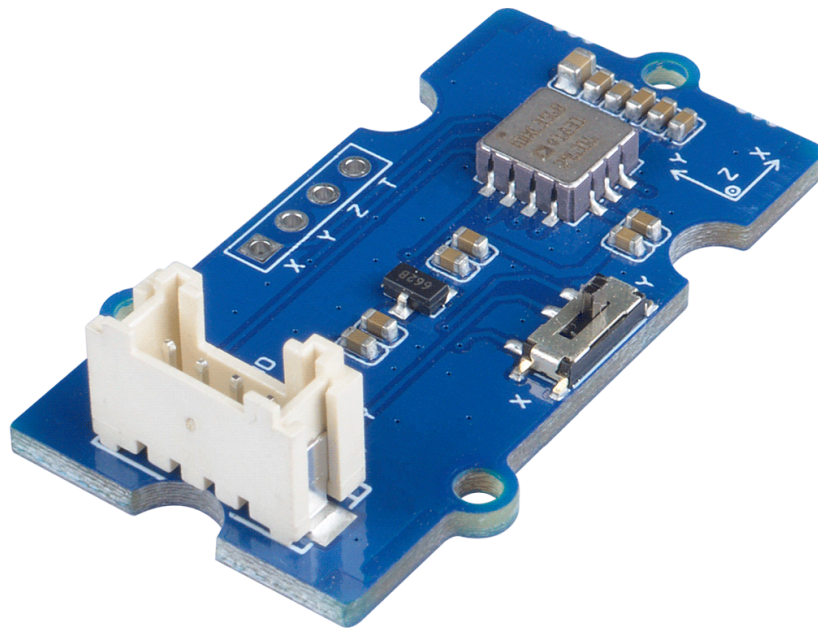


Grove - 3-Axis Analog Accelerometer 20g (ADXL356B)



You can find a variety of [3-axis accelerometers](#) [<https://www.seeedstudio.com/tag/accelerometer.html>] on our website that can meet different scenarios and needs. This time, we bring you the industrial grade, high stability, high precision and low power ADI ADXL series three-axis accelerometers.

The Grove - 3-Axis Analog Accelerometer $\pm 20g$ (ADXL356B) is a analog output **MEMS** [\[https://www.seeedstudio.com/tag/MEMS.html\]](https://www.seeedstudio.com/tag/MEMS.html) Accelerometer. This sensor has three two selectable measuring ranges: $\pm 10g$, $\pm 20g$. You just need to do little calibration work to get a relatively accurate result. The On-Board grove port can output two channel analog data: one for Z-axis, one for X/Y-axis. You can choose output the X-axis or Y-axis signal with the on-board switch. Also you can use the 4-pin welding hole to output X/Y/Z axis at the same time. The power consumption of this sensor is extremely low, $150 \mu A$ in normal operation mode and even only $21 \mu A$ in standby mode. You can switch the operating mode by changing the pad connection on the back.

The ADI ADXL Series Accelerometer includes four products that will meet your different range and output needs:

Product	Measurement Range	Output Port	Power Consumption
Grove - 3-Axis Analog Accelerometer \pm20g (ADXL356B) [https://www.seeedstudio.com/Grove-3-Axis-Analog-Accelerometer-20g-ADXL356B-p-4004.html]	\pm 10 \pm 20g	Analog	mea moc star moc
Grove - 3-Axis Analog Accelerometer \pm40g (ADXL356C) [https://www.seeedstudio.com/Grove-3-Axis-Analog-Accelerometer-40g-ADXL356C-p-4006.html]	\pm 10g \pm 40g	Analog	mea moc star moc
Grove - 3-Axis Digital Accelerometer \pm40g (ADXL357) [https://www.seeedstudio.com/Grove-3-Axis-Digital-Accelerometer-40g-ADXL357-p-4005.html]	\pm 10g@51200 LSB/g \pm 20g@25600 LSB/g \pm 40g@12800 LSB/g	Digital I2C	mea moc
Grove - 3-Axis Digital Accelerometer \pm200g (ADXL372) [https://www.seeedstudio.com/Grove-3-Axis-Digital-Accelerometer-200g-ADXL372-p-4003.html]	\pm 200g	Digital I2C	mea moc



