



# **Club de Robotique et d'Electronique Programmable de Ploemeur**

**Support des ateliers  
d'Electronique**


Nicolas Le Guerroué

30 septembre 2024


# Table des matières


<b>Liste des figures</b>	<b>2</b>
<b>Préambule</b>	<b>3</b>
<b>Glossaire</b>	<b>4</b>
<b>1 Utilisation de VSCode</b>	<b>5</b>
1.1 Installation de Visual Studio Code . . . . .	5
1.1.1 Téléchargement . . . . .	5
1.1.2 Installation . . . . .	5
1.2 Installation de PlatformIO . . . . .	5
1.2.1 Installation de l'extension PlatformIO . . . . .	6
1.2.2 Installation des drivers . . . . .	6
1.3 Configuration du projet pour la carte ESP12 . . . . .	9
1.3.1 Création d'un nouveau projet PlatformIO . . . . .	9
1.3.2 Structure du projet . . . . .	11
1.4 Écriture et téléversement du code sur l'ESP12 . . . . .	11
1.4.1 Écriture du code . . . . .	11
1.4.2 Compilation et téléversement . . . . .	12
1.5 Installation d'une bibliothèque sous PlatformIO . . . . .	13
1.6 Mise en français du logiciel VSCode . . . . .	17

# Liste des figures


1.1	Menu des extensions . . . . .	6
1.2	Recherche de l'extension . . . . .	6
1.3	Un ESP12 avec un module CP2102 . . . . .	7
1.4	Interface de téléchargement pour driver CP2102 . . . . .	7
1.5	Installation du driver CP2102 . . . . .	8
1.6	Un ESP12 avec un module CH340 . . . . .	8
1.7	Interface de téléchargement pour driver CH340 . . . . .	9
1.8	Installation du driver CH340 . . . . .	9
1.9	Ouverture de l'environnement . . . . .	10
1.10	Sélection de la carte . . . . .	11
1.11	Compilation du code . . . . .	12
1.12	Message signifiant le téléversement . . . . .	13
1.13	Ouverture du gestionnaire de bibliothèque PlatformIO . . . . .	14
1.14	Accès au gestionnaire de bibliothèque PlatformIO . . . . .	15
1.15	Ajout d'une bibliothèque . . . . .	15
1.16	Sélection du projet pour la bibliothèque . . . . .	16
1.17	Contenu du fichier  platformio.ini . . . . .	16
1.18	Mise en français du logiciel VSCode . . . . .	17


# Préambule


 Document réalisé en  $\text{\LaTeX}$  par Nicolas Le Guerroué pour le Club de Robotique et d'Électronique Programmable de Ploemeur (CREPP)


 Version du 30 septembre 2024


 Taille de police : 11pt (carlito)

 N'hésitez pas à faire des retours sur le document, cela permettra de l'améliorer

 [nicolasleguerroue@gmail.com](mailto:nicolasleguerroue@gmail.com)

 <https://github.com/CREPP-PLOEMEUR><sup>1</sup>

 Permission vous est donnée de copier, distribuer et/ou modifier ce document sous quelque forme et de quelque manière que ce soit.

 **Dans la mesure du possible, évitez d'imprimer ce document si ce n'est pas nécessaire. Il est optimisé pour une visualisation sur un ordinateur et contient beaucoup d'images (18 images)**

---

1. Click-droit et **Copier l'adresse du lien**

# Section 1

## Utilisation de VSCode

Dans ce document, nous allons décrire les étapes nécessaires pour installer Visual Studio Code (VS-Code), PlatformIO, et leur utilisation avec une carte ESP12. Ces outils permettent un développement facile de firmware pour des microcontrôleurs comme l'ESP12, qui est basé sur le populaire ESP8266.

### Installation de Visual Studio Code

Visual Studio Code (VSCode) est un éditeur de texte léger et puissant qui prend en charge un large éventail de langages de programmation et d'extensions. Suivez ces étapes pour l'installer :

#### Téléchargement

- ▶ Accédez à la page de téléchargement officielle de VSCode : <https://code.visualstudio.com/>
- ▶ Sélectionnez la version appropriée pour votre système d'exploitation (Windows, macOS, Linux).
- ▶ Téléchargez l'installateur et lancez-le.

#### Installation

- ▶ Suivez les instructions de l'installateur.
- ▶ À la fin de l'installation, cochez l'option pour ouvrir VSCode.

