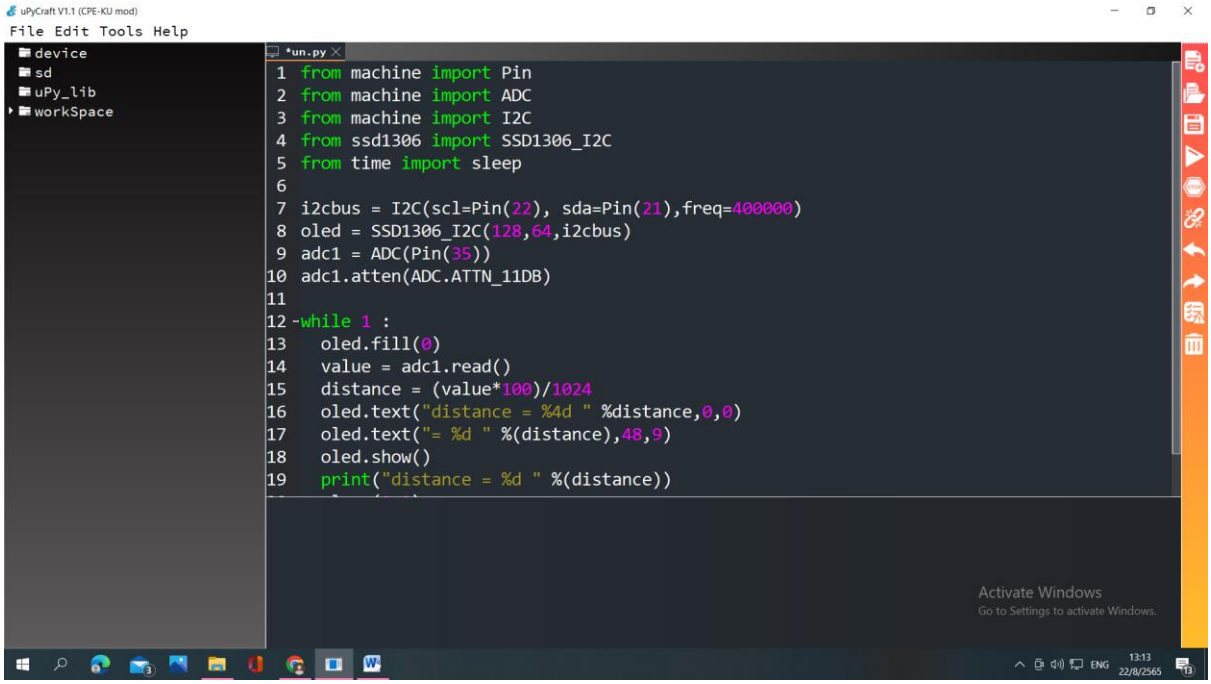
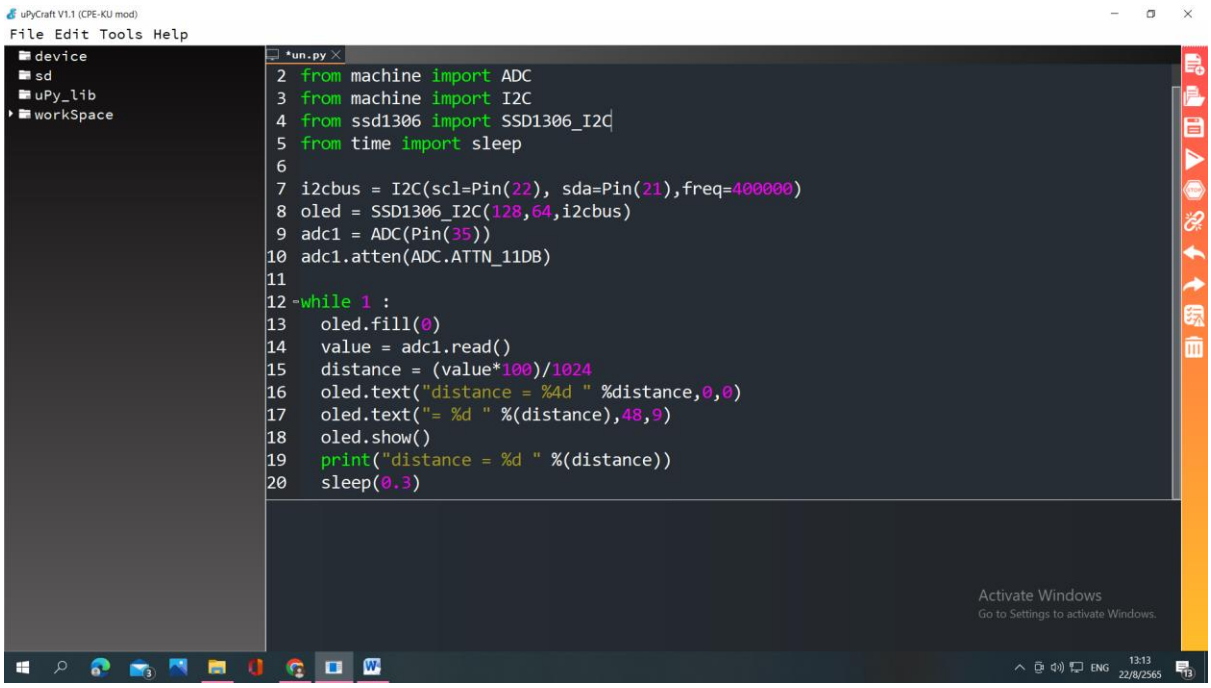
การเขียนโปรแกรม ultrasonic





