

# Home Assistant et ESP32

On aurait voulu utiliser le [serveur YunoHost du FabLab](#), mais seule [la version Core de HomeAssistant](#) est disponible sur YunoHost et celle-ci [ne permet pas l'installation de l'add-on ESPHome](#). Une [application en cours de création](#) devrait permettre d'installer le dashboard ESPHome sur YunoHost.

Par ailleurs le serveur YunoHost du FabLab est configuré pour se connecter au réseau wifi du FabLab qui n'est accessible que depuis les salles du FabLab. Étant donné que les ESP32 que l'on veut piloter doivent être sur le même réseau wifi que la Raspberry ESPHome, il est plus flexible d'utiliser un routeur mobile pour pouvoir faire ça de n'importe où dans ou hors de l'IUT.

## Installation de Home Assistant

**Home Assistant peut être installé sur un Raspberry 3B+ moyennant de rajouter 1Go de SWAP**

**MAIS à l'usage** l'installation des paquets, la compilation et l'upload des programmes sur les ESP ont tendance à échouer (ou timeout). Il faut donc lancer les opérations à plusieurs reprises pour qu'elles aboutissent, ce qui rend la **solution peu utilisable**

- [installer Home Assistant OS](#) sur un Raspberry 4 ou 5 de préférence
- Connecter le Raspberry et un PC au même routeur, avec serveur DHCP activé
- [Se connecter depuis le navigateur du PC à l'interface Web](#) :  
<http://homeassistant.local:8123>
  - Déconnecter le PC de tout autre réseau pour éviter les conflits d'IP
  - Attention, connexion en http sur le réseau local
  - En dernier recours, chercher l'IP de la Raspberry sur l'interface du routeur et se connecter à, par ex. <http://192.168.1.100:8123>
- Configurer l'installation
  - Ajout du premier utilisateur
  - Géolocalisation
  - etc.

# Configuration d'un HotSpot Wifi

Home Assistant étant installé sur un Raspberry, on peut exploiter sa carte Wifi pour émettre un réseau wifi plutôt que de dépendre du routeur externe.

Le réseau wifi émis par le Raspberry n'est pas très puissant, il faudra donc que les objets connectés soient dans la même pièce ou à proximité immédiate.

- [Suivre ces instructions](#) : Paramètres --> modules complémentaires --> Boutique des modules complémentaires
  - ou cliquer simplement sur : <http://homeassistant.local:8123/hassio/store>
  - puis --> trois points --> dépôts
- Ajouter <https://github.com/mattlongman/hassio-addons-directory>
- Attendre la mise-jour du store
- Installer le module `Hass.io Access Point`
- Aller dans Paramètres --> modules complémentaires --> Hass.io Access Point --> Configuration
- Définir un SSID et mot-de-passe
- Laisser les autres paramètres par défaut
- Activer le serveur DHCP pour que votre PC et les ESP32 récupèrent une adresse IP automatiquement
- Autoriser l'accès internet pour que vous puissiez avoir internet via le Raspberry lorsque vous connectez votre PC au Hotspot

Le HotSpot wifi ne se monte que si la Raspberry est connectée à Internet via RJ45

## Configuration de ESPHome

- Paramètres --> Modules Complémentaires (add-on) --> Boutique des modules complémentaires --> Installer ESPHome

Les ESP32 avec Home Assistant ne supportent pas le Wifi 5GHz, rester sur du 2,4GHz

- Pour changer le réseau WIFI auquel se connectent les ESP32 --> modifier SECRETS.yaml

## Ajout d'un ESP32 et d'un capteur DHT22

L'ESP32-E firebeetle 2 de DFRobot n'est pas officiellement supporté car son schéma n'a pas été ajouté à platformio.

Les I/O ne peuvent être désignés par leur nom A6.

De manière générale sur ESPHome, préférer les numérotations GPIO plutôt que les noms de pin type D1, A2, etc.

