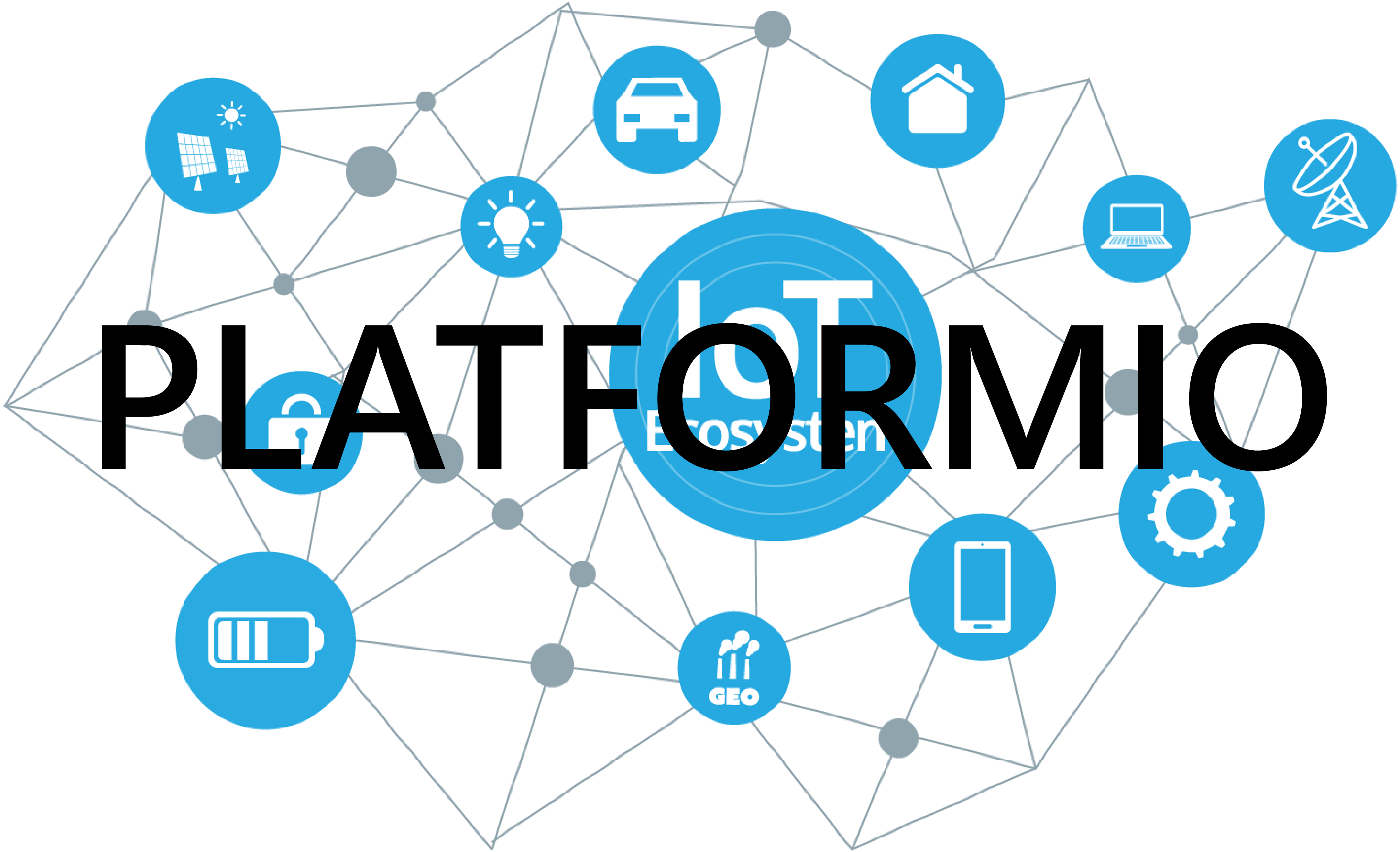
PLATFORMIO



Platformio

- **Apakah itu?**
  - PlatformIO is an open source ecosystem for IoT development
  - Cross-platform IDE and unified debugger.
  - Remote unit testing and firmware updates (http://platformio.org/)
- **Spesifikasi**
  - **Cross-platform** build system without external dependencies to the OS software: 400+ embedded boards, 15+ development platforms, 10+ frameworks
  - C/C++ Intelligent Code Completion and Smart Code Linter for rapid professional development
  - Multi-projects workflow with multiple panes and Themes support with dark and light colors
  - Built-in Terminal with PlatformIO Core and powerful Serial Port Monitor

Platformio

- **Kebutuhan Sistem**
  - OS : Windows, macOS, Linux, FreeBSD, Linux ARMv6+
  - Python Interpreter : Python 3.5 above
  - CLang Interpreter : 3.9.1 version
- **PERINGATAN**
  MATIKAN SEMUA ANTIVIRUS DAN FIREWALL SEBELUM MEMULAI INSTALASI
  SEMUA PROGRAM RUN AS ADMINISTRATOR

Embedded Development. Easier Than Ever.

PlatformIO is well suited for embedded development and has pre-configured settings for the most popular Embedded Boards.

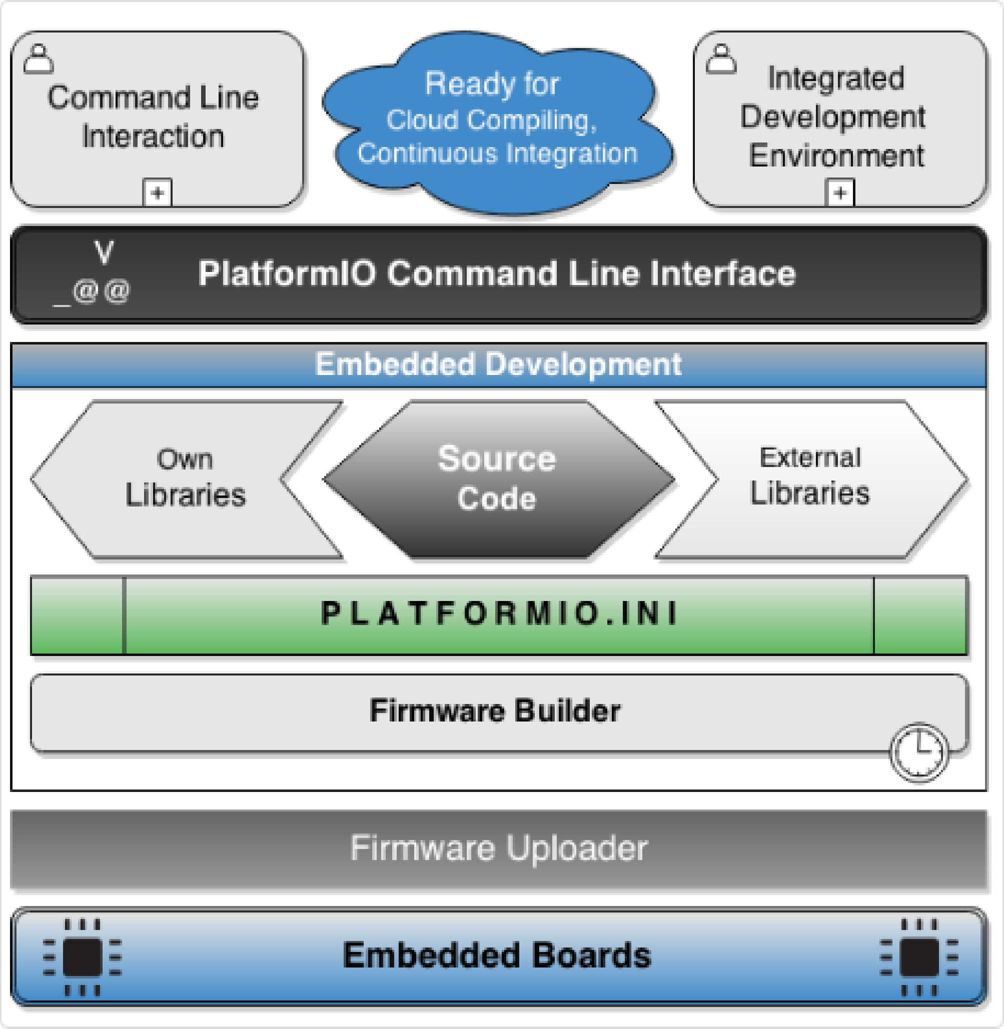
✓ Colourful command-line output

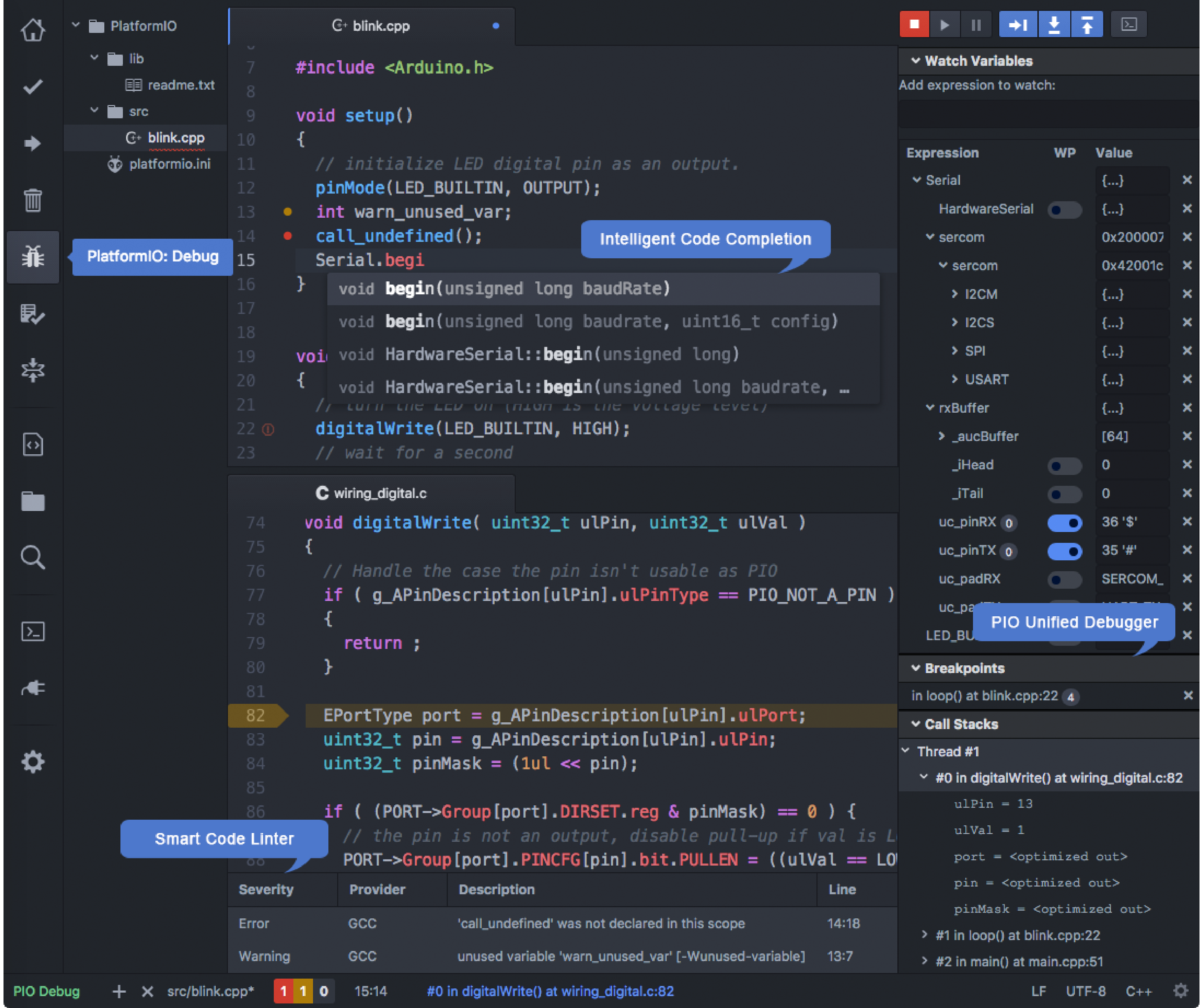
✓ IDE Integration with *Cloud9, Codeanywhere, Eclipse Che, Atom, CLion, CodeBlocks, Eclipse, Emacs, NetBeans, Qt Creator, Sublime Text, Vim, Visual Studio*

