for details), and via the commands 0x5B ZUPT, 0x5C Tunnel_guide (see Table 6.80).

When "**Tunnel guide**" feature is used, to ensure proper Kalman Filter calculation, set noise of horizontal speed data "STD_Vh_ext (m/s)" to 1.5 and vertical speed data "STD_Vv_ext (m/s)" to 0.3 by sending relevant command 0x2F STD_Vh_ext, 0x30 STD_Vv_ext (see Table 6.80).

The next parameters should be set to configure ZUPT option (see "2.1 Settings" portion in Table 6.80):

- **ZUPT (0x58)** – allow use "Zero Velocity Update" option to reduce INS accumulated errors when stop of the carrier object is detected.
- thresholds for filtered gyros signal (**Gyro_1_zupt (0x62), Gyro_2_zupt (0x62)**), accelerometers signal (**Accel_zupt (0x63)**), horizontal and vertical speeds (**Vh_zupt (0x58), Vv_zupt (0x58)**), that are used to detect stop of a carrier object. The default value of every parameter is shown in Table 6.87.
- time constants (in seconds) of filters for gyroscopes (**Tgyr_1_zupt (0x62), Tgyr_2_zupt (0x62)**), accelerometers (**Tacc_1_zupt (0x63), Tacc_2_zupt (0x63)**), horizontal and vertical speeds (**T_zupt (0x58)**), which are used to detect stops of a carrier object. The default value of every parameter is shown in Table 6.87.
- threshold **Gyro_3_zupt (0x64)** for filtered gyros signal that allows INS to keep the "stop" mode at carrier object jerks or oscillations. For this purpose please set Gyro_3_zupt value greater than allowed angular rate of carrier object at its stop. Usually Gyro_3_zupt = 4 deg/s is enough threshold. To exclude this feature please set Gyro_3_zupt value the same as for Gyro_2_zupt (this is the default value of the parameter, see Table 6.87).
- **STD_Vh_zupt (0x60), STD_Vv_zupt (0x60)** – is noise standard deviation of horizontal and vertical virtual velocity measurements The default value of every parameter is shown in Table 6.87.
- switch **Freeze_PV (0x64)** allows to freeze INS position and to zero INS velocity output at stop detected (ZUPT mode). Be careful at such freezing because INS loses sensitivity to slow motion of the carrier object. The default value of Freeze_PV parameter is shown in Table 6.87.

Table 6.87. The default value of ZUPT parameters

Parameter	Default value
Thresholds	
Gyro_1_zupt	0.05 deg/s
Gyro_2_zupt	0.05 deg/s
Gyro_3_zupt	0.05 deg/s
Accel_zupt	0.15 m/s^2
Vh_zupt	0.15 m/s
Vv_zupt	0.15 m/s
Filter time constants	
Tgyr_1_zupt	0.16 s
Tgyr_2_zupt	0.20 s

Tacc_1_zupt	2.0 s
Tacc_2_zupt	0.4 s
T_zupt	0.1 s
Virtual measurement noise	
STD_Vh_zupt	0.1 m/s
STD_Vv_zupt	0.1 m/s
Switch for freezing INS output at stop	
Freeze_PV	false

Notes:
1. "Tunnel guide" fits only for fixed axle land vehicles like a car.
2. For certain applications where it is known the system will never be stationary, such as marine or airborne applications, ZUPT option should be disabled.
3. "Tunnel guide" feature and ZUPT can be added to aiding data input through the main COM1 port.
4. INS algorithm of operation with odometer already includes "Tunnel guide" and ZUPT, so it is not necessary to switch on these options separately.

6.6.6. Switching off/on the GNSS receiver data input to INS algorithm

The Inertial Labs INS allows switching off/on the GNSS receiver input to the INS algorithm during the INS operation since firmware version 2.6.0.6. This feature can be used to check INS operation at GNSS outage. Note physically the receiver is not disconnected and INS unit continues transferring GNSS data to output independently of the GNSS receiver input to the INS algorithm is switched on or off.

To switch the GNSS receiver off during the INS operation the command **GNSS_switch_off** (code 0x72 in the "Payload" field) should be sent.

To switch the GNSS receiver on during the INS operation the command **GNSS_switch_on** (code 0x71 in the "Payload" field) should be sent.

Exact structure of these commands is shown in Appendix C.

After receiving the GNSS_switch_off command the INS unit does not use GNSS data in its calculations. INS indicates this by setting number of satellites used in navigation solution, #SolnSVs, to zero in output data package.

Though GNSS_info2 field of INS output data package contains true information about GNSS data (see Table 6.7). If necessary number of GNSS satellites is available then "Solution status" of the GNSS_info2 field indicates "GNSS solution is computed" independent on sent command GNSS_switch_off or GNSS_switch_on.

6.7. Acceleration compensation at object swaying

It is possible to increase the INS orientation accuracy at the carrier object swaying if to compensate linear acceleration at place of the INS mounting. For this purpose, please set coordinates of the INS mounting relative to the center of the object swaying (usually this is object center of gravity). Reference point of the INS unit is its accelerometer mass-center (see Fig. 1.6, Fig. 1.7).

These coordinates are set in meters in such sequence of the object directions: right, forward, up. For this please use the LoadINSpar command (see Table 6.59, bytes #29-34) or the INS GUI (that is easier).

6.8. Calibration of the Inertial Labs™ INS on hard and soft iron

The Inertial Labs™ INS software allows compensation of hard and soft iron effects of the carrier object on the heading determination accuracy. For this purpose, field calibration of the INS magnetometers is provided (see Appendix A, The INS calibration). In case of using external magnetic compass OS3D-FG SAMC, the hard/soft iron calibration is the same as for INS magnetometers.

<u>Note</u>: INS does not require calibration of its magnetometers or SAFG ones on hard/soft iron if "Use_mags" switch is disabled in the "Settings" tab of «Correction options…» window of the INS GUI or is disabled by the command 0x59 Use_mags (see Table 6.80).

Inertial Labs utilizes several types of field calibration depending on the carrier object type:

- 3D calibration;
- 2D-2T calibration;
- 2D calibration;
- VG3D calibration (since firmware version 2.6.2.2);
- on-the-fly VG3D calibration (since firmware version 2.6.2.2);
- automatic 2D and 3D calibration (since firmware version 3.4.3.0).

The next commands are used for the INS calibration:

Start3DClb;	StopClbRun;	GetClbRes;
StartVG3DClb;	FinishClb;	StartVG3Dclb_flight;
Start2D2TClb;	AcceptClb;	StopVG3Dclb_flight;
Start2DClb;	ClearClb;	StopMagAutoClb.
StartClbRun;	ExitClb;	

All these commands have the byte structure shown in the Table 6.2. Payload for all commands has length 1 byte and contains code of the command. See Appendix C for examples of these commands.

6.8.1. Start3DClb command for INS 3D calibration

The **3D calibration** is designed for carrier objects that can operate in full heading, pitch and roll ranges. At this calibration the carrier object should be rotated in all these ranges.

To start the 3D calibration the host computer sends to the INS the Start3DClb command (code 0x23 in the “Payload” field) followed by message with block of parameters listed in the Table 6.88. This message have the byte structure shown in the Table 6.2, and should be sent without pause after sending the Start3DClb command.

Table 6.88. Payload of the message following after the Start3DClb, StartVG3DClb, Start2D2TClb and Start2DClb commands (the block of parameters loaded to the INS)

Byte	Parameter	Format	Length	*Note*
0-3	Reserved	**byte**	**4**	
4-5	Time of data accumulation in one run	**word**	**2**	Seconds
6-9	Latitude	**float**	**4**	Degrees
10-13	Longitude	**float**	**4**	Degrees
14-17	Altitude	**float**	**4**	Meters
18-21	Date (Year, Month, Day)	**float**	**4**	Year + (Month-1)/12 + (Day-1)/365

The INS calculates the check sum of received parameters and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

Then the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don’t move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out the block of the initial alignment data (see the Section 6.3.12) and starts data accumulation during time specified in message sent after the Start3DClb command (see the Table 6.88).

During the INS data accumulation, the object should be rotated in full azimuth, pitch and roll ranges. For example, the object is rotated in the horizon plane (the Z-axis is up) with periodical stops about each 90 degrees for tilting in pitch and roll. After full 360° rotation the object with the INS is turned over (the Z-axis is down) and the procedure described above should be repeated. During this calibration the range of pitch and roll angles changing must be as much as possible.

Note: there is estimation of 3D calibration quality in terms of possible INS heading accuracy. To allow this possibility it is necessary to include additional rotation of the INS with the carrier object in the horizon plane on about 360 degrees or more with pitch and roll near the level. Acceptable pitch and roll change can be set using INS GUI Software by the "Pitch/Roll threshold" parameter in the "Magnetometers calibration options" or by the command 0x81 PR_threshold (see Table 6.80).

After set accumulation time is reached or StopClbRun command is sent to the INS (see section 6.8.2 for details) the INS finishes data accumulation and calculates the calibration parameters.

After calculation of the calibration parameters that takes <0.5 seconds, the INS gives out message with the calibration results (see the Table 6.89) and it waits one of the next commands:

- the AcceptClb command (see section 6.8.3) to accept and save the calibration parameters (usually if the "Calibration success" byte in the INS message is nonzero and corresponds to satisfactory INS heading accuracy (see the Table 6.89 and Note below it));
- or the ExitClb command (see section 6.8.4 to exit from calibration procedure without accepting and saving its results (usually if the "Calibration success" byte in the INS message is equal to 0 or corresponds to not satisfactory INS heading accuracy (see the Table 6.89 and Note below it)).

The INS answers on these commands with checksum and goes to idle mode.

Table 6.89. Payload of the INS message after calibration completed

Byte	Parameter	Format	Length	*Note*
0	Type of calibration	**byte**	**1**	1 for 2D calibration; 2 for 2D-2T calibration; 3 for 3D calibration; 5 for VG3D calibration
1	Number of used calibration runs	**byte**	**1**	
2	Percent of used data points	**byte**	**1**	for 2D and 3D calibrations only
3	Calibration success	**byte**	**1**	0 – calibration is not successful >0 – calibration is successful (see Note below)
4-39	Matrix for soft iron correction	**float**	**9*4**	Matrix Tm_c (3×3) by rows
39-51	Matrix for hard iron correction	**float**	**3*4**	Matrix Hm_0 (3×1)

Note: there is estimation of the calibration quality as predicted INS heading accuracy. So nonzero value of byte #3 “Calibration success” is predicted maximum (3 sigma) heading error of the INS after calibration, in degrees*10. For example, byte #3 equal to 5 corresponds to the INS accuracy ±0.5 deg. If calibration is successful but INS cannot estimate predicted accuracy, it returns byte #3 equal to 255.

6.8.2. StopClbRun command

After receiving the StopClbRun command (code 0x20 in the “Payload” field) the INS early stops data accumulation in the calibration run before set accumulation time is reached.

Then the calibration procedure continues in the same way as after set accumulation time was reached.

6.8.3. AcceptClb command

The AcceptClb command (code 0x2E in the “Payload” field) is applied to accept the calibration parameters and to save them to the INS nonvolatile memory. This command can be used in the end of the calibration procedure.

The INS answers on this command. The INS calculates the check sum of the message (without its header and check sum) and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

6.8.4. ExitClb command

The ExitClb command (code 0xFE in the “Payload” field) is used to exit from the calibration without any calculations in the INS and without saving any calibration parameters. The INS stops work in operating mode and goes into the idle mode.

The INS answers on this command. The INS calculates the check sum of the message (without its header and check sum) and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

6.8.5. StartVG3DClb command for INS VG3D calibration

Since firmware version 2.6.2.2 the INS provides **VG3D calibration**. The **VG3D calibration** is designed for carrier objects that can operate in full heading, pitch and roll ranges. VG3D calibration is similar to 3D calibration but allows performing simpler rotation than is necessary for 3D calibration.

Note: VG3D calibration is at the testing stage. Please contact Inertial Labs about the possibility of using the VG3D calibration.

To start the VG3D calibration the host computer sends to the INS the StartVG3DClb command (code 0x25 in the "Payload" field) followed by message with block of parameters listed in the Table 6.88. This message have the byte structure shown in the Table 6.2, and should be sent without pause after sending the StartVG3DClb command.

