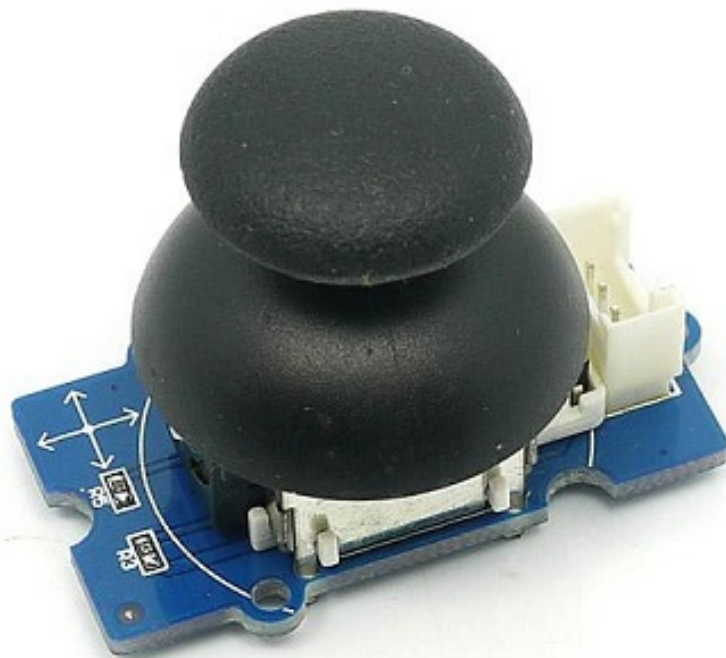


Grove - Thumb Joystick



Grove - Thumb Joystick is a Grove compatible module which is very similar to the 'analog' joystick on PS2 (PlayStation 2) controllers. The X and Y axes are two ~10k potentiometers which control 2D movement by generating analog signals. The joystick also has a push button that could be used for special applications. When the module is in working mode, it will output two analog values,

representing two directions. Compared to a normal joystick, its output values are restricted to a smaller range (i.e. 200~800), only when being pressed that the X value will be set to 1023 and the MCU can detect the action of pressing.

[Get One Now !\[\]\(99f58673407353e96a019fbca558fd72_img.jpg\)](#)

[<https://www.seeedstudio.com/Grove-Thumb-Joystick-p-935.html>]

Version

Product Version	Changes	Released Date
Grove - Thumb Joystick V1.1	Initial	Oct 2016

Specifications


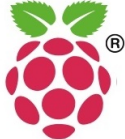
Item	Min	Typical	Max	Unit
Working Voltage	4.75	5.0	5.25	V
Output Analog Value (X coordinate)	206	516	798	\
Output Analog Value (Y coordinate)	203	507	797	\



Tip

More details about Grove modules please refer to [Grove System](https://wiki.seeedstudio.com/Grove_System/) [https://wiki.seeedstudio.com/Grove_System/]

Platforms Supported

Arduino	Raspberry Pi		
			



**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.

Getting Started

**Note**

If this is the first time you work with Arduino, we firmly 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.

Play With Arduino

Demonstration

The Grove - Thumb Joystick is an analog device that outputs analog signal ranging from 0 to 1023. That requires us to use the analog port of Arduino to take the readings.

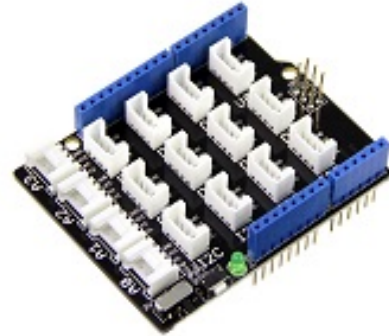
Hardware

- **Step 1.** Prepare the below stuffs:

Seeeduino V4.2



Base Shield



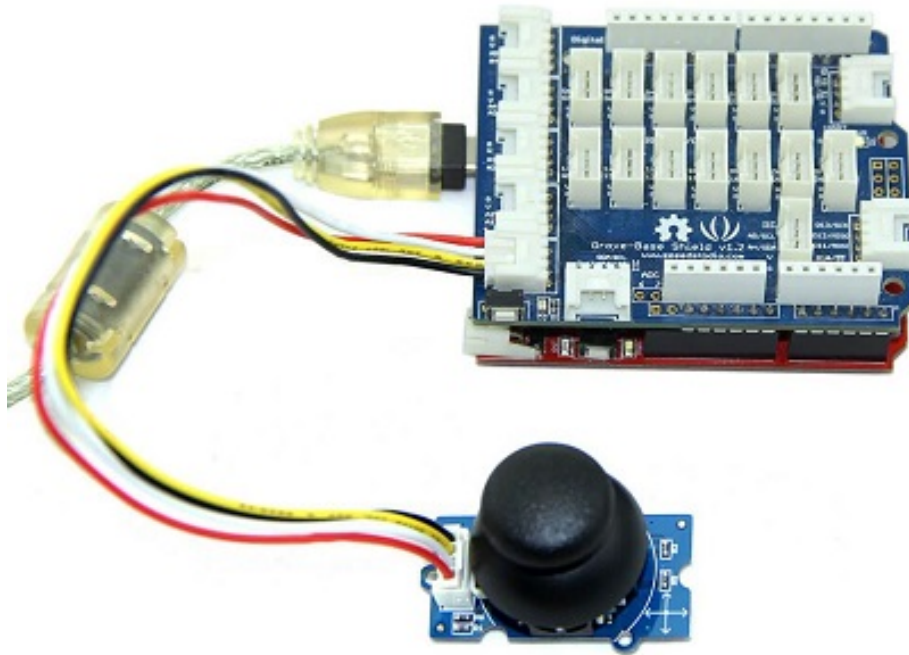
[Get One Now](#)

[<https://www.seeedstudio.com/Seeeduino-V4.2-p-2517.html>]

[Get One Now](#)

[<https://www.seeedstudio.com/Base-Shield-V2-p-1378.html>]

- **Step 2.** Connect the module to the **A0/A1** of Grove - Base Shield by using the 4-pin grove cable.
- **Step 3.** Plug Grove - Base Shield into Seeeduino.
- **Step 4.** Connect Seeeduino to PC via a USB cable.

**Note**

If we don't have Grove Base Shield, We also can directly connect Grove-Thumb Joystick to Seeduino as below.

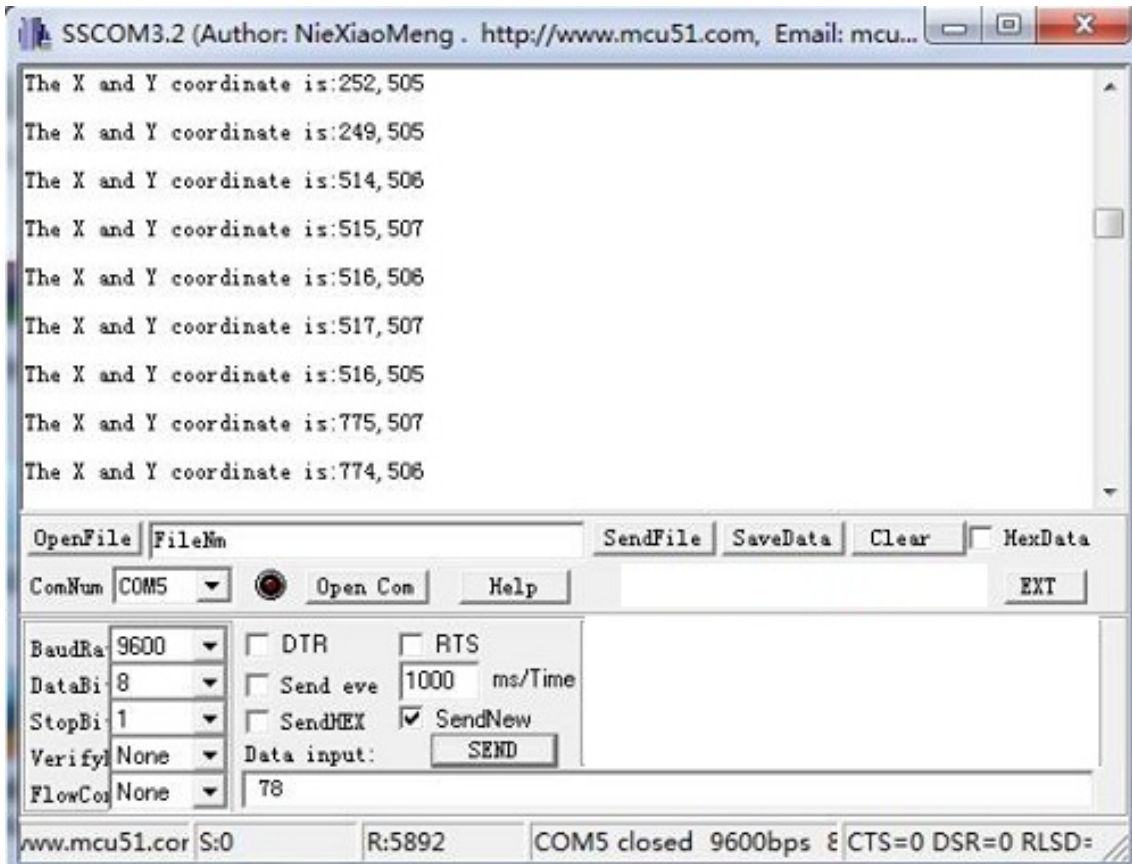
Seeduino	Grove - Thumb Joystick
5V	Red
GND	Black
A1	White
A0	Yellow

