Adafruit MMC5603 - Triple-axis Magnetometer

Created by Liz Clark

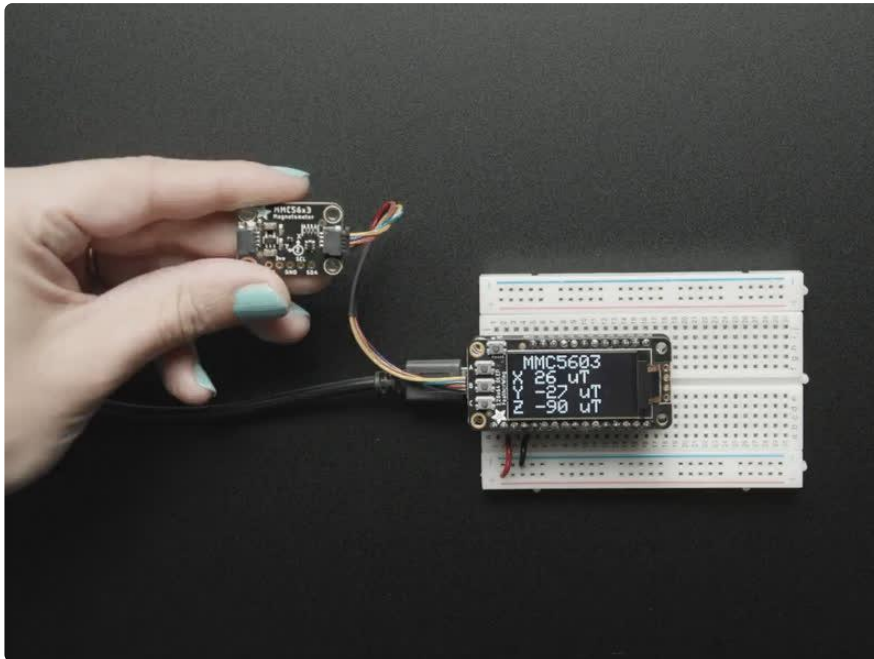https://learn.adafruit.com/adafruit-mmc5603-triple-axis-magnetometer
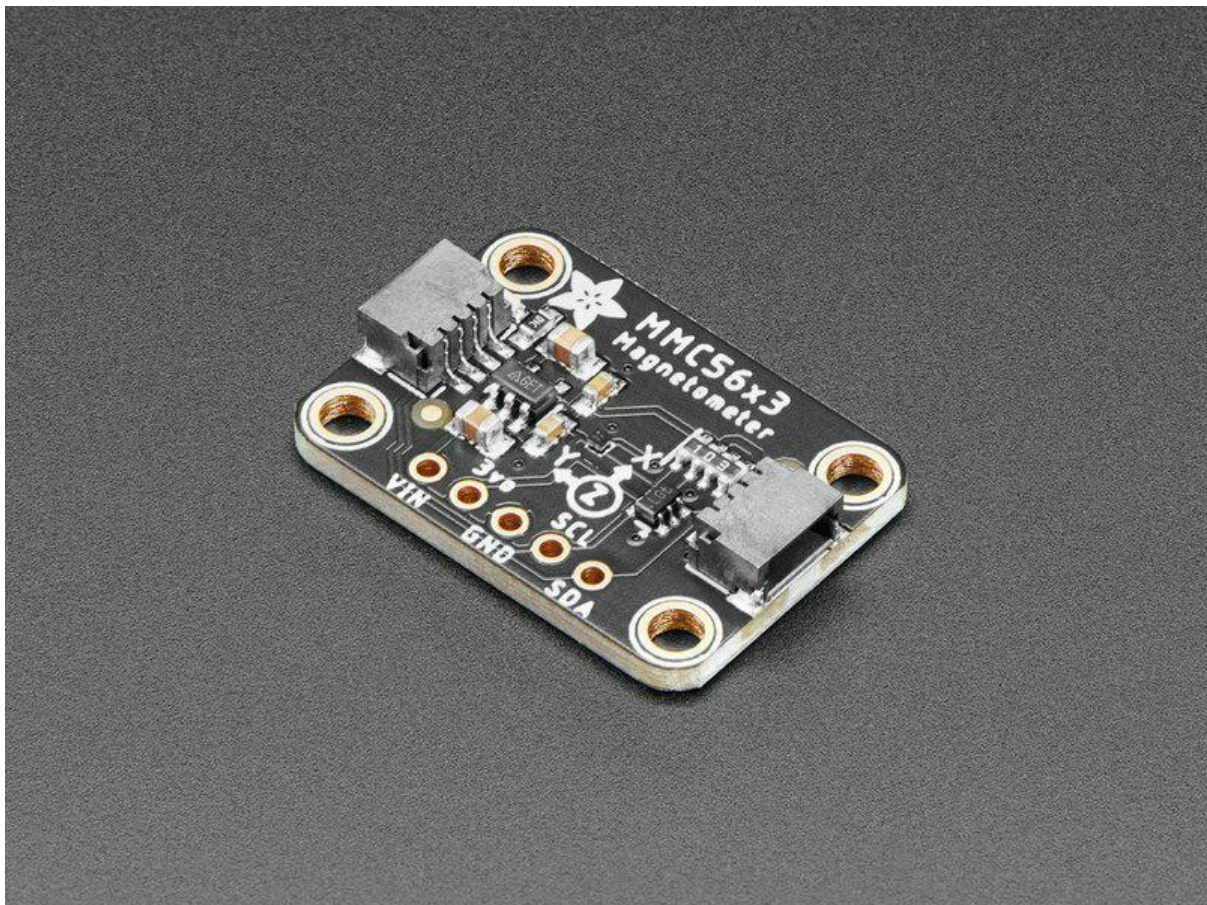
Last updated on 2022-08-31 10:48:55 AM EDT

