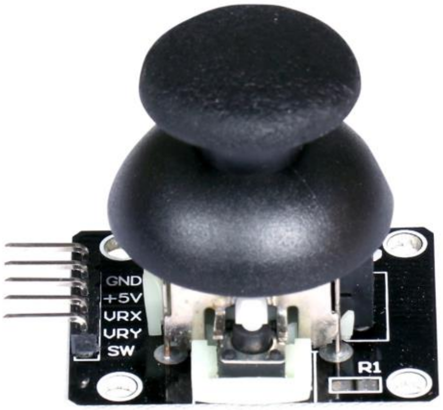
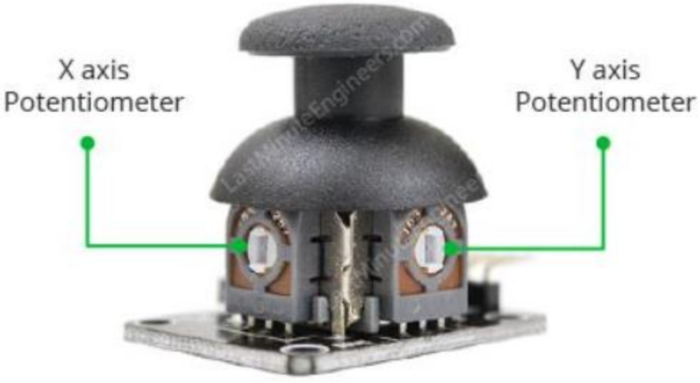


Description:

ADIY Joystick is a self-centering spring-loaded, which means that when you release it, it will center itself. It also has a comfortable cup-type knob/cap that feels like a thumb-stick.

The basic idea behind a joystick is to convert the stick’s position on two axes — the X-axis (left to right) and the Y-axis (up and down) — into an electrical signal that a microcontroller can process. This is accomplished by incorporating two 5K potentiometers (one for each axis) connected with a gymbal mechanism that separates “horizontal” and “vertical” movements.



The two gray boxes on either side of the joystick are the potentiometers. If you move the joystick while keeping an eye on the potentiometer, you’ll notice that each potentiometer only detects movement in one direction.

Features:

1. 2.54mm pin interface leads
2. Long service life and stable performance
3. Standard interface and electronic building blocks
4. Widely use in Arduino DIY projects
5. Cross rocker as a two-way 10K resistor, with the rocker in a different direction.

Specifications:

- Operating Voltage (VDC):5
- Height (mm): 32
- Weight (gm):10
- PCB Size (mm): 34 x 32

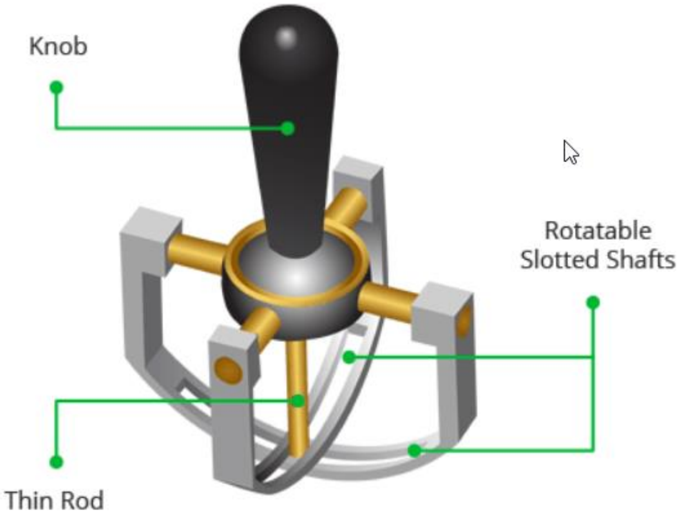
How it works:

It is truly remarkable how a joystick can translate every tiny motion of your fingertips. This is made possible by the joystick’s design, which consists of two potentiometers and a gimbal mechanism.

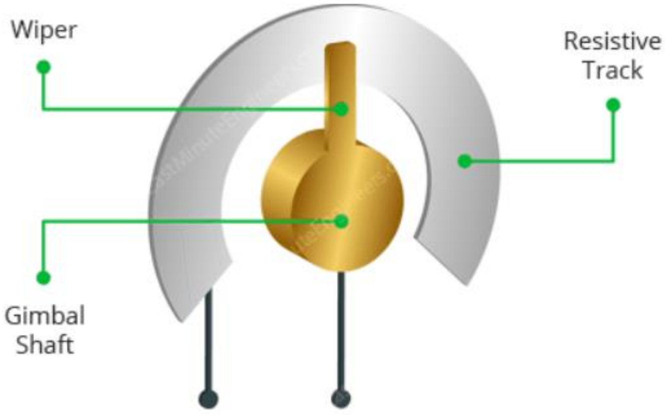
Gimbal Mechanism

When you move the joystick, a thin rod that sits between two rotatable slotted shafts (Gimbal) moves. One of the shafts allows movement along the X-axis (left and right), while the other allows movement along the Y-axis (up and down).

