

# Understanding the 3-axis acceleration, accelerometer, term called 'Mapping', Understanding ADXL 335 sensor its construction, connections and coding.

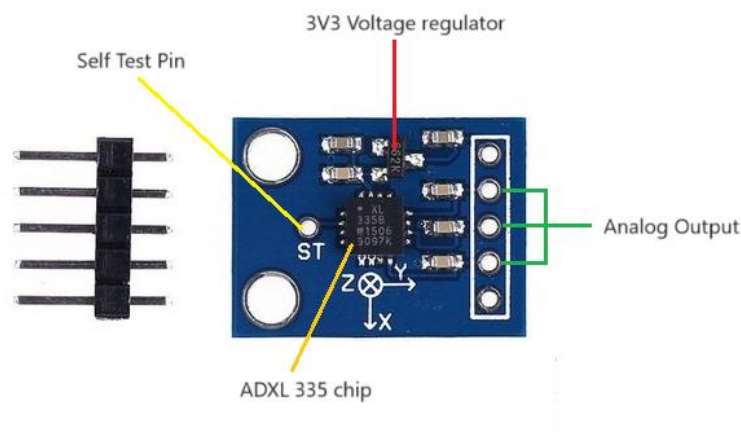
In this tutorial we will learn about ADXL335 accelerometer, its types and its interfacing with Arduino. We will also learn how does a sensor works and also its applications.

## Components required:

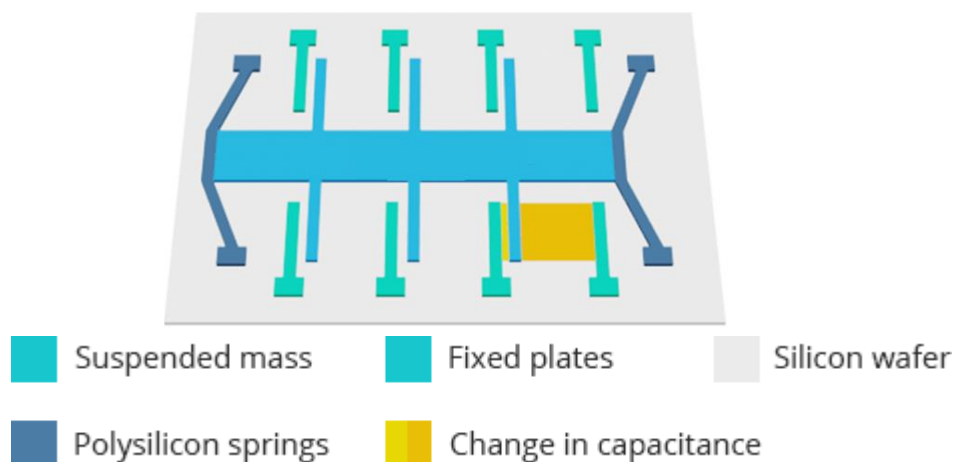
- 1) Arduino UNO – 01
- 2) Breadboard - 01
- 3) ADXL335 – 01
- 4) Servo motor SG90 – 01
- 5) USB A-B cable - 01
- 6) Jumper wires – as per the requirement
- 7) LED bulb – 03



