# Breakout - 3-axis Analog Accelerometer ADXL335

## Introduction

The ADXL335 is a small, thin, low power, complete 3-axis accel-erometer with signal conditioned voltage outputs. The product measures acceleration with a minimum full-scale range of ±3 g.

The module was designed as breakout board because ADXL335's signal is analog( more ports requested ),but the board outline is grove module that you can fix it convenient like others grove. The sensor combine 3.3 and 5V power supply,can be used in standard arduino device and seeduino stalker. The following program code include first-order filter which can make the output smoothly if the sensor was used in robot or toy car.


## Features

- Wide power range DC3V to 5V
- Grove outline
- 3 axis sensing
- Small, low-profile package: 4×4×1.45mm LFCSP
- Low power 350µA at 3V (typical)
- High sensitive
- 10,000 g shock survival
- BW adjustment with a single capacitor per axis
- RoHS/WEEE lead-free compliant

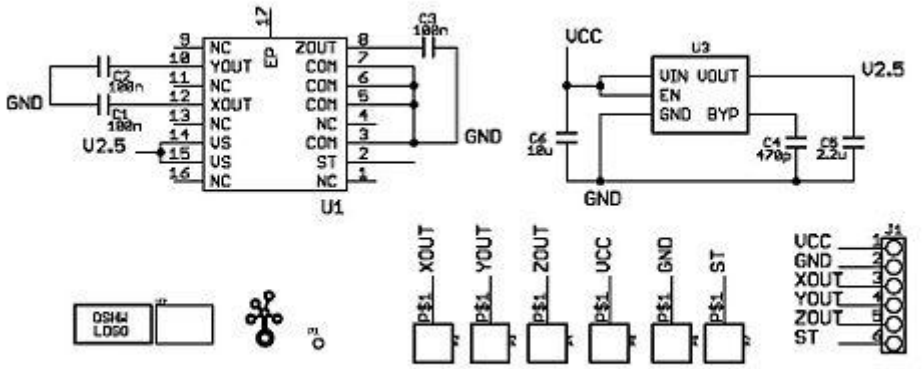
## Application Ideas

- Motion Sensor
- Shock detector
- Vibration sensor
- Toy car
- Robot

## Cautions

The warnings and wrong operations possible cause dangerous.

# Schematic




# Usage

- **How to connect**

The sensor's outline is breakout board, you can welding wire in the board or use jumper wire to connect the sensor. The VCC connect to power source（DC5V or DC3.3V）, GND to ground, X to arduino annalog port A0, Y to A1, Z to A2. Downloading the demo code,then open serial monitor, turn the sensor any angle, you can see the digital angle value which feedbacked by accelerometer from monitor.

- **How to regulate the sensor**

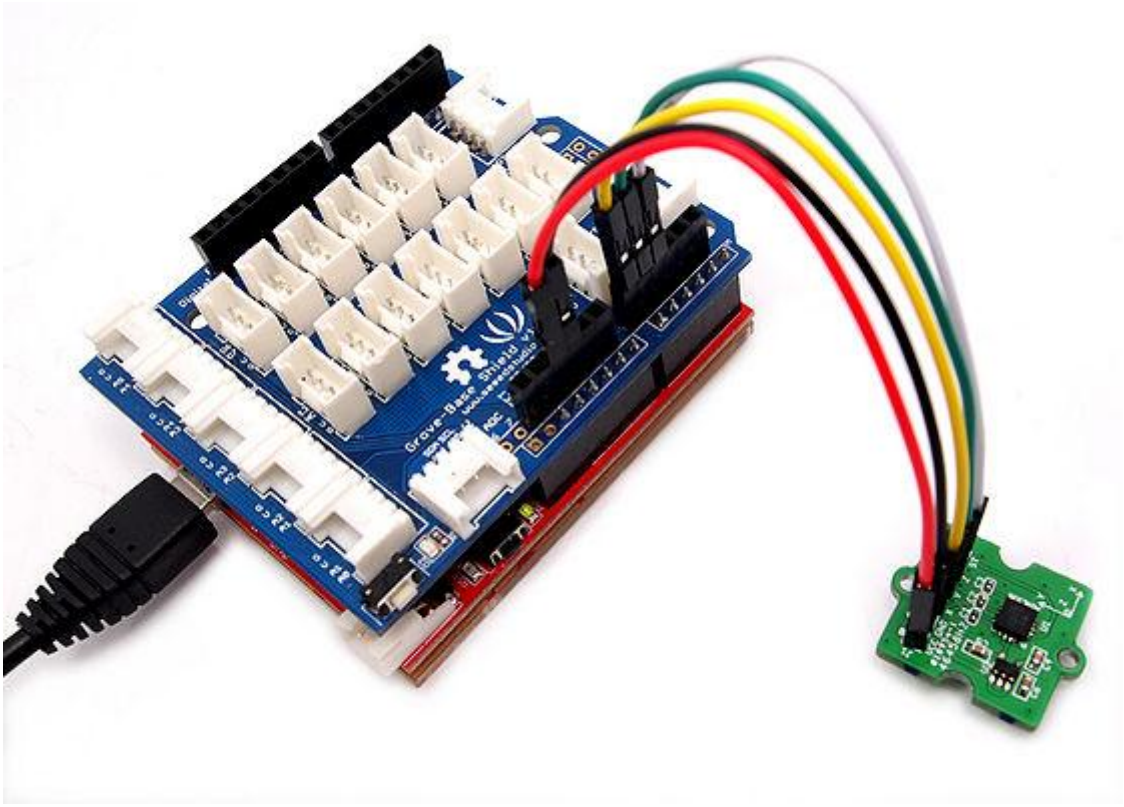
**The sensor is analog device, you should regulate the sensor before combine it to you system.** We set six variances(correspond to 3 axis) in program, e.g. X axis,variance **xminVal,xmaxVal**. At first used, we suggest you running a measure code to detect the 3 axis's max and min value of the sensor. The following step will show you how to operate.

