Inhaltsverzeichnis

Wemos D1 Mini: NeoPixel mit Micropython

Die aktuelle Firmware („latest stable Firmware“) für ESP8266-Boards hier herunterladen: https://micropython.org/download/esp8266/

(Und wenn man aus Versehen die nur für 512K compilierte Version erwischt, wundert man sich einen ganzen Vormittag, warum nichts funktioniert....)

```
esptool.py --port /dev/ttyUSB0 erase_flash
esptool.py --port /dev/ttyUSB0 --baud 115200 write_flash --flash_size=detect
0 //media/ram/esp8266-20220117-v1.18.bin
```

Und dann kann man entweder mit Thonny [1] ganz gut arbeiten, es soll wohl auch mit dem CLI tool ampy funktionieren

```
sudo pip3 install adafruit-ampy --upgrade
export AMPY_PORT=/dev/ttyUSB0
```

Pinout:

Pin	Name	Signal	Funktion
18	GPIO 0	D3	D+

```
# boot.py -- run on boot-up
from machine import Pin, reset
from neopixel import NeoPixel

from time import sleep, ticks_ms
import network



import time


def hexMac(byteMac):
    """
    Die Funktion hexMAC nimmt die MAC-Adresse im Bytecode
    entgegen und bildet daraus einen String fuer die Rueckgabe
    """
    macString =""
    for i in range(0,len(byteMac)):
        macString += hex(byteMac[i])[2:]
        if i <len(byteMac)-1 :
            macString +="-"
    return macString

# ***************** Connect to WLAN **********************
connectStatus = {
```

```
    0: "STAT_IDLE",
    1: "STAT_CONNECTING",
    5: "STAT_GOT_IP",
    2:  "STAT_WRONG_PASSWORD",
    3:  "NO AP FOUND",
    4:  "STAT_CONNECT_FAIL",
    }

client_id = "micropython"
#mqtt_server = "192.168.1.112"
mqtt_server = "schnipsl"
mqtt_server_port = 0
topic_sub = "neopixel"
mySSID = 'SKWLANAP2'
myPass = 'password'

# Unbedingt das AP-Interface ausschalten
nac=network.WLAN(network.AP_IF)
nac.active(False)
nac=None

# Wir erzeugen eine Netzwerk Interface-Instanz
nic = network.WLAN(network.STA_IF)
nic.active(False)
# Abfrage der MAC-Adresse zum Eintragen im Router,
# damit die Freigabe des Zugangs erfolgen kann
MAC = nic.config('mac')
myID=hexMac(MAC)
print("Client-ID",myID)

client_id += myID
# Wir aktivieren das Netzwerk-Interface
nic.active(True)


# Anmelden am WLAN-Router
nic.connect(mySSID, myPass)

if not nic.isconnected():
   # warten bis die Verbindung zum Accesspoint steht
   while not nic.isconnected():
       print("{}.".format(nic.status()),end='')
       sleep(2)

print("joo")

# Wenn verbunden, zeige Verbindungsstatus & Config-Daten
print("\nconnected: ",nic.isconnected())
print("\nVerbindungsstatus: ",connectStatus[nic.status()])
# War die Konfiguration erfolgreich? Kontrolle
STAconf = nic.ifconfig()
```

```
print("STA-IP:\t\t",STAconf[0],"\nSTA-NETMASK:\t",\
     STAconf[1],"\nSTA-GATEWAY:\t",STAconf[2] ,sep='')

#
https://github.com/RuiSantosdotme/ESP-MicroPython/blob/master/code/MQTT/umqt
tsimple.py
from umqttsimple import MQTTClient
import esp
esp.osdebug(None)
import gc
gc.collect()


def demo(np):
    n = np.n

    # cycle
    for i in range(4 * n):
        for j in range(n):
            np[j] = (0, 0, 0)
        np[i % n] = (255, 255, 255)
        np.write()
        time.sleep_ms(25)

    # bounce
    for i in range(4 * n):
        for j in range(n):
            np[j] = (0, 0, 128)
        if (i // n) % 2 == 0:
            np[i % n] = (0, 0, 0)
        else:
            np[n - 1 - (i % n)] = (0, 0, 0)
        np.write()
        time.sleep_ms(60)

    # fade in/out
    for i in range(0, 4 * 256, 8):
        for j in range(n):
            if (i // 256) % 2 == 0:
                val = i & 0xff
            else:
                val = 255 - (i & 0xff)
            np[j] = (val, 0, 0)
        np.write()

    # clear
    for i in range(n):
        np[i] = (0, 0, 0)
    np.write()

pin = Pin(0, Pin.OUT)   # set GPIO0 to output to drive NeoPixels
```

```
np = NeoPixel(pin, 25)   # create NeoPixel driver on GPIO0 for 8 pixels
#np = NeoPixel(pin, 150,timing=0)   # create NeoPixel driver on GPIO0 for 8
pixels


def sub_cb(topic, msg):
  print((topic, msg))
  demo(np)

def connect_and_subscribe():
  global client_id, mqtt_server, topic_sub,mqtt_server_port
  # MQTT keepalive error:
https://github.com/micropython/micropython-lib/issues/445
  client = MQTTClient(client_id,
mqtt_server,port=mqtt_server_port,keepalive=60)
  client.set_callback(sub_cb)
  print(client_id, mqtt_server)
  client.connect()
  client.subscribe(topic_sub)
  print('Connected to %s MQTT broker, subscribed to %s topic' %
(mqtt_server, topic_sub))
  return client

def restart_and_reconnect():
  print('Failed to connect to MQTT broker. Reconnecting...')
  time.sleep(10)
  reset()

try:
    client = connect_and_subscribe()
    ping_counter=0
    while True:
        new_message = client.check_msg()
        if new_message:
            print("new_message",new_message)
            #client.publish(topic_pub, b'received')
        sleep(1)
        ping_counter-=1
        if ping_counter<0:
            ping_counter=30
            client.ping()
except OSError as e:
  restart_and_reconnect()




#demo(np)
#n = np.n

# cycle
```

```
#for i in range( n):
#    np[i] = (128, i , i)
#    np.write()
#    time.sleep_ms(750)

#np[0] = (128, 128, 128) # set the first pixel to white
#np.write()              # write data to all pixels
#r, g, b = np[0]         # get first pixel colour
```

1)

https://randomnerdtutorials.com/getting-started-thonny-micropython-python-ide-esp32-esp8266/