✓ Cloud compiling and Continuous Integration with *AppVeyor, Circle CI, Drone, Shippable, Travis CI*

✓ Built-in Serial Port Monitor and configurable build -flags/-options

👍 Pre-built toolchains, frameworks for the popular development platforms





PlatformIO: Debug

Intelligent Code Completion

```
 7  #include <Arduino.h>
 8
 9  void setup()
10  {
11    // initialize LED digital pin as an output.
12    pinMode(LED_BUILTIN, OUTPUT);
13    int warn_unused_var;
14    call_undefined();
15    Serial.begi
16  }
17
18
19  voi
20  {
21    // turn the LED on (HIGH is the voltage level)
22    digitalWrite(LED_BUILTIN, HIGH);
23    // wait for a second
```

```
void begin(unsigned long baudRate)
void begin(unsigned long baudrate, uint16_t config)
void HardwareSerial::begin(unsigned long)
void HardwareSerial::begin(unsigned long baudrate, …
```

```
74  void digitalWrite( uint32_t ulPin, uint32_t ulVal )
75  {
76    // Handle the case the pin isn't usable as PIO
77    if ( g_APinDescription[ulPin].ulPinType == PIO_NOT_A_PIN )
78    {
79      return ;
80    }
81
82    EPortType port = g_APinDescription[ulPin].ulPort;
83    uint32_t pin = g_APinDescription[ulPin].ulPin;
84    uint32_t pinMask = (1ul << pin);
85
86    if ( (PORT->Group[port].DIRSET.reg & pinMask) == 0 ) {
87      // the pin is not an output, disable pull-up if val is L
88      PORT->Group[port].PINCFG[pin].bit.PULLEN = ((ulVal == LO
```

Smart Code Linter

Severity	Provider	Description	Line
Error	GCC	'call_undefined' was not declared in this scope	14:18
Warning	GCC	unused variable 'warn_unused_var' [-Wunused-variable]	13:7

Watch Variables

Add expression to watch:

Expression	WP	Value
Serial		{...}
HardwareSerial	○	{...}
sercom		0x200007
sercom		0x42001c
I2CM		{...}
I2CS		{...}
SPI		{...}
USART		{...}
rxBuffer		{...}
_aucBuffer		[64]
_iHead	○	0
_iTail	○	0
uc_pinRX 0	●	36 '$'
uc_pinTX 0	●	35 '#'
uc_padRX	○	SERCOM_
uc_pa		
LED_BU		

PIO Unified Debugger

Breakpoints

in loop() at blink.cpp:22 4

Call Stacks

```
Thread #1
  #0 in digitalWrite() at wiring_digital.c:82
      ulPin = 13
      ulVal = 1
      port = <optimized out>
      pin = <optimized out>
      pinMask = <optimized out>
  #1 in loop() at blink.cpp:22
  #2 in main() at main.cpp:51
```

PIO Debug + × src/blink.cpp* 1 1 0 15:14 #0 in digitalWrite() at wiring_digital.c:82 LF UTF-8 C++

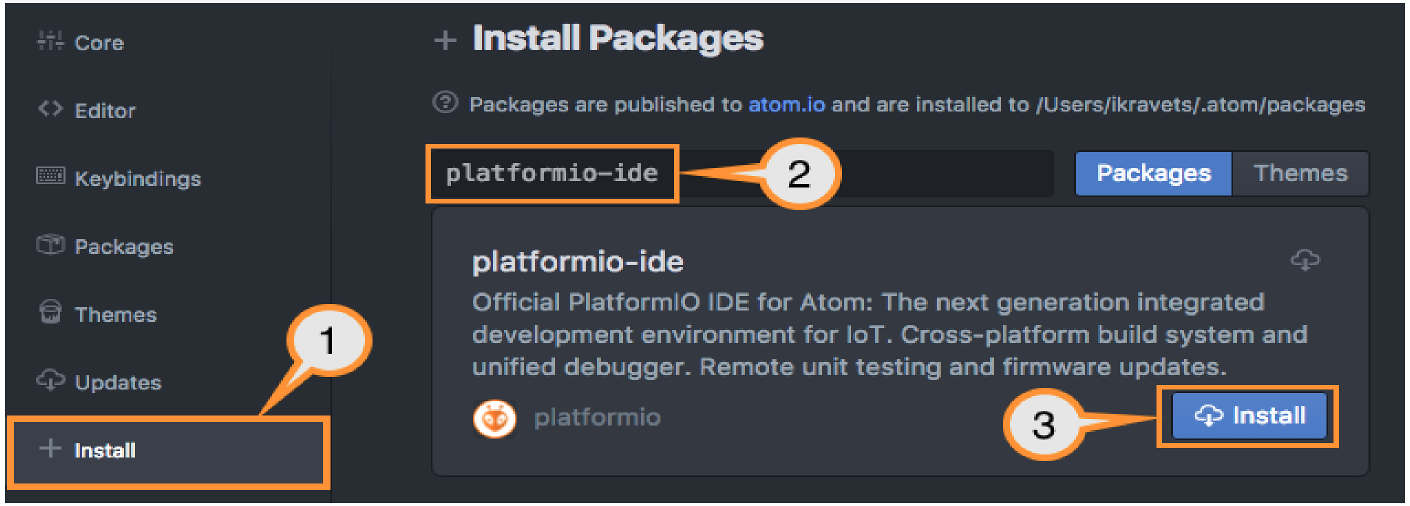
Platformio + Atom Editor

1. Install Python 3.7.9 (32bit atau 64bit sesuaikan OS Anda)
   https://www.python.org/downloads/release/python-379/
2. Install CLang 3.9.1 (**Install di path tanpa spasi. Restart PC Anda**)
   http://releases.llvm.org/download.html

Platformio + Atom Editor

3. Install Atom Editor. Pilih portable version saja
   https://github.com/atom/atom/releases/download/v1.54.0/atom-windows.zip
4. Install Platformio

```
Menu: File > Settings > Install
```



+ Install Packages

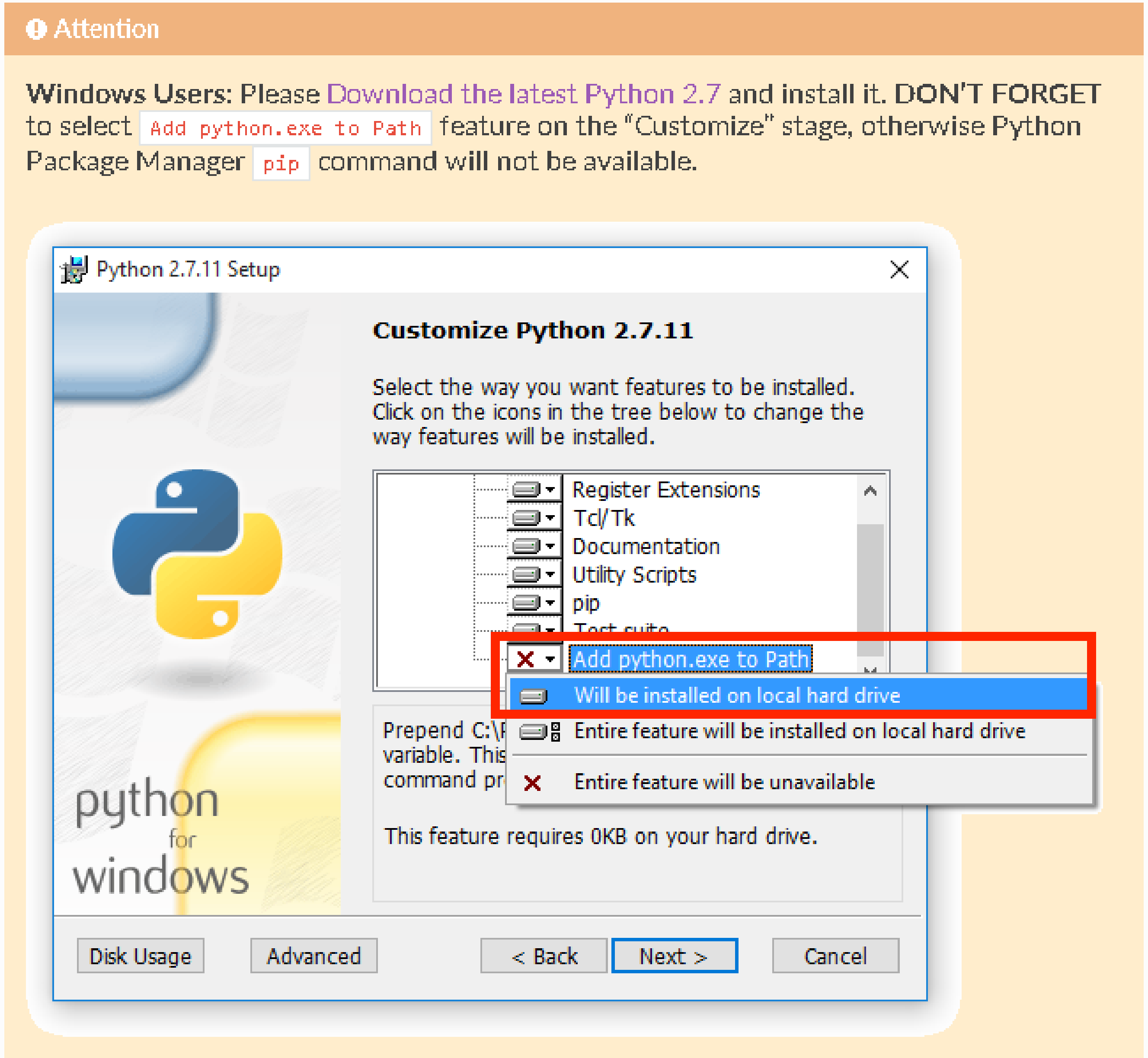
Packages are published to atom.io and are installed to /Users/ikravets/.atom/packages

platformio-ide

platformio-ide

Official PlatformIO IDE for Atom: The next generation integrated development environment for IoT. Cross-platform build system and unified debugger. Remote unit testing and firmware updates.

Jangan lupa ini!



Attention

Windows Users: Please Download the latest Python 2.7 and install it. DON'T FORGET to select `Add python.exe to Path` feature on the "Customize" stage, otherwise Python Package Manager `pip` command will not be available.

Python 2.7.11 Setup

Customize Python 2.7.11

Select the way you want features to be installed. Click on the icons in the tree below to change the way features will be installed.

- Register Extensions
- Tcl/Tk
- Documentation
- Utility Scripts
- pip
- Test suite
- Add python.exe to Path