### Installation de PlatformIO

PlatformIO est une extension de Visual Studio Code qui permet de gérer les environnements de développement pour divers microcontrôleurs, y compris l'ESP12. Voici les étapes pour installer PlatformIO.

## Installation de l'extension PlatformIO

- ▶ Ouvrez Visual Studio Code.
- ▶ Allez dans le panneau des extensions en cliquant sur l'icône d'extensions sur le côté gauche ( 5ème en partant du haut ) ou utilisez le raccourci clavier **Ctrl+Shift+X** .

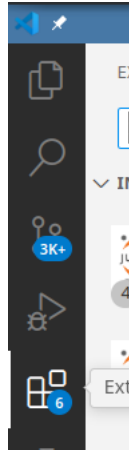


Figure 1.1 – Menu des extensions

- ▶ Dans la barre de recherche, tapez *PlatformIO IDE*.

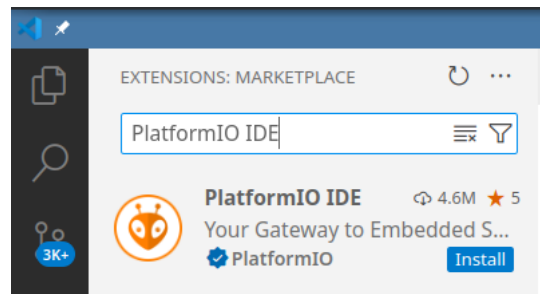


Figure 1.2 – Recherche de l'extension

- ▶ Cliquez sur *Installer* pour installer l'extension.
- ▶ Une fois l'installation terminée, redémarrez VSCode si nécessaire.



Pour les personnes ayant déjà une carte ESP-12 NodeMCU, il est possible de poursuivre le document. Le cas échéant, il faudra continuer le document une fois que la carte sera en votre possession.

## Installation des drivers

Il existe principalement 2 versions de drivers de la carte ESP12 :

- Le driver CP1024 On le reconnaît à sa forme carré sur la carte ESP12, juste au dessus du port USB.

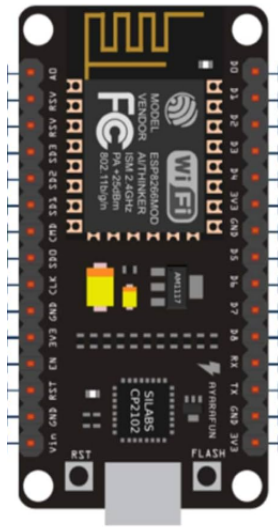


Figure 1.3 – Un ESP12 avec un module CP2102

Ce driver est plus ancien mais on peut le retrouver assez souvent. Voici la procédure d'installation :

- Se rendre sur le site à l'adresse suivante :

<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers>

- Cliquer sur la section **Download**

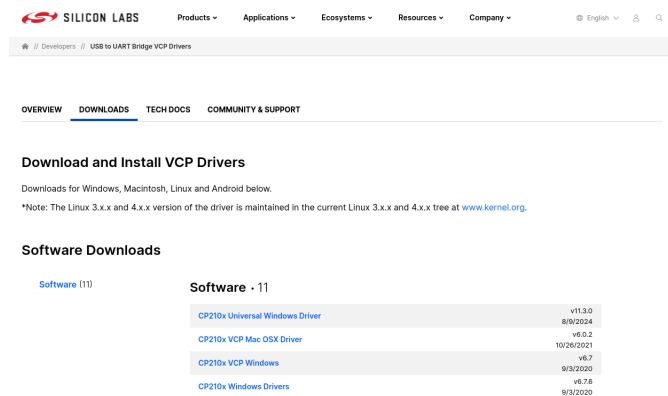


Figure 1.4 – Interface de téléchargement pour driver CP2102

- Cliquer sur **CP210x Windows Drivers**
- Extraire l'archive
- Exécuter le fichier **CP210xVCPInstaller\_64.exe**

Nom	Modifié le	Type	Taille
Aujourd'hui			
SLAB_License_Agreement_VCP_Windows.txt	29/09/2024 22:22	Document texte	9 Ko
slabvcp.inf	29/09/2024 22:22	Informations sur l'ins...	8 Ko
v6-7-6-driver-release-notes.txt	29/09/2024 22:22	Document texte	16 Ko
CP210xVCPInstaller_x64.exe	29/09/2024 22:22	Application	1 026 Ko
CP210xVCPInstaller_x86.exe	29/09/2024 22:22	Application	903 Ko
dpinst.xml	29/09/2024 22:22	xmlfile	12 Ko
slabvcp.cat	29/09/2024 22:22	Catalogue de sécurité	11 Ko
x64	29/09/2024 22:22	Dossier de fichiers	
x86	29/09/2024 22:22	Dossier de fichiers	