Table of Contents

Overview

Sense the magnetic fields surrounding us with this handy triple-axis magnetometer (compass) module. Magnetometers can sense where the strongest magnetic force is coming from, generally used to detect magnetic north, but can also be used for measuring magnetic fields. This sensor tends to be paired with a 6-DoF (degree of freedom) accelerometer/gyroscope to create a 9-DoF inertial measurement unit that can detect its orientation in real-space, thanks to Earth's stable magnetic field. It's a great match for the LSM6DSOX from ST! (https://adafru.it/HNb)

We based this breakout on the MMC5603, a great general purpose magnetometer with a very wide range and both I2C and SPI interfaces. This compact sensor is very easy to use. Simply download our library and connect the SCL pin to your I2C clock pin, and SDA pin to your I2C data pin and upload our test program to read out magnetic field data. If you'd like, you can also use SPI to receive data (we just happen to prefer I2C here).
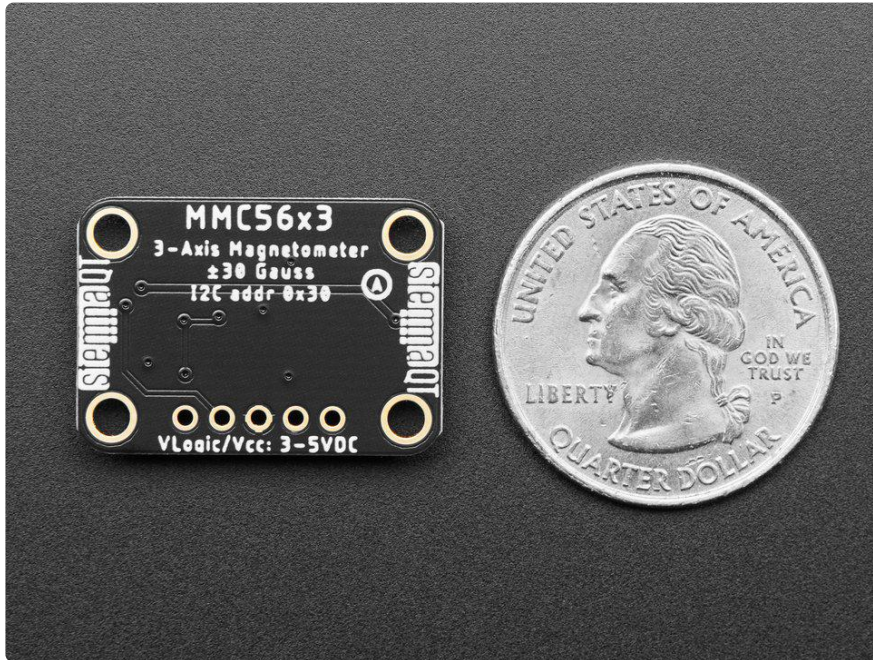

