Grove - Piezo Vibration Sensor

Grove-Piezo Vibration Sensor is suitable for measurements of flexibility, vibration, impact and touch. The module is based on PZT film sensor LDT0-028. When the sensor moves back and forth, a certain voltage will be generated by the voltage comparator inside of it. A wide dynamic range (0.001Hz~1000MHz) guarantees an excellent measuring performance. And, you can adjust its sensitivity by adjusting the on-board potentiometer with a screw.

**Version**

<table>
<thead>
<tr>
<th>Product Version</th>
<th>Changes</th>
<th>Released Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grove - Piezo Vibration Sensor V1.1</td>
<td>Initial</td>
<td>Jul 2014</td>
</tr>
</tbody>
</table>

**Features**

- Standard grove socket
- Wide dynamic range : 0.1Hz~180Hz
- Adjustable sensitivity
- High receptivity for strong impact

**Tip**

More details about Grove modules please refer to [Grove System](#)
Platforms Supported

<table>
<thead>
<tr>
<th>Arduino</th>
<th>Raspberry Pi</th>
<th>BeagleBone</th>
<th>Wio</th>
<th>Linkit ONE</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Arduino Icon]</td>
<td>![Raspberry Pi Icon]</td>
<td>![BeagleBone Icon]</td>
<td>![Wio Icon]</td>
<td>![Linkit ONE Icon]</td>
</tr>
</tbody>
</table>

**Caution**
The platforms mentioned above as supported is/are an indication of the module’s software or theoretical compatibility. We only provide software library or code examples for Arduino platform in most cases. It is not possible to provide software library / demo code for all possible MCU platforms. Hence, users have to write their own software library.

**Applications**

- Vibration Sensing in Washing Machine
- Low Power Wakeup Switch
- Low Cost Vibration Sensing
- Car Alarms
- Body Movement
- Security Systems

**Getting Started**

**Play With Arduino**

**Hardware**

The Grove - Piezo Vibration Sensor outputs a logic HIGH when vibration was detected. We can use any of Arduino pins to read the data. Here is an example of Piezo Vibration Sensor controlling LED. When the vibration was detected, this sensor outputs a logic high signal (the sensitivity can be changed by adjusting the potentiometer), an LED lights up.

- Step 1. Prepare the below stuffs:
<table>
<thead>
<tr>
<th>Seeeduino V4</th>
<th>Base Shield</th>
<th>Grove - Piezo Vibration</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image of Seeeduino V4" /></td>
<td><img src="image2.png" alt="Image of Base Shield" /></td>
<td><img src="image3.png" alt="Image of Grove - Piezo Vibration" /></td>
</tr>
</tbody>
</table>

- Step 2. Connect the module to the **D2** of base shield using the 4-pin grove cable, we use **digital pin13 on board LED** as output.
- Step 3. Plug the Basic Shield into Arduino.
- Step 4. Connect Arduino to PC by using a USB cable.

**Note**
It may output low level even though originally output high level when you increase the threshold voltage by clockwise adjusting the potentiometer.
Software

- Step 1. Copy and paste code below to a new Arduino sketch.

```cpp
const int ledPin = 13;
void setup() {
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  int sensorState = digitalRead(2);
  Serial.println(sensorState);
  delay(100);
  if(sensorState == HIGH) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }
}
```

- Step 2. The LED will be on when vibration is detected.

Play With Raspberry Pi (With Grove Base Hat for Raspberry Pi)

Hardware

- Step 1. Things used in this project:

<table>
<thead>
<tr>
<th>Raspberry pi</th>
<th>Grove Base Hat for RasPi</th>
<th>Grove - Piezo Vibration</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Raspberry Pi" /></td>
<td><img src="image2.png" alt="Grove Base Hat for RasPi" /></td>
<td><img src="image3.png" alt="Grove - Piezo Vibration" /></td>
</tr>
</tbody>
</table>

- Step 2. Plug the Grove Base Hat into Raspberry.
- Step 3. Connect the Grove - Piezo Vibration Sensor to port 12 of the Base Hat.
- Step 4. Connect the Raspberry Pi to PC through USB cable.
**Note**
For step 3 you are able to connect the piezo vibration sensor to **any GPIO Port** but make sure you change the command with the corresponding port number.

