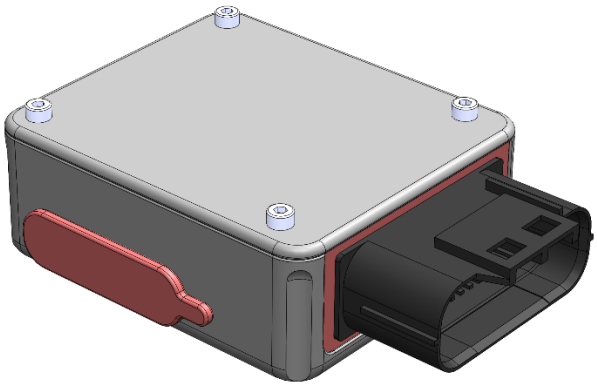
GUBENAV



High dynamic 100Hz GPS sensor & 6-Axis IMU with CAN 2.0 interface

1. Introduction

The **GUBENAV** is an advanced positioning and dynamics module, designed for professional applications in the motorsport, automotive, naval, precision agriculture, industrial, autonomous guided vehicles (AGV) and research and development (R&D) sectors.

The device combines a highly sensitive GPS-GNSS module with a 6-axis inertial measurement unit (IMU) (ICM-20649). Through a sophisticated *sensor fusion algorithm* , the GUBENAV calculates and transmits geographic coordinates, velocity, heading angle, and accelerations on the three spatial axes with an update rate of up to 100Hz. Data is transmitted in real time via a CAN 2.0 bus, ensuring maximum reliability and compatibility with existing acquisition systems.

2. Technical specifications

Positioning module: GPS-GNSS

Inertial Sensor (IMU): ICM-20649 (6-Axis: 3 accelerometers, 3 gyroscopes)

Calculation frequency (Sensor Fusion): Refresh output selectable at 1-10-50-100Hz

Processed data: Latitude, Longitude, Speed, Heading, Acceleration (X, Y, Z), Angular velocity (Pitch, Roll, Yaw)

