CNS16460

High Performance Inertial Measurement Unit

Main Features

- High Precision 6 DoF Mini IMU with Full Calibration
- Fully Compatible with ADIS16460, Better Performance
- 3 Angular Rates and 3 Linear Acceleration Outputs
- Data Output Rate: 200Hz
- Range: Gyro ±400° /s, Accelerometer ±6g
- Bias Instability: Gyro 4° /h, Accelerometer ±10μg
- Compact and Light weight 23.4 x 22.4 x 7.4 (mm), <15g
- Wide Working Temperature: -40° C～+85° C

CNS16460 High Performance Inertial Measurement Unit is a rugged industrial inertial navigation system providing precise yaw, pitch, roll as well as 6-axis outputs of angular rate and acceleration at 200Hz. CNS16460 adopts MEMS-based inertial sensors and extended Kalman filter algorithm, which provides super cost-effective solution for customers

CNS16460 Inertial Measurement Unit enjoys small size and light weight, and it is housed in an ultra-durable and compact aluminum housing, it enjoys high reliability in harsh environment, it has been widely applied in dynamic control systems and attitude systems including Unmanned Aircraft and ROVs, drones,Smart Agriculture, Machine Control, Platform Stabilization, AGV, robotics, etc.

- ✓ **12-Step Quality Control, Super Reliability, More Functions**
- ✓ **Adopting Original Big Brand Component, High-class Material, Competitive Price**
- ✓ **Real Actual Precise after Calibration, Perfect Performance**
- ✓ **Successful Applications in Tens of Fields, More than 1000 Customers are Using**

Typical Applications

Unmanned Aircraft

Smart Agriculture

Robotics

Unmanned Vehicle

Add:Room B-1120 No.15-1 Jian Guo Road, Chao Yang District, Beijing, China.100024
Tel: 4006-910-991 Mob: 13911707235 E-mail:bjsensors@gmail.com www.cnsens.com

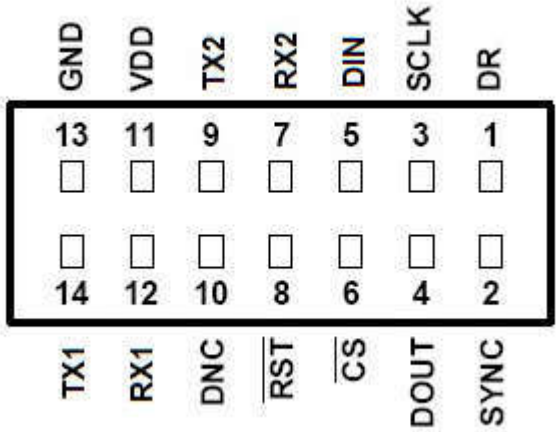
CNS16460

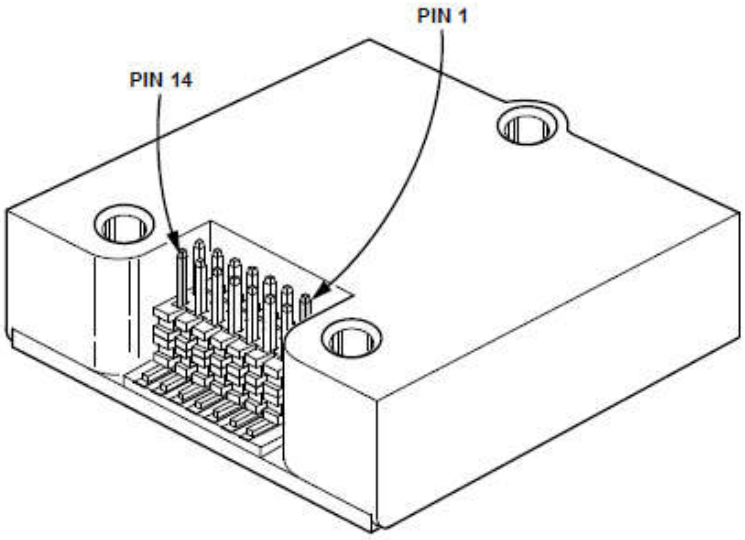
High Performance Inertial Measurement Unit

