กด swA ให้เลื่อนตำแหน่งไฟ swB ให้เปลี่ยสี

from machine import Pin

from neopixel import NeoPixel

from time import sleep

# Your NeoPixel initialization

np = NeoPixel(Pin(13, Pin.OUT), 25)

p = 0  # Initialize p to 0

sw1 = Pin(5, Pin.IN, Pin.PULL\_UP)

sw2 = Pin(12, Pin.IN, Pin.PULL\_UP)

colors = [(30, 0, 0), (0, 0, 30), (20, 0, 20), (0, 30, 0)]  # Red, Blue, Purple, Green

color\_index = 0  # Initial color index

while True:

    # Check if sw1 is pressed to change the pattern

    if sw1.value() == 0:

        p = (p + 1) % 24

    # Check if sw2 is pressed to change the color

    if sw2.value() == 0:

        # Cycle through the color list

        color\_index = (color\_index + 1) % len(colors)

    # Show the pattern with the selected color

    for i in range(24):

        if i == p:

            np[i] = colors[color\_index]

        else:

            np[i] = (0, 0, 0)

    np.write()

    sleep(0.2)

กดปุ่ม A neopixel +1 iLED4 +1 กดปุ่ม B เปลี่ยนสี

import iLED4

from machine import Pin,ADC

from neopixel import NeoPixel

from time import sleep

adc = ADC(Pin(33))  # P2

adc.atten(ADC.ATTN\_11DB)

# Your NeoPixel initialization

np = NeoPixel(Pin(13, Pin.OUT), 25)

p = 0  # Initialize p to 0

sw1 = Pin(4, Pin.IN, Pin.PULL\_UP)

sw2 = Pin(2, Pin.IN, Pin.PULL\_UP)

colors = [(30, 0, 0), (0, 0, 30), (20, 0, 20), (0, 30, 0)]  # Red, Blue, Purple, Green

color\_index = 0  # Initial color index

while True:

    # Check if sw1 is pressed to change the pattern

    if sw1.value() == 0:

        p = (p + 1) % 24

        iLED4.writeDigitNum(3,p%10,0)

        iLED4.writeDigitNum(2,p%100/10,0)

    # Check if sw2 is pressed to change the color

    if sw2.value() == 0:

        # Cycle through the color list

        color\_index = (color\_index + 1) % len(colors)

    # Show the pattern with the selected color

    for i in range(24):

        if i == p:

            np[i] = colors[color\_index]

        else:

            np[i] = (0, 0, 0)

    np.write()

    sleep(0.2)

กดปุ่ม A neopixel +1 ZX-pot เปลี่ยนสี

from machine import Pin, ADC

from neopixel import NeoPixel

from time import sleep

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

sw1 = Pin(5, Pin.IN, Pin.PULL\_UP)

# Your NeoPixel initialization

np = NeoPixel(Pin(13), 25)

p = 0  # Initialize p to 0

colors = [(30, 0, 0), (0, 0, 30), (20, 0, 20), (0, 30, 0)]  # Red, Blue, Purple, Green

color\_index = 0  # Initial color index

while True:

    print(adc.read())

    # Check if sw1 is pressed to change the pattern

    if adc.read() <= 1000:

        np[p] = colors[0]  # Assigning Red color to the specific NeoPixel index

    # Check if sw2 is pressed to change the color

    elif adc.read() <= 2000 :

        # Cycle through the color list

        np[p] = colors[1]

    elif adc.read() <= 3000 :

        # Cycle through the color list

        np[p] = colors[2]

    elif adc.read() <= 4000 :

        # Cycle through the color list

        np[p] = colors[3]

    if sw1.value() == 0:

        p = (p + 1) % 24

    # Show the pattern with the selected color

    np.write()

    sleep(0.2)

neopixel rgb + ZX-POT

from machine import Pin, ADC

from neopixel import NeoPixel

from time import sleep

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

# Your NeoPixel initialization

np = NeoPixel(Pin(13), 25)

p = 0  # Initialize p to 0

colors = [(30, 0, 0), (0, 0, 30), (20, 0, 20), (0, 30, 0)]  # Red, Blue, Purple, Green

color\_index = 0  # Initial color index

while True:

    print(adc.read())

    # Check if sw1 is pressed to change the pattern

    if adc.read() <= 1000:

        np[p] = colors[0]  # Assigning Red color to the specific NeoPixel index

    # Check if sw2 is pressed to change the color

    elif adc.read() <= 2000 :

        # Cycle through the color list

        np[p] = colors[1]

    elif adc.read() <= 3000 :

        # Cycle through the color list

        np[p] = colors[2]

    elif adc.read() <= 4000 :

        # Cycle through the color list

        np[p] = colors[3]

    # Show the pattern with the selected color

    np.write()

    sleep(0.2)