Communication interface: Data sent via CAN 2.0B / Configuration sent via USB

Transceiver **:** High-Speed CAN 2.0B

Supported protocols: GUBENAV Proprietary and Standard NMEA2000

Supported CAN Baud Rates : 1 Mbps, 500 kbps, 250 kbps

Power supply voltage: 9V - 24V DC

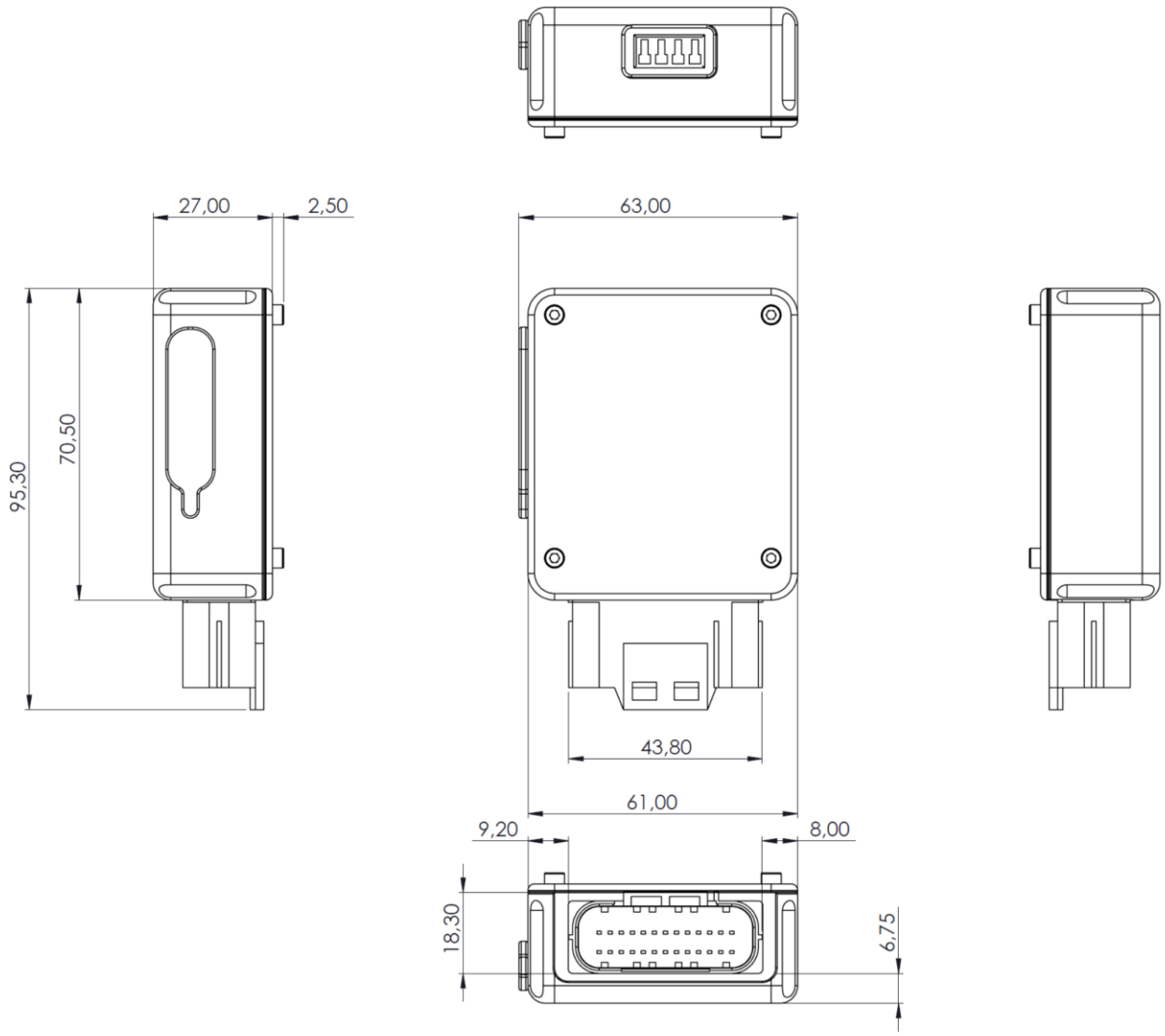
Current consumption: 150mA

Degree of protection: IPX7 (Fully protected against dust and temporary immersion)

The device is equipped with two integrated sensors that are essential for the dynamic analysis of the vehicle:

- **GPS/GNSS Receiver:** Integrated module with internal GPS antenna.
- **Inertial Measurement Unit (IMU):** 6-axis sensor (3-axis accelerometers + 3-axis gyroscopes) rigidly mounted on the internal PCB. It measures longitudinal/lateral accelerations and angular velocities of roll, pitch, and yaw.
  Accelerometers +/- 30g, internal sampling rate 1125Hz, low-pass filter 68.8Hz
  Gyroscopes +/- 2000°/sec internal sampling rate 1125Hz, low-pass filter 73.3Hz

