An accelerometer is a device that measures proper acceleration. Proper acceleration, being the acceleration (or rate of change of velocity) of a body in its own instantaneous rest frame,[2] is not the same as coordinate acceleration, being the acceleration in a fixed coordinate system. For example, an accelerometer at rest on the surface of the Earth will measure an acceleration due to Earth's gravity, straight upwards (by definition) of $g \approx 9.81$ m/s2. By contrast, accelerometers in free fall (falling toward the center of the Earth at a rate of about 9.81 m/s2) will measure zero.

Accelerometers have multiple applications in industry and science. Highly sensitive accelerometers are components of inertial navigation systems for aircraft and missiles. Accelerometers are used to detect and monitor vibration in rotating machinery. Accelerometers are used in tablet computers and digital cameras so that images on screens are always displayed upright. Accelerometers are used in drones for flight stabilisation. Coordinated accelerometers can be used to measure differences in proper acceleration, particularly gravity, over their separation in space; i.e., gradient of the gravitational field. This gravity gradiometry is useful because absolute gravity is a weak effect and depends on local density of the Earth which is quite variable.

Single- and multi-axis models of accelerometer are available to detect magnitude and direction of the proper acceleration, as a vector quantity, and can be used to sense orientation (because direction of weight changes), coordinate acceleration, vibration, shock, and falling in a resistive medium (a case where the proper acceleration changes, since it starts at zero, then increases). Micromachined accelerometers are increasingly present in portable electronic devices and video game controllers, to detect the position of the device or provide for game input.

How they work

An accelerator looks like a simple circuit for some larger electronic device. Despite its humble appearance, the accelerometer consists of many different parts and works in many ways, two of which are the piezoelectric effect and the capacitance sensor. The piezoelectric effect is the most common form of accelerometer and uses microscopic crystal structures that become stressed due to accelerative forces. These crystals create a voltage from the stress, and the accelerometer interprets the voltage to determine velocity and orientation.

The capacitance accelerometer senses changes in capacitance between microstructures located next to the device. If an accelerative force moves one of these structures, the capacitance will change and the accelerometer will translate that capacitance to voltage for interpretation.

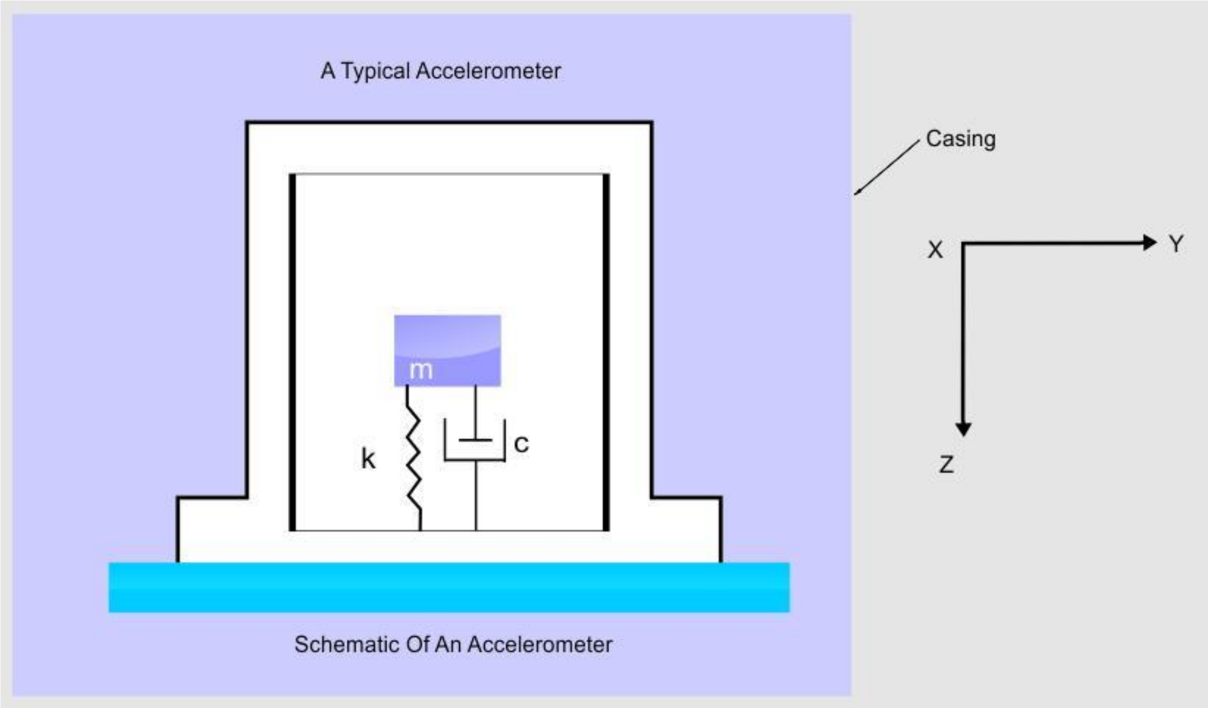
Accelerometers are made up of many different components, and can be purchased as a separate device. Analog and digital displays are available, though for most technology devices, these components are integrated into the main technology and accessed using the governing software or operating system.

Typical accelerometers are made up of multiple axes, two to determine most two-dimensional movement with the option of a third for 3D positioning. Most smartphones typically make use of three-axis models, whereas cars simply use only a two-axis to determine the moment of impact. The sensitivity of these devices is quite high as they're intended to measure even very minute shifts in acceleration. The more sensitive the accelerometer, the more easily it can measure acceleration.

Accelerometers, while actively used in many electronics in today's world, are also available for use in custom projects. Whether you're an engineer or tech geek, the accelerometer plays a very active role in a wide range of functionalities. In many cases you may not notice the presence of this simple sensor, but odds are you may already be using a device with it.