Technical Specifications

Technical Specs				
Parameter	**Value**			**Comments**
	Min	**Typical**	**Max**	
Gyroscopes				
Range:	-400°/s		+400°/s	
Bias Instability	3°/h	4°/h	5°/h	
Initial Bias Error	0.08°/s	0.1°/s	0.2°/s	
Resolution		0.01°/s		
Scale Factor Accuracy		0.1%		
Non-linearity		0.5%FS		
Random Walk	0.2°/√hr	0.3°/√hr	0.5°/√hr	
Bandwidth		50Hz		
Accelerometers				
Range: X, Y, Z	-6g		+6g	g=9.80665 m/s^2
Bias Instability	5μg	10μg	15μg	
Initial Bias Error	3mg	4mg	5mg	
Resolution	-	0.5mg	-	
Scale Factor Accuracy	-	0.1%	-	
Non-linearity	-	0.1%FS	-	
Random Walk	0.03m/s/√hr	0.05m/s/√hr	0.07m/s/√hr	
Bandwidth	-	50Hz	-	
Environment Conditions				
Working Temperature	-40~+85°C			
Storage Temperature	-55~+100°C			
Housing Material	aluminum			
Electrical Features				
Input Power Supply	3.3VDC			
Power Consumption	<0.15W			
Interface	SPI / UART			Optional
Output Frequency	200Hz			
Physical Parameter				
Size	23.4 x 22.4 x 7.4			
Weight	<15grams			
Connector	7*2 pins			

