



CS-100 Series VRU, IMU and AHRS



CTi SENSOR, INC.

Document Revision 1.2

CTi Sensors Technical Document

This is CS-100 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	4
1.1. FEATURES	4
1.2. APPLICATIONS	4
2. SPECIFICATIONS	5
2.1. ANGLES (VRU AND AHRS ONLY)	5
2.2. ACCELEROMETER	5
2.3. GYROSCOPE	5
2.4. MAGNETOMETER (IMU AND AHRS ONLY)	6
2.5. SYSTEM	6
2.6. MECHANICAL	6
3. TERMINAL ASSIGNMENT	7
4. CTI SENSORS CONNECT SOFTWARE	8
5. SERIAL INTERFACE AND DATA FORMAT	9
5.1. ASCII MESSAGE FORMAT	9
5.2. BINARY MESSAGE FORMAT	11
5.3. MODBUS RTU FORMAT	13
6. 8-BIT CHECKSUM	14
7. CONFIGURATION COMMANDS	15
8. DIMENSIONAL DRAWING	17
9. PART NUMBER	18
10. WIRED CABLES	19
11. REVISION HISTORY	20
12. WARRANTY INFORMATION	20

1. Introduction

A **Vertical Reference Unit (VRU)** is an inertial measurement device designed to measure tilt angles (roll and pitch) using a combination of accelerometers and gyroscopes. An onboard fusion algorithm is implemented to take the raw sensor data and convert it into angles. These devices are fairly simple to work with but also very reliable and an excellent choice in applications which only need to monitor one or two angles.

An **Inertial Measurement Unit (IMU)** is a 9 Degree-of-Freedom (DOF) inertial measurement device. The 3-axis accelerometer can be used to sense gravitational acceleration as well as motion. Rotation-rate can be sensed by the 3-axis gyroscope and the magnetometer provides 3-axis measurements of the magnetic field. This calibrated MEMS sensing package provides a wealth of dynamic measurements to support a wide range of applications.

An **Attitude Heading and Reference System (AHRS)** is a 9 DOF system measuring 3-dimensional orientation (roll, pitch, and yaw angles). The combination of accelerometers, gyroscope, and magnetometers are used internally by CTi Sensors fusion algorithm to achieve this. This sensor has applications in attitude control and robotics as well as in vehicle localization and as part of a telemetry module. The advanced user has access to the underlying 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer measurements. The robust housing, interface options, and simple ASCII communication protocol means setup is minimal, allowing the user to begin testing quicker.

The **CS-100 Series** uses the latest high performance, high resolution miniature MEMS sensor technology.

1.1. Features

- 6 DOF sensor data: 3-axis accelerometer and 3-axis gyroscopic turn rate
- Additional 3 DOF on IMU and AHRS: 3-axis magnetic field intensity
- Output data rate up to 200 Hz
- Selectable accelerometer and gyroscope range
- Simple ASCII interface language
- IP 67 compliant connector, cable, and housing
- Robust aluminum housing
- Low power consumption: 250 mW (50 mA @ 5 V)

1.2. Applications

- Inertial navigation and GPS compensation
- Avionics and unmanned vehicles
- Motion and dynamics measurements
- Vehicle control: marine, robotics, automotive
- Agricultural and industrial vehicle tilt monitoring
- Virtual/augmented reality

2. Specifications

2.1. Angles (VRU and AHRS only)

Table 1. Angles (VRU and AHRS only)

Parameter	Value
Range	Pitch: $\pm 90^\circ$, roll: $\pm 180^\circ$ Heading (AHRS only): $\pm 180^\circ$
Static Accuracy (RMS)	Pitch and roll: 0.2° Heading (AHRS only) ¹ : 1.5°
Dynamic Accuracy (RMS)	Pitch and roll: 1.0° Heading (AHRS only): 1.5°
Angular resolution	Pitch and roll: $< 0.025^\circ$ Heading (AHRS only): $< 0.1^\circ$
Zero offset error (pitch and roll)	$\pm 0.1^\circ$ (@ 20°C)
Offset change versus temperature (pitch and roll)	$\pm 0.01^\circ/\text{C}$ (typical)