Software

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

```
1  /*
2   Thumb Joystick demo v1.0
3   by:https://www.seeedstudio.com
4   connect the module to A0&A1 for using;
5  */
6
7  void setup()
8  {
9      Serial.begin(9600);
10 }
11
12 void loop()
13 {
14     int sensorValue1 = analogRead(A0);
15     int sensorValue2 = analogRead(A1);
16
17     Serial.print("The X and Y coordinate is:");
18     Serial.print(sensorValue1, DEC);
19     Serial.print(",");
20     Serial.println(sensorValue2, DEC);
21     Serial.println(" ");
22     delay(200);
23 }
```

- **Step 2.** You can check the values of the output analog signals by opening the Serial Monitor.



The output value from the analog port of Arduino can be converted to the corresponding resistance by using the formula: $R=(float)(1023-sensorValue)*10/sensorValue$.

Play with Codecraft

Hardware

Step 1. Connect a Grove - Thumb Joystick to port A0 of a Base Shield.

Step 2. Plug the Base Shield to your Seeduino/Arduino.

Step 3. Link Seeduino/Arduino to your PC via an USB cable.

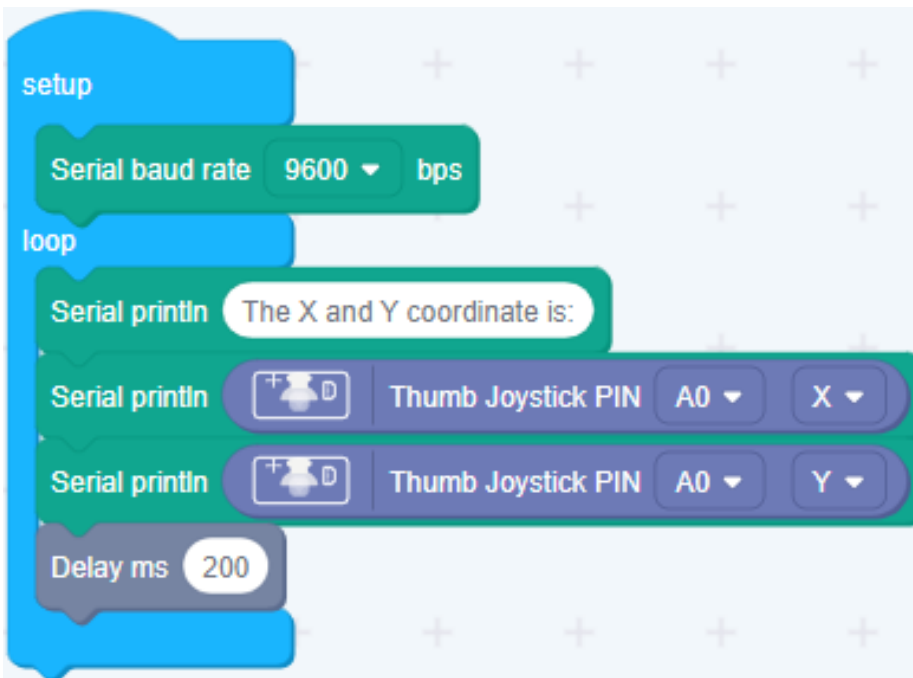
Software

Step 1. Open [Codecraft](https://ide.chmakered.com/) [https://ide.chmakered.com/], add Arduino support, and drag a main procedure to working area.