## Construction of the ADXL 335-ADXL 335



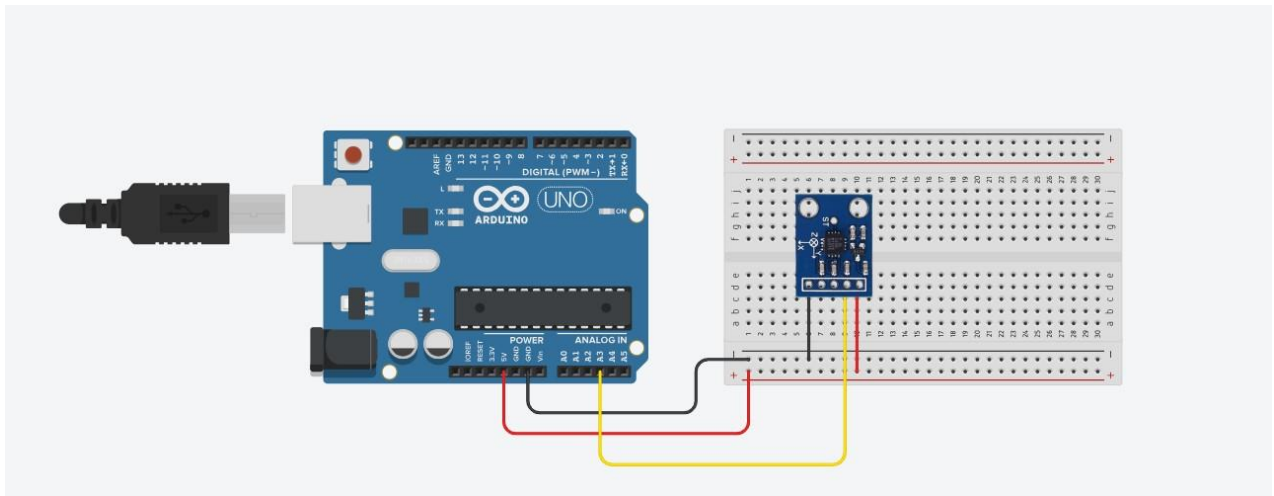
## Chip Construction demonstartion



Read out the sensing range values of the sensor-

Connect the circuit as per the details given below.

X pin – pin no. A3



### Arduino source code-

Copy and paste the Arduino source code given here to the Arduino IDE software from there you will upload the code to the Arduino UNO. Once uploaded, you will start getting the reading on the serial monitor. Hold the sensor flat to the surface and check what is the value, then turn to the complete left side vertical position and note the reading. Similarly bring it back to the flat position and then again turn it to the other side vertical to check the reading.

Here at both the position you will get two extreme values beyond which the sensor will not increase the value, instead it will start decreasing it or either it will start increasing.

Now you have understood both the values, hence take a mean of them and consider that value as the stable position of the sensor.

In my case I have got 266 on the left-hand side and 400 on the right-hand side. Hence-

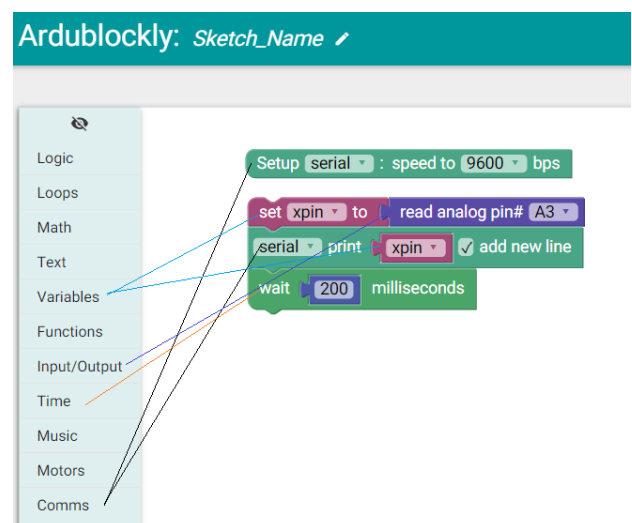
$$\text{Mean value} = \frac{266+400}{2} = 333$$

Use the given code to find out your values-

```
int xpin;

void setup() {
  Serial.begin(9600);
  pinMode(A3, INPUT);
}

void loop() {
  xpin = analogRead(A3);
  Serial.println(xpin);
  delay(200);
}
}
```

