Adafruit SensorLab - Gyroscope Calibration

Created by lady ada

Uncalibrated Gyro

Calibrated Gyro

https://learn.adafruit.com/adafruit-sensorlab-gyroscope-calibration

Last updated on 2023-08-29 04:23:15 PM EDT
# Table of Contents

- **Gyroscope Calibration**
- **Install SensorLab**
  - Install SensorLab
- **Simple Gyro Calibration**
  - Step 1 - Upload the SensorLab zero rate simplecal Example
  - Step 2 - Open Serial Port
- **Gyro Calibration with Jupyter**
  - Step 1 - Download Calibration Notebook
  - Step 2 - Upload the SensorLab imucal Example
  - Configure the notebook
- **Comparing Gyroscopes**
  - ST ISM330DHC
  - LSM6DSOX
  - LSM6DS33
  - LSM6DS3TR-C
  - LSM9DS1
  - MPU-6050
  - NXP FXAS21002
  - ICM-20649

©Adafruit Industries  
Page 2 of 20
Digital gyroscopes are just like those spinner toys you've seen when you're a kid, where you pull the string and you can balance the whole thing on your fingertip easily. The ones in chips don't have metal rings, instead they use very very small springs that help it measure rotation.

Compared to accelerometers and even magnetometers, digital gyros are surprisingly complex, but thanks to their existence we can match them up with other sensors to create orientation sensors. Magnetometers are too slow and are easily messed up by magnets and accelerometers cannot measure spin along the gravitational axis (they can only measure tilt) so give a hand to the gyroscope!

When gyros are manufactured, they have some zero-offset error, much like magnetometer hard-offset error, that can make measurements difficult. It's easy to detect and remove this offset, we just take many measurements and look for the 'offset' from zero.

For example, here's a common MPU-6050. If we put it down flat on a table, and take measurements, we will see that neither X, Y or Z (red, green, blue) are at zero. Even though its not moving! That's the zero offset. In this case, its about -0.43 for X, 0.32 for Y and 0.34 for Z.
Once calibrated, you can see that there's still a little noise (there always is!) but its only 10% as much as the offset, so we'll get much better measurements. We can try to improve the noisy measurements with filtering if necessary.

Install SensorLab

Since there's dozens of different sensor manufacturers out there, and we don't want to have a ton of #ifdef's in our code to manage each kind, we'll be using Adafruit SensorLab to manage detecting the various magnetometers, accelerometers, pressure sensors... etc!

Adafruit SensorLab automatically detects a wide range of sensors, over I2C, no matter what I2C address it's on. It will return an Adafruit Unified Sensor object that we can query for events. You can't do advanced stuff like manually setting ranges or internal filters, but for many projects the basics will do just fine!

We'll be assuming you have the sensor on the main I2C port, and of course use the matching Adafruit library to verify the sensor is working and powered right before you continue!

A list of supported sensors is available here [link]

Remember, only supported sensors on I2C will be detected!
Install SensorLab

Since there are a ton of sensors, and we also use Arcada in a few examples, there's a lot of libraries to install.

No really, we have a lot of software involved here - probably 20 or so libraries total!

For that reason we really strongly recommend you use Arduino 1.8.10 or greater which handles automatic library dependency installation. Otherwise you will be frustrated...

Select the Sketch -> Include Library -> Manage Libraries...

Search for Sensor Lab and install the Adafruit library you see

Simple Gyro Calibration

If you don't want to set up a graphical interface for calibration a gyroscopic sensor, you can do a simple zero-g offset calibration using just the serial interface. The nice thing about this is it will work for any and all boards, and does not require any additional software installation!

This example can be run by any Arduino compatible, from a Arduino UNO/ATmega328 or better
Step 1 - Upload the SensorLab zero rate simplecal Example

We have a simple sketch that will read a few seconds of gyroscope data and calculate zero-rate offset for you.

