CS-GN300

GNSS-Aided Inertial Navigation System

CTi Sensor, Inc.

Document Revision 1.1

CTi Sensors Technical Document

This is our CS-GN300 product-specific technical datasheet. The following information is available to assist CTi Sensors customers in product development.

Technical Support Contact Information:

Website: www.CTiSensors.com

Email: info@ctisensors.com

Phone: (440) 264-2370

TABLE OF CONTENTS

1. INTRODUCTION **6**
1.1. FEATURES 6
1.2. APPLICATIONS 6

2. SPECIFICATIONS **7**
2.1. PERFORMANCE 7
2.2. GNSS RECEIVER 7
2.3. ACCELEROMETER 8
2.4. GYROSCOPE 8
2.5. MAGNETOMETER 8
2.6. SYSTEM 9
2.7. MECHANICAL 9
2.8. TERMINAL ASSIGNMENT 10

3. COORDINATE SYSTEMS **11**
3.1. SENSOR COORDINATE FRAME 11
3.2. LATITUDE, LONGITUDE, ALTITUDE (LLA) 12
3.3. NORTH, EAST, DOWN (NED) 12

4. FACTORY CALIBRATION **13**
4.1. ACCELEROMETER AND GYROSCOPE 13
4.2. MAGNETOMETER 13
4.3. TEMPERATURE CALIBRATION 13

5. OPERATION **14**
5.1. MOUNTING 14
5.2. GNSS ANTENNA SETUP 14
5.3. OPERATION MODES 15
5.4. SYNCHRONIZATION SIGNALS 15
5.5. OUTPUT CONFIGURATION 15

6. COMMUNICATION **16**
6.1. CTI SENSOR CONNECT SOFTWARE 16
6.2. ASCII MESSAGE FORMAT 18
6.3. BINARY MESSAGE FORMAT 20
6.4. 8-BIT CHECKSUM 21
6.5. CONFIGURATION COMMANDS 22

7. MECHANICAL **25**
7.1. DIMENSIONAL DRAWING 25
7.2. ALIGNMENT HOLES 26
7.3. SENSOR MOUNTING 27
7.4. CONNECTOR SPECIFICATIONS 28
7.5. CABLE SPECIFICATIONS 29
7.6. GNSS ANTENNA SPECIFICATIONS 30

8. PART NUMBER **31**
8.1. DEVICE PART NUMBER 31
8.2. ACCESSORIES PART NUMBER 31

9. REVISION HISTORY **32**

10. WARRANTY INFORMATION **32**

[This Page Intentionally Left Blank]

1. Introduction

The CS-GN300 is a small-sized, high-performance GNSS-Aided Inertial Navigation System (GNSS-INS). The CS-GN300 combines the latest in inertial and GNSS sensors with advanced Kalman filter fusion algorithms into a beautifully compact form factor. Inside there are 3-axis accelerometers, 3-axis gyroscope, and 3-axis magnetometer sensors alongside the latest in GNSS multi-constellation GNSS receivers. A continuous 200 Hz stream of 3D position, velocity, and orientation are available via the embedded navigation algorithm.

The CS-GN300 is easily configured for many applications and includes all that is needed to hit the ground running. Use the included CTi Sensor Connect GUI software to begin testing within minutes or integrate the sensor into your own data acquisition workflow. It has a robust aluminum housing and several interface options such as USB, RS232, RS485, RS422, and 3.3V TTL UART.

1.1. Features

- **Sensor Fusion:** Combined GNSS and inertial navigation solution
- **High Data Rate:** 200Hz navigation solution (position, velocity, pitch, roll, heading)
- **High Performance Sensors:** state-of-the-art 3-axis MEMS gyroscope, accelerometer, and magnetometer
- **Measuring Range:** roll, heading/yaw: ±180°, pitch: ±90°
- **Multi-Constellation Receiver:** GPS, GLONASS, BeiDou, Galileo, SBAS, and QZSS
- **Industry Standard Parts:** IP 67 compliant connector, cable, and housing
- **Robust Enclosure:** aluminum housing and connectors
- **Low Power Consumption:** < 1 W (180 mA @ 5 V, including active antenna)

1.2. Applications

- Mapping and surveying
- Automotive and ground-vehicle testing
- Off-road vehicle testing
- Marine applications
- Unmanned Autonomous Vehicles (UAVs)
- Simulation Localization And Mapping (SLAM)
- Beyond Visual Line Of Sight navigation (BVLOS)
- Precision agriculture
- Construction applications
- Machine control and automation
- Robotics

2. Specifications

2.1. Performance

Table 1. Angles and Position

Parameter	Value
Range	Roll and heading/yaw: ±180° Pitch: ±90°
Static accuracy (RMS)	Pitch and roll: < 0.05° Heading[1]: 1.5°
Dynamic accuracy (sufficient motion)	Pitch and roll: 0.15°, 1σ Heading: 0.25°, 1σ
Angular resolution	Pitch and roll: < 0.003° Heading: < 0.1°
Zero offset error (pitch and roll)	±0.02° (@20°C)
Offset change versus temperature (pitch and roll)	±0.002°/°C (typical)
Horizontal position accuracy	1.5 m CEP
Vertical position accuracy	2.0 m
Velocity accuracy	0.05 m/s
Free inertial position drift	3.0 cm/s^2

2.2. GNSS Receiver

Table 2. GNSS Receiver

Parameter	Value
Receiver type	184-channel GPS L1C/A L2C, GLO L1OF L2OF, GAL E1B/C E5b, BDS B1I B2I, QZSS L1C/A L1S L2C, SBAS L1C/A
Constellations	GPS, GLONASS, Galileo, BeiDou, SBAS, QZSS
Time –to –first –fix (TTFF)	Cold start: 25 second, Reacquisition: 2 second
Altitude limit	80,000 m
Velocity limit	500 m/s

[1] Measured with a valid magnetometer calibration in a suitable magnetic environment

2.3. Accelerometer

Table 3. Accelerometer

Parameter	Value
Range	±2 g/±4 g/±8 g selectable
Zero offset error	< ±0.5 mg (@20°C)
In-run bias stability	X & Y: < 5 µg, Z: < 10 µg
Velocity random walk	X & Y: 5.3 mm/sec/√hr, Z: 7.7 mm/sec/√hr
Nonlinearity	±0.1 % FS
Bias change versus temperature	±0.02 mg/°C (typical)
Noise density	25 µg/√Hz (@200Hz)

2.4. Gyroscope

Table 4. Gyroscope

Parameter	Value
Range	±125/250/500/1000/2000 °/s selectable
In-run bias stability	10 °/hr
Angle random walk	0.35 °/√hr
Initial bias error	< 0.1 °/s (@±500°/s range)
Noise density	0.007 dps/√Hz (@ 10 Hz)
Nonlinearity	0.1 % FS
g- Sensitivity	0.1 °/s/g
Bias change versus temperature[2]	±0.05 °/s/°C

2.5. Magnetometer

Table 5. Magnetometer

Parameter	Value
Range	±800 µT
Nonlinearity	±0.2 % FS
Noise density	0.06 uT/√Hz (@ 100 Hz)
Bandwidth	100 Hz

[2] CS-GN300 has gyro in-run bias compensation

2.6. System

Table 6. System

Parameter	Value
Power source	4.5-38 VDC
Power consumption (Including GNSS antenna)	900 mW (180 mA @ 5 V)
Data format	ASCII and binary
Output data rate	1 Hz to 200 Hz selectable
GUI software	CTi Sensor Connect®
Serial interface options	RS232, RS422, RS485, USB, UART, RS485 with multi-drop networked
LED indicators	LED1[3], red: CPU heartbeat, flashing at 1 Hz LED2, green: GNSS 3D fix; On: 3D fix, Off: no 3D fix LED3, green: Operation mode; setup: off, Initialization: blinking, aided nav: on
Temperature sensor resolution	0.5°C

2.7. Mechanical

Table 7. Mechanical

Parameter	Value
Protection	IP 67 (housing, connector and cable)
Dimension	2.16” x 2.16” x 1.18” (55 x 55 x 30 mm)
Material	Enclosure: anodized aluminum Connector: brass/nickel Cable molded head: TPU Cable carrier: TPU or nylon Conductor insulation: PVC
Temperature range	-40°C to +85°C (-40°F to +185°F)
Connection	Cable gland connector M8, 10-contact (female)

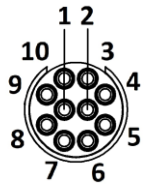
[3] Please see figure 5 for LED numbers.

