

On peut tout à fait programmer un ESP depuis l'IDE Arduino. Cependant, l'arrivée du framework ESPHome a redéfini la donne : le recours pour programmer un ESP à un fichier descriptif en Yaml plutôt qu'à un code C++ a permis d'améliorer la qualité des programmes produits et élargir l'assiette des utilisateurs en intégrant les débutant·e·s et les personnes en apprentissage.

Principaux ESP

- ESP8266 : microcontrôleur Wi-Fi largement utilisé pour les projets IoT. Il est connu pour sa faible consommation d'énergie, son faible coût et sa compatibilité avec de nombreuses plates-formes de développement.
- ESP8285 : variante de l'ESP8266 avec un petit espace de stockage Flash intégré, utile pour les projets où l'espace de stockage est limité.
- ESP32 : microcontrôleur plus avancé, doté de capacités de Wi-Fi et de Bluetooth intégrées. Il offre également une puissance de traitement plus élevée, davantage de broches GPIO et une mémoire plus importante par rapport à l'ESP8266.

Développement avec ESPHome

Les ESP peuvent se programmer avec l'IDE Arduino en C++. Comme les Raspberry Pi Pico, les ESP peuvent se programmer également en LUA et en micro python.

ESPHome simplifie considérablement le processus de développement en permettant aux développeurs de configurer les fonctionnalités de leur dispositif IoT à l'aide de fichiers de configuration YAML, plutôt que de devoir écrire manuellement le code source C++ pour le microcontrôleur. Les fichiers de configuration ESPHome définissent les broches matérielles, les capteurs, les actionneurs, les communications réseau, etc.

Une fois que vous avez créé votre configuration ESPHome, vous pouvez compiler et flasher le firmware sur votre microcontrôleur ESP à l'aide de l'outil ESPHome. Cela simplifie grandement le processus de développement, en particulier pour ceux qui ne sont pas familiers avec la programmation embarquée.

ESPHome offre une intégration directe avec Home Assistant, la plateforme domotique open source. C'est l'environnement choisi pour la majorité des projets de domotique et de surveillance environnementale développé au BIK'LAB.

- [Construction d'un nœud basique de mesures environnementales avec ESPHome](#)
 - [ESPHome / BME 280](#)
 - [ESPHome / BME 680](#)
 - [ESPHome / ds18b20](#)
 - [ESPHome / SHT31](#)
 - [ESPHome / tsl2561](#)

From:

<https://wiki.lebiklab.fr/> - **Wiki Le BIK'LAB**

Permanent link:

<https://wiki.lebiklab.fr/doku.php?id=projets:home-assistant:esphome:start&rev=1698163604>

Last update: **04/04/2024 15:35**