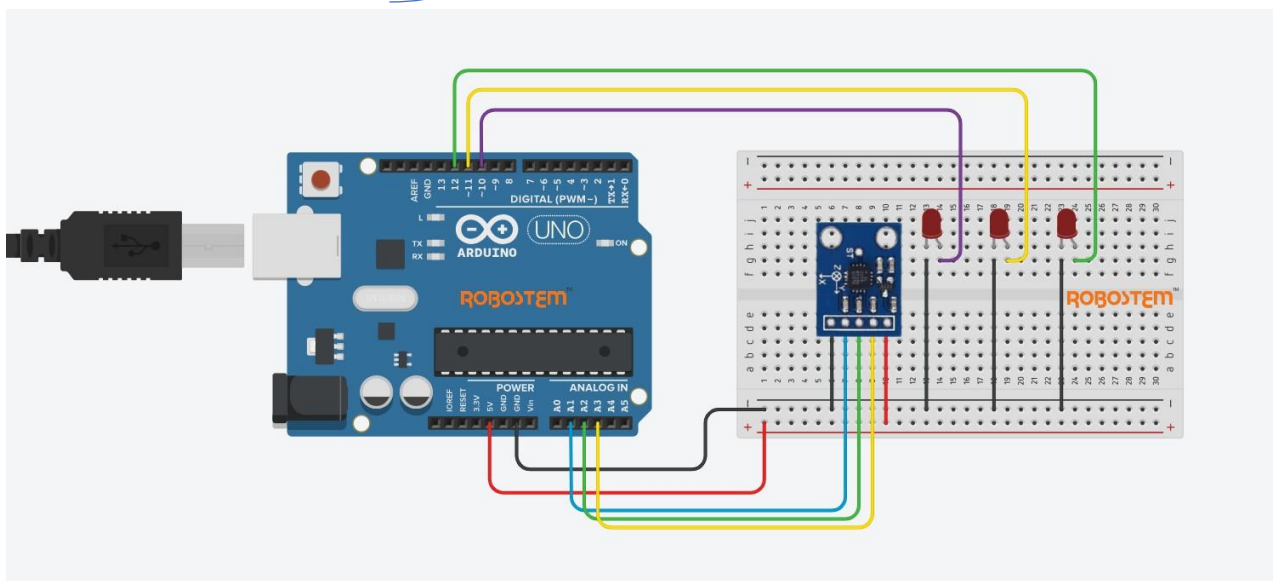


Some sample projects using ADXL 335

## 1) Control LED with ADXL 335: -

Connect the circuit as per the details given below.

X pin	- pin no. A3	} All these pins from Analog section of an Arduino
Y pin	- pin no. A2	
Z pin	- pin no. A1	
LED 1	- pin no. 10	} All these pins from Digital section of an Arduino
LED 2	- pin no. 11	
LED 3	- pin no. 12	



Do the circuit connections as shown in the picture.

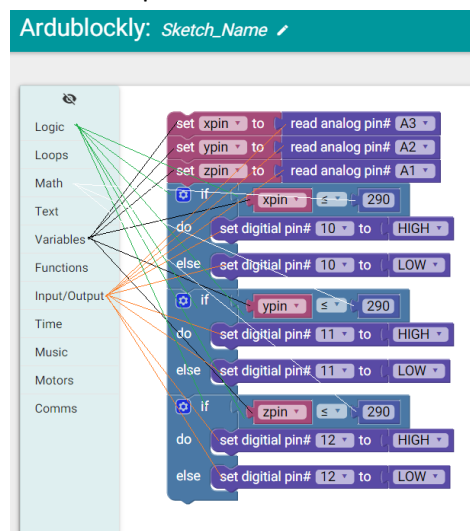
Arduino source code:

Copy and paste the Arduino source code given here to control glowing of an LED. LED will turn ON whenever there will be a tilt in the sensor with respect to the respective assigned axis only on single decided side and goes off whenever the axis comes back to its normal position.

```
int xpin;
int ypin;
int zpin;

void setup() {
  pinMode(A3, INPUT);
  pinMode(A2, INPUT);
  pinMode(A1, INPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() {
  xpin = analogRead(A3);
  ypin = analogRead(A2);
  zpin = analogRead(A1);
}
```



```

if (xpin <= 290) {
  digitalWrite(10, HIGH);
} else {
  digitalWrite(10, LOW);
}
if (ypin <= 290) {
  digitalWrite(11, HIGH);
} else {
  digitalWrite(11, LOW);
}
if (zpin <= 290) {
  digitalWrite(12, HIGH);
} else {
  digitalWrite(12, LOW);
}
}

```

### Arduino source code to check tilt on either side of an axis:

Copy and paste the Arduino source code given here to control glowing of an LED. LED will turn ON whenever there will be a tilt in the sensor with respect to the respective assigned axis on either side and goes off whenever the axis comes back to its normal position

