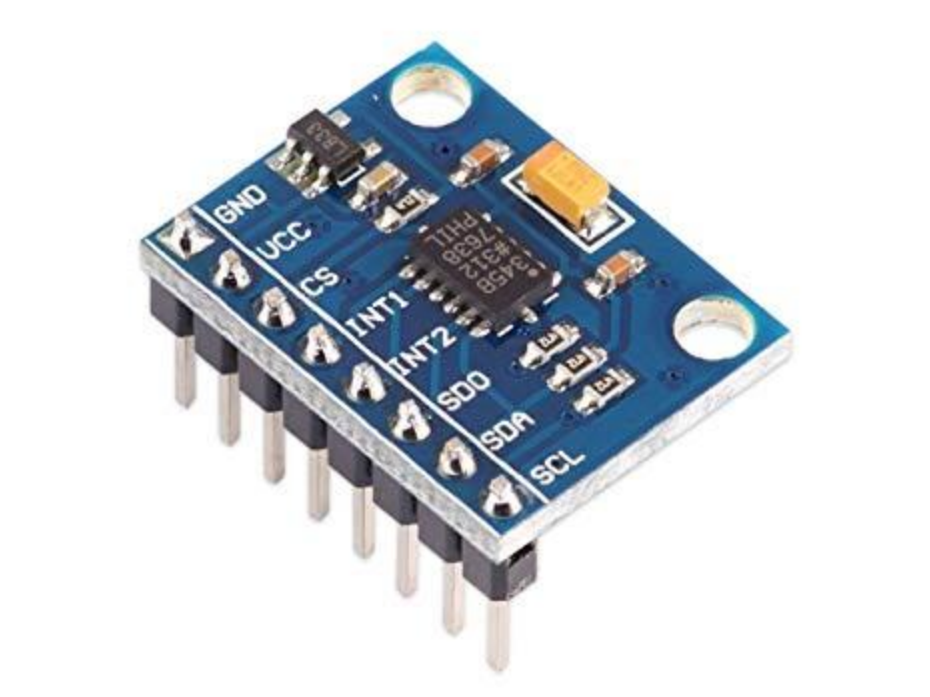
Interfacing of ADXL345 Accelerometer



Accelerometer(ADXL345)

- The ADXL345 is a small, thin, ultralow power, 3-axis accelerometer with high resolution (13-bit) measurement at up to ±16 g.
- Digital output data is formatted as 16-bit two's complement and is accessible through either a SPI (3- or 4-wire) or I2C digital interface.
- The ADXL345 is well suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock.