2.2. Accelerometer

Table 2. Accelerometer

Parameter	Value
Range	$\pm 2\text{ g}/\pm 4\text{ g}/\pm 8\text{ g}$ selectable
Zero offset error	$< 1.0\text{ mg}$ (@ 20°C)
In-run bias stability	$< 10\text{ }\mu\text{g}$
Velocity random walk	X & Y: $0.075\text{ m/sec}/\sqrt{\text{hr}}$, Z: $0.095\text{ m/sec}/\sqrt{\text{hr}}$
Nonlinearity	$\pm 0.5\%$ FS
Bias change versus temperature	$\pm 0.2\text{ mg}/^\circ\text{C}$ (typical)
Noise density	$125\text{ }\mu\text{g}/\sqrt{\text{Hz}}$ (@ 200 Hz)

2.3. Gyroscope

Table 3. Gyroscope

Parameter	Value
Range	$\pm 250/500/1000/2000^\circ/\text{s}$ selectable
In-run bias stability	$< 20^\circ/\text{hr}$
Angle random walk	$1.4^\circ/\sqrt{\text{hr}}$
Initial bias error	$< 0.2^\circ/\text{s}$ (@ $\pm 500^\circ/\text{s}$ range)
Bias change versus temperature	X & Y ² : $\pm 0.02^\circ/\text{s}/^\circ\text{C}$ Z: $\pm 0.01^\circ/\text{s}/^\circ\text{C}$
Noise density	$0.025\text{ dps}/\sqrt{\text{Hz}}$ (@ 64 Hz)
Nonlinearity	$< 0.5\%$ FS

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

² AHRS and VRU have the in-run bias compensation for gyro axes X and Y

2.4. Magnetometer (IMU and AHRS only)

Table 4. Magnetometer (IMU and AHRS only)

Parameter	Value
Range	$\pm 800 \mu\text{T}$
Nonlinearity	$\pm 0.1 \% \text{ FS}$
Noise density	$0.06 \mu\text{T}/\sqrt{\text{Hz}}$ (@ 100 Hz)
Bandwidth	100 Hz

2.5. System

Table 5. System

Parameter	Value
Power source	4.1-38 VDC
Power consumption	250 mW (50 mA @ 5 V)
Data format	ASCII and binary
Output data rate	1 Hz to 200 Hz selectable
GUI software	CTi Sensors Connect®
Serial interface options	RS232, RS422, RS485, UART/ USB, Modbus RTU, Wireless, RS485 with multi-drop networking
LED indicators	Green: CPU heartbeat, Flashing at 1 Hz Red: Data transmission rate, Flashing at current data rate
Temperature sensor resolution	1°C

2.6. Mechanical

Table 6. Mechanical

Parameter	Value
Protection	IP 67 (housing, connector and cable)
Dimension	1.65" x 2.15" x 1.00"
Material (cable is optional as a third-party product)	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 ¹	Cable gland connector M8, 6-contact (female)

¹ Cable is a third-party product with temperature tolerance from -40°C to +105°C (-40°F to +221°F).

2.7. Terminal Assignment

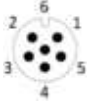
Table 7. Terminal Assignment

Connector	Wireless	RS232/UART/USB ¹	RS422	RS485	Wire Color
Pin 1	+Vin	+Vin	+Vin	+Vin	Brown
Pin 2	GND	GND	GND	GND	White
Pin 3	–	TX	TX+	D+	Blue
Pin 4	–	–	TX-	D-	Black
Pin 5	–	RX	RX+	D+	Gray
Pin 6	–	–	RX-	D-	Pink



Device:
M 8 – 6-contact (female)

Cable:
M 8 – 6-pin (male)



¹ USB uses UART interface and a UART to USB cable.

3. CTi Sensors Connect Software

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



4. Serial Interface and Data Format

The CS-100 series can transmit data in either ASCII or binary format. ASCII is designed to be very easy to read and parse, while binary is designed to maximize the efficiency in transmitting the message. For this reason, binary messages may generally be run at higher data rates than ASCII messages at the same baud rate. CS-100 series devices also have a RTU Modbus mode which can be enabled using ASCII or binary prompts. The Modbus mode is described in section 5.3. This sensor has three available ASCII messages. Two output angular data, one in Euler angles and one in quaternions. The remaining message outputs sensor data. It has two binary messages. These binary messages output both angular and sensor data together, one formats the angular data as Euler angles, the other as quaternions. All commands and responses to commands are transmitted in ASCII, regardless of the current message format. The sensor will default to its angular ASCII message.

Binary data is in IEEE 754 format, with the exception of magnetometer and temperature data which is transmitted as a two-byte integer multiplied by a factor of 16.

