



INHOUDSTAFEL

1	Doelstelling.....	3
2	Opstelling van de hardware	3
2.1	Materiaal.....	3
2.2	Opbouw	3
3	Arduino IDE – bibliotheken downloaden.....	4
4	ESP32 programmeren	4
5	Node-RED downloaden	5
6	MQTT connecteren met ESP32.....	5
6.1	Connectie via Node-RED	5
6.2	Connectie via Arduino	6
6.2.1	Server adres.....	6
6.2.2	Poort.....	6
7	Node-RED – Flows importeren.....	6
8	ESP32 - Instellingen	7
9	ESP32 - Uploaden	8
10	Testen.....	8

1 DOELSTELLING

Dit document beschrijft het volgende :

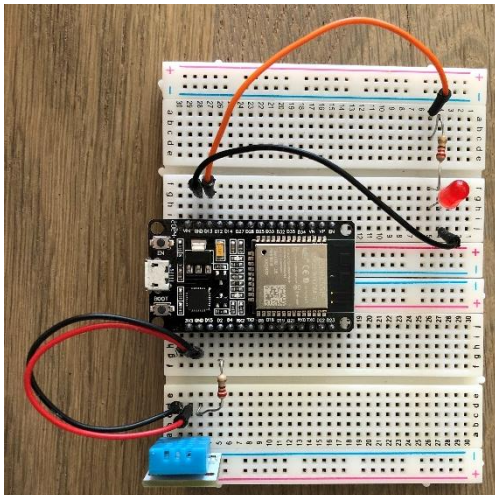
- Hoe een DHT11, een Led en een ESP32 wordt opgesteld op een Arduino Board.
- Hoe je de verschillende software en bibliotheken download en installeert
- Hoe connectie maken met MQTT broker
- Hoe connectie maken met het Wifi Access Point
- Hoe de hardware programmeren in Node-RED
- Hoe testen in Node-RED

2 OPSTELLING VAN DE HARDWARE

2.1 Materiaal

- Led
- Weerstand (100Ω)
- DHT11
- ESP32

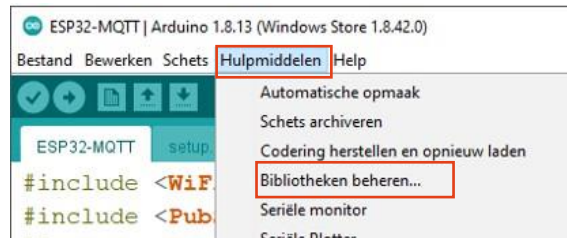
2.2 Opbouw



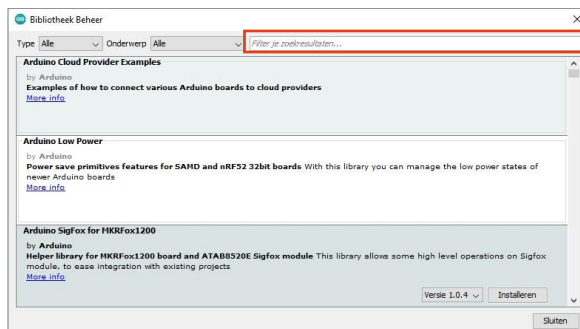
3 ARDUINO IDE – BIBLIOTHEKEN DOWNLOADEN

In deze stap ga je de volgende bibliotheken downloaden :

- MQTT van Joel Gaehwiler
- ArduinoJson van Benoit Blanchon
- Open de software Arduino IDE
- Klik op de knop “Hulpmiddelen”
- Kies “Bibliotheken beheren...”