The INS calculates the check sum of received parameters and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

Then the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don't move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out the block of the initial alignment data (see the Section 6.3.12) and starts data accumulation during time specified in message sent after the StartVG3DClb command (see the Table 6.88).

During the INS data accumulation, the object should be rotated in full azimuth range and maximum possible pitch and roll ranges. Allowed object motion should be agreed with Inertial Labs.

After set accumulation time is reached or StopClbRun command is sent to the INS (see section 6.8.2 for details) the INS finishes data accumulation and calculates the calibration parameters.

After calculation of the calibration parameters that takes <0.5 seconds, the INS gives out message with the calibration results (see the Table 6.89) and it waits one of the next commands:

- the AcceptClb command (see section 6.8.3) to accept and save the calibration parameters (usually if the “Calibration success” byte in the INS message is nonzero and corresponds to satisfactory INS heading accuracy (see the Table 6.89 and Note below it));
- or the ExitClb command (see section 6.8.4) to exit from calibration procedure without accepting and saving its results (usually if the “Calibration success” byte in the INS message is equal to 0 or corresponds to not satisfactory INS heading accuracy (see the Table 6.89 and Note below it)).

The INS answers on these commands with checksum and goes to idle mode.

6.8.6. Start2D2TClb command for INS 2D-2T calibration

The **2D-2T calibration** is designed for objects that operate in full azimuth range but with limited range of pitch and roll angles. This calibration procedure involves a few full 360° rotations of the object in azimuth with different pitch angles.

To start the 2D-2T calibration the host computer sends to the INS the Start2D2TClb command (code 0x22 in the “Payload” field) followed by message with block of parameters listed in the Table 6.88. This message have the byte structure shown in the Table 6.2, and should be sent without pause after sending the Start2D2TClb command.

The INS calculates the check sum of received parameters and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

The 2D-2T calibration procedure involves a few runs with full 360° rotations of the object with installed INS in heading with different pitch angles.

Set the object to the first pitch angle (usually the minimum pitch angle is set first). Then send the StartClbRun command followed by message (see section 6.8.7) to start the first run of the calibration.

After receiving the StartClbRun command with its message, the INS calculates the check sum of received block of parameters and returns it for checking. This check sum should be equal to the check sum in the StartClbRun command message that was sent to the INS. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

Then the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don't move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out the block of the initial alignment data (see Section 6.3.12) and starts data accumulation during time specified in message sent after the Start2D2TClb command (see the Table 6.88). Rotate object in azimuth with approximately constant pitch and roll. This rotation must include one or more full 360 deg turns. Please, correct the time required for such rotation in the «Time of data accumulation» field of the message (Table 6.88) to provide necessary rotation.

After set accumulation time is reached or StopClbRun command is sent to the INS (see section 6.8.2 for details) the INS gives out message with result of the calibration run (see the Table 6.90).

Table 6.90. Payload of the INS message after each calibration run of the 2D-2T calibration

Byte	Parameter	Format	Length	*Note*
0	Type of calibration	**byte**	**1**	2 for 2D-2T calibration
1	Calibration run	**byte**	**1**	1, 2, …
2	Percent of used data points	**byte**	**1**	
3	Calibration success	**byte**	**1**	0 – unsuccessful; >0 – successful (see Note below)
4-7	Reserved	**float**	**4**	
8-11	Average pitch, deg	**float**	**4**	
12-15	Average roll, deg	**float**	**4**	
16-27	Reserved	**float**	**3*4**	
28-29	USW	**word**	**2**	See section 6.10

If the "Calibration success" byte is zero (calibration run is not successful) in the INS answer Table 6.90 then this run will be excluded from calculations in the 2D-2T calibration procedure. To complete this procedure, it is necessary to perform at least two successful runs with essentially different pitch angles.

Note: there is estimation of the calibration quality as predicted INS heading accuracy. So nonzero value of byte #3 “Calibration success” is predicted maximum (3 sigma) heading error of the INS after calibration, in degrees*10. For example, byte #3 equal to 5 corresponds to the INS accuracy ±0.5 deg. If calibration is successful but INS cannot estimate predicted accuracy, it returns byte #3 equal to 255.

After each calibration run completed the INS sends message with payload shown in the Table 6.90, and it waits one of the next three commands from the host computer:

1. StartClbRun command followed by its message (see section 6.8.7) to start new calibration run. Before send this command the object should be turned to the next pitch angle. After sending this command the above described procedure of the calibration run with object rotation in heading should be performed.
2. FinishClb command (see section 6.8.8 for details) to finish the calibration procedure and to calculate calibration parameters. After that the INS gives out message with the calibration results (see the Table 6.89) and waits one of the two commands:

   a. the AcceptClb command (see section 6.8.3) to accept and save the calibration parameters (usually if the “Calibration success” byte in the INS message is nonzero and corresponds to satisfactory INS heading accuracy (see the Table 6.89 and Note below it));
   b. or the ExitClb command (see section 6.8.4) to exit from calibration procedure without accepting and saving its results (usually if the “Calibration success” byte in the INS message is equal to 0 or corresponds to not satisfactory INS heading accuracy (see the Table 6.89 and Note below it)).

   The INS answers on these commands with checksum and goes to idle mode.

3. ExitClb command (see section 6.8.4) In this case the calibration finishes without any calculations in the INS and without saving any calibration parameters. The INS answers on this command with checksum and goes into the idle mode.

Notes:
1. Rotation of the object with the INS in heading must include one or more full 360° turns. Please, correct the time required for saving data in the «**Accumulation time»** window to attain necessary rotations.
2. During calibration run pitch and roll angles should be approximately constant.
3. If place of the INS mounting on the object is changed, or if the object is changed, then the INS should be re-calibrated on the hard and soft iron of this object.

6.8.7. StartClbRun command

If calibration procedure includes more than one run (like 2D-2T calibration) then the StartClbRun command (code 0x2B in the “Payload” field) is used to start each run.

For unification with the StartClbRun command for some other calibration types, this command must be followed by message with block of parameters listed in the Table 6.91. But for the 2D-2T calibration the values of those 6 bytes don’t influence, so these 6 bytes may be any, for example zeros. Only requirement is that this message should have the byte structure shown in the Table 6.2, and should be sent without pause after sending the StartClbRun command.

Table 6.91. Payload of the message following after the StartClbRun command (block of parameters loaded to the INS)

Byte	Parameter	Format	Length	*Note*
0-3	Reserved	**float**	**4**	
4-5	Reserved	**word**	**2**	

After receiving the StartClbRun command the INS calculates the check sum of received parameters and returns it for a checking. This check sum should be equal to the check sum in the StartClbRun command message that was sent to the INS. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

6.8.8. FinishClb command for INS 2D-2T calibration

After receiving the FinishClb command (code 0x2C in the “Payload” field) the INS finishes the calibration procedure with multiple runs (like 2D-2T) and calculates the calibration parameters. After that the INS gives out message with the calibration results (see the Table 6.89).

Then the INS waits one of the next commands:

- the AcceptClb command (see section 6.8.3) to accept and save the calibration parameters;
- or the ExitClb command (see section 6.8.4) to exit from calibration procedure without accepting and saving its results.

6.8.9. Start2DClb command for INS 2D calibration

The **2D calibration** is designed for carrier objects that operate in full azimuth range but with small pitch and roll angles (not more than a few degrees). This calibration procedure involves full 360° rotation of the carrier object in azimuth. During this rotation pitch and roll angles must be as close to zero as possible.

To start the 2D calibration the host computer sends to the INS the Start2DClb command (code 0x21 in the “Payload” field) followed by message with block of parameters listed in the Table 6.88. This message have the byte structure shown in the Table 6.2, and should be sent without pause after sending the Start2DClb command.

The INS calculates the check sum of received parameters and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

Then the INS starts process of initial alignment that takes usually 30 seconds. This process includes the INS gyros bias estimation, therefore don’t move the INS during its initial alignment. If this requirement disregarded, then large errors may occur in orientation angles calculation.

Note: If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

After completing of the initial alignment the INS gives out the block of the initial alignment data (see the Section 6.3.12) and starts data accumulation during time specified in message sent after the Start2DClb command (see the Table 6.88). Rotate carrier object in azimuth with pitch and roll angles close to zero as possible. This rotation must include one or more full 360 deg turns. Please, correct the time required for such rotation in the «Time of data accumulation» field of the message (Table 6.88) to provide necessary rotation.

After set accumulation time is reached or StopClbRun command is sent to the INS (see section 6.8.2 for details) the INS finishes data accumulation and calculates the calibration parameters.

After calculation of the calibration parameters that takes <0.5 seconds, the INS gives out message with the calibration results (see the Table 6.89) and it waits one of the next commands:

- the AcceptClb command (see section 6.8.3) to accept and save the calibration parameters (usually if the “Calibration success” byte in the INS message is nonzero and corresponds to satisfactory INS heading accuracy (see the Table 6.89 and Note below it));
- or the ExitClb command (see section 6.8.4) to exit from calibration procedure without accepting and saving its results (usually if the “Calibration success” byte in the INS message is equal to 0 or corresponds to not satisfactory INS heading accuracy (see the Table 6.89 and Note below it)).

The INS answers on these commands with checksum and goes to idle mode.

6.8.10. ClearClb command

The ClearClb command (code 0x2F in the “Payload” field) is used to clear parameters of the hard and soft iron calibration from the INS nonvolatile memory.

The INS answers on this command. The INS calculates the check sum of the message (without its header and check sum) and returns it for a checking. Byte structure of this message is shown in the Table 6.2 where payload is the calculated check sum (1 word).

You should clear parameters of the soft and hard iron calibration if you uninstall the INS from object to avoid incorrect azimuth determination with standalone INS.

6.8.11. GetClbRes command

The GetClbRes command (code 0x2A in the “Payload” field) can be send from the host computer to check the last calibration results of the INS. As answer on this command the INS sends out the message with the data block near the same as after completing calibration, see the Table 6.92.

Table 6.92. Payload of the INS answer on request GetClbRes about calibration results

Byte	Parameter	Format	Length	*Note*
0	Type of calibration performed	**byte**	**1**	0 – INS is not calibrated; 1 – 2D calibration; 2 – 2D-2T calibration; 3 – 3D calibration; 5 – VG3D calibration; 6 – automatic calibration; >0x80 – INS is calibrated by loading calibration parame-ters from other software (e.g.

				GUI software).
1	Number of used calibration runs	byte	1	
2	Reserved	byte	1	
3	Calibration success	byte	1	0 – not successful calibration >0 – successful calibration (see Note below)
4-39	Matrix for soft iron correction	float	9*4	Matrix Tm_c (3×3) by rows
39-51	Matrix for hard iron correction	float	3*4	Matrix Hm_0 (3×1)

Note: There is estimation of the calibration quality as predicted INS heading accuracy. So nonzero value of byte #3 "Calibration success" is predicted maximum (3 sigma) heading error of the INS after calibration, in degrees*10. For example, byte #3 equal to 5 corresponds to the INS accuracy ±0.5 deg. If calibration is successful but INS cannot estimate predicted accuracy, it returns byte #3 equal to 255.

6.8.12. StartVG3Dclb_flight and StopVG3Dclb_flight commands for start and finish INS on-the-fly VG3D calibration

Since firmware version 2.6.2.2 the INS provides **on-the-fly VG3D calibration**. It allows to calibrate INS unit during INS ordinary operation without interruption of INS navigation data calculation and output.

To start the on-the-fly VG3D calibration the host computer sends to the INS the **StartVG3Dclb_flight** command (see Appendix C):

AA 55 00 00 07 00 26 2D 00

Note: these are hexadecimal numbers but not ASCII text symbols.

After receiving of this command INS starts accumulation of magnetometers data for VG3D calibration. At this the bit #7 is set to 1 in INS status word USW for indication of data accumulation process (see Fig. 6.4 and section 6.10).

The carrier object with INS unit should be rotated in full azimuth range with maximum possible pitch and roll ranges. For example, airplane should perform at least two full 360° coordinated turns (on the right and on the left) with maximum roll angles.

After finishing of calibration rotation of the carrier object it is necessary to send the StopVG3Dclb_flight command (see Appendix C):

AA 55 00 00 07 00 27 2E 00

Note: these are hexadecimal numbers but not ASCII text symbols.