1.from machine import Pin

โมดูลของช่อง Pin

2.from machine import ADC

โมดูลของช่องพอร์ต ADC

3.from machine import I2C

4.from ssd1306 import SSD1306_I2C

เรียกชุดคำสั่ง I2C เพื่อควบคุมคำสั่งที่ใช้ในการเชื่อมต่อ oled

5.from time import sleep

โมดูลของเวลา

6.

7.i2cbus = I2C(scl=Pin(22), sda=Pin(21),freq=400000)

คำสั่งสำหรับกำหนดจุดเชื่อมต่อจอ oled

```
8.oled = SSD1306_I2C(128,64,i2cbus)
```

กำหนดการแสดงหน้าจอ oled อยู่ตำแหน่งภายใน 128,64

```
9.adc1 = ADC(Pin(35))
```

กำหนดตัวแปร adc1 ของ ADC ช่อง Pin ที่ใช้คือ 35

```
10.adc1.atten(ADC.ATTN_11DB)
```

กำหนดค่าการลดทอนแรงดัน เพื่อให้รับแรงดันสูงสุด 3.3V

```
11.
12.while 1 :
```

การวนซ้ำของแผงวงจร โดยใช้คำสั่ง while

```
13.  oled.fill(0)
```

โมดูลของหน้าจอ oled จะทำการเคลียร์พื้นที่หน้าจอ

0 คือ ดับ

```
14.  value = adc1.read()
```

ตัวแปร adc1 ทำการอ่านค่าแรงดันไฟตรงที่ขาพอร์ตนั้นๆ

```
15.  distance = (value*100)/1024
```