**Software**

- **Step 1.** Follow [Setting Software](#) to configure the development environment.

- **Step 2.** Download the source file by cloning the grove.py library.
  
  ```
  cd ~
git clone https://github.com/Seeed-Studio/grove.py
  ```

- **Step 3.** Execute below commands to run the code.
  
  ```
  cd grove.py/grove
  python grove_piezo_vibration_sensor.py 12
  ```

Following is the grove_piezo_vibration_sensor.py code.

```python
import time
from grove.gpio import GPIO

class GrovePiezoVibrationSensor(GPIO):
    def __init__(self, pin):
        super(GrovePiezoVibrationSensor, self).__init__(pin, GPIO.IN)
        self._on_detect = None

    @property
    def on_detect(self):
        return self._on_detect
```
@on_detect.setter
def on_detect(self, callback):
    if not callable(callback):
        return
    if self.on_event is None:
        self.on_event = self._handle_event
    self._on_detect = callback
def _handle_event(self, pin, value):
    if value:
        if callable(self._on_detect):
            self._on_detect()

Grove = GrovePiezoVibrationSensor
def main():
    import sys
    if len(sys.argv) < 2:
        print('Usage: {} pin'.format(sys.argv[0]))
        sys.exit(1)
    pir = GrovePiezoVibrationSensor(int(sys.argv[1]))
    def callback():
        print('Detected.')
    pir.on_detect = callback
    while True:
        time.sleep(1)

if __name__ == '__main__':
    main()

Success
If everything goes well, you will be able to see the following result

pi@raspberrypi:~/grove.py/grove $ python grove_piezo_vibration_sensor.py 12
Detected.
Detected.
Detected.
Detected.
Detected.
Detected.
Detected.
Detected.
Detected.
^CTraceback (most recent call last):
Play With Raspberry Pi (with GrovePi_Plus)

**Hardware**

- Step 1. Prepare the below stuffs:
  - Raspberry pi
  - GrovePi_Plus
  - Grove - Piezo Vibration

- Step 2. Plug the GrovePi_Plus into Raspberry.
- Step 3. Connect Grove-Piezo Vibration to A0 port of GrovePi_Plus.
- Step 4. Connect the Raspberry to PC through USB cable.
**Software**

- Step 1. Follow Setting Software to configure the development environment.
- Step 2. Git clone the Github repository.
  ```
  cd ~
git clone https://github.com/DexterInd/GrovePi.git
  ```
- Step 3. Execute below commands to detect the vibration.
  ```
  cd ~/GrovePi/Software/Python
  python grove_piezo_vibration_sensor.py
  ```

Here is the grove_piezo_vibration_sensor.py code.

```python
import time
import grovepi
# Connect the Grove Piezo Vibration Sensor to analog port A0
# OUT,NC,VCC,GND
piezo = 0
grovepi.pinMode(piezo, "INPUT")
while True:
    try:
        # When vibration is detected, the sensor outputs a logic high signal
        print grovepi.analogRead(piezo)
        time.sleep(.5)
    except IOError:
        print "Error"
```

- Step 4. We will see the vibration display on terminal as below.
  ```
pi@raspberrypi:~/GrovePi/Software/Python $ python grove_piezo_vibration_sensor.py
1023 1023 1023 1023 18 17 18 17
```
Note
We also can use grovepi.digitalRead(2) to read the vibration status with attaching the sensor to D2 port of GrovePi.

FAQs

Q1: Is it digital or analog output?
A1: It is digital output, Low or High.

Resources

- [Eagle] Grove - Piezo Vibration Sensor Eagle File
- [PDF] Grove - Piezo Vibration Sensor Schematic PDF File
- [PDF] Grove - Piezo Vibration Sensor PCB PDF File
- [Datasheet] Piezo Vibration Sensor Datasheet

Projects

Grove Starter Kit For Arduino - Piezo Vibration Sensor: Teaches you how to use the Piezo vibration sensor in the Arduino Grove starter kit.

Seat Monitor: Using ARTIK cloud to monitor cabin seat state.

Tech Support

Please submit any technical issue into our forum or drop mail to techsupport@seeed.cc.