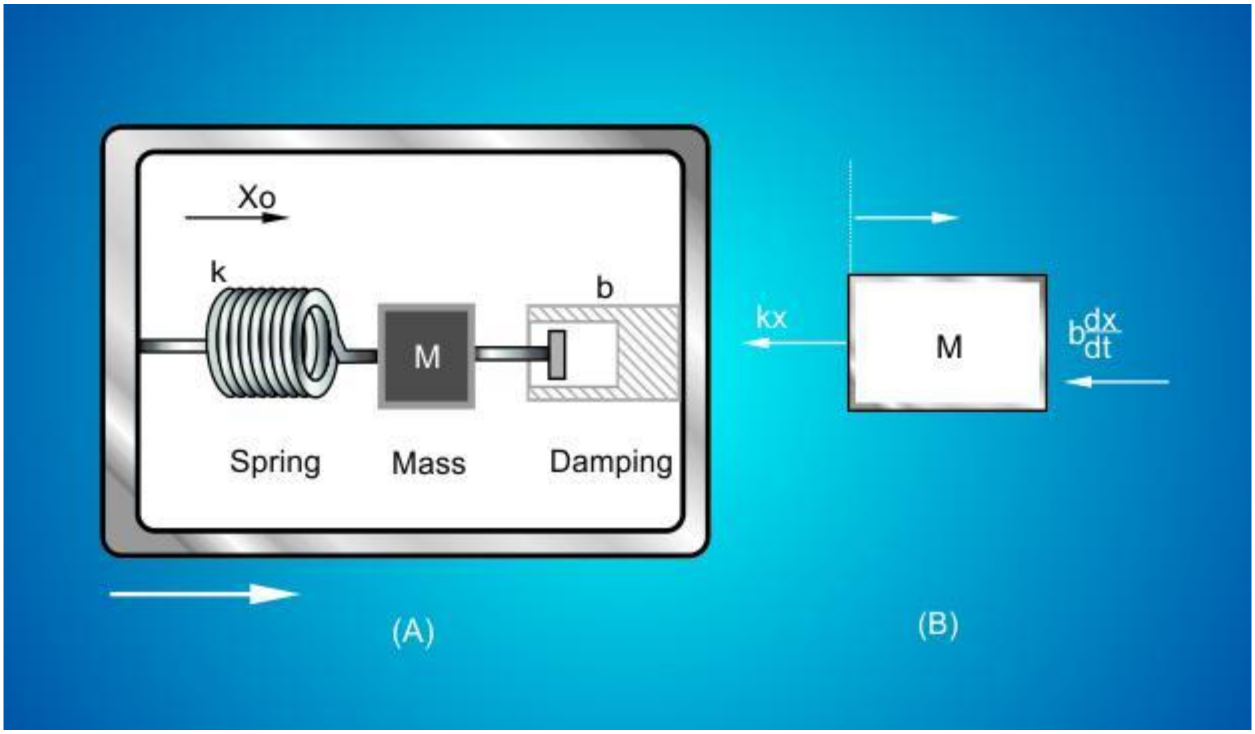
The term '**Accelerometers**' refer to the transducers which comprises of mechanical sensing element and a mechanism which converts the mechanical motion into an electrical output.

Theory behind working of accelerometers can be understood from the mechanical model of accelerometer, using Newtonian mechanics. The sensing element essentially is a proof mass (also known as seismic mass). The proof mass is attached to spring which in turn is connected to its casing. In addition, a dashpot is also included in a system to provide desirable damping effect; otherwise system may oscillate at its natural frequency. The dashpot is attached (in parallel or in series) between the mass and the casing. The unit is rigidly mounted on the body whose acceleration is of interest.



When the system is subjected to linear acceleration, a force (= mass * acceleration) acts on the proof-mass. This causes it to deflect; the deflection is sensed by a suitable means and is converted into an equivalent electrical signal.

When force is applied on the body, proof mass moves. Its movement is countered by spring and damper.



Therefore, if m = proof mass of the body
x = relative movement of the proof-mass with respect to the frame
c = damping coefficient
k = spring stiffness

then

Summation of all forces on Proof mass = 0

$$m\,a + Fd + Fs = 0$$

$$m\,a = -Fd - Fs$$

$$m\,a = -c\,\dot{x} - k\,x$$

$$a = -\left(\frac{c}{m}\right)\dot{x} - \left(\frac{k}{m}\right)x$$

Thus, with the knowledge of damping coefficient(c), spring stiffness (k), and proof mass (m), for a useful acceleration sensor, it is sufficient to provide a component that can move relative to sensors housing and a means to sense the movement.

