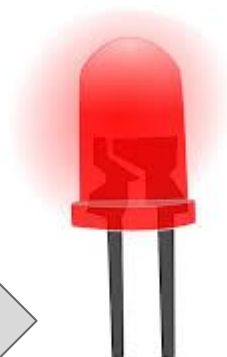
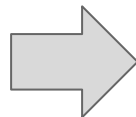
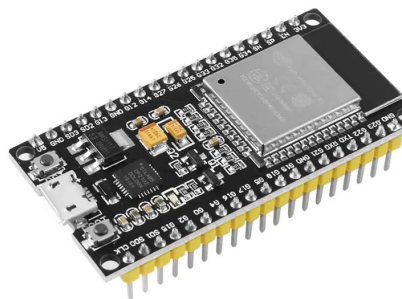
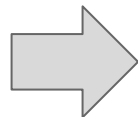


# ESP32 klient til MQTT

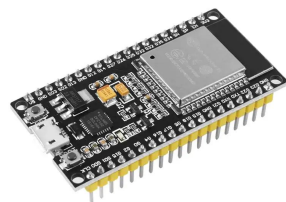


Coding Pirates 2024-10-30

**Hvad?**



# Hvordan?



Forbind til  
WiFi

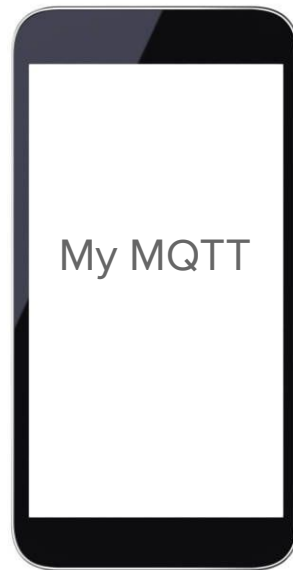
Forbind til  
MQTT

Subscribe

WiFi:  
frivillig  
test1234



Publish



# Hvordan forbinder man ESP32 til WiFi?

```
import time
import network

SSID = 'frivillig'
PASSWORD = 'test1234'

sta_if = network.WLAN(network.STA_IF)

sta_if.active(True)
sta_if.connect(SSID, PASSWORD)

print(f'Connecting to WiFi {SSID}', end='')
while not sta_if.isconnected():
    time.sleep(1)
    print('.', end='')
```

# Hvordan forbinder man ESP32 til MQTT?

```
from umqtt.robust import MQTTClient

MQTT_BROKER = '192.168.100.129'

CLIENT_ID = 'esp32-test-client'

client = MQTTClient(CLIENT_ID, MQTT_BROKER)

client.connect()

print(f'Connected to MQTT Broker: {MQTT_BROKER}')
```

# Forbered ESP32 pakker

I Thonny Shell skriv flg. linier

```
import network
sta_if = network.WLAN(network.STA_IF)
sta_if.active(True)
sta_if.connect('frivillig', 'test1234')
```

Så skulle du gerne få noget som 🙌

```
MicroPython v1.19.1 on 2022-06-18; ESP32S3 module with ESP32S3
Type "help()" for more information.
```

```
>>> import network
>>> sta_if = network.WLAN(network.STA_IF)
>>> sta_if.active(True)
```

```
True
```

```
>>> sta_if.connect('frivillig', 'test1234')
>>> sta_if.isconnected()
```

```
True
```

```
>>>
```

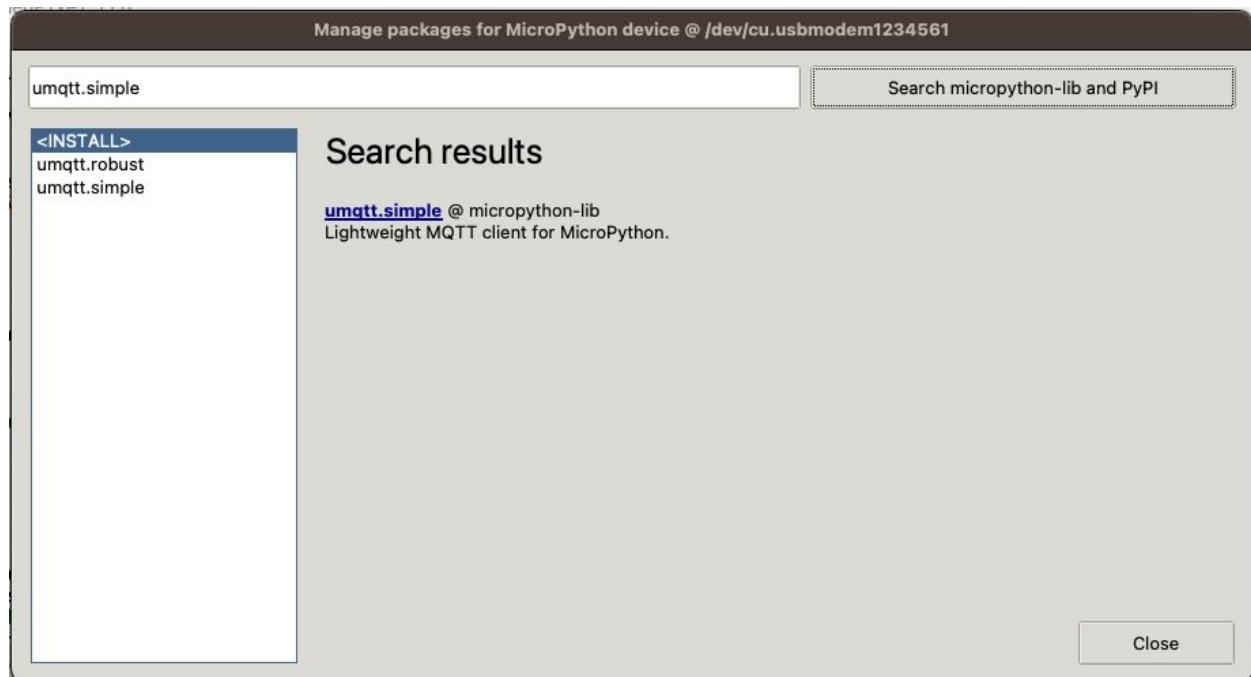
# Installer pakker

Thonny vælg Tools > Manage packages...