**Note**

If this is your first time using Codecraft, see also [Guide for Codecraft using Arduino](https://wiki.seeedstudio.com/Guide_for_Codecraft_using_Arduino/) [https://wiki.seeedstudio.com/Guide_for_Codecraft_using_Arduino/].

Step 2. Drag blocks as picture below or open the cdc file which can be downloaded at the end of this page.



Upload the program to your Arduino/Seeeduino.


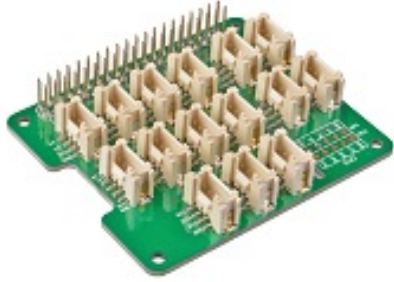
**Success**

When the code finishes uploaded, you will see the coordinate of X and Y displayed in the Serial Monitor.

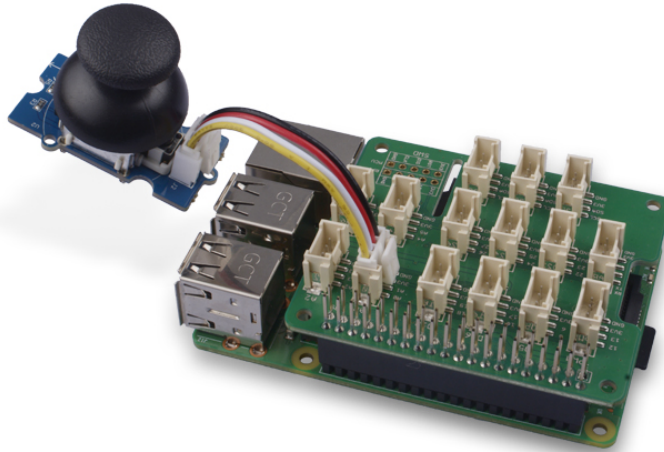
Play With Raspberry Pi (With Grove Base Hat for Raspberry Pi)

Hardware

- **Step 1.** Things used in this project:

Raspberry pi	Grove Base Hat for RasPi
	
<p>Get ONE Now [https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html]</p>	<p>Get ONE Now [https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-p-3186.html]</p>

- **Step 2.** Plug the Grove Base Hat into Raspberry.
- **Step 3.** Connect the Thumb Joystick to port A0 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 the thumb joystick to **any Analog Port** but make sure you change the command with the corresponding port number.

Software



Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- **Step 1.** Follow [Setting Software](https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation) [https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation] to configure the development environment.
- **Step 2.** Download the source file by cloning the grove.py library.

```
1 cd ~
2 git clone https://github.com/Seeed-Studio/grove.py
```

- **Step 3.** Excute below commands to run the code.

```
1 cd grove.py/grove
2 python3 grove_thumb_joystick.py 0
```



Note

you can excute the program with ++python grove_thumb_joystick.py pin++, where pin could be one of {0, 2, 4, 6} in the ADC group and connect the device to the corresponding slot {A0, A2, A4, A6}.

Following is the grove_thumb_joystick.py code.

```
1 import math
2 import sys
3 import time
4 from grove.adc import ADC
5
6
7 class GroveThumbJoystick:
8
9     def __init__(self, channelX, channelY):
10         self.channelX = channelX
11         self.channelY = channelY
12         self.adc = ADC()
13
14     @property
15     def value(self):
16         return self.adc.read(self.channelX), self.adc.re
17
18 Grove = GroveThumbJoystick
19
20
21 def main():
```

```
22     from grove.helper import SlotHelper
23     sh = SlotHelper(SlotHelper.ADC)
24     pin = sh.argv2pin()
25
26     sensor = GroveThumbJoystick(int(pin), int(pin + 1))
27
28     while True:
29         x, y = sensor.value
30         if x > 900:
31             print('Joystick Pressed')
32             print("X, Y = {0} {1}".format(x, y))
33             time.sleep(.2)
34
35 if __name__ == '__main__':
36     main()
```



Success

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

```
1 pi@raspberrypi:~/grove.py/grove $ python3 grove_thumb_jo
2 Hat Name = 'Grove Base Hat RPi'
3 X, Y = 506 484
4 X, Y = 484 484
5 X, Y = 506 484
6 X, Y = 506 487
7 Joystick Pressed
8 X, Y = 999 485
9 X, Y = 310 736
10 X, Y = 681 484
11 Joystick Pressed
12 X, Y = 999 277
13 Joystick Pressed
14 X, Y = 999 487
15 X, Y = 506 484
16 X, Y = 501 486
17 X, Y = 509 484
18 X, Y = 511 486
19 X, Y = 510 485
```

```
20 ^CTraceback (most recent call last):
21   File "grove_thumb_joystick.py", line 69, in <module>
22     main()
23   File "grove_thumb_joystick.py", line 66, in main
24     time.sleep(.2)
25 KeyboardInterrupt
```

You can quit this program by simply press `Ctrl + C`.