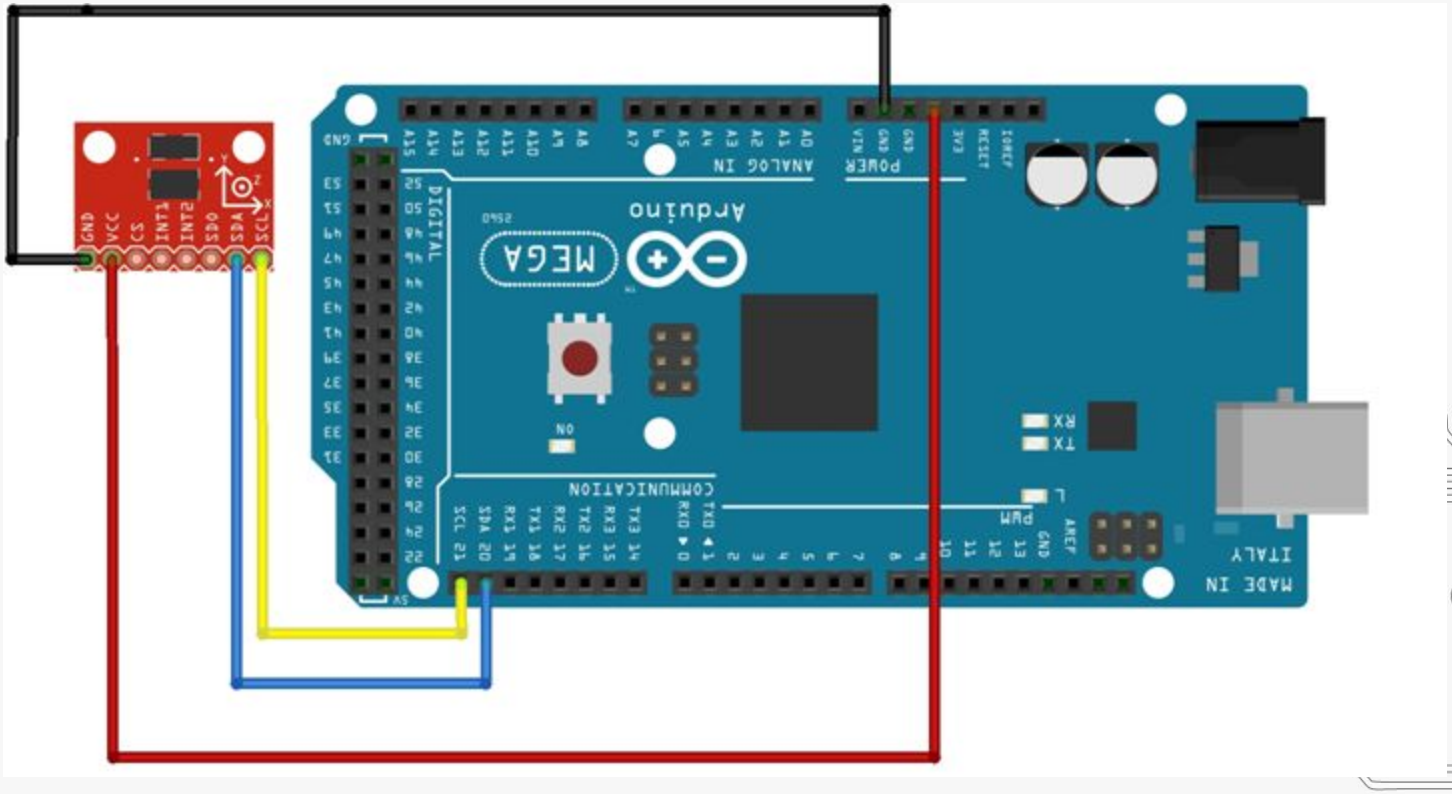
Working of Accelerometer

- This is a 3-axis accelerometer which can measure both static and dynamic forces of acceleration.
- The unit of measurement for acceleration is meter per second squared (m/s^2). However, accelerometer sensors usually express the measurements in “g” or gravity. One “g” is the value of the earth gravitational force which is equal to 9.8 meters per second squared.
- So, if we have an accelerometer positioned flat, with its Z-axis pointing upwards, opposite to the gravitational force, the Z-axis output of the sensor will be 1g. On the other hand, the X and Y outputs will be zero, because the gravitational force is perpendicular to these axes and doesn’t affect them at all.

Components required

- Arduino mega
- ADXL345 Accelerometer
- Jumper wires

Connection Diagram



Connections

1. Connect SDA pin of ADXL345 with 20 pin of Arduino.
2. Connect SCL pin of ADXL345 with 21 pin of Arduino.
3. Connect its Vcc with Arduino (+5V).
4. Connect its GND with Arduino GND.

Code

Interfacing_of_accelerometer | Arduino 1.8.19

File Edit Sketch Tools Help

Interfacing_of_accelerometer

```
/*****************************************************************************/

#include <Wire.h>
#include <ADXL345.h>


ADXL345 adxl; //variable adxl is an instance of the ADXL345 library

void setup(){
  Serial.begin(9600);
  adxl.powerOn();

  //set activity/ inactivity thresholds (0-255)
  adxl.setActivityThreshold(75); //62.5mg per increment
  adxl.setInactivityThreshold(75); //62.5mg per increment
  adxl.setTimeInactivity(10); // how many seconds of no activity is inactive?

  //look of activity movement on this axes - 1 == on; 0 == off
  adxl.setActivityX(1);
  adxl.setActivityY(1);
  adxl.setActivityZ(1);

  //look of inactivity movement on this axes - 1 == on; 0 == off
  adxl.setInactivityX(1);
```