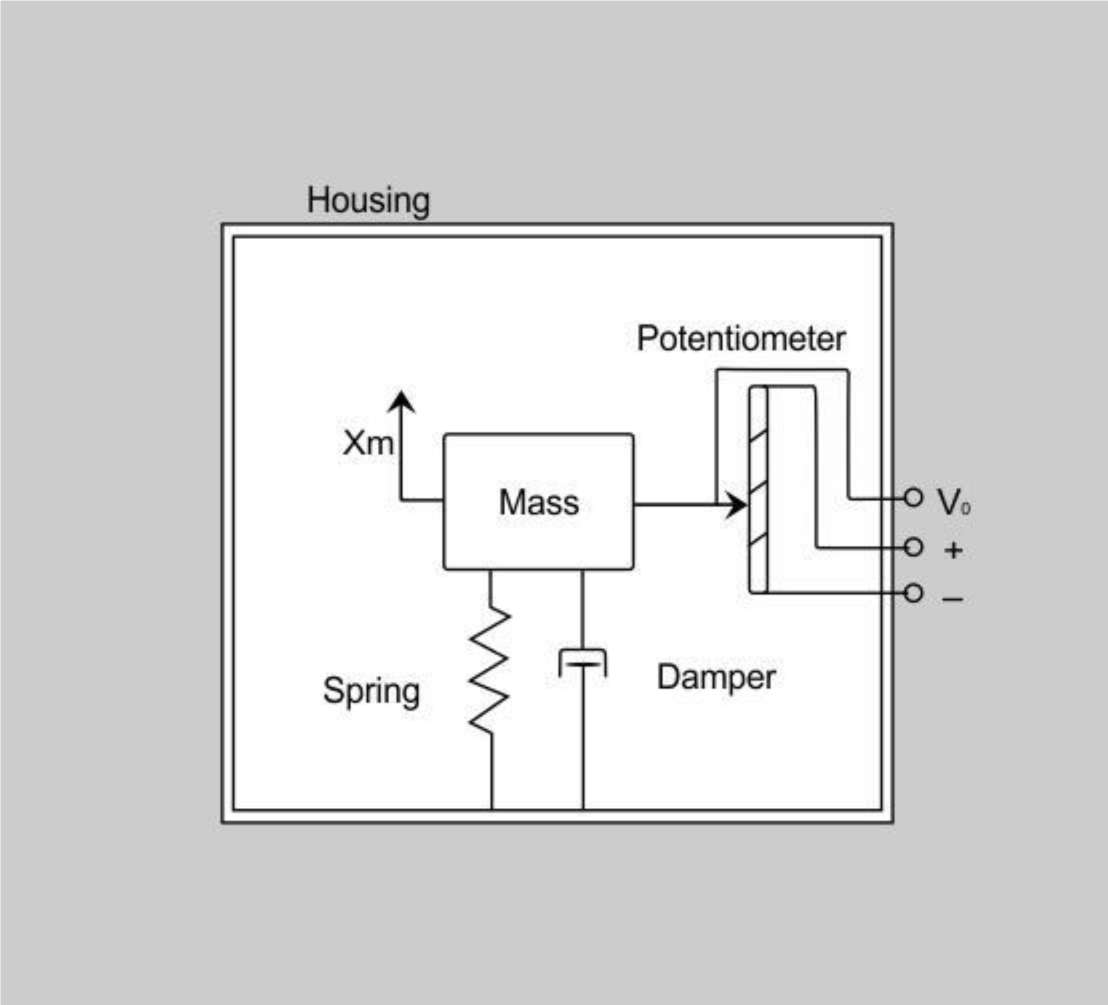
Displacement and acceleration are related by fundamental scaling law. A higher resonant frequency implies less displacement or low sensitivity.

TYPES OF ACCELEROMETERS

As movement of the proof mass is sufficient for an accelerometer, **accelerometers are designed using various sensing principles**.

- **Potentiometric**

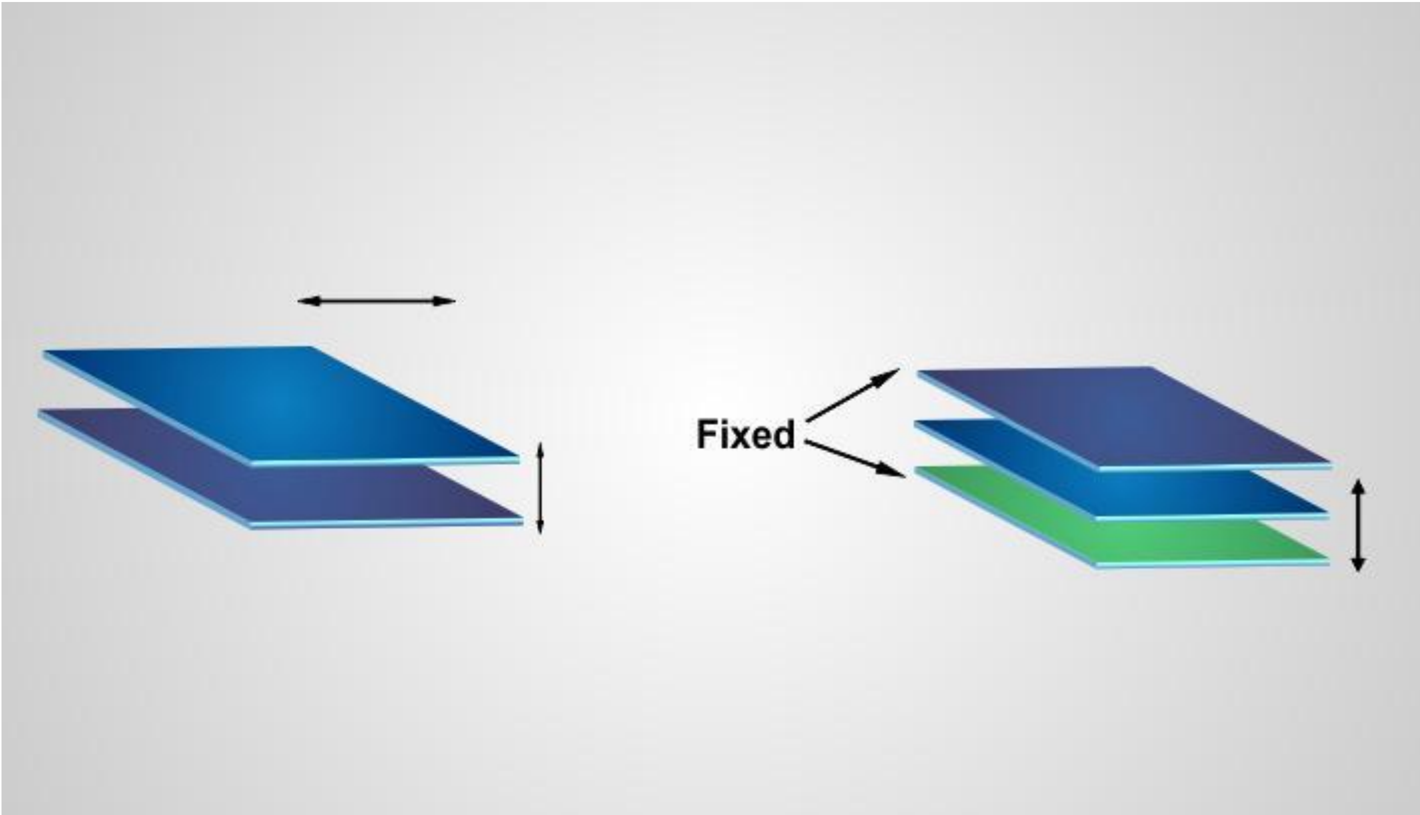
One of the simplest accelerometer type - it measures motion of the proof mass motion by attaching the spring mass to the wiper arm of a potentiometer. Thus position of the mass and thereby, changing acceleration is translated to changing resistance.