Will be installed on local hard drive

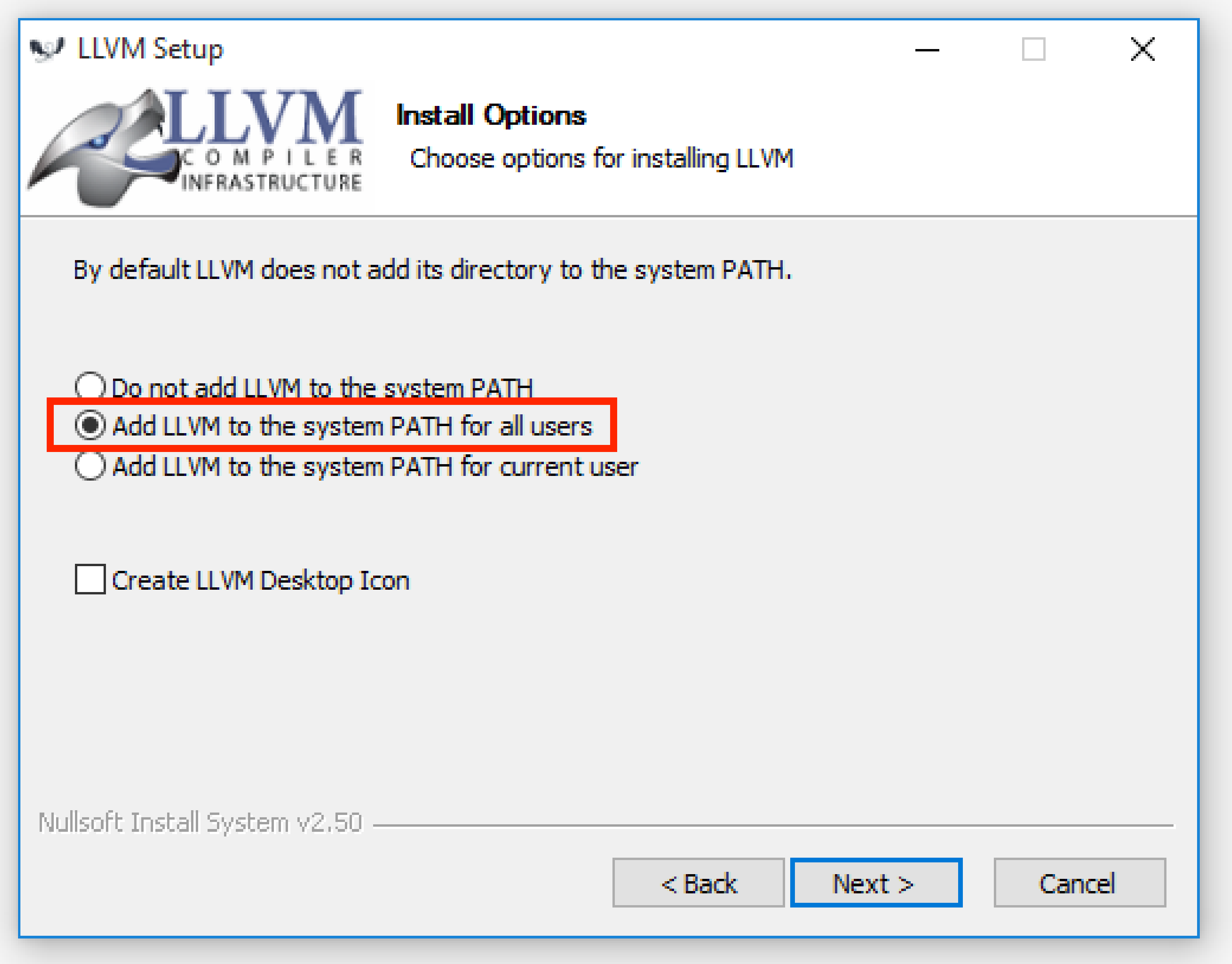
Entire feature will be installed on local hard drive

Entire feature will be unavailable

Prepend C:\P
variable. This
command pr

This feature requires 0KB on your hard drive.

Jangan lupa ini!



Install Options

Choose options for installing LLVM

By default LLVM does not add its directory to the system PATH.

○ Do not add LLVM to the system PATH

● Add LLVM to the system PATH for all users

○ Add LLVM to the system PATH for current user

□ Create LLVM Desktop Icon

Driver CP2102

1. Download dan *install* dari :

https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

Driver CH340G

1. Download dan *install* dari :

https://kelasrobot.com/cara-install-usb-driver-ch340g-ch340-untuk-arduino/

https://www.dropbox.com/s/q0u00s6yd4bhw8w/USB Driver CH340G.rar?dl=0

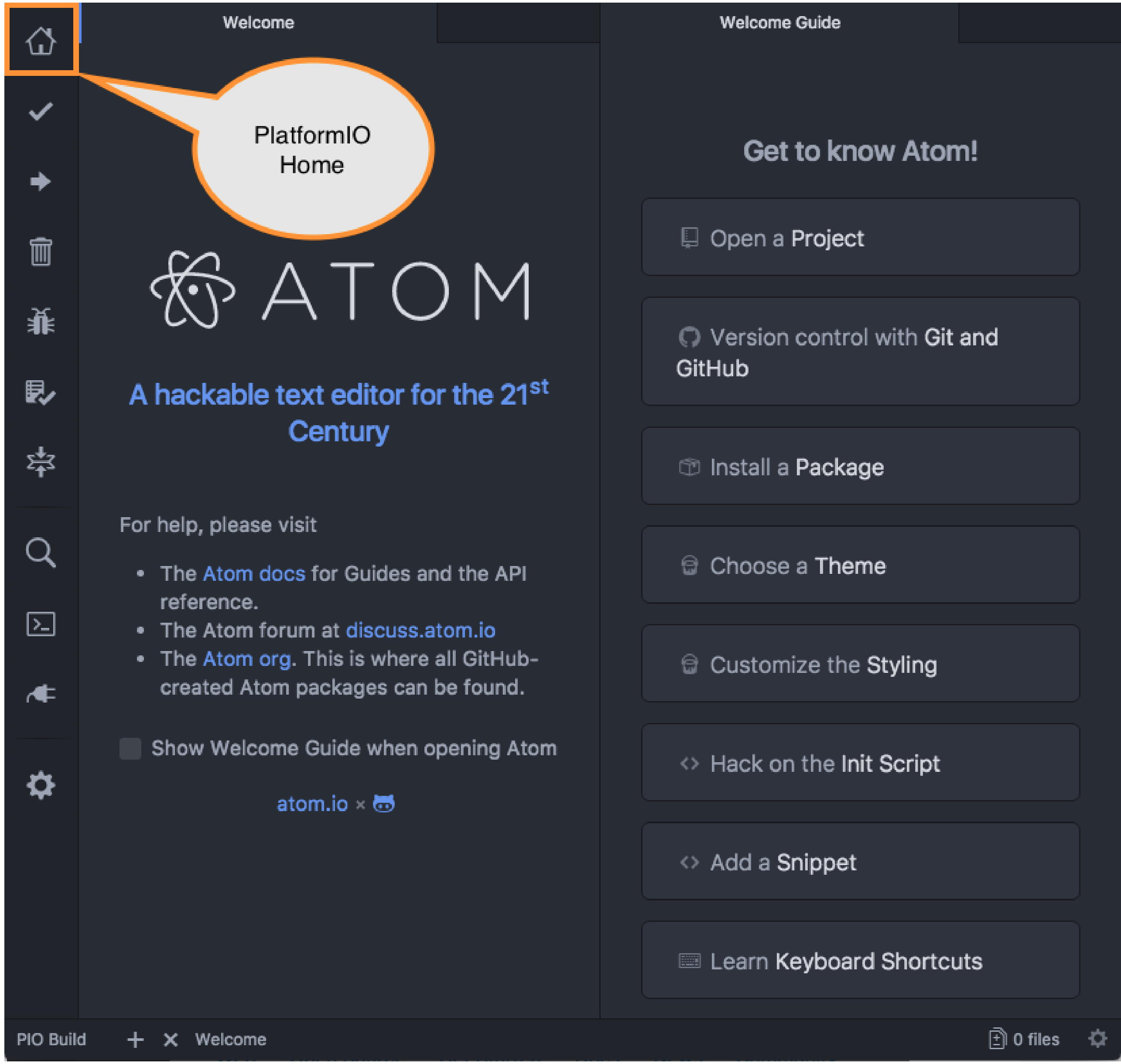
SDK ARDUINO ESP8266 PLATFORMIO

- Proses instalasinya lama banget,hehehehe….
- Instruktur akan memberikan file SDK dalam format **zip**
- Ekstrak ke folder di PC Anda. **Nama folder tidak boleh ada spasi!!!**
- Misal : **D:\PIO**

STOP

Platformio Quick Start

1. Jalankan Atom, Platformio akan mulai terbuka. Harap bersabar bagi yang *low spec* PC



A hackable text editor for the 21st Century

For help, please visit

- The Atom docs for Guides and the API reference.
- The Atom forum at discuss.atom.io
- The Atom org. This is where all GitHub-created Atom packages can be found.

□ Show Welcome Guide when opening Atom

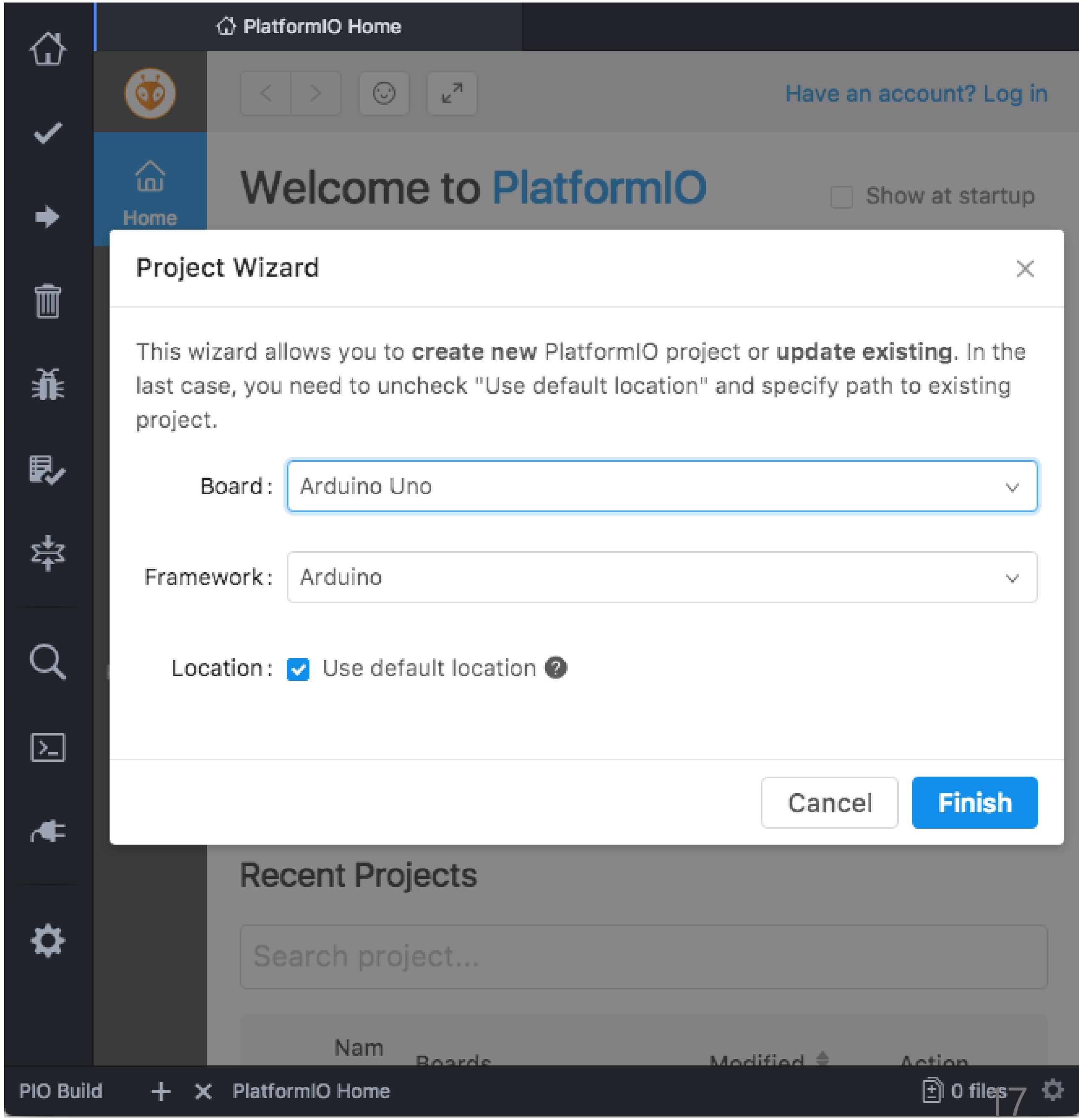
atom.io ×

Get to know Atom!

Platformio Quick Start

2. Klik **New Project**, pilih nama modulnya (Nodemcu v1.0), dan finish

3. Tunggu sampai selesai instalasi SDK-nya dan software pendukungnya. **SANGAT LAMA KETIKA PERTAMA KALI**. Silakan ngopi dulu ^_^



Project Wizard

This wizard allows you to **create new** PlatformIO project or **update existing**. In the last case, you need to uncheck "Use default location" and specify path to existing project.