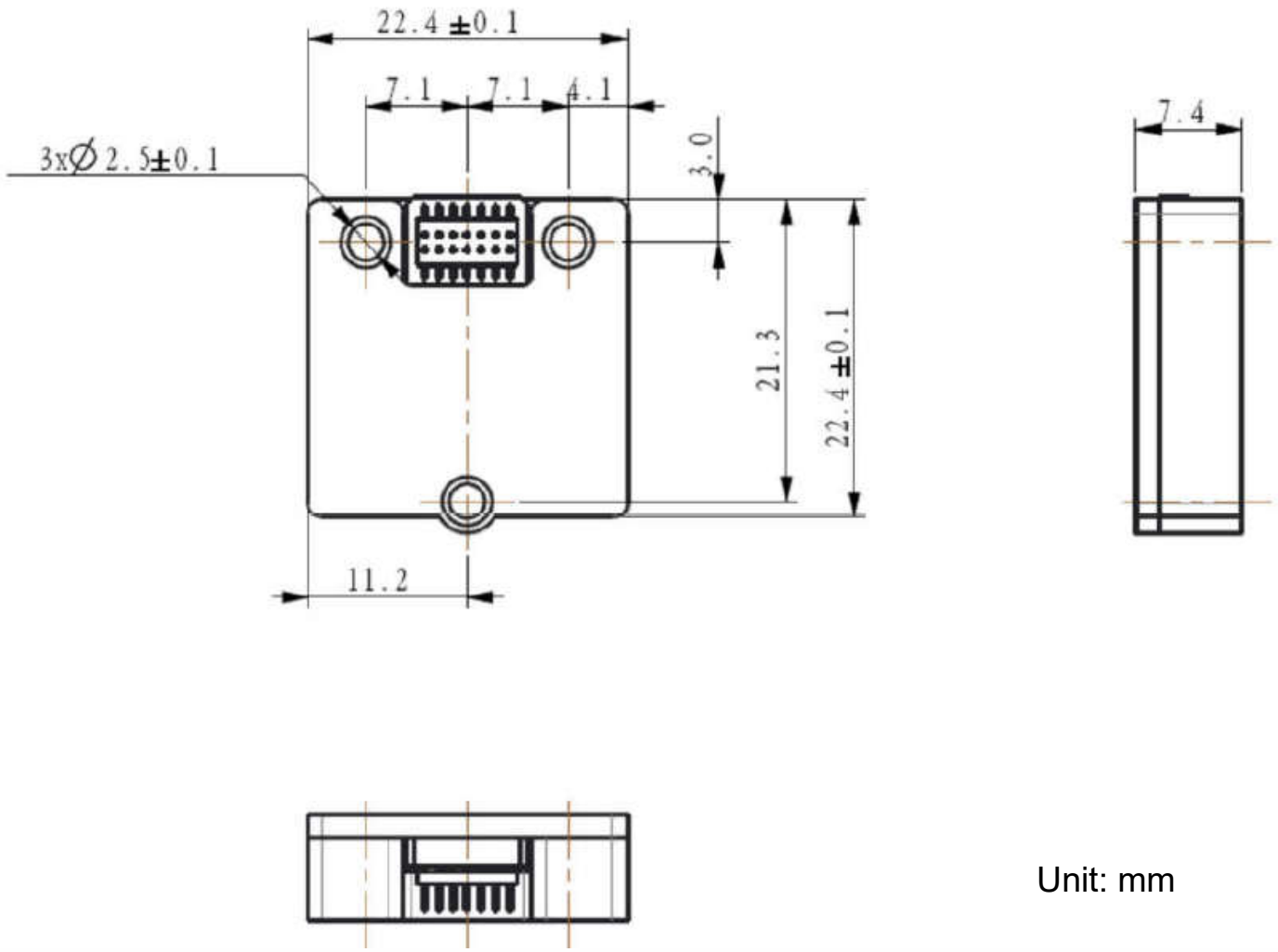
Pins Definition





Pins Definition		
Pins No.	**Name**	**Description**
1	DR	Data ready indicator
2	SYNC	External synchronization signal
3	SCLK	SPI serial clock
4	DOUT	SPI data output
5	DIN	SPI data input
6	$\overline{CS}$	SPI chip select
7	RX2	GPS signal UART_RX (LVTTL)
8	$\overline{RST}$	Reset
9	TX2	GPS signal UART_TX (LVTTL)
10	DNC	Do not connect
11	VDD	Power supply
12	RX1	IMU data UART_RX (LVTTL)
13	GND	Power ground
14	TX1	IMU data UART_TX (LVTTL)

Dimension



Unit: mm

Communication Protocol - UART

the electronic parameters of UART RS232 communicaiton protocol are as follows:

- Initial bit: 1
- Data bit: 8
- Stop bit: 1
- Check bit: none

each data string includes 23 bytes, the detailed description see as follows:

Data String Definition		
Byte No.	**Name**	**Description**
1	Synchronous byte 1	7FH (fixed value)
2	Synchronous byte 2	80H (fixed value)
3	X axis acceleration, low 8 bits	(20/65536)*N N=acceleration high 8 bits* 256+acceleration low 8 bits
4	X axis acceleration, high 8 bits	
5	Y axis acceleration, low 8 bits	(20/65536)*N N=acceleration high 8 bits* 256+acceleration low 8 bits
6	Y axis acceleration, high 8 bits	
7	Z axis acceleration, low 8 bits	(20/65536)*N N=acceleration high 8 bits* 256+acceleration low 8 bits
8	Z axis acceleration, high 8 bits	
9	X axis angular rate, low 8 bits	(1260/65536)*N N=angular rate high 8 bits* 256+angular rate low 8 bits
10	X axis angular rate, high 8 bits	
11	Y axis angular rate, low 8 bits	(1260/65536)*N N=angular rate high 8 bits* 256+angular rate low 8 bits
12	Y axis angular rate, high 8 bits	
13	Z axis angular rate, low 8 bits	(1260/65536)*N N=angular rate high 8 bits* 256+angular rate low 8 bits
14	Z axis angular rate, high 8 bits	
15	X axis roll angle, low 8bits	(360/65536)*N N= angle high 8 bits* 256+angle low 8 bits
16	X axis roll angle, high 8bits	
17	Y axis pitch angle, low 8bits	(360/65536)*N N= angle high 8 bits* 256+angle low 8 bits
18	Y axis pitch angle, high 8bits	
19	Z axis yaw angle, low 8bits	(360/65536)*N N= angle high 8 bits* 256+angle low 8 bits
20	Z axis yaw angle, high 8bits	
21	temperature sensor, low 8bits	(200/65536)*N N= temperature high 8 bits* 256+temperature low 8 bits
22	temperature sensor, low 8bits	
23	Checksum	sum of all bytes except synchronous bytes (3~22 bytes), then bit inversion