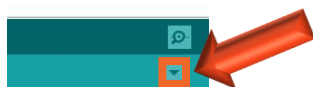
- Zoek de de MQTT en de ArduinoJson in het bibliotheekbeheer



- Selecteer de bibliotheek en installeer de hoogste versie

4 ESP32 PROGRAMMEREN

- Surf naar de website : <https://github.com/cedric-carels/Cloud-and-Security>
- Klik op “Assignment 1”
- Klik op ESP32-MQTT
- Klik op ESP32-MQTT.ino
- Kopieer de code
- Open of activeer de software Arduino IDE
- Maak een nieuw bestand aan (via de knop “Nieuw”)
- Plak de gekopieerde code in dit bestand
- Maak een nieuwe bijlage aan door op het pijltje rechts boven het scherm te klikken



- Kies uit het menu : “Nieuw Tabblad”
- Geef het nieuw tabblad de volgende benaming : **setup.h**
- Ga terug naar “Github” en ga een stap terug zodat je terug op de pagina Cloud-and-Security/ Assignment 1/ ESP32-MQTT
- Klik op “setup.h”
- Kopieer de code
- Activeer Arduino IDE
- Plak de gekopieerde code in het nieuw aangemaakte tabblad
- Pas je ssid en password aan

```
#ifndef WIFI_H_INCLUDE
#define WIFI_H_INCLUDE

//gebruik van WIFI of hotspot
#define ssid "je eigen ssid"
#define password "het wachtwoord van je ssid"

#endif
```

5 NODE-RED DOWNLOADEN

Je hebt Node-RED nodig om de ESP32 te kunnen aansturen.
Dit gebeurt door de software “Nodejs command prompt”

- Ga naar de website : <https://nodejs.org/en>
- Download de LTS en installeer op je pc

14.16.0 LTS
Recommended For Most Users

6 MQTT CONNECTEREN MET ESP32

6.1 Connectie via Node-RED

- Open de software “Nodejs command prompt”
- Voer in : NODE-RED
- Kopieer het IP-adres in de lijn : Server now running at **http://xxx.x.x.x:xxx/**

```
node-red
Your environment has been set up for using Node.js 14.15.5 (x64) and npm.

C:\Users\Carels Cédric>NODE-RED
8 Mar 10:17:16 - [info]

Welcome to Node-RED
=====

8 Mar 10:17:16 - [info] Node-RED version: v1.2.9
8 Mar 10:17:16 - [info] Node.js version: v14.15.5
8 Mar 10:17:16 - [info] Windows_NT 10.0.18363 x64 LE
8 Mar 10:17:17 - [info] Loading palette nodes
8 Mar 10:17:18 - [info] Dashboard version 2.28.1 started at /ui
8 Mar 10:17:18 - [info] Settings file : C:\Users\Carels Cédric\.node-red\settings.js
8 Mar 10:17:18 - [info] Context store : 'default' [module=memory]
8 Mar 10:17:18 - [info] User directory : \Users\Carels Cédric\.node-red
8 Mar 10:17:18 - [warn] Projects disabled : editorTheme.projects.enabled=false
8 Mar 10:17:18 - [info] Flow file : \Users\Carels Cédric\.node-red\flows_LAPTOP-AJ8G79A7.json
8 Mar 10:17:19 - [info] Server now running at http://xxx.x.x.x:xxx/
```

- Open je internetbrowser
- Klik in de balk plak het gekopieerde IP-adres in deze balk
De Node-RED wordt automatisch geopend.

6.2 Connectie via Arduino

In de Arduino-IDE moet je in de ESP32-MQTT de MQTT server en poort aanduiden. Deze gegevens moet je ophalen in de Nodejs command prompt.