Get One Now 

[\[https://www.seeedstudio.com/Grove-3-Axis-Analog-Accelerometer-20g-ADXL356B-p-4004.html\]](https://www.seeedstudio.com/Grove-3-Axis-Analog-Accelerometer-20g-ADXL356B-p-4004.html)

Features

- Industry leading noise, minimal offset drift over temperature, and long-term stability, enabling precision applications with minimal calibration.
- Hermetic package offers excellent long-term stability 0 g offset vs. temperature (all axes): 0.75 mg/°C maximum
- The low noise of the ADXL356 over higher frequencies is ideal for wireless condition monitoring.
- Ultra low power consumption: Normal operation mode-150 μ A, Standby mode 21 μ A
- Low drift, low noise

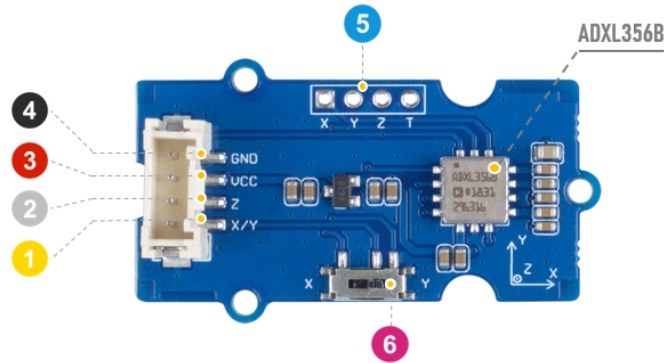
Specification

Parameter	Value
Supply voltage	3.3V / 5V
Operating ambient temperature	-40 – 125°C
Output Full-Scale Range (FSR)	±10g / ±20g
Sensitivity at X _{OUT} , Y _{OUT} , Z _{OUT} (Ratiometric to V1P8ANA)	±10 g@80 mv/g (Typ.) ±20 g@40 mv/g (Typ.)
Sensitivity Change due to Temperature	±0.01%/°C (TA = -40°C to +125°C)
0g OFFSET (Referred to V1P8ANA/2)	±125 mg(Typ.)
Output interface	Analog

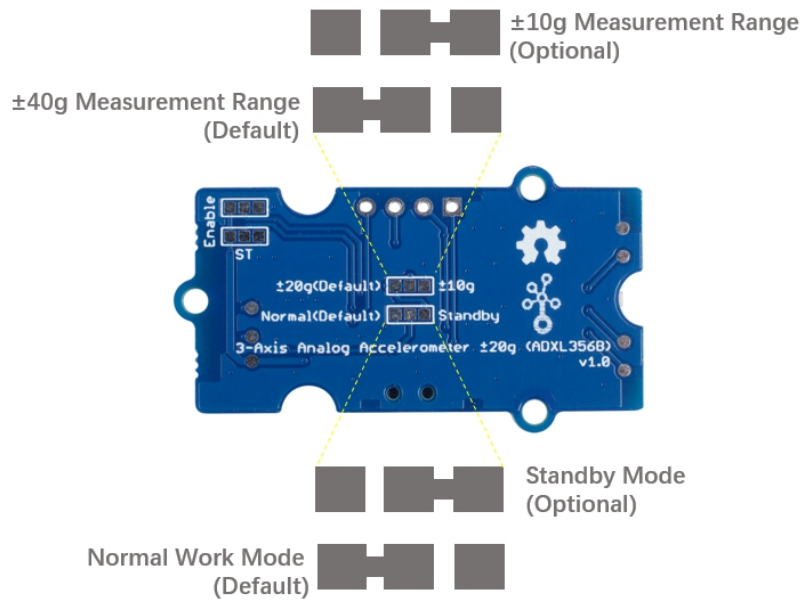
Applications

- Inertial measurement units (IMUs)/altitude and heading reference systems (AHRSS)
- Platform stabilization systems
- Structural health monitoring
- Condition monitoring
- Seismic imaging
- Tilt sensing
- Robotics


