Activity 3: Digital Input

Below is a step-by-step guide to program and wire a Raspberry Pi Pico for a push button on pin 18 and a slide switch on pin 15 using MicroPython. I'll include comments in the code to explain each step:

Wiring:

1. Push Button Wiring:

- Connect the 1.r leg of the push button to pin 18 (GP18) on the Raspberry Pi Pico.
- Connect the 2.r leg of the push button to the GND (ground) pin on the Raspberry Pi Pico.

2. Slide Switch Wiring:

- Connect the middle terminal of the slide switch to pin 15 (GP15) on the Raspberry Pi Pico.

- Connect the one terminal of the slide switch to the 3V3 (3.3V) pin on the Raspberry Pi Pico and the other terminal to GND (ground). The 2 terminals are interchangeable.

Programming:

3. Install Thonny IDE:

- If you haven't already, install the Thonny IDE on your computer.

4. Connect Raspberry Pi Pico:

- Connect your Raspberry Pi Pico to your computer using a USB cable.

5. Install MicroPython Firmware:

- Make sure your Raspberry Pi Pico has the MicroPython firmware installed. If not, you can download the firmware from the [official MicroPython website](https://micropython.org/download/rp2-pico/) and follow the installation instructions.

6. Open Thonny:

- Open the Thonny IDE on your computer.

7. Create a New Script:

- Create a new Python script by selecting "File" > "New" in Thonny.

8. Write the Code for Push Button and Slide Switch:

- Copy the following code into the script:

```
from machine import Pin
import utime

# Define the GPIO pins for the push button and slide switch
button_pin = Pin(18, Pin.IN, Pin.PULL_UP)  # Pin 18 for the push button with pull-up resistor
switch_pin = Pin(15, Pin.IN)  # Pin 15 for the slide switch


while True:
    # Check the state of the push button
    print("Push button:")
    print(button_pin.value())

    # Check the state of the slide switch
    print("Slide Switch:")
    print(switch_pin.value())
    # Delay for a short duration to avoid continuous printing
    utime.sleep(0.5)

```

9. Save the Script:

- Save the script with a .py extension, for example, `button_switch.py`.

10. Run the Script:

- Click on the "Run" button (green triangle) in Thonny to execute the script.

11. Observe the Output:

- Observe the output in the Thonny console. It should print messages indicating the state of the push button and slide switch.

12. Understand the Code:

- Review the comments in the code to understand each step. The code continuously checks the state of the push button and slide switch and prints messages accordingly.

13. Exercise:

- Add another slide switch and push button to the system and display their output.

This step-by-step guide should help you wire and program your Raspberry Pi Pico to interact with a push button on pin 18 and a slide switch on pin 15 using MicroPython.