2.8. Terminal Assignment

Table 8. Terminal Assignment

Connector	RS232/UART/USB[4]	RS422	RS485	Wire Color
Pin 1	+Vin			Orange
Pin 2	GND			Yellow
Pin 3	TX	TX+	D+	Green
Pin 4	–	TX-	D-	Blue
Pin 5	RX	RX+	D+	Black
Pin 6	–	RX-	D-	White
Pin 7	GPS-PPS			Red
Pin 8	SYNC-IN			Brown
Pin 9	SYNC-OUT			Purple
Pin 10	NA			Gray

Device:
10-contact (female)



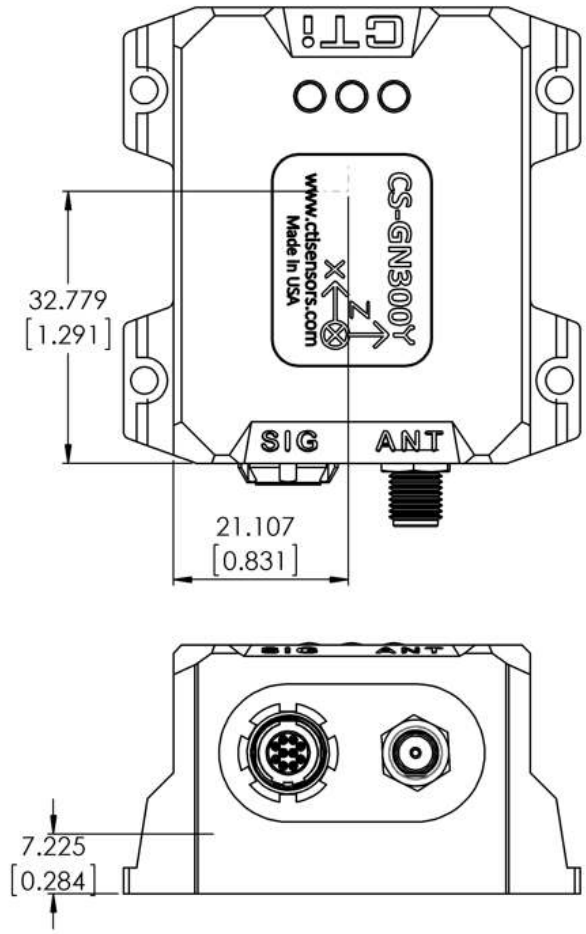
[4] USB uses UART interface and a UART to USB cable.

3. Coordinate Systems

Inside the CS-GN300 are multiple sensor types (e.g. accelerometer, gyroscope, magnetometer, GNSS) and the device navigation outputs include multiple measurement types (e.g. position, velocity, orientation). This section details the basic coordinate frames used by the CS-GN300.

3.1. Sensor Coordinate Frame

The sensor coordinate frame is a frame-of-reference attached to the CS-GN300. This is sometimes referred to as the "Center of Navigation" or "IMU Center of Measurement" and is shown in Figure 1. All inertial measurements are referenced with respect to this origin. The CS-GN300 position and velocity outputs represent the position and velocity of this point. If utilized, the GNSS antenna offset is also specified with respect to this point and expressed in the sensor coordinate frame.



Millimeter [Inch]

Figure 1. Location of the IMU Center of Measurement

3.2. Latitude, Longitude, Altitude (LLA)

The position output is expressed in geodetic latitude, longitude, and altitude coordinates. The earth is modeled as an ellipsoid and the ellipsoid model used is WGS 84, the standard model for GPS receivers. Any physical point of interest can be projected onto the surface of the reference ellipsoid, where the connecting vector is normal to the reference ellipsoid. From here we can define the position using three parameters:

- Latitude: along the meridian plane, the angle between the equatorial plane and the ellipsoidal normal vector
- Longitude: along the equatorial plane, the angle between the prime meridian and the projection of the physical point onto the equatorial plane
- Altitude: the distance above the ellipsoidal surface, along the normal vector, to the physical point of interest

Further details on the ellipsoidal model, sign convention, and alternative coordinate frames (e.g. Earth Centered Earth Fixed or ECEF) can be found in navigation or GNSS reference books.

3.3. North, East, Down (NED)

The velocity output is expressed in a local North, East, Down (NED) rectangular coordinate system. This is known as a local geodetic or tangent plane coordinate system. The coordinate system moves with the sensor, as projected onto the earth model, and is defined as tangent to the surface of the reference ellipsoid.

Note that velocity expressed as NED is independent of the sensor orientation. There may be applications where velocity is needed to be expressed in the sensor coordinate frame. For example, to specifically know the forward or lateral velocity. This transformation is possible using the sensor orientation. Our support team can assist users who require this.

4. Factory Calibration

The CS-GN300 internally stores and applies the calibration parameters determined at manufacturing. The factory calibration process involves three general steps:

1. **Defining the Sensor Error Model:** the sensor error model connects the reference and measured quantities.
2. **Collecting Calibration Data:** the data collection routine is designed to observe all error model parameters.
3. **Estimating Calibration Parameters:** a least-squares fitting process is used to estimate all error model parameters.

4.1. Accelerometer and Gyroscope

The following sensor error model is used for both accelerometer and gyroscope sensors:

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} 1+s_X & 0 & 0 \\ 0 & 1+s_Y & 0 \\ 0 & 0 & 1+s_Z \end{bmatrix} \begin{bmatrix} 1 & m_{yx} & m_{zx} \\ m_{xy} & 1 & m_{zy} \\ m_{xz} & m_{yz} & 1 \end{bmatrix} \left(\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} - \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} \right)$$

Where:

- Subscript *"t"* and *"m"* are the true and measured quantities, respectively.
- With the exception of sensor bias, all other error parameters are unitless. The bias parameters have the same unit as the sensor output (e.g. m/s^2 for the accelerometer or rad/s for the gyroscope).
- The scale-factor parameter is chosen to represent error (i.e. a perfect sensor has a scale factor parameter equal to 0).

4.2. Magnetometer

Magnetometer calibration requires special treatment. Unlike the accelerometer and gyroscope, the magnetometer is impacted by the nearby sensor environment. The magnetometer calibration is designed to compensate for all disturbances to the Earth magnetic field. These disturbances may be both internal and external to the sensor. For example, installing the CS-GN300 in proximity to a ferrous bolt can introduce a magnetic bias and degrade the ability to sense the undisturbed Earth magnetic field. As such, the CS-GN300 magnetometer is both factory-calibrated, and the user is empowered with a magnetometer calibration utility. This calibration process can be completed by the user using the CTi Sensor Connect GUI utility (See section 6.1).

4.3. Temperature Calibration

Calibration for thermal impact is not part of the standard calibration. Applications operating over wide temperature ranges may benefit from thermal calibration. If interested, please contact support to discuss thermal calibration options.

5. Operation

A simple setup and usage experience is our goal and the CS-GN300 is configured to operate well for a range of applications. Improved performance is possible if certain operation guidelines are followed. This section briefly describes these guidelines.

5.1. Mounting

1. The CS-GN300 should be rigidly attached to the vehicle or platform.
2. Excessive vibration can potentially degrade navigation performance.
3. Proximity to ferrous materials or electromagnetic noise sources can negatively impact the magnetic heading.

