Coding part

Connection Explanation

The program controls **3 servo motors** using a **joystick** and a **button**.

- **Servo 1** → Moves the arm **left and right** (X-axis). Joystick works like Pushing it **left or right** moves Servo 1.

- **Servo 2** → Moves the arm **up and down** (Y-axis). Joystick works like Pushing it **up or down** moves Servo 2.
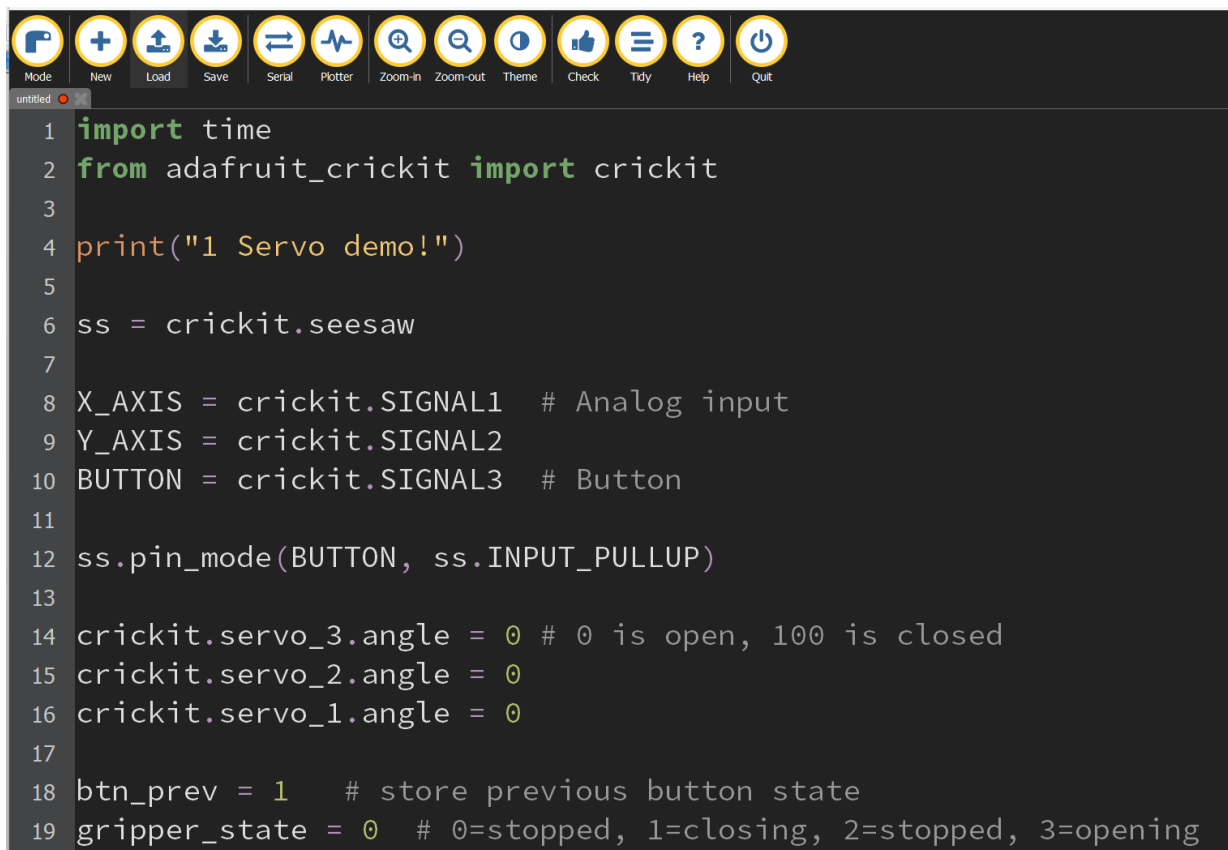
Servo 3 → Works as the **gripper hand** (open and close).

The **button** changes the gripper's state:

- First press → gripper **closes**.

- Second press → gripper **stops**.

- Third press → gripper **opens**.

- Fourth press → gripper **stops again** (cycle repeats).

Coding part

Here is the code you can see if you face any doubt while doing this task.

```python
import time
from adafruit_crickit import crickit

print("1 Servo demo!")

ss = crickit.seesaw

X_AXIS = crickit.SIGNAL1   # Analog input
Y_AXIS = crickit.SIGNAL2
BUTTON = crickit.SIGNAL3   # Button

ss.pin_mode(BUTTON, ss.INPUT_PULLUP)

crickit.servo_3.angle = 0 # 0 is open, 100 is closed
crickit.servo_2.angle = 0
crickit.servo_1.angle = 0

btn_prev = 1    # store previous button state
gripper_state = 0  # 0=stopped, 1=closing, 2=stopped, 3=opening
```

```python
gprangle = 0
xservo = 0
yservo = 0

while True:
    y = ss.analog_read(Y_AXIS)
    x = ss.analog_read(X_AXIS)
    btn_val = ss.analog_read(BUTTON)

    # treat button as digital (pressed < 50)
    if btn_val < 50:
        btn = 0  # pressed
    else:
        btn = 1  # released

    # Detect new press (edge trigger)
    if btn == 0 and btn_prev == 1:
        # cycle through 0→1→2→3→0
        gripper_state = (gripper_state + 1) % 4
        print("New state:", gripper_state)

    # Move gripper depending on state
    if gripper_state == 1:    # closing
        print("closing gripper")

    # Limit angles
    if gprangle < 0:
        gprangle = 0
    if gprangle > 100:
        gprangle = 100

    crickit.servo_3.angle = gprangle

    # X joystick
    if x < 20:
        xservo += 10
    if x > 1000:
        xservo -= 10
    xservo = max(0, min(180, xservo))
    crickit.servo_1.angle = xservo

    # Y joystick
    if y < 20:
        yservo += 10
    if y > 1000:
        yservo -= 10
    yservo = max(0, min(180, yservo))
    crickit.servo_2.angle = yservo

    btn_prev = btn
    time.sleep(0.1)
```

You can adjust the value of checking condition and modify and see the output.