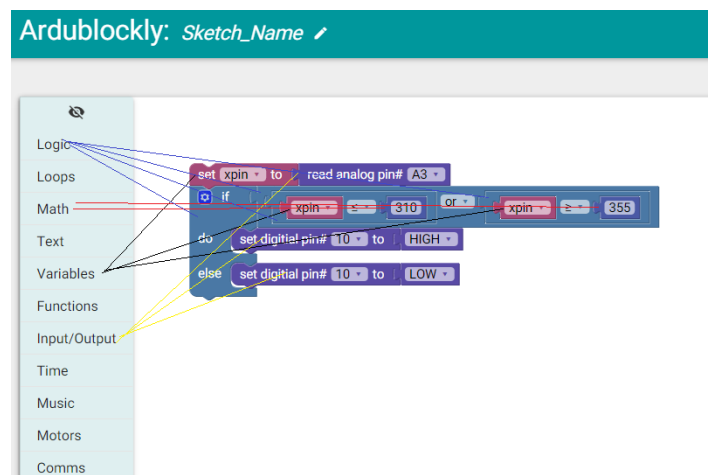
```

int xpin;

void setup() {
  pinMode(A3, INPUT);
  pinMode(10, OUTPUT);
}

void loop() {
  xpin = analogRead(A3);
  if (xpin <= 310 || xpin >= 355) {
    digitalWrite(10, HIGH);
  } else {
    digitalWrite(10, LOW);
  }
}

```

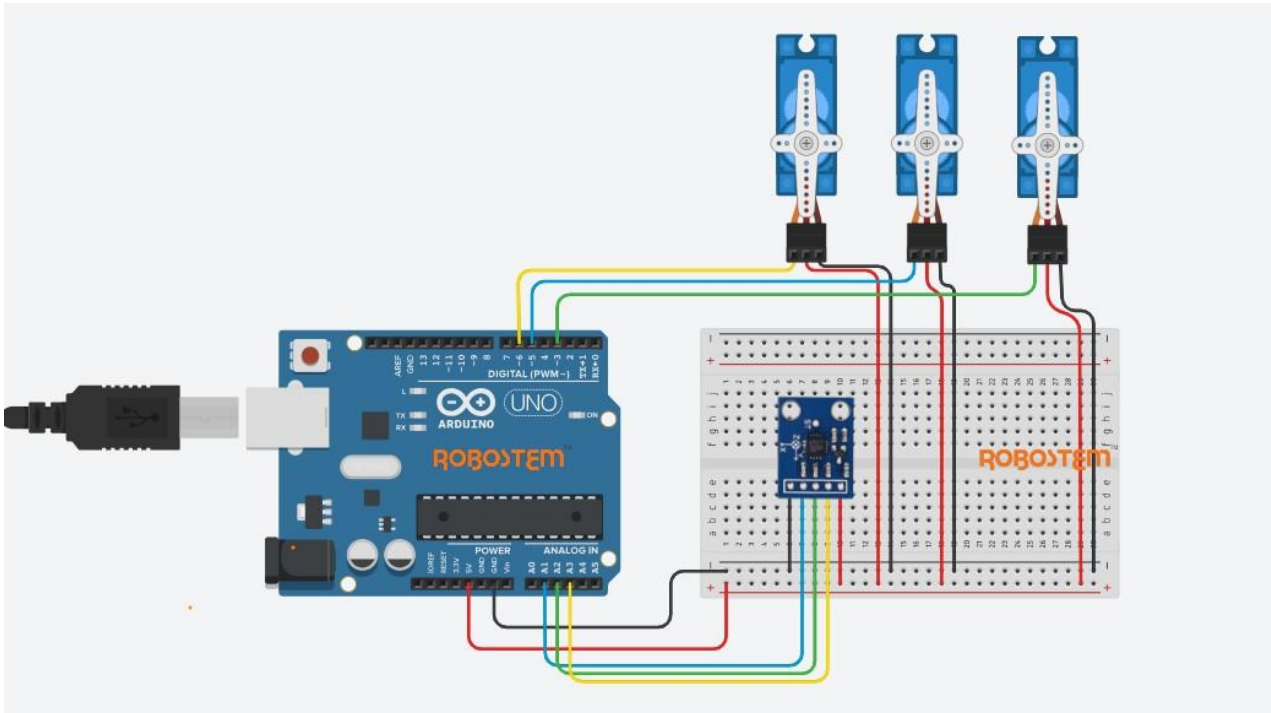


### 2) Control servo motor SG90 with ADXL 335: -

Connect the circuit as per the details given below.