5.2. GNSS Antenna Setup

1. The GNSS antenna should be rigidly attached to the same vehicle or platform as the sensor.
2. The antenna should be installed for maximum sky visibility and above all (to the extent possible) nearby objects.
3. Proximity to other radio frequency (RF) transmission sources (e.g. telemetry radio) can negatively impact the GNSS signal-to-noise ratio.
4. If the separation between the GNSS antenna and the CS-GN300 sensor is significant (e.g. > 0.5 m), the GNSS antenna offset, also known as the GNSS lever arm, should be configured using the GNSS lever-arm command (table 9). This involves defining the vector between the CS-GN300 sensor and the GNSS antenna, expressed in the sensor coordinate frame, as shown in Figure 2. In this example scenario, the GNSS antenna offset is ℓ_{ax}, ℓ_{ay}, and ℓ_{az} (here ℓ_{az} is a negative number). The command `$nGLA,` ℓ_{ax}, ℓ_{ay}, ℓ_{az} `*<cr>` can be used to configure the GNSS lever arm.

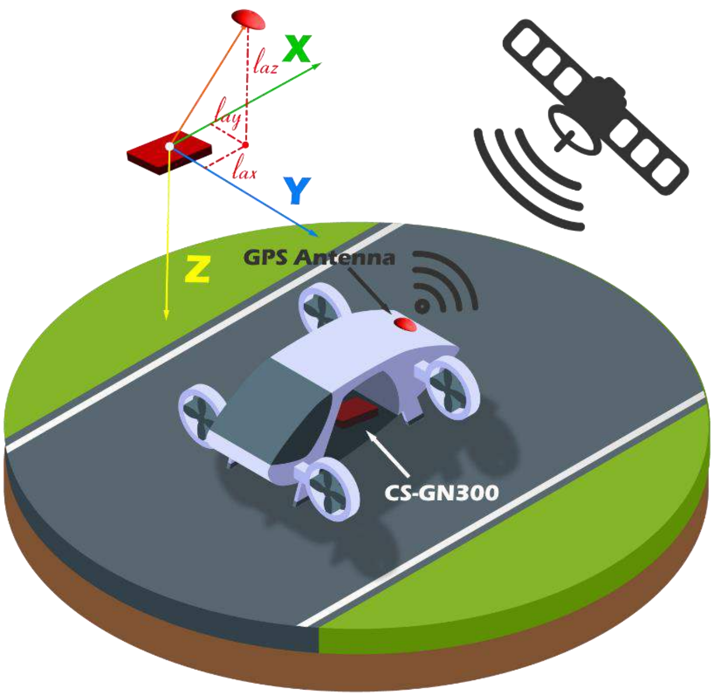


Figure 2. Example GNSS antenna lever-arm scenario

5.3. Operation Modes

The CS-GN300 automatically switches between these three modes:

1. Setup
2. Initialization
3. Aided Navigation

Once the sensor is powered, a suitable GNSS position fix is required to complete *Initialization* and switch to *Aided Navigation* mode. Short term losses to GNSS are seamlessly handled. However, the extended loss of GNSS measurements causes the navigation states to drift and become unreliable. In this case the device will automatically switch back to *Initialization* mode until reliable GNSS measurements are regained. The sensor mode can be monitored via the device LEDs (Table 6).

5.4. Synchronization Signals

5.4.1. GNSS-PPS

GNSS pulse per second is available on pin 7 of the connector. This pin is a TTL voltage level (3V) output directly connected to the PPS (pulse per second) pin on GNSS receiver. The signal is synchronized on rising edge with the pulse width of 100 ms. It triggers on a GPS PPS event (1 Hz) when a 3D fix is valid.

5.4.2. SYNC-IN

SYNC-IN is on pin 8 of the connector and enables polling a selected message using an input pulse. Once enabled (see Interface Commands, Table 9), the CS-GN300 transmits the selected message on rising edge of SYNC-IN signal.

5.4.3. SYNC-OUT

The SYNC-OUT signal can be programmed (see Interface Commands, Table 9) by user with one of the following functionalities.

IMU-START: The signal on SYNC-OUT triggers at start of IMU sampling.

IMU-READY: The signal on SYNC-OUT triggers when IMU measurements are available.

INS-READY: The signal on SYNC-OUT triggers when attitude measurements are available.

The pulse width of the signal is 1 ms.

5.5. Output Configuration

The default sensor output is GNSS-aided navigation ASCII message. The sensor can be configured to output either ASCII or binary messages. Generally, the binary messages have more detailed outputs for advanced users. The best way to explore these messages is using the CTi Sensor Connect Software. Once a suitable setup is found, the configuration can be saved to flash memory. See section 6.5 for the commands for both saving or resetting the default configuration.

6. Communication

The CS-GN300 can transmit data in both ASCII and binary formats. All commands and responses to commands are transmitted in ASCII. The fastest way to view and log data is to use the CTi Sensor Connect Software package. Alternatively, the serial and binary data formats detailed in this manual can be used to develop a custom interface.

6.1. CTi Sensor Connect Software

CTi Sensor Connect® is the Graphical User Interface (GUI) software provided by CTi Sensor Inc. for visualization, device configuration, and data logging for CS-100 product series, CS-200 product series, and CS-GN300. CTi Sensor Connect® is designed to be intuitive to users. The package can be downloaded from the CTi Sensors website.

Launching CTi Sensor Connect and prior to connecting any sensors, user can select the device COM port via the dropdown box and configure the Baud Rate (default: 115200 bps) to initialize serial communication the sensor. Figure 3 shows a screenshot after establishing serial communication. Figure 4 illustrates additional features of the CTi Sensor Connect GUI.