The natural frequency of these devices is generally less than 30 Hz, limiting their application to low frequency vibration measurements. Dynamic range is also limited. But they can measure down to 0 Hz (DC response).

- **Capacitive accelerometers**

Capacitive accelerometers sense a change in electrical capacitance, with respect to acceleration. Single capacitor or differential capacitors can be used; differential ones being more common



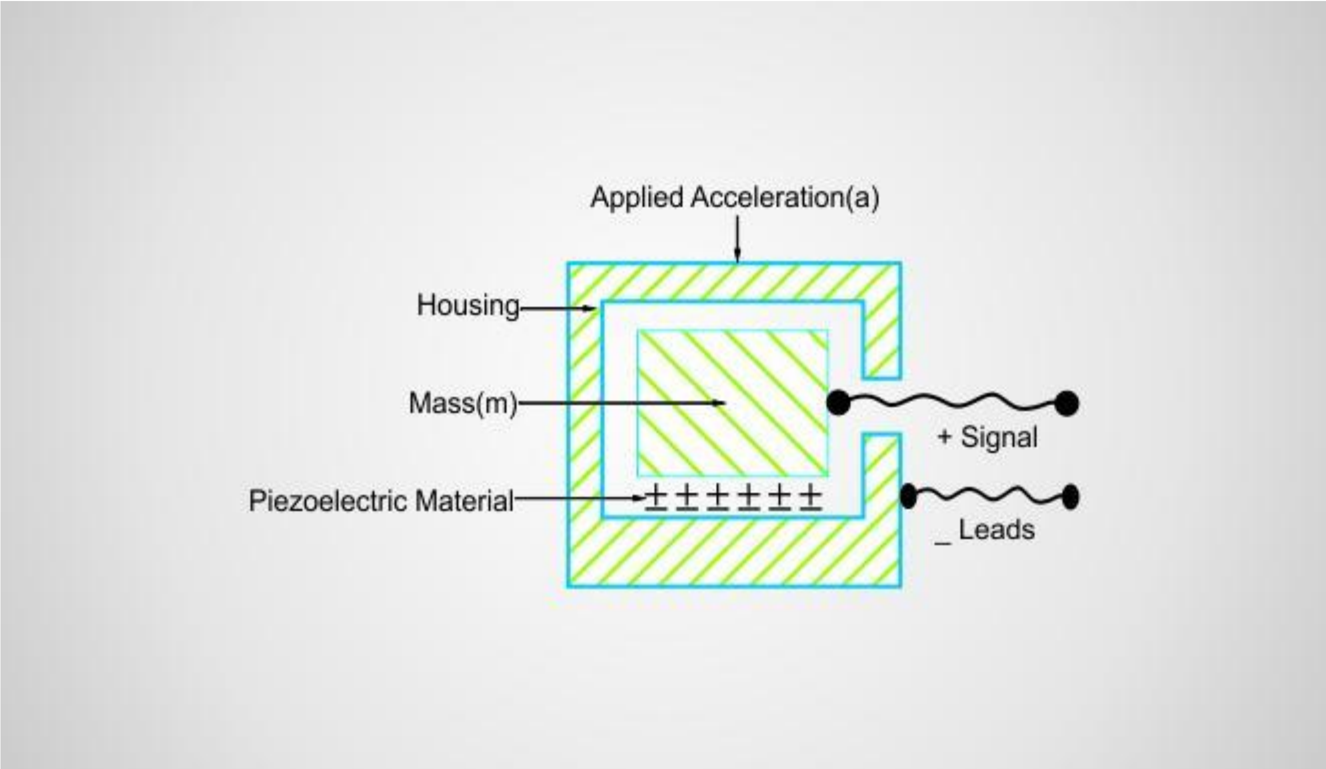
In these accelerometers, a diaphragm acting as a mass moves in the presence of acceleration. The diaphragm is sandwiched between the two fixed plates creating two capacitors; each with an individual fixed plate and each sharing the diaphragm as a movable plate. Movement of the diaphragm causes a capacitance shift by altering the distance between two parallel plates, the diaphragm itself being one of the plates.
The two capacitors form the two arms of the bridge; the output of the bridge varies with the acceleration.

Capacitive sensing is most commonly used in MEMS accelerometers. Like potentiometric accelerometers, capacitive accelerometers have true DC response but limited frequency range and limited dynamic range.

- **Piezoelectric accelerometers**

Piezoelectric accelerometers employ piezoelectric effect. When piezoelectric materials are stressed, they are deformed and an electric charge is generated on the piezoelectric materials.

In piezoelectric accelerometers, piezoelectric material is used as an active element. One side of the piezoelectric material is connected to rigid base. Seismic or proof mass is attached to the other side. When force (generated due to acceleration) is applied, piezoelectric material deforms to generate the charge. This charge is proportional to the applied force or in other words, proportional to acceleration (as mass is constant). The charge is converted to voltage using charge amplifiers and associated signal conditioning circuit.