เมื่อให้ sonar มีค่า = 0 ให้ iled วิ่งจากขวาไปซ้าย

import iLED4

from machine import Pin, ADC

from time import sleep

adc = ADC(Pin(32, 33))

adc.atten(ADC.ATTN\_11DB)

iii = 3

while True:

    print('ADC=', adc.read(), 'RANGE', adc.read() / 40.95)

    if adc.read() == 0:

        for i in range (4):

            iLED4.writeDigitRaw(i,0x01)

            iLED4.writeDisplay()

            sleep(1)

            for ii in range (4):

                iLED4.writeDigitRaw(ii,0x00)

                iLED4.writeDisplay()

        iLED4.writeDigitRaw(iii, 0x02)

        iLED4.writeDisplay()

        sleep(1)

        iLED4.writeDigitRaw(iii, 0x04)

        iLED4.writeDisplay()

        sleep(1)

        for i in range(3, -1, -1):

            iLED4.writeDigitRaw(i, 0x08)

            iLED4.writeDisplay()

            sleep(1)

            for ii in range(4, -1, -1):

                iLED4.writeDigitRaw(ii, 0x00)

                iLED4.writeDisplay()

        iLED4.writeDigitRaw(0, 0x10)

        iLED4.writeDisplay()

        sleep(1)

        iLED4.writeDigitRaw(0, 0x20)

        iLED4.writeDisplay()

        sleep(1)

    else :

        iLED4.writeDigitRaw(0, 0x20)

        iLED4.writeDisplay()

        sleep(1)

        iLED4.writeDigitRaw(0, 0x10)

        iLED4.writeDisplay()

        sleep(1)

        for i in range (4):

            iLED4.writeDigitRaw(i, 0x08)

            iLED4.writeDisplay()

            sleep(1)

            for ii in range (4):

                iLED4.writeDigitRaw(ii, 0x00)

                iLED4.writeDisplay()

        iLED4.writeDigitRaw(iii, 0x04)

        iLED4.writeDisplay()

        sleep(1)

        iLED4.writeDigitRaw(iii, 0x02)

        iLED4.writeDisplay()

        sleep(1)

        for i in range(3, -1, -1):

            iLED4.writeDigitRaw(i,0x01)

            iLED4.writeDisplay()

            sleep(1)

            for ii in range(4, -1, -1):

                iLED4.writeDigitRaw(ii,0x00)

                iLED4.writeDisplay()

**POT เปลี่ยน หลอด LED โชว์เลขที่จอ OLED**

import math

from machine import Pin, ADC

from time import sleep

import display

import OLED

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

oled = OLED.SSD1306\_I2C(128, 64)

led1 = Pin(4, Pin.OUT)

led2 = Pin(15, Pin.OUT)

led3 = Pin(2, Pin.OUT)

while True:

    sensor\_value = adc.read()

    print(sensor\_value)

    oled.text(str(sensor\_value), 0, 0, 1)

    oled.show()

    sleep(0.5)

    if sensor\_value <= 1365:

        led1.on()

        led2.off()

        led3.off()

    elif 1365 < sensor\_value <= 2047:

        led1.off()

        led2.on()

        led3.off()

    elif sensor\_value > 2047:

        led1.off()

        led2.off()

        led3.on()

**เมื่อให้ sonar มีค่า = 0 ให้แสดงค่า iled 4 +1**

from machine import Pin,ADC

import iLED4

from time import sleep

import display

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

led=Pin(2,Pin.OUT)

x = 0

while True :

    print('ADC=',adc.read(),'RANGE',adc.read()/40.95)

    sleep(0.1)

    if adc.read() == 0 :

        led(1)

        x += 1

        iLED4.writeDigitNum(3,x%10,0)

        iLED4.writeDigitNum(2,x%100//10,0)

        iLED4.writeDigitNum(1,x%1000//100,0)

        iLED4.writeDigitNum(0,x%10000//1000,0)

    else :

        led(0)

**ควบคุม zx pot ให้แสดงบนiled4แล้วให้ servoหมุนตาม**

import iLED4

from machine import Pin,ADC,PWM

from time import sleep

adc = ADC(Pin(32))  # P2

adc.atten(ADC.ATTN\_11DB)

servo=PWM(Pin(15),freq=50)

while True :

    x=(adc.read()/22.75)

    iLED4.writeDigitNum(0,x//1000,0)

    iLED4.writeDigitNum(1,x%1000//100,0)

    iLED4.writeDigitNum(2,x%100//10,0)

    iLED4.writeDigitNum(3,x%10,0)

    iLED4.writeDisplay()

    sleep(0.1)

    servo.duty(int(x))

**เมื่อ sonar มีค่า = 0 ให้ไฟ ZX RGB3R ติด**

from machine import Pin,ADC

from time import sleep

from neopixel import NeoPixel

import display

np = NeoPixel(Pin(15, Pin.OUT), 3)

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

while True :

    print('ADC=',adc.read(),'Rang',adc.read()/40.95)

    sleep(0.5)

    if adc.read() == 0:

        np.fill((30,0,0))

        np.write()

    else :

        np.fill((0,0,0))

        np.write()

**POT เปลี่ยน หลอด LED**

import math

from machine import Pin,ADC

from time import sleep

import display

adc = ADC(Pin(32))

adc.atten(ADC.ATTN\_11DB)

led1=Pin(4,Pin.OUT)

led2=Pin(15,Pin.OUT)

led3=Pin(2,Pin.OUT)

while True :

    print(adc.read())

    if adc.read() <= 1365 :

        led1(1)

        led2(0)

        led3(0)

    elif adc.read() <= 2047 :

        led1(0)

        led2(1)

        led3(0)

    elif adc.read() >= 2047 :

        led1(0)

        led2(0)

        led3(1)