

과제 2

온도데이터를 시간과 함께 파일에 기록하기

실습) Code#1은 전에 과제1로 내준 온도 데이터를 파일에 기록하는 코드입니다. Code#2는 수업시간에 진행한 RTC 시간을 컴퓨터 화면에 출력하는 코드입니다.

두 코드를 참고해서 온도 센서와 함께 시간도 같이 파일에 쓰여지도록 만들어서 코드를 실습 과제에 제출하세요.

Code#1

```
from time import sleep
from machine import I2C, Pin
import onewire, ds18x20
import time
from ds3231_port import DS3231

led = Pin("LED", Pin.OUT)

sdaPIN = Pin(4)
sclPIN = Pin(5)

i2c = I2C(0, sda=sdaPIN, scl=sclPIN)

ds3231 = DS3231(i2c)

# GPIO 26번 핀에 OneWire 버스를 생성합니다.
data = machine.Pin(26)
temp_wire = onewire.OneWire(data) # GPIO 26번 핀에서 OneWire 버스 생성

# OneWire 버스를 사용하여 DS18X20 온도 센서 객체를 생성합니다.
temp_sensor = ds18x20.DS18X20(temp_wire)

# 버스에서 연결된 장치(온도 센서)를 스캔합니다.
roms = temp_sensor.scan()
print(len(roms), 'temperature sensor found') # 연결된 온도 센서의 수를 출력합니다.

def writeline(text):
    file = open("log.txt", "a")
    file.write(text + "\n")
    file.close()

while True:
    #light = bh1750.measurement
    #print(light)
    temp_sensor.convert_temp()
    time.sleep_ms(100)
    for rom in roms:
        t = temp_sensor.read_temp(rom)
        writeline(str(t))
    #writeline(str(light))
    print(t)
    led.value(1)
    sleep(0.5)
    led.value(0)
    sleep(0.5)
```

Code#2

```
import network
from machine import RTC
from machine import Pin
from machine import I2C
import utime as time
import usocket as socket
import ustruct as struct
from ds3231_port import DS3231

rtc = RTC()

# Connect to DS3231
print ('Syncing with DS3231')
sdaPIN = Pin(4) # SDA pin
sclPIN = Pin(5) # SCL pin

i2c = I2C(0, sda=sdaPIN, scl=sclPIN) # Init I2C using pins sda and scl

ds3231 = DS3231(i2c) # Create DS3231 object

print('Initial values')
print('DS3231 time:', ds3231.get_time())
print('RTC time:   ', time.localtime())

print('Setting DS3231 from RTC')
# DS3231와 RTC 시간의 차이를 확인하고 싶으면 아래 주석을 하고 실행
#ds3231.save_time() # Set DS3231 from RTC
print('DS3231 time:', ds3231.get_time())
print('RTC time:   ', time.localtime())
```

감사합니다.

공학도서관

www.gongdo.kr