After receiving of this command INS stops data accumulation for VG3D calibration and sets the USW bit #7 to 0 (see Fig. 6.4).

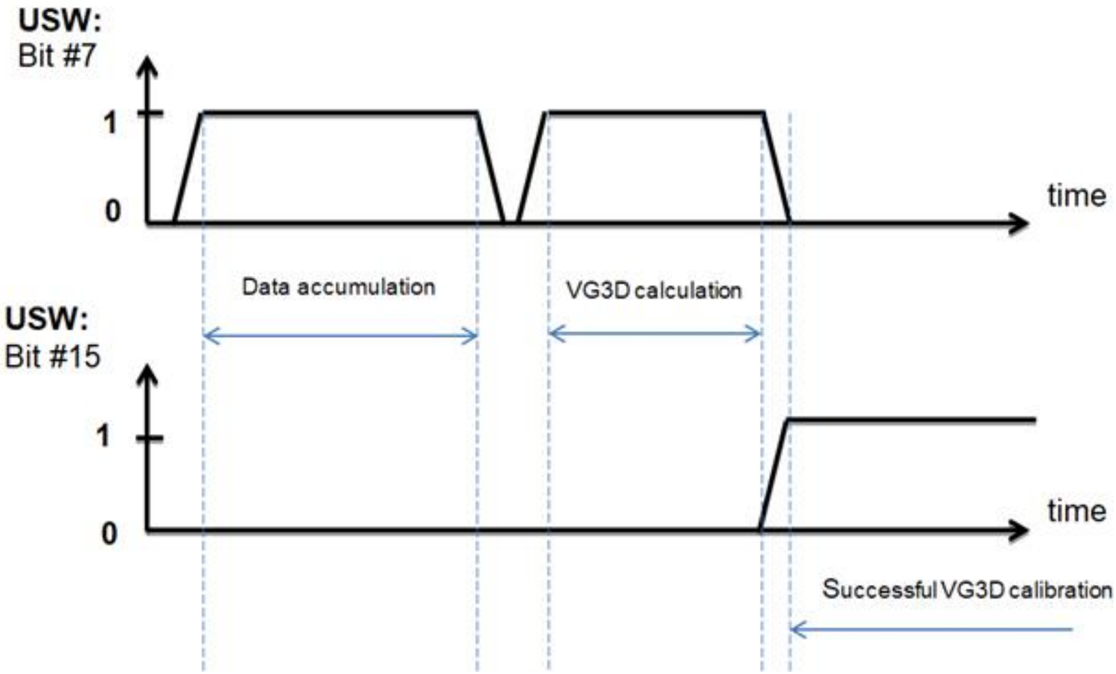


Fig. 6.4. The diagram of indication of on-the-fly VG3D calibration process in the USW (Unit Status Word, see section 6.10)

Then INS starts calculations for VG3D calibration and sets USW bit #7 to 1 again. In the end of these calculations, if calibration is successful, INS calculates calibration parameters for compensation of hard and soft iron, and stores them to INS nonvolatile memory. Also INS sets the USW bit #15 to 1 to inform the host system that on-the-fly VG3D calibration is performed and successful (see Fig. 6.4). If this calibration was estimated as unsuccessful then the USW bit #15 is set to 0.

Calculated calibration parameters are applied immediately to INS magnetometers data for compensation of hard and soft iron of the carrier object.

Note during all steps of on-the-fly VG3D calibration the INS unit continues calculation of navigation data and their output. Because these calibration parameters are stored to INS nonvolatile memory then they will be applied at all the next INS operations until new calibration is performed or parameters are cleared using the ClearClb command (see section 6.8.10).

On the other hand, if INS unit is uninstalled from the carrier object then it is necessary to clear parameters of the soft and hard iron calibration using the ClearClb command (see section 6.8.10).

Parameters of the on-the-fly VG3D calibration can be checked after INS stop using the GetClbRes command (see section 6.8.11).

6.8.13. Automatic calibration of magnetometers

Since firmware version 3.4.3.0 the INS provides automatic 3D calibration of INS magnetometers during INS ordinary operation without interruption of INS navigation data calculation and output. In contrast to the on-the-fly VG3D calibration, automatic calibration is being performed during all time of INS run. The same as for VG3D calibration, it is necessary the carrier object changes its heading, pitch and (or) roll enough during run. Also, since INS firmware version 3.4.3.3 it is possible to perform automatic 2D calibration of magnetometers.

Automatic 2D or 3D calibration is allowed only if magnetometers are not used for INS heading correction in current INS run. So only following variants of INS heading correction are acceptable for INS automatic calibration of magnetometers (see parameter 0x58 H_corr_type in Table 6.80): GNSS track, Dual GNSS, Inertial.

To allow automatic calibration of magnetometers set parameter 0x84 Mag_auto_clb to 1 for 2D or to 2 for 3D calibration (see Table 6.80) or set it using INS GUI (see INS GUI User's Manual, section "10.4.5. Automatic 2D and 3D calibration").

Automatic calibration is in progress when GNSS data are available. Parameters of this calibration are applied immediately for compensation of the carrier's hard and soft iron in magnetometers data. At GNSS data lost this compensation is continued using the last estimated parameters of calibration.

Automatic calibration can be stopped during INS run by sending **StopMagAutoClb** command (see Appendix C):

AA 55 00 00 07 00 3B 42 00

This makes sense when the rest run of INS unit is estimated to be in poor magnetic environment (like landing of UAV).

Parameters of automatic calibration of magnetometers will be saved to the INS flash memory after INS unit is stopped by Stop command (see section 6.3.3). If INS unit is stopped by power off, then parameters of automatic calibration cannot be stored to the INS flash memory.

6.8.14. Choice of set of INS calibration parameters

INS has several procedures to perform different variants of magnetometers calibration on hard and soft iron of the carrier object. INS keeps in its flash memory four sets of calibration parameters for magnetometers. User can choose one of them for the next INS operation by one of two means:

- using LoadINSpar_RAM command to change parameter 0x7E **InitClbType**, then SaveINSpar_FLS command to save parameters from RAM to flash memory (see section 6.3.14);
- or using INS GUI (see INS GUI User's Manual, section "10.4.6. Choice of set of INS calibration parameters").

Parameter InitClbType specifies set of hard/soft iron calibration parameters for the next INS start (see Table 6.80):

- **Last INS Clb** – parameters of the last calibration of magnetometers;
- **Factory Clb** – factory calibration parameters;
- **2D-2T, 3D, VG3D, 2D Clb** – parameters calculated after one of these calibrations;
- **Previous Auto Clb** – parameters calculated in the end of automatic calibration.

<u>Notes</u>:

1. Parameter InitClbType is set automatically to appropriate value "2D-2T, 3D, VG3D, 2D Clb" or "Previous Auto Clb" after each calibration performed.
2. "Last INS Clb" is always copy of parameters of the last performed calibration.
3. "Factory Clb" parameters correspond to cleared calibration of magnetometers on hard and soft iron. User can switch InitClbType to "Factory Clb" set if he wants INS work with magnetometers without calibration on hard and soft iron, but he doesn't want to clear calibration parameters.
4. After cleaning of magnetometers calibration on hard and soft iron, parameters "Last INS Clb", "2D-2T, 3D, VG3D, 2D Clb", "Previous Auto Clb" become cleared, and parameter InitClbType is set to "Factory Clb".

6.9. INS automatic start

Since firmware version 1.0.2.0 the Inertial Labs^{TM} INS auto start is implemented that allows start of its operation and data output after power on without any command from the host computer. There is possible to choose desirable output data format for auto start (see section 6.2).

The auto start option can be enabled or disabled, via INS GUI (see INS GUI User's Manual, section "10.6. INS automatic start"), and via the command 0x07 AutoStart (see Table 6.80) since INS firmware version 3.3.0.1.

If the auto start option is enabled, then after the INS power on the next operations take place:

- Initialization of the on-board GNSS receiver and IMU that takes not more than 15 seconds. The INS LED indicator lights yellow.
- Then the INS automatically starts operation from sending out the message AA 55 01 **XX** 08 00 00 00 **YY YY** (in hexadecimal format) where **XX** is the command code that corresponds to output data format which INS started with (see matching Table 6.58). **YY YY** is the check sum of this message (the arithmetical sum of bytes 2 to 7, the low byte is transmitted first). The INS LED indicator changes color to green.
- After that the initial alignment procedure starts when initial orientation angles are calculated and gyros bias is estimated for its next compensation. Therefore, don't move the INS during initial alignment process. If this requirement disregarded, then large errors may occur in orientation angles calculation.

  Note: Default time of the initial alignment is 30 seconds. If the device starts while moving or for some other reason it is impossible to achieve absolutely stationarity standing at the start, you should set the initial alignment time to 0 seconds (see section 6.3.4 LoadINSpar command) to skip initial alignment. But INS dynamic accuracy may be decreased in such case.

- After completing of the initial alignment the INS gives out message with block of the initial alignment data (see section 6.3.12) and starts data output according to the chosen data format. The INS LED indicator lights green.
- Valid INS position and velocity data appears only after the on-board GNSS receiver starts output navigation data. The receiver requires 50 seconds after power on for this so-called cold start.

Note: To identify the INS output data format at auto start mode use the INS data identifier in the data block structure (see Table 6.2, byte #3) which is equal to the command code and corresponds to data format according to the Table 6.58. This is implemented in the INS firmware since version 2.1.2.0.

To stop the INS please send the Stop command (see section 6.3.3). After receiving the Stop command, the INS stops data calculation and goes to the idle mode. The INS LED indicator changes its color to red. The INS is ready to receive any command from the host computer.

6.10. The Unit Status Word definition

The Unit Status Word (USW) provides the INS state information. The low byte (bits 0-7) of USW indicates failure of the INS. If this byte is 0, the INS operates correctly, if it is not 0, see the Table 6.93 for type of failure. The high byte (bits 8-15) contains a warning or is informative for the user. Status of each bit of the USW warning byte is specified in the Table 6.93.

Table 6.93. The Unit Status Word description

	Bit	Parameter	Description
Low (failure) byte	0	Initial Alignment	0 – Successful initial alignment 1 – Unsuccessful initial alignment due to INS moving or large changing of outer magnetic field
	1	Software status	0 – Correct operation 1 – Incorrect data appeared at calculations
	2	Gyroscope Unit	0 – No failure 1 – Failure detected
	3	Accelerometer Unit	0 – No failure 1 – Failure detected
	4	Magnetometer Unit	0 – No failure 1 – Failure detected
	5	Electronics	0 – No failure 1 – Failure detected
	6	GNSS receiver	0 – No failure 1 – Failure detected
	7	On-the-fly calibration	1 – during data accumulation and calculation 0 -- otherwise
High (warning) byte	8	Incorrect Power Supply	0 – Supply voltage is not less than minimum level 1 – Low supply voltage detected
	9		0 – Supply voltage is not greater than max level 1 – High supply voltage detected
	10	Angular Rate Exceeding Detect	0 – X-angular rate is within the range 1 – X-angular rate is outrange
	11		0 – Y-angular rate is within the range 1 – Y-angular rate is outrange
	12		0 – Z-angular rate is within the range 1 – Z-angular rate is outrange
	13	Large Magnetic Field Detect	0 – Total magnetic field is within the normal range 1 – Total magnetic field limit is exceeded
	14	Environmental Temperature	0 – Temperature is within the operating range 1 – Temperature is out of the operating range
	15	On-the-fly calibration	0 – No on-the-fly calibration 1 – Successfully calibrated during current run

6.11. Post-processing of the INS and GNSS data

For applications requiring highly accurate postmission position, velocity and orientation, the INS and GNSS data post-processing can be used. This feature is provided by NovAtel software, see http://www.novatel.com/products/software/ .

For such post-processing the raw GNSS and raw IMU data should be used.

The Inertial Labs™ INS uses additional COM ports (COM2 or COM3) for output the raw GNSS receiver data (see section 6.11.1). For these data recording from receiver an external program **GNSS_Reader** can be used. The GNSS_Reader is supplied with the Inertial Labs INS GUI software.

File with raw IMU data can be created from files .bin, .prm saved by INS GUI. Use “Convert to IMU data” item in the “Convert” menu – see the Inertial Labs™ INS GUI User’s Manual, section “12.2. Raw IMU data generation”.