Figure 1.5 – Installation du driver CP2102

### ► Le driver CH340

On le reconnaît à sa forme rectangulaire sur la carte ESP12



Figure 1.6 – Un ESP12 avec un module CH340

Ce driver est plus récent et est de plus en plus répandu. Cependant, si vous devez l'installer, voici la procédure :

- Se rendre sur le site <https://sparks.gogo.co.nz/ch340.html>





The CH340 chip is used by a number of Arduino compatible boards to provide USB connectivity, you may need to install a driver, don't panic, it's easier than falling off a log, and much less painful.

## Windows

(Manufacturer's Chinese Info Link)

- Download the [Windows CH340 Driver](#)
- Unzip the file
- Run the installer which you unzipped
- In the Arduino IDE when the CH340 is connected you will see a COM Port in the Tools > Serial Port menu, the COM number for your device may vary depending on your system.

Figure 1.7 – Interface de téléchargement pour driver CH340

- Cliquer sur **Download the Windows CH340 Driver**
- Extraire l'archive
- Exécuter le fichier et cliquer sur **Install**

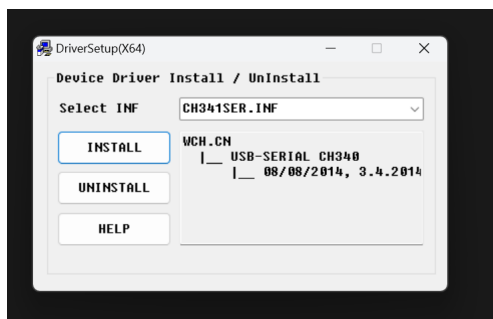


Figure 1.8 – Installation du driver CH340

## Configuration du projet pour la carte ESP12

Voici comment configurer un projet PlatformIO pour l'ESP12.

### Création d'un nouveau projet PlatformIO

- ▶ Ouvrez Visual Studio Code avec l'extension PlatformIO installée.
- ▶ Cliquez sur l'icône de PlatformIO dans la barre latérale de gauche pour accéder à la commande rapide de PlatformIO.

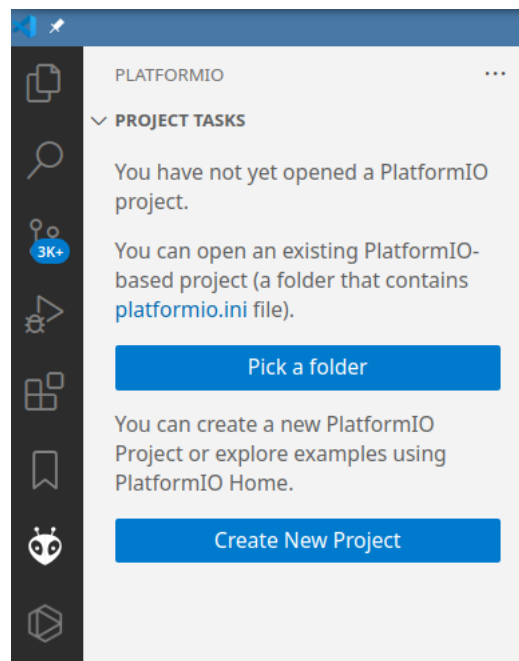


Figure 1.9 – Ouverture de l'environnement

- ▶ Sélectionnez *New Project*.
- ▶ Donnez un nom à votre projet.
- ▶ Dans la section *Board*, recherchez et sélectionnez *ESP12E* ou *ESP8266*.
- ▶ Sélectionnez le framework *Arduino*.