4.1. ASCII Message Format

The **CS-VR100** uses the following ASCII format, based on the widely used NMEA 0183 protocol, for data output:

- Angular message (default): `$CSPRA, $\alpha_X$ , $\alpha_Y$ ,T*CC<CR><LF>`
- Sensor data message (optional): `$CSAGD,AX,AY,AZ,GX,GY,GZ,T*CC<CR><LF>`

The **CS-IM100** uses the following ASCII format, based on the widely used NMEA 0183 protocol, for data output:

- Sensor data message (default): `$CSIMU,AX,AY,AZ,GX,GY,GZ,MX,MY,MZ,T*CC<CR><LF>`

The **CS-AH100** uses the following ASCII format, based on the widely used NMEA 0183 protocol, for data output:

- Angular Euler message (default): `$CSPRY, $\alpha_X$ , $\alpha_Y$ , $\alpha_Z$ ,T*CC<CR><LF>`
- Angular Quaternion message (optional): `$CSQTN,Q0,Q1,Q2,Q3,T*CC<CR><LF>`
- Sensor data message (optional): `$CSIMU,AX,AY,AZ,GX,GY,GZ,MX,MY,MZ,T*CC<CR><LF>`

Where:

- α_X , α_Y , α_Z : Roll, Pitch and Heading angles in degrees
- Q_0 , Q_1 , Q_2 , Q_3 : Quaternions
- 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 microTesla (three-axis magnetometer data)
- T: Internal temperature in degrees Celsius
- CC: Checksum (Two ASCII characters)
- <CR> <LF>: Carriage return, and line feed characters

Examples:**VRU message:**

- \$CSRPA,+000.19,-000.98,+029*60
- \$CSVRU,-0017.4,-0003.6,-0992.9,+0000.37,+0001.19,-0000.74,+030*47

IMU message:

- \$CSIMU,-0027.0,+0508.2,-0845.2,-0018.11,+0000.18,-0001.58,+012.0,-010.9,-041.3,+027*6A

AHRS message:

- \$CSRPY,+000.17,+000.03,+081.08,+029*6A
- \$CSQTN,+00.866,+00.005,-00.001,-00.499,+029*6F
- \$CSIMU,+0001.1,-0002.9,-0993.0,+0000.66,+0001.07,-0000.72,+001.9,-013.0,-030.9,+029*62

4.2. Binary Message Format

Binary data is in IEEE 754 format, with the exception of magnetometer and temperature data which is transmitted as a two-byte integer multiplied by a factor of 16.

The **CS-VR100** only has one binary message, which outputs all available data from the sensor in the format:

- 1 byte start character: 0xAA
- 1 byte message type: 0xEA for the CS-VR100
- 1 byte data length N: N is 0x20 (34 in base 10) for the CS-VR100
- N byte data (α_x , α_y , A_x , A_y , A_z , G_x , G_y , G_z , and T in that order. Angular, accelerometer and gyroscope data are 4 bytes, magnetometer and temperature are 2 bytes.)
- 1 byte checksum
- 1 byte stop character: 0x55

The **CS-IM100** only has one binary message, which outputs all available data from the sensor in the format:

- 1 byte start character: 0xAA
- 1 byte message type: 0xDA for the CS-IM100
- 1 byte data length N: N is 0x1E (32 in base 10) for the CS-IM100
- N byte data (A_x , A_y , A_z , G_x , G_y , G_z , M_x , M_y , M_z , and T in that order. Angular, accelerometer and gyroscope data are 4 bytes, magnetometer and temperature are 2 bytes.)
- 1 byte checksum
- 1 byte stop character: 0x55

The **CS-AH100** has two binary messages, which output data from the sensor in the format:

Euler Message:

- 1 byte start character: 0xAA
- 1 byte message type: 0xCE for the CS-AH100 Euler Message
- 1 byte data length N: N is 0x2C (44 in base 10) for the CS-AH100 Euler Message
- N byte data (α_x , α_y , α_z , A_x , A_y , A_z , G_x , G_y , G_z , M_x , M_y , M_z , and T in that order. Angular, accelerometer and gyroscope data are 4 bytes, magnetometer and temperature are 2 bytes.)
- 1 byte checksum
- 1 byte stop character 0x55

Quaternion Message:

- 1 byte start character: 0xAA
- 1 byte message type: 0xBC for the CS-AH100 Quaternion Message
- 1 byte data length N: N is 0x30 (48 in base 10) for the CS-AH100 Quaternion Message
- N byte data (Q_0 , Q_1 , Q_2 , Q_3 , A_x , A_y , A_z , G_x , G_y , G_z , M_x , M_y , M_z , and T in that order. Angular, accelerometer and gyroscope data are 4 bytes, magnetometer and temperature are 2 bytes.)
- 1 byte checksum
- 1 byte stop character 0x55

Where:

α_x , α_y , α_z : Roll, Pitch and Heading angles in degrees

Q_0 , Q_1 , Q_2 , Q_3 : Quaternion representation of angles

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 microTesla (three-axis magnetometer data)

T: Internal temperature in degrees Celsius

Examples:**VRU message:**

– AA EA 20 BF AE 62 12 3F A1 99 BA 41 AF 97 1B 41 BD E6 FD C4 79 26 33 3D C1 A5 30 BE 02
6B 44 BF 8D 79 02 80 00 08 55

IMU message:

– AA DA 1E 41 AA 88 44 41 C4 13 01 C4 79 3D 2E BF 84 B8 52 3E 01 99 9A BF 81 0A 3D FF 5C
00 1C 01 DC 80 00 DC 55

AHRS message:

– AA CE 2C BF A8 5E 0C 3F B3 C8 6A C2 F1 0D 3E 41 C3 F3 E8 41 B7 06 61 C4 79 49 E3 BD D4
FF D8 3E 23 4A BA BF 91 2D 1E FF 9D 00 6F 03 5B 80 00 FD 55

– AA BC 30 3E 89 AF AB 3C 0F F3 1F 3C 6C F4 B6 BF 76 88 1B 41 C4 67 EA 41 B4 B6 3C C4 79
3C A9 3E D2 C0 0A 3E E1 A5 5D BF 52 5A 3B FF A3 00 7A 03 5A 80 00 85 55

4.3. Modbus RTU Format

All three CS-100 devices have an RTU Modbus mode which can be enabled using ASCII or binary prompts. While in Modbus mode, the device will not normally respond to ASCII commands. Settings such as baud rate should be selected before entering Modbus mode.

Additionally, while in Modbus mode, the device will perform a CRC check on incoming data. The device uses CRC-16 Modbus, output in two bytes. If the input CRC is not correct, or if the message is not recognized, then the device will not respond.

- **\$1MM1* (0x24 0x31 0x4d 0x4d 0x31 0x2a 0x0d)**

This command will switch the sensor from ASCII mode to Modbus mode. It can be input as either ASCII or hexadecimal.

- **\$1MM0* (0x24 0x31 0x4d 0x4d 0x30 0x2a 0x0d)**

This command, which can be input as either ASCII or hexadecimal, will cause the sensor to leave Modbus mode. No other ASCII commands will be processed or acknowledged while the sensor is in Modbus mode.

- **0x03: Read Holding Registers**

Command 0x03 is the command to read sensor data in Modbus Mode. The sensor is expecting an 8-byte command where:

Byte 1 is the unit number, 0x01 by default

Byte 2 is the command number, 0x03 in this case

Bytes 3-4 are the data addresses

Bytes 5-6 are the number of response bytes expected

Bytes 7-8 are the CRC

The data addresses are as follows. The data format is the same as the binary message, where most data is transferred in IEEE 754, with temperature and magnetometer data transferred as a 2-byte integer multiplied by a factor of 16.

Table 8. Modbus data Addresses

Address	Parameter	Address	Parameter	Address	Parameter
0x01	Temperature	0x0b	Ax MSB	0x15	Psi MSB
0x02	Mx	0x0c	Ax LSB	0x16	Psi LSB
0x03	My	0x0d	Ay MSB	0x17	Q0 MSB
0x04	Mz	0x0e	Ay LSB	0x18	Q0 LSB
0x05	Gx MSB	0x0f	Az MSB	0x19	Q1 MSB
0x06	Gx LSB	0x10	Az LSB	0x1a	Q1 LSB
0x07	Gy MSB	0x11	Phi MSB	0x1b	Q2 MSB
0x08	Gy LSB	0x12	Phi LSB	0x1c	Q2 LSB
0x09	Gz MSB	0x13	Theta MSB	0x1d	Q3 MSB
0x0a	Gz LSB	0x14	Theta LSB	0x1e	Q3 LSB

- **0x10: Preset Registers**

This command will toggle the sensor between defaulting to Modbus mode on startup or not. The choice is automatically saved to the flash.

