05050554 鄒學緯

1.

#!/usr/bin/python

# Copyright (c) 2014 Adafruit Industries

# Author: Tony DiCola

# Permission is hereby granted, free of charge, to any person obtaining a copy

# of this software and associated documentation files (the "Software"), to deal

# in the Software without restriction, including without limitation the rights

# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

# copies of the Software, and to permit persons to whom the Software is

# furnished to do so, subject to the following conditions:

# The above copyright notice and this permission notice shall be included in all

# copies or substantial portions of the Software.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

# The above copyright notice and this permission notice shall be included in all

# copies or substantial portions of the Software.

# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

# SOFTWARE.

import sys

import Adafruit\_DHT

import time

import sys

import httplib, urllib

import json

deviceId = "DG0SgWPy"

# SOFTWARE.

import sys

import Adafruit\_DHT

import time

import sys

import httplib, urllib

import json

deviceId = "DG0SgWPy"

deviceKey = "pIvbcebyniUgtPDX"

def post\_to\_mcs(payload):

headers = {"Content-type": "application/json", "deviceKey": deviceKey}

not\_connected = 1

while (not\_connected):

try:

conn = httplib.HTTPConnection("api.mediatek.com:80")

conn.connect()

not\_connected = 0

except (httplib.HTTPException, socket.error) as ex:

def post\_to\_mcs(payload):

headers = {"Content-type": "application/json", "deviceKey": deviceKey}

not\_connected = 1

while (not\_connected):

try:

conn = httplib.HTTPConnection("api.mediatek.com:80")

conn.connect()

not\_connected = 0

except (httplib.HTTPException, socket.error) as ex:

print ("Error: %s" % ex)

time.sleep(10)

# sleep 10 seconds

conn.request("POST", "/mcs/v2/devices/" + deviceId + "/datapoints", jso$

response = conn.getresponse()

print( response.status, response.reason, json.dumps(payload), time.strf$

data = response.read()

conn.close()

sensor\_args = { '11': Adafruit\_DHT.DHT11,

'22': Adafruit\_DHT.DHT22,

'2302': Adafruit\_DHT.AM2302 }

if len(sys.argv) == 3 and sys.argv[1] in sensor\_args:

sensor = sensor\_args[sys.argv[1]]

pin = sys.argv[2]

else:

print('Usage: sudo ./Adafruit\_DHT.py [11|22|2302] <GPIO pin number>')

print('Example: sudo ./Adafruit\_DHT.py 2302 4 - Read from an AM2302 connect$

sys.exit(1)

# Try to grab a sensor reading. Use the read\_retry method which will retry up

# to 15 times to get a sensor reading (waiting 2 seconds between each retry).

while 1:

humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

# Un-comment the line below to convert the temperature to Fahrenheit.

# temperature = temperature \* 9/5.0 + 32

# Try to grab a sensor reading. Use the read\_retry method which will retry up

# to 15 times to get a sensor reading (waiting 2 seconds between each retry).

while 1:

humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

# Un-comment the line below to convert the temperature to Fahrenheit.

# temperature = temperature \* 9/5.0 + 32

# Note that sometimes you won't get a reading and

# the results will be null (because Linux can't

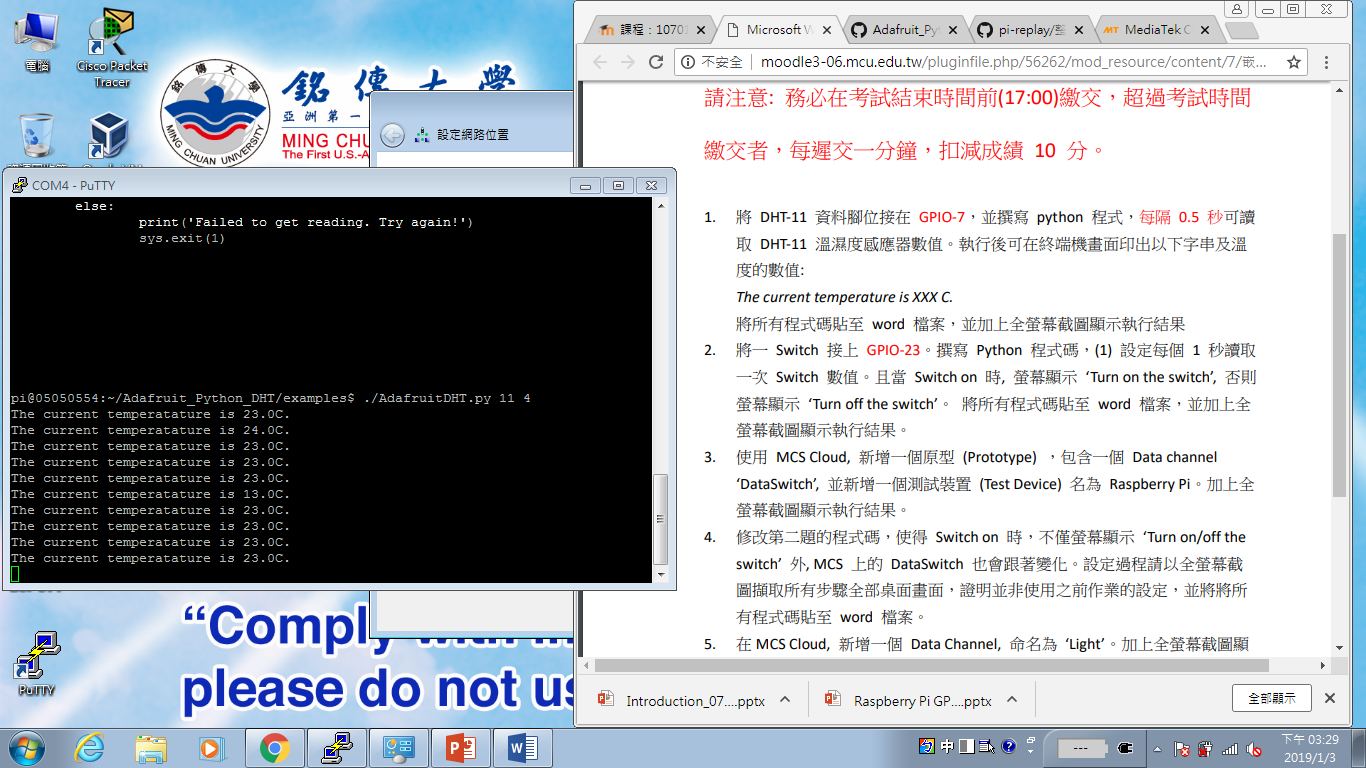
# guarantee the timing of calls to read the sensor).

# If this happens try again!

if humidity is not None and temperature is not None:

print('The current temperatature is {0:0.1f}C. '.format(temper$

time.sleep(5)



2.

import sys

import Adafruit\_DHT

import time

import sys

import httplib, urllib

import json

import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)

GPIO.setup(23,GPIO.IN, pull\_up\_down=GPIO.PUD\_UP)

deviceId = "DG0SgWPy"

deviceKey = "pIvbcebyniUgtPDX"

def post\_to\_mcs(payload):

headers = {"Content-type": "application/json", "deviceKey": deviceKey}

not\_connected = 1

while (not\_connected):

try:

conn = httplib.HTTPConnection("api.mediatek.com:80")

conn.connect()

not\_connected = 0

except (httplib.HTTPException, socket.error) as ex:

print ("Error: %s" % ex)

time.sleep(1)

# sleep 10 seconds

conn.request("POST", "/mcs/v2/devices/" + deviceId + "/datapoints", jso$

response = conn.getresponse()

print( response.status, response.reason, json.dumps(payload), time.strf$

data = response.read()

conn.close()

# Parse command line parameters.

sensor\_args = { '11': Adafruit\_DHT.DHT11,

'22': Adafruit\_DHT.DHT22,

'2302': Adafruit\_DHT.AM2302 }

if len(sys.argv) == 3 and sys.argv[1] in sensor\_args:

sensor = sensor\_args[sys.argv[1]]

pin = sys.argv[2]

else:

print('Usage: sudo ./Adafruit\_DHT.py [11|22|2302] <GPIO pin number>')

print('Example: sudo ./Adafruit\_DHT.py 2302 4 - Read from an AM2302 connect$

sys.exit(1)