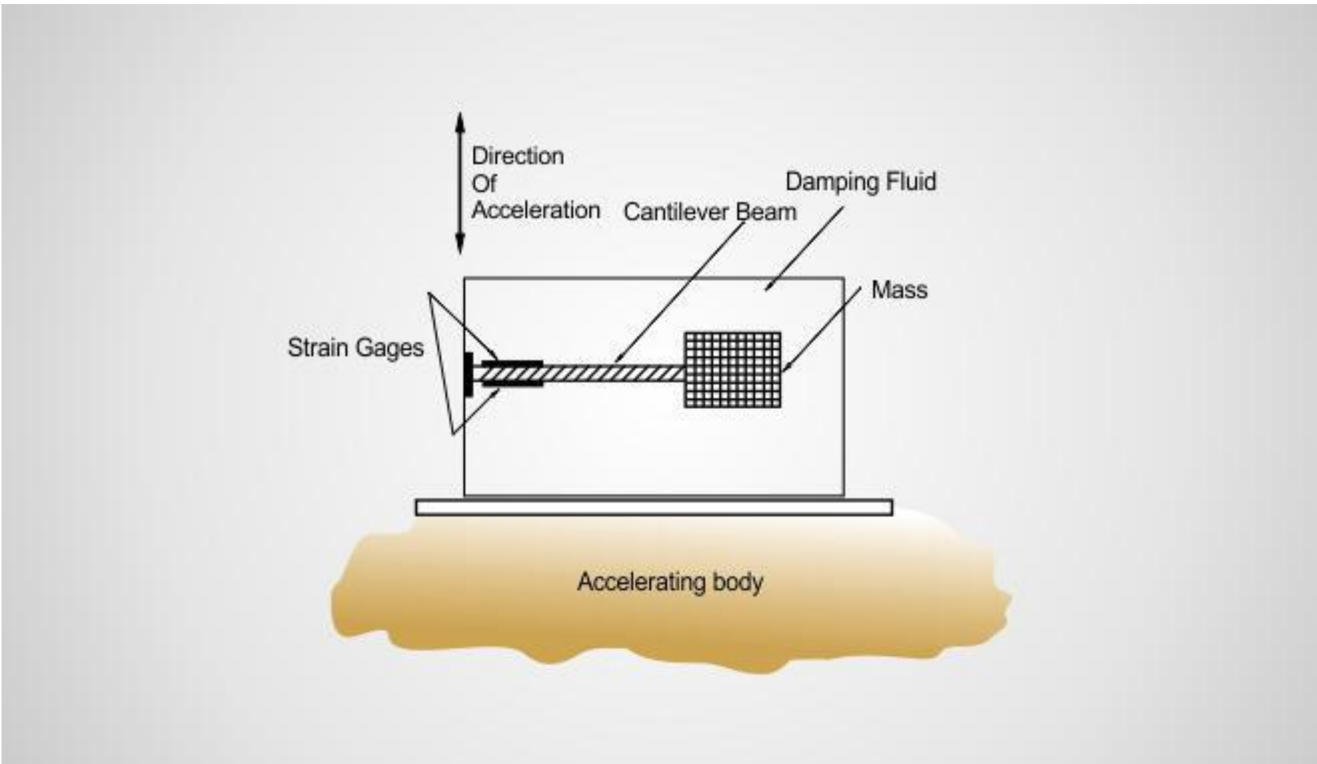


Compared to other type of accelerometers, piezoelectric accelerometers offer unique advantages –
Wide dynamic range
Excellent linearity
Wide frequency range
No wear and tear due to absence of moving parts
No external power requirement

However, alternating acceleration only can be measured with piezoelectric accelerometers. These accelerometers are not capable of measuring DC response.

- **Piezo-resistive accelerometers**

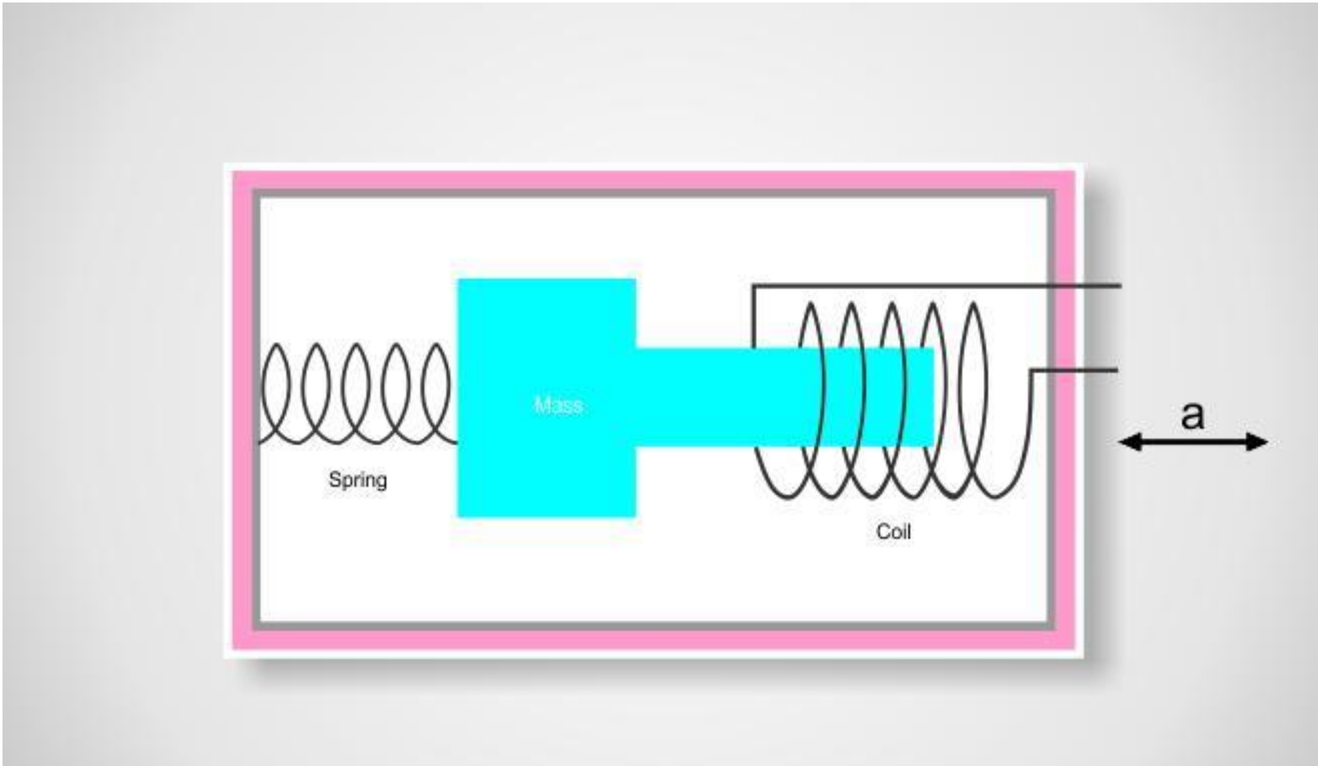
Piezo-resistive accelerometers use piezo-resistive materials, i.e., strain gauges. On application of the force (due to acceleration), resistance of these strain gages changes. The change in resistance is monitored to measure the acceleration.