This sensor can sense ranges from ±30 Gauss (±3000uT or ±3mT) with no range-setting required and full 20 bit output, up to 1000 Hz rate reading. The range makes it good for reading Earth's magnetic field (which maxes at about 0.6 Gauss) or some basic magnets. It isn't good for very strong rare earth magnets, for that check out the TLV293 (https://adafru.it/IES).
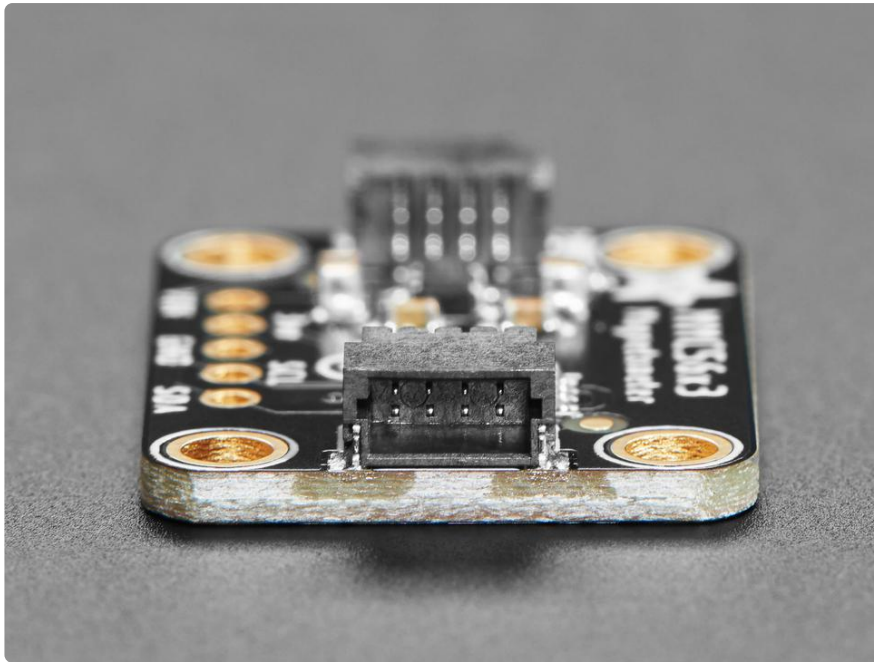
To make life easier, so you can focus on your important work, we've taken the MMC5603 and put it onto a breakout PCB along with support circuitry to let you use this little wonder with 3.3V (Feather/Raspberry Pi) or 5V (Arduino/ Metro328) logic levels. Additionally, since it speaks I2C, you can easily connect it up with two wires (plus power and ground!). We've even included SparkFun qwiic (https://adafru.it/Fpw) compatible STEMMA QT (https://adafru.it/Ft4) connectors for the I2C bus, so you don't even need to solder! Just wire up to your favorite micro and you can use our CircuitPython/Python (https://adafru.it/10vb) or Arduino drivers (https://adafru.it/10ne) to easily interface with the MMC5603 and get magnetic measurements ASAP.

It's fully assembled and tested. Comes with a bit of 0.1" standard header in case you want to use it with a breadboard or perfboard. Four 2.5mm (0.1") mounting holes for easy attachment.