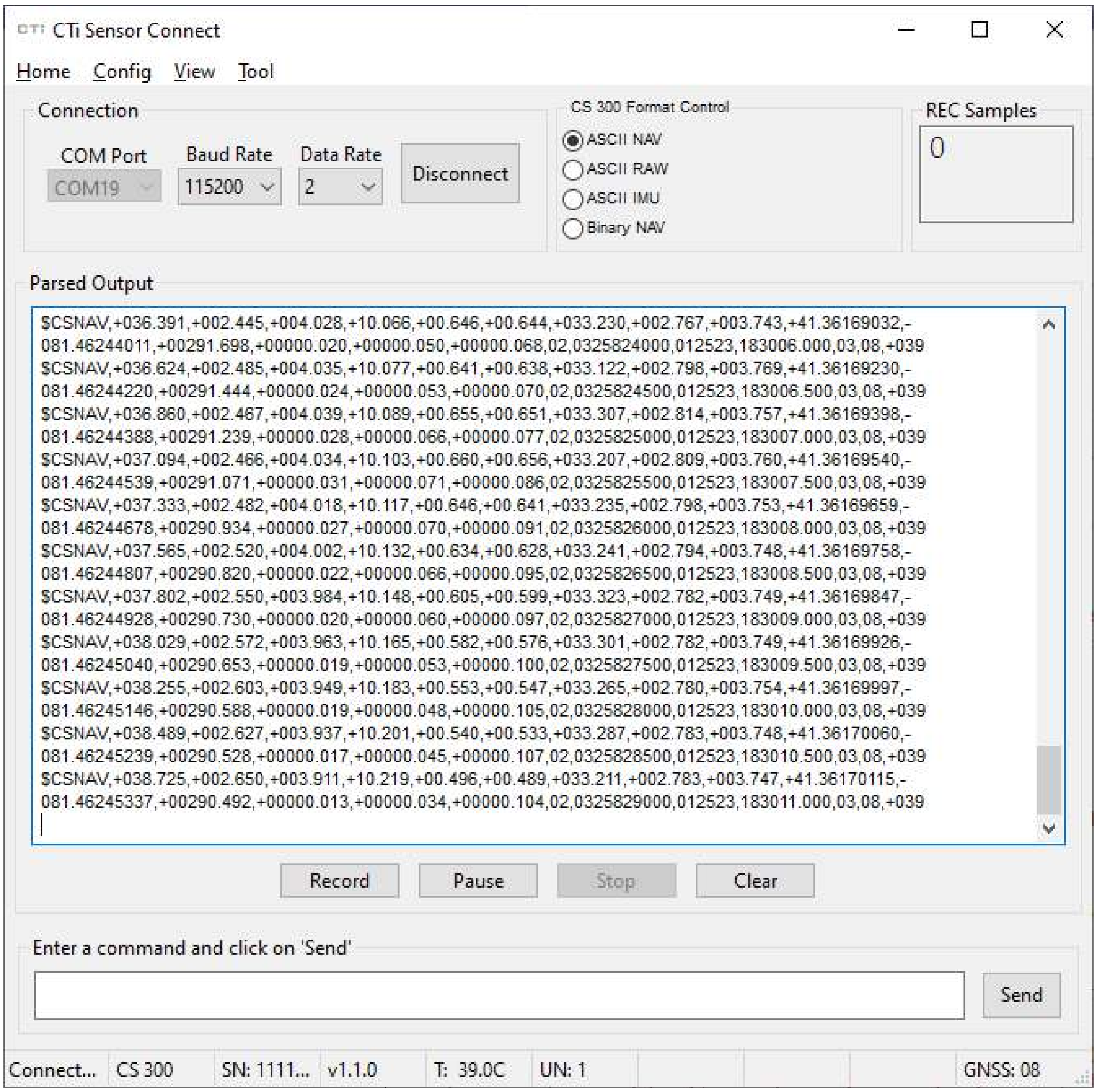


Figure 3. CTi Sensor Connect® Software, main screen

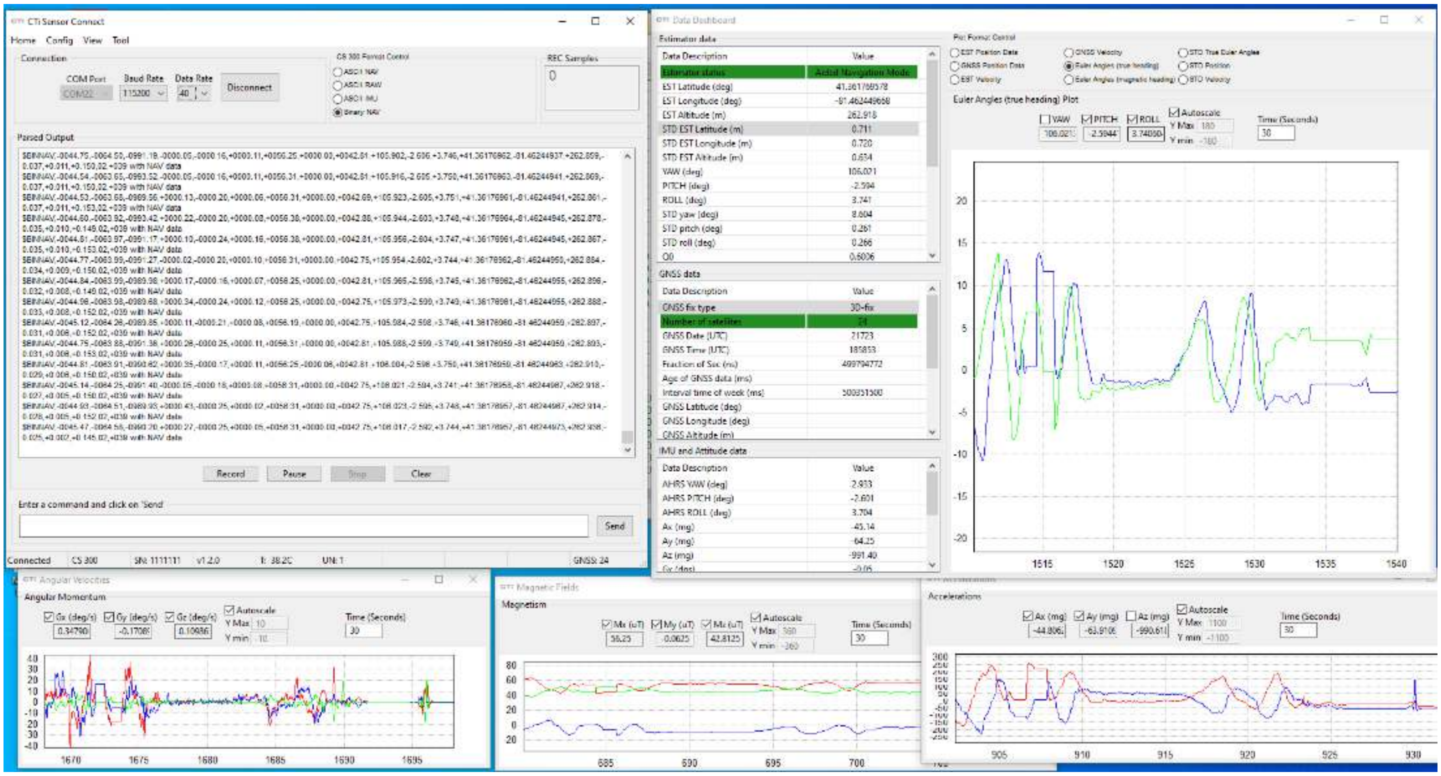


Figure 4. Additional features of the CTi Sensor Connect®

6.2. ASCII Message Format

The CS-GN300 uses the following ASCII format, based on the widely used NMEA 0183 protocol for data output. The following three ASCII messages are available on CS-GN300.

- GNSS-aided navigation message (default):
  $CSNAV,yaw,pitch,roll,ahrs_yaw,ahrs_pitch,ahrs_roll,est_lat,est_lon,est_alt,est_velN,est_velE,est_velD,est_status,i_tow,date,time,gnss_fix_type,num_gnss,T*CC<CR><LF>
- Sensor data message (optional):
  $CSIMU,$A_X$,$A_Y$,$A_Z$,$G_X$,$G_Y$,$G_Z$,$M_X$,$M_Y$,$M_Z$,T*CC<CR><LF>
- GNSS and imu raw data message(optional):
  $CSGIR,$A_X$,$A_Y$,$A_Z$,$G_X$,$G_Y$,$G_Z$,$M_X$,$M_Y$,$M_Z$,i_tow,date,time,gnss_fix_type,num_gnss,gnss_lat,gnss_lon,gnss_alt,h_acc,v_acc,gnss_velN,gnss_velE,gnss_velD,s_acc,age_of_gnss_data,T*CC<CR><LF>

Where:

- yaw, pitch, roll: ZYX-Euler-angles (yaw is true heading) from GNSS-aided INS Kalman filter in degree
- ahrs_yaw, ahrs_pitch, ahrs_roll: magnetic heading, pitch and roll angles from AHRS complementary filter in degree
- est_lat: estimated latitude in degree
- est_lon: estimated longitude in degree
- est_alt: estimated altitude in meter
- est_velN, est_velE, est_velD: NED estimated north, east, and down velocities in m/s
- est_status: mode of estimator operation
  - 0: Setup mode
  - 1: Initialization mode
  - 2: GNSS-aided navigation mode
- i_tow: GPS time of week of the navigation epoch.
- date: month, day, and year in MMDDYY format (UTC date)
- time: hour, minute, and second in HHMMSS.ss format (UTC time)
- gnss_fix_type: GNSS fix type
  - 0: no fix
  - 1: dead reckoning only
  - 2: 2D-fix
  - 3: 3D-fix
  - 4: GNSS + dead reckoning combined
  - 5: time only fix
- num_gnss: number of GNSS satellites used in navigation solution
- A_X, A_Y, A_Z: X, Y and Z accelerations in milli g (three-axis accelerometer data)
- G_X, G_Y, G_Z: X, Y and Z angular velocities in deg/s (three-axis gyroscope data)
- M_X, M_Y, M_Z: X, Y and Z magnetic fields in micro-Tesla (three-axis magnetometer data)
- gnss_lat: latitude from GNSS receiver in degree
- gnss_lon: longitude from GNSS receiver in degree
- gnss_alt: altitude from GNSS receiver in meter
- h_acc: GNSS horizontal accuracy estimate in mm
- v_acc: GNSS vertical accuracy estimate in mm
- gnss_velN: GNSS north velocity in m/s
- gnss_velE: GNSS east velocity in m/s
- gnss_velD: GNSS down velocity in m/s
- s_acc: GNSS speed accuracy estimate in mm/s