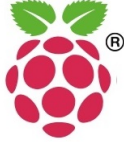
Pinout



- 4 GND:** connect this module to the system GND
- 3 VCC:** you can use 5V or 3.3V for this module
- 2 Z:** Z-axis analog data output
- 1 X/Y:** X-axis or Y-axis analog data output
- 5 4 Pin Output:** X/Y/Z axis analog data output, Temperature output
- 6 Switch:** switch X/Y output



Platforms Supported


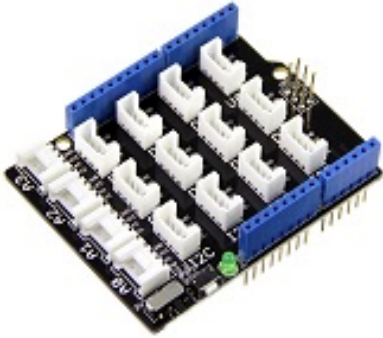
Arduino	Raspberry Pi		
			



Getting Started

Play With Arduino

Materials required

Seeeduino V4.2	Base Shield
	
<p>Get ONE Now [https://www.seeedstudio.com/Seeeduino-V4.2-p-2517.html]</p>	<p>Get ONE Now [https://www.seeedstudio.com/Base-Shield-V2-p-1378.html]</p>



In addition, you can consider our new [Seeeduino Lotus M0+](https://www.seeedstudio.com/Seeeduino-Lotus-Cortex-M0-p-2896.html) [https://www.seeedstudio.com/Seeeduino-Lotus-Cortex-M0-p-2896.html], which is equivalent to the combination of Seeeduino V4.2 and Baseshield.

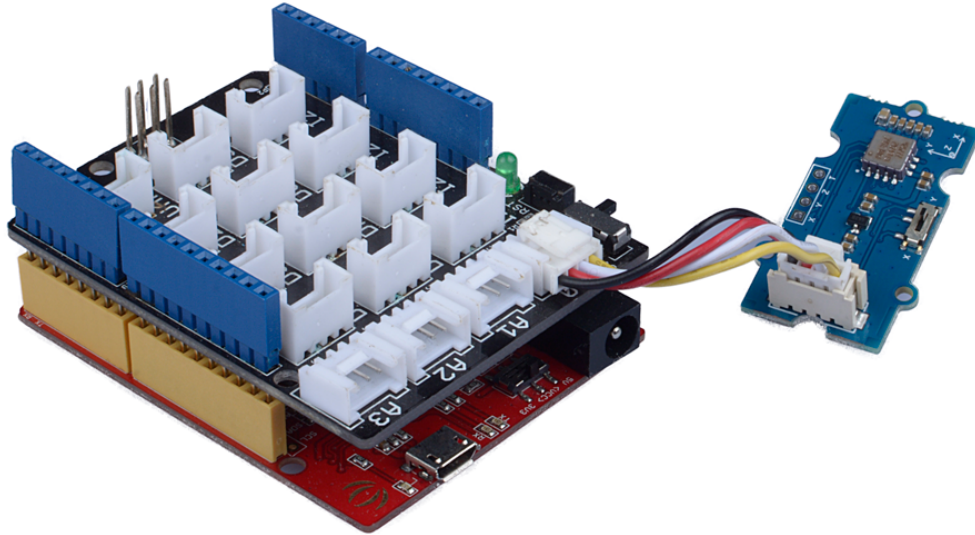
**Note**

1 Please plug the USB cable gently, otherwise you may damage the port. Please use the USB cable with 4 wires inside, the 2 wires cable can't transfer data. If you are not sure about the wire you have, you can click [here](https://www.seeedstudio.com/Micro-USB-Cable-48cm-p-1475.html) [https://www.seeedstudio.com/Micro-USB-Cable-48cm-p-1475.html] to buy