Notice

You may have noticed that for the analog port, the silkscreen pin number is something like **A1**, **A0**, however in the command we use parameter **0** and **1**, just the same as digital port. So please make sure you plug the module into the correct port, otherwise there may be pin conflicts.

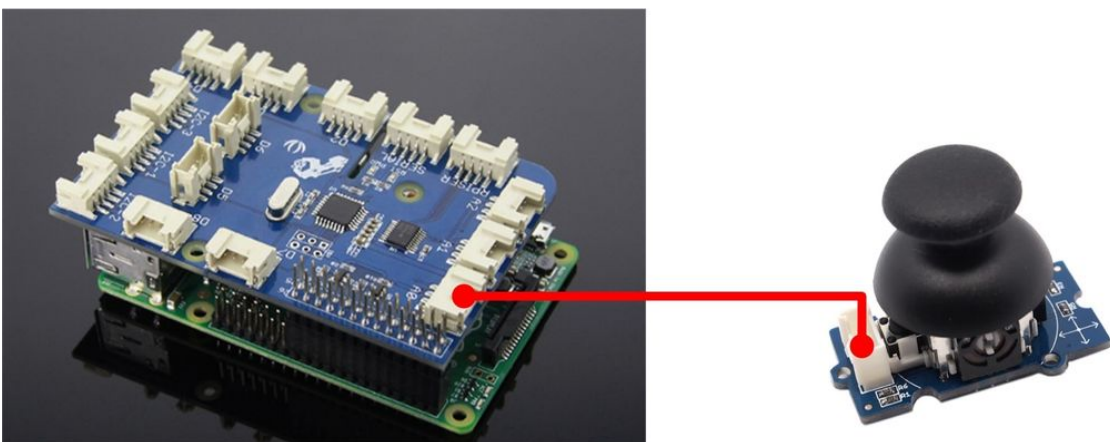
Play With Raspberry Pi (with GrovePi_Plus)

Hardware

- **Step 1.** Prepare the below stuffs:

Raspberry pi	GrovePi_Plus
	
<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>

- **Step 2.** Plug the GrovePi_Plus into Raspberry.
- **Step 3.** Connect Grove-Thumb Joystick ranger to **A0** port of GrovePi_Plus.
- **Step 4.** Connect the Raspberry to PC through USB cable.



Software



Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- **Step 1.** Navigate to the demos' directory:

```
cd yourpath/GrovePi/Software/Python/
```

- **Step 2.** To see the code

```
nano grove_thumb_joystick.py # "Ctrl+x" to exit #
```

```
1  import time
2  import grovepi
3
4  # Connect the Grove Thumb Joystick to analog port A0
5
6  # GrovePi Port A0 uses Arduino pins 0 and 1
7  # GrovePi Port A1 uses Arduino pins 1 and 2
8  # Don't plug anything into port A1 that uses pin 1
9  # Most Grove sensors only use 3 of their 4 pins, which i.
10 # If the sensor has a pin definition SIG,NC,VCC,GND, the
11
12 # If you wish to connect two joysticks, use ports A0 and
13
14 # Uses two pins - one for the X axis and one for the Y a.
15 # This configuration means you are using port A0
16 xPin = 0
17 yPin = 1
18 grovepi.pinMode(xPin,"INPUT")
19 grovepi.pinMode(yPin,"INPUT")
20
21 # The Grove Thumb Joystick is an analog device that outp
22 # The X and Y axes are two ~10k potentiometers and a mom
23
24 # My joystick produces slightly different results to the
25 # I've listed both here:
```