Board : Arduino Uno

Framework : Arduino

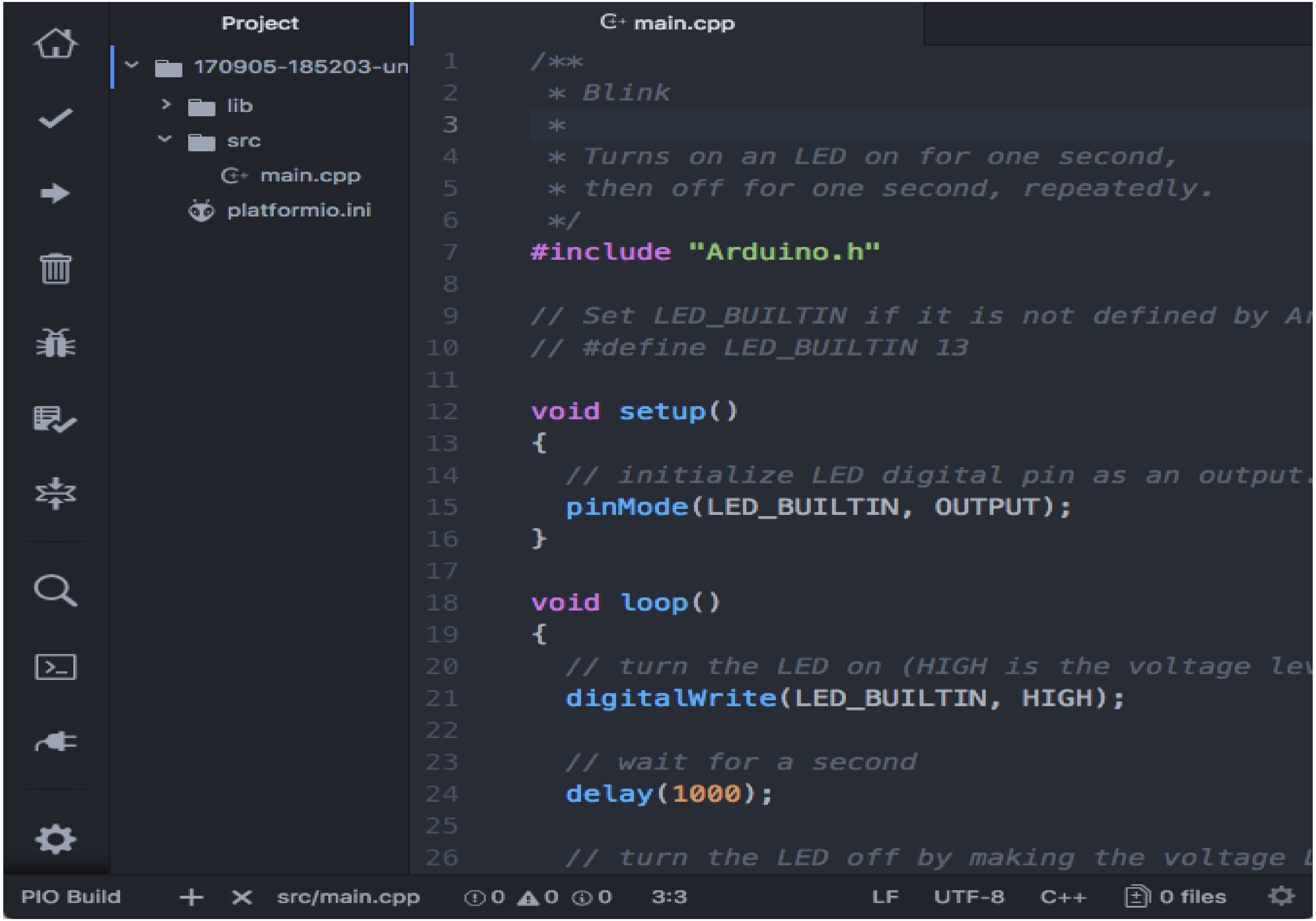
Location : ☑ Use default location

Platformio Quick Start

4. Buat file baru di folder **src** , misal : *main.cpp*

```
/**
 * Blink
 *
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */
#include <Arduino.h>

// Set LED_BUILTIN if it is not defined by Arduino framework
// #define LED_BUILTIN 13

void setup()
{
        Serial.begin(57600);

  // initialize LED digital pin as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}
```

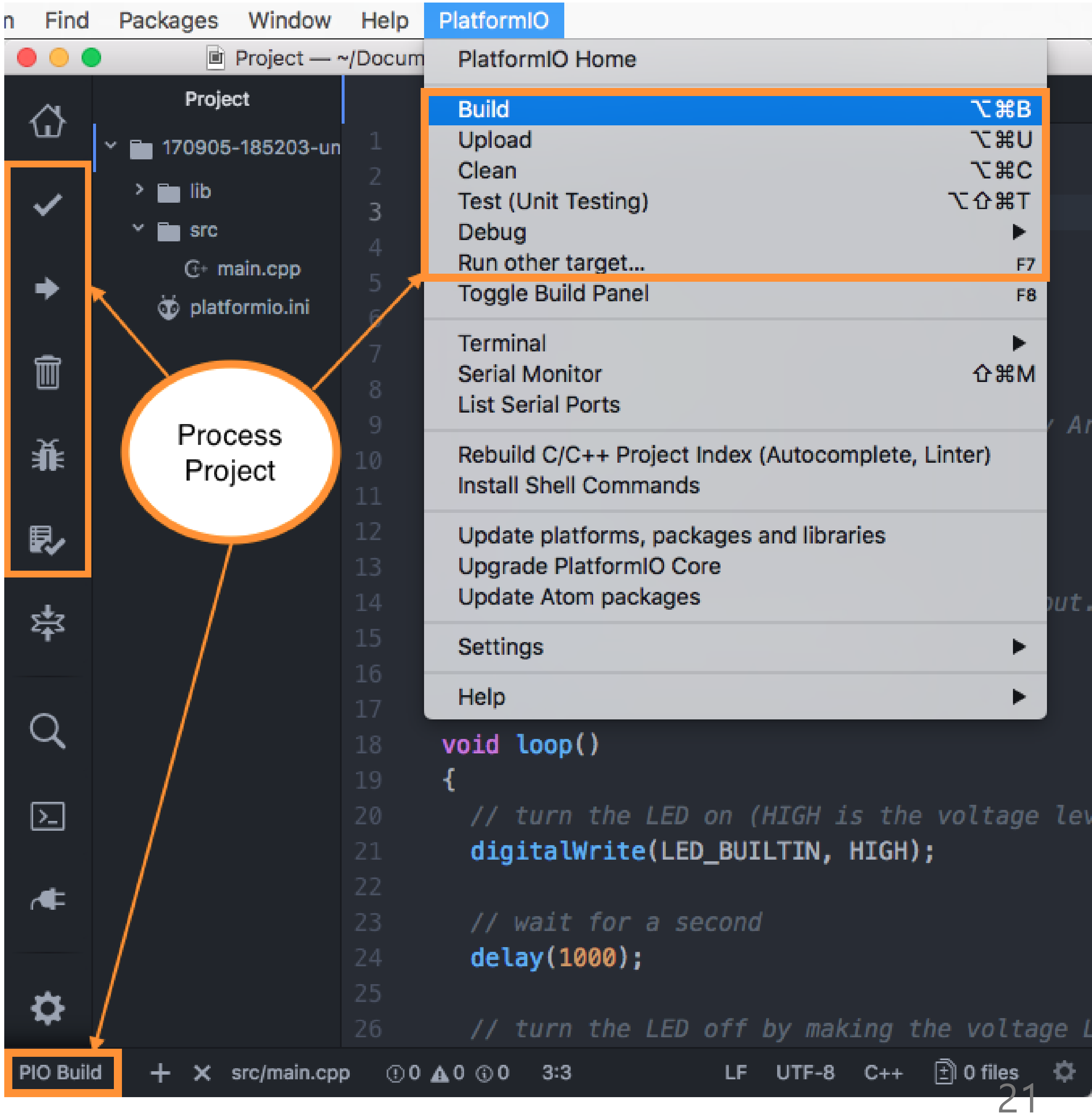
```
void loop()
{
  // turn the LED on (HIGH is the voltage level)
  digitalWrite(LED_BUILTIN, HIGH);

  // wait for a second
  delay(1000);

  // turn the LED off by making the voltage LOW
  digitalWrite(LED_BUILTIN, LOW);

   // wait for a second
  delay(1000);

  Serial.println(millis());
}
```



```
/**
 * Blink
 *
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */
#include "Arduino.h"

// Set LED_BUILTIN if it is not defined by Ar
// #define LED_BUILTIN 13

void setup()
{
  // initialize LED digital pin as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
{
  // turn the LED on (HIGH is the voltage lev
  digitalWrite(LED_BUILTIN, HIGH);

  // wait for a second
  delay(1000);

  // turn the LED off by making the voltage L
```

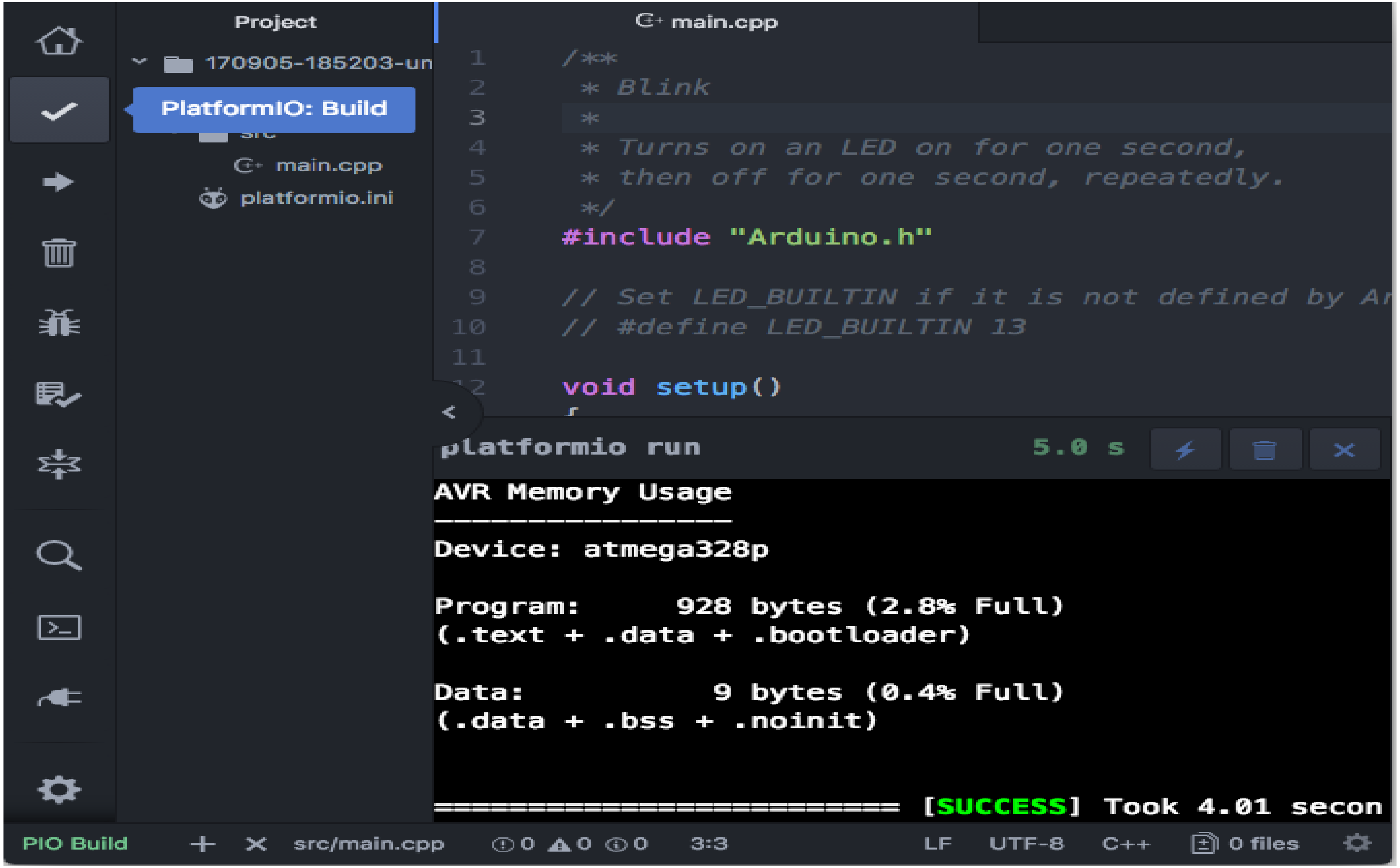
Platformio Quick Start

5. Kompile dan upload kode di atas via menu **Build**, **Upload**



```
void loop()
{
  // turn the LED on (HIGH is the voltage lev
  digitalWrite(LED_BUILTIN, HIGH);

  // wait for a second
  delay(1000);

  // turn the LED off by making the voltage L
```