- |         |              |   |
|---------|--------------|---|
| X pin   | – pin no. A3 | } All these pins from Digital section of an Arduino |
| Y pin   | – pin no. A2 |   |
| Z pin   | – pin no. A1 |   |
| SERVO 1 | – pin no. 3  | } All these pins from Digital section of an Arduino |
| SERVO 2 | – pin no. 5  |   |
| SERVO 3 | – pin no. 6  |   |

Do the circuit connections as shown in the picture



Arduino source code:

Copy and paste the Arduino source code given here to control servo motor movement. The servo motors will move according to the movement of the axis. Its beautiful to check with the practically experience, enjoy it.

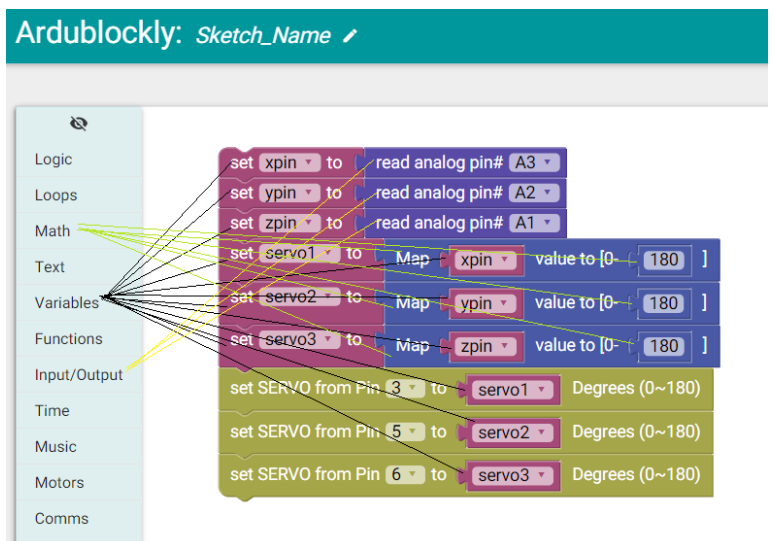
```
#include <Servo.h>
```

```
int xpin;
int ypin;
int zpin;
int servo1;
int servo2;
int servo3;
```

```
Servo myServo3;
Servo myServo5;
Servo myServo6;
```

```
void setup() {
  pinMode(A3, INPUT);
  pinMode(A2, INPUT);
  pinMode(A1, INPUT);
  myServo3.attach(3);
  myServo5.attach(5);
  myServo6.attach(6);
}
```

```
void loop() {
  xpin = analogRead(A3);
  ypin = analogRead(A2);
```



```
zpin = analogRead(A1);  
servo1 = (map(xpin, 266, 400, 0,  
180));  
servo2 = (map(ypin, 266, 400, 0,  
180));  
servo3 = (map(zpin, 266, 400, 0,  
180));  
myServo3.write(servo1);  
myServo5.write(servo2);  
myServo6.write(servo3);  
  
}
```

Here little changes we have to make in the mapping values after copy and pasting the code from the ardublockly to the Arduino IDE software. Instead of (0, 1024, 0, 180)), which we generally get in the ardublockly after coding, we will be mapping the vales that we have taken initially in the testing with one axis, i.e. (266, 400, 0, 180)),value here taken are 266 and 400 in my case as those are my two extreme values of an X-axis.



## Extra Dose



Design and build your own robotic controlled arm using the servo motor and ADXL 335 which will work according to the movement.



Build balancing robotic Dall using the ADXL 335.

HAPPY TINKERING

ROBOSTEM LEARNING SOLUTIONS

TF#2, 3<sup>rd</sup> floor, Oneness empire,

Sadashiv nagar, main road, Belgavi. KA, India. 590019.

[www.robostem.co.in](http://www.robostem.co.in) | Ph: 0831 3568302

admin@robostem.co.in