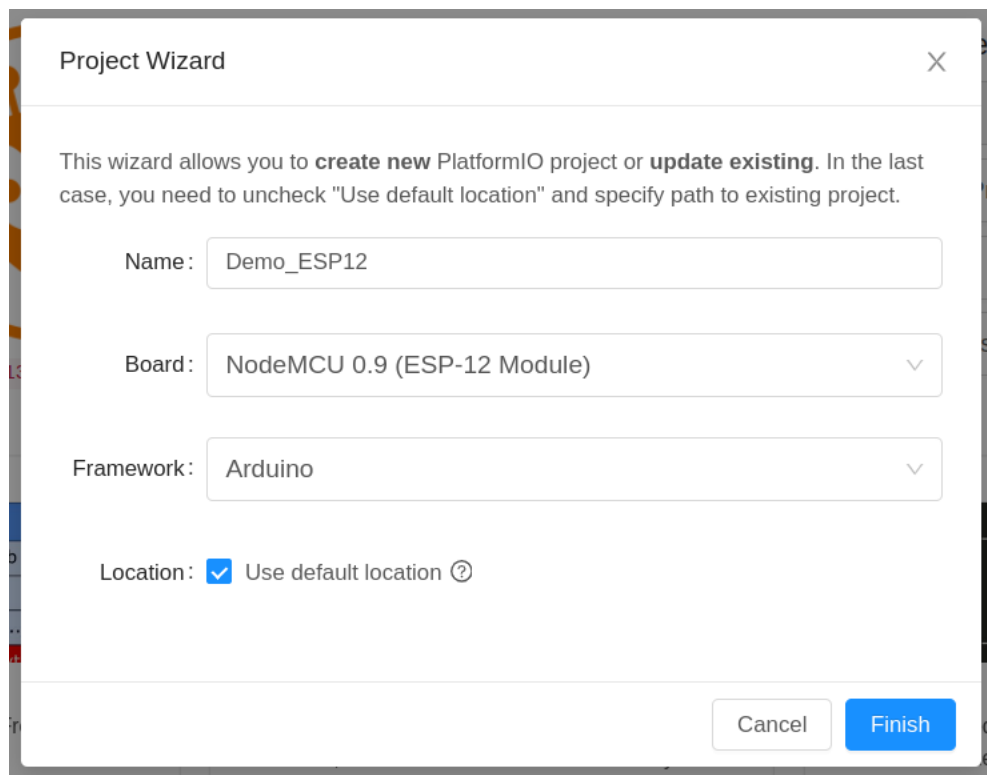


Figure 1.10 – Sélection de la carte

- Cliquez sur *Finish* pour générer le projet.

## Structure du projet

Une fois le projet créé, vous verrez une structure de dossiers typique pour un projet PlatformIO :

- **src/** : Contient les fichiers source, y compris le fichier `main.cpp` où vous écrirez votre code.
- **include/** : Contient les fichiers d'en-tête (`.h`).
- **platformio.ini** : Fichier de configuration du projet.

## Écriture et téléversement du code sur l'ESP12

### Écriture du code

Le fichier `main.cpp` situé dans le dossier `src/` est le fichier principal où vous allez écrire votre code. Voici un exemple simple qui fait clignoter une LED connectée à la broche `D4` de l'ESP12 :

```
#include <Arduino.h>

void setup()
{
    pinMode(D4, OUTPUT);
}

void loop()
{
    digitalWrite(D4, HIGH); // Allume la LED
    digitalWrite(D4, LOW);  // Éteindre la LED
    delay(100);             // Attendre 0.1 seconde
}
```

Code 1 - Code minimaliste

## Compilation et téléversement

- Branchez votre carte ESP12 à votre ordinateur via un câble USB.
- Cliquez sur l'icône de *Check* dans la barre inférieure de PlatformIO pour compiler votre code.

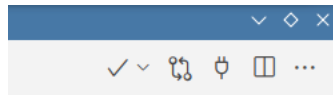


Figure 1.11 – Compilation du code



L'explication de la compilation se fera à travers les ateliers d'électronique.  
L'objectif de ce tutoriel est d'installer correctement VSCode et ses dépendances.

- Une fois la compilation réussie, cliquez sur l'icône de *Flèche* pour sélectionner **Upload**<sup>1</sup> pour téléverser le code sur la carte.

Un texte défilant apparaît en bas de la fenêtre VSCode pour indiquer la progression du chargement du code sur la carte.

1. Le raccourci  **ctrl+Alt+u** au clavier.

```
Compressed 20000 bytes to 10000...  
Writing at 0x00000000... (8 %)  
Writing at 0x00004000... (16 %)  
Writing at 0x00008000... (25 %)  
Writing at 0x0000c000... (33 %)  
Writing at 0x00010000... (41 %)  
Writing at 0x00014000... (50 %)  
Writing at 0x00018000... (58 %)  
Writing at 0x0001c000... (66 %)  
Writing at 0x00020000... (75 %)  
Writing at 0x00024000... (83 %)
```

Figure 1.12 – Message signifiant le téléversement

Lorsque le défilement s'arrête, le code est envoyé et la LED bleue devrait clignoter à 5Hz.