Piezo-resistive elements are typically used in micro-machined structures. They have true DC response. They can be designed to measure upto ±1000 g.

- **Variable inductance accelerometers**

Using the concept very similar to the one used in LVDTs, variable inductance accelerometers can be designed. In these accelerometers, proof mass is made of ferromagnetic materials. The proof mass is designed in the form of core which can move in or out of the coil.

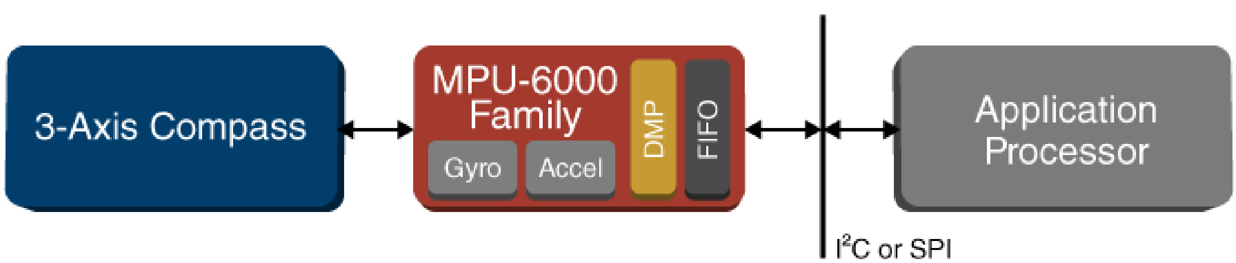


When the body is accelerated, the proof mass moves. In other words, portion of the core inside the coil changes and so the coil impedance. Thus, the coil impedance is a function of the applied acceleration.

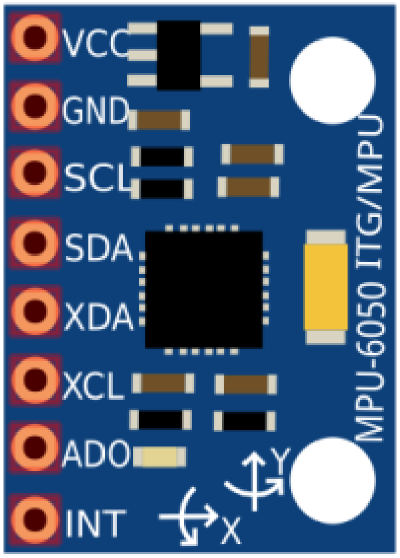
Using a accelerometer with Microcontroller(Here I will be using Arduino for the explanation purpose)

I am trying to explain the use and interfacing of the gyroscope with the Arduino by using a MPU6050 IMU sensor which houses both a gyroscope as well as an accelerometer on the same board

The MPU6050 uses the I2C communication protocol to send and receive the data from the microcontroller which we are using



The MPU-6050 is a serious little piece of motion processing tech! By combining a MEMS 3-axis gyroscope and a 3-axis accelerometer on the same silicon die together with an onboard Digital Motion Processor™ (DMP™) capable of processing complex 9-axis MotionFusion algorithms, the MPU-6050 does away with the cross-axis alignment problems that can creep up on discrete parts.



