# CMPS10 - Tilt Compensated Compass Module

Introduction

The CMPS10 module is a tilt compensated compass. Employing a 3-axis magnetometer and a 3-axis accelerometer and a powerful 16-bit processor, the CMPS10 has been designed to remove the errors caused by tilting of the PCB. The CMPS10 produces a result of 0-3599 representing 0-359.9 or 0 to 255. The output of the three sensors measuring x, y and z components of the magnetic field, together with the pitch and roll are used to calculate the bearing, each of these components are also made available in there raw form. We have also written examples of using the CMPS10 module with a wide range of popular controllers. The CMPS10 module requires a power supply at 3.3 - 5v and draws a nominal 25mA of current. There are three ways of getting the bearing from the module. A serial interface, an I2C interface or a PWM output.
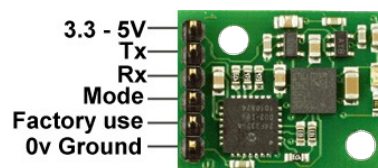
Mode selection

For data on each mode please click the mode heading. Note the CMPS10 looks at the mode selection pins at power-up only.
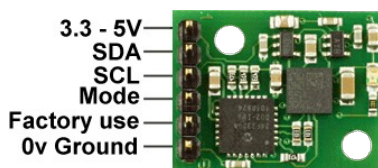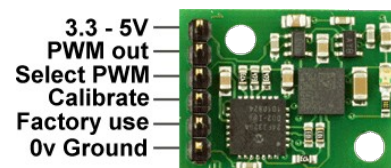
| I2C mode	Serial mode	PWM mode
To enter the I2C mode of operation leave the mode pin unconnected	To enter the serial mode of operation connect the mode pin to ground	To enter the PWM mode of operation connect the select PWM pin to ground

The **Factory use** pin is for our internal testing only, this pin should remain unconnected.

Data update frequency

Updates of the tilt compensated heading occur at 75hz with the data is filtered by means of a 45 sample buffer, this means a complete refresh of the buffer is achieved every 640ms. Raw data from the magnetometer and accelerometer is available every 13.3ms.

PCB Drilling Plan

The following diagram shows the CMPS10 PCB mounting hole positions.
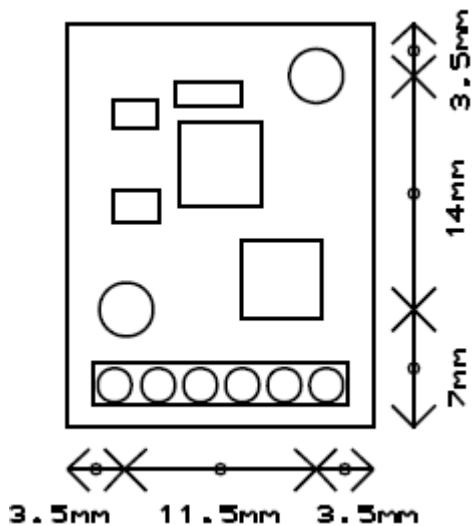
We have examples of using the Compass module with a wide range of popular controllers.

I2C mode

Connections

To enter the I2C mode of operation leave the mode pin unconnected

I2C Communication

I2C communication protocol with the compass module is the same as popular eeprom's such as the 24C04. First send a start bit, the module address with the read/write bit low, then the register number you wish to read. This is followed by a repeated start and the module address again with the read/write bit high. You now read one or two bytes for 8bit or 16bit registers respectively. 16bit registers are read high byte first. The compass has a 23 byte array of registers. organized as below:

Register	Function
0	Software version
1	Compass Bearing as a byte, i.e. 0-255 for a full circle
2,3	Compass Bearing as a word, i.e. 0-3599 for a full circle, representing 0-359.9 degrees.
4	Pitch angle - signed byte giving angle in degrees from the horizontal plane
5	Roll angle - signed byte giving angle in degrees from the horizontal plane
6	Unused
7	Unused
8	Unused
9	Unused
10,11	Magnetometer X axis raw output, 16 bit signed integer with register 10 being the upper 8 bits
12,13	Magnetometer Y axis raw output, 16 bit signed integer with register 12 being the upper 8 bits
14,15	Magnetometer Z axis raw output, 16 bit signed integer with register 14 being the upper 8 bits
16,17	Accelerometer X axis raw output, 16 bit signed integer with register 16 being the upper 8 bits
18,19	Accelerometer Y axis raw output, 16 bit signed integer with register 18 being the upper 8 bits
20,21	Accelerometer Z axis raw output, 16 bit signed integer with register 20 being the upper 8 bits
22	Command register

Register 1 is the bearing converted to a 0-255 value. This may be easier for some applications than 0-3599 which requires two bytes. For those who require better resolution registers 2 and 3 (high byte first) form a 16 bit unsigned integer in the range 0-3599. This represents 0-359.9°. Register 4 is the pitch angle, giving an angle of 0 when the board is flat and up to +/- 85° at maximum tilt in either direction. Register 5 works the same way but

with results for the Roll angle. There is then an array of registers (10-21) providing all the raw sensor data from the magnetic and acceleration sensors. Finally Register 22 is the command register and is used to calibrate the compass, change address and if necessary restore the factory default calibration.

Calibration the CMPS10

I would recommend evaluating the CMPS10 performance first before implementing this function. Its purpose is to remove offsets caused by constant magnetic sources around the CMPS10. First of all you need to determine North and align the CMPS10 with it, then enter the calibration mode by writing 0xF0 to the command register (22). To calibrate the first point write 0xF5 to the command register, this should also light the LED, then move the object through 90° and write 0xF5 to the command register. Repeat this twice more so four points are calibrated and the LED should also turn off to confirm calibration completion. Please make sure that the CMPS10 is not located near to ferrous objects as this will distort the magnetic field and induce errors in the reading.

Restoring Factory Calibration

Should you need to revert to the factory calibration then write the following to the command register in 3 separate transactions with 100ms between each transaction: 0x20,0x2A,0x60. These commands must be sent in the correct sequence to restore the calibration, additionally, No other command may be issued in the middle of the sequence. The sequence must be sent to the command register at location 22, which means 3 separate write transactions on the I2C bus. (A write transaction is [start sequence] [I2C address] [register address] [command byte] [stop sequence] then a 100mS delay).

Changing the I2C Bus Address

To change the I2C address of the CMPS10 you must have only one module on the bus. Write the 3 sequence commands in the correct order followed by the address with 100ms between writes. Example; to change the address of a compass currently at 0xC0 (the default shipped address) to 0xC2, write the following to address 0xC0; (0xA0, 0xAA, 0xA5, 0xC2) with a 100ms delay after each of the first three bytes. These commands must be sent in the correct sequence to change the I2C address, additionally, No other command may be issued in the middle of the sequence. The sequence must be sent to the command register at location 22, which means 4 separate write transactions on the I2C bus. When done, you should label the CMPS10 with its address, however if you do forget, just power it up without sending any commands. The CMPS10 will flash its address out on the LED. One long flash followed by a number of shorter flashes indicating its address. The flashing is terminated immediately on sending a command the CMPS10.
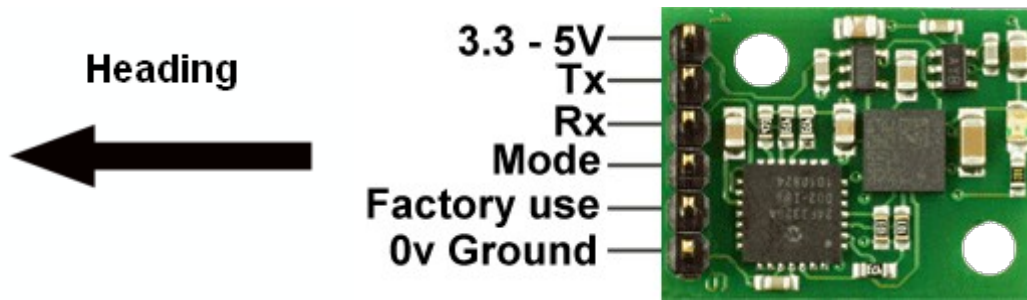
Address		Long Flash	Short flashes
Decimal	Hex		
192	C0	1	0
194	C2	1	1
196	C4	1	2
198	C6	1	3
200	C8	1	4
202	CA	1	5
204	CC	1	6
206	CE	1	7