Code

Interfacing_of_accelerometer | Arduino 1.8.19

File Edit Sketch Tools Help

Interfacing_of_accelerometer

```
//look of inactivity movement on this axes - 1 == on; 0 == off
adxl.setInactivityX(1);
adxl.setInactivityY(1);
adxl.setInactivityZ(1);

//look of tap movement on this axes - 1 == on; 0 == off
adxl.setTapDetectionOnX(0);
adxl.setTapDetectionOnY(0);
adxl.setTapDetectionOnZ(1);

//set values for what is a tap, and what is a double tap (0-255)
adxl.setTapThreshold(50); //62.5mg per increment
adxl.setTapDuration(15); //625us per increment
adxl.setDoubleTapLatency(80); //1.25ms per increment
adxl.setDoubleTapWindow(200); //1.25ms per increment

//set values for what is considered freefall (0-255)
adxl.setFreeFallThreshold(7); //(5 - 9) recommended - 62.5mg per increment
adxl.setFreeFallDuration(45); //(20 - 70) recommended - 5ms per increment

//setting all interrupts to take place on int pin 1
//I had issues with int pin 2, was unable to reset it
adxl.setInterruptMapping( ADXL345_INT_SINGLE_TAP_BIT,   ADXL345_INT1_PIN );
```

Code

Interfacing_of_accelerometer | Arduino 1.8.19

File Edit Sketch Tools Help

Interfacing_of_accelerometer

```
  //setting all interrupts to take place on int pin 1
  //I had issues with int pin 2, was unable to reset it
  adxl.setInterruptMapping( ADXL345_INT_SINGLE_TAP_BIT,   ADXL345_INT1_PIN );
  adxl.setInterruptMapping( ADXL345_INT_DOUBLE_TAP_BIT,   ADXL345_INT1_PIN );
  adxl.setInterruptMapping( ADXL345_INT_FREE_FALL_BIT,    ADXL345_INT1_PIN );
  adxl.setInterruptMapping( ADXL345_INT_ACTIVITY_BIT,     ADXL345_INT1_PIN );
  adxl.setInterruptMapping( ADXL345_INT_INACTIVITY_BIT,   ADXL345_INT1_PIN );

  //register interrupt actions - 1 == on; 0 == off
  adxl.setInterrupt( ADXL345_INT_SINGLE_TAP_BIT, 1);
  adxl.setInterrupt( ADXL345_INT_DOUBLE_TAP_BIT, 1);
  adxl.setInterrupt( ADXL345_INT_FREE_FALL_BIT,  1);
  adxl.setInterrupt( ADXL345_INT_ACTIVITY_BIT,   1);
  adxl.setInterrupt( ADXL345_INT_INACTIVITY_BIT, 1);
}

void loop(){

  //Boring accelerometer stuff
  int x,y,z;
  adxl.readXYZ(&x, &y, &z); //read the accelerometer values and store them in variables  x,y,z
  // Output x,y,z values
  Serial.print("values of X , Y , Z: ");
  Serial.print(x);
```

Code

Interfacing_of_accelerometer | Arduino 1.8.19

File Edit Sketch Tools Help

Interfacing_of_accelerometer

```
  Serial.print(" , ");
  Serial.print(y);
  Serial.print(" , ");
  Serial.println(z);

  double xyz[3];
  double ax,ay,az;
  adxl.getAcceleration(xyz);
  ax = xyz[0];
  ay = xyz[1];
  az = xyz[2];
  Serial.print("X=");
  Serial.print(ax);
    Serial.println(" g");
  Serial.print("Y=");
  Serial.print(ay);
    Serial.println(" g");
  Serial.print("Z=");
  Serial.print(az);
    Serial.println(" g");
  Serial.println("**********************");
  delay(500);

}
```

Project Link : https://youtu.be/TLrNSPZuTJM