Lesson 3 I2C

Teacher Guide

**Teacher guide to I2C**

The I2C (Inter Integrated Circuit) protocol is a widely used communication protocol that facilitates data transfer between microcontrollers and various peripherals, such as sensors, displays, and other integrated circuits. The I2C has the benefit of only needing two pins for communication a serial data connection and a serial clock. If you look on the pic out diagram you will see the Pico has two separate channels for I2C each channel can handle many devices as long as the devices have different address. Referencing to the pin diagram you will see the pins are in pairs and are belong to either channel 0 or 1. We will be using Pin 0&1 which are on channel 0.

**How data is sent to an OLED screen using the I2C protocol:**

1. Initialisation:

The OLED screen is connected to the Raspberry Pi Pico via the SDA and SCL pins. The I2C bus requires pullup resistors on both lines to ensure proper signal integrity.

2. Addressing:

Each I2C device has a unique address. The OLED screens address is specified in the code. When the Raspberry Pi Pico wants to communicate with the OLED screen, it sends a start condition on the I2C bus followed by the 7bit device address and a write bit (0).

3. Data Transmission:

After addressing, the Raspberry Pi Pico can send data to the OLED screen. This may include commands to control the display or actual pixel data for rendering images or text. The data is sent in bytes with the acknowledgment from the OLED screen.

4. Stop Condition:

Once the data transmission is complete, the Raspberry Pi Pico sends a stop condition on the I2C bus. This informs the OLED screen that the communication session has ended.

The I2C protocol provides efficient way to communicate between the Raspberry Pi Pico and the OLED screen, allowing for the exchange of data and commands with minimal wiring and hardware complexity. Other screens and protocols are available on the Pico

**Breakdown of the two lines of code:**

**i2c = I2C(0, scl=Pin(1), sda=Pin(0), freq=400000)**

This line initialises an I2C object (`i2c`) using theI2C` class provided by the MicroPython machine` module. Here is a breakdown of the parameters:

0: This parameter specifies the I2C bus number. The Raspberry Pi Pico has two I2C buses, numbered 0 and 1. In this case, it’s using I2C bus 0.

scl=Pin(1): This parameter specifies the pin for the I2C serial clock line (SCL). In this case, the SCL pin is set to GPIO pin 1.

sda=Pin(0): This parameter specifies the pin for the I2C serial data line (SDA). In this case, the SDA pin is set to GPIO pin 0.

freq=400000: This parameter sets the frequency of the I2C communication. The value400000` represents 400 kHz, which is a common I2C frequency.

The i2c object is then used as the communication interface for the OLED screen.

**oled = SSD1306\_I2C(WIDTH, HEIGHT, i2c)**

This line creates an object (`oled`) of theSSD1306\_I2C` class.

WIDTH and HEIGHT: These parameters represent the width and height of the OLED display in pixels. These values should match the specifications of your specific OLED screen.

i2c: This parameter specifies the I2C interface (`i2c` object) that the OLED screen is connected to. The communication with the OLED display will be performed over the I2C bus using the specified I2C object.