For more details about post-processing see Section “12. INS and GNSS data post-processing” in the INS GUI User’s Manual.

6.11.1. Raw GNSS receiver data

The Inertial Labs™ INS uses the COM2 or COM3 port for output the raw GNSS receiver data. The INS starts output of these data after power on and completing of the receiver initialization (when the INS LED indicator switches from yellow to red).

Raw GNSS data consist of necessary logs for post-processing. There are synchronous and asynchronous logs. The data for synchronous logs are generated with set frequency. In order to output the most current data as soon as they are available, asynchronous data are generated at irregular intervals. Full list of generated logs is shown in the Table 6.94.

Since firmware version 3.2.5.8 user can select a minimal set of raw GNSS data. Appropriate list of generated logs is shown in the Table 6.95.

Table 6.94. Logs of raw GNSS data

Log	Description
Asynchronous	
CLOCKSTEERING	Clock steering status
GLOCLOCK	GLONASS clock information
ALMANAC (1)	Decoded GPS Almanac
GPSEPHEM (1)	Decoded GPS ephemerides
RAWALM	Raw Almanac data
RAWEPHEM (1)	Raw ephemeris
RAWGPSSUBFRAME (1)	Raw subframe data
RAWCNAVFRAME	Raw CNAV frame data
RAWGPSWORD	Raw navigation word
GLOALMANAC (1)	Decoded GLONASS Almanac
GLOEPHEMERIS (1)	Decoded GLONASS ephemeris
GLORAWALM	Raw GLONASS Almanac data
GLORAWEPHEM	Raw GLONASS Ephemeris data
GLORAWFRAME	Raw GLONASS frame data
GLORAWSTRING (1)	Raw GLONASS string
MARK2POS (2)	Position at time of mark input event
MARK2TIME (2)	Time of mark input event
BDSEPHEMERIS	Decoded BeiDou ephemeris
GALINAVEPHEMERIS	Decoded Galileo INAV ephemeris
GALFNAVEPHEMERIS	Decoded Galileo FNAV ephemeris
HEADING2	Heading data (INS-D only)
Synchronous	
CLOCKMODEL	Current clock model status
TIMESYNC (1)	Synchronize time between GNSS receivers
TIME (1)	Time data
RANGE (1)	Satellite range information
RANGEGPSL1	L1 version of the RANGE log
TRACKSTAT (1)	Tracking status

Table 6.95. Minimal set of raw GNSS data logs

<table>
<tr><th colspan="2">Log</th><th>Description</th></tr>
<tr><th colspan="3">Asynchronous</th></tr>
<tr><td colspan="2">RAWEPHEM</td><td>Raw ephemeris</td></tr>
<tr><td colspan="2">GLOEPHEMERIS</td><td>Decoded GLONASS ephemeris</td></tr>
<tr><td rowspan="4">INS-B / P / D</td><td>BDSEPHEMERIS</td><td>Decoded BeiDou ephemeris</td></tr>
<tr><td>GALINAVEPHEMERIS</td><td>Decoded Galileo INAV ephemeris</td></tr>
<tr><td>GALFNAVEPHEMERIS</td><td>Decoded Galileo FNAV ephemeris</td></tr>
<tr><td>QZSSEPHEMERIS</td><td>Decoded QZSS parameters</td></tr>
<tr><td rowspan="2">INS-DL</td><td>BD2EPHEM</td><td>Decoded BeiDou ephemeris</td></tr>
<tr><td>GALEPHEMERIS</td><td>Decoded Galileo ephemeris</td></tr>
<tr><td colspan="2">MARK2POS (2)</td><td>Position at time of mark input event</td></tr>
<tr><td colspan="2">MARK2TIME (2)</td><td>Time of mark input event</td></tr>
<tr><th colspan="3">Synchronous</th></tr>
<tr><td colspan="2">RANGECMP</td><td>Compressed version of the RANGE log (Satellite range information)</td></tr>
</table>

Notes:

1. INS-DL supports only logs marked with [(1)] in the Table 6.94.

2. MARK2POS[(2)] and MARK2TIME[(2)] logs are not used in post processing, but they can be added to the raw GNSS data if input marks are enabled. MARK2POS and MARK2TIME logs appear when a pulse is detected at mark input (see section “5.6. Mark input description”).

6.12. Synchronization of INS data with LiDAR and other devices

Synchronization of the Inertial Labs™ INS measurements with data form other devices is very important in many applications. The INS can trigger other devices, or an external device can trigger the INS measurements.

6.12.1. Synchronization pulses issued by INS

To trigger external devices, the Inertial Labs™ INS outputs accurate pulse per second (PPS) signal generated by on-board GNSS receiver. The PPS signal is provided by appropriate pin of the INS main connector (see Table 5.1). See section “5.5. PPS description” for details.

Adjustment of the PPS signal (pulse polarity, period, width) can be done via INS GUI (see INS GUI User’s Manual, section “13.1. Control of PPS output signal”), and via the command 0x51 PPS_switch, PPS_polarity, PPS_period, PPS_pulse_width (see Table 6.80) since INS firmware version 3.3.0.1.

6.12.2. Trigging of INS by external devices

The Inertial Labs™ INS output data can be get on request by Request command. For this purpose, the INS unit should be started in the “On Request” (on demand) mode. In such case the INS sends one data block after each Request command issued from the host computer. See section “6.3.2. SetOnRequestMode command – getting INS data on request (on demand)” for details.

6.12.3. Synchronization of INS data with LiDAR

For Inertial Labs INS operation with LiDAR it is necessary to make the next connections:

- use INS COM1 port for output of the main INS data;
- use INS COM2 port for output of GNSS raw data or NMEA data set generated by INS onboard GNSS receiver;
- connect INS COM3 port for output of $GPRMC messages issued by INS onboard GNSS receiver to LiDAR;
- connect pulse-per-second (PPS) signal generated by INS onboard GNSS receiver to LiDAR;
- optionally, for camera synchronization – connect mark input (MARK IN pin) line for input signal from camera to trigger specific GNSS receiver data (MARK2POS and MARK2TIME logs)

All these data and signals are available on the main INS connector – see section “5. Electrical Interface”. There are two ways to configure INS data and signals, via INS GUI (see INS GUI User’s Manual, section “13.3. INS operation with LiDAR” for details), and via appropriate commands since INS firmware version 3.3.0.1 (see section 6.3.14).

6.13. Change of the main COM port baud rate

COM1 is the main COM port. It is used for commands and data transfer between the Inertial Labs™ INS and the host computer.

The default baud rate for the INS COM1 port is set to 115200 bps (maximum for the standard COM-port). Since firmware version 2.2.0.0 the INS supports different baud rates: 4800, 9600, 14400, 19200, 38400, 57600, 115200, 230400, 460800 bps.

There are two ways to change INS COM1 port baud rate, via INS GUI (see INS GUI User’s Manual, section “4.2.7. Change of the main COM port baud rate”), and via the command 0x01 COM1_bps (see Table 6.80) since INS firmware version 3.3.0.1.

Note the same baud rate must be set for COM port of the host computer.

6.14. Limitation of the INS maximum output data rate

When setting of the output data rate for the INS unit using LoadINSPar command (see section 6.3.4) or using the Inertial Labs™ INS GUI it is essential to ensure the chosen baud rate of the main COM port is capable of handling the data throughput with desirable data rate. The maximum data rate (Hz) can be calculated using the baud rate and data package length:

$$max\_data\_rate = \frac{COM\_baud\_rate}{bits\_per\_byte * package\_length}, \qquad (6.2)$$

where COM_baud_rate is COM port baud rate (bits/s); bits_per_byte = 11 bits per one transferred byte of data; package_length for binary data = payload length plus 8 bytes of overhead. See Tables 6.4, 6.9, 6.10, 6.11, 6.12, 6.13, 6.15, 6.21, 6.22, 6.27 for payload length of binary output data formats. The package_length of the text output data formats correspond to their structure shown in sections 6.2.1 to 6.2.14.

Below Table 6.96 contains data package length for each output data format and also maximum data rate calculated using formula (6.2), with some spare. Note the maximum rate of INS data is limited by 200 Hz.

Table 6.96. INS maximum data rate for different output data formats

Output data format	Data package length, bytes	COM-port baud rate, bps					
		9600	19200	38400	**115200**	230400	460800
		Maximum data rate, Hz					
INS Sensors Data	84+8	9	10	30	**100**	200	200
INS OPVT	92+8	8	10	30	**100**	200	200
INS QPVT	94+8	8	10	30	**100**	200	200
INS OPVT2A	101+8	8	10	30	**90**	190	200
INS OPVT2AW	103+8	7	10	30	**90**	180	200
INS OPVT2AHR	129+8	6	10	20	**70**	150	200
INS OPVTAD	177+8	4	10	10	**50**	100	200
INS Minimal Data	42+8	10	30	60	**200**	200	200
INS_OPVT_rawIMUdata	90+8	5	10	40	**100**	200	200
INS OPVT GNSSext	186+8	4	8	10	**50**	100	200
SPAN_rawimu	72	20	25	50	**100**	200	200
INS NMEA	93	9	10	30	**100**	200	200
INS Sensors NMEA	141	6	10	20	**70**	140	200
Cobham UAV 200 Satcom	128	6	10	25	**80**	150	200
NAV440	39	20	40	80	**200**	200	200

Notes:

1. INS unit controls correctness of the data rate setting. If user sets data rate which exceeds limit shown in Table 6.96, then its value is corrected. True data rate is given out

in the byte #3 of INS message after completing of the initial alignment procedure (see section 6.3.12, Table 6.69, Table 6.71).
2. Since firmware version 3.2.0.0, there are available only data rates that are factors of 200: (1, 2, 4, 5, 8, 10, 20, 25, 40, 50, 100, 200) Hz.

6.15. INS solution status

“INS solution status” can be chosen for INS output as one of data types in the “User Defined Data” output format since firmware version 3.2.4.3. See section “6.2.12. The “User Defined Data” output format”, Table 6.30, data type #0x54. Values of the INS solution status are listed in Table 6.97.

Table 6.97. INS solution status (data type # 0x54)

Value	Description
0	INS solution is good
1	INS solution is satisfactory, KF convergence is in progress
2	INS started at absence of GNSS solution, it is still absent and no other aiding data were received
3	GNSS data are available but heading correction did not start yet. Maybe vehicle speed is not enough (at “GNSS track” or “Inertial” type of correction), or dual antenna position did not reach “Integer narrow-lane ambiguity solution” or “Integer L1 ambiguity solution” yet (at “Dual antenna GNSS” type of correction)
4	INS operates in autonomous mode (without GNSS or any other aiding data)
5	INS operates without GNSS data but with other external aiding data
6	INS froze position and velocity calculation because time of INS autonomous operation exceeded set maximum time
7	Zero velocity update (ZUPT) is applied to INS solution at detected stop
8	INS solution is invalid

INS operates with some features if the INS solution status is not zero:

- at “INS solution status” equal to 3, the INS outputs just GNSS position and velocity;
- at “INS solution status” equal to 6, the INS freezes position and velocity output after time of INS autonomous operation (without GNSS data or any other aiding data) exceeded maximum time;
- at “INS solution status” equal to 2, 3, 6, the INS calculates orientation using AHRS or Vertical Gyro algorithm depending on enabled or disabled magnetometer data.

Also, since firmware version 3.2.2.7, “simple INS solution status” is added to output data in OPVT2A, OPVT2AW, OPVT2AHR, OPVTAD data formats to indicate good or poor INS data. Good solution means the INS Kalman filter is converged, and INS outputs valid position, velocity and orientation data.

Poor solution is set in one of such cases:

1. If INS started at absence of valid GNSS data, until these data appear.
2. At absence of valid GNSS data during long time that exceeds the maximum time of INS autonomous operation.
3. If INS uses GNSS track angle as heading reference for INS correction or "Inertial" type of INS correction is set, but the carrier object did not reach yet necessary speed to start using the track angle.
4. If INS-D uses dual-antenna GNSS heading as reference, but valid GNSS heading did not appeared yet after INS start.
5. INS solution is invalid.