Pinouts
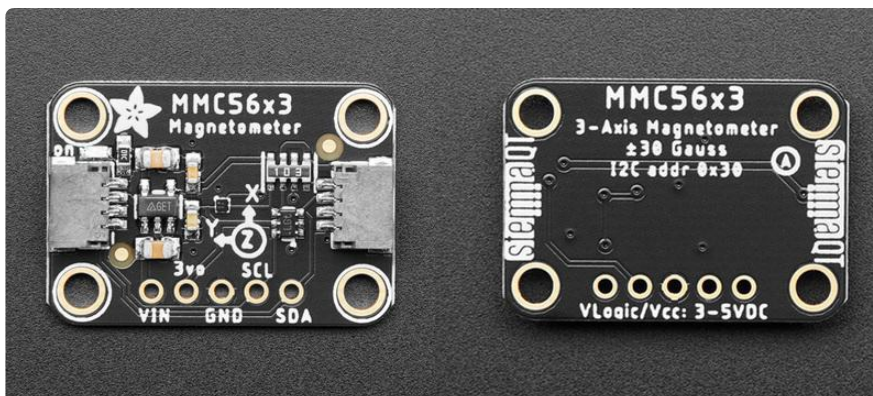
The default I2C address is 0x30.

Power Pins

- VIN - this is the power pin. Since the sensor chip uses 3 VDC, we have included a voltage regulator on board that will take 3-5VDC and safely convert it down. To power the board, give it the same power as the logic level of your microcontroller - e.g. for a 5V microcontroller like Arduino, use 5V.
- 3vo - this is the 3.3V output from the voltage regulator, you can grab up to 100mA from this if you like.
- GND - common ground for power and logic.

I2C Logic Pins

- SCL - I2C clock pin, connect to your microcontroller I2C clock line. This pin is level shifted so you can use 3-5V logic, and there's a 10K pullup on this pin.
- SDA -I2C data pin, connect to your microcontroller I2C data line. This pin is level shifted so you can use 3-5V logic, and there's a 10K pullup on this pin.
- STEMMA QT (https://adafru.it/Ft4) - These connectors allow you to connectors to dev boards with STEMMA QT connectors or to other things with various associated accessories (https://adafru.it/Ft6).

Power LED

- Power LED - In the upper left corner, above the STEMMA connector, on the front of the board, is the power LED, labeled on. It is the green LED.

Python & CircuitPython

It's easy to use the MMC5603 with Python or CircuitPython, and the Adafruit_CircuitPython_MMC56x3 (https://adafru.it/10vb) module. This module allows you to easily write Python code that reads the values from the MMC5603's magnetometer and ambient temperature sensor. You can use this sensor with any CircuitPython microcontroller board or with a computer that has GPIO and Python thanks to Adafruit_Blinka, our CircuitPython-for-Python compatibility library (https://adafru.it/BSN).

CircuitPython Microcontroller Wiring

First, wire up a MMC5603 to your board exactly as shown below. Here's an example of wiring a Feather M4 to the sensor with I2C using one of the handy STEMMA QT (htt ps://adafru.it/Ft4) connectors:
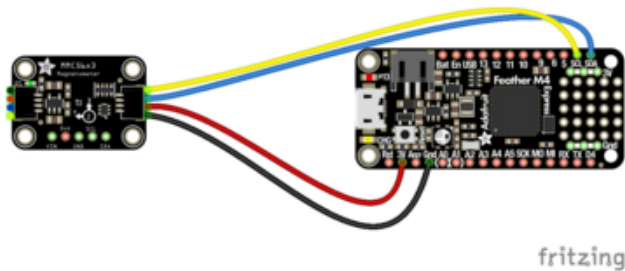
Board 3V to sensor VIN (red wire)
Board GND to sensor GND (black wire)
Board SCL to sensor SCL (yellow wire)
Board SDA to sensor SDA (blue wire)