Platformio Quick Start

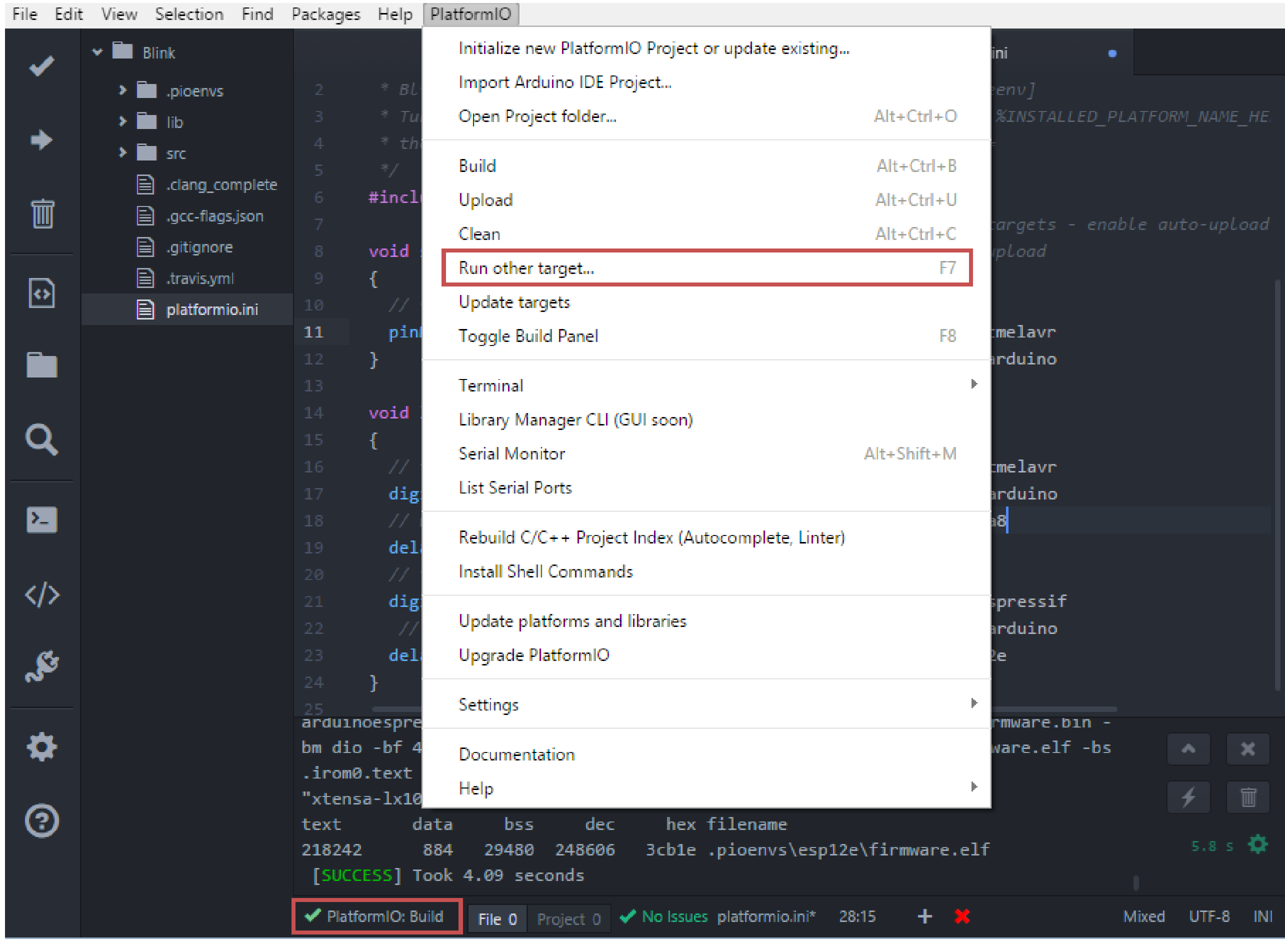


```
/**
 * Blink
 *
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */
#include "Arduino.h"

// Set LED_BUILTIN if it is not defined by Ar
// #define LED_BUILTIN 13

void setup()
{
```

```
platformio run                                  5.0 s

AVR Memory Usage
----------------
Device: atmega328p

Program:      928 bytes (2.8% Full)
(.text + .data + .bootloader)

Data:           9 bytes (0.4% Full)
(.data + .bss + .noinit)


========================== [SUCCESS] Took 4.01 secon
```

Platformio Quick Start

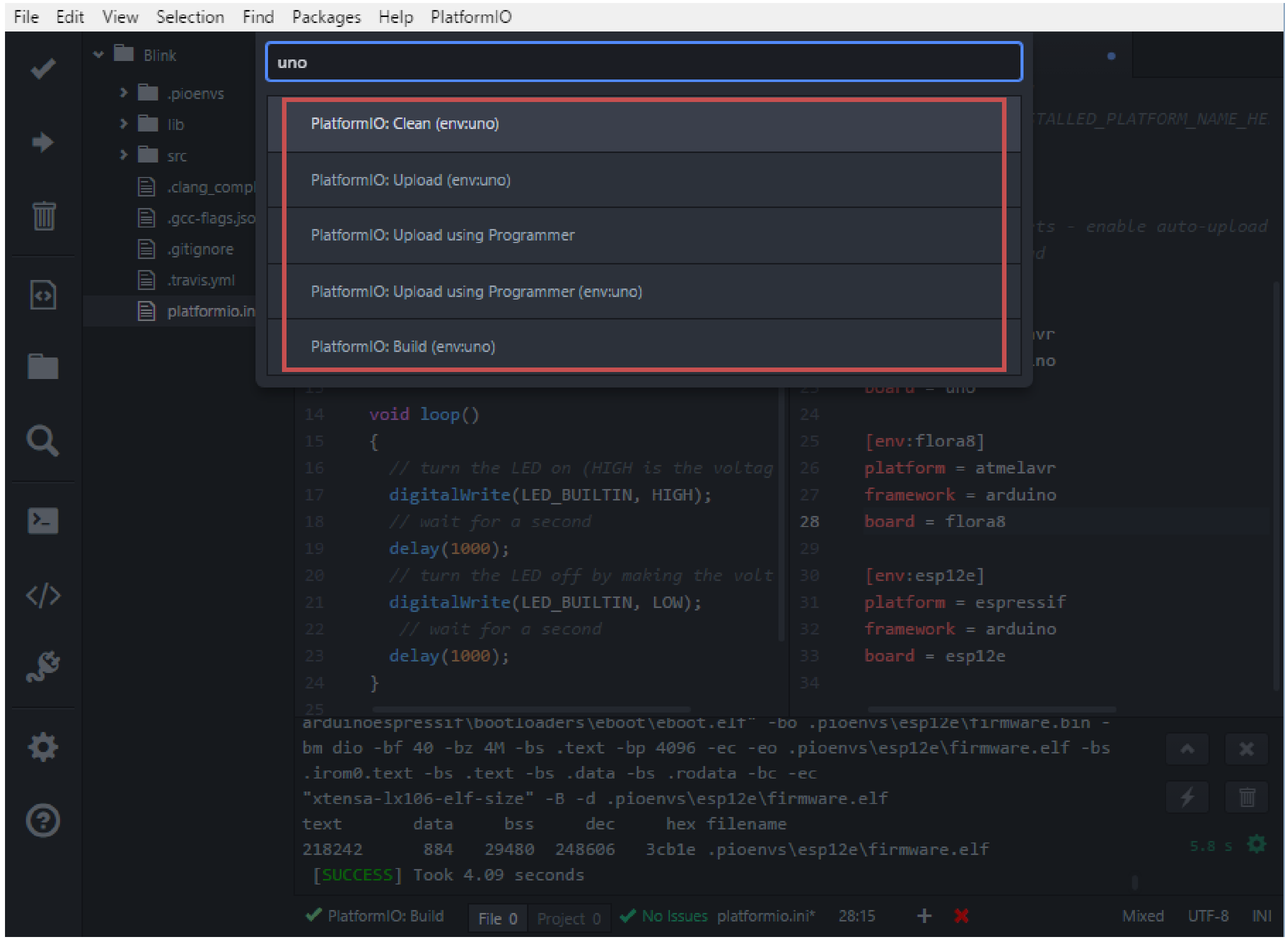
6. Untuk target dan opsi lainnya maka tekan tombol **F7**



```
text       data     bss    dec     hex filename
218242      884   29480 248606   3cb1e .pioenvs\esp12e\firmware.elf
 [SUCCESS] Took 4.09 seconds
```

Platformio Quick Start

7. Pilih dan jalankan sesuai keinginan Anda. **Ini di versi baru tidak muncul kalau di Atom** 😕



uno

- PlatformIO: Clean (env:uno)
- PlatformIO: Upload (env:uno)
- PlatformIO: Upload using Programmer
- PlatformIO: Upload using Programmer (env:uno)
- PlatformIO: Build (env:uno)

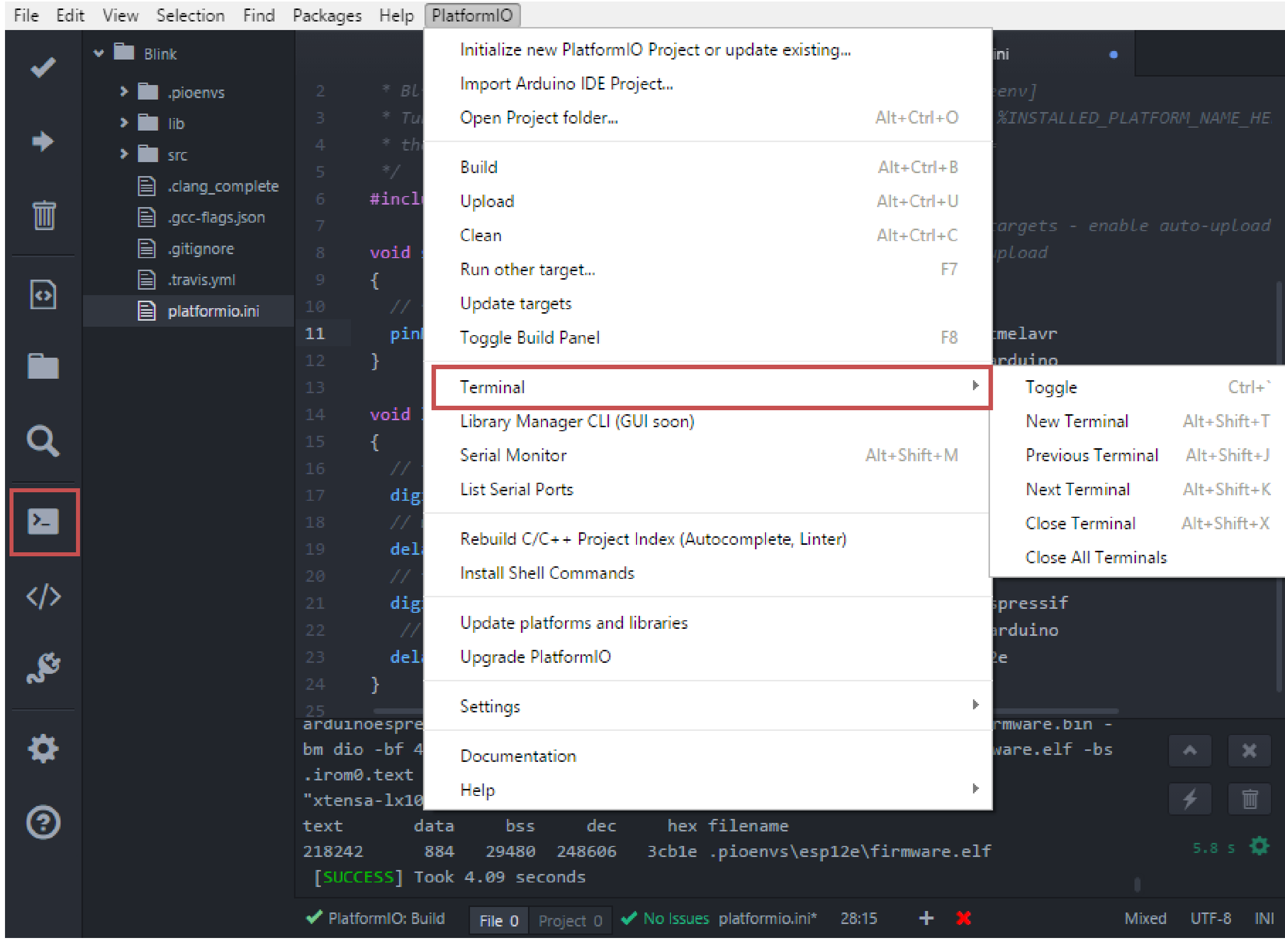
```
void loop()
{
  // turn the LED on (HIGH is the voltag
  digitalWrite(LED_BUILTIN, HIGH);
  // wait for a second
  delay(1000);
  // turn the LED off by making the volt
  digitalWrite(LED_BUILTIN, LOW);
   // wait for a second
  delay(1000);
}
```

```
[env:flora8]
platform = atmelavr
framework = arduino
board = flora8

[env:esp12e]
platform = espressif
framework = arduino
board = esp12e
```

```
arduinoespressif\bootloaders\eboot\eboot.elf" -bo .pioenvs\esp12e\firmware.bin -
bm dio -bf 40 -bz 4M -bs .text -bp 4096 -ec -eo .pioenvs\esp12e\firmware.elf -bs
.irom0.text -bs .text -bs .data -bs .rodata -bc -ec
"xtensa-lx106-elf-size" -B -d .pioenvs\esp12e\firmware.elf
text       data     bss     dec      hex filename
218242      884   29480  248606   3cb1e .pioenvs\esp12e\firmware.elf
 [SUCCESS] Took 4.09 seconds
```