Communication Protocol - UART Data Analysis Example

```
double[] sensors = new double[9];

sensors[0] = (short)((byte)buffer[0] + ((byte)buffer[1] << 8));
sensors[1] = (short)((byte)buffer[2] + ((byte)buffer[3] << 8));
sensors[2] = (short)((byte)buffer[4] + ((byte)buffer[5] << 8));
sensors[3] = (short)((byte)buffer[6] + ((byte)buffer[7] << 8));
sensors[4] = (short)((byte)buffer[8] + ((byte)buffer[9] << 8));
sensors[5] = (short)((byte)buffer[10] + ((byte)buffer[11] << 8));
sensors[6] = (short)((byte)buffer[12] + ((byte)buffer[13] << 8));
sensors[7] = (short)((byte)buffer[14] + ((byte)buffer[15] << 8));
sensors[8] = (short)((byte)buffer[16] + ((byte)buffer[17] << 8));

// Accel x, y, z
Ax = sensors[0] * Accel_Scale / Sensor_Scale;
Ay = sensors[1] * Accel_Scale / Sensor_Scale;
Az = sensors[2] * Accel_Scale / Sensor_Scale;

// Gyro x, y, z
Gx = sensors[3] * Rate_Scale / Sensor_Scale;
Gy = sensors[4] * Rate_Scale / Sensor_Scale;
Gz = sensors[5] * Rate_Scale / Sensor_Scale;

// Roll, Pitch, Yaw
Roll = sensors[6] * Angle_Scale / Sensor_Scale;
Pitch = sensors[7] * Angle_Scale / Sensor_Scale;
Yaw = sensors[8] * Angle_Scale / Sensor_Scale;
```

Remarks:

1. acceleration unit is g, angular value unit is degree/s (°/s), attitude angle degree is degree (°).
2. Accel_Scale = 20, Rate_Scale = 1260, Angle_Scale = 360, Sensor_Scale = 65536

Communication Protocol - SPI

CNS16460 can be configured as SPI interface to communicate, it is used as slave device, and the master device should be configured as the following settings to communicate with CNS16460:

• 16 bit word length and the most significant bit (MSB) transmission

• $f_{CLK} \leq 1MHz$

• CPOL = 1 (Clock Polarity) and CPHA = 1 (Clock Phase)

1. SPI Burst-mode Read Data

SPI master device will access to SPI bus through the following mode: Burst mode

In Burst mode, CNS16460 will transmit a group of continuous SPI periodic predefined data block.

In Burst mode, the predefined data block that is returned with single group format by CNS16460 is called data package, and it doesn't need to send multiple read commands. The lengths of these groups depends on the selected data package

Standard Data Package SPI Burst-mode Read Mode			
Data Package	**Register Address**	**Data Length (16bytes)**	**Remarks**
Standard	0x46	8	the detailed introduction refers to the following content

The standard data package includes 8 predefined registers' data. The following table indicates the data name and analysis method, these registers will be listed according to the read sequence of Burst mode read.