- `age_of_gnss_data`: time between sending message and receiving latest GNSS data from GNSS module in milli second
- `T`: Internal temperature in degrees Celsius
- `CC`: Checksum (two ASCII characters)
- `<CR><LF>`: Carriage return, and line feed characters

ASCII message examples:

- GNSS-aided navigation message:
  - GNSS available
    - $CSNAV,+144.576,+000.007,+000.001,+31.892,+00.183,+00.182,+036.627,+000.008,+000.013,+41.36173454,-081.46245174,+00266.249,-00000.011,-00000.002,+00000.093,02,0163129600,012323,211831.600,03,18,-003*45
  - GNSS unavailable
    - $CSNAV, +nan, +nan, +nan, +nan, +nan, +nan,+055.732,-000.026,+000.010, +nan, +nan, +nan, +nan, +nan, +nan,01,0000018000,032220,000018.000,00,00,-010*4B

- Sensor data message:
  - $CSIMU,+0063.79,-0068.99,-0995.53,+0000.00,+0000.00,-4640.04,+0045.32,-0033.66,+0051.40,-003*5E

- GNSS and IMU raw data message:
  - GNSS available
    - $CSGIR,+0024.89,-0089.31,-0995.35,+0000.00,+0000.00,+3039.96,+0047.82,-0031.99,+0051.01,0163369500,012323,212231.500,03,20,+41.36174890,-081.46245680,+00264.891,000324,000266,+00000.011,-00000.013,+00000.004,00089,099,-002*51
  - GNSS unavailable
    - $CSNAV, +nan, +nan, +nan, +nan, +nan, +nan,+055.732,-000.026,+000.010, +nan, +nan, +nan, +nan, +nan, +nan,01,0000018000,032220,000018.000,00,00,-010*4B

6.3. Binary Message Format

The CS-GN300 has one binary message (binary navigation message) making all system data available in the following format. All data is sent with the first byte being the most significant.

Table 9. Binary Message

Offset	Type	Name	Scale	Unit	Description
0	uint8	Start character			The start character is 0xAA
1	uint8	Message type			The message type is 0xBE
2	uint8	Data length			The data length is 0x9E (158 in base 10)
3	float32	AHRS Yaw	1	deg	AHRS yaw (magnetic heading, complementary filter)
7	float32	AHRS Pitch	1	deg	AHRS pitch (complementary filter)
11	float32	AHRS Roll	1	deg	AHRS roll (complementary filter)
15	float32	A_X	1	milli g	Acceleration data at X-axis
19	float32	A_Y	1	milli g	Acceleration data at Y-axis
23	float32	A_Z	1	milli g	Acceleration data at Z-axis
27	float32	G_X	1	deg/s	Angular velocity at X-axis
31	float32	G_Y	1	deg/s	Angular velocity at Y-axis
35	float32	G_Z	1	deg/s	Angular velocity at Z-axis
39	int16	M_X	16	micro-Tesla	Magnetic field at X-axis
41	int16	M_Y	16	micro-Tesla	Magnetic field at Y-axis
43	int16	M_Z	16	micro-Tesla	Magnetic field at Z-axis
45	uint32	iTOW	1	milli second	GPS time of week of the navigation epoch
49	uint16	Year	1	year	Year (UTC)
51	uint8	Month	1	month	Month of year 1 … 12 (UTC)
52	uint8	Day	1	day	Day of month 1 … 31 (UTC)
53	uint8	Hour	1	hour	Hour of day 0 … 23 (UTC)
54	uint8	Minute	1	minute	Minute of hour 0 … 59 (UTC)
55	uint8	Second	1	second	Second of minute 0 … 59 (UTC)
56	int32	Fraction of second	1	nano second	Fraction of second, -1e9 … 1e9 (UTC)
60	uint8	Fix type	1		GNSS fix type (see section 6.2)
61	uint8	Num of satellites	1		Number of satellites used in solution
62	float32	Quaternion 0	1		Estimated quaternion 0
66	float32	Quaternion 1	1		Estimated quaternion 1
70	float32	Quaternion 2	1		Estimated quaternion 2
74	float32	Quaternion 3	1		Estimated quaternion 3
78	float32	Yaw	1	deg	Estimated yaw
82	float32	Pitch	1	deg	Estimated pitch
86	float32	Roll	1	deg	Estimated roll
90	float32	STD EST yaw	1	deg	Standard deviation of estimated yaw
94	float32	STD EST pitch	1	deg	Standard deviation of estimated pitch
98	float32	STD EST roll	1	deg	Standard deviation of estimated roll
102	float64	Latitude	1	deg	Estimated latitude
110	float64	Longitude	1	deg	Estimated longitude
118	float32	Altitude	1	meter	Estimated altitude

122	float32	STD EST latitude	1	meter	Standard deviation of estimated latitude
126	float32	STD EST longitude	1	meter	Standard deviation of estimated longitude
130	float32	STD EST altitude	1	meter	Standard deviation of estimated altitude
134	float32	Velocity North	1	m/s	Estimated velocity in north direction
138	float32	Velocity East	1	m/s	Estimated velocity in east direction
142	float32	Velocity Down	1	m/s	Estimated velocity in down direction
146	float32	STD EST VelN	1	m/s	Standard deviation of estimated velocity north
150	float32	STD EST VelE	1	m/s	Standard deviation of estimated velocity east
154	float32	STD EST VelD	1	m/s	Standard deviation of estimated velocity down
158	uint8	Estimator status	1		Status of estimator
159	int16	Temperature	16	Celsius	Temperature in degree Celsius
161	uint8	Checksum			8-bit checksum (see section 6.4)
162	uint8	Stop character			Stop character of message is 0x55

6.4. 8-Bit Checksum

The checksum is calculated in ASCII by XORing bitwise all bytes (each character is represented by 1 byte) between the $ and * (not including the $ or * characters) based on the NMEA standard. It results in two hexadecimal characters, which are sent in ASCII format.

In binary, the checksum is calculated by XORing bitwise all bytes (each number is represented by one byte) from AA to 55 (not including the AA or 55 bytes). The result is 1 byte of data, represented by two hexadecimal characters, which are sent in binary format.

The following example code calculates the checksum for both ASCII and binary messages.

```
unsigned char cti_checksum(unsigned char * msg)
{
     unsigned int i;
     unsigned char crc = 0;
     for (i = 0; i < strlen((char *)msg); i++)
          crc ^= msg[i];
     return crc;
}
```

6.5. Configuration Commands

The CS-GN300 Series uses a simple command format which allows the user to change the device configuration and request specific information or data. All commands start with a '$' character, and end with a '*' character followed by a carriage return. All responses end with a carriage return and newline character. Table shows the list of the interface commands for the CS-GN300 Series.

In the table below, the lowercase letter 'n' represents the unit number, which is set to 1 by default, and can be set by user to any number from 1 to 9. The lowercase letters 'm', 'x', and 'y' represent variable inputs that can be used to set the properties of the device. The lowercase letter 'd' represents variable outputs. In the commands, uppercase letters and other characters do not change.