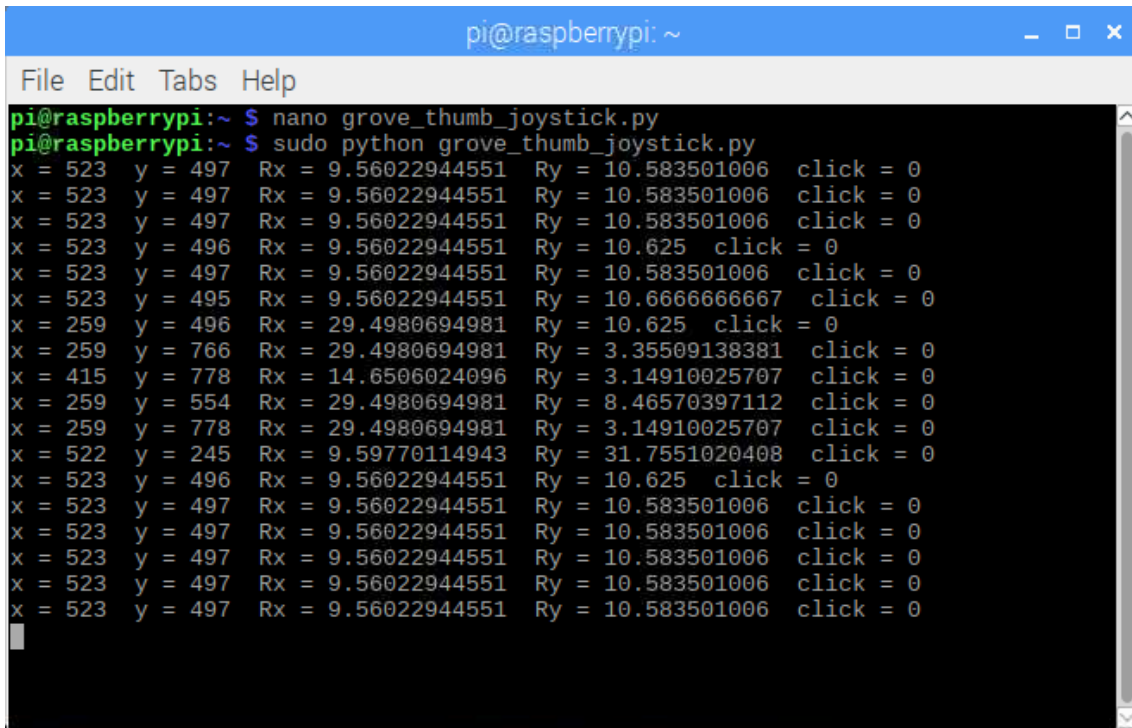
```
26
27 # Specifications
28 #   Min Typ Max Click
29 # X 206 516 798 1023
30 # Y 203 507 797
31
32 # My Joystick
33 #   Min Typ Max Click
34 # X 253 513 766 1020-1023
35 # Y 250 505 769
36 while True:
37     try:
38         # Get X/Y coordinates
39         x = grovepi.analogRead(xPin)
40         y = grovepi.analogRead(yPin)
41
42         # Calculate X/Y resistance
43         Rx = (float)(1023 - x) * 10 / x
44         Ry = (float)(1023 - y) * 10 / y
45
46         # Was a click detected on the X axis?
47         click = 1 if x >= 1020 else 0
48
49         print "x =", x, " y =", y, " Rx =", Rx, " Ry =",
50         time.sleep(.5)
51
52     except IOError:
53         print "Error"
```

- **Step 3.** Run the demo.

```
sudo python3 grove_thumb_joystick.py
```



- **Step 4.** We will see the output display on terminal as below.