- Server adres : de cijfers voor het dubbel punt
- Poort : de cijfers na het dubbel punt

6.2.1 Server adres

- In de Nodejs command prompt kopieer je het IP-adres in de lijn : Connect to broker : mqtt://**xx.xx.xx.xxx**:xxxx (cijfers voor :)

```
21 Mar 13:49:10 - [info] Starting flows
21 Mar 13:49:10 - [error] [mqtt out:837f37b6.ea0578] missing broker configuration
21 Mar 13:49:10 - [info] Started flows
21 Mar 13:49:10 - [info] [mqtt-broker:CC.C] Connected to broker: mqtt://xx.xx.xx.xxx:xxxx
21 Mar 13:49:10 - [info] [mqtt-broker:CC.C] Connected to broker: mqtt://xx.xx.xx.xxx:xxxx
```

- Activeer de Arduino-IDE
- Activeer het tabblad ESP32-MQTT
- Plak de gekopieerde tekst na "mqtt_server ="

```
const char* mqtt_server = "xxx.x.x.x"; // IP adres van de mqtt dat je uit nodejs comand
const int mqtt_port = xxxx; // poort van de mqtt dat je uit nodejs comand uit
```

6.2.2 Poort

- In de Nodejs command prompt kopieer je het IP-adres in de lijn : Connect to broker : mqtt://xx.xx.xx.xxx:**xxxx** (cijfers na :)

```
21 Mar 13:49:10 - [info] Starting flows
21 Mar 13:49:10 - [error] [mqtt out:837f37b6.ea0578] missing broker configuration
21 Mar 13:49:10 - [info] Started flows
21 Mar 13:49:10 - [info] [mqtt-broker:CC.C] Connected to broker: mqtt://xx.xx.xx.xxx:xxxx
21 Mar 13:49:10 - [info] [mqtt-broker:CC.C] Connected to broker: mqtt://xx.xx.xx.xxx:xxxx
```

- Activeer de Arduino-IDE
- Activeer het tabblad ESP32-MQTT
- Plak de gekopieerde tekst na "mqtt_port ="

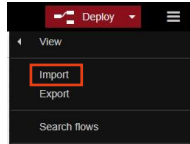
```
const char* mqtt_server = "xxx.x.x.x"; // IP adres van de mqtt dat je uit nodejs comand
const int mqtt_port = xxxx; // poort van de mqtt dat je uit nodejs comand uit
```

7 NODE-RED – FLOWS IMPORTEREN

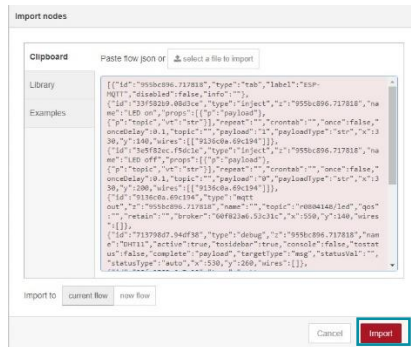
- Ga naar <https://github.com/cedric-carels/Cloud-and-Security>
- Klik op "Assignment 1"
- Klik op Node-red

- Klik op ESP-MQTT.json
- Kopieer lijn 1 volledig
- Ga naar Node-RED
- Klik op  (volledig rechts bovenaan het scherm van Node-RED)

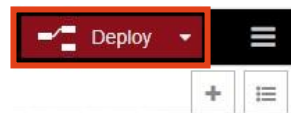
- Klik op "Import"



- Plak de gekopieerde tekst in het "roze" kader
- Klik op "Import"



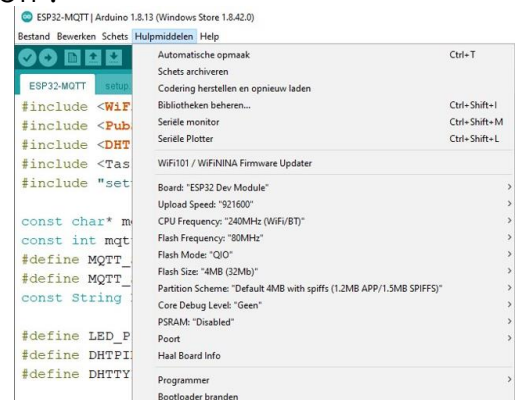
- Klik op de toets "Deploy" om af te ronden



Je kan natuurlijk zelf de flows aanmaken en configureren. Deze stap-voor-stap behandelingen vind je terug in bijlage 1.