## Installation d'une bibliothèque sous PlatformIO

Pour installer une bibliothèque sous PlatformIO, il suffit de se rendre dans le gestionnaire de bibliothèque :

- ▶ Cliquer sur l'icône PlatformIO
- ▶ Cliquer sur **Librarie**

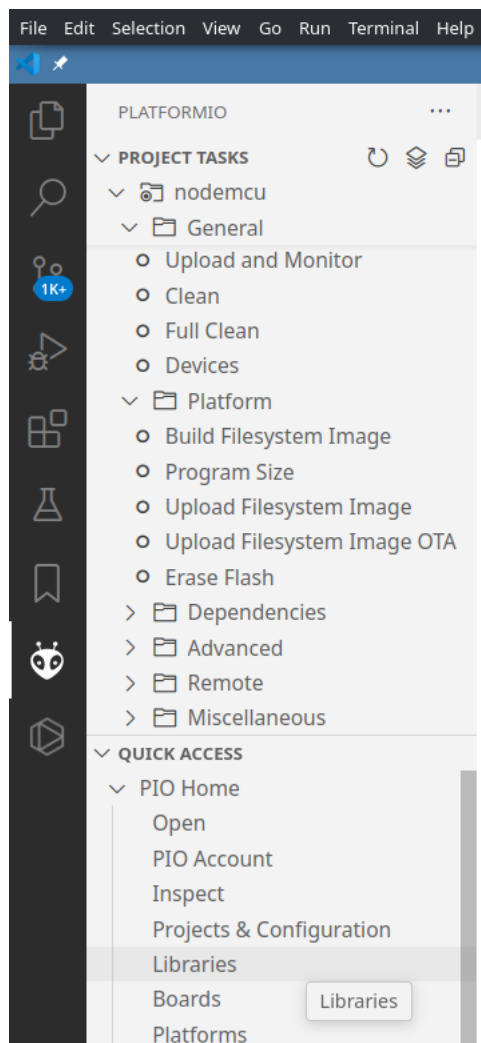


Figure 1.13 – Ouverture du gestionnaire de bibliothèque PlatformIO

Nous prendrons l'exemple de la bibliothèque  **SSD1306** , qui permet d'utiliser un écran OLED basé sur le circuit **SSD1306** .

Il suffit d'écrire le nom de la bibliothèque dans la barre de recherche :