- Connecter l'ESP32 en USB-C au Raspberry
- Ajouter un appareil "New Device"
- Cliquer sur continue
- donner un nom parlant à l'ESP32, par ex. `station_eau_1`
- Sélectionner ESP32
- Valider la clé de chiffrement, elle pourra être récupérée par la suite au besoin
- Choisir une connexion par USB au Raspberry "Plug into the computer running ESPHome Dashboard"
- Sélectionner le port USB où est connecté l'ESP32
- La configuration initiale est créée
- cliquer sur EDIT pour la modifier pour ajouter les capteur
- Voir la discussion <https://community.home-assistant.io/t/esphome-on-the-firebeetle-2-esp32-e/601747/15>
- Trouver le numéro GPIO correspondant au port I/O utilisé, [voir ce tableau](#)
  - Ex. : D6 --> GPIO14
- Trouver le code pour le capteur à ajouter, par exemple [pour le DHT22](#) :

```
# Configuration DHT22
sensor:
  - platform: dht
    pin: GPIO14
    temperature:
      name: "Température FabLab"
    humidity:
      name: "Humidité FabLab"
    update_interval: 60s
```

- Update --> Validate --> Install l'ESP32
- Voici les grandes étapes de compilation et déploiement

```
INFO ESPHome 2024.6.3
```

```
INFO Reading configuration /config/esphome/station-eau-1.yaml...
```

```
INFO Generating C++ source...
```

```
INFO Compiling app...
Processing station-eau-1 (board: esp32dev; framework: arduino; platform:
platformio/espressif32@5.4.0)
-----
Platform Manager: Installing platformio/espressif32 @ 5.4.0
INFO Installing platformio/espressif32 @ 5.4.0
Downloading [ #####] 100%
Unpacking [ #####] 100%
Platform Manager: espressif32@5.4.0 has been installed!
INFO espressif32@5.4.0 has been installed!
...
Library Manager: Resolving dependencies...
INFO Resolving dependencies...
Library Manager: Installing esphome/libsodium @ 1.10018.1
INFO Installing esphome/libsodium @ 1.10018.1
Downloading [ #####] 100%
Unpacking [ #####] 100%
Library Manager: libsodium@1.10018.1 has been installed!
INFO libsodium@1.10018.1 has been installed!
HARDWARE: ESP32 240MHz, 320KB RAM, 4MB Flash
- toolchain-xtensa-esp32 @ 8.4.0+2021r2-patch5
Dependency Graph
|-- AsyncTCP-esphome @ 2.1.3
|-- WiFi @ 2.0.0
|-- FS @ 2.0.0
|-- Update @ 2.0.0
|-- ESPAsyncWebServer-esphome @ 3.2.2
|-- DNSServer @ 2.0.0
|-- ESPmDNS @ 2.0.0
|-- noise-c @ 0.1.4
Compiling .pioenvs/station-eau-1/src/esphome/components/api/api_connection.cpp.o
...
Compiling .pioenvs/station-eau-1/src/esphome/components/wifi/wifi_component_pico_w.cpp.o
Compiling .pioenvs/station-eau-1/src/esphome/core/application.cpp.o
...
Compiling .pioenvs/station-eau-1/src/esphome/core/util.cpp.o
Compiling .pioenvs/station-eau-1/src/main.cpp.o
Building .pioenvs/station-eau-1/bootloader.bin
Creating esp32 image...
Successfully created esp32 image.
```

```
Generating partitions .pioenvs/station-eau-1/partitions.bin
Compiling .pioenvs/station-eau-1/libbeef/AsyncTCP-esphome/AsyncTCP.cpp.o
Archiving .pioenvs/station-eau-1/libbeef/libAsyncTCP-esphome.a
Compiling .pioenvs/station-eau-1/lib64d/WiFi/WiFi.cpp.o
...
Compiling .pioenvs/station-eau-1/FrameworkArduino/wiring_shift.c.o
Archiving .pioenvs/station-eau-1/libFrameworkArduino.a
Linking .pioenvs/station-eau-1/firmware.elf
RAM:  [ =          ] 12.4% (used 40648 bytes from 327680 bytes)
Flash: [ ===== ] 49.3% (used 904333 bytes from 1835008 bytes)
Building .pioenvs/station-eau-1/firmware.bin
Creating esp32 image...
Successfully created esp32 image.
esp32_create_combined_bin([".pioenvs/station-eau-1/firmware.bin"], [".pioenvs/station-eau-1/firmware.elf"])
Wrote 0xee320 bytes to file /data/build/station-eau-1/.pioenvs/station-eau-1/firmware.factory.bin, ready to flash to offset 0x0
esp32_copy_ota_bin([".pioenvs/station-eau-1/firmware.bin"], [".pioenvs/station-eau-1/firmware.elf"])
===== [ SUCCESS] Took 41.34 seconds =====
INFO Successfully compiled program.
esptool.py v4.7.0
Serial port /dev/ttyUSB0
Connecting...
Chip is ESP32-D0WD-V3 (revision v3.0)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
Crystal is 40MHz
MAC: 40:22:d8:66:91:7c
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 460800
Changed.
Configuring flash size...
Auto-detected Flash size: 4MB
Flash will be erased from 0x00010000 to 0x000eefff...
...
Compressed 8192 bytes to 47...
Wrote 8192 bytes (47 compressed) at 0x0000e000 in 0.1 seconds (effective 561.8 kbit/s)...
Hash of data verified.
```

Leaving...