8 ESP32 - INSTELLINGEN

- Verbind je ESP32 met je laptop of pc
- Open Arduino IDE
- Selecteer het tabblad ESP32-MQTT
- Zorg ervoor dat je de correcte instellingen meegeeft :
 - Board : ESP32 Dev Module
 - Upload Speed : 921600
 - CPU Frequency : 240MHz (WiFi/BT)
 - Flash Frequency : 80MHz
 - Flash Mode : QIO
 - Flash Size : 4MB (32Mb)
 - Partition Scheme : Default 4MB with Spiffs (1.2MB APP/1.5MB SPIFFS)
 - Core Debug Levels : Green
 - PSRAM : Disabled



9 ESP32 - UPLOADEN

- In Arduino-IDE, klik je op het groene pijltje (➔)



- Als de upload loopt krijg je Connecting.....

```
De schets gebruikt 706549 bytes (53%) programma-opslagruimte. Maximum is 1310720 bytes.  
Globale variabelen gebruiken 38484 bytes (11%) van het dynamisch geheugen. Resteren 289196 bytes voor lokale variabelen. Maximum is 327680 bytes.  
esptool.py v3.0-dev  
Serial port COM5  
Connecting.....  
chip is ESP32-D0W0Q6 (revision 1)
```

- Op de ESP32 moet je op het knopje "BOOT" blijven drukken tot de Connecting..... niet meer wordt weergegeven



10 TESTEN

In de Node-RED kan je de Led aansturen (aan en uitzetten). Hiervoor klik je op de knoppen "LED on" en "LED OFF".

BIJLAGE

Aanmaken van flows
Programmeren van flows
Koppelen van flows

FLows AANMAKEN

Via Node-RED ga je een flows aanmaken door functie blokken toe te voegen. Je moet een "common" flow en een "network" flow aanmaken.

- Volledig links van het scherm zie je een balk met "Common" en "Network"
- Common flows :



- Inject : Deze stuurt de Led aan. Aangezien de Led moet kunnen aangezet en afgezet worden, heb je dus 2 injects nodig. Sleep de knop "inject" in je flow. Herhaal dit zodat je 2 flows hebt.
- Debug : Stuurt de DHT11 aan. Sleep deze knop in de flow

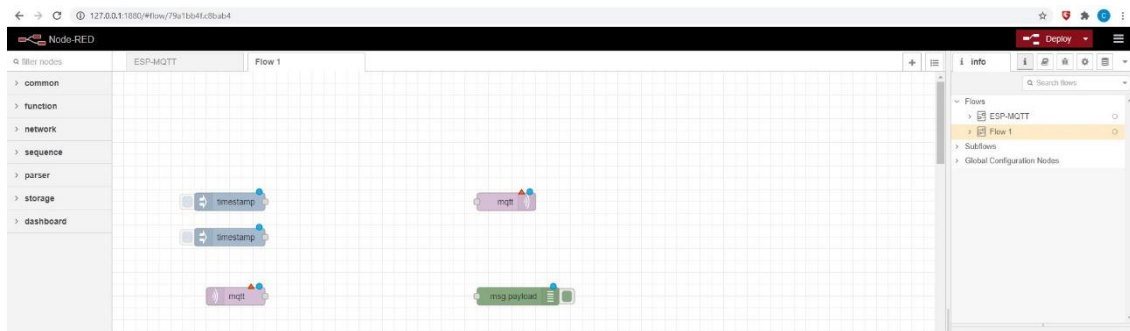
- Network flows :



