Lesson 2 OLED ACTIVITY Worksheet

**OLED screen activity**

Up until this point we have used libraries that are built into the Micro-python programming language. In this activity you will import an external module into Thonny. The ssd1306 screen needs its own set of “drivers” instructions and “protocols” rules to be able to communicate with the Pico.

Thonny has a package manager to simplify this process. Follow these steps to import the package into Thonny.

**Note: make sure your Pico is plugged into your computer's USB port** and you are **running the Micropython editor** for Pico at the **bottom right hand** side of the Thonny editor.

1. From the **Thonny toolbar** locate the **Tools menu**, in the drop down menu select **manage packages**

A screenshot of a computer

Description automatically generated

1. **Type SSD1306** into the **search bar** and press enter or **click the search** **button**

A screenshot of a computer

Description automatically generated

1. Click on the **blue link** in **the results**, a **pop up** will show a **button** where you can **click to install.**
2. Once the **module is installed close** the pop up and you are ready to start the coding.A screenshot of a computer

   Description automatically generated

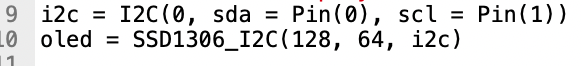
**Coding Activity 1:** install the libraries below paying careful attention to the case of **I2c** and the **ssd1306. Note how they change**

A close up of a text

Description automatically generated

**Next**

line 9 and 10 declare a variable i2c to hold the I2C object and pass in the channel and the PINS for the clock and data connections



That’s the tricky part over!

**Now add the remaining code**

A screenshot of a computer program

Description automatically generated

**Save and run your code.**

***If it does not work as expected check your code and your wiring connections with a friend.***

Now look at the amended code below. Can you predict what will happen when this code is executed? Record your idea here before running the code:

A screenshot of a computer program

Description automatically generated

don't forget to add **oled.show()to the end of your code**

Was the outcome as you predicted? If not, how did it differ?

The rectangle you drew was a simple vector image. Vector images are made of shapes using maths. Now you know how to draw a rectangle, let's combine some shapes to form a logo.

**Before you run the code below** what do you think the logo will look like?

Try and sketch your idea in the box below by reading the parameters of the rectangles and vlines(vertical lines) ?

A screenshot of a computer program

Description automatically generated

don't forget to add **oled.show()to the end of your code**



Now run the code and see how close you were to your prediction. What differed from your sketch?

Can you identify the code that made the sketch vary from the actual image displayed on the screen? **Record your answers here**

Now add the line **oled.invert(1)** before the **oled.show()**

**Run the code and record what happens to the screen**

**Challenge activity:**

Experiment with some of the basic Use the basic OLED functions below such as rotating and object.

Can you create an image of your own design?

Can you use the button from lesson 2 to clear the screen?

**Remember to display anything on the OLED you must use oled.show()** where oled is the name of your screen object

here are some common to help you

oled.fill(0) # fill entire screen with colour=0

oled.pixel(0, 10) # get pixel at x=0, y=10

oled.pixel(0, 10, 1) # set pixel at x=0, y=10 to colour=1

oled.hline(0, 8, 4, 1) # draw horizontal line x=0, y=8, width=4,colour=1

oled.vline(0, 8, 4, 1) # draw vertical line x=0, y=8, height=4, colour=1

oled.line(0, 0, 127, 63, 1) # draw a line from 0, 0 to 127, 63

oled.rect(10, 10, 107, 43, 1) # draw a rectangle outline 10,10 to 117,53,colour=1

oled.fill\_rect(10, 10, 107, 43, 1) # draw a solid rectangle 10 , 10 to 117, 53, colour=1

oled.text('Hello World', 0, 0, 1) # draw some text at x=0, y=0, colour=1

oled.scroll(20, 0) # scroll 20 pixels to the right

**Plenary**

**In your own words define the following two key terms:**

Mono OLED screen

Protocol