Table 10. Interface Commands for the CS-GN300 Series

Command	Description	Response	Description
$n*<cr>	Ping unit number n.	>!n	Acknowledge ping.
$N?*<cr>	Request unit number.	>Unit Number: n	Returns unit number, default: n=1.
$n#m*<cr>	Change unit number from n to m, 1≤ m ≤ 9.	>New Unit Number: n	n=old unit number, m=new unit number, default: n=1.
$n#FW*<cr>	Save current unit number into flash memory as the new default.	>Default Unit Number set to n. Changes Written to Flash Memory.	Unit number will be changed permanently, and the current unit number will be saved into flash memory as the default unit number. PLEASE BE CAREFUL WHEN USING THIS COMMAND.
$nV*<cr>	Request firmware version.	>Firmware Version: d.d.d	Returns firmware version.
$nS*<cr>	Request serial number.	>Device n Serial Number: ddddddd	Returns 7-digit serial number.
$nST*<cr>	Request sensor type.	Sensor Type: dddd ddd	For example: *Sensor Type: GNSS 300*
$nBxxx*<cr>	Set baud rate: xxx = 2:2400, 4:4800, 9:9600, 19:19200, 38:38400, 57:57600, 115:115200, 230:230400, 460:460800, 921:921600 (bps).	>New Baud Rate: dddddd	Selected baud rate should support current data rate. Otherwise, baud rate will not be changed. The default baud rate is 115200 bps.
$nDxxx*<cr>	Set data rate: xxx= 1, 2, 5, 10, 20, 25, 40, 50, 100, 200 Hz.	>New Output Data Rate: ddd	The default data rate is 2 Hz.
$nBFW*<cr>	Save current data rate and baud rate to flash memory as the new default.	>Default Baud Rate set to dddddd. Default Data Rate set to ddd. Changes Written to Flash Memory.	Baud rate and data rate will be changed permanently and saved into the flash memory. PLEASE BE CAREFUL WHEN USING THIS COMMAND.
$nARx*<cr>	Set accelerometer measurement range: x= ±2, ±4, ±8 g.	>New Accelerometer Range: +/-d g	The new accelerometer range will be saved into flash memory (default: ±4 g).
$nAR?*<cr>	Request accelerometer measurement range.	>Accelerometer Range: +/-d g	The default range is ± 4g.
$nGRx*<cr>	Set gyroscope measurement range: x= 0: 2000, 1: 1000, 2: 500, 3: 250, 4: 125 °/s.	>New Gyroscope Range: +/-dddd deg/sec	The new gyroscope range will be saved into the flash memory (default: ±500 °/s).
$nGR?*<cr>	Request gyroscope measurement range.	>Gyroscope Range: +/-dddd deg/sec	The default range is ± 500 °/sec.

<table>
<tr><th>Command</th><th>Description</th><th>Response</th><th>Description</th></tr>
<tr><td>$nZA*<cr></td><td>Set g offset correction to 0 for X and Y axes.</td><td>>New Accelerometer Zero Offset:
X Offset: ddd,
Y Offset: ddd</td><td>Current values of X and Y accelerometers will be saved into the flash memory as the zero g offset.</td></tr>
<tr><td>$nxyz*<cr></td><td>Set output message format:
x = A: ASCII format
x = B: Binary format
y = N: Navigation data (GNSS aided navigation)
y = S: IMU Sensor data
y = V: GNSS IMU Raw data
z = S: Single message
z = C: Continuous message</td><td>>Output Message set to xxxxx | yyyyy | zzzzz

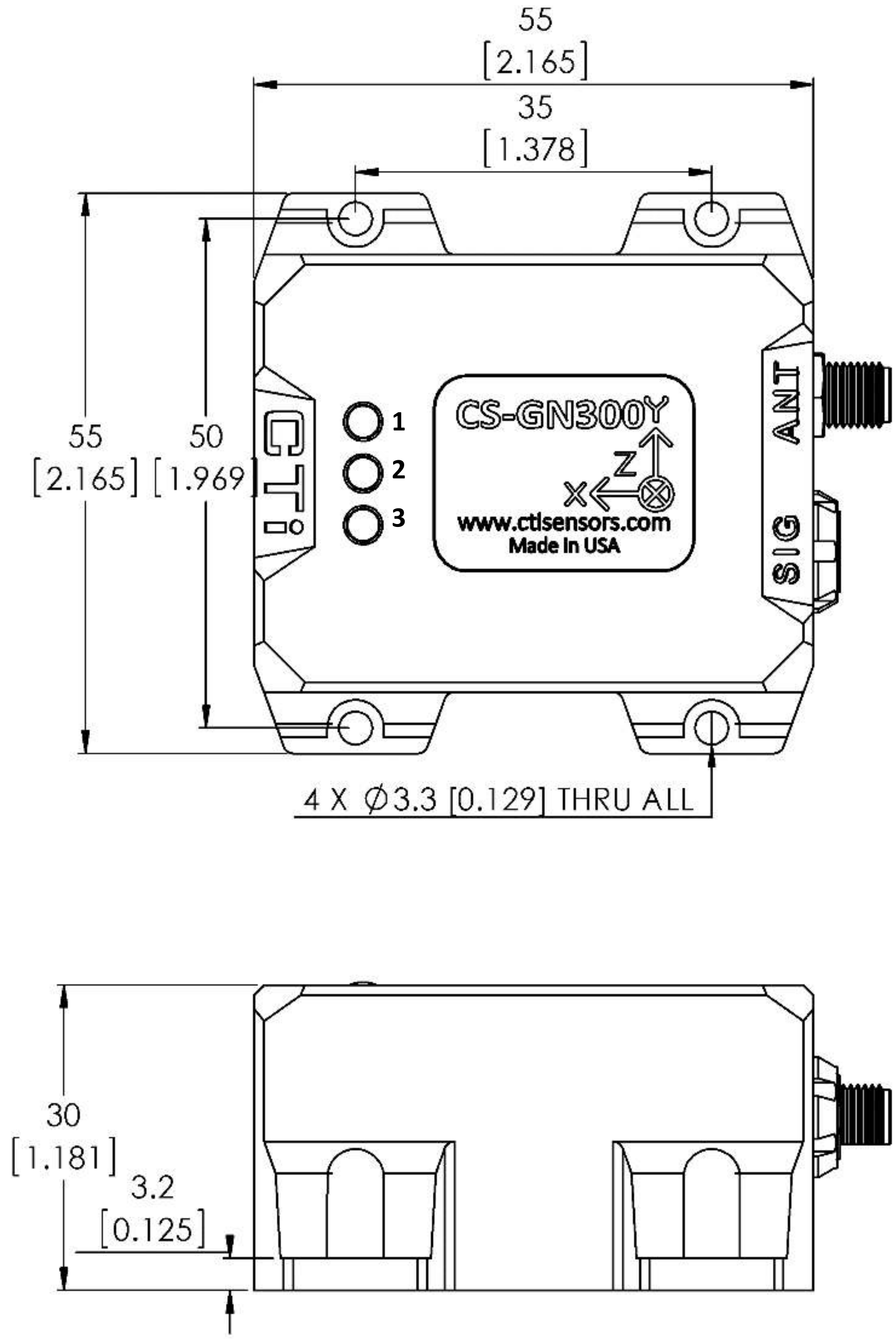
For example:
>Output Message set to BINARY FORMAT | EULER ANGLES | CONTINUOUS MESSAGE</td><td>Example for inclinometer data:
Example for inclinometer data:
$1ASC*: Continuously sends out sensor data message in ASCII
$1AEC*: Continuously sends out GNSS aided navigation message in ASCII

