Activity 6: Digital Input

Below is a step-by-step guide to wire and program a Raspberry Pi Pico with the provided code using the MPU6050 sensor and MicroPython.

Materials Needed:

- Raspberry Pi Pico

- MPU6050 sensor

- Breadboard and jumper wires

Wiring:

1. Connect the SDA pin of the MPU6050 to GPIO pin 2 on the Raspberry Pi Pico.

2. Connect the SCL pin of the MPU6050 to GPIO pin 3 on the Raspberry Pi Pico.

3. Make sure to connect the VCC and GND pins of the MPU6050 to the appropriate power and ground rails on the breadboard.

4. Double-check the wiring to ensure everything is connected correctly.

Programming:

1. Install Required Libraries:

- Make sure you have the necessary libraries installed on your Raspberry Pi Pico. The code relies on the `imu.py` module, so ensure it's available.

Save the imu.py on the Pico from:

https://github.com/ump-stemlab/stemcube/blob/f682b5db118ba9b00c7c2445da3a1601bb206530/imu.py

Save the vector3d.py on the Pico from:

https://github.com/ump-stemlab/stemcube/blob/f682b5db118ba9b00c7c2445da3a1601bb206530/vector3d.py

-Make sure the name saved is exactly the same.

2. Create a New MicroPython Script:

- Open Thonny or your preferred MicroPython editor.

3. Copy the Code:

- Copy the provided code into your new MicroPython script.

```
from imu import MPU6050
from time import sleep
from machine import Pin, I2C

# Initialize I2C and MPU6050
i2c = I2C( 1, sda=Pin(2), scl=Pin(3), freq=400000) # Pico
imu = MPU6050(i2c)

try:
# Open or create a file for writing data
    file = open("data.txt", "a") # "w" to overwrite, "a" to append

except OSError:
# If file opening fails, create a new file
    file = open("data.txt", "w") # "w" to overwrite, "a" to append

count = 0

while True:
    # Read accelerometer, gyroscope, and temperature data
    ax = round(imu.accel.x, 2)
    ay = round(imu.accel.y, 2)
    az = round(imu.accel.z, 2)
    gx = round(imu.gyro.x)
    gy = round(imu.gyro.y)
    gz = round(imu.gyro.z)
    tem = round(imu.temperature, )

    # Print data to console
    print("ax", ax, "\tay", ay, "\taz", az, "\tgx",gx,"\tgy",gy,"\tgz",gz,"\tTemperature", tem)

    # Write data to file
    file.write(str(count) + "," + str(ax) + ","+ str(ay) + "," + str(az) + ","+ str(gx) + "," + str(gy) + "," + str(gz) + "," + str(tem) + "\n")
    file.flush()
    count += 1

    # Wait for 1 second
    sleep(1)
```

4. Save the Script:

- Save the script with a meaningful name, such as `mpu6050_data_logger.py`.

5. Run the Script:

- Save the script and run it on your Raspberry Pi Pico.

- Open the console to view the accelerometer, gyroscope, and temperature data being printed.

6. Check Data File:

- The script is set to write data to a file named `data.txt`. Check if the file is being created and if data is being appended.

8. Observations:

- Observe the console output and the data file to ensure that the sensor data is being recorded accurately.

Additional Notes:

- Adjust Sampling Rate:
    - You can adjust the sampling rate by modifying the `sleep` duration in the loop.