Such "simple INS solution status" is combined with the "Angles position type" value in above listed output data formats. If the "simple INS solution status" is poor, then 100 is added to the value of the "Angles position type". So use the next simple formula to get "simple INS solution status":

If "Angles position type" < 100 then "simple INS solution status" is good, otherwise it is poor.

6.16. Time stamps in INS messages

The Inertial Labs™ INS provides internal and output data synchronization. If the GPS reference time status is fine (see Table 6.24) then INS uses 1PPS signal from the onboard GNSS receiver for data synchronization and for adjustment of INS processor clock. When the GPS reference time status is not fine then INS use its processor system timer for data synchronization.

All INS output data in the "Continuous" and "On Request" operating modes have accurate time stamps (see section 6.2). There are three variants of time stamps for INS output data:

- **GPS INS Time** is <u>time of INS solution</u>, in milliseconds from the beginning of the GPS reference week;
- **GPS INS Time (round)** is <u>time of INS solution</u>, in milliseconds from the beginning of the GPS reference week, rounded to 1000/(output data rate);
- **GPS IMU Time** is <u>time of the last IMU data package</u>, in milliseconds from the beginning of the GPS reference week.

Resolution of these timestamps is 1 nanosecond. But these time stamps can appear in INS output data in seconds, milliseconds or nanoseconds depending on output data format.

GPS INS Time is the most accurate time stamp. It corresponds to time of IMU data acquisition, so calculated INS position, velocity, orientation are conjunct to this time point. It is recommended for using for post-processing of INS data and for high-dynamic applications where accurate time stamps are very important.

GPS INS Time (round) is more convenient for use and comparison of INS and other device data. GPS time of INS solution in milliseconds is rounded to 1000/(output data rate) value. For example, time stamps of 200 Hz data are rounded to 5 milliseconds step, time stamps of 50 Hz data are rounded to 20 milliseconds step.

GPS IMU Time is subsidiary time. It is the time of the last IMU data package received. Because IMU clock differs from GPS one, the GPS IMU Time drifts relative to GPS INS time. INS calculates this drift and takes it into account at processing of IMU data.

Before firmware version 3.2.4.1 INS output only round timestamps in the most of output data formats except “INS OPVT & Raw IMU Data” and “SPAN rawimu” which included accurate time stamps necessary for post-processing. Since INS firmware version 3.2.4.1 user can choose variant of time stamps in output data. This can be done, via INS GUI by check or uncheck “Round time stamp” checkbox (see INS GUI User’s Manual, section “4.3. Correction options” for details), and via the command 0x5D Round_time_stamp (see Table 6.80) since INS firmware version 3.3.0.1.

Also, in “User Defined Data” output format user can choose any of above three variants of INS time stamps (see section 6.2.12, Table 6.29).

6.17. Using external GNSS receiver

Since firmware version 3.2.8.6, it is possible to use an external GNSS receiver with INS unit. The INS unit with an internal GNSS receiver does not support external receiver. The INS without internal receiver is available upon request.

6.17.1. Adjustment of the external GNSS receiver

The external GNSS receiver should have COM port with RS-232 interface for connection to INS unit. Also, it is highly desirable the external GNSS receiver has PPS output.

It is necessary to configure the GNSS receiver COM port to output data that can be accepted by INS unit. Two variants of external receiver data are supported:

- binary messages from NovAtel receiver;

- NMEA messages from any GNSS receiver.

The best variant is binary data which are more accurate, more quick and more complete than NMEA messages.

To allow INS operate with **NovAtel binary messages**, the external NovAtel GNSS receiver should be configured for output of the next logs:

- *BESTPOS*. To request this log, it is necessary to send the next command to the receiver: `log bestposb ontime X` where X is time period in seconds between BESTPOS logs.
- *BESTVEL*. To request this log, it is necessary to send the next command to the receiver: `log bestvelb ontime X` where X is time period in seconds between BESTVEL logs.
- If dual-antenna GNSS receiver will be used that allows to measure heading, then optional HEADING log can be requested by the command `log headingb onnew`.

It is desirable to set the same logging period X for BESTPOS and BESTVEL logs. If less logging period can be set, then it is better. For example, if the GNSS receiver supports 20 Hz logging, then set minimum logging period as 1/20 Hz or X = 0.05 seconds.

NMEA messages are universal, and almost all GNSS receivers can output such messages that allows so it is possible to connect any GNSS receiver with RS-232 output to the Inertial Labs™ INS unit. But Inertial Labs recommend to avoid using NMEA messages where this is possible. NMEA messages are slow, incomplete, less accurate than binary data, have poor error checking of the format.

To allow INS operate with **NMEA messages**, the external GNSS receiver should be configured for output of the next messages:

- *GPGGA* (time, 3D position and fix related data of the GNSS receiver);
- *GPRMC* (time, 2D position, track made good and speed data provided by the GNSS navigation);
- *GPHDT* – (heading provided by dual-antenna GNSS receiver), it is not obligatory.

It is highly desirable to set maximum data rate for these NMEA messages that the GNSS receiver supports.

6.17.2. Adjustment of the INS unit for operations with the external GNSS receiver

INS hardware should be configured for operations with external GNSS receiver.

1. Connect RS232 port of the external GNSS receiver to COM2 port of the INS unit (see Fig. 5.3 and Table 5.1, pins #8, 15, 16) .
2. It is highly recommended to connect the PPS output of the GNSS receiver to pin #5 of the INS connector.
3. Use Inertial Labs[TM] INS GUI software to configure INS. Go to “Options” / “Devices options” menu item, “External sensors” tab (see Fig. 6.6) and set the next parameters:
   - **Receiver data** – there is drop-down list for choice of variant of the receiver data (see Fig. 6.5).
   - **Port bps** – is COM port baud rate which should be the same as for the receiver’s COM port.
   - **Data rate** – is receiver data rate.

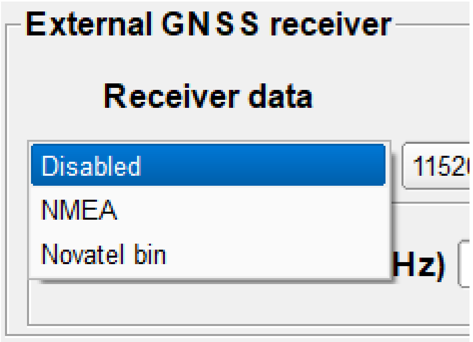


Fig. 6.5. Choice variant of the external GNSS receiver data

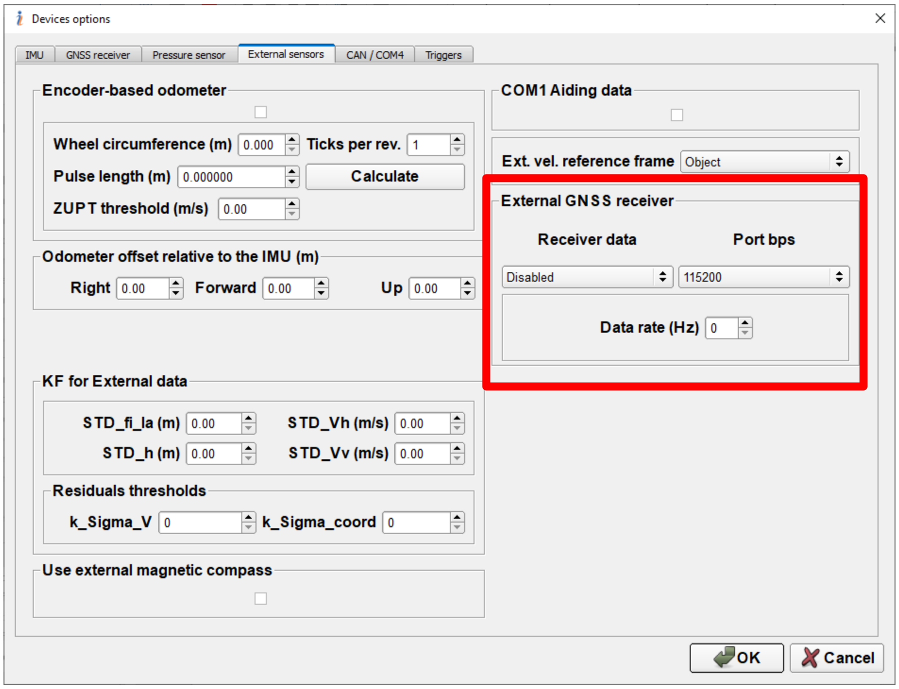


Fig. 6.6. Settings for the external GNSS receiver

Antenna installation process and its configuration is the same as for INS with internal GNSS receiver. See sections 4.2 and 4.3 for a detailed description.

APPENDIX A.
The Inertial Labs™ INS Calibration

The Inertial Labs™ INS GUI software allows to take into account influence of the carrier object soft and hard iron on the heading determination. For this purpose, field calibration of the INS magnetometers on hard and soft iron is provided. This calibration does not require any additional equipment, but it requires setting of the carrier object, where the INS is mounted, in specified positions.

There are several types of the calibration implemented onboard the INS:

- 3D calibration;
- 2D-2T calibration;
- 2D calibration;
- VG3D calibration;
- on-the-fly VG3D calibration;
- automatic 2D and 3D calibration.

3D calibration is designed for carrier objects that can operate in full heading, pitch and roll ranges. For this calibration the carrier object is rotated in the horizon plane (the Z-axis is up) with periodical stops about each 90 degrees for tilting in pitch and roll. After full 360° rotation the carrier object with the INS is turned over (the Z-axis is down) and the procedure described above should be repeated. During this calibration the range of pitch and roll angles changing must be as much as possible.

VG3D calibration is similar to 3D calibration but allows performing simpler rotation than it is necessary for 3D calibration.

2D-2T calibration is designed for carrier objects that operate in full heading range but with limited range of pitch and roll angles. This calibration procedure involves a few full 360° rotations of the carrier object with installed INS in heading with different pitch angles. During each rotation, pitch and roll angles should be as constant as possible.

2D calibration is designed for carrier objects that operate in full azimuth range but with small pitch and roll angles (not more than a few degrees). This calibration procedure involves full 360° rotation of the carrier object with installed INS in the horizon plane. During this rotation pitch and roll angles must be as close to zero as possible.

On-the-fly VG3D calibration allows to calibrate INS unit during INS ordinary run without interruption of INS navigation data calculation and output. It is necessary to send commands to start and to finish the on-the-fly VG3D calibration.

Automatic calibration allows to calibrate INS unit during INS ordinary run without interruption of INS navigation data calculation and output. In contrast to the on-the-fly VG3D calibration, automatic calibration is being performed during all time of INS run.

If place of the INS mounting on the carrier object is changed, or if the carrier object is changed, then the INS should be re-calibrated on the hard and soft iron of the carrier object.

See section 6.8 for detailed description of embedded calibration procedures.

APPENDIX B.
Variants of the Inertial Labs™ INS mounting relative to the object axes

The Inertial Labs™ INS can be mounted on the object in any known position (up to upside-down, upright etc.) relative to the object axes. Such mounting doesn't change right determination of the object orientation if angles of the INS mounting are correctly stored in the INS nonvolatile memory.

To store angles of the INS mounting to its nonvolatile memory please use the Inertial Labs™ INS GUI (item «Device option …» from the «Options» menu) or send LoadINSPar command to the INS directly (see structure of the message following after the LoadINSPar command in the Table 6.59). In both cases these angles are denoted as "Alignment angles".

Angles of the INS position (alignment angles) are set in next order (like heading, pitch and roll setting):

- first alignment angle sets position of the INS longitudinal axis Y relative to longitudinal axes of the object measured in the horizontal plane of the object. Clockwise rotation is positive;
- second alignment angle is equal to angle of inclination of the INS longitudinal axis Y relative to the horizontal plane of the object. Positive direction is up;
- third alignment angle is equal to inclination angle of the INS lateral axis X measured around INS' longitudinal axis. Positive rotation is X axis moving down.

All angles are set in degrees. Some examples of the Inertial Labs INS mounting relative the carrier object are shown on Fig. B.1.

To check correctness of the alignment angles please run the INS using the Inertial Labs INS GUI or using ReadINSpar_RAM command (see section 6.3.14). Default values of the INS alignment angles are all zero.

