

# Triple Axis Magnetometer Breakout

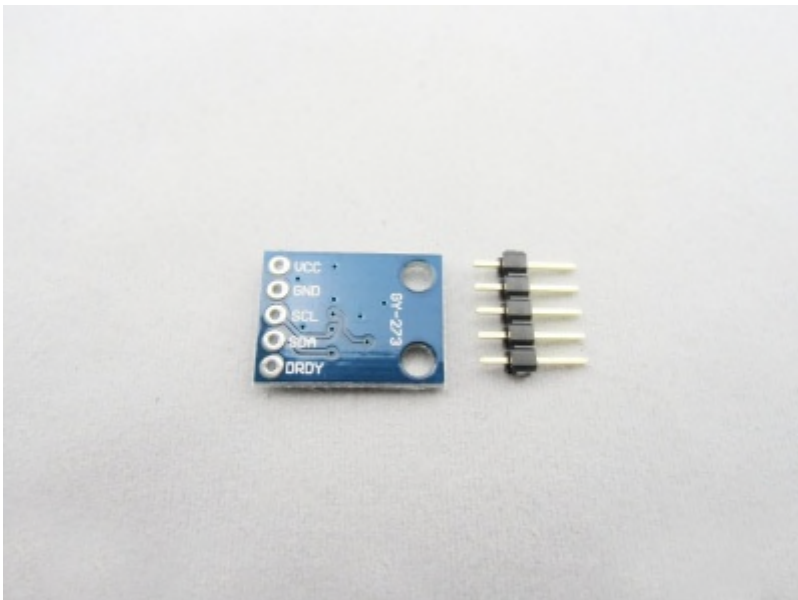
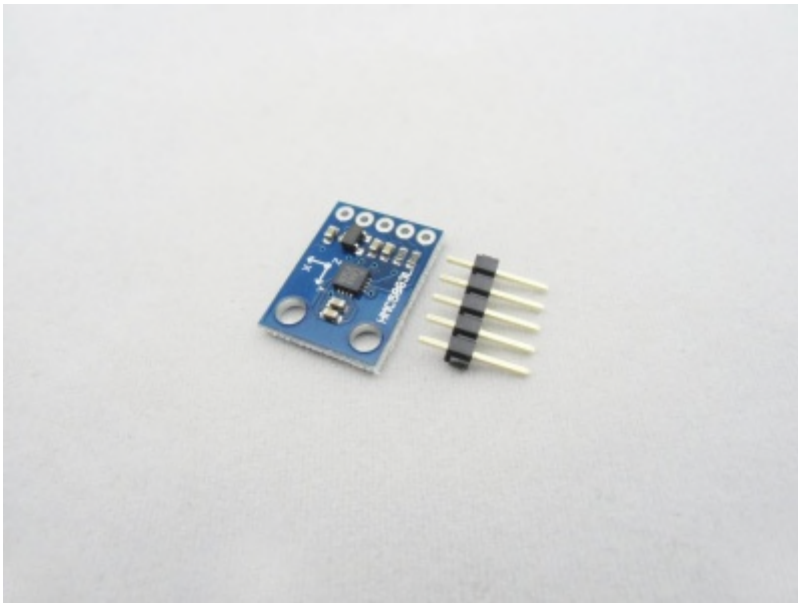
---

## Introduction

---

Description: This is a breakout board for Honeywell's HMC5883L, a 3-axis digital compass, provides compass heading accuracy up to 1° to 2°. Communication with the HMC5883L is simple and all done through an I2C interface. An Arduino Demo is given out to help users learn how to use this sensor.

The breakout board includes the HMC5883L sensor and all filtering capacitors as shown. This module also includes a DC regulator for power supply requirement. Hence user can connect any 3.3V to 6V DC power supply.



