# Accelerometer using arduino

Introduction

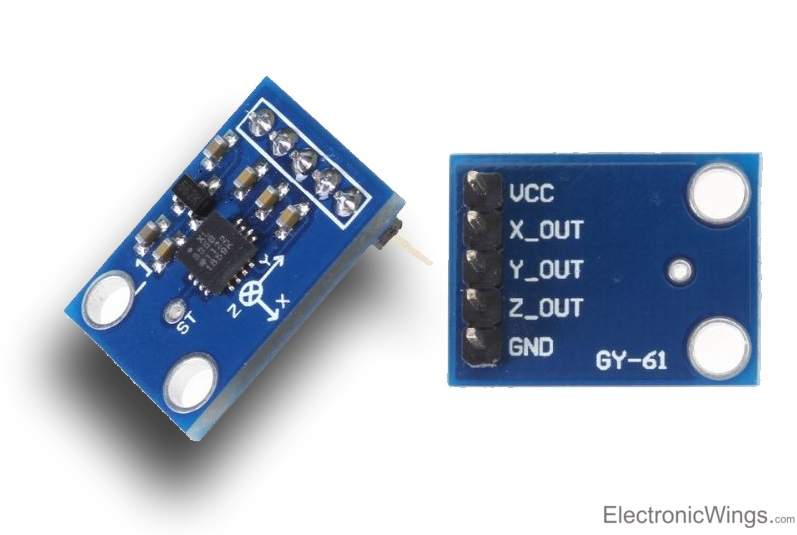
Accelerometer is an electromechanical device that measures the force of acceleration due to gravity in g unit.

It can be used in applications requiring tilt sensing.

The ADXL335 measures acceleration along X, Y and Z axes and gives analog voltage output proportional to the acceleration along these 3 axes.

Microcontrollers can process these voltages by converting them to digital signals using ADC.

The Accelerometer module consists of **ADXL335** ic along with a 3.3v voltage regulator and some capacitors and resistors, so it helps the module measure the acceleration in the range of ±3g and output proportional analog values of range 0 t0 1023.



Components

* Arduino Uno
* Adxl335 Accelerometer Module
* LEDs(Red, Green, Blue, and Yellow)
* 220-ohm Resistor
* Breadboard
* Jumper Wires
* USB cable to connect Arduino Uno with Computer

Application

* Accelerometers such as in Drones (to maintain stability in the air), smartphones (to adjust the orientation of the screen from portrait to landscape),
* Game controllers and pads,
* Joysticks (to track the movement and direction),
* in Disc Drives( to prevent data loss when free fall is detected),
* Sports devices and trackers

Objective

**During this activity ,you will help students to achieve following objectives**

**1.** Understanding the principle and operation of Adxl335 Accelerometer Module

2. Design algorithm and flowchart for Adxl335 Accelerometer Module

3. Programming Adxl335 Accelerometer Module using Arduino uno

4. Interfacing Adxl335 Accelerometer Module with Arduino uno

Programming steps

1. Initialise variable m,n for accelerometer x,y position
2. Define input and output port
3. Read analog input values
4. If analog input (n>=375) ,then make LED1 ON
5. IF analog input (n<=320) , then make LED2 ON
6. if analog input (m>=375), then make LED3 ON

else analog input (m<=315), then make LED4 ON

Program

int m=0, n=0;

void setup()

{

pinMode(A0, INPUT);

pinMode(A1, INPUT);

pinMode(4,OUTPUT);

pinMode(5,OUTPUT);

pinMode(6,OUTPUT);

pinMode(7,OUTPUT);

Serial.begin(9600);

}

void loop()

{

m = analogRead(A0);

n = analogRead(A1);

Serial.println(m);

delay(100);

Serial.println(n);

delay(100);

if(n>=375)

{

digitalWrite(4,HIGH);

Serial.println("F");

}

else if(n<=320)

{

digitalWrite(5,HIGH);

Serial.println("B");

}

else if(m>=375)

{

digitalWrite(6,HIGH);

Serial.println("R");

}

else if(m<=315)

{

digitalWrite(7,HIGH);

Serial.println("L");

}

else

{

digitalWrite(4, LOW);

digitalWrite(5, LOW);

digitalWrite(6, LOW);

digitalWrite(7, LOW);

Serial.println("N");

}

}

Hardware

Instruction

* VCC of Module to 5V pin of Arduino UNO.
* GND of Module to GND pin of Arduino UNO.
* X\* of Module to A0 of Arduino UNO.
* Y\* of Module to A1 of Arduino UNO.
* Positive’+’ ends of LEDs to pins 4,5,6,7 of Arduino UNO Respectively.
* Negative ‘-‘ ends of LEDs to GND pin of Arduino UNO using 220-ohm Resistor.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Arduino** | **LED B** | **LED G** | **LED R** | **LED Y** | **220 Ohm Resistor** |
| D7 Pin | Anode Pin |  |  |  |  |
| D6 Pin |  | Anode Pin |  |  |  |
| D5 Pin |  |  | Anode Pin |  |  |
| D5 Pin |  |  |  | Anode Pin |  |
| GND |  |  |  |  | Terminal 1 |
|  | Cathode Pin | Cathode Pin | Cathode Pin | Cathode Pin | Terminal 2 |