Søg og installer flg

`umqtt.simple`

`umqtt.robust`



# WiFi hjælper

Hent denne fil og gem den som wifi.py på din ESP32

<https://raw.githubusercontent.com/veggerby/cpmqtt/refs/heads/main/pymakr/wifi.py>



**Nu skal der  
kodes...**

```

import time
from umqtt.robust import MQTTClient
from machine import Pin
from neopixel import NeoPixel
import wifi
import random

NAVN = 'skriv-dit-navn-her'

SSID = 'frivillig'
PASSWORD = 'test1234'
MQTT_BROKER = '192.168.100.129'
MQTT_PORT = 1883
MQTT_USERNAME = None
MQTT_PASSWORD = None

CLIENT_ID = f'esp32-{NAVN}-{time.time_ns()}'

INFO_TOPIC = 'info'

wifi.try_connect_to_wifi(SSID, PASSWORD)

client = MQTTClient(CLIENT_ID, MQTT_BROKER, port=MQTT_PORT, user=MQTT_USERNAME, password=MQTT_PASSWORD)

def subscribe(topic, msg):
    topic_name = topic.decode('utf-8')
    message = msg.decode('utf-8')
    print(f'Received message {message} on {topic_name}')

client.connect()
client.set_callback(subscribe)

print(f'Connected to MQTT Broker: {MQTT_BROKER}')

client.subscribe(INFO_TOPIC)

while True:
    client.check_msg()

```

Forbind herefter din telefon til broderen og publish en besked på topic "info"

```
# Konfiguration af NeoPixel LED
pin = 48 # GPIO 48
num_leds = 1
np = NeoPixel(Pin(pin), num_leds)

def skift_farve(farve):
    print(f'Skifter LED til {farve}')
    color = None
    if (farve == 'red'):
        color = (255, 0, 0)
    elif (farve == 'green'):
        color = (0, 255, 0)
    elif (farve == 'blue'):
        color = (0, 0, 255)
    elif (farve == 'random'):
        color = (random.randrange(180), random.randrange(180), random.randrange(180))
    elif (farve == 'off'):
        color = (0, 0, 0)
    else:
        print(f'Ukendt farve {farve}')
        return

    np[0] = color
    np.write()
```

Indsæt følgende  
kode før “def  
subscribe(topic,  
msg):”

Ændre subscribe  
funktionen til

```
def subscribe(topic, msg):  
    topic_name = topic.decode('utf-8')  
    message = msg.decode('utf-8')  
    print(f'Received message {message} on {topic_name}')  
    if topic_name == LED_TOPIC:  
        skift_farve(message)
```

Tilføj

```
client.subscribe(f'led/{NAVN}')
```

Under

```
client.subscribe(INFO_TOPIC)
```

# Så skulle din fil gerne indeholde dette...

```
import time
from umqtt.robust import MQTTClient
from machine import Pin
from neopixel import NeoPixel
import wifi
import random

NAVN = 'jesper'

## ESP32 CP network
SSID = 'frivillig'
PASSWORD = 'test1234'
MQTT_BROKER = '192.168.100.129'
MQTT_PORT = 1883
MQTT_USERNAME = None
MQTT_PASSWORD = None

CLIENT_ID = f'esp32-{NAVN}-{time.time_ns()}'

INFO_TOPIC = 'info'
LED_TOPIC = f'led/{NAVN}'

wifi.try_connect_to_wifi(SSID, PASSWORD)

client = MQTTClient(CLIENT_ID, MQTT_BROKER, port=MQTT_PORT, user=MQTT_USERNAME, password=MQTT_PASSWORD)

# Konfiguration af NeoPixel LED
pin = 48 # GPIO 48
num_leds = 1
np = NeoPixel(Pin(pin), num_leds)
```

```
def skift_farve(farve):
    print(f'Skifter LED til {farve}')
    color = None
    if (farve == 'red'):
        color = (255, 0, 0)
    elif (farve == 'green'):
        color = (0, 255, 0)
    elif (farve == 'blue'):
        color = (0, 0, 255)
    elif (farve == 'random'):
        color = (random.randrange(180), random.randrange(180), random.randrange(180))
    elif (farve == 'off'):
        color = (0, 0, 0)
    else:
        print(f'Ukendt farve {farve}')
        return

    np[0] = color
    np.write()

def subscribe(topic, msg):
    topic_name = topic.decode('utf-8')
    message = msg.decode('utf-8')
    print(f'Received message {message} on {topic_name}')
    if topic_name == LED_TOPIC:
        skift_farve(message)

client.connect()
client.set_callback(subscribe)

print(f'Connected to MQTT Broker: {MQTT_BROKER}')

client.subscribe(INFO_TOPIC)
client.subscribe(LED_TOPIC)

while True:
    client.check_msg()
```