For other settings, such as baud rate, set and save those before switching into Modbus Mode.

5. 8-Bit Checksum

The checksum is calculated in ASCII by XORing bitwise all bytes (each character is represented by 1 byte) between \$ 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 code for calculating and checking the checksum is as follows:

```
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. Configuration Commands

The CS-100 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 '*' character followed by a carriage return. All responses end with a carriage return and newline character. Table I shows the list of the interface commands for the CS-100 Series.

In the table below, lowercase '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.

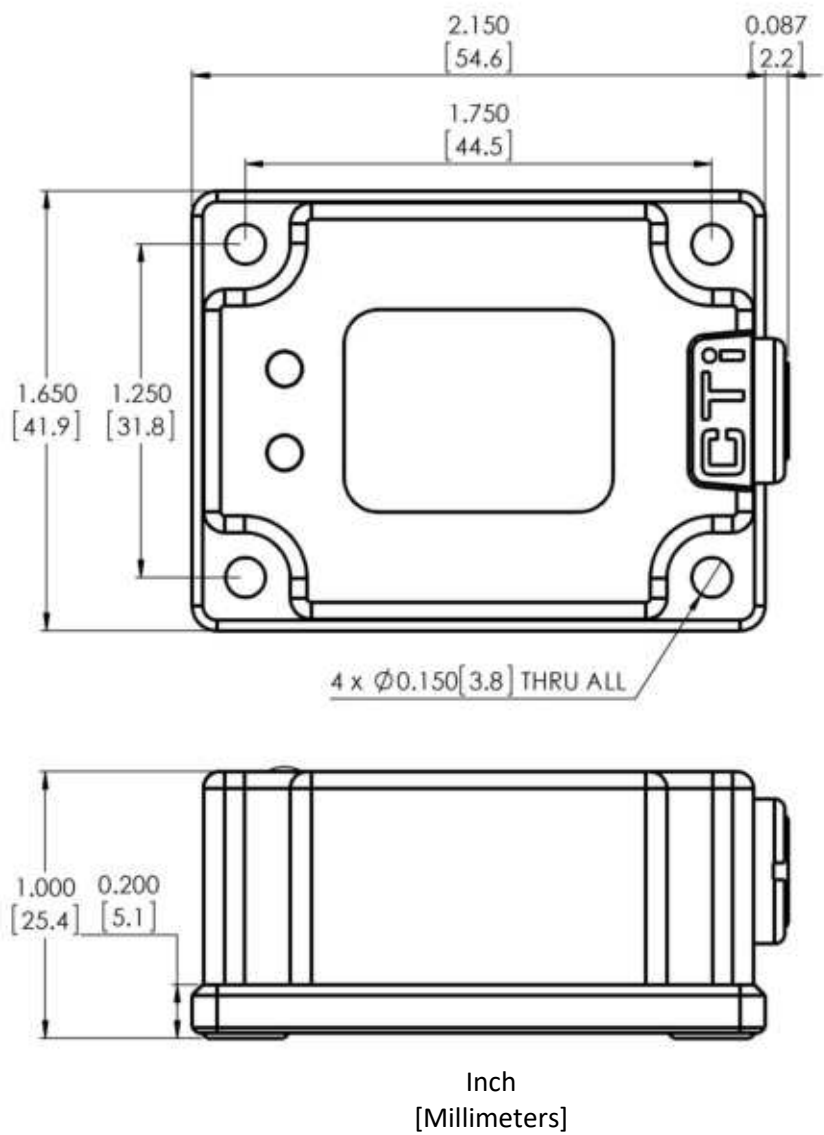
Table 9. Interface Commands for the CS-100 Series

Command	Description	Response	Description
\$ <u>n</u> *<cr>	Ping unit number n	>! <u>n</u>	Acknowledge ping
\$N?*<cr>	Request unit number	>Unit Number: <u>n</u>	Returns unit number, default: n=1
\$ <u>n</u> # <u>m</u> *<cr>	Change unit number from n to m, 1 ≤ m ≤ 9	>New Unit Number: <u>m</u>	n=old unit number, m=new unit number, default: n=1
\$ <u>n</u> #FW*<cr>	Save current unit number into flash memory as the new default.	>Default Unit Number set to <u>n</u> . Changes Written to Flash Memory.	Unit number will be changed permanently, and current unit number will be saved into the flash memory as the default unit number. PLEASE BE CAREFUL WHEN USING THIS COMMAND.
\$ <u>n</u> V*<cr>	Request firmware version	>Firmware Version: <u>d.dd</u>	Returns firmware version
\$ <u>n</u> S*<cr>	Request serial number	>Device <u>n</u> Serial Number: <u>ddddddd</u>	Returns 7-digit serial number
\$ <u>n</u> ST*<cr>	Request sensor type	Sensor Type: <u>ddd ddd</u>	For example: <i>Sensor Type: AHRS 100</i>
\$ <u>n</u> Bxxx*<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: <u>dddddd</u>	Selected baud rate should support current data rate. Otherwise, baud rate will not be changed. Default baud rate is 115200 bps.
\$ <u>n</u> Dxxx*<cr>	Set data rate: xxx= 1, 2, 5, 10, 20, 25, 40, 50, 100, and 200 Hz	>New Output Data Rate: <u>ddd</u>	Default data rate is 2 Hz.
\$ <u>n</u> BFW*<cr>	Save current data rate and baud rate to flash memory as the new default.	>Default Baud Rate set to <u>dddddd</u> . Default Data Rate set to <u>dd</u> . 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.