2 Each Grove module comes with a Grove cable when you buy. In case you lose the Grove cable, you can click [here](https://www.seeedstudio.com/Grove-Universal-4-Pin-Buckled-20cm-Cable-%285-PCs-pack%29-p-936.html) [https://www.seeedstudio.com/Grove-Universal-4-Pin-Buckled-20cm-Cable-%285-PCs-pack%29-p-936.html] to buy.

Hardware Connection

- **Step 1.** Connect the Grove - 3-Axis Analog Accelerometer $\pm 20g$ (ADXL356B) to the **A0** port of the Base Shield.
- **Step 2.** Plug Grove - Base Shield into Seeeduino.
- **Step 3.** Connect Seeeduino to PC via a USB cable.



Software



Attention

If this is the first time you work with Arduino, we strongly recommend you to see [Getting Started with Arduino](https://wiki.seeedstudio.com/Getting_Started_with_Arduino/) [https://wiki.seeedstudio.com/Getting_Started_with_Arduino/] before the start.

- **Step 1.** Download the [Seeed_ADXL356.ino](https://github.com/linux-downey/Seeed_ADXL356/blob/master/Seeed_ADXL356/Seeed_ADXL356.ino) [https://github.com/linux-downey/Seeed_ADXL356/blob/master/Seeed_ADXL356/Seeed_ADXL356.ino] from Github.

or you can just copy the code below:

```
1  #include <Arduino.h>
2
3  #ifdef ARDUINO_SAMD_VARIANT_COMPLIANCE
```



```
4   #define SERIAL SerialUSB
5   #define SYS_VOL    3.3
6   #else
7   #define SERIAL Serial
8   #define SYS_VOL    5
9   #endif
10
11  float cali_data_xy;
12  float cali_data_z;
13  int16_t scale;
14
15  #define MODUEL_RANGE      20
16
17  #define MODULE_VOL        1.8
18
19  #define CALI_BUF_LEN      15
20  #define CALI_INTERVAL_TIME 250
21
22  float cali_buf_xy[CALI_BUF_LEN];
23  float cali_buf_z[CALI_BUF_LEN];
24
25  #define XY_PIN    A0
26  #define Z_PIN     A1
27
28  float deal_cali_buf(float *buf)
29  {
30      float cali_val = 0;
31
32      for(int i = 0;i < CALI_BUF_LEN;i++)
33      {
34          cali_val += buf[i];
35      }
36      cali_val = cali_val/CALI_BUF_LEN;
37      return (float)cali_val;
38  }
39
40
41  void calibration(void)
42  {
43      SERIAL.println("Please Place the module horizontall
44      delay(1000);
```

```
45     SERIAL.println("Start calibration.....");
46
47     for(int i=0;i<CALI_BUF_LEN;i++)
48     {
49         cali_buf_xy[i] = analogRead(XY_PIN);;
50         cali_buf_z[i] = analogRead(Z_PIN);
51         delay(CALI_INTERVAL_TIME);
52     }
53     cali_data_xy = deal_cali_buf(cali_buf_xy);
54     cali_data_z = (float)deal_cali_buf(cali_buf_z);
55     SERIAL.println("Calibration OK!!");
56     scale = (float)1000 / (cali_data_z - cali_data_xy);
57     cali_data_z -= (float)980 / scale;
58     SERIAL.println(cali_data_xy);
59     SERIAL.println(cali_data_z);
60     SERIAL.println(scale);
61
62 }
63
64
65
66 void AccMeasurement(void)
67 {
68     int16_t val_xy = 0;
69     int16_t val_z = 0;
70     val_xy = analogRead(XY_PIN);
71     val_z = analogRead(Z_PIN);
72
73     SERIAL.print("Raw data xy = ");
74     SERIAL.println(val_xy);
75     SERIAL.print("Raw data z = ");
76     SERIAL.println(val_z);
77     SERIAL.println(" ");
78
79     val_xy -= cali_data_xy;
80     val_z -= cali_data_z;
81     SERIAL.print("x or y position acc is ");
82     SERIAL.print(val_xy * scale / 1000.0);
83     SERIAL.println(" g ");
84     SERIAL.print("z position acc is ");
85     SERIAL.print(val_z * scale / 1000.0);
```

```
86     SERIAL.println(" g ");
87     SERIAL.println(" ");
88     SERIAL.println(" ");
89     SERIAL.println(" ");
90     delay(1000);
91 }
92
93 void setup()
94 {
95     SERIAL.begin(115200);
96     #ifdef ARDUINO_SAMD_VARIANT_COMPLIANCE
97     analogReadResolution(12);
98     #endif
99     calibration();
100    SERIAL.print("Scale = ");
101    SERIAL.println(scale);
102
103 }
104
105
106
107 void loop()
108 {
109     AccMeasurement();
110 }
```