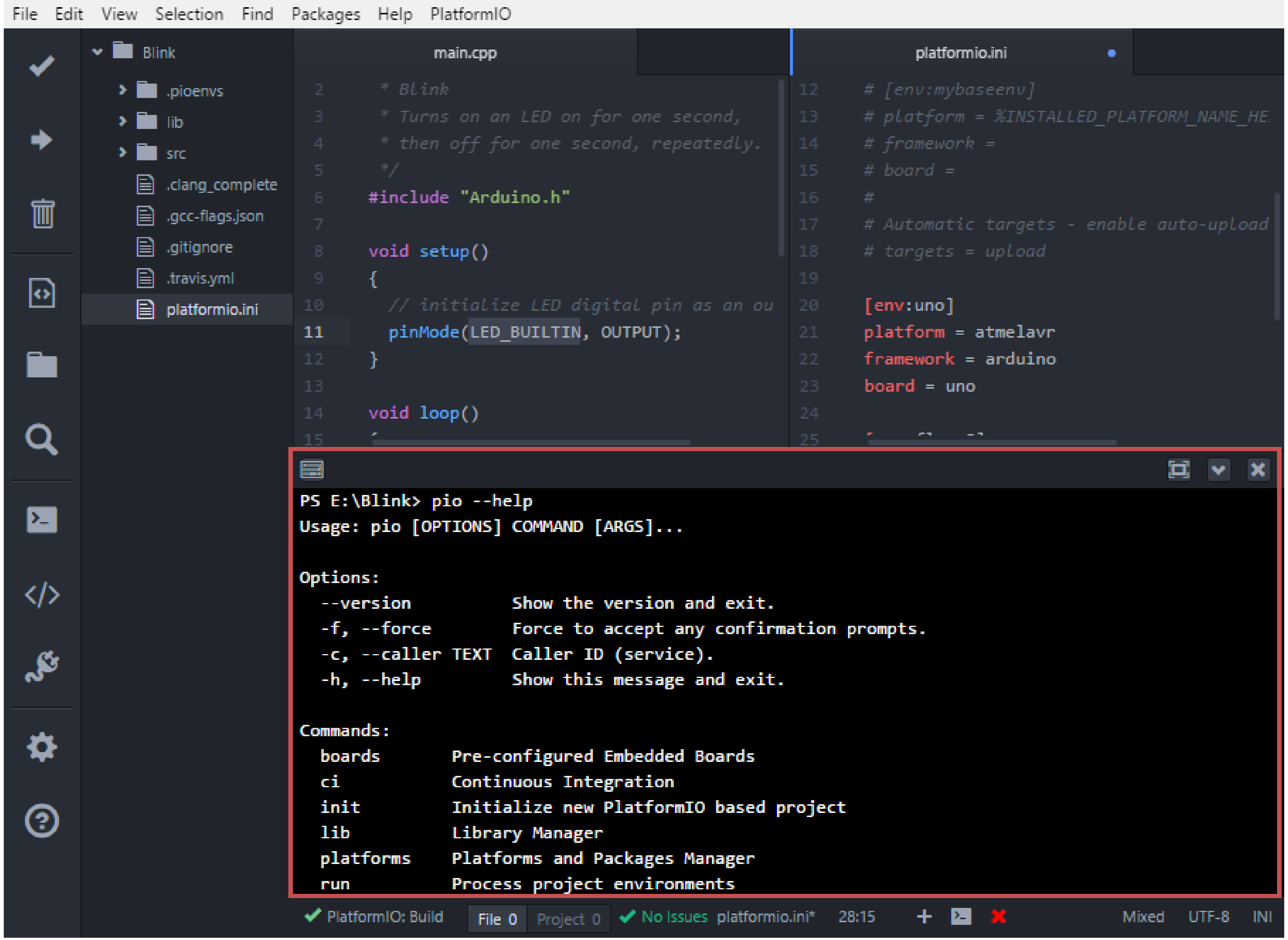
Platformio Quick Start

8. Untuk kebutuhan yang lebih maka pergunakan **Terminal**



```
arduinoespre                                          rmware.bin -
bm dio -bf 4                                          ware.elf -bs
.irom0.text
"xtensa-lx10
text       data     bss     dec     hex filename
218242      884   29480  248606   3cb1e .pioenvs\esp12e\firmware.elf
 [SUCCESS] Took 4.09 seconds
```

Platformio Quick Start



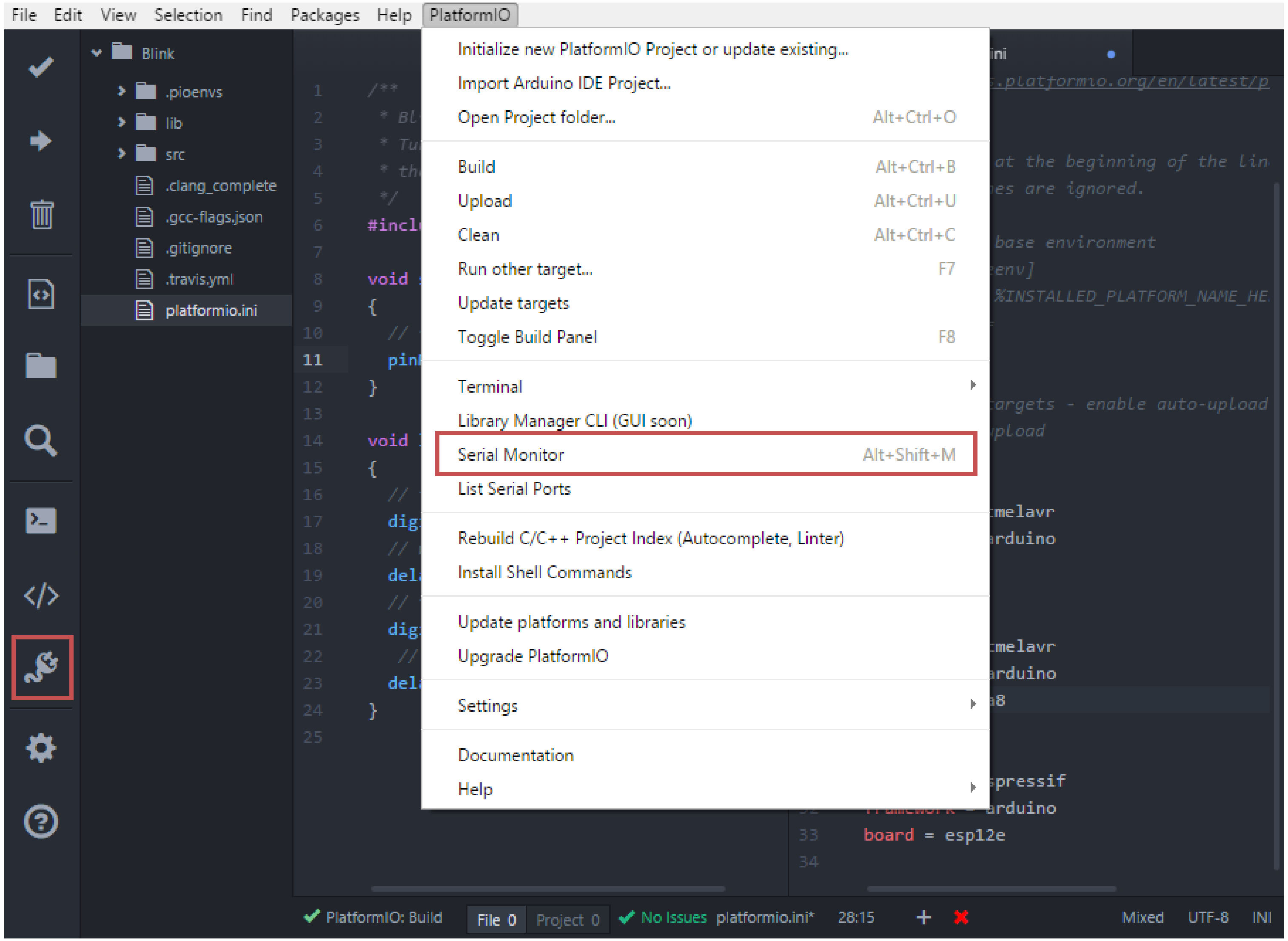
```
 * Blink
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */
#include "Arduino.h"

void setup()
{
  // initialize LED digital pin as an ou
  pinMode(LED_BUILTIN, OUTPUT);
}

void loop()
```

```
# [env:mybaseenv]
# platform = %INSTALLED_PLATFORM_NAME_HE
# framework =
# board =
#
# Automatic targets - enable auto-upload
# targets = upload

[env:uno]
platform = atmelavr
framework = arduino
board = uno
```

```
PS E:\Blink> pio --help
Usage: pio [OPTIONS] COMMAND [ARGS]...

Options:
  --version          Show the version and exit.
  -f, --force        Force to accept any confirmation prompts.
  -c, --caller TEXT  Caller ID (service).
  -h, --help         Show this message and exit.

Commands:
  boards       Pre-configured Embedded Boards
  ci           Continuous Integration
  init         Initialize new PlatformIO based project
  lib          Library Manager
  platforms    Platforms and Packages Manager
  run          Process project environments
```