3 . Dimensions



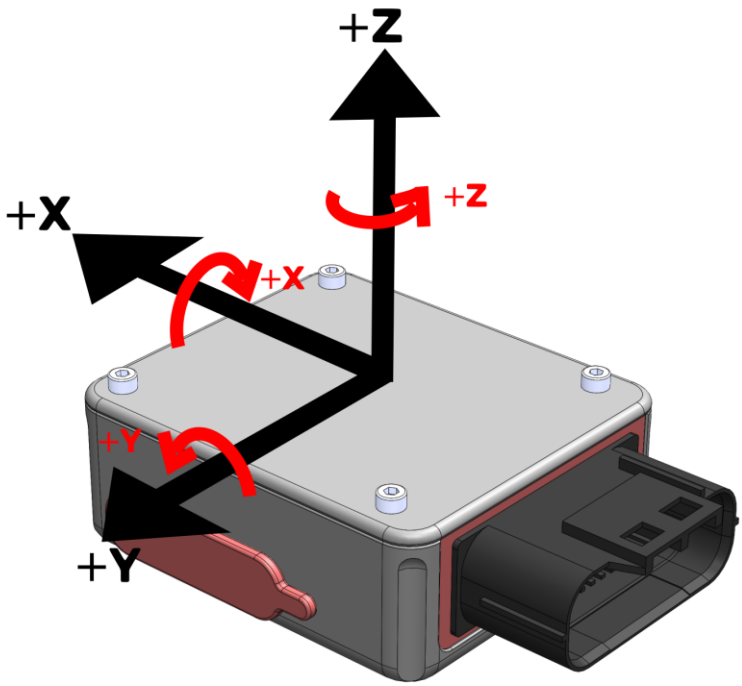
Note: The dimensions shown are expressed in millimeters (mm).

Mechanical specifications

To ensure system reliability (IPX7 certified against immersion) and correct reading of the inertial sensors (IMU), the device must be installed respecting the dimensions and tolerances reported below.

- Casing Material: Nylon PA12.
- External Dimensions (L x W x H): E.g. 93 mm x 63 mm x 30 mm.
- Weight: 180 grams.
- Protection rating: IPX7.
- Operating Temperatures: From -20°C to +85°C.

4. Positioning and Fixing (Guidelines)

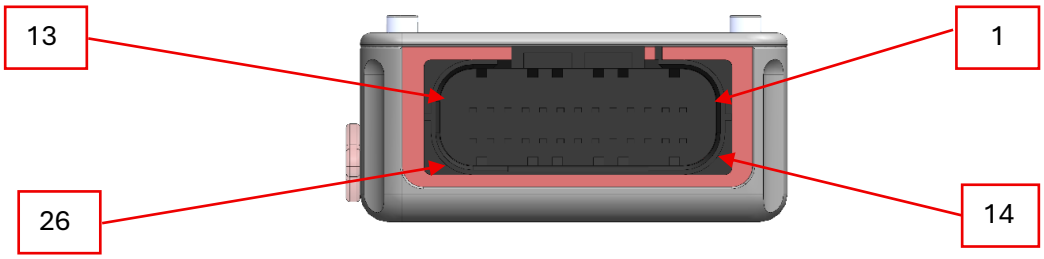


- IMU Orientation: Since the device contains three accelerometers and three gyroscopes, it must be mounted orthogonally to the vehicle axes
- Fix the GUBENAV in a place where the upper part of the sensor has as clear and uninterrupted a view of the sky as possible (e.g. roof, dome, roll-bar).
- Avoid installation under metal panels, thick carbon or near sources of strong electromagnetic interference (e.g. ignition coils, power inverters, radio transmitting antennas).

Orientation and vibrations

- Alignment: The sensor must be mounted perfectly aligned and square with the vehicle. The X+ axis must point in the direction of travel and the Y+ axis must point skyward.
- Vibration Management: GUBENAV should be rigidly mounted to the main frame or a structural bracket not subject to flexure or high-frequency resonance. Avoid mounting it on soft plastics or moving fairings, which would introduce mechanical "noise" that would invalidate the accelerometer data. If the frame is subject to extreme vibration (e.g., single-cylinder engines or heavy agricultural machinery), consider using lightweight anti-vibration mounts (silent mounts). block) specific for electronics.