Default is ASCII format, GNSS aided navigation, continuous message.</td></tr>
<tr><td>$nMXX*<cr></td><td>Turns off output message.</td><td>> Output Message turned OFF</td><td>Message can be turned back on by inputting any output message format command.</td></tr>
<tr><td>$nMFW*<cr></td><td>Save current output message to flash memory as the new default.</td><td>> Current Output Message was set to Default. Changes Written to Flash Memory.</td><td>The current message status will be saved into flash memory.</td></tr>
<tr><td>$nMAVx*<cr></td><td>Toggle internal averaging:
x = 1: Averaging On
x = 0: Averaging Off</td><td>>Data output averaging filter is ON/OFF</td><td>Averaging selection will be saved into flash memory. Only sensor data has averaged data. All data in GNSS aided navigation format are real time data without averaging.</td></tr>
<tr><td>$nALPFx*<cr></td><td>Set accelerometer low pass filter bandwidth (Hz):
x = 0:1, 1:2, 2,:4, 3:8, 4:16, 5:31, 6:62, 7:125, 8:250, 9:500, 10:1000</td><td>>Accelerometer low pass filter bandwidth: dddd Hz</td><td>The default filter bandwidth is 31 Hz. New low pass filter bandwidth will be saved into flash memory.</td></tr>
<tr><td>$nALPF?*<cr></td><td>Request accelerometer low pass filter bandwidth.</td><td>>Accelerometer low pass filter bandwidth: dddd Hz</td><td>The default filter bandwidth is 31 Hz.</td></tr>
<tr><td>$nGLPFx*<cr></td><td>Set gyroscope low pass filter bandwidth (Hz):
x = 0:11, 1:21, 2:40, 3:75, 4:137, 5:255, 6:524, 7:890</td><td>>Gyroscope low pass filter bandwidth: ddd Hz</td><td>The default filter bandwidth is 40 Hz. New low pass filter bandwidth will be saved into flash memory.</td></tr>
<tr><td>$nGLPF?*<cr></td><td>Request gyroscope low pass filter setting.</td><td>>Gyroscope low pass filter bandwidth: ddd Hz</td><td>The default filter bandwidth is 40 Hz.</td></tr>
<tr><td>$nTRx*<cr></td><td>Toggles RS422 and RS485 Termination Resistor:
x = 1: Connected
x = 0: Disconnected</td><td>> RS422/RS485 Termination Resistor connected/disconnected.</td><td>By default, the Termination Resistor is connected.</td></tr>
<tr><td>$nLEG,x.xxx*<cr></td><td>Set local earth gravity.</td><td>>Local earth gravity: x.xxx</td><td>Sets the assumed local earth gravity magnitude.</td></tr>
<tr><td>$nLEG?<cr></td><td>Request local earth gravity.</td><td>>Current local earth gravity: x.xxx</td><td>Returns current assumed local earth gravity.</td></tr>
<tr><td>$nGLA,x.xxx,y.yyy,z.zzz*<cr></td><td>Set GNSS antenna lever arm.</td><td>>GNSS lever arm vector: X=x.xxx Y=y.yyy, Z=z.zzz</td><td>Sets the GNSS antenna lever arm vector. Please see section 5.2 for more details.</td></tr>
<tr><td>$nGLA?*<cr></td><td>Request GNSS antenna lever arm.</td><td>>Current GNSS lever arm vector: X=x.xxx, Y=y.yyy, Z=z.zzz</td><td>Returns the assumed GNSS antenna lever arm vector.</td></tr>
</table>

Command	Description	Response	Description
$nSIx*<cr>	Enable or disable synchronization input (SYNC-IN). x = 0: OFF , x = 1: ON	>SYNC-IN Signal Is On/OFF.	By default, the SYNC-IN is OFF. Requesting any other data messages will automatically disable SYNC-IN.
$nSOx*<cr>	Enable or disable synchronization output (SYNC-OUT). x = 0: OFF x = 1: IMU-START x = 2: IMU-READY x = 3: INS-READY	>SYNC-OUT signal is OFF/triggers on IMU-START/IMU-READY/INS-READY.	By default, the SYNC-OUT is OFF.
$nRFD*<cr>	Reset device to factory default settings.	>Reset to factory default.	Resets the selectable parameters to their default values.

7. Mechanical

7.1. Dimensional Drawing



Millimeter
[Inch]

Figure 5. Dimensions

7.2. Alignment Holes

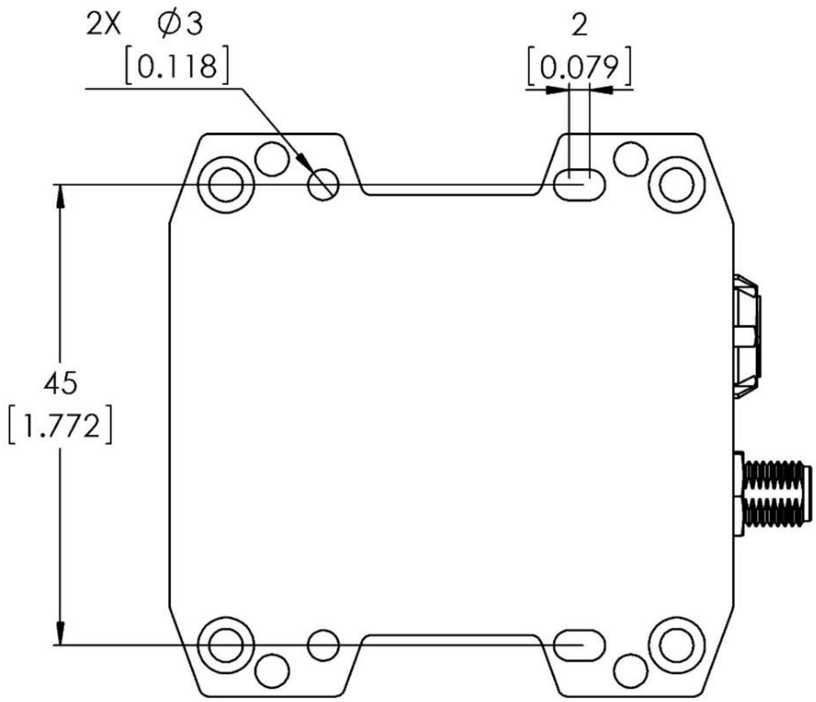


Figure 6. Alignment Holes

7.3. Sensor Mounting

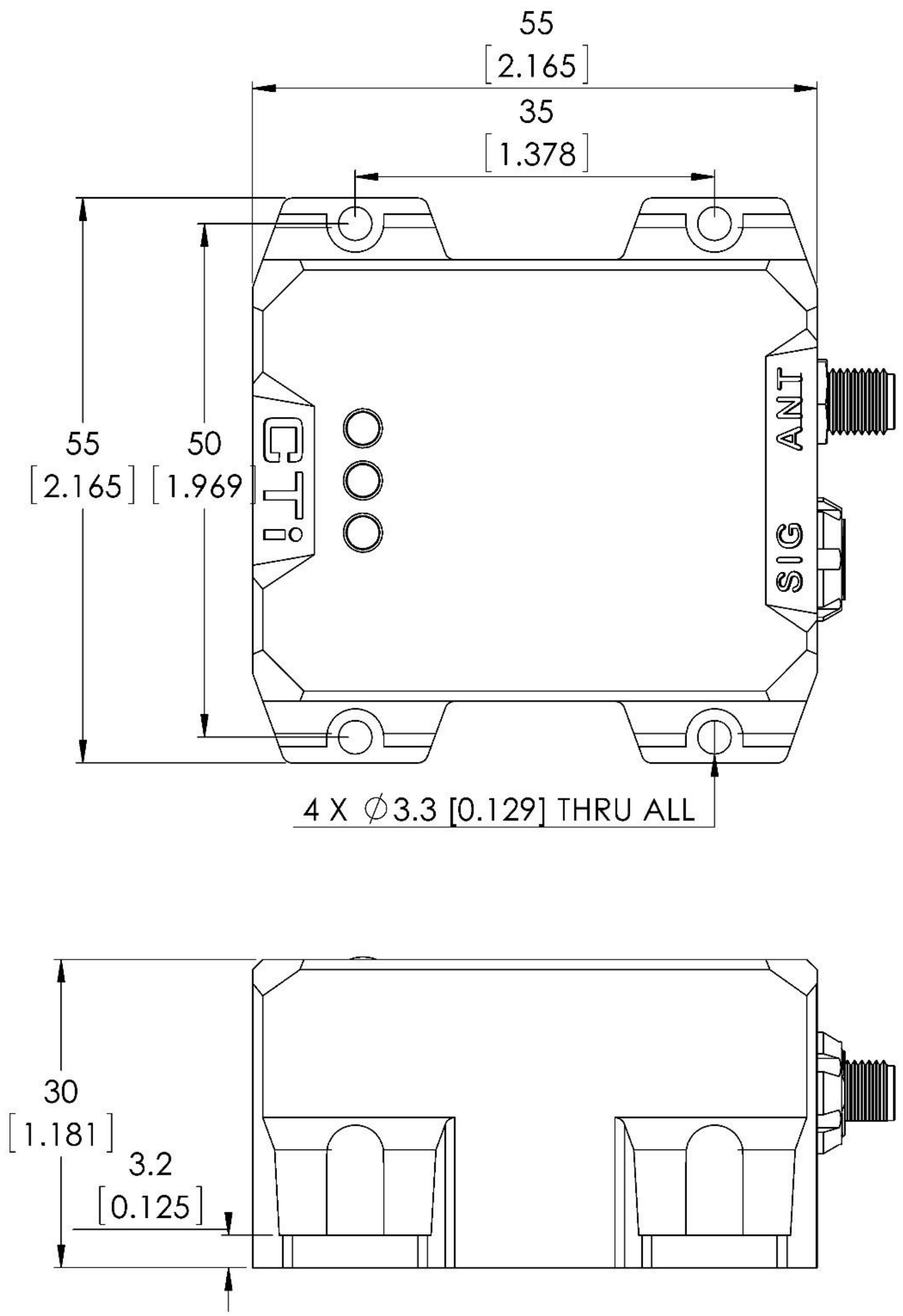


Figure 7. Sensor Mounting

- Suggested mounting screws are 4-40 socket head screw.