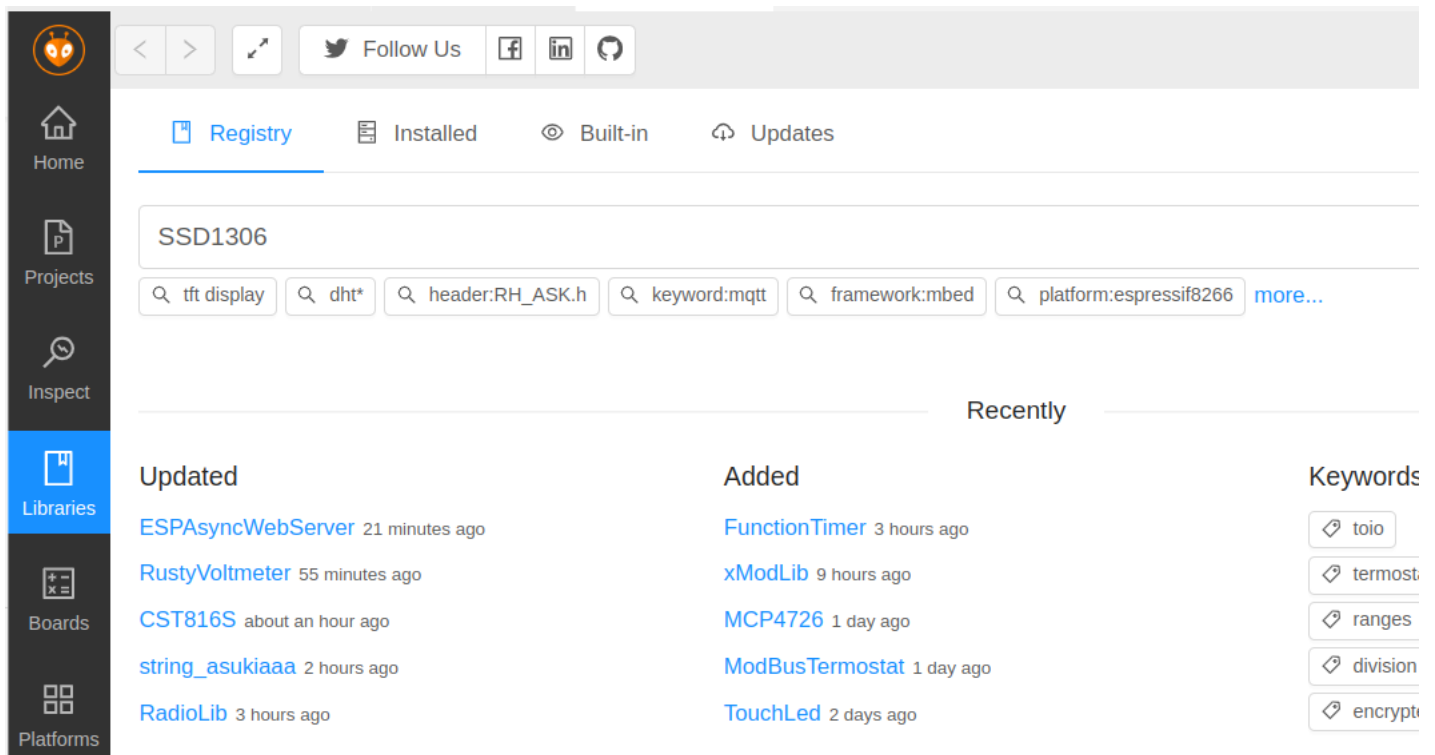


Figure 1.14 – Accès au gestionnaire de bibliothèque PlatformIO

Puis de cliquer sur la première bibliothèque  **SSD1306** et de cliquer sur **Add to Project**.

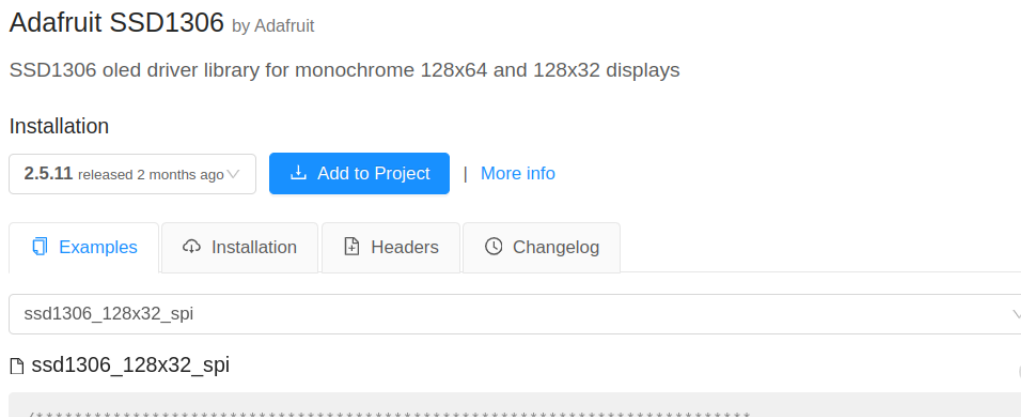


Figure 1.15 – Ajout d'une bibliothèque

A ce moment là, on va vous demander de lier la bibliothèque à un projet. Sélectionner sur le projet sur lequel vous utiliser un écran OLED **SSD1306**.