You can also use standard 0.100" pitch headers to wire it up on a breadboard:
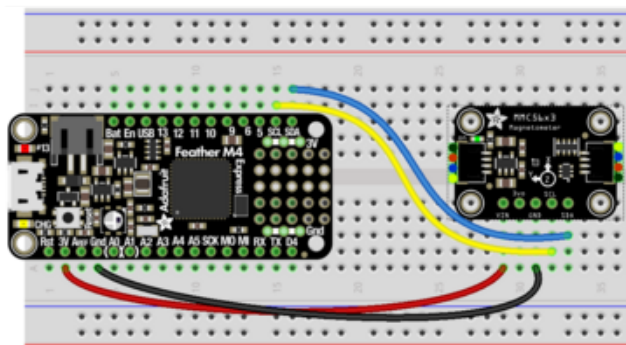
Board 3V to sensor VIN (red wire)
Board GND to sensor GND (black wire)
Board SCL to sensor SCL (yellow wire)
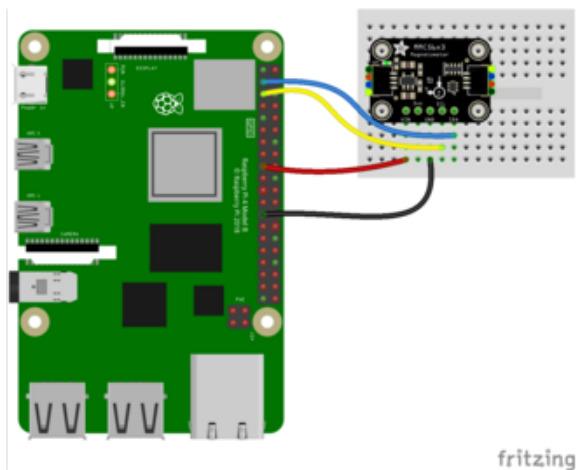Board SDA to sensor SDA (blue wire)

Python Computer Wiring

Since there's dozens of Linux computers/boards you can use, below shows wiring for Raspberry Pi. For other platforms, please visit the guide for CircuitPython on Linux to see whether your platform is supported (https://adafru.it/BSN).

Here's the Raspberry Pi wired to the sensor using I2C and a STEMMA QT (https:// adafru.it/Ft4) connector:
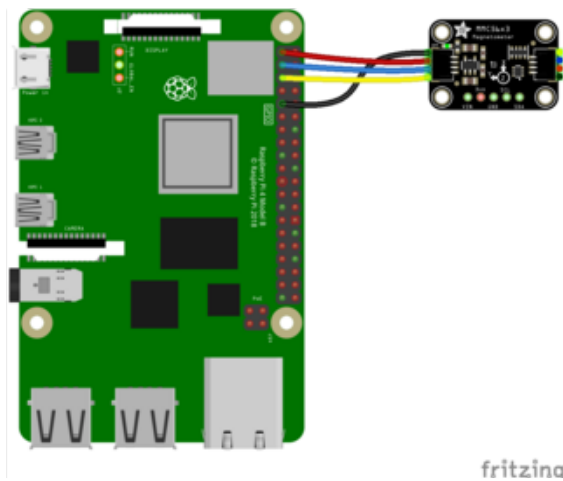
Pi 3V to sensor VIN (red wire)
Pi GND to sensor GND (black wire)
Pi SCL to sensor SCL (yellow wire)
Pi SDA to sensor SDA (blue wire)

Finally here is an example of how to wire up a Raspberry Pi to the sensor using a solderless breadboard:

Pi 3V to sensor VIN (red wire)
Pi GND to sensor GND (black wire)
Pi SCL to sensor SCL (yellow wire)
Pi SDA to sensor SDA (blue wire)

Python Installation of MMC56x3 Library

You'll need to install the Adafruit_Blinka library that provides the CircuitPython support in Python. This may also require enabling I2C on your platform and verifying you are running Python 3. Since each platform is a little different, and Linux changes often, please visit the CircuitPython on Linux guide to get your computer ready (https://adafru.it/BSN)!