Open up the Adafruit_SensorLab->calibration->gyro_zerorate_simplecal

Step 2 - Open Serial Port

Open the serial port to launch the SensorLab calibration. You should see your gyroscope detected.

You'll see a countdown to warn you that the gyro must be kept perfectly still during calibration.

Place the board on a stable flat surface and hold it down with a book or something. The board will take 5 seconds of data and print at each reading.

The first three numbers are the current gyro readings.

The middle three numbers are the average values (zero rate offsets) in radians/second.

The last numbers are the peak-to-peak values.

At the end, you'll get a summary:
In this case shown above, the screenshot indicates \( x = 0.0003, \ y = 0.01, \ z = 0.0004 \) rad/s

You may want to take a few measurements to get a couple calculations!

---

**Gyro Calibration with Jupyter**

*Jupyter Notebooks are a powerful cross-platform method for analyzing data using Python.*

You can definitely use Jupyter to plot, analyze and calibrate your sensor data. This method is the most powerful because you can do plotting and calculations. However, we assume you already have Jupyter installed (either desktop or thru Anaconda) and have some familiarity with running 'notebook' style Python!

This example runs on chips with at least 64 KB of flash, and will not fit on an UNO (Atmega328) or Leonardo (Atmega32u4) - try the simple calibration instead!

---

**Step 1 - Download Calibration Notebook**

The gyro/magnetometer notebook lives in the [SensorLab Arduino library, in the notebooks folder](https://www.adafruit.com). Look for this section in the website:

![Click to download the iPython/Jupyter Notebook](https://www.adafruit.com)

You must open this notebook within Jupyter - you cannot run it direct from github or from the command line as a text file!

Once open, your browser will look like this:
Step 2 - Upload the SensorLab imucal Example

Next we have to tell the microcontroller board to send the magnetometer (and, if there is one, accelerometer and gyroscope) data out over serial in the right format.

Open up the Adafruit_SensorLab->calibration->imucal

Select your desired board & port from the Tools menu then click Upload
Open up the serial console, you'll see SensorLab initialization and detection of whatever magnetometer is available. In this case is a LSM9DS1, but any gyroscope can be calibrated!

You'll then see a stream of data that looks like:

- **Raw**: -58, -815, 8362, 76, -121, -95, -375, -159, -24
- **Uni**: -0.07, -0.98, 10.00, 0.0832, -0.1327, -0.1046, -37.50, -15.93, -2.50

The first three numbers are accelerometer data - if you don't have an accelerometer, they will be 0.

The middle three numbers are gyroscope data, they should definitely not be zeros!

The last three numbers are magnetometer - if you don't have an magnetometer, they will be 0.

Configure the notebook

Close the serial port, and go back to Jupyter. In the first cell, find where we define the PORT and change the port to match your serial/COM port. For windows it'll be something like COM4 for Mac/Linux it'll be like /dev/cu.USBSERIAL or something
Run the first cell so the serial port is set
Then run the second cell, you should see output like this - the serial port is opened and IMU raw data is output as numbers
If you get errors or no numbers, hard-reset the board (click the reset button once) then try re-running the cell again.

Skip past the magnetometer calibration section, to the Gyro calibration cells

Skip the magnetometer cells, and find the Gyroscope offset calibration cell

Place the board down on a flat stable surface, we like to put a book on top of it to keep it flat and steady. You don’t want any shifts or movement while you run the calibrator!

You’ll see a 3 second countdown, then the serial port is opened and data collected

Put down the board and do not touch or move it!
3...2...1... COLLECTING GYRO DATA
Opened COM6

Once it’s done you’ll see a plotting of the data output. You should see something like this with red, green and blue lines. There may be some offset as expected, but the lines should wobble around a point without big spikes or increases/decreases.

Above the graph you’ll see analysis of the average value (zero-g offset) like so:
In this case, the zero-g calibration is $X = 0.044$, $Y = -0.0148$, and $Z = -0.0111$

The units are in SI units radians / second not degree/s (dps)!