Hard resetting via RTS pin...

INFO Successfully uploaded program.

INFO Starting log output from /dev/ttyUSB0 with baud rate 115200

[11:14:57][I][logger:156]: Log initialized

[11:14:57][C][safe\_mode:079]: There have been 0 suspected unsuccessful boot attempts

[11:14:57][D][esp32.preferences:114]: Saving 1 preferences to flash...

[11:14:57][D][esp32.preferences:143]: Saving 1 preferences to flash: 0 cached, 1 written, 0 failed

[11:14:57][I][app:029]: Running through setup()...

[11:14:57][C][wifi:047]: Setting up WiFi...

[11:14:57][C][wifi:060]: Starting WiFi...

[11:14:57][C][wifi:061]: Local MAC: 40:22:D8:66:91:7C

[11:14:57][D][wifi:481]: Starting scan...

[11:14:57][W][component:157]: Component wifi set Warning flag: scanning for networks

[11:15:03][D][wifi:496]: Found networks:

[11:15:03][I][wifi:540]: - 'homeassistant-rasp3-hotspot' (B8:27:EB:46:5C:65) [redacted] 

[11:15:03][D][wifi:541]: Channel: 6

[11:15:03][D][wifi:542]: RSSI: -48 dB

...

[11:15:09][C][logger:185]: Logger:

[11:15:09][C][logger:186]: Level: DEBUG

[11:15:09][C][logger:188]: Log Baud Rate: 115200

[11:15:09][C][logger:189]: Hardware UART: UART0

[11:15:09][C][captive\_portal:088]: Captive Portal:

[11:15:09][C][mdns:115]: mDNS:

[11:15:09][C][mdns:116]: Hostname: station-eau-1

[11:15:09][C][esphome.ota:073]: Over-The-Air updates:

[11:15:09][C][esphome.ota:074]: Address: station-eau-1.local:3232

[11:15:09][C][esphome.ota:075]: Version: 2

[11:15:09][C][esphome.ota:078]: Password configured

[11:15:09][C][safe\_mode:018]: Safe Mode:

[11:15:09][C][safe\_mode:020]: Boot considered successful after 60 seconds

[11:15:09][C][safe\_mode:021]: Invoke after 10 boot attempts

[11:15:09][C][safe\_mode:023]: Remain in safe mode for 300 seconds

[11:15:09][C][api:139]: API Server:

[11:15:09][C][api:140]: Address: station-eau-1.local:6053

[11:15:09][C][api:142]: Using noise encryption: YES

[11:15:57][I][safe\_mode:041]: Boot seems successful; resetting boot loop counter

```
[11:15:57][D][esp32.preferences:114]: Saving 1 preferences to flash...  
[11:15:57][D][esp32.preferences:143]: Saving 1 preferences to flash: 0 cached, 1 written, 0  
failed
```

- Noter l'adresse de l'ESP32 sur le réseau wifi local :

```
[11:15:09][C][api:140]: Address: station-eau-1.local:6053
```

- Aller dans Paramètres --> Appareils et Services --> ESPHome -->

## ESPHome



Veuillez saisir les paramètres de connexion de votre nœud [ESPHome](#).

Hôte\*

station\_eau\_1.local

Port

6053

VALIDER

- Aller dans Paramètres --> Appareils et Services --> ESPHome --> sélectionner l'ESP32 par son nom, par ex. station\_eau\_1
- Les données du capteur s'affichent sur tout dashboard qui affiche la salle où le capteur est positionné

## Débuggage

- Impossible de flasher un ESP32 `Error: Could not find one of 'package.json' manifest files in the package`
  - Les dépendances ce sont mal installées/téléchargées
    - Solution 1 : [désinstaller et réinstaller ESPHome](#)
    - Solution 2 : [supprimer le dossier](#) `/home/esphome/.platformio/packages`

## Sources

- <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>
- [https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e\\_esp32-wroom-32ue\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf)

- <https://rdr-it.io/domotique/home-assistant-configuration-en-tant-que-point-dacces-wifi/>
- 

Revision #11

Created 27 June 2024 09:52:44 by admin\_idf

Updated 18 February 2025 08:28:38 by admin\_idf