When you move the joystick back and forth, the Y-axis shaft pivots. When you move it left to right, the X-axis shaft pivots. And when you move it diagonally, both shafts pivot.



Each shaft is connected to a potentiometer so that moving the shaft rotates the corresponding potentiometer's wiper. In other words, pushing the knob all the way forward will cause the potentiometer wiper to move to one end of the resistive track, and pulling it back will cause it to move to the opposite end.

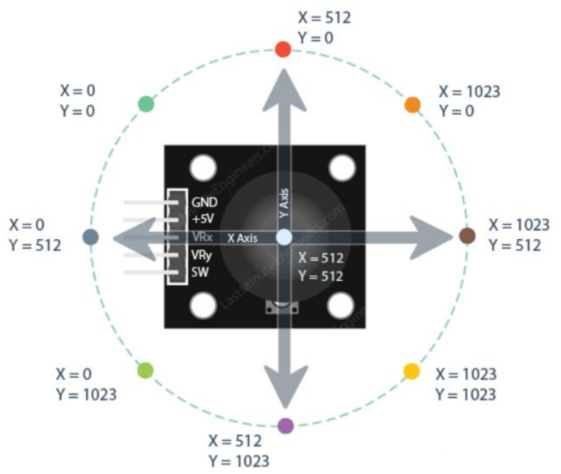


The joystick outputs an analog signal whose voltage varies between 0 and 5V. When you move the joystick along the X axis from one extreme to the other, the X output changes from 0 to 5V, and the same thing happens when you move it along the Y axis. And, when the joystick is centered (rest position), the output voltage is approximately half of VCC, or 2.5V.

This output voltage can be fed to an ADC on a microcontroller to determine the physical position of the joystick.

Because the Arduino board has a 10-bit ADC resolution, the values on each analog channel (axis) can range from 0 to 1023. Therefore, when the joystick is moved from one extreme to the other, it will read a value between 0 and 1023 for the corresponding channel. When the joystick is centered, the vertical and horizontal channels will both read 512.

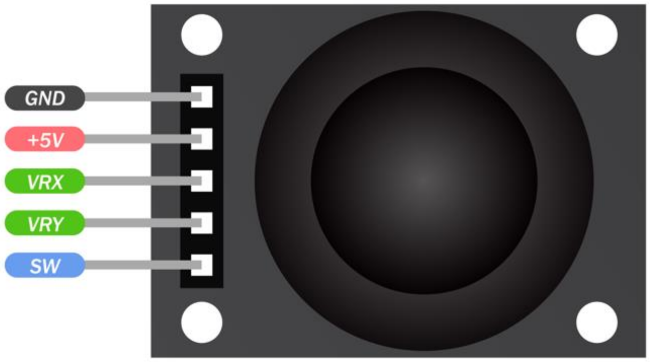
The figure below depicts the X and Y axes as well as how the outputs will respond when the joystick is moved in different directions.



Pin Configuration:

GND: Ground

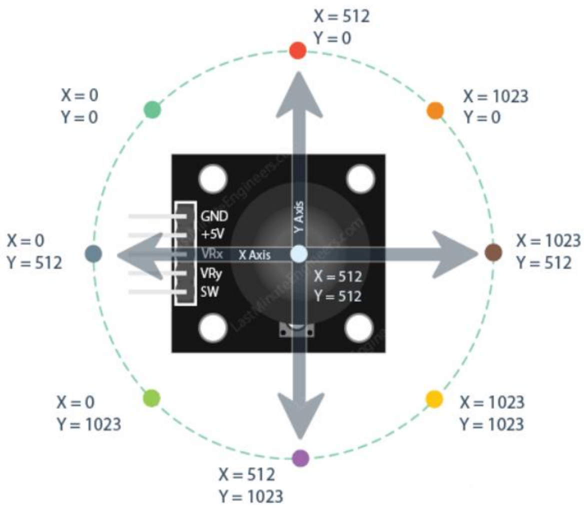
+5V: provides power to the module. Connect this to your positive supply (usually 5V or 3.3V depending on your logic levels).



VRX: is the horizontal output voltage. Moving the joystick from left to right causes the output voltage to change from 0 to VCC. The joystick will read approximately half of VCC when it is centered (rest position).

VRY: is the vertical output voltage. Moving the joystick up and down causes the output voltage to change from 0 to VCC. The joystick will read approximately half of VCC when it is centered (rest position).

SW: is the output from the pushbutton switch. By default, the switch output is floating. To read the switch, a pull-up resistor is required so that when the joystick knob is pressed, the switch output becomes LOW otherwise it remains HIGH. Keep in mind that the input pin to which the switch is connected must have the internal pull-up enabled, or an external pull-up resistor must be connected.



Applications:

- As square wave signal generator which generates a square wave signal
- To provide a signal to the stepping motor driver
- Adjustable pulse generation for chip use
- Produce variable pulse signal, the control-related circuit (PWM dimming, speed)