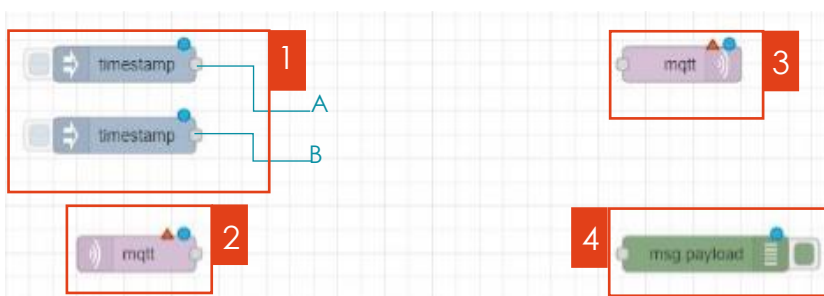
- Mqtt in : ontvangt de data van de DHT11. Sleep deze knop in je flow.
- Mqtt out: verstuurt de data naar de Led (aan of uit).

Dit is het resultaat :

- De "injects" worden weergegeven als "timestamp"
- De "mqtt in" wordt weergegeven als "mqtt"
- De "mqtt out" wordt weergegeven als "mqtt"
- De "debug" wordt weergegeven als "msg.payload"



FLows PROGRAMMEREN



TIMESTAMP PROGRAMMEREN [1]

Hier ga je ervoor zorgen dat de Led wordt aangestuurd (aan of uit).

Timestamp (A) = Led on

- Dubbel klik op de 1^e timestamp
- Vul de gegevens in onder de “properties” (volledig rechts van het scherm):
 - Name : Led on
 - Msg.payload = timestamp (staat reeds ingevuld)
 - Msg.topic = 1
- Klik op de rode “Done” knop

The screenshot shows a 'Properties' dialog box with the following fields:

- Name:** Led on
- msg. payload:** timestamp
- msg. topic:** 1

At the top right, there are buttons for 'Delete', 'Cancel', and 'Done'. A red arrow points to the 'Done' button.

De benaming “timestamp” is nu veranderd in “LED on”.

Timestamp (B) = Led off

- Dubbel klik op de 2^{de} timestamp
- Vul de gegevens in onder de “properties” (volledig rechts van het scherm):
 - Name : LED off
 - Msg.payload = timestamp (staat reeds ingevuld)
 - Msg.topic = 0
 - Klik op de rode “Done” knop

The screenshot shows a 'Properties' dialog box with the following fields:

- Name:** Led off
- msg. payload:** timestamp
- msg. topic:** 0

At the top right, there are buttons for 'Delete', 'Cancel', and 'Done'. A red arrow points to the 'Done' button.

De benaming “timestamp” is nu veranderd in “LED off”.


MQTT IN PROGRAMMEREN [2]

Hier ga je ervoor zorgen dat de MQTT broker de gegevens van de DHT11 (data via Arduino) inleest.

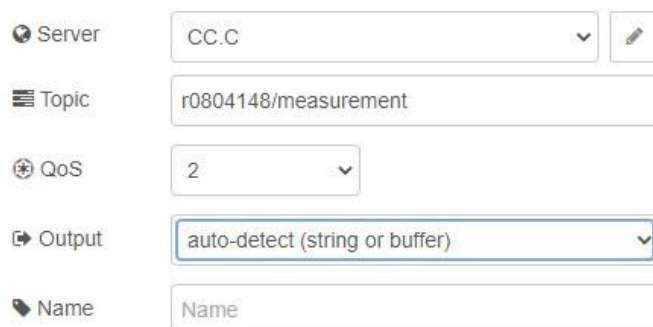
Omdat de DHT11 moet kunnen communiceren met de MQTT, moet de benaming die je in de programmatie gebruikt dezelfde zijn dan die in Arduino IDE.

Voor de MQTT in moet je dus de benaming van de “serial receiver” invoeren.

```
#define MQTT_SERIAL_PUBLISH_CH "r0804148/measurement"
#define MQTT_SERIAL_RECEIVER_CH "r0804148/led"
const String DEVICE_ID = "r0804148";
```



- Dubbel klik op “mqtt”
- Vul de gegevens in :
 - Server : CC.C (staat standaard ingevuld)
 - Topic : r0804148/measurement
 - QoS : 2 (staat standaard ingevuld)
 - Output : auto-detect (string or buffer) (staat standaard ingevuld)



De benaming “mqtt” is nu veranderd in “r0804148/measurement”.


