2023

5. Move the motor with a joystick

Project number: **2021-1-FR01-KA220-SCH-000031617**

Co-funded by the European Union

SCRAPY Partnership

31/05/2023

Table of Contents

Experiment 5: Move the motor with a joystick

Short Description

With this experiment students will be able to create a moving pointer that is controlled by a joystick module.

Extended Description

In this experiment we will control a servo motor using a joystick and a Raspberry Pi Pico.

The joystick we are using is analog and it provides more accurate readings than the simple directional joysticks.

The joystick will allow the students to rotate the servo motor in a defined angle, and the program will allow them to understand the amplitude of angles and the orientation.

Objectives:

In terms of knowledge, students will:

1. Understand how a circuit works.
2. Identify the components of the circuit.
3. Understand the measure of angles.
4. Understand the orientation of an angle and the effect it has on the angle measured.

Materials to be used:

- 1 x Raspberry Pi Pico
- 1 x Pico breadboard kit
- 1 x Full-size breadboard
- 1 x SG90 servo motor
- 1 x joystick module
- Jumper wires

Steps to be followed:

Let's create a command to move a servo motor and know more about angles and rotation.

For this, let's star by

1. Connect the joystick module to the Raspberry Pi Pico board using connection wires.
2. Connect the SG90 servo motor to the Raspberry Pi Pico board.

3. Write a Python program to control the Raspberry Pi Pico board and use the joystick module to control the servo motor.
4. Test the joystick module by moving it in different directions and seeing what happens to the servo motor.

Raspberry Pi Pico Board:

- GP12: Signal of the SG90 servo motor
- GP26: Vert Pin of the Joystick module
- GP16: SW pin of the Joystick module
- GND: Ground pin of the board

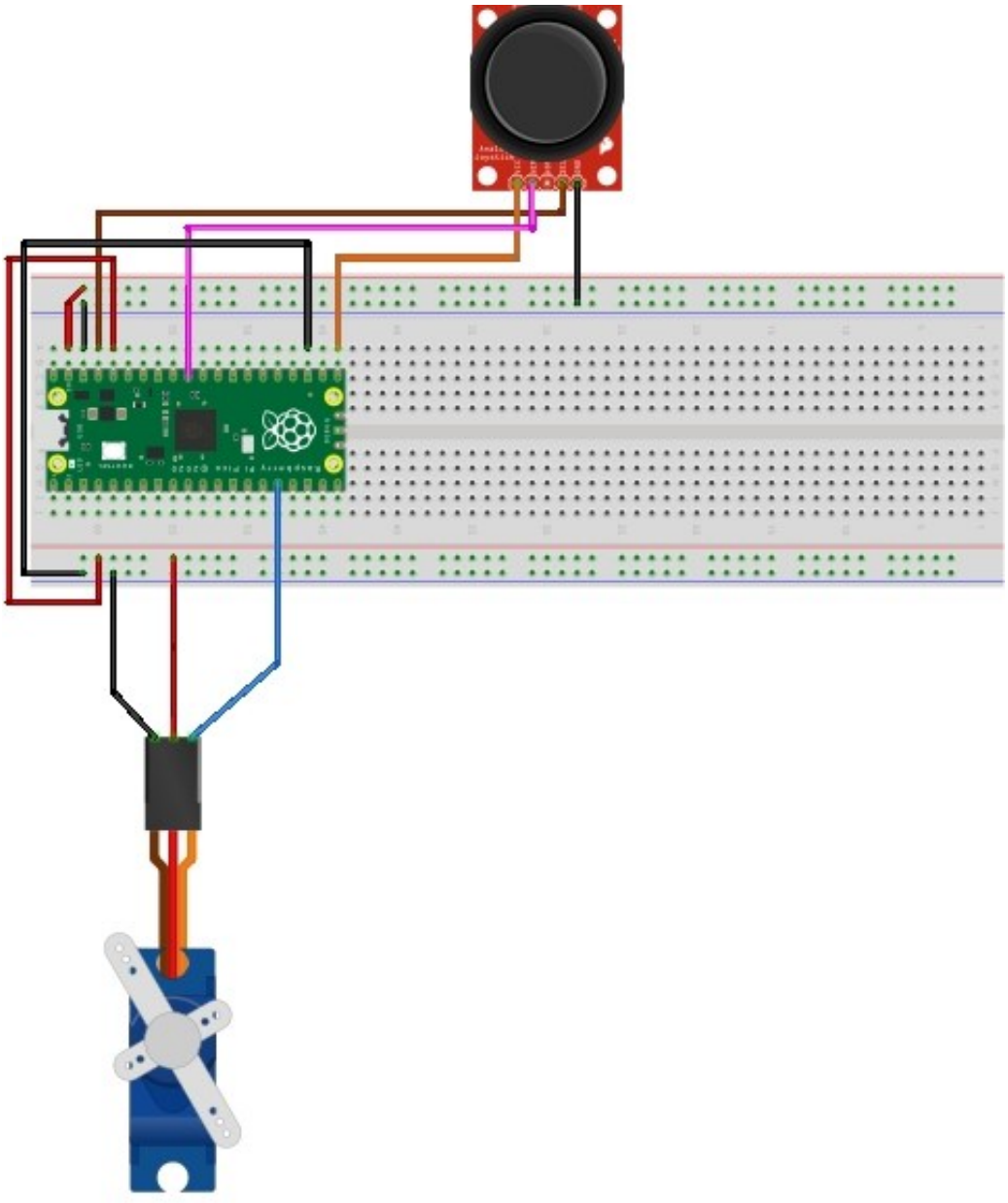
SG90 servo motor:

- Power: Connect to 3V3 power source
- GND: Connect to GND of Raspberry Pi Pico board
- Signal: Connect to GP12 of Raspberry Pi Pico board

Joystick module:

- VCC: Connect to 3V3_EN power source
- GND: Connect to GND of Raspberry Pi Pico board
- Horiz: Not connected
- Vert: Connect to GP26 of Raspberry Pi Pico board
- SW: Connect to GP16 of Raspberry Pi Pico board

Wiring diagram



Code

```
import machine
import utime

# Define ADC pin numbers
X_AXIS_PIN = 26

# Define servo pin number
SERVO_PIN = 13

# Servo angle limits
SERVO_MIN_ANGLE = 0
SERVO_MAX_ANGLE = 45

# Create ADC instances
x_axis_adc = machine.ADC(machine.Pin(X_AXIS_PIN))

# Create PWM instance for servo control
servo_pwm = machine.PWM(machine.Pin(SERVO_PIN))

# Configure PWM frequency and duty cycle range for servo control
servo_pwm.freq(30)
servo_pwm.duty_u16(0)

# Function to map a value from one range to another
def map_value(value, in_min, in_max, out_min, out_max):
    return int((value - in_min) * (out_max - out_min) / (in_max -
in_min) + out_min)

# Main loop
while True:
    # Read joystick values
    x_val = x_axis_adc.read_u16()
    # Map joystick values to servo angle
    angle_x = map_value(x_val, 0, 65535, SERVO_MIN_ANGLE,
SERVO_MAX_ANGLE)
    # Set servo position
    servo_pwm.duty_u16(map_value(angle_x, SERVO_MIN_ANGLE,
SERVO_MAX_ANGLE, 50, 5000))

    # Delay for stability
    utime.sleep_ms(10)
```

Conclusion

If needed, the teacher can use the program and the servo motor as a game, attaching a colored roulette to the servo motor and challenging the students to rotate the pointer to point to a certain color.