5. Pinout

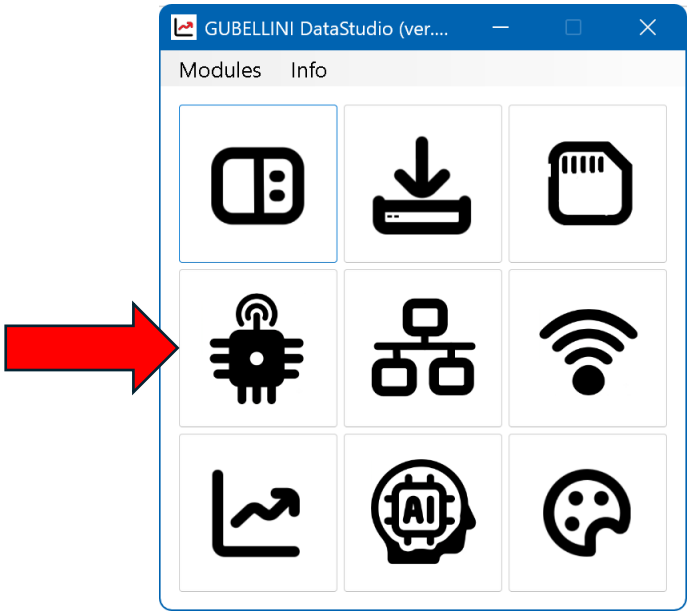


Main Connector : JAE Electronics MX23A26NF1 26-pin)

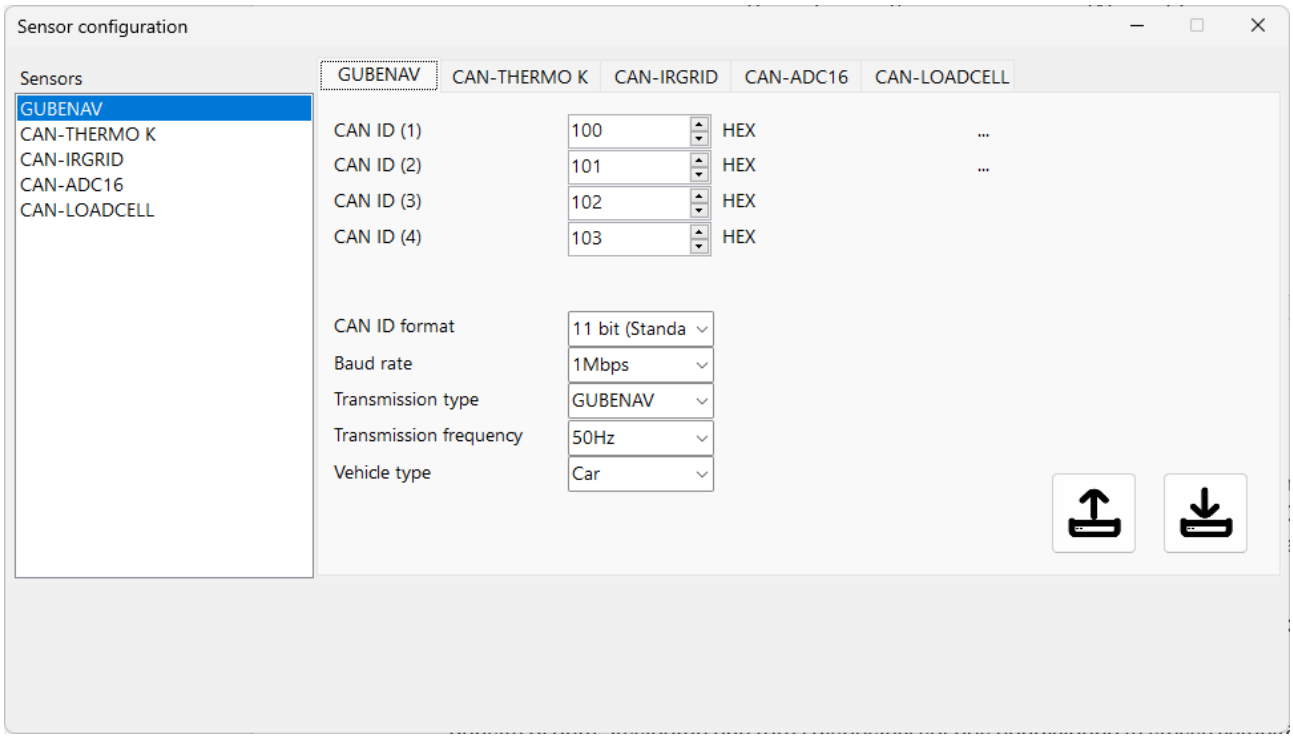
- Pin 11: CAN High
- Pin 12: CAN Low
- Pin 13: +V (9-24V power supply)
- Pin 26: GND (Ground)