Register Data		
Register Name	**Description**	**Analysis Method**
STATUS	*Unused*	*Unused*
X_RATE	X axis gyro output	(1260/65536) * N Note: N is integal with symbol N= (short) (angular rate high 8 bits* 256+angular rate low 8 bits)
Y_RATE	Y axis gyro output	
Z_RATE	Z axis gyro output	
X_ACCEL	X axis accelerometer output	(20/65536)*N Note: N is integal with symbol N= (short) acceleration high 8 bits* 256+acceleration low 8 bits
Y_ACCEL	Y axis accelerometer output	
Z_ACCEL	Z axis accelerometer output	
SENSOR_TEMP	Sensor Temperature	(200/65536)*N Note: N is integal with symbol N= (short) temperature high 8 bits* 256+temperature low 8 bits

Burst mode will be activated wen the master device requests to read data grouping (0x46) from Burst mode. It will take 8 extra SPI periods to finish the read. The following diagram indicates the Burst mode sequence. Attention: if SPI periods number is not correct, then the SPI transmission will be finished in advance or kept at Burst mode, and the following Read / Write will be not synchronized with CNS16460's SPI transmission period.

Communication Protocol - SPI

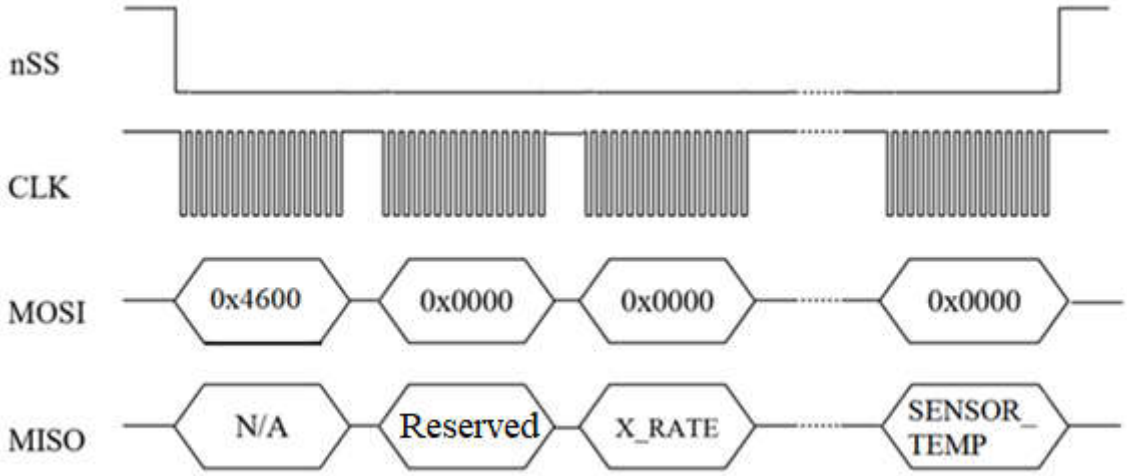
Attitude Data Package SPI Burst-mode Read Mode			
Data Package	**Register Address**	**Data Length (6bytes)**	**Remarks**
Angle	0x47	3	the detailed introduction refers to the following content

Attitude data package includes 3 predefined registers' data. The following table indicates the data name and analysis method, these registers will be listed according to the read sequence of Burst mode read.

Register Data		
Register Name	**Description**	**Analysis Method**
ROLL	Roll angle	Roll angle=N* 360 / 65536 Note: Unit:°, N is integal with symbol N= (short) (ROLL high 8 bits* 256+ ROLL low 8 bits)
PITCH	Pitch angle	Pitch angle=N* 360 / 65536 Note: Unit:°, N is integal with symbol N= (short) (PITCH high 8 bits* 256+ PITCH low 8 bits)
YAW	Yaw angle	Yaw angle=N* 360 / 65536 Note: Unit:°, N is integal with symbol N= (short) (YAW high 8 bits* 256+ YAW low 8 bits)

Burst mode will be activated wen the master device requests to read data grouping (0x46) from Burst mode. It will take 16 extra SPI periods to finish the read. The following diagram indicates the Burst mode sequence. Attention: if SPI periods number is not correct, then the SPI transmission will be finished in advance or kept at Burst mode, and the following Read / Write will be not synchronized with CNS16460's SPI transmission period.