## Features

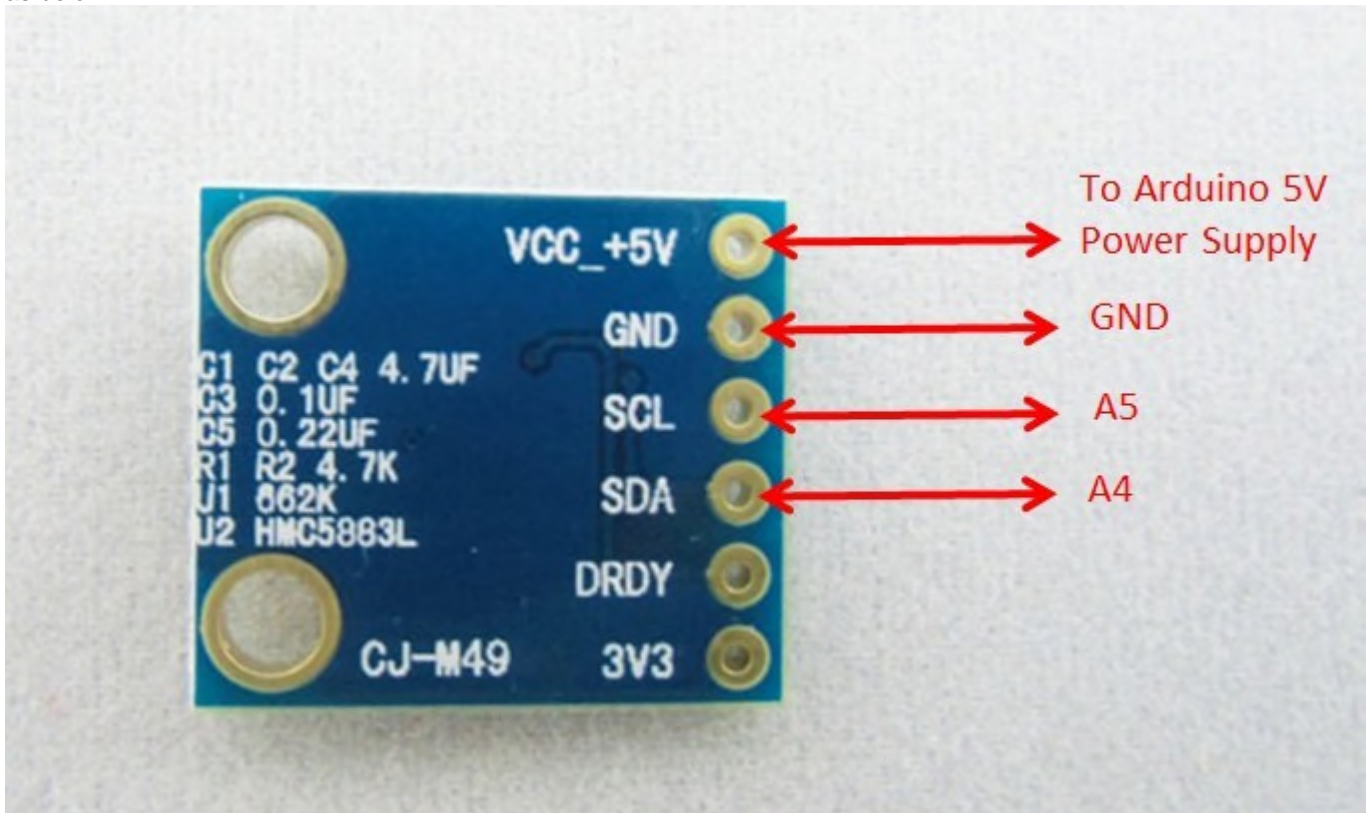
---

- Simple I2C interface.
  - 3.3–6V supply range.
  - Low current draw.
  - Dimensions: 18 \* 16 \* 1.6mm.
-

# Usage

## Hardware

Connect this triple axis magnetometer breakout module to your Arduino/AZLduino I2C wires (SDA:A4, D18;SCL:A5, D19) as below :



## Programming

First download the [HMC5883L library](#) to your PC see - [Triple Axis Magnetometer Breakout \(pdf file\)](#)

please refer to : [How to install the library and upload programs to Arduino \(pdf file\)](#)

you can begin your user application program as below:

```
#include <Wire.h>
#include <HMC5883L.h>
HMC5883L compass;
int error = 0;
void setup()
{
  Serial.begin(9600);

  Serial.println("Starting the I2C interface.");
  Wire.begin();

  Serial.println("Constructing new HMC5883L");
  compass = HMC5883L();

  Serial.println("Setting scale to +/- 1.3 Ga");
  error = compass.SetScale(1.3);
```

```

if(error != 0)
    Serial.println(compass.GetErrorText(error));

Serial.println("Setting measurement mode to continuous.");
error = compass.SetMeasurementMode(Measurement_Continuous); // Set the measurement
mode to Continuous
if(error != 0)
    Serial.println(compass.GetErrorText(error));
}
void loop()
{
    // Retrieve the raw values from the compass (not scaled).
    MagnetometerRaw raw = compass.ReadRawAxis();
    // Retrieved the scaled values from the compass (scaled to the configured scale).
    MagnetometerScaled scaled = compass.ReadScaledAxis();

    int MilliGauss_OnThe_XAxis = scaled.XAxis; // (or YAxis, or ZAxis)

    // Calculate heading when the magnetometer is level, then correct for signs of axis.
    float heading = atan2(scaled.YAxis, scaled.XAxis);

    // Once you have your heading, you must then add your 'Declination Angle', which is
the 'Error' of the magnetic field in your location.
    // Find yours here: http://www.magnetic-declination.com/
    float declinationAngle = -0.0457;
    heading += declinationAngle;
    if(heading < 0)
        heading += 2*PI;
    if(heading > 2*PI)
        heading -= 2*PI;

    // Convert radians to degrees for readability.
    float headingDegrees = heading * 180/M_PI;

    // Output the data via the serial port.
    Output(raw, scaled, heading, headingDegrees);
    delay(66); //of course it can be delayed longer.
}

// Output the data down the serial port.
void Output(MagnetometerRaw raw, MagnetometerScaled scaled, float heading, float
headingDegrees)
{
    Serial.print("Raw:\t");
    Serial.print(raw.XAxis);
    Serial.print("  ");
    Serial.print(raw.YAxis);
    Serial.print("  ");

```

```
Serial.print(raw.ZAxis);  
Serial.print(" \tScaled:\t");  
  
Serial.print(scaled.XAxis);  
Serial.print(" ");  
Serial.print(scaled.YAxis);  
Serial.print(" ");  
Serial.print(scaled.ZAxis);  
  
Serial.print(" \tHeading:\t");  
Serial.print(heading);  
Serial.print(" Radians \t");  
Serial.print(headingDegrees);  
Serial.println(" Degrees \t");  
}
```

Open a Sscom32 terminal, and set the baud rate to 9600, you will see the output change with the sensor orientation

## Resource

See - [Datasheet - HMC5883L-FDS \(pdf file\)](#)

[File : Compass breakout demo code for Arduino 10.zip](#)