You'll also see the results of removing the offset, this should be 3 noisy but nearly-zero-centered lines. The higher the quality of the gyro, the lower the jitter in the measurements, and the lower the offset.

### Comparing Gyroscopes

When making an inertial measurement unit (IMU) especially one used for orientation calculations, the quality of the gyroscope has a big impact on the jitter/speed/drift. Accelerometers are pretty good quality these days, we've figured out how to make them decades ago. And magnetometers are not as important - they don't update often and are used for orientation correction. Gyro's have the most error possibilities.

We compare two basic measurements - the zero offset and the zero noise. Zero offset is easy to correct for, and you should calibrate your gyroscope once its mounted in the final project/PCB - simply take a lot of measurements, find the offset that would bring the gyro to zero. The noise that you get when not moving can be minimized with low pass filtering (sometimes handled in-chip).

For these tests we took a random board out of stock and plotted performance. Your boards may be better..or worse! Nothing beats the datasheet for detailed information, individual test/calibration. This page is inteneded to give you a sense of what sort of performance you can expect!
ST ISM330DHC

This is an industrial 6-DoF IMU with great performance - at an expected higher price

Adafruit ISM330DHCX - 6 DoF IMU - Accelerometer and Gyroscope

Behold, the ST ISM330DHCX: an industrial quality Accelerometer+Gyroscope 6-DOF IMUs (inertial measurement unit) from ST. This IMU sensor has 6 degrees of freedom - 3...

https://www.adafruit.com/product/4502

Uncalibrated offsets are low, the one we grabbed had 0.006 rad/s (0.35 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±1 deg/s.

No-motion observed noise was an incredibly low ±0.002 rad/s (±0.06 deg/sec) when running at 104 Hz and no filters on. Check the datasheet for more details!

LSM6DSOX

This is an high quality 6-DoF IMU with great performance - at an expected higher price than the LSM6DS33. This is the 'commercial usage' version of the ISM330DHC
Adafruit LSM6DSOX 6 DoF Accelerometer and Gyroscope
Behold, the ST LSM6DSOX: The latest in a long line of quality Accelerometer+Gyroscope 6-DOF IMUs from ST. This IMU sensor has 6 degrees of freedom - 3 degrees each of linear...
https://www.adafruit.com/product/4438

Uncalibrated offsets are low, the one we grabbed had 0.007 rad/s (0.42 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±1 deg/s.

No-motion observed noise was an incredibly low ±0.003 rad/s (±0.17 deg/sec) when running at 104 Hz and no filters on. Check the datasheet for more details!

LSM6DS33

This basic 6-DoF IMU is inexpensive and works well for basic projects. We have it both in 6-DoF and paired with a magnetometer for 9-DoF,
Adafruit LSM6DS33 6-DoF Accel + Gyro IMU
Add motion and orientation sensing to your Arduino project with this affordable 6 Degree of Freedom (6-DoF) sensor with sensors from ST. The board includes an... https://www.adafruit.com/product/4480

Adafruit LSM6DS33 + LIS3MDL - 9 DoF IMU with Accel / Gyro / Mag
Discontinued - you can grab Adafruit LSM6DS3 + LIS3MDL - Precision 9 DoF IMU https://www.adafruit.com/product/4485

Uncalibrated offsets are fair, the one we grabbed had 0.034 rad/s (2 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±10 deg/s!