เมื่อได้ค่า value นำมา * 100 และหาร 1024

จะได้ค่า distance

```
16.  oled.text("distance = %4d " %distance,0,0)
```

โมดูลของหน้าจอ oled จะแสดงค่า distance เป็นจำนวนเต็ม

ตำแหน่งที่ 0,0

```
17.  oled.text("= %d " %(distance),48,9)
```

โมดูลของหน้าจอ oled จะแสดงค่า **%d** เป็นจำนวนเต็ม

ตำแหน่งที่ 48,1

```
18.  oled.show()
```

โมดูลของหน้าจอ oled ที่โชว์

```
19.  print("distance = %d " %(distance))
```

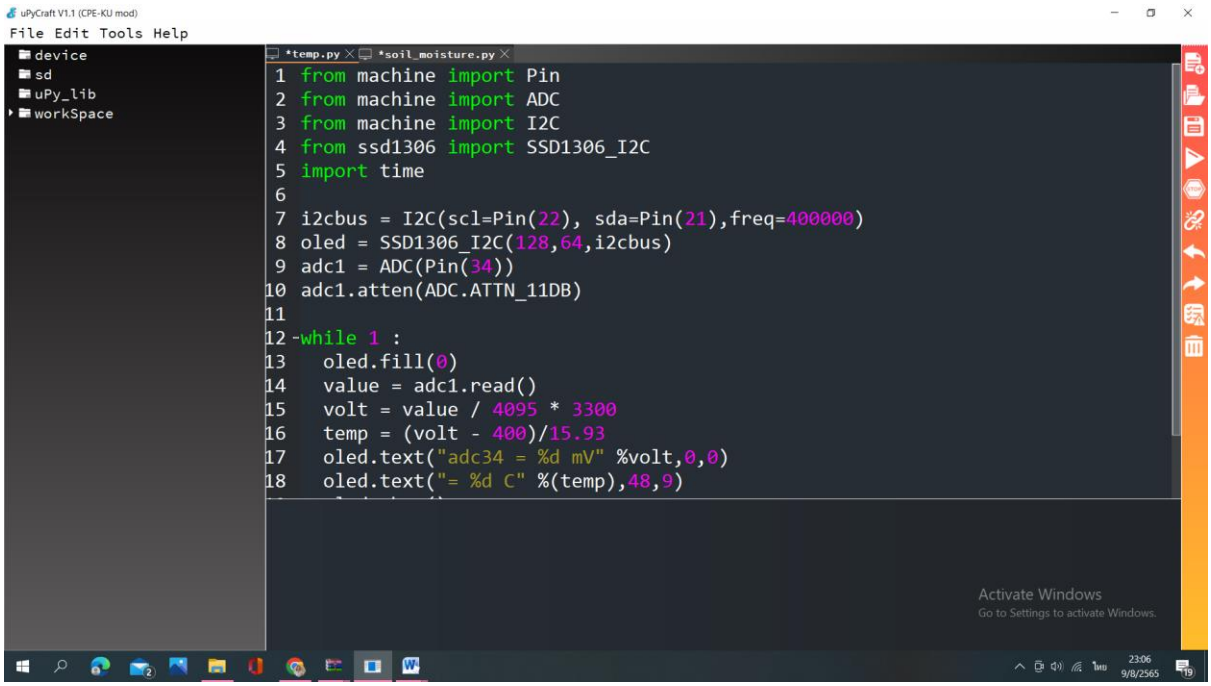
แสดงค่าจากการแปลงสัญญาณออกทางหน้าจอ Terminal

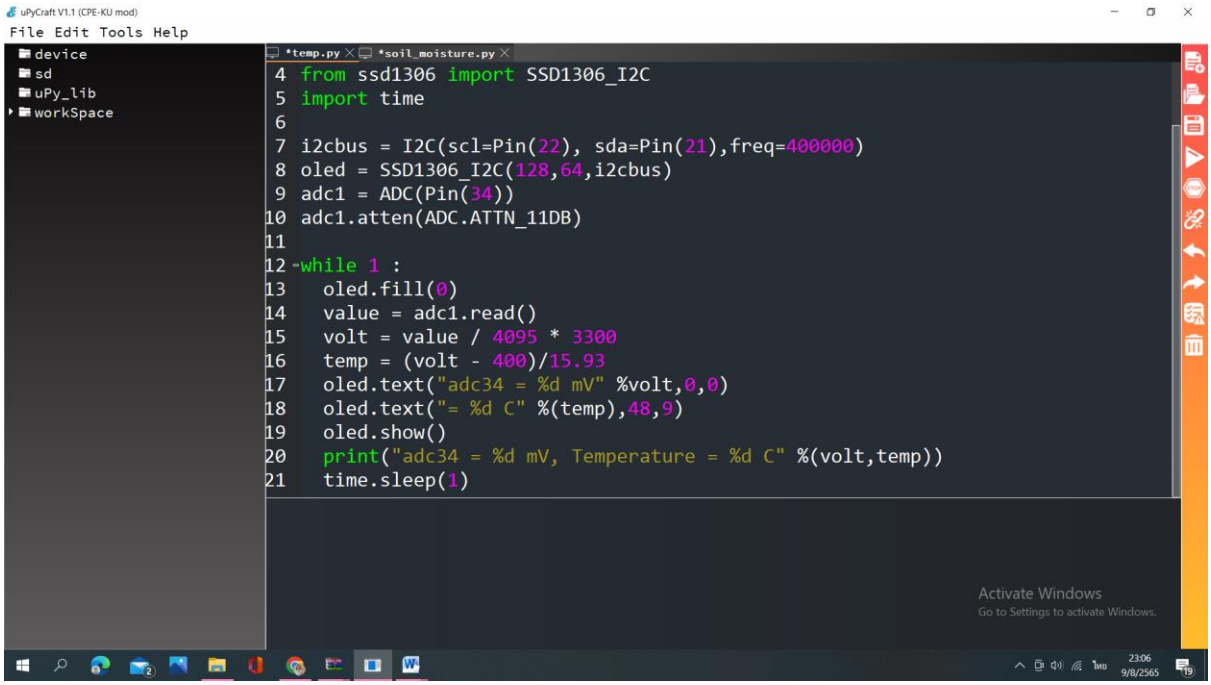
โดยค่าที่ได้จะเป็นเลขจำนวนเต็ม

```
20.  sleep(0.3)
```