Communication Protocol - SPI



Operation Remarks:

- The Burst mode read of other data groups will be similar as that of the standard group. The only difference with the above method is the corresponding registers' address and the data word number is different.

- It should be careful when switching the data packages, cause the returned value is invalid during the new data package's first Burst read time. It requires one read period to fill inner Bust mode register; and the follow-up information from the data package will include the valid information.

- During Burst read period, the chip select line (nSS) can be controlled through the following method:

  • During whole read period, pull down nSS and keep low level. After the transmission is finished, the chip select should be set as high level.

Communication Protocol - SPI

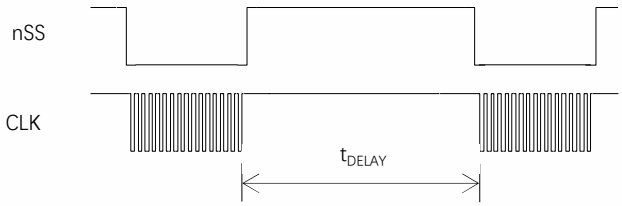
2. SPI Time Sequence

SPI time sequency requirements please see the following table and figure. and the following operation constraints apply to SPI communication:

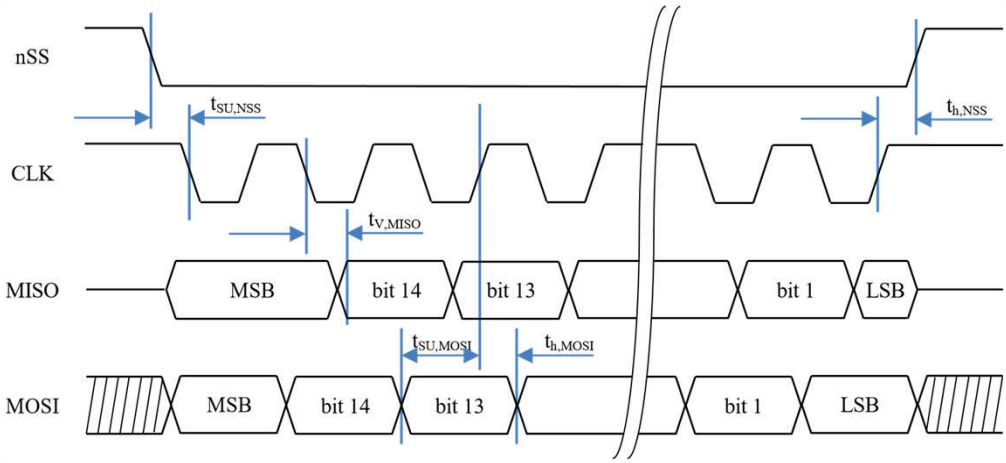
• CPOL = 1 (Clock Polarity) and CPHA = 1 (Clock Phase)

• 16 bit word length and the most significant bit (MSB) transmission

SPI Time Sequence Requirement			
Parameter	**Description**	**Value**	**Unit**
f_{CLK}	SPI clock frequency	1(max)	MHz
t_{DELAY}	Time between successive clock cycles	20(min)	usec
$t_{SU,NSS}$	nSS setup time prior to clocking data	1	usec
$t_{h,NSS}$	nSS hold time following clock signal	67	nsec
$t_{V,MISO}$	Time after falling edge of previous clock-edge that MISO data bit is invalid	25	nsec
$t_{SU,MOSI}$	Data input setup time prior to rising edge of clock	5	nsec
$t_{h,MOSI}$	Data input hold time following rising edge of clock	4	nsec



Relay time



SPI Time Sequence Figure