Take care not to set more than one device to the same address, there will be a bus collision and very unpredictable results.

If you need more information on the I2C bus, have a look at our I2C tutorial.

Serial mode

Connections

To use the serial mode of operation the mode pin must be connected to ground.

Communication settings

The Serial mode operates over a link with a default baud rate of 9600 bps (no parity, 2 stop bits) and 3.3v-5v signal levels.

This is not RS232. Do not connect RS232 to the module, the high RS232 voltages will irreversibly damage the module.

Commands

Below is a table describing commands that can be sent to the CMPS10 and the data it will respond with.

Commands for Serial

Command	Name	Bytes returned	Returned data description
0x11	GET VERSION	1	Software version
0x12	GET ANGLE 8 BIT	1	Angle as a single byte 0-255
0x13	GET ANGLE 16 BIT	2	Angle as two bytes, high byte first 0-3600
0x14	GET PITCH	1	Pitch angle +/- 0-85°
0x15	GET ROLL	1	Roll angle +/- 0-85°
0x21	GET MAG RAW	6	Raw magnetic data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x22	GET ACCEL RAW	6	Raw accelerometer data, 16 bit signed: X high, X low, Y high, Y low, Z high, Z low
0x23	GET ALL	4	angle high, angle low (0-3600), pitch (+/- 0-85), roll (+/- 0-85)
0x31	CALIBRATE EN1	1	returns ok (0x55)
0x45	CALIBRATE EN2	1	returns ok (0x55)
0x5A	CALIBRATE EN3	1	returns ok (0x55)
0x5E	CALIBRATE	1	returns ok (0x55)
0x6A	RESTORE 1	1	returns ok (0x55)
0x7C	RESTORE 2	1	returns ok (0x55)
0x81	RESTORE 3	1	returns ok (0x55)
0xA0	BAUD 19200	1	returns ok (0x55)
0xA1	BAUD 38400	1	returns ok (0x55)

Calibration the CMPS10

I would recommend evaluating the CMPS10 performance first before implementing this function. Its purpose is to remove offsets caused by constant magnetic sources around the

CMPS10. First of all you need to determine North and align the CMPS10 with it, then write a sequence of 3 commands in the correct order with a small delay between bytes, 100ms will be more than adequate. The sequence to enter calibration mode is 0x31,0x45,0x5A, then calibrate the first point by sending 0x5E to the command register, this should also light the LED. The Compass should then be rotated 90° and 0x5E sent to the command register again, repeat for two further 90° rotations and the calibration completes and the LED turns off. Please make sure that the CMPS10 is not located near to ferrous objects as this will distort the magnetic field and induce errors in the reading.

Restore of factory calibration of the CMPS10
To perform a restore of the factory calibration write a sequence of 3 commands in the correct order with a small delay between bytes, 100ms will be more than adequate. The sequence is 0x6A,0x7C,0x81.

Changing the baud rate
While the CMPS10 operates at a default serial bus baud rate of 9600 you may wish to change this. There are two other baud rates that can be used, for 19200 just send 0xA0 or alternatively for 38400 send 0xA1. Please note that the CMPS10 will always default to its 9600kbps rate after power cycling and after setting a new baud rate the ok response (0x55) will be sent at the newly selected speed.