โมดูลของเวลาที่แสดง 0.3 วินาที

การเขียนโปรแกรมวัดค่าอุณหภูมิ





1.from machine import Pin

โมดูลของช่อง Pin

2.from machine import ADC

โมดูลของช่องพอร์ต ADC

3.from machine import I2C

4.from ssd1306 import SSD1306_I2C

เรียกชุดคำสั่ง I2C เพื่อควบคุมคำสั่งที่ใช้ในการเชื่อมต่อ oled

5.import time

โมดูลของเวลา

6.

7.i2cbus =
I2C(scl=Pin(22),sda=Pin(21),freq=400000)

คำสั่งสำหรับกำหนดจุดเชื่อมต่อจอ oled

```
8.oled = SSD1306_I2C(128,64,i2cbus)
```

กำหนดการแสดงหน้าจอ oled อยู่ตำแหน่งภายใน 128,64

```
9.adc1 = ADC(Pin(34))
```

กำหนดตัวแปร **adc1** ของ **ADC** ช่อง **Pin** ที่ใช้คือ 34

```
10.adc1.atten(ADC.ATTN_11DB)
```

กำหนดค่าการลดทอนแรงดัน เพื่อให้รับแรงดันสูงสุด 3.3V

```
11.
12.while 1 :
```

การวนซ้ำของแผงวงจร โดยใช้คำสั่ง **while**

```
13.  oled.fill(0)
```

โมดูลของหน้าจอ oled จะทำการเคลียร์พื้นที่หน้าจอ

0 คือ ดับ

```
14.  value = adc1.read()
```

ตัวแปร **adc1** ทำการอ่านค่าแรงดันไฟตรงที่ขาพอร์ตนั้นๆ

```
15.  volt = value / 4095 * 3300
```

แปลงค่าที่ได้จาก **ADC** ให้เป็นแรงดัน 0 ถึง 3.3V

นำค่าที่ได้หารด้วย 4095 * 3300

```
16.  temp = (volt - 400)/15.93
```

นำค่า **volt** ที่ได้จากการแปลงค่ามาลบ 400 และหาร 15.93

จะได้ค่า **temp**

```
17.  oled.text("adc34 = %d mV" %volt,0,0)
```

โมดูลของหน้าจอ oled จะแสดงค่าแรงดันไฟ ตำแหน่งที่ 0,0

```
18.  oled.text("= %d C" %(temp),48,9)
```

โมดูลของหน้าจอ oled จะแสดงค่าอุณหภูมิ ตำแหน่งที่ 48,9

```
19.  oled.show()
```

โมดูลของหน้าจอ oled ที่โชว์

```
20.  print("adc34 = %d mV, Temperature =
%d C" %(volt,temp))
```