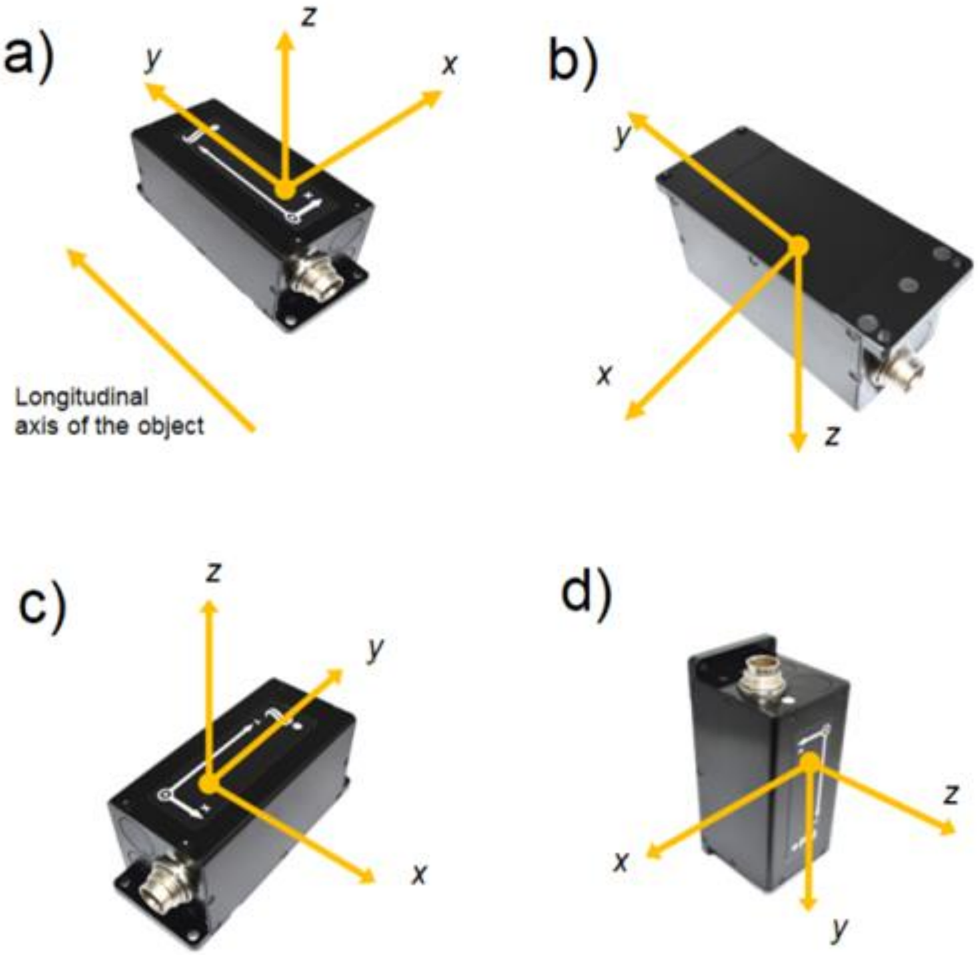


Fig. B.1. Examples of the Inertial Labs™ INS mounting on the carrier object

a – alignment angles are 0, 0, 0 (degrees);
b – alignment angles are 0, 0, 180 (degrees);
c – alignment angles are 90, 0, 0 (degrees);
d – alignment angles are 180, -90, 0 (degrees).

APPENDIX C.
Full list of the Inertial Labs™ INS commands

All the INS commands have the byte structure shown in the Table 6.2. Payload for all commands has length 1 byte and contains code of the command. Below Table C.1 lists all commands with their exact structure in hexadecimal numbers.

Table C.1. List of the INS commands with exact structure

Command name	Code	Exact structure (hex)
Commands for Inertial Labs™ INS control		
INS_SensorsData	0x50	AA 55 00 00 07 00 50 57 00
INS_OPVTdata	0x52	AA 55 00 00 07 00 52 59 00
INS_QPVTdata	0x56	AA 55 00 00 07 00 56 5D 00
INS_OPVT2Adata	0x57	AA 55 00 00 07 00 57 5E 00
INS_OPVT2AWdata	0x59	AA 55 00 00 07 00 59 60 00
INS_OPVT2AHRdata	0x58	AA 55 00 00 07 00 58 5F 00
INS_OPVTADdata	0x61	AA 55 00 00 07 00 61 68 00
INS_minData	0x53	AA 55 00 00 07 00 53 5A 00
INS_OPVT_rawIMUdata	0x66	AA 55 00 00 07 00 66 6D 00
INS_OPVT_GNSSextdata	0x67	AA 55 00 00 07 00 67 6E 00
SPAN_rawimu	0x68	AA 55 00 00 07 00 68 6F 00
User_Def_Data	0x95	AA 55 00 00 07 00 95 9C 00
User_Def_Data_config	0x96	AA 55 00 00 07 00 96 9D 00
Get_User_Def_Data_struct	0x97	AA 55 00 00 07 00 97 9E 00
CAN1_message_set_config	0x98	AA 55 00 00 07 00 98 9F 00
Get_CAN1_message_set_struct	0x99	AA 55 00 00 07 00 99 A0 00
INS_NMEA	0x54	AA 55 00 00 07 00 54 5B 00
INS_Sensors_NMEA	0x55	AA 55 00 00 07 00 55 5C 00
Cobham_UAV200_Satcom	0x46	AA 55 00 00 07 00 46 4D 00
INS_NAV440	0x6A	AA 55 00 00 07 00 6A 71 00
SetOnRequestMode	0xC1	AA 55 00 00 07 00 C1 C8 00
Stop	0xFE	AA 55 00 00 07 00 FE 05 01
LoadINSpar	0x40	AA 55 00 00 07 00 40 47 00
ReadINSpar	0x41	AA 55 00 00 07 00 41 48 00
GetBIT	0x1A	AA 55 00 00 07 00 1A 21 00
GetDevInfo	0x12	AA 55 00 00 07 00 12 19 00
DevSelfTest	0x13	AA 55 00 00 07 00 13 1A 00
LoadINSpar_RAM	0xB0	Has a variable structure, see section 6.3.14
ReadINSpar_RAM	0xB1	Has a variable structure, see section 6.3.14
SaveINSpar_FLS	0xB2	AA 55 00 00 07 00 B2 B9 00
PPPSEED_STORE	0xB6	AA 55 00 00 07 00 B6 BD 00
PPPSEED_RESTORE	0xB7	AA 55 00 00 07 00 B7 BE 00
PPPSEED_CLEAR	0xB8	AA 55 00 00 07 00 B8 BF 00
CAN2_message_set_config	0xB9	AA 55 00 00 07 00 B9 C0 00
Get_CAN2_message_set_struct	0xBA	AA 55 00 00 07 00 BA C1 00
GNSS_switch_on	0x71	AA 55 00 00 07 00 71 78 00
GNSS_switch_off	0x72	AA 55 00 00 07 00 72 79 00
Commands for Inertial Labs™ INS calibration on hard/soft iron		

Start2DClb	0x21	AA 55 00 00 07 00 21 28 00
Start2D2TClb	0x22	AA 55 00 00 07 00 22 29 00
Start3DClb	0x23	AA 55 00 00 07 00 23 2A 00
StartVG3DClb	0x25	AA 55 00 00 07 00 25 2C 00
StartVG3Dclb_flight	0x26	AA 55 00 00 07 00 26 2D 00
StopVG3Dclb_flight	0x27	AA 55 00 00 07 00 27 2E 00
StartClbRun	0x2B	AA 55 00 00 07 00 2B 32 00
StopClbRun	0x20	AA 55 00 00 07 00 20 27 00
FinishClb	0x2C	AA 55 00 00 07 00 2C 33 00
AcceptClb	0x2E	AA 55 00 00 07 00 2E 35 00
ExitClb	0xFE	AA 55 00 00 07 00 FE 05 01
ClearClb	0x2F	AA 55 00 00 07 00 2F 36 00
GetClbRes	0x2A	AA 55 00 00 07 00 2A 31 00
StopMagAutoClb	0x3B	AA 55 00 00 07 00 3B 42 00
Commands for odometer calibration		
Start_Odom_Clb	0x28	AA 55 00 00 07 00 28 2F 00
Stop_Odom_Clb	0x29	AA 55 00 00 07 00 29 30 00

APPENDIX D. Forms of the Inertial Labs™ INS orientation presentation

Define coordinate system $Ox_oy_oz_o$ to be fixed to the carrier object where Ox_o axis is lateral and directed to the right, Oy_o axis is longitudinal and directed forward, Oz_o axis is normal and directed vertical. At usual installation of the INS on carrier object the INS appropriate axes should be parallel to the axes as above Fig. 1.5 shows. Also, it is possible to install the INS in any known position relative to the object with known alignment angles (see Appendix B for details).

The Inertial Labs™ INS calculates orientation of the coordinate system $Ox_oy_oz_o$ fixed to the carrier object with respect to Cartesian geographical reference frame *Oxyz* where axes *Ox* and *Oy* are in the level and directed to the East and North, and *Oz* axis is directed up. Such reference frame is also known as ENU (East-North-Up) Earth-level frame.

Measured angles are the standard Euler angles of rotation from the Earth-level frame to the object frame: heading *K* is first, then pitch θ, and then roll γ -- see Fig. D.1.

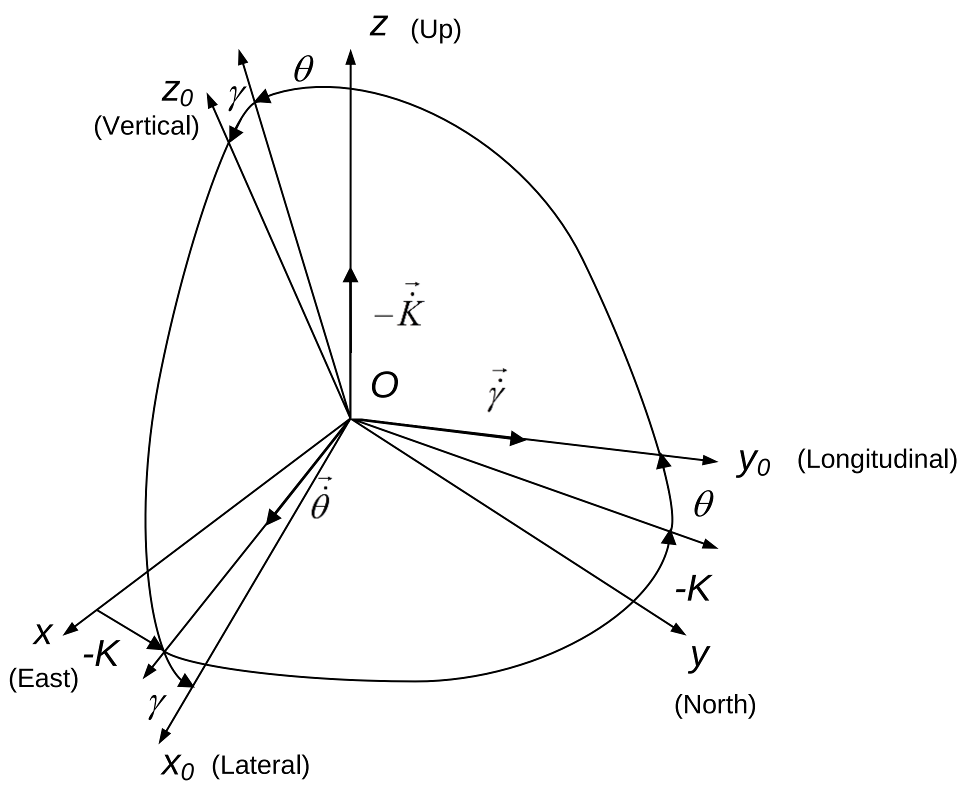


Fig. D.1. Transformation of coordinate systems

Notes:

1. Positive direction of heading is clock-wise. So heading *K* is shown with minus sign on Fig. D.1.

2. In different applications "heading" is also known as "azimuth" or "yaw"; "pitch" is also known as "elevation" or "tilt"; "roll" is also known as "bank".

Due to the definition of Euler angles there is a mathematical singularity when the object longitudinal y_0-axis is pointed up or down (i.e. pitch approaches ±90°). This singularity is not present in the quaternion or directional cosine matrix (rotation matrix) presentation.

