# Assignment 1

IP: 10.54.32.1  
Wireless Access Point: GraduaatIOT  
MQTT Broker

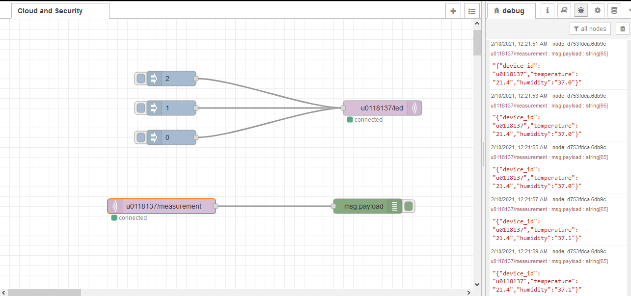
ESP32 – DHT22 – MQTT

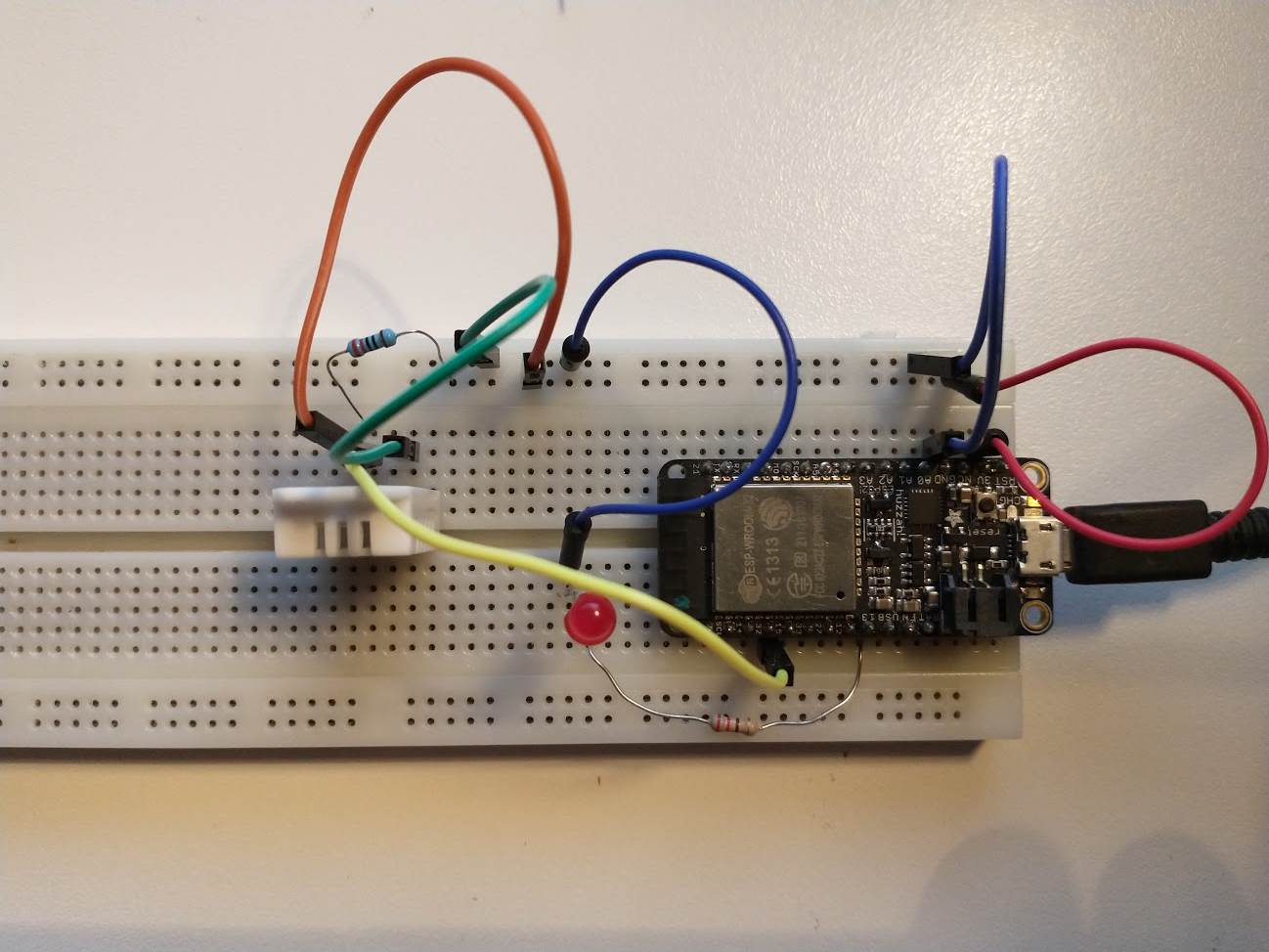


Control LED

Control LED

Publish measurements





Read measurements

In deze opdracht maken we de opstelling klaar om te gebruiken in volgende lessen. In deze les werken we nog volledig lokaal. In de volgende lessen zullen we deze opstelling stap voor stap verbinden met AWS (= Cloud Service Provider).

Een DHT22 temperatuur- en vochtigheidssensor wordt aangesloten aan een ESP32. Ook een led wordt aangesloten zoals te zien is in de afbeelding.

De ESP32 wordt m.b.v. idf of arduino IDE geprogrammeerd om onderstaande stappen uit te voeren. De voorbeeldcode kan je vinden op <https://github.com/dust555/TM_GIOT_CLOUD/>

* Connectie maken met het Wifi Access Point
* Connectie maken met MQTT broker
* Subscriben op jouw persoonlijk topic om de led aan te sturen
* Waarde van de DHT22 sensor lezen en deze in JSON formaat publishen naar jouw persoonlijk topic
* Gebruik node-red of een mqtt client op je laptop om de setup te testen
  + Subscribe op het measurement topic om te zien of je metingen correct worden verstuurd
  + Publish 1 of 0 naar het led topic om te zien of je zo de led kan sturen

MQTT:

URI: mqtt://10.54.32.1

Publish topic temp+humid: <U-NUMMER>/measurement

Subscribe topic voor led: <U-NUMMER>/led

Wifi:

SSID: GraduaatIOT

Password: CloudAndSecurity101

Netwerk: 10.54.32.1/24

## Extra

* Voeg, indien je werkt met node-red, in een dashboard een gauge en een grafiek toe die de temperatuur en vochtigheid laat zien.
* Voeg een drukknop toe aan je hardware opstelling.
* Publish de staat van de knop (0 voor niet ingedrukt, 1 voor ingedrukt) naar het topic   
  <U-NUMMER>/button.
* Zorg ervoor dat je de led kan sturen met de knop (via de MQTT broker, niet rechtstreeks binnen de ESP32 code).
* Werk met een andere student samen zodat de knop van student 1 de led van student 2 bedient.