

Rocker print data

1. Learning target

- 1.1 In this course, we will learn how to use pins of the Raspberry Pi Pico board.
- 1.2 How to use drive servo by PWM.

2. Preparation

Raspberry Pi Pico board *1
Pico sensor expansion board *1
PC *1
USB data cable *1
Rocker module*1
Male-to-male DuPont line *5

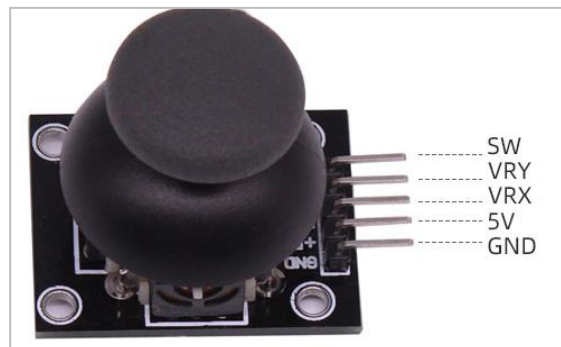
Working principle:

The rocker module can be regarded as a potentiometer with two channels, which can output the analog values of the X and Y axes. When moving to the left, the smaller the X value of the output. When moving to the right, the larger the X value of the output. When moving to the up, the larger the Y value of the output. When moving to the down, the smaller the Y value of the output.

The output analog range of the X and Y axes is from 0 to 1023.

Output signal:

The dual-axis XY joystick module has two analog outputs and one digital output interface. The joystick is remotely controlled from left to right, the output value corresponds to the offset X/Y, the button indicates the direction of the Z axis, press it for low level, and release it for high level.



Module parameters:

Name: Dual-axis XY joystick module

Input voltage: DC 3.3V to 5V (note that the PICO board must be connected to 3.3V voltage)

Output signal: X/Y dual-axis analog quantity + Z-axis digital quantity

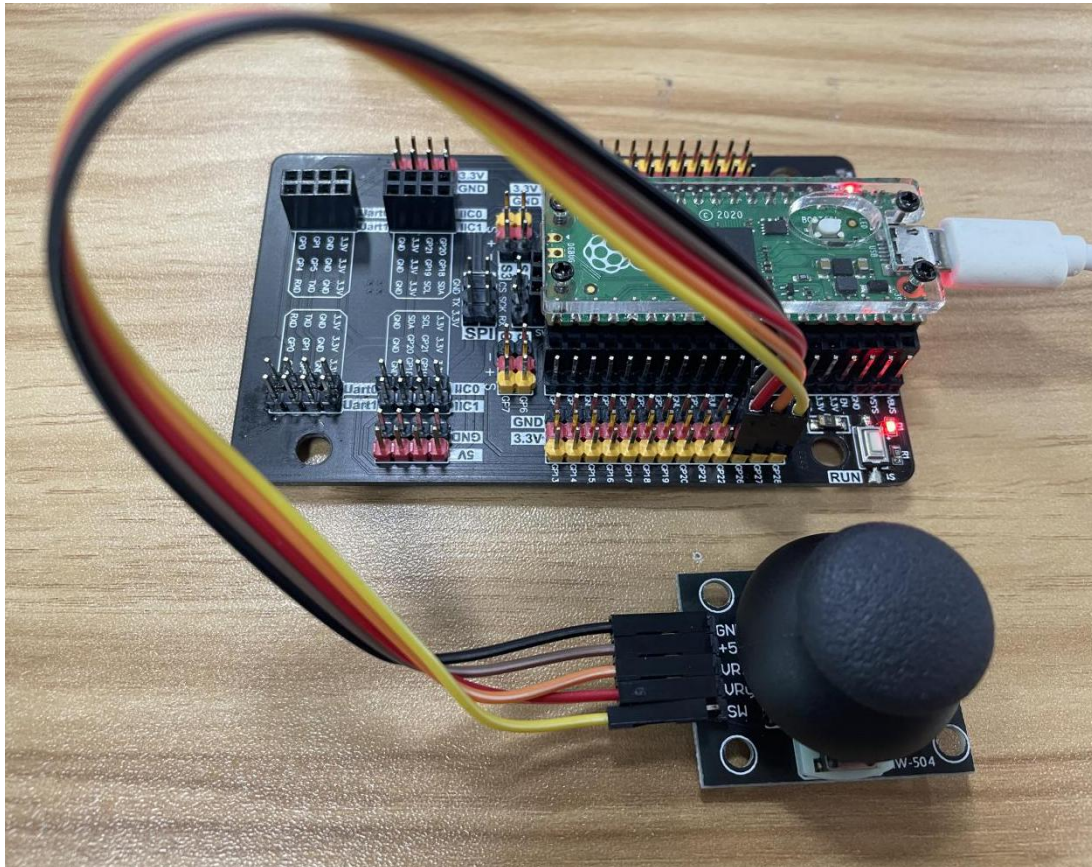
Scope of application: suitable for the movement of control point coordinates in the plane, control of dual-degree-of-freedom steering gear, etc.

Product features: exquisite appearance, superior rocker feel, simple operation, sensitive response, long service life

Size: 34mm*26mm

3. About wiring

Rocker module	Pico sensor expansion board
SW	GP28
VRX	GP27
VRX	GP26
GND	GND
+5V	3.3V



4. About code

Thonny programming

About how to using ThonnyIDE, please check the tutorials in 【2.Development environment】

```

from machine import Pin, ADC
import utime
# Initialize the joystick module (ADC function)
rocker_x = ADC(27)
rocker_y = ADC(26)
button = Pin(28, Pin.IN, Pin.PULL_UP)

# Read the value of the X axis and return [0, 255]
def read_x():
    value = int(rocker_x.read_u16()) * 256 / 65536

```

```

    return value

# Read the value of Y axis and return [0, 255]
def read_y():
    value = int(rocker_y.read_u16() * 256 / 65536)
    return value


# Read the state of the button, press to return to True, release to return to False
def btn_state():
    press = False
    if button.value() == 0:
        press = True
    return press

# Print the current value of the potentiometer cyclically, value=[0, 255]
while True:
    value_x = read_x()
    value_y = read_y()
    state = btn_state()
    print("x:%d, y:%d, press:%s" % (value_x, value_y, state))
    utime.sleep(.1)

```

5. Phenomenon

Click the green run button  of Thonny IDE to start running the program. Click the red stop

button  to stop the program. When the program is running, the shell window under Thonny IDE will print the current analog value of the rocker.