6. Software Configuration (GUBELLINIDataStudio)

The GUBENAV is fully configurable using the free GUBELLINIDataStudio software suite. To configure the device, download and install the free suite and launch the program. Connect the device to your PC's USB-C port to power it. From the main window, click the **Sensor Configuration** button **.**



By connecting the device via USB, you can access the setup screen. As shown in the reference image, the "Sensor configuration " panel allows you to set:



- CAN ID (1) - (4): GUBENAV divides the data transmission into 4 distinct CAN identifiers (e.g. 100, 101, 102, 103 in HEX format). These fields allow you to customize the ID on which the CAN packets are transmitted (so as to avoid conflicts with other control units already present on your CAN network).
- CAN ID format: Select the identifier format between 11 bits (Standard) or 29 bits (Extended), depending on the requirements of your logger (e.g. GUBELOG-03 or third - party devices).
- Baud rate: Select the communication speed of the CAN network (e.g., 1Mbps for high-density networks). Make sure all devices on the bus share the same speed.
- Transmission type : Select the output protocol. Selecting GUBENAV activates the optimized proprietary protocol. (Alternatively, select NMEA2000 if required by the application.)
- Transmission frequency: Sets the CAN packet update rate (e.g. 50Hz, or up to the maximum value of 100Hz depending on the load allowed by the bus).
- Vehicle type : Define the vehicle type (e.g. Car) to optimize the internal “ sensor fusion” filters based on the expected dynamics (car, motorbike, boat).
- Read/Write Buttons: Use the lower right buttons (and) to read the current configuration from the device or write the new configuration to the GUBENAV's permanent memory.

7. Communication Protocols (Output Data)

The GUBENAV is designed to interface with any CAN network via two distinct standards:

7.1 GUBENAV Proprietary Protocol

An efficient protocol, ideal for motorsport applications or generic data acquisition (e.g. via GUBELOG-03). The data is structured in Little- Endian on the 4 configured CAN IDs:

- CAN ID 1 (e.g. 0x100): Contains position data (Latitude and Longitude).
- CAN ID 2 (e.g. 0x101): Contains data derived from GPS (Speed, Course Angle, Connected Satellites).
- CAN ID 3 (e.g. 0x102): Contains linear accelerations (X, Y, Z).
- CAN ID 4 (e.g. 0x103): Contains the gyroscope angular velocities (Pitch, Roll, Yaw). *(Note: The final manual will include tables with the conversion factors, offsets, and bit format for each channel.)*

This mode is designed for proprietary software interfaces (e.g., PC software) or specific loggers . The CAN IDs and frame types (Standard or Extended) are not fixed, but are configured by the user via PC. The complete package is divided into 4 distinct frames:

Frame 1: GPS coordinates

Byte(s)	Description	Data Format	Notes
0..3	Latitude	LongInt 32-bit	Value = LongInt / 10000000.0
4..7	Longitude	LongInt 32-bit	Value = LongInt / 10000000.0

Frame 2: Kinematics and Sensor Status

Byte(s)	Description	Data Format	Notes
0..1	Speed	16-bit integer (Word)	Resolution **0.01 km/h** . Value = Word / 100.0
2..3	Yaw / Heading	16-bit integer (Word)	Resolution **0.01°** . Value = Word / 100.0
4	Data Reliability	Byte	**1** = GPS Fix Valid, **0** = GPS Timeout (IMU Only).