7.4. Connector Specifications

Table 11. Connector

Parameter	Value
Manufacture	Fischer
Part number	UR02W07 F010P BK1 E2AB
Mating part number	UP01L07 F010S BK1 Z2ZB
Number of contacts	10
Connector type	Receptacle
Polarity	Female
Sealing level	IP68
Housing material	Brass

7.5.

7.5. Cable Specifications

There are two types of cable terminations available for the CS-GN300: USB and flying wires. The former is suitable for UART/USB interface and the latter can be used for other communication interfaces.

Table 12. Cable Specifications

Parameter	Value
Number of contacts	10
Cable length	6 feet (180 cm)
Wire gage	10 x 28 AWG
Connector type	Plug
Locking system	Push-pull
Connector polarity	Male
Housing material	Brass

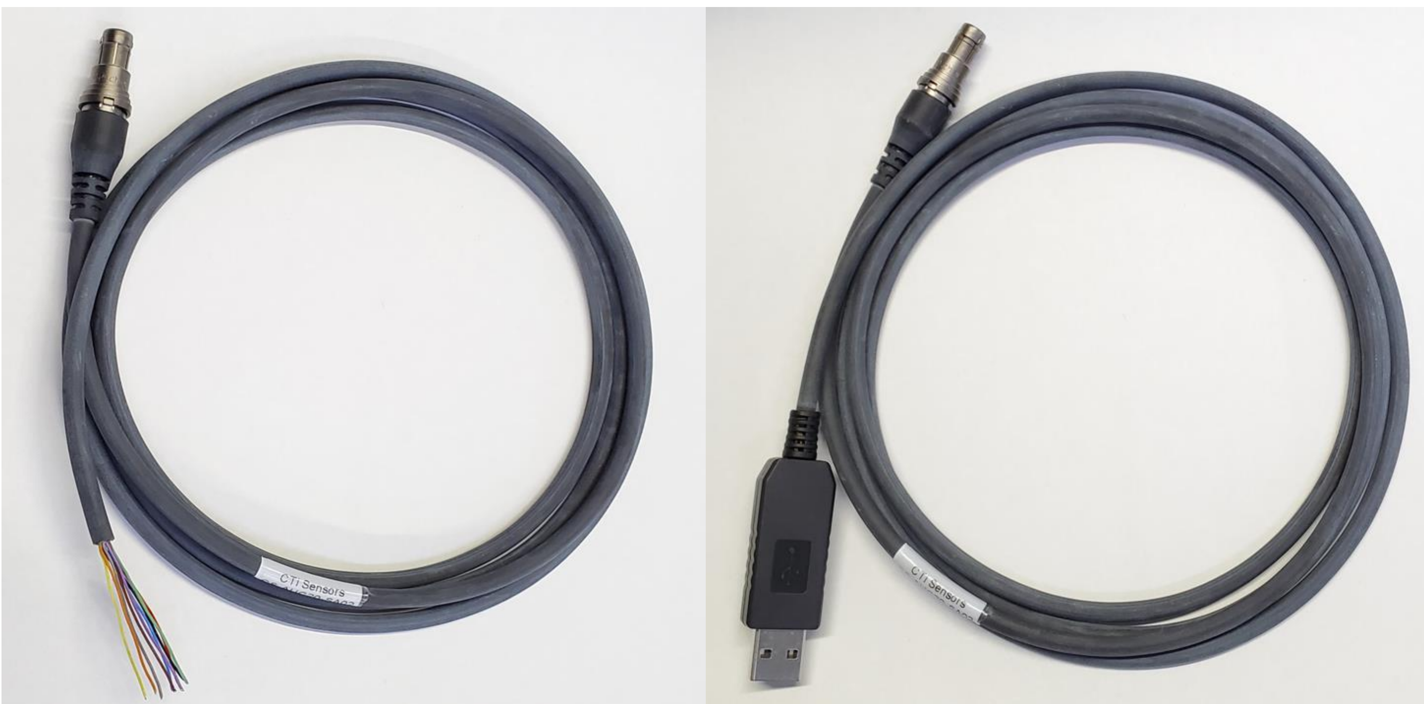


Figure 8. Two types of cable available for CS-GN300

7.6. GNSS Antenna Specifications

CS-GN300 requires an active GNSS antenna (available for order as an accessory part). The u-blox ANN-MB multi-band (L1, L2/E5b/B2I) active GNSS antenna is the default supplied by CTi sensors. It is also possible to use a different GNSS antenna.

Table 13. GNSS Antenna Specifications

Parameter	Value
Manufacture	U-blox
Part number	ANN-MB1-00
Dimension	60.0 x 82.0 x 22.5 mm
Cable	5 m RG174 standard
Connector	SMA
Impedance	50 Ω
Weight	173 g (typ. including cable)
Mounting	Magnetic base, fixed installation option
Temperature	-40°C to +85°C (-40°F to +185°F)
Protection	IP 67

Figure 9. GNSS antenna

8. Part Number

8.1. Device Part Number

CS – XX YYY – X Y – XX

Design Model (XX)

A1

Interface (Y)

3 *RS232*
4 *RS422*
8 *RS485*
U *UART/USB*

Housing Material (X)

A *Aluminum*
O *OEM (No Housing)*

Family Series	Accelerometer	Gyroscope	Magnetometer	Quaternions	Euler Angles	Velocity	Position	Temperature
CS – GN 300	*Three Axis*	*Three Axis*	*Three Axis*	✓	✓	✓	✓	✓

8.2. Accessories Part Number

- **CS-AC300-U-A1**: includes 6 feet signal cable with USB connector and one ANN-MB1-00 GNSS antenna.
- **CS-AC300-G-A1**: includes 6 feet signal cable with flying wires and one ANN-MB1-00 GNSS antenna.

9. Revision History

Table 14. Revision History

Revision Number	Revision Date	Description of Changes
1.0	Mar. 2022	Draft version
1.1	Feb. 2023	Initial release

10. Warranty Information

CTi SENSOR, INC. "CTi" warrants its products against defects in material and workmanship for a period of 18 months from the date of the shipment to the customer provided the products have been stored, handled, installed and used under proper conditions. CTi's liability under this limited warranty shall extend only to repair or replace the defective product, at CTi's option. This warranty does not cover misuse or careless handling and it is void if the product has been altered or repaired by personnel not authorized by CTi. CTi disclaims all liability for any affirmation, promise, or consequential damages caused by the product. No warranties, expressed or implied, are created with respect to CTi's products except those expressly contained herein. The customer acknowledges the disclaimers and limitation contained herein, and relies on no other warranties or affirmations.

For more information, please refer to the following link:

www.CTiSensors.com/warranty

All contents of this document are subject to change without notice.
This product was designed and manufactured in the U.S.A.

CTi SENSOR, INC.
30301 Emerald Valley Parkway, Unit B
Solon, OH 44139
Phone: (440) 264-2370
Email: sales@CTiSensors.com