Directional cosine matrix (DCM) is the rotation matrix ***C*** from the object body reference frame $Ox_oy_oz_o$ to the geographical reference frame *Oxyz.* According to Fig. D.1, DCM can be represented through Euler angles as

$$\boldsymbol{C} = \begin{bmatrix} \cos K\cos\gamma + \sin K\sin\gamma\sin\theta & \sin K\cos\theta & \cos K\sin\gamma - \sin K\cos\gamma\sin\theta \\ -\sin K\cos\gamma + \cos K\sin\gamma\sin\theta & \cos K\cos\theta & -\sin K\sin\gamma - \cos K\cos\gamma\sin\theta \\ -\cos\theta\sin\gamma & \sin\theta & \cos\theta\cos\gamma \end{bmatrix}. \quad \text{(D.1)}$$

Or, Euler angles can be calculated from elements c_{ij} of directional cosine matrix ***C***:

$$K = \arctan\frac{c_{12}}{c_{22}};\ \theta = \arcsin c_{32};\ \gamma = -\arctan\frac{c_{31}}{c_{33}}. \quad \text{(D.2)}$$

Also the Inertial Labs™ INS provides orientation output in quaternion ***Q*** form which is a hyper-complex number with four components

$$\boldsymbol{Q} = (q_0, q_1, q_2, q_3), \quad \text{(D.3)}$$

where q_0 is real part, q_1, q_2, q_3 are vector part. In other words, q_0 represents the magnitude of the rotation, and the other three components represent the axis about which that rotation takes place. With only four components, quaternion representation of orientation is computationally efficient. However, manipulation of quaternions is not intuitive, so their use in place of directional cosine matrices may increase the chances of mistakes being made.

Quaternion ***Q*** is converted to directional cosine matrix ***C*** using the next expressions:

$$C = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix}. \quad \text{(D.4)}$$

The reverse conversation from directional cosine matrix ***C*** to quaternion ***Q*** is following:

$$q_0 = \frac{1}{2}\sqrt{1 + c_{11} + c_{22} + c_{33}}; \tag{D.5}$$

$$q_1 = \frac{c_{32} - c_{23}}{4q_0}; \quad q_2 = \frac{c_{13} - c_{31}}{4q_0}; \quad q_3 = \frac{c_{21} - c_{12}}{4q_0}.$$

Expressions (D.5) are wide used but they have singularity at $q_0 = 0$. Therefore, the Inertial Labs™ INS uses other expressions that have no singularity:

$$q_0 = \frac{1}{2}\sqrt{1 + c_{11} + c_{22} + c_{33}}; \quad q_1 = \frac{1}{2}\sqrt{1 + c_{11} - c_{22} - c_{33}} \cdot sign(c_{32} - c_{23}); \tag{D.6}$$

$$q_2 = \frac{1}{2}\sqrt{1 - c_{11} + c_{22} - c_{33}} \cdot sign(c_{13} - c_{31}); \quad q_3 = \frac{1}{2}\sqrt{1 - c_{11} - c_{22} + c_{33}} \cdot sign(c_{21} - c_{12}).$$

At necessity to calculate Euler angles from quaternion, calculate elements c_{12}, c_{22}, c_{31}, c_{32}, c_{33}, according to (D.6), and then use formulas (D.2):

$$K = \arctan\frac{2(q_1 q_2 - q_0 q_3)}{q_0^2 + q_2^2 - q_1^2 - q_3^2}; \quad \theta = \arcsin(2q_2 q_3 + 2q_0 q_1); \tag{D.7}$$

$$\gamma = -\arctan\frac{2(q_1 q_3 - q_0 q_2)}{q_0^2 + q_3^2 - q_1^2 - q_2^2},$$

where arctan is four-quadrant inverse tangent.

APPENDIX E. Instant and averaged output data of the Inertial Labs^{TM} INS

E.1. Variants of INS orientation, position and velocity data in the Inertial Labs^{TM} INS

If the output data rate is set to less than 200Hz, the Inertial Labs^{TM} INS can provide instant or averaged output data:

- heading, pitch, roll and quaternion of orientation;
- position and velocity data;
- barometer data: pressure and barometric height.

Instant or average variant of output data can be set, via INS GUI (see INS GUI User's Manual, section "4.2. Devices options" for details), and via command 0x03 Output_data_avrg (see Table 6.80) since INS firmware version 3.3.0.1.

If **instant output data** variant is chosen, then INS outputs the latest sample of INS data in the data message.

If **average output data** variant is chosen, then INS outputs the average of the internal 200 Hz samples between two data messages.

E.2. Variants of sensors output data in the Inertial Labs^{TM} INS

The gyros in INS measure angular rate (in deg/s), the accelerometers measure acceleration (in g) and the magnetometers measure magnetic field (in nT). Internal measurement rate is 800 Hz (or 2000 Hz). Then sensors data are processed to provide 200 Hz data for INS orientation calculation.

In order to make the INS data more versatile, there are three forms of INS sensors data output:

- instant sensors data;
- average sensors data;
- incremental sensors data

That appears in INS output data described in section "6.2 Output Data Formats of the Inertial Labs^{TM} INS in the Operating Modes".

Variant of sensors output data can be set, via INS GUI, in “Sensors options” window (see User Manual, section 4.4), and via command 0x77 Sensors_output_var (see Table 6.80) since INS firmware version 3.3.0.1.

If **instant sensors data** variant is chosen, then INS outputs the latest sample of sensors data in the data message.

If **average sensors data** variant is chosen, then INS outputs the average of the internal 800 Hz (or 2000 Hz) samples between two data messages.

If **incremental sensors data** variant is chosen, then INS outputs integrals of sensors data calculated for 800 Hz (or 2000 Hz) samples divided by time step of data output. In fact, gyros and accelerometers data are presented here as angle increment and velocity increment (Delta Theta and Delta Velocity) between two data messages divided by time step of data output. Magnetometers data are the same as at average sensors data variant.

In all variants of sensors data output their measurement units are the following:

- gyros data are in deg/s;
- accelerometers data are in g (g=9.8106 m/s^2);
- magnetometers data are in nT.

APPENDIX F.
Electrical interface of the housed INS unit with 12 or 19 pin connector

The Inertial Labs™ INS can have 12 pin or 19 pin connector for electrical connection to the host system:

- The Binder Series 723 male 12 pin connector, part # 09 0131 80 12 can be used in Inertial Labs™ INS with RS-232 interface of COM1 port.
- The Binder Series 723 male 19 pin connector, part # 09 0463 80 19, can be used in Inertial Labs™ INS with RS-422 interface of COM1 port.

The host system should have a cable with appropriate mating connector:

- For INS with RS-232 interface: the Binder Series 423, 425 or 723 female 12 pin connector (or cordset), part # 09 0130 70 12, # 99 5130 40 12, or # 79 6130 20 12.
- For INS with RS-422 interface: the Binder Series 423 or 723 female 19 pin connector (or cordset), part # 99 5662 00 19, # 99 5662 75 19 or # 09 0462 70 19, # 99 0462 75 19.

Fig. F.1 shows 12 pin connector pinout of the Inertial Labs™ INS with **RS-232** interface. Table F.1 contains pin diagram of this connector and appropriate color of wires in cable with mating Binder Series 425 Female plug, part # 79 6130 20 12.

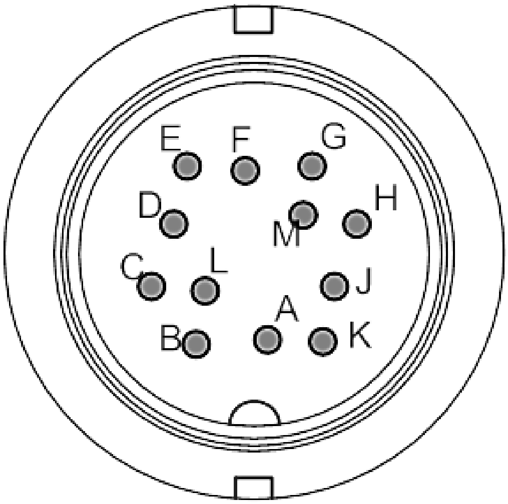


Fig. F.1. The Inertial Labs™ INS RS-232 connector pinout (mating side of the connector)

Table F.1. Pin diagram of the Inertial Labs™ INS RS-232 connector

Pin	Wire color	Signal
A	White	RS232 – RX2
B	Brown	RS232 – TX2
C	Green	RS232 – RX3
D	Yellow	RS232 – TX3
E	Grey	Power
F	Pink	Ground
G	Blue	RS232 – RX1
H	Red	RS232 – TX1
J	Black	PPS
K	Violet	MARK IN
L	Grey/pink	Do not connect
M	Red/blue	Do not connect

Note: Do not connect anything to pins #L or #M that are connected to INS PCB for firmware updates.

Fig. F.2 shows 19 pin connector pinout of the Inertial Labs™ INS with **RS-422** interface. Table F.2 contains pin diagram of this connector and appropriate color of wires in Alpha Wire cable part number 5478C with 16 conductors.

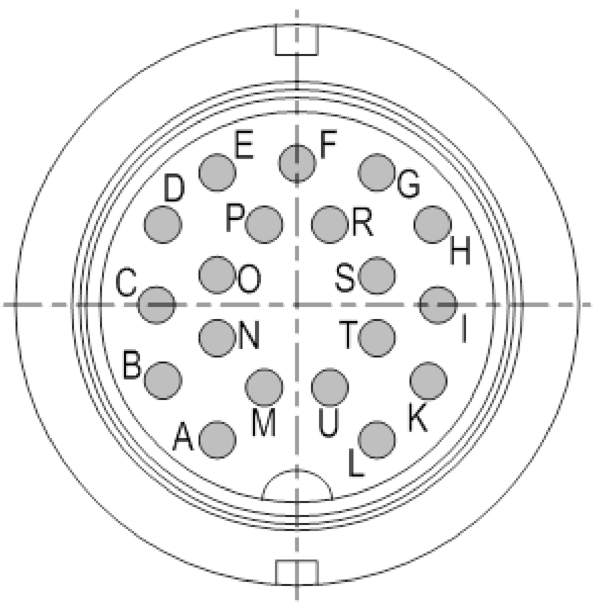


Fig. F.2. The Inertial Labs™ INS RS-422 connector pinout (mating side of the connector)

Table F.2. Pin diagram of the Inertial Labs™ INS RS-422 connector

Pin	Pairs color	Wire color	Signal
G	Yellow + Black	Yellow	RS422-A
R		Black	RS422-B
F	Orange + Black	Orange	RS422-X
E		Black	RS422-Z
P			Reserve1
D	Red + Black	Red	POWER
O		Black	GND
C	Blue+ Black, Brown+ Black	Blue	RS232-RX2
B		Brown	RS232-TX2
N		2xBlack	GND2
I	Green+ Black, White+ Black	Green	RS232-RX3
K		White	RS232-TX3
T		2xBlack	GND3
L	Red +White	Red	PPS
U		White	MARK IN
A			Do not connect
M			Do not connect
H			Reserve2
S			Reserve3

Note: Do not connect anything to pins #A and #M that are connected to INS PCB for firmware updates.

F.1 Connection of the Inertial Labs™ INS with 12-pin connector (RS-232 interface) to the host computer for tests

The delivery set for the INS with RS-232 interface electrical connection to PC is provided by the Inertial Labs and includes:

- interface cable for the Inertial Labs™ INS connection to the COM-ports of PC or another device, with branch wires for the Inertial Labs™ INS DC powering;

- COM-to-USB converter for connection of the INS to PC through the USB port;
- AC/DC adapter.

Also Inertial Labs INS GUI software is included in the delivery set for quick evaluation of the Inertial Labs™ INS.

For usual operations connect INS COM1 port to PC using interface cable as Fig. F.3 show. In addition, to use the raw GNSS data and NMEA messages COM2 should be connected with PC too. To provide the INS operation with DGPS mode the COM3 should be connected to Radio modem.

<u>Note</u>: According to customer's specific requirements the cable structure can be changed from 3 to 1 output.

For the Inertial Labs™ INS powering the AC/DC adapter can be used which receives the power from the 100…240V 50…60Hz AC power source.