**Step 1:** The six variances is xminVal,xmaxVal,yminVal,ymaxVal,zminVal,zmaxVal from line 17 to 22 in demo code. This variances saved the max and min value of each axis which in the positive and negative position. As the second diagram showed.

**Step 2*:*** Changing you demo code, running the function(**valueprint**) in demo code line 96 and cancel function nameed **outputdata**(line 95) then download it.As the third diagram showed.

**Step 3:** Open your serial monitor, make sure the sensor is connected. Follow the axis institutions printed on sensor's board, place the X arrow straight up then note down the value as xmaxValue. Changed the sensor position, make the x arrow in opposite directions then note down the value as xminValue from the serial monitor. repeat above operation to obtain Y and Z axes value. Now, you charge the characters of the sensor.

**Step 4:** Make your sensor work under regulated. Change the regulate variances(xminVal and others) according the feedback data in step 3.



_3_axis_annalog_accelerometer_demo_code_pde_pde | Ardui...

File Edit Sketch Tools Help

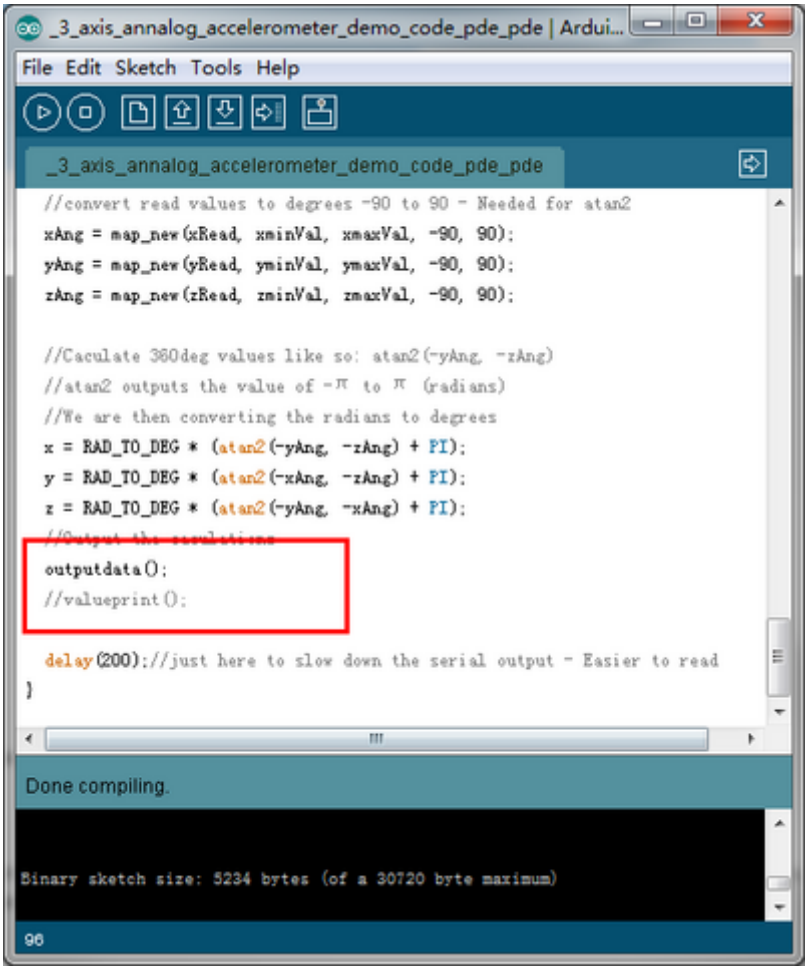
_3_axis_annalog_accelerometer_demo_code_pde_pde