No-motion observed noise was ±0.015 rad/s (±0.85 deg/sec) when running at 104 Hz and no filters on. Check the datasheet for more details!
LSM6DS3TR-C

The ST LSM6DS3TR-C is a great entry-level 6-DoF IMU accelerometer + gyro and is very similar to the now-discontinued LSM6DS33. We have it both in 6-DoF and paired with a magnetometer for 9-DoF

Adafruit LSM6DS3TR-C 6-DoF Accel + Gyro IMU
Add motion and orientation sensing to your Arduino project with this affordable 6 Degree of Freedom (6-DoF) sensor with sensors from ST. The board includes an... https://www.adafruit.com/product/4503

Adafruit LSM6DS3TR-C + LIS3MDL - Precision 9 DoF IMU
Add high-quality motion, direction, and orientation sensing to your Arduino project with this all-in-one 9 Degree of Freedom (9-DoF) sensor with sensors from ST. This little... https://www.adafruit.com/product/5543

The one used here had 0.03 rad/s (1.7 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±3 deg/s.

No-motion observed noise was ±0.0015 rad/s (±0.085 deg/sec) when running at 104 Hz and no filters on. Check the datasheet for more details! ()
LSM9DS1

This popular sensor is a 9-DoF all-in-one with a LIS3MDL for the magnetometer, and a 'LSM6DS-like-but-not-the-same-register-map-at-all' accelerometer/gyro IMU

Adafruit 9-DOF Accel/Mag/Gyro+Temp Breakout Board - LSM9DS1
Add motion, direction and orientation sensing to your Arduino project with this all-in-one 9-DOF sensor. Inside the chip are three sensors, one is a classic 3-axis accelerometer, which...

https://www.adafruit.com/product/3387

Uncalibrated offsets are pretty good, the one we grabbed had 0.02 rad/s (1.2 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±30 deg/s!

However, we noticed spikes of gyro data well outside the expected range. When those spikes are ignore, the no-motion observed noise was + - 0.007 rad/s (±0.4 deg/sec) at 1 KHz with the 408 Hz bandwidth filter on. Check the datasheet for more details!
MPU-6050

This is a fairly old sensor, but very popular and low cost! The quality of the sensor is pretty good considering how old/low cost it is
Adafruit MPU-6050 6-DoF Accel and Gyro Sensor - STEMMA QT Qwiic

I mew, you mew we all mew for IMU! The MPU-6050 is what we call "an oldie but goodie" - this popular triple-axis accelerometer plus gyro combo has been a...

https://www.adafruit.com/product/3886

Uncalibrated offsets are fair, the one we grabbed had 0.04 rad/s (2.3 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±20 deg/s!

No-motion observed noise was +/- 0.05 rad/s (±0.29 deg/sec) with the 260 Hz bandwidth filter on. Check the datasheet for more details!

NXP FXAS21002

This 3-axis gyroscope sensor is no longer available but we use it in our NXP 9-DoF (and will continue selling it until we can't get the chip anymore) At the time of release it was pretty good, since then better gyros have come out, but it's still not too bad
Adafruit Precision NXP 9-DOF Breakout Board

The NXP Precision 9DoF breakout combines two of the best motion sensors we've tested here at Adafruit: The FXOS8700 3-Axis accelerometer and magnetometer, and the...

https://www.adafruit.com/product/3463

Uncalibrated offsets are pretty good - the one we grabbed had 0.01 rad/s (0.57 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical post-mount ±50 LSB (NOT deg/s) - at 250 deg/s rate, that translates to ±0.4 deg/s

No-motion observed noise was +- 0.01 rad/s (±0.55 deg/sec) at 100 Hz output. Check the datasheet for more details!

![Uncalibrated Gyro](image1.png)

![Calibrated Gyro](image2.png)
ICM-20649

Adafruit ICM-20649 Wide Range ±30g ±4000dps 6-DoF IMU
Most accelerometers have a similar range of measurements that they can make: often around 2G - 16G. Similarly, most gyros can measure in the range of 250 degree/s to 2000 degrees/s....
https://www.adafruit.com/product/4464

Uncalibrated offsets are not bad, the one we grabbed had 0.023 rad/s (1.3 deg/s) max offset. Datasheet's Angular rate zero-rate level is typical ±5 deg/s

No-motion observed noise was +- 0.015 rad/s (±0.86 deg/sec) at 1.1KHz. Check the datasheet for more details!