Platformio Quick Start

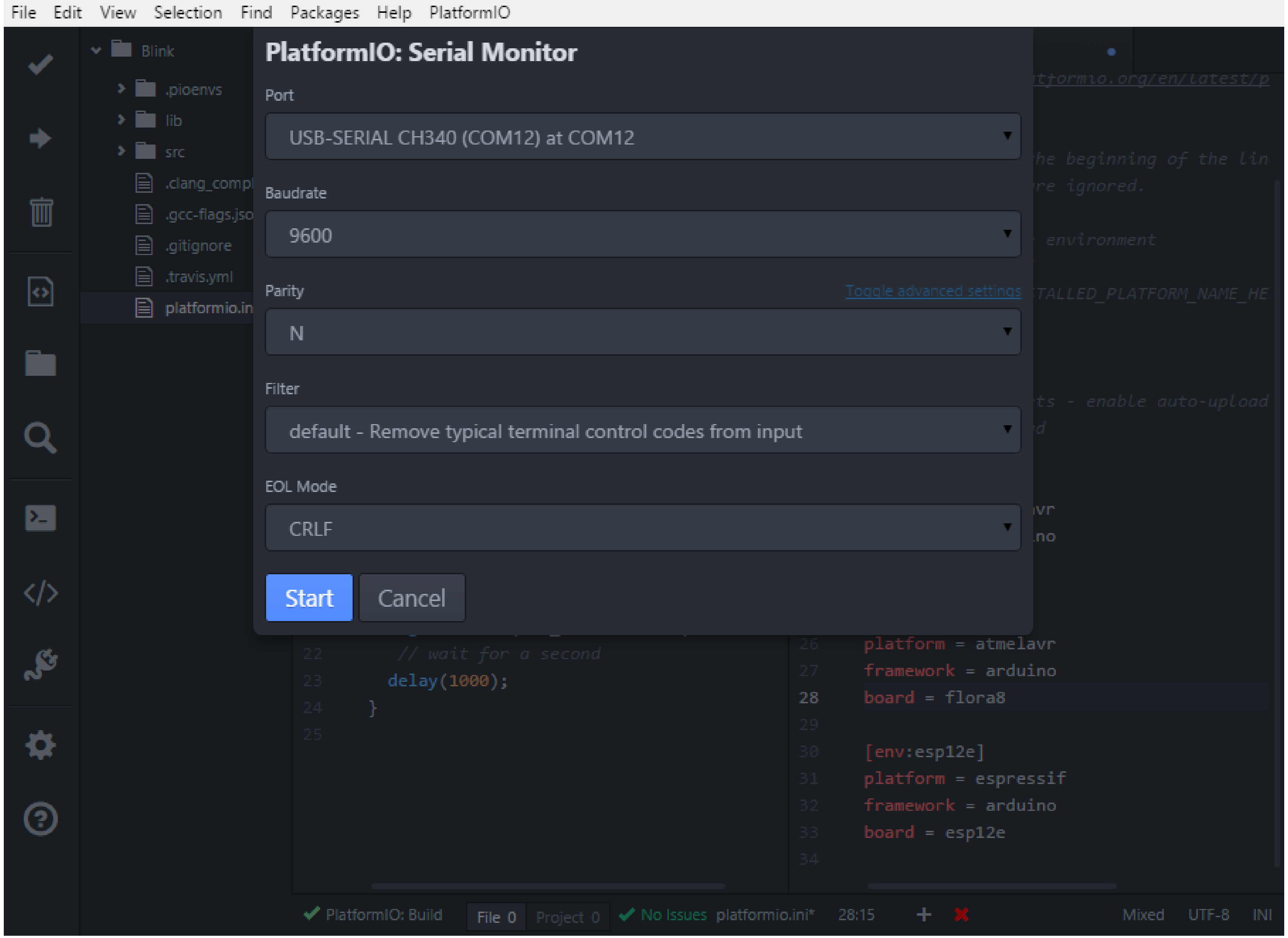
9. Debug program Anda menggunakan **Serial Port Monitor**



```
/**
 * Bl
 * Tu
 * th
 */
#incl

void
{
  //
  pin
}

void
{
  //
  dig
  //
  del
  //
  dig
   //
  del
}
```

```
.platformio.org/en/latest/p

at the beginning of the lin
es are ignored.

base environment
env]
%INSTALLED_PLATFORM_NAME_HE

argets - enable auto-upload
pload

melavr
rduino

melavr
rduino
a8

spressif
rduino
board = esp12e
```

Platformio Quick Start



PlatformIO: Serial Monitor

Port: USB-SERIAL CH340 (COM12) at COM12

Baudrate: 9600

Parity: N

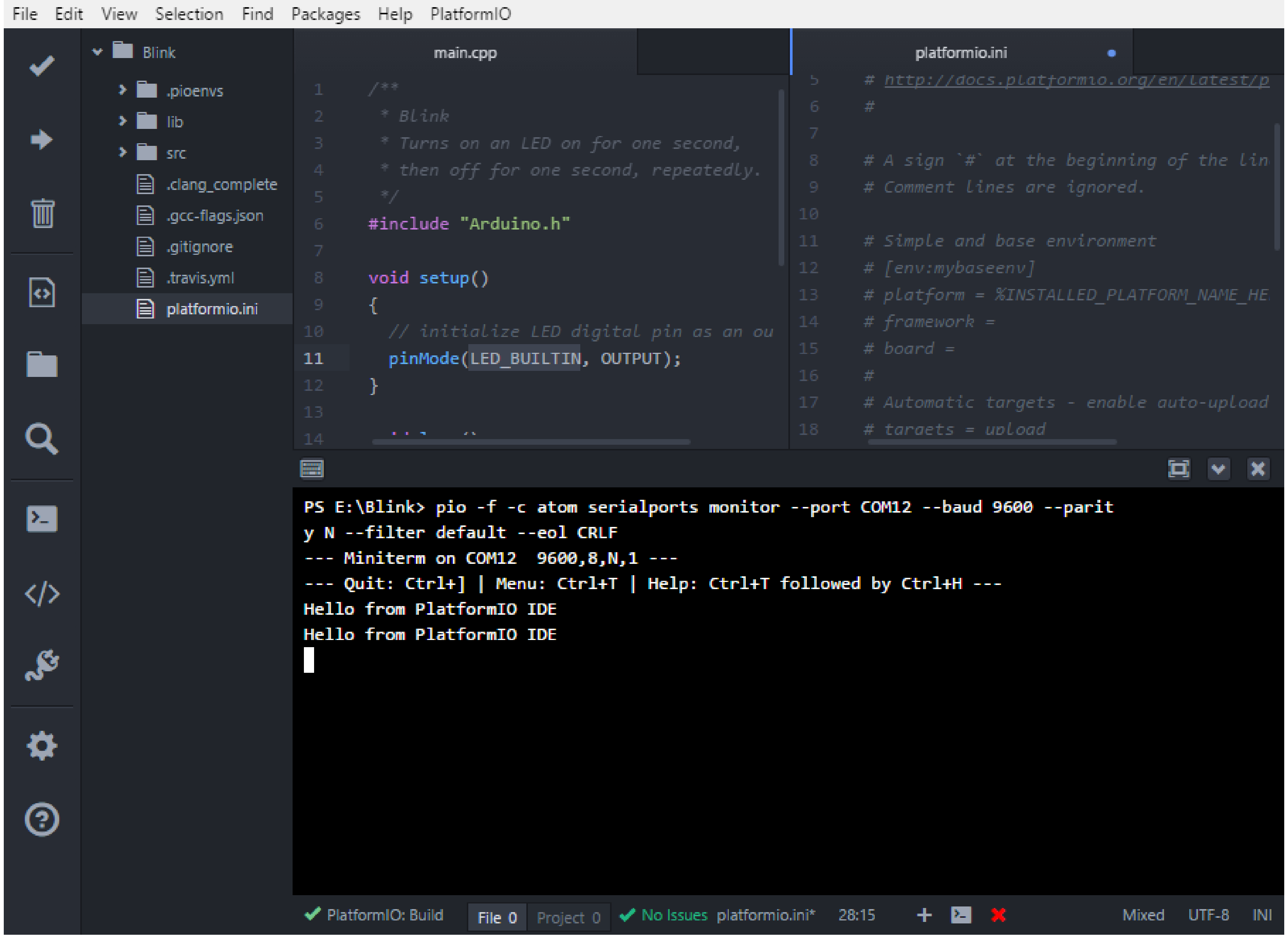
Filter: default - Remove typical terminal control codes from input

EOL Mode: CRLF

```
    // wait for a second
    delay(1000);
}
```

```
platform = atmelavr
framework = arduino
board = flora8

[env:esp12e]
platform = espressif
framework = arduino
board = esp12e
```

Platformio Quick Start



```
/**
 * Blink
 * Turns on an LED on for one second,
 * then off for one second, repeatedly.
 */
#include "Arduino.h"

void setup()
{
  // initialize LED digital pin as an ou
  pinMode(LED_BUILTIN, OUTPUT);
}

```

```
# http://docs.platformio.org/en/latest/p
#

# A sign `#` at the beginning of the lin
# Comment lines are ignored.

# Simple and base environment
# [env:mybaseenv]
# platform = %INSTALLED_PLATFORM_NAME_HE
# framework =
# board =
#
# Automatic targets - enable auto-upload
# targets = upload
```

```
PS E:\Blink> pio -f -c atom serialports monitor --port COM12 --baud 9600 --parit
y N --filter default --eol CRLF
--- Miniterm on COM12  9600,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
Hello from PlatformIO IDE
Hello from PlatformIO IDE
```

Platformio Core

1. Buka Atom Platformio
2. Buka **Terminal** menggunakan **CTRL+`**
3. Cek info *board* yang dipakai : `pio boards nodemcu` or `pio boards esp32doit`
4. Inisialiasi *project*-nya : `pio init --board uno --board nodemcuv1.0 --board esp32doit-devkit-v1`
5. Konfigurasi **platformio.ini** untuk kustomisasi opsi lebih lanjut