Byte(s)	Description	Data Format	Notes
5..7	Padding	3 Bytes	Reserved, set to 0.

Frame 3: Accelerometer Data (ICM-20649)

Byte(s)	Description	Data Format	Notes
0..1	X-Axis Acceleration	16-bit integer	Value = Integer / 1000.0
2..3	Y-Axis Acceleration	16-bit integer	Value = Integer / 1000.0
4..5	Z-Axis Acceleration	16-bit integer	Value = Integer / 1000.0
6..7	Padding	2 Bytes	Reserved, set to 0.

Frame 4: Gyroscope Data (ICM-20649)

Byte(s)	Description	Data Format	Notes
0..1	Angular Velocity X-Axis	16-bit integer	Value = Integer / 10.0
2..3	Y-Axis Angular Velocity	16-bit integer	Value = Integer / 10.0
4..5	Z-Axis Angular Velocity	16-bit integer	Value = Integer / 10.0
6..7	Padding	2 Bytes	Reserved, set to 0.

7.2 NMEA2000 Protocol

The NMEA2000 protocol transmits standardized messages (PGNs) universally used in the marine, agricultural (ISOBUS compatible in some layers) and fleet sectors. When set to this mode, the GUBENAV automatically generates the PGNs needed for rapid positioning and attitude data, formatted in 29-bit according to industry specifications.

In this mode, the device interfaces with standard nautical networks using **29-bit extended CAN identifiers** and setting the message priority to **2.** The default Source Address of the node is *fixed at* **0x33** .

Two main frames are transmitted:

Frame 1: Location (PGN 129025)

- **CAN ID (Hex):** 0x09F50333
- **Length (DLC):** 8 bytes

Byte(s)	Description	Data Format	Notes
0..3	Latitude	32-bit integer (LongInt)	Highly accurate data derived from Sensor Fusion.
4..7	Longitude	32-bit integer (LongInt)	Highly accurate data derived from Sensor Fusion.

Frame 2: Course and Speed - COG & SOG (PGN 129026)

- **CAN ID (Hex):** 0x09F50433
- **Length (DLC):** 8 bytes

Byte(s)	Description	Data Format	Notes
0	SID (Sequence ID)	Byte	Fixed at 0.
1	COG Reference	Byte	0 = True North.
2..3	COG (Course Over Ground)	16-bit integer (Word)	Resolution **0.0001 rad** .
4..5	SOG (Speed Over Ground)	16-bit integer (Word)	Resolution **0.01 m/s** .
6..7	Reserved	2 Bytes	Fixed NMEA value 0xFFFF.

7.3 Code for the .DBC (CAN Database) file

You can copy the following text and save it in a text file renamed gubenav.dbc . *Note: In your code, the CAN IDs are dynamic (CAN ID 1, CAN ID 2, etc.). In the DBC file, I've inserted temporary default IDs (e.g., 256 in decimal for 0x100).*

How to interpret the .DBC file (quick explanation for the manual):

- @1+: Indicates Intel (Little- Endian) format , Unsigned .
- @1-: Indicates Intel (Little- Endian) format , Signed .
- (0.01,0): Represents the *Multiplication Factor* (0.01) and the *Offset* (0) respectively. The customer software will automatically perform the calculation: (CAN_Value * 0.01) + 0.
- In the DBC file, the IMU data is left with a multiplier (1.0) and the unit of measurement in LSB. This allows customers to apply the correct divisor in their analysis software depending on the dynamic range (FSR) they decide to set on the device.