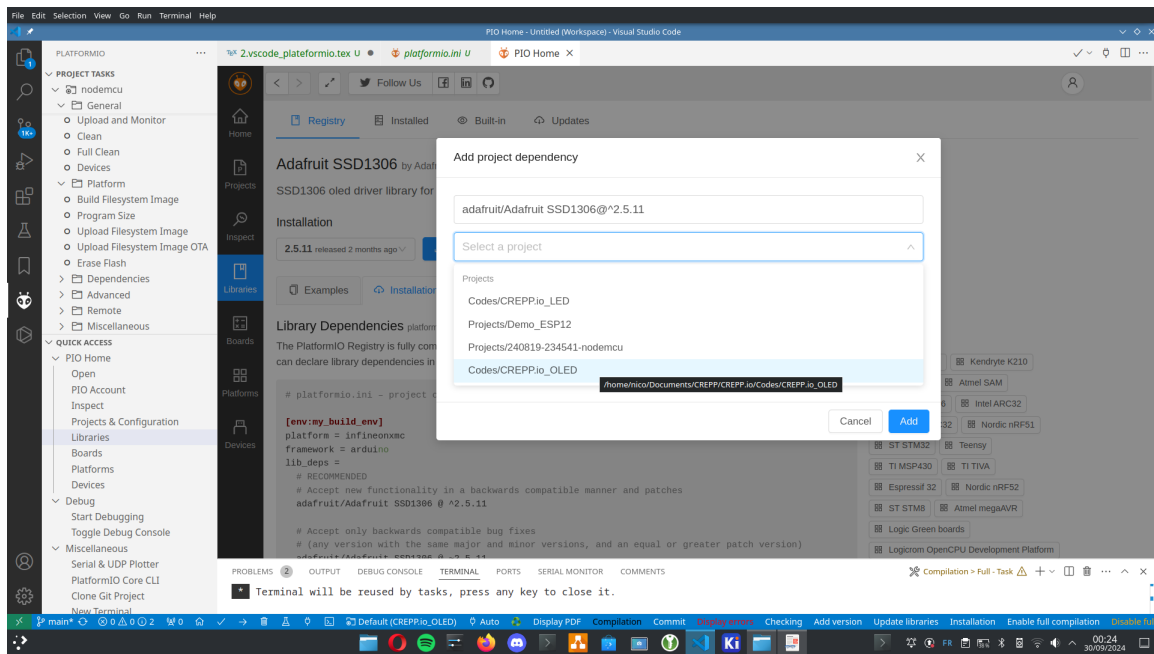


Figure 1.16 – Sélection du projet pour la bibliothèque

Le grand avantage de cette méthode est que pour tout code ouvert via PlatformIO, si des bibliothèques sont nécessaires, elles seront automatiquement ajoutées.

Le fichier `platformio.ini`, situé à la racine du projet, contient la liste des bibliothèques du projet :

```

CREPPio_OLED > platformio.ini
1  ; PlatformIO Project Configuration File
2  ;
3  ; Build options: build flags, source filter
4  ; Upload options: custom upload port, speed and extra flags
5  ; Library options: dependencies, extra library storages
6  ; Advanced options: extra scripting
7  ;
8  ; Please visit documentation for the other options and examples
9  ; https://docs.platformio.org/page/projectconf.html
10
11 [env:nodemcu]
12 platform = espressif8266
13 board = nodemcu
14 framework = arduino
15 lib_deps = adafruit/Adafruit SSD1306@^2.5.11
16

```

Figure 1.17 – Contenu du fichier `platformio.ini`

## Mise en français du logiciel VSCode



Enfin, pour mettre en français le logiciel, vous pouvez installer l'extension suivante :

 **French language Pack for Visual Studio Code** :

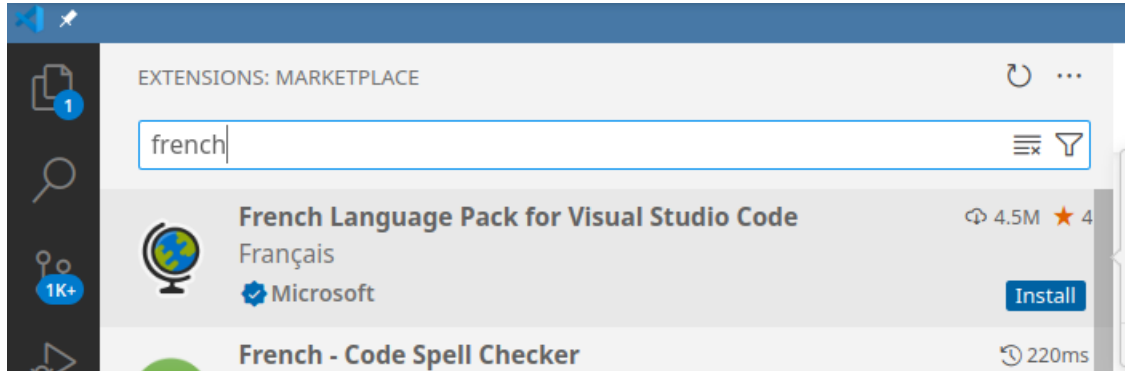


Figure 1.18 – Mise en français du logiciel VSCode