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~$ nano grove_thumb_joystick.py  
pi@raspberrypi:~$ sudo python grove_thumb_joystick.py  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 496 Rx = 9.56022944551 Ry = 10.625 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 495 Rx = 9.56022944551 Ry = 10.6666666667 click = 0  
x = 259 y = 496 Rx = 29.4980694981 Ry = 10.625 click = 0  
x = 259 y = 766 Rx = 29.4980694981 Ry = 3.35509138381 click = 0  
x = 415 y = 778 Rx = 14.6506024096 Ry = 3.14910025707 click = 0  
x = 259 y = 554 Rx = 29.4980694981 Ry = 8.46570397112 click = 0  
x = 259 y = 778 Rx = 29.4980694981 Ry = 3.14910025707 click = 0  
x = 522 y = 245 Rx = 9.59770114943 Ry = 31.7551020408 click = 0  
x = 523 y = 496 Rx = 9.56022944551 Ry = 10.625 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0  
x = 523 y = 497 Rx = 9.56022944551 Ry = 10.583501006 click = 0
```

|

Schematic Online Viewer



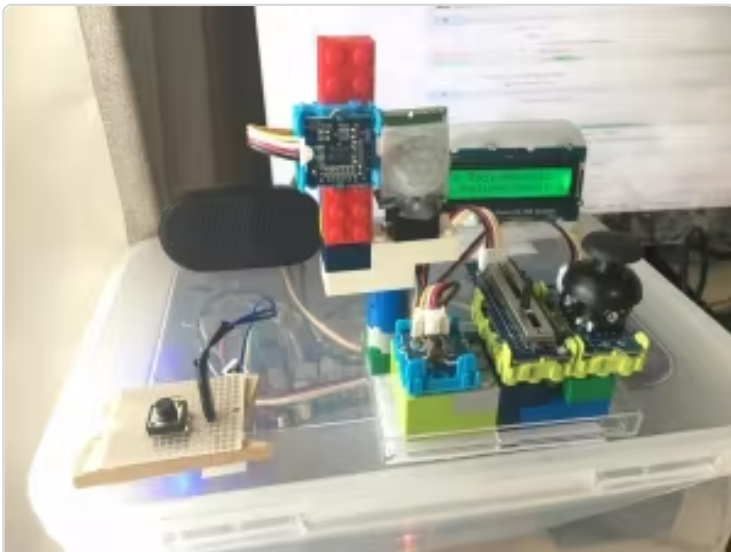
Resources

- **[Eagle]** [Grove-Thumb Joystick Schematic](https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Eagle_Design_Files.zip)
[https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Eagle_Design_Files.zip]
- **[Datasheet]** [Analog Joystick Datasheet](https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Analog_Joystick_Datasheet.jpg)
[https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Analog_Joystick_Datasheet.jpg]

- **[PDF]** [Joystick Schematic PDF File](https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Joystick.pdf)
[https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Joystick.pdf]
- **[Codecraft]** [CDC File](https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Grove_Thumb_Joystick_CDC_File.zip) [https://files.seeedstudio.com/wiki/Grove-Thumb_Joystick/res/Grove_Thumb_Joystick_CDC_File.zip]

Projects

Raspberry pi music server: A first step to Raspberry Pi project.

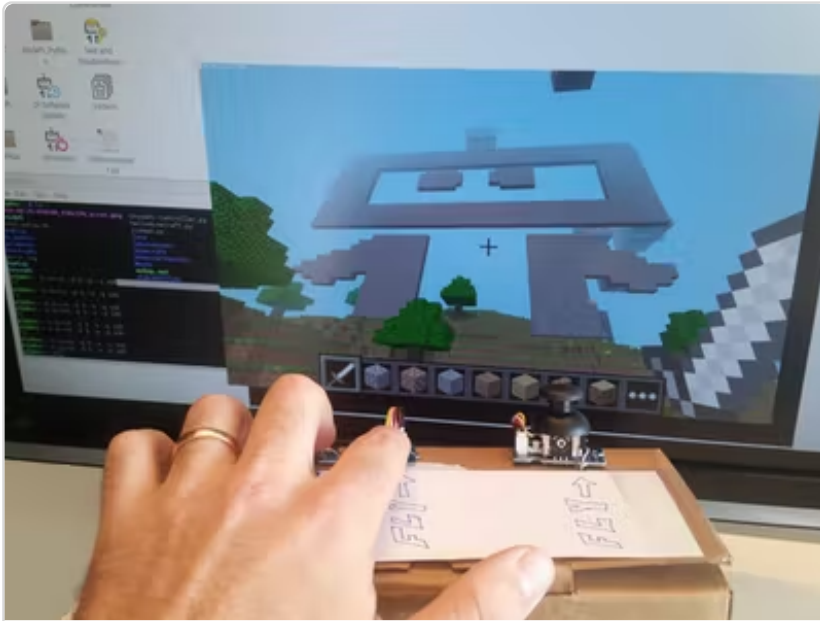


(<https://www.hackster.io/kishima7/raspberry-pi-music-server-f5a0ae>)

Raspberry pi music server

(<https://www.hackster.io/kishima7/raspberry-pi-music-server-f5a0ae>)

Build a Custom Minecraft Controller: Build a Custom Minecraft Controller With the GrovePi.



(<https://www.hackster.io/dexterindustries/build-a-custom-minecraft-controller-d55d9c>)

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]