Start of .DBC file code

```
VERSION ""


NS_ :

NS_DESC_

CM_

BA_DEF_

BA_

VAL_

CAT_DEF_

CAT_

FILTER

BA_DEF_DEF_

EV_DATA_

ENVVAR_DATA_

SGTYPE_

SGTYPE_VAL_

BA_DEF_SGTYPE_

BA_SGTYPE_

SIG_TYPE_REF_

VAL_TABLE_

SIG_GROUP_

SIG_VALTYPE_

SIGTYPE_VALTYPE_

BO_TX_BU_

BA_DEF_REL_

BA_REL_

BA_DEF_DEF_REL_
```

```
BU_SG_REL_
BU_EV_REL_
BU_BO_REL_
    SG_MUL_VAL_

BS_:

BU_: GUBENAV_Device PC_Logger

BO_ 256 GUBENAV_Coordinates : 8 GUBENAV_Device
 SG_ Latitude : 0|32@1- (0.0000001,0) [-90|90] " deg " Vector__XXX
SG_ Longitude : 32|32@1- (0.0000001,0) [-180|180] " deg " Vector__XXX

BO_ 257 GUBENAV_Kinematics : 8 GUBENAV_Device
SG_ Speed : 0|16@1+ (0.01,0) [0|600] "km/h" Vector__XXX
SG_ Yaw_ Heading : 16|16@1+ (0.01,0) [0|360] " deg " Vector__XXX
SG_ GPS_ Reliability : 32|8@1+ (1.0) [0|1] " boolean " Vector__XXX

BO_ 258 GUBENAV_Accelerometers : 8 GUBENAV_Device
SG_ Accel_
 SG_ Accel_ Y : 16|16@1- (1,0) [-32768|32767] "LSB" Vector__XXX
 SG_ Accel_ Z : 32|16@1- (1,0) [-32768|32767] "LSB" Vector __XXX

BO_ 259 GUBENAV_Gyroscopes : 8 GUBENAV_Device
 SG_ Gyro_
SG_ Gyro_ Y : 16|16@1- (1,0) [-32768|32767] "LSB" Vector__XXX
SG_ Gyro_ Z : 32|16@1- (1,0) [-32768|32767] "LSB" Vector__XXX

CM_ BU_ GUBENAV_Device " Gubenav Telemetric Unit ";
CM_ BO_ 256 "High resolution GPS/GNSS coordinate frame";
CM_ SG_ 256 Latitude "Resolution 10^-7 degrees (integer format)";
CM_ SG_ 256 Longitude "Resolution 10^-7 degrees (integer format)";
CM_ BO_ 257 "Frame Kinematics and Status";
CM_ SG_ 257 Speed " Over ground speed ";
CM_ SG_ 257 Yaw_Heading "Heading Angle";
CM_ SG_ 257 GPS_Reliability "1 = Fix Valid, 0 = IMU Only (Fix Timeout )";
CM_ BO_ 258 " ICM-20649 Accelerometer Raw Data. Divide by Sensitivity LSB based on FSR.";
CM_ BO_ 259 " ICM-20649 Gyroscope Raw Data. Divide by Sensitivity LSB based on FSR.";
```

```
VAL_ 257 GPS_Reliability 1 "FIX_OK" 0 "TIMEOUT_IMU_ONLY " ;
```

8. Resolution of the Problems (Troubleshooting)

- No data on the CAN network: Check that the 120 Ohm terminating resistors are present on the CAN backbone. Make sure that the Baud Rate configured in GUBELLINIDataStudio matches that of the receiver.
- GPS coordinates invalid or slow to update: Verify that the sensor is not covered by shielding materials. Wait for the initial *cold start* (up to 2 minutes outdoors) for satellite constellation acquisition.
- Incorrect or very noisy accelerometer data: Check the rigidity of the mechanical mount. An unsuitable mount can amplify engine vibrations, saturating the IMU reading.

GUBELLINI s.a.s. di Diego Gubellini & C.

Via Euridia Bergianti 10B, 40059 Medicina (BO) Italy | VAT: IT 03466001207

Web: https://www.gubellinielectronics.com **|** https://www.gripone.com
Mail: info@gubellinielectronics.com | info@gripone.com