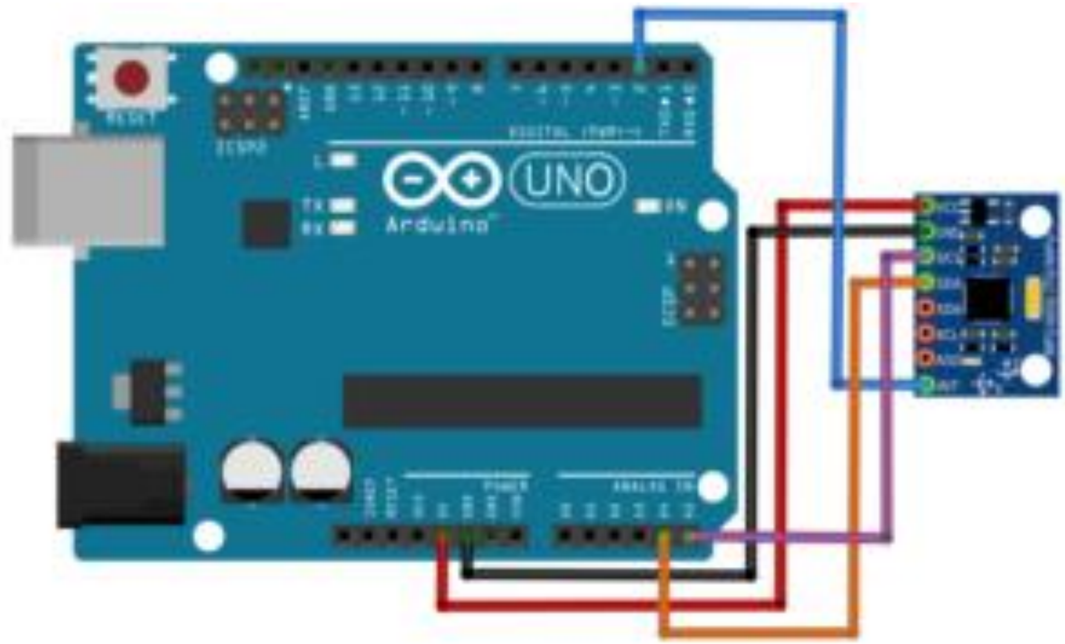
Arduino MPU 6050 Pin out

How Does it Work?

IMU sensors usually consist of two or more parts. Listing them by priority, they are the accelerometer, gyroscope, magnetometer, and altimeter. The MPU 6050 is a 6 DOF (Degrees of Freedom) or a six-axis IMU sensor, which means that it gives six values as output. Three values from the accelerometer and three from the gyroscope. The MPU 6050 is a sensor based on MEMS (Micro Electro Mechanical Systems) technology. Both the accelerometer and the gyroscope are embedded inside a single chip. This chip uses I2C (Inter-Integrated Circuit) protocol for communication.

Interfacing the Arduino MPU 6050

The MPU 6050 communicates with the Arduino through the I2C protocol. The MPU 6050 is connected to Arduino as shown in the following diagram. If your MPU 6050 module has a 5V pin, then you can connect it to your Arduino's 5V pin. If not, you will have to connect it to the 3.3V pin. Next, the GND of the Arduino is connected to the GND of the MPU 6050.



Arduino MPU 6050 connections

The program we will be running here, also takes advantage of the Arduino's interrupt pin. Connect your Arduino's digital pin 2 (interrupt pin 0) to the pin labeled as INT on the MPU 6050. Next, we need to set up the I2C lines. To do this, connect the pin labeled SDA on the MPU 6050 to the Arduino's analog pin 4 (SDA) and the pin labeled as SCL on the MPU 6050 to the Arduino's analog pin 5 (SCL).

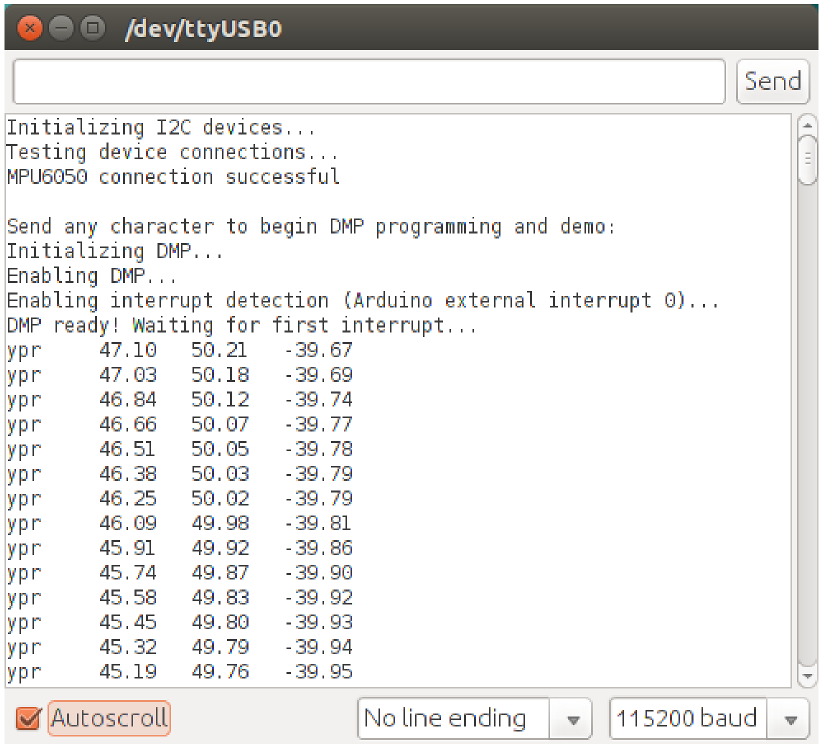
Uploading the Code and Testing the Arduino MPU 6050

To test the Arduino MPU 6050, first download the Arduino library for MPU 6050, developed by Jeff Rowberg. You can find the library here. Next, you have to unzip/extract this library and take the folder named "MPU6050" and paste it inside the Arduino's "library" folder. To do this, go to the location where you have installed Arduino (Arduino –> libraries) and paste it inside the libraries folder. You might also have to do the same thing to install the I2Cdev library if you don't already have it for your Arduino. Do the same procedure as above to install it, you can find the file here: I2Cdev library.

If you have done this correctly, when you open the Arduino IDE, you can see "MPU6050" in File –> Examples. Next, open the example program from: File –> Examples –> MPU6050 –> Examples –> MPU6050_DMP6.