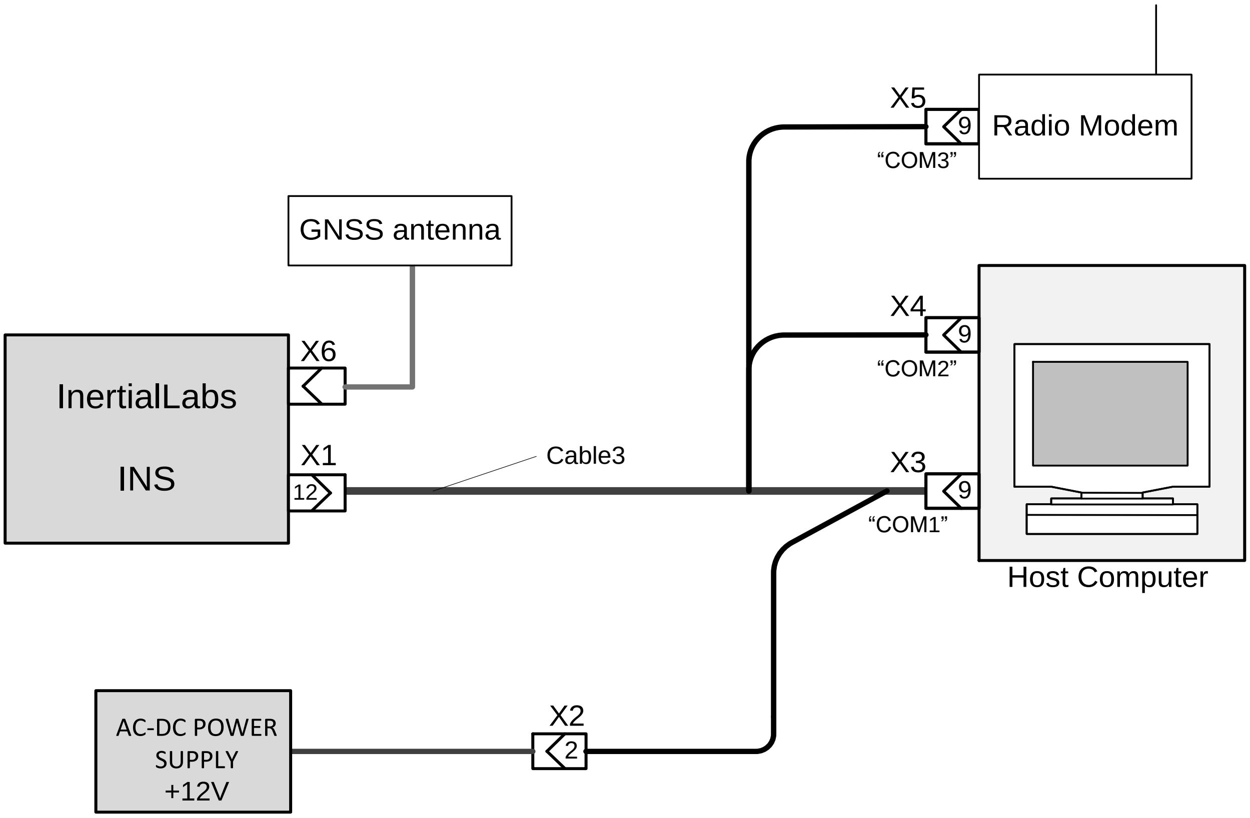


Fig. F.3. The diagram of electric connection of the Inertial Labs™ INS with DGPS mode to PC for tests

Fig. F.4 show the diagram of the interface cables 3 for the Inertial Labs™ INS connections to the COM-ports of host computer and to the DC power source.

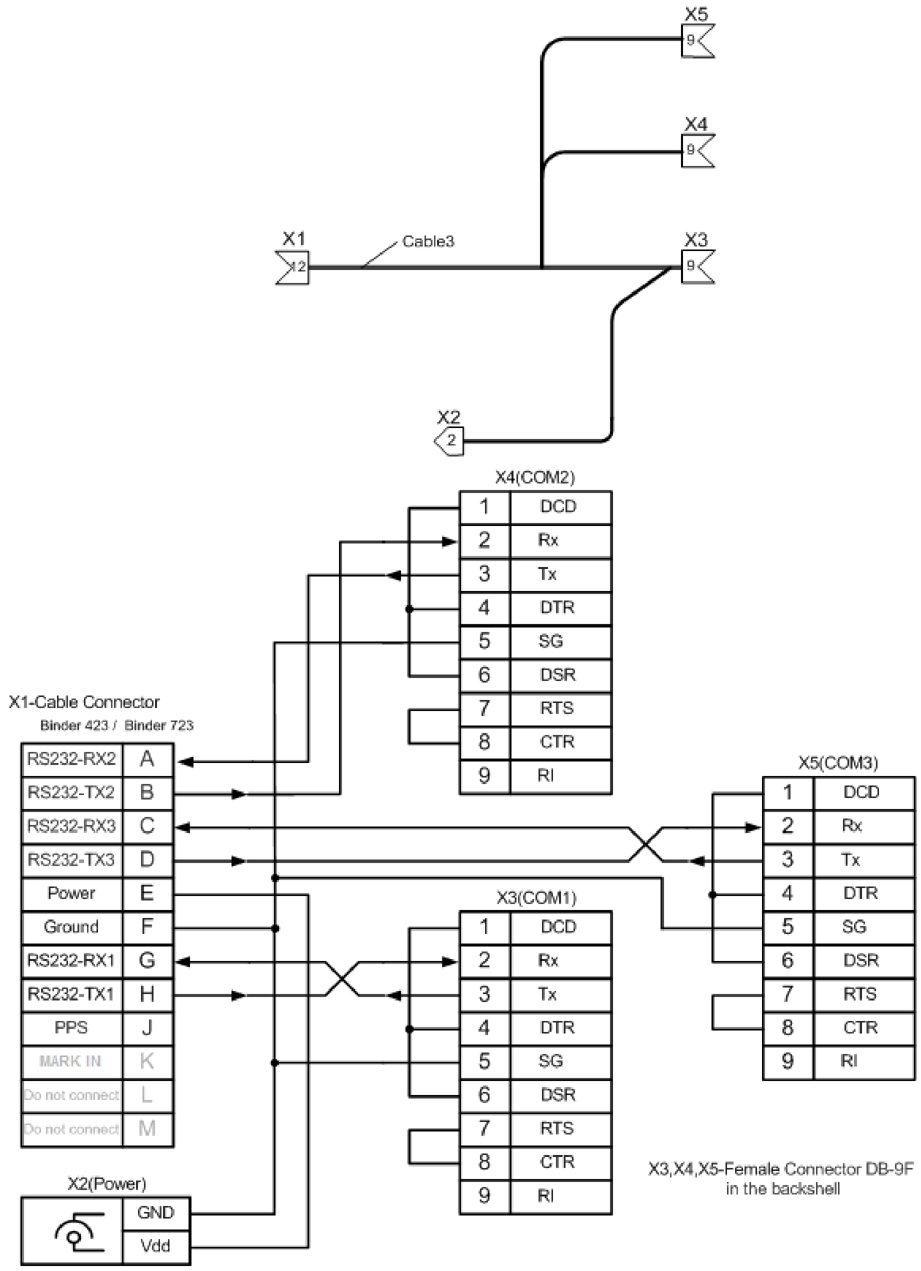


Fig. F.4. The diagram of the interface cable for the Inertial Labs™ INS connections to two COM-ports of the host computer, to radio modem and to the AC/DC adapter

F.2 Connection of the Inertial Labs™ INS with 19-pin connector (RS-422 interface) to the host computer for tests

Usual PC has no possibility of devices connection through RS-422 interface directly. Therefore, for the Inertial Labs™ INS with RS-422 interface connection to PC it is necessary to use some converter, for example Serial-to-USB MOXA 1130 converter, which is supplied with INS unit by the Inertial Labs. In other part above diagram Fig. F.3 is still valid. Fig. F.5 show the diagram of the interface cables 3 for the Inertial Labs™ INS with RS-422 interface connections to the COM-ports of host computer and to the DC power source.

Note: According to customer's specific requirements the cable structure can be changed from 3 to 1 output.

For the Inertial Labs™ INS powering the AC/DC adapter can be used which receives the power from the 100…240V 50…60Hz AC power source.

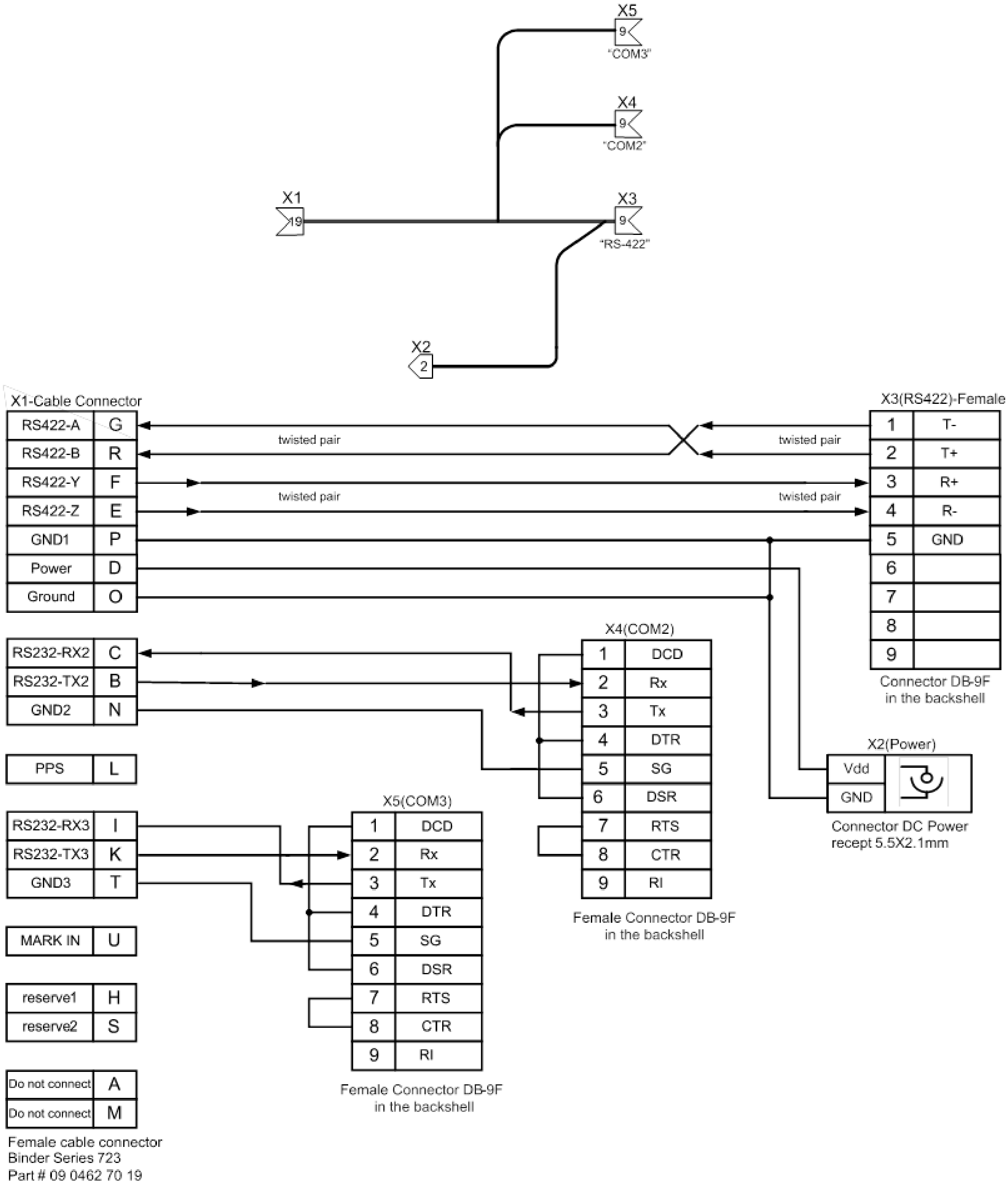


Fig. F.5. The diagram of the interface cable for the Inertial Labs™ INS RS-422 connections to two COM-ports of the host computer, to radio modem and to the AC/DC adapter

Inertial Labs
Attitude is Everything