MQTT OUT PROGRAMMEREN [3]

Hier ga je ervoor zorgen dat de MQTT broker de gegevens (aan of uit) doorgeeft aan de Led op je Arduino.


Omdat de MQTT moet kunnen communiceren met de Led, moet de benaming die je in de programmatie gebruikt dezelfde zijn dan die in Arduino IDE.

Voor de MQTT in moet je dus de benaming van de “serial receiver” invoeren.

```
#define MQTT_SERIAL_PUBLISH_CH "r0804148/measurement"
#define MQTT_SERIAL_RECEIVER_CH "r0804148/led"
const String DEVICE_ID = "r0804148";
```



- Dubbel klik op “mqtt”
- Vul de gegevens in :
 - Server : CC.C (staat standaard ingevuld)
 - Topic : r0804148/led
 - QoS : blanco laten
 - Retain : blanco laten
 - Name : blanco laten



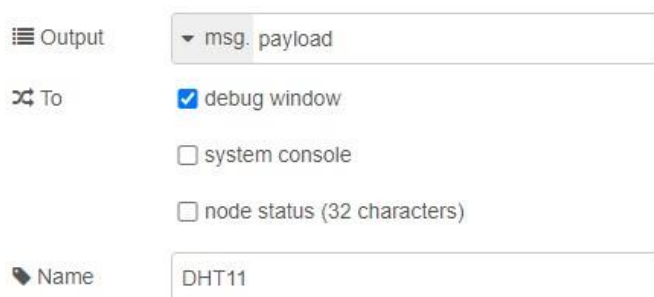
The screenshot shows a configuration interface for an MQTT client. It has four rows, each with a label and a text input field. The first row is labeled 'Server' and contains 'CC.C'. The second row is labeled 'Topic' and contains 'r0804148/led'. The third row is labeled 'QoS' and contains a dropdown menu with a downward arrow, followed by a 'Retain' checkbox and another dropdown menu with a downward arrow. The fourth row is labeled 'Name' and contains 'Name'.

De benaming “mqtt” is nu veranderd in “r0804148/led”.

DEBUG PROGRAMMEREN [4]

Hier ga je ervoor zorgen dat de gegevens die de MQTT broker heeft ingelezen van de DHT11 weergeeft.

- Dubbel klik op msg.payload
- Vul de gegevens in :
 - Output : msg.payload (staat standaard ingevuld)
 - To : “debug window” (staat standaard aangevinkt)
 - Name : DHT11



The screenshot shows a configuration interface for a debug window. It has three rows. The first row is labeled 'Output' and contains a dropdown menu with 'msg. payload' selected. The second row is labeled 'To' and contains three checkboxes: 'debug window' (checked), 'system console' (unchecked), and 'node status (32 characters)' (unchecked). The third row is labeled 'Name' and contains 'DHT11'.

De benaming “msg.payload” is nu veranderd in “DHT11”.

FLAWS AAN ELKAAR KOPPELEN

Om ervoor te zorgen dat de Led wordt aangestuurd en de DHT11 kan ingelezen worden, moeten de aangemaakte flows met elkaar verbonden worden.

LED ON & LED OFF

De Led on en de Led off moeten gekoppeld worden aan de r0804148/led

- Klik op het grijze blokje van de Led en sleep dit naar het grijze blokje van r0804148/led



DHT11

De DHT11 moet gekoppeld worden aan de r0804148/measurement

- Klik op het grijze blokje van de r0804148/measurement en sleep dit naar het grijze blokje van DHT11



De koppelingen zijn gemaakt :

