Names _____

INTRODUCTION

In the previous labs, you interfaced your PicoW using the microcontroller's ADC input, and PWM output. This lab will introduce you the world of external sensors, displays, and controllers. There are hundreds of external boards that expand microcontrollers to measure, control, or display almost anything imaginable. Sensors include sound, like, motion, radiation, air quality, temperature, humidity and more. Display boards range from two lines of 16 characters, to touchscreens, to dual monitors. Controllers include driving motors, valves, heaters, etc. Most of these boards communicate with microcontrollers using digital interfaces.

CHAPTER 10 – DIGITAL COMMUNICATION PROTOCOLS: I2C AND SPI

To use an external sensor board, microcontrollers communicate using digital signal *protocols*. Most of these protocol are *serial*, that is they send data one bit at a time to the board. Some of these protocols commonly used include *USB*, *One-Wire*, *I2C*, and *I2S*. In this lab, based on Chapter 10 in the Getting Started book, we will use I2C and SPI.

1) Display using I2C

You are following the directions in Ch. 11 with a few small changes.

- Unplug the Pico W!
- Use pins **`GP8`** for **`SDA`** and **`GP9`** for **`SCL`**. On the DAC board, they are in the 1x10 connector on the left of the Pico W.
- Follow the directions for plugging in the 128 x 32 OLED display. (It is the long rectangular display.) It is set up to use the I2C protocol.
- The library procedure did not work for me. Instead google *micropython-ssd1306* and click on the GitHub link **`https://github.com/stlehmann/micropython-ssd1306/blob/master/ssd1306.py`** Right above the code window there is a download button. Copy this file to your working directory, then use Thonny to create a folder on the Pico W named **`lib`**. Use Thonny to upload the file to this lib folder.
- Test if you installed it correctly by going to the Pico W python prompt and typing **`import ssd1306`**.

2) Challenge

Before going on to the temperature sensor part, there are more things you can do with the display than write test.

Go to the MicroPython web page
`https://docs.micropython.org/en/latest/esp8266/tutorial/ssd1306.html`

On that page you will see examples of **`contrast`**, **`invert`**, **`pixel`**, **`hline`**, **`vline`**, **`line`**, **`rect`**, and more. Try each method at least once. Remember, use **`fill(0)`** before, and show after to see your results. You can draw multiple things on the display if your don't use **`fill`**.

Finally, write a program that will make a personal logo for you and your partner and show it to the instructor.

3) Display Using SPI

- Unplug the Pico W!
- Use pins **`GP8`** for **`SDA`** and **`GP9`** for **`SCL`**. On the DAC board, they are in the 1x10 connector on the left of the Pico W.

- This display is 128 x 64 and uses the SPI protocol.
- Work through the exercises in the rest of the chapter.

4) Challenge

- **Challenge 1**
- This display has more than one color. Use **`fill_rect`** to completely fill the display. Describe what the display looks like below.
- Write a program to make you logo with the SPI display and show your instructor.

- **Challenge 2**
- Write a program to randomly fill the display with dots. When it is full, start randomly deleting pixels.
- **Hints:**
  - make a list with a list from 0 to the number of pixels minus 1. Use **`list`** function and **`range`** function.
  - The function **`random.choice`** randomly picks an item from an array.
  - **`array.remove(i)`** removes item with value **`i`** from an array.
  - For a pixel **`i`**, the "row" of the pixel is **`i // 128`**, and the column of the pixel is **`i % 12`**.
  - Go through all of the pixels twice, once for each color.

MATERIALS LIST

- One 0.91” 128x32 pixel OLED display
- One 0.96” 128x64 pixel OLED display
- Male to male jumper wires.