Once that's done, from your command line run the following command:

- `pip3 install adafruit-circuitpython-mmc56x3`

If your default Python is version 3, you may need to run `pip` instead. Make sure you aren't trying to use CircuitPython on Python 2.x, it isn't supported!
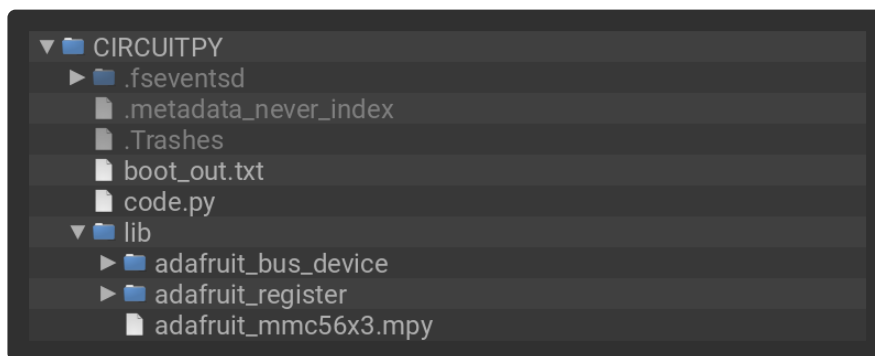
CircuitPython Usage

To use with CircuitPython, you need to first install the MMC56x3 library, and its dependencies, into the lib folder on your CIRCUITPY drive. Then you need to update code.py with the example script.

Thankfully, we can do this in one go. In the example below, click the Download Project Bundle button below to download the necessary libraries and the code.py file in a zip file. Extract the contents of the zip file, and copy the entire lib folder and the code.py file to your CIRCUITPY drive.

Your CIRCUITPY/lib folder should contain the following folders and file:

- adafruit_bus_device/
- adafruit_register/
- adafruit_mmc56x3.mpy

```
▼ 📁 CIRCUITPY
   ▶ 📁 .fseventsd
     📄 .metadata_never_index
     📄 .Trashes
     📄 boot_out.txt
     📄 code.py
   ▼ 📁 lib
     ▶ 📁 adafruit_bus_device
     ▶ 📁 adafruit_register
       📄 adafruit_mmc56x3.mpy
```

Python Usage

Once you have the library `pip3` installed on your computer, copy or download the following example to your computer, and run the following, replacing code.py with whatever you named the file:

`python3 code.py`

Example Code

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

""" Display magnetometer data once per second """

import time
```

```
import board
import adafruit_mmc56x3

i2c = board.I2C()  # uses board.SCL and board.SDA
sensor = adafruit_mmc56x3.MMC5603(i2c)

while True:
    mag_x, mag_y, mag_z = sensor.magnetic
    temp = sensor.temperature