\$ <u>n</u> ARx*<cr>	Set accelerometer measurement range: x= ±2, ±4, ±8 g	>New Accelerometer Range: +/- <u>d</u> g	New accelerometer range will be saved into the flash memory (Default: ±4 g).
\$ <u>n</u> AR?<cr>	Request accelerometer measurement range.	>Accelerometer Range: +/- <u>d</u> g	Default range is ±4g.
\$ <u>n</u> GRx*<cr>	Set gyroscope measurement range: x= 0:2000, 1:1000, 2:500, 3:250 °/s	>New Gyroscope Range: +/- <u>ddd</u> deg/sec	New gyroscope range will be saved into the flash memory (Default: ±500 °/s).
\$ <u>n</u> GR?<cr>	Request gyroscope measurement range.	>Gyroscope Range: +/- <u>ddd</u> deg/sec	Default range is ± 500 °/sec.

Continued...

Command	Description	Response	Description
\$ <u>n</u> ZA*<cr>	Set g offset correction to 0 for X and Y axes.	>New Accelerometer Zero Offset: X Offset: ddd, Y Offset: ddd	Current values of Ax and Ay will be saved into the flash memory as the zero g offset.
\$ <u>n</u> xyz*<cr>	Set output message format: x = A: ASCII format x = B: Binary format y = E: Euler angles (AHRS and VRU only) y = Q: Quaternions (AHRS only) y = S: Sensor data (ASCII format only) z = S: Single message z = C: Continuous message	>Output Message set to <u>xxxxx</u> <u>yyyyy</u> <u>zzzzz</u> For example: >Output Message set to BINARY FORMAT EULER ANGLES CONTINUOUS MESSAGE	Example for inclinometer data: \$1ASC*: Continuously sends out sensor data message in ASCII \$1AEC*: Continuously sends out Euler angle message in ASCII \$1BQS*: Sends out one quaternion message in binary Default is ASCII format, Euler angles, continuous message.
\$ <u>n</u> MXX*<cr>	Turns off output message.	> Output Message turned OFF	Message can be turned back on by inputting any output message format command.
\$ <u>n</u> MFW*<cr>	Save current output message to flash memory as the new default.	> Current Output Message was set to Default. Changes Written to Flash Memory.	Current message status will be saved into flash memory.
\$ <u>n</u> MAVx*<cr>	Toggle internal averaging: x = 1: Averaging On x = 0: Averaging Off	>Data output averaging filter is ON/OFF	Averaging selection will be saved into flash memory.
\$ <u>n</u> RFD*<cr>	Reset device to factory default settings.	>Reset to factory default.	Resets the selectable parameters to their default values.

7. Dimensional Drawing