# Try to grab a sensor reading. Use the read\_retry method which will retry up

# to 15 times to get a sensor reading (waiting 2 seconds between each retry).

while 1:

humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

print('Usage: sudo ./Adafruit\_DHT.py [11|22|2302] <GPIO pin number>')

print('Example: sudo ./Adafruit\_DHT.py 2302 4 - Read from an AM2302 connect$

sys.exit(1)

# Try to grab a sensor reading. Use the read\_retry method which will retry up

# to 15 times to get a sensor reading (waiting 2 seconds between each retry).

while 1:

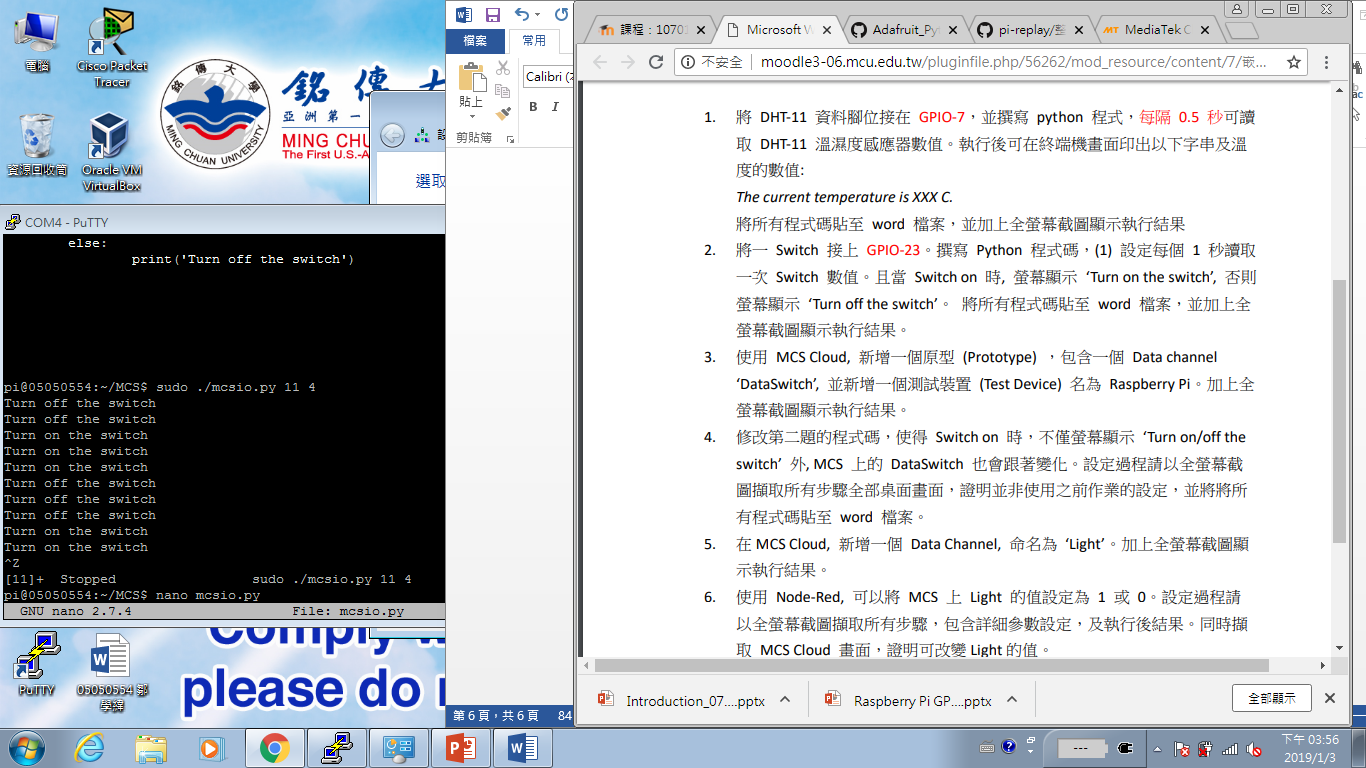
humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

SwitchStatus = GPIO.input(23)

if( SwitchStatus == 0):

print('Turn on the switch')

else:

print('Turn off the switch')

3.



4.