    print(
        "X:{0:10.2f}, Y:{1:10.2f}, Z:{2:10.2f} uT\tTemp:{3:6.1f}*C".format(
            mag_x, mag_y, mag_z, temp
        )
    )
    print("")
    time.sleep(1.0)
```

If running CircuitPython: Once everything is saved to the CIRCUITPY drive, connect to the serial console (https://adafru.it/Bec) to see the data printed out!

If running Python: The console output will appear wherever you are running Python.

Twist and turn your MMC5603 and see the values from the magnetometer and temperature sensor print out to the REPL!

First you import the necessary modules and libraries. Then you instantiate the sensor on I2C.

Then you're ready to read data from the sensor's magnetometer and temperature sensor.

Finally, inside the loop, you see the x, y and z values from the magnetometer alongside the ambient temperature reading.

The magnetometer measures force on the x, y and z axes in micro-Teslas (uT) and the temperature sensor reads the ambient temperature in Celsius.

That's all there is to using the MMC5603 with CircuitPython!

Python Docs

Python Docs (https://adafru.it/10nd)

Arduino

Using the MMC5603 with Arduino involves wiring up the sensor to your Arduino-compatible microcontroller, installing the Adafruit_MMC56x3 (https://adafru.it/10ne) library and running the provided example code.

Wiring

Wire as shown for a 5V board like an Uno. If you are using a 3V board, like an Adafruit Feather, wire the board's 3V pin to the MMC5603 VIN.

Here is an Adafruit Metro wired up to the MMC5603 using the STEMMA QT connector:
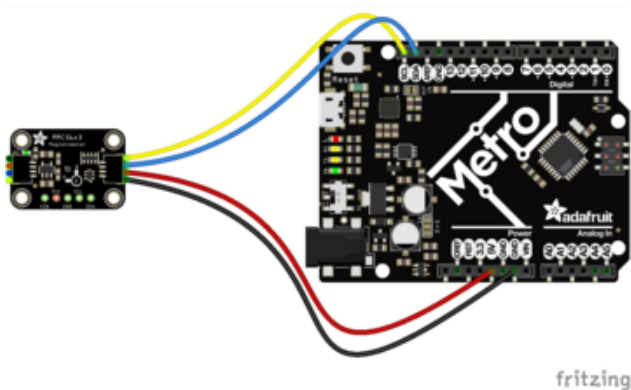
Board 5V to sensor VIN (red wire)
Board GND to sensor GND (black wire)
Board SCL to sensor SCL (yellow wire)
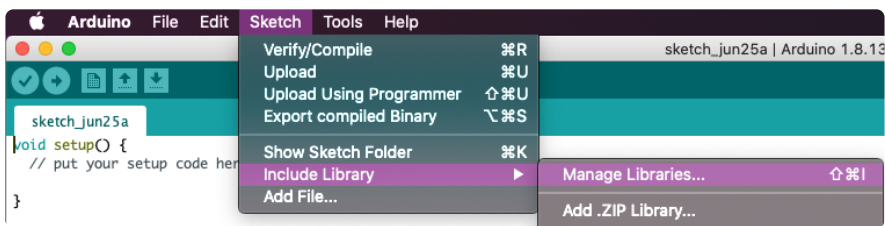Board SDA to sensor SDA (blue wire)

Here is an Adafruit Metro wired up using a solderless breadboard:
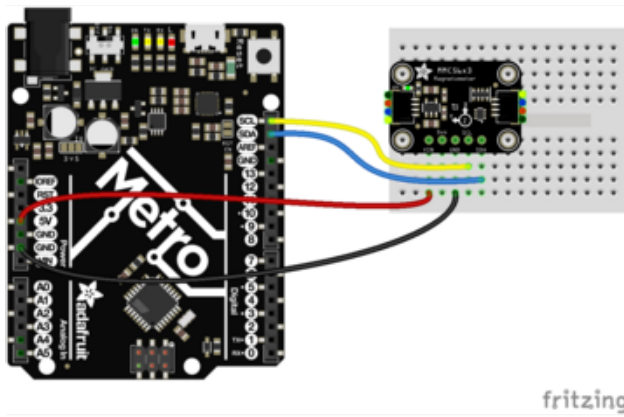
Board 5V to sensor VIN (red wire)
Board GND to sensor GND (black wire)
Board SCL to sensor SCL (yellow wire)
Board SDA to sensor SDA (blue wire)
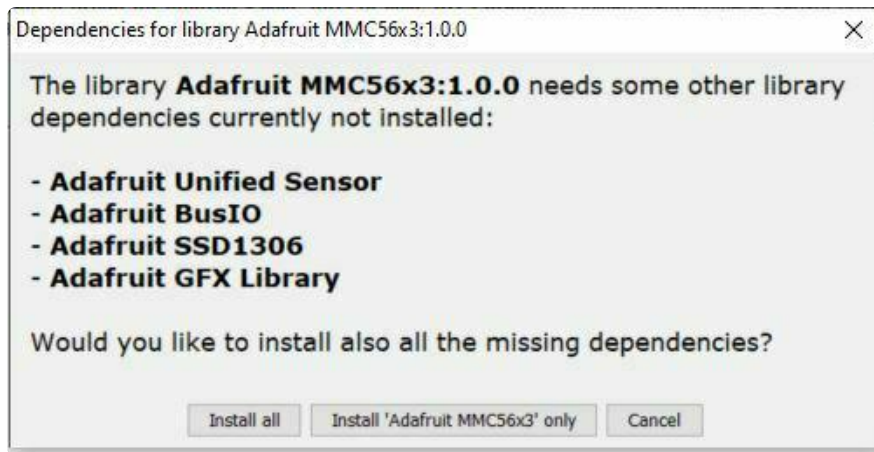
Library Installation

You can install the Adafruit MMC56x3 library for Arduino using the Library Manager in the Arduino IDE.

Click the Manage Libraries ... menu item, search for Adafruit MMC56x3, and select the Adafruit MMC56x3 library:
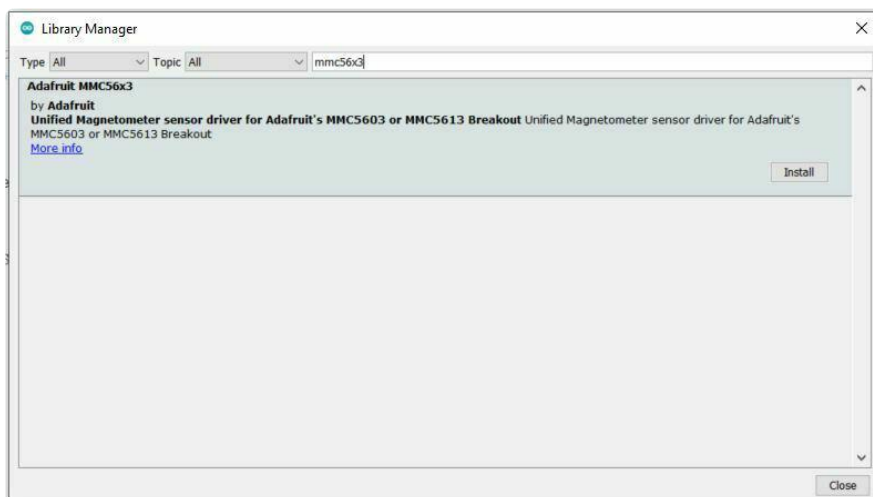
If asked about dependencies, click "Install all".

If the "Dependencies" window does not come up, then you already have the dependencies installed.

> If the dependencies are already installed, you must make sure you update them through the Arduino Library Manager before loading the example!

Example Code