8. Part Number

CS – XX YYY

–

XY

–

XX

Design Model

A1

Interface

3 RS2324 RS4228 RS485U UART/USBW Wireless¹

Housing Material

A AluminumO OEM (No Housing)

Family Series

CS – VR 100CS – VR 200CS – IM 100CS – IM 200CS – AH 100CS – AH 200

Accelerometer

Three Axis

Gyroscope

Three Axis

Magnetometer

✗

Quaternions

✗

Pitch/Roll

✓

Heading

✗

Temperature

✓

Three Axis

Three Axis

Three Axis

✗

✗

✗

✓

Three Axis

Three Axis

Three Axis

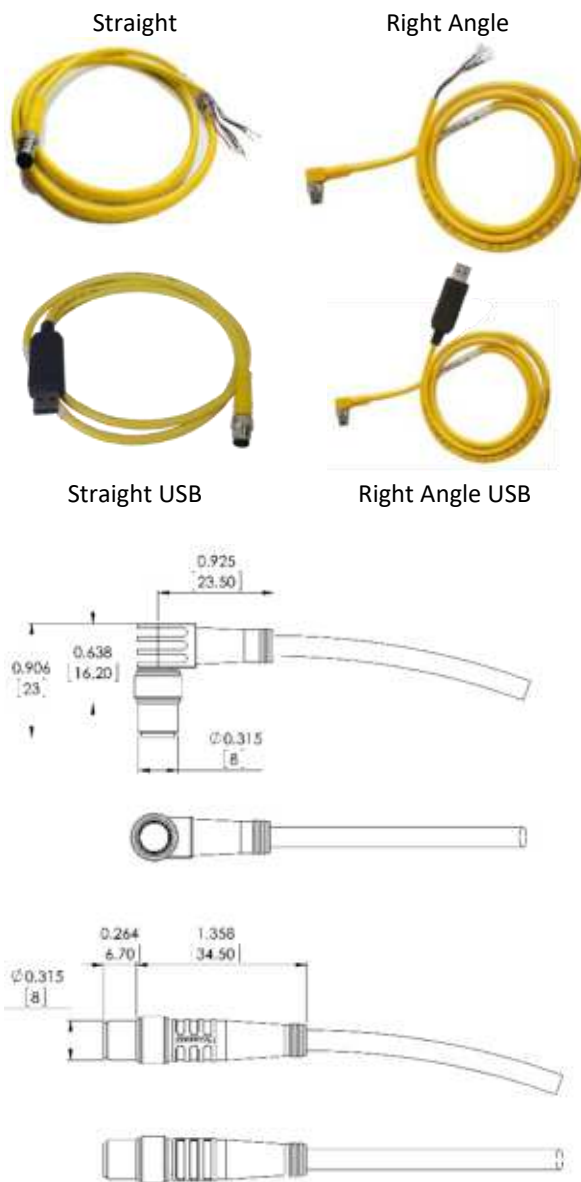
✓

✓

✓

✓

¹ Wireless communication requires both wireless cable and CTi USB dongle. Neither wireless cable nor CTi USB dongle can connect to another device. To work properly, the Wireless Module on the device side must be powered.

9. Wired Cables¹

Inch

[millimeter]

Communication Cable Part Number ²

XXX	XX	-	X	-	X	-	CS
							Interface
							<u>G</u> <u>UART / RS232 / RS422 / RS485</u>
							<u>U</u> <u>USB</u>
							<u>W</u> <u>Wireless</u> ³
							Length
							<u>1</u> <u>meter</u>
							<u>2</u> <u>meter</u>
							<u>3</u> <u>meter</u>
							<u>4</u> <u>meter (non-stock)</u>
							<u>5</u> <u>meter (non-stock)</u>
							<u>6</u> <u>meter (non-stock)</u>
							<u>10</u> <u>meter (Only PSG)</u>
							<u>15</u> <u>meter (non-stock)</u>
							<u>30</u> <u>meter (non-stock)</u>
							Type
							<u>PSG 6M</u> <u>Straight</u>
							<u>PSW 6M</u> <u>Right Angle</u>

Specifications

Protection	IP 67
	Connector: brass / nickel
	Cable molded head: TPU
	Cable carrier: TPU or nylon
	Conductor insulation: PVC
Operational Temperature range	-40°C to +85°C (-40°F to +185°F)
Connection	Cable gland Connector M8, 6-contact (male)

Wireless Cables:

<https://ctisensors.com/Documents/Wireless-Datasheet.pdf>
¹ Cable is a third-party product.² Available options for this model are underlined.³ Wireless module on device side has to be powered.

10.Revision History

Table 10. Revision History

Revision Number	Revision Date	Description of Changes
1.1	Jan. 2020	Initial release
1.2	Oct. 2022	Updated information and formatting

11.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

WARRANTY: This product has 18 months limited warranty. For more information, please visit:
www.CTiSensors.com/warranty

This product is 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

All contents of this document are subject to change without notice.