Arduino MPU 6050 DMP code

Next, you have to upload this code to your Arduino. After uploading the code, open up the serial monitor and set the baud rate as 115200. Next, check if you see stuff like "Initializing I2C devices..." on the serial monitor. If you don't, just press the reset button. Now, you'll see a line saying "Send any character to begin DMP programming and demo." Just type in any character on the serial monitor and send it and you should start seeing the yaw, pitch, and roll values coming in from the MPU 6050. Like so:



Arduino MPU 6050 Serial Monitor

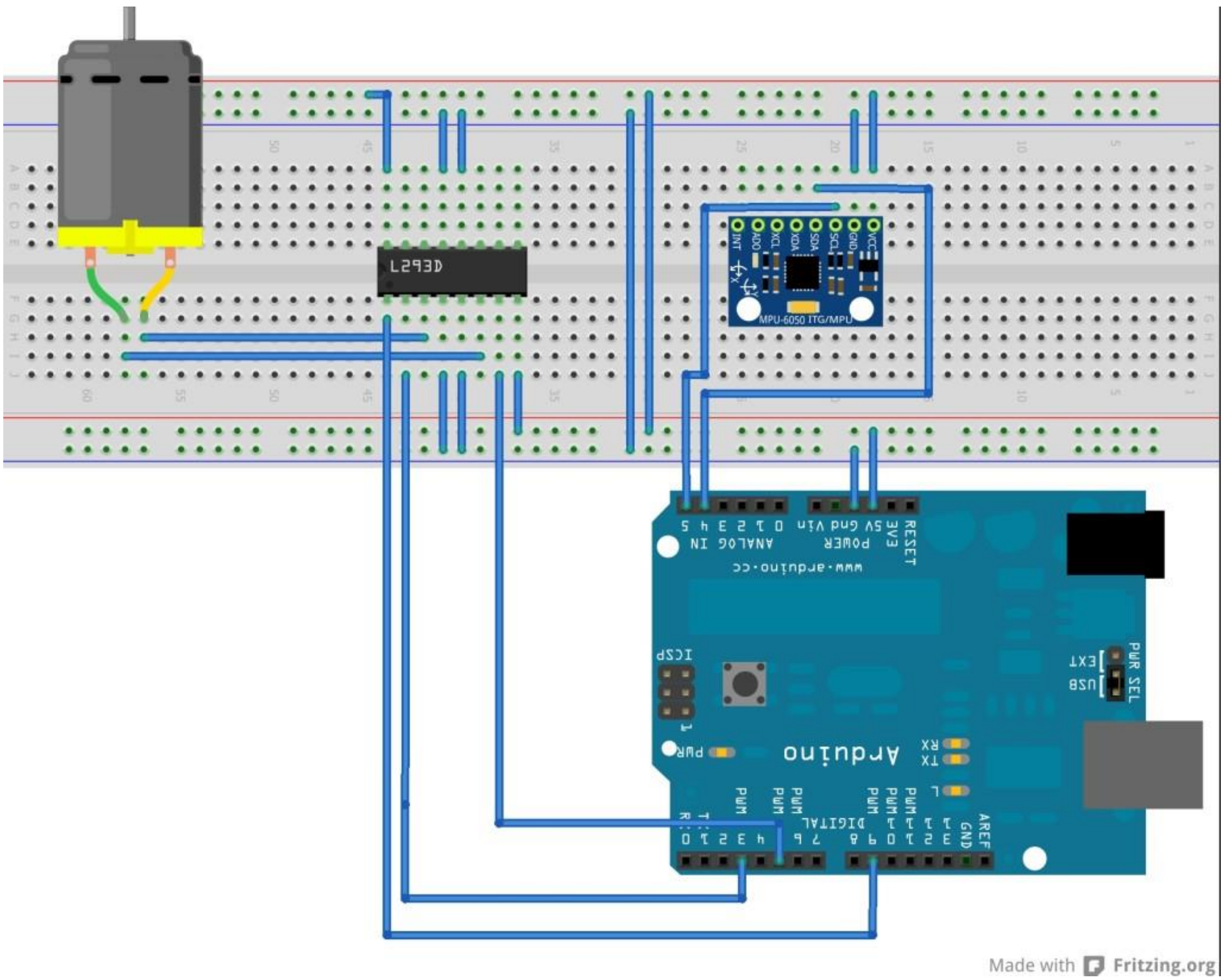
DMP stands for Digital Motion Processing. The MPU 6050 has a built-in motion processor. It processes the values from the accelerometer and gyroscope to give us accurate 3D values.

Also, you will need to wait about 10 secs before you get accurate values from the Arduino MPU 6050. After which, the values will begin to stabilize.

We can use the accelerometer and the gyroscope readings from MPU6050 IMU and control the motor speed, The weight balance, Thrust etc for different cases to maintain the Balance and stability of the system etc or to implement the machine learning algorithms or the PID Controller mechanism in our bots.

We can see the use by designing small prototype of a system in which the motor speed is controlled by the MPU6050 IPU'S Reading.

To make this prototype I am using:

1. Arduino UNO
2. GY-521 (MPU-6050)
3. L29dD Driver IC
4. DC Motor
5. Breadboard
6. Jumper wire
7. 

```
#include <Wire.h>
#include<I2Cdev.h>
#include<MPU6050.h>

MPU6050 mpu;

int16_t ax, ay, az;
int16_t gx, gy, gz;

#define pin1 3
#define pin2 5

void setup(){
 Serial.begin(9600);
 Serial.println("Initialize MPU");
 mpu.initialize();
 //Serial.println(mpu.testConnection() ? "Connected" : "Connection failed");
pinMode(pin1,OUTPUT);
 pinMode(pin2,OUTPUT);

}

void loop(){
 mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);

 ax = map(ax, -17000, 17000, -1500, 1500);

 //Serial.println(ax);
 if(ax > 0){
  if(ax<255){
  Serial.println(ax);
  analogWrite(pin2,ax);
 }
 else{
  Serial.println("+255");
  analogWrite(pin2,255);
 }

 }
 if(ax<0){
  if(ax>-255){
  Serial.println(ax);
  analogWrite(pin1, ax-ax-ax);
 }
 else{
  Serial.println("-255");
  analogWrite(pin1, 255);
 }
 }
 delay(1000);
}
```

The following code(algorithm) can be extended to make the robot self balancing and smart so that our bot can be stable and we can avoid some errors which might arise at the last moment in the Robocon competition by using the Feedback system and the PID control for the best outputs.