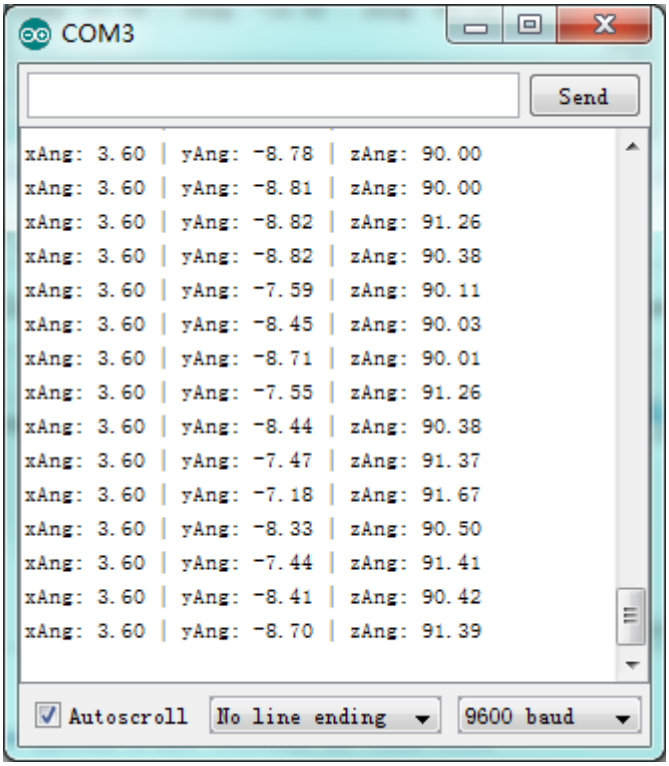
```
const unsigned char zPin = 2;
float xRead = 0;
float yRead = 0;
float zRead = 0;
//The minimum and maximum values that came from
//the accelerometer while standing still
//You very well may need to change these
float xminVal = 200;
float xmaxVal = 300;
float yminVal = 205;
float ymaxVal = 307;
float zminVal = 205;
float zmaxVal = 306;

float xAng = 0;
float yAng = 0;
float zAng = 0;
```

Done compiling.

```
Binary sketch size: 5234 bytes (of a 30720 byte maximum)
```

19




# Programming

```
//////////////////////////////////////////////////////////////////////
//This code reference from bildr,thanks!
//Released under the MIT License - Please reuse change and share
//Simple code for the ADXL335, prints calculated orientation via serial
//////////////////////////////////////////////////////////////////////

//Analog read pins
const unsigned char xPin = 0;
const unsigned char yPin = 1;
const unsigned char zPin = 2;
float xRead = 0;
float yRead = 0;
float zRead = 0;
//The minimum and maximum values that came from
//the accelerometer while standing still
//You very well may need to change these
float xminVal = 200;
float xmaxVal = 300;
float yminVal = 205;
float ymaxVal = 307;
float zminVal = 205;
float zmaxVal = 306;

float xAng = 0;
float yAng = 0;
float zAng = 0;

float FTX=0;
float FTY=0;
float FTZ=0;
```

```
float a=0.3;

//to hold the caculated values
double x;
double y;
double z;

void valueprint()
{
  Serial.print(xRead);
  Serial.print('\t');
  Serial.print(yRead);
  Serial.print('\t');
  Serial.println(zRead);
}

float map_new(float x, float in_min, float in_max, float out_min, float out_max)
{
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}

void outputdata()
{
  Serial.print("xAng: ");
  Serial.print(xAng);
  Serial.print(" | yAng: ");
  Serial.print(yAng);
  Serial.print(" | zAng: ");
  Serial.println(zAng);
}
void setup()
{
  Serial.begin(9600);
  FTX=analogRead(xPin);
  FTY=analogRead(yPin);
  FTZ=analogRead(zPin);
}

void loop(){

  //read the analog values from the accelerometer
  xRead = analogRead(xPin);
  yRead = analogRead(yPin);
  zRead = analogRead(zPin);

  xRead=a*xRead+(1-a)*FTX;
  yRead=a*yRead+(1-a)*FTY;
  zRead=a*zRead+(1-a)*FTZ;
  FTX=xRead;
  FTY=yRead;
  FTZ=zRead;

  //convert read values to degrees -90 to 90 - Needed for atan2
  xAng = map_new(xRead, xminVal, xmaxVal, -90, 90);
  yAng = map_new(yRead, yminVal, ymaxVal, -90, 90);
  zAng = map_new(zRead, zminVal, zmaxVal, -90, 90);

  //Caculate 360deg values like so: atan2(-yAng, -zAng)
  //atan2 outputs the value of -π to π (radians)
  //We are then converting the radians to degrees
  x = RAD_TO_DEG * (atan2(-yAng, -zAng) + PI);
  y = RAD_TO_DEG * (atan2(-xAng, -zAng) + PI);
  z = RAD_TO_DEG * (atan2(-yAng, -xAng) + PI);
  //Output the caculations
  outputdata();
  //valueprint();

  delay(200);//just here to slow down the serial output - Easier to read
}
```

## Support

If you have questions or other better design ideas, you can go to our forum or wish to discuss.

## Resources

Breakout - 3-axis analog accelerometer V1.0 Eagle file.zip

3 axis annalog accelerometer demo code pde.zip

ADXL335 datasheet.pdf

## See Also

Other Accelerometer

## External Links

Sensing Orientation With The ADXL335 + Arduino in bildr blog.

Categories: Sensors | GROVE