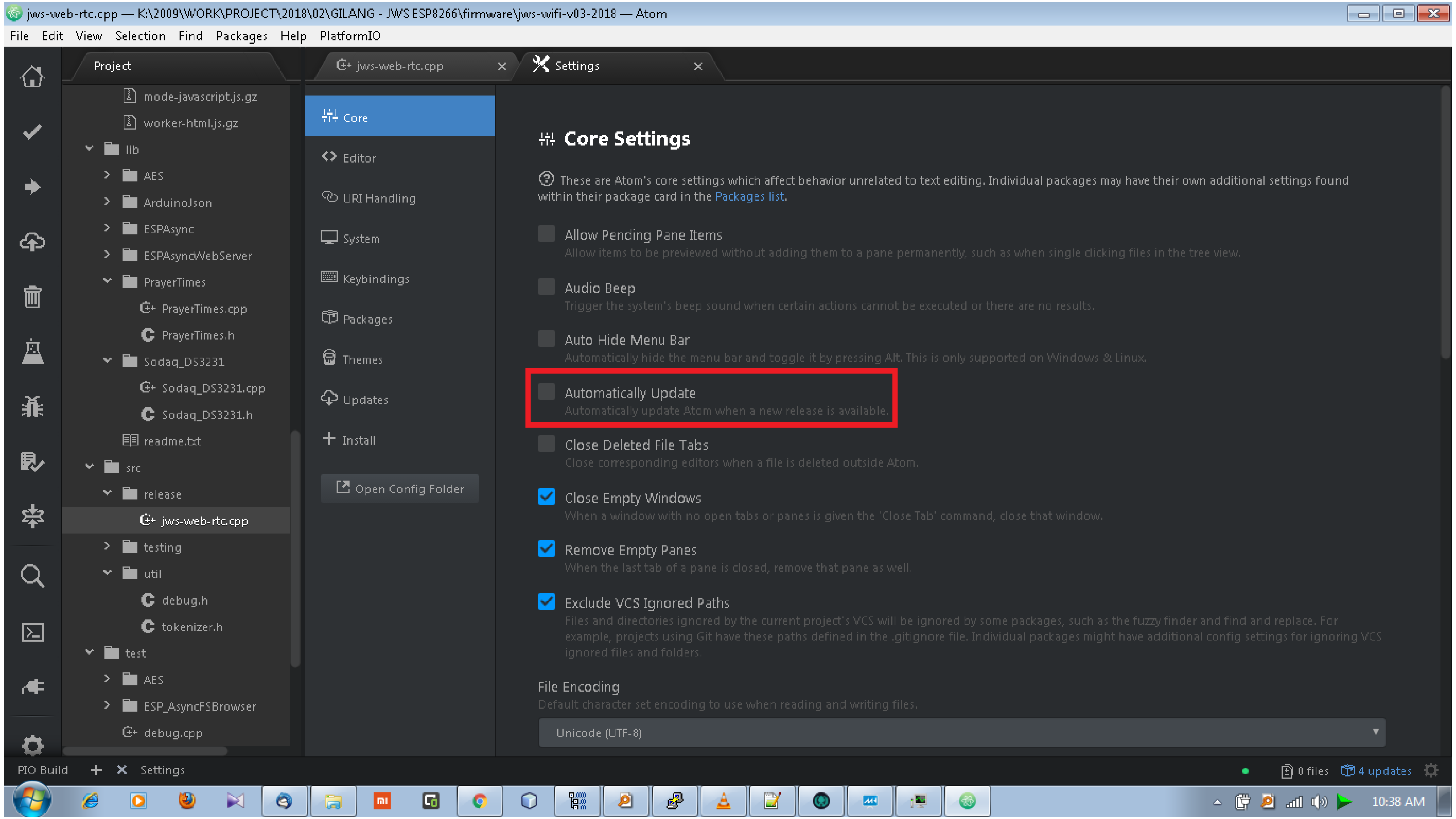
Platformio Core

platformio.ini

```
; PlatformIO Project Configuration File
;
;   Build options: build flags, source filter
;   Upload options: custom upload port, speed and extra flags
;   Library options: dependencies, extra library storages
;   Advanced options: extra scripting
;
; Please visit documentation for the other options and examples
; http://docs.platformio.org/page/projectconf.html
[platformio]
default_envs = iot_esp8266
core_dir = D:\PIO\.platformio

; if you want add aonther library folder
; lib_extra_dirs = K:\2009\WORK\PROJECT\PlatformIO_Library\
[env:iot_esp8266]
platform = espressif8266
board = nodemcuv2
framework = arduino
; 4M (1M SPIFFS)
build_flags = -Wl,-Tesp8266.flash.4m1m.ld

; 4M (3M SPIFFS)
; build_flags = -Wl,-Tesp8266.flash.4m.ld -lc

; make sure the library search it deep enough
lib_ldf_mode = deep+

; customize upload port
; upload_port = COM4
; upload_speed = 480600

; src_filter = -<test/>
; src_filter = +<*> -<arduino> -<test>
; you environment, name it as you wish, no space
[env:esp32dev]
; platform used
platform = espressif32
; platform = espressif32@0.11.1
; platform = https://github.com/platformio/
;             platform-espressif32.git

; framework use for this environment
; depend on your board and platform
framework = arduino

; Board name used
board = esp32doit-devkit-v1

; http://docs.platformio.org/en/latest/projectconf/
; section_env_library.html
; how the platformio search for Library
; deep+ is the most advance and
; complete for auto search Library
lib_ldf_mode = deep+
; which library you want to ignore or exclude from compilation
; lib_ignore = ESPAsync

; which library you want to use specifically for this environment
; lib_deps = DHT

; how the library is automagically choose,
; which one is going to be compiled,
; which one is not
; by default soft mode,
; platformio will ignore library which is
; not meant for the framework
; lib_compat_mode = soft

; if you got many folder inside src, then this will filter
; which one you want to include(+) or exclude(-)
; src_filter = +<*> -<release>

; add custom build flags
; -fexceptions needed if you want to use BLE
; build_flags = -fexceptions

; customize upload port, if needed
; upload_port = COM4
; upload_speed = 115200
```

Platformio Tips

- Disable auto update -- **Sangat menyebalkan**



Core Settings

These are Atom's core settings which affect behavior unrelated to text editing. Individual packages may have their own additional settings found within their package card in the Packages list.

□ Allow Pending Pane Items
Allow items to be previewed without adding them to a pane permanently, such as when single clicking files in the tree view.

□ Audio Beep
Trigger the system's beep sound when certain actions cannot be executed or there are no results.

□ Auto Hide Menu Bar
Automatically hide the menu bar and toggle it by pressing Alt. This is only supported on Windows & Linux.

□ Automatically Update
Automatically update Atom when a new release is available.

□ Close Deleted File Tabs
Close corresponding editors when a file is deleted outside Atom.

☑ Close Empty Windows
When a window with no open tabs or panes is given the 'Close Tab' command, close that window.

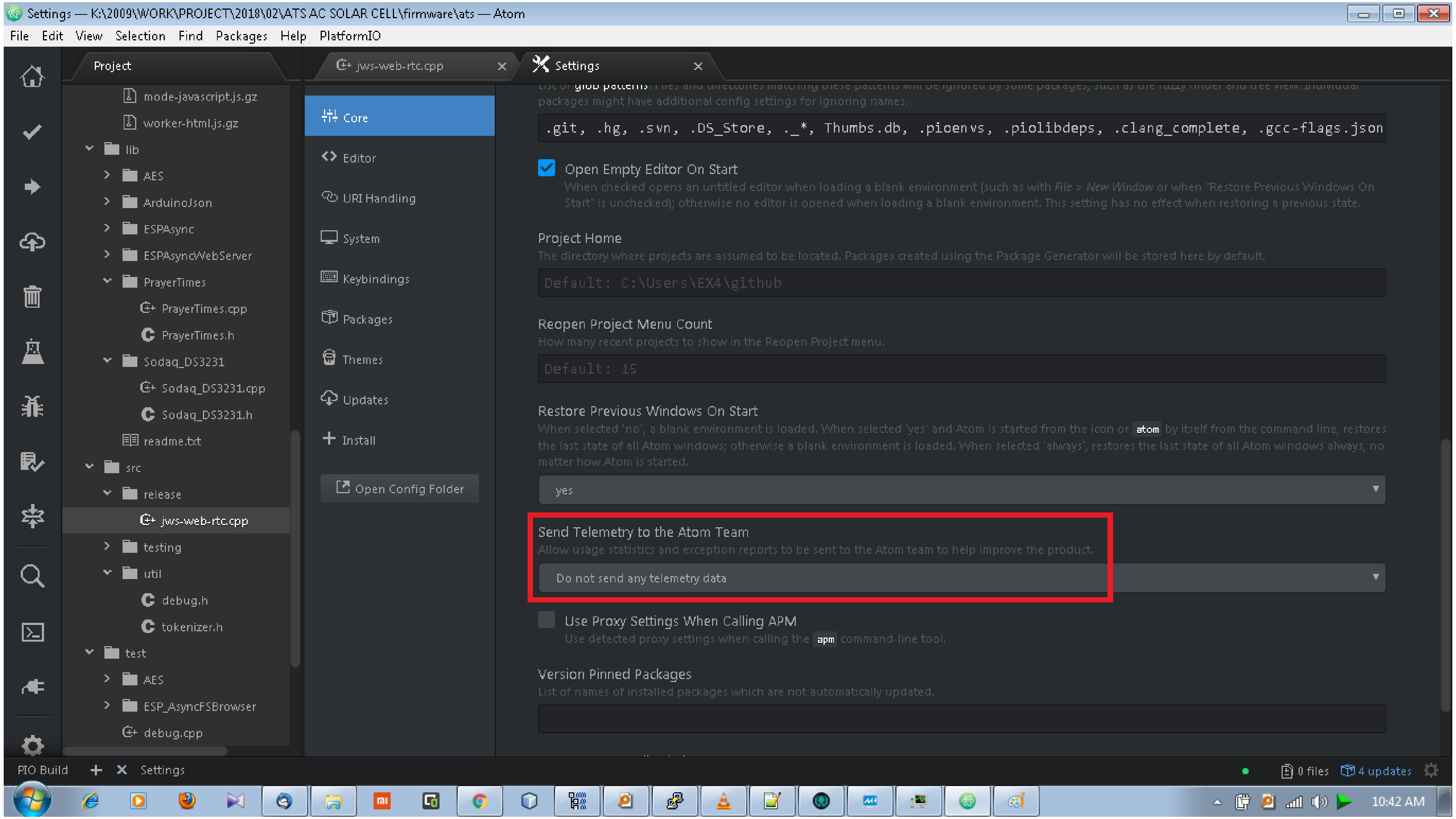
☑ Remove Empty Panes
When the last tab of a pane is closed, remove that pane as well.

☑ Exclude VCS Ignored Paths
Files and directories ignored by the current project's VCS will be ignored by some packages, such as the fuzzy finder and find and replace. For example, projects using Git have these paths defined in the .gitignore file. Individual packages might have additional config settings for ignoring VCS ignored files and folders.

File Encoding
Default character set encoding to use when reading and writing files.
Unicode (UTF-8)

Platformio Tips

- Disable Telemetry -- **Makan *bandwitdh* jaringan**



List of glob patterns. Files and directories matching these patterns will be ignored by some packages, such as the fuzzy finder and tree view. Individual packages might have additional config settings for ignoring names.

`.git, .hg, .svn, .DS_Store, ._*, Thumbs.db, .pioenvs, .piolibdeps, .clang_complete, .gcc-flags.json`

☑ Open Empty Editor On Start
When checked opens an untitled editor when loading a blank environment (such as with *File > New Window* or when "Restore Previous Windows On Start" is unchecked); otherwise no editor is opened when loading a blank environment. This setting has no effect when restoring a previous state.

Project Home
The directory where projects are assumed to be located. Packages created using the Package Generator will be stored here by default.
`Default: C:\Users\EX4\github`

Reopen Project Menu Count
How many recent projects to show in the Reopen Project menu.
`Default: 15`

Restore Previous Windows On Start
When selected 'no', a blank environment is loaded. When selected 'yes' and Atom is started from the icon or `atom` by itself from the command line, restores the last state of all Atom windows; otherwise a blank environment is loaded. When selected 'always', restores the last state of all Atom windows always, no matter how Atom is started.
yes ▾

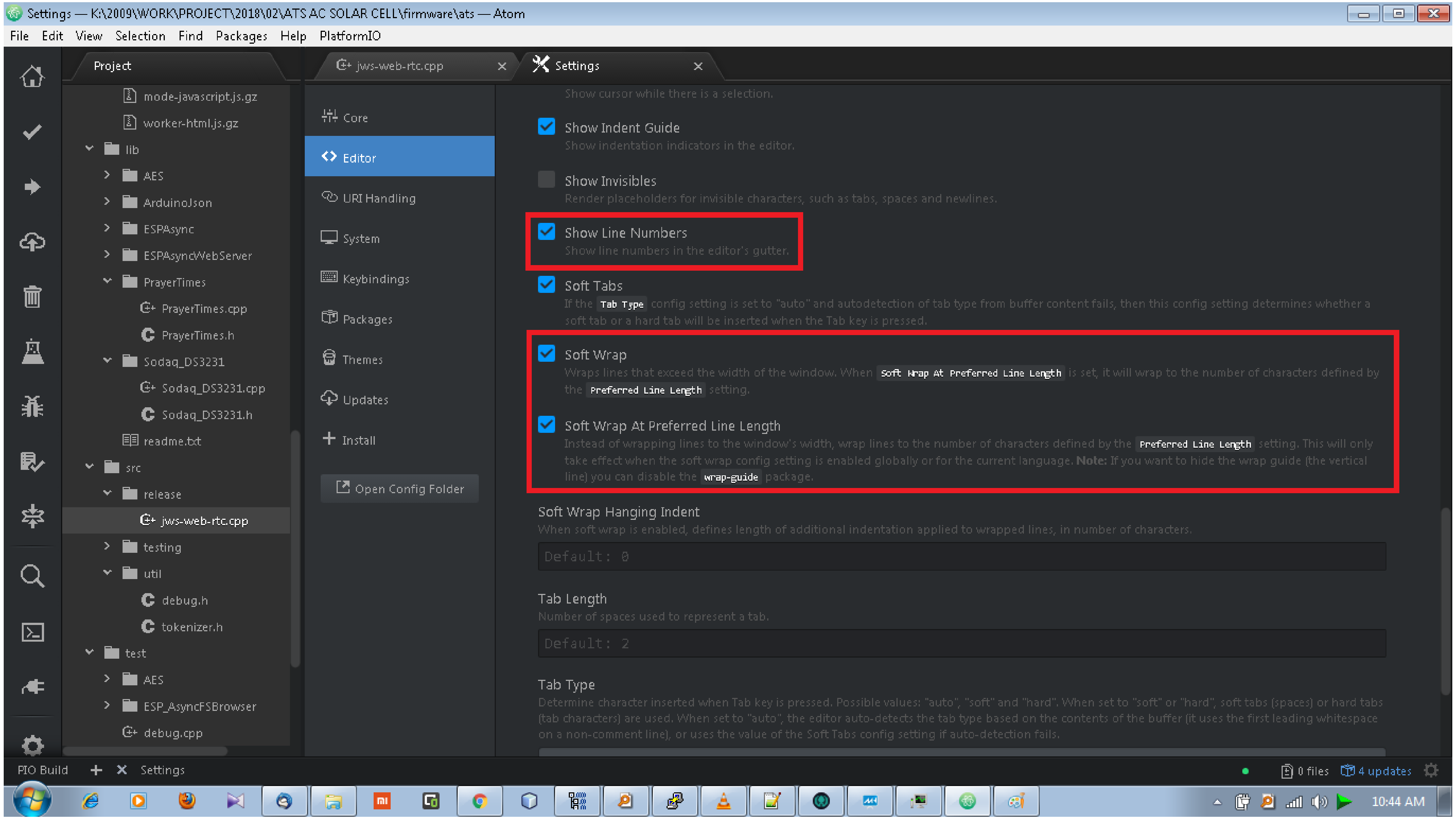
Send Telemetry