- **Step 2.** Upload the demo. If you do not know how to upload the code, please check [How to upload code](https://wiki.seeedstudio.com/Upload_Code/) [https://wiki.seeedstudio.com/Upload_Code/].
- **Step 3.** Open the **Serial Monitor** of Arduino IDE by click **Tool->Serial Monitor**. Or tap the **Ctrl + Shift + M** key at the same time. Set the baud rate to **115200**.
- **Step 4. Calibration** Follow the calibration tips in the Serial Monitor, just few step the calibration will be finished

- **Step 5.** Now you can use this sensor, and the output will be like this:

```
1 Please Place the module horizontally!
2 Start calibration.....
3 Calibration OK!!
4 184.93
5 185.03
6 121
7 Scale = 121
8 Raw data xy = 185
9 Raw data z = 193
10
11 x or y position acc is 0.00 g
12 z position acc is 0.85 g
13
14
15
16 Raw data xy = 188
17 Raw data z = 196
18
19 x or y position acc is 0.36 g
20 z position acc is 1.21 g
```



Attention

If you use Grove port to output the data, the X axis and the Y axis can not be output at the same time, you can use the on-board switch to select the output channel. If you want to output X/Y/Z at the same time, you can use the 4 pin welding hole.

FAQ

Q1: How to select the $\pm 10g$ measurement range?

A1: To change the measurement range you need to modify both the hardware and software. First, cut the back pad which connected to $\pm 20g$ and re-solder it to $\pm 10g$. Then modify the code in the code block line 12

```
#define MODUEL_RANGE      20
```



\downdownarrows

```
#define MODUEL_RANGE      10
```



Schematic Online Viewer



Resources

- **[ZIP]** [Grove - 3-Axis Analog Accelerometer ±20g \(ADXL356B\) Schematic file](https://files.seeedstudio.com/wiki/Grove-3-Axis_Analog_Accelerometer-20g-ADXL356B/res/Grove%20-%203-Axis%20Analog%20Accelerometer%20%C2%B120g%20(ADXL356B).zip) [https://files.seeedstudio.com/wiki/Grove-3-Axis_Analog_Accelerometer-20g-ADXL356B/res/Grove%20-%203-Axis%20Analog%20Accelerometer%20%C2%B120g%20(ADXL356B).zip]

- **[PDF] ADXL 356B Datasheet**
[https://files.seeedstudio.com/wiki/Grove-3-Axis_Analog_Accelerometer-20g-ADXL356B/res/ADXL356B.pdf]

Tech Support

Please submit any technical issue into our [forum](https://forum.seeedstudio.com/)
[<https://forum.seeedstudio.com/>]



[https://www.seeedstudio.com/act-4.html?utm_source=wiki&utm_medium=wikibanner&utm_campaign=newproducts]