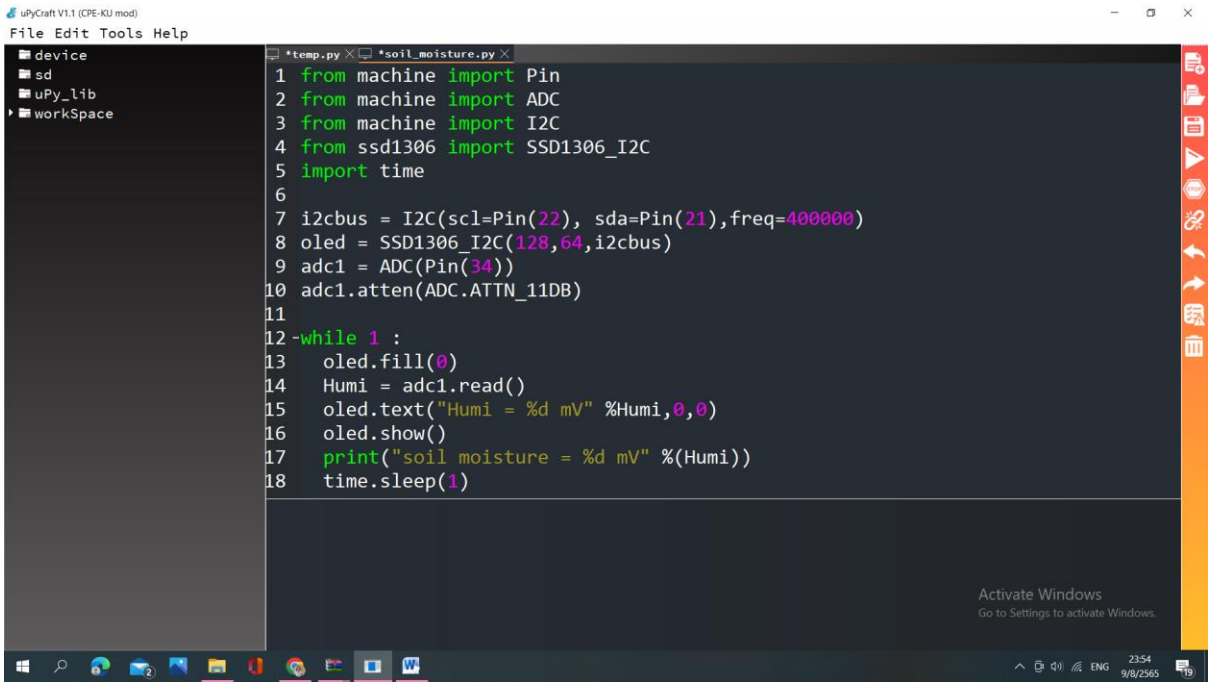
แสดงค่าจากการแปลงสัญญาณออกทางหน้าจอ Terminal

โดยค่าที่ได้จะเป็นเลขจำนวนเต็ม

```
21.  time.sleep(1)
```

โมดูลของเวลาที่แสดง 1 วินาที

การเขียนโปรแกรมวัดค่าความชื้นในดิน



1.from machine import Pin

โมดูลของช่อง Pin

2.from machine import ADC

โมดูลของช่องพอร์ต ADC

```
3.from machine import I2C
```

```
4.from ssd1306 import SSD1306_I2C
```

เรียกชุดคำสั่ง I2C เพื่อควบคุมคำสั่งที่ใช้ในการเชื่อมต่อ oled

```
5.import time
```

โมดูลของเวลา

```
6.
7.i2cbus =
I2C(scl=Pin(22),sda=Pin(21),freq=400000)
```

คำสั่งสำหรับกำหนดจุดเชื่อมต่อจอ oled

```
8.oled = SSD1306_I2C(128,64,i2cbus)
```

กำหนดการแสดงหน้าจอ oled อยู่ตำแหน่งภายใน 128,64

```
9.adc1 = ADC(Pin(34))
```

กำหนดตัวแปร **adc1** ของ **ADC** ช่อง **Pin** ที่ใช้คือ 34

```
10.adc1.atten(ADC.ATTN_11DB)
```

กำหนดค่าการลดทอนแรงดัน เพื่อให้รับแรงดันสูงสุด 3.3V

```
11.
12.while 1 :
```

การวนซ้ำของแผงวงจร โดยใช้คำสั่ง **while**

```
13.  oled.fill(0)
```

โมดูลของหน้าจอ oled จะทำการเคลียร์พื้นที่หน้าจอ

0 คือ ดับ

```
14.  Humi = adc1.read()
```

ตัวแปร **adc1** ทำการอ่านค่าแรงดันไฟตรงที่ขาพอร์ตนั้นๆ

```
15.  oled.text("Humi = %d mV" %Humi,0,0)
```

โมดูลของหน้าจอ oled จะแสดงค่าความชื้นในดินเป็นจำนวนเต็ม

ตำแหน่งที่ 0,0

```
16.  oled.show()
```

โมดูลของหน้าจอ oled ที่โชว์

```
17. print("soil moisture = %d mV"
%(Humi))
```

แสดงค่าจากการแปลงสัญญาณออกทางหน้าจอ Terminal

โดยค่าที่ได้จะเป็นเลขจำนวนเต็ม

```
18. time.sleep(1)
```

โมดูลของเวลาที่แสดง 1 วินาที