```cpp
#include <Adafruit_MMC56x3.h>

/* Assign a unique ID to this sensor at the same time */
Adafruit_MMC5603 mmc = Adafruit_MMC5603(12345);

void setup(void) {
  Serial.begin(115200);
  while (!Serial)
    delay(10); // will pause Zero, Leonardo, etc until serial console opens

  Serial.println("Adafruit_MMC5603 Magnetometer Test");
  Serial.println("");

  /* Initialise the sensor */
  if (!mmc.begin(MMC56X3_DEFAULT_ADDRESS, &Wire)) {  // I2C mode
    /* There was a problem detecting the MMC5603 ... check your connections */
    Serial.println("Ooops, no MMC5603 detected ... Check your wiring!");
    while (1) delay(10);
  }

  /* Display some basic information on this sensor */
  mmc.printSensorDetails();
}

void loop(void) {
  // Get a new sensor event
  sensors_event_t event;
  mmc.getEvent(&event);

  // Display the results (magnetic vector values are in micro-Tesla (uT))
  Serial.print("X: ");
  Serial.print(event.magnetic.x);
  Serial.print("  ");
  Serial.print("Y: ");
  Serial.print(event.magnetic.y);
  Serial.print("  ");
```

```
  Serial.print("Z: ");
  Serial.print(event.magnetic.z);
  Serial.print("  ");
  Serial.println("uT");

  // Read and display temperature
  float temp_c = mmc.readTemperature();
  Serial.print("Temp: "); Serial.print(temp_c); Serial.println(" *C");
  // Delay before the next sample
  delay(100);
}
```


Upload the sketch to your board and open up the Serial Monitor (Tools -> Serial Monitor) at 115200 baud. You should see the values from the embedded temperature sensor and magnetometer being printed out. You'll see the values change depending on the movement of the sensor.

Arduino Docs

Arduino Docs (https://adafru.it/10ne)

Downloads

Files

- MMC5603 Datasheet (https://adafru.it/10vc)
- EagleCAD PCB files on GitHub (https://adafru.it/10vd)
- Fritzing object in the Adafruit Fritzing Library (https://adafru.it/10ve)

Schematic and Fab Print