PWM mode

To aid the more basic controllers we have included a PWM mode of operation. As the compass is rotated a high pulse will be generated that is proportional to the current angle. The pulse width varies from 1mS (0°) to 36.99mS (359.9°) $\frac{3}{8}$ in other words 100uS/° with a +1mS offset. The signal goes low for 65mS between pulses, so the cycle time is 65mS + the pulse width - ie. 66ms-102ms.

Pin connections and mode selection for PWM operation
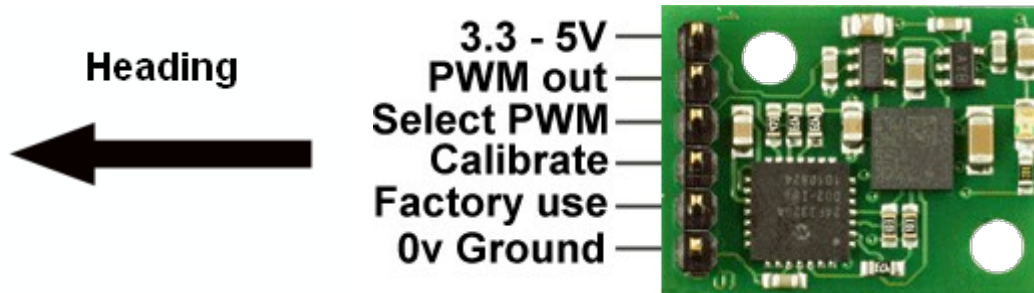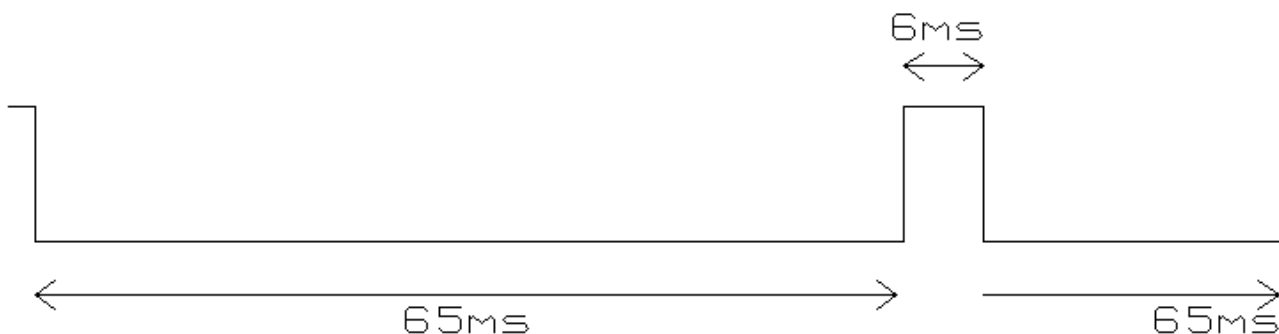

To enter the PWM mode of operation you are required to connect the Select PWM pin to ground. After 500ms the module will enter PWM mode and output a variable high pulse from the PWM out pin dependant on the angle of the PCB.

Example 1: Pulse of 20ms - 1ms offset = 19ms = 190°

Example 2: Pulse of 6ms - 1ms offset = 5ms = 50°

Calibration

In some environments it may be necessary to perform a calibration to remove hard iron distortion, however I would recommend evaluating the CMPS10 performance first before implementing the routine. To achieve this a normally open switch must be wired between the calibrate pin and ground. The first step is to line the compass up with north, then press the switch and the CMPS10 will light is LED. Rotate the module by 90° and repeat 3 further times, the LED should now go out and the module will be recalibrated. Should you wish to restore the factory calibration then press and hold the switch for at least 5 seconds before releasing. Please make sure that the CMPS10 is not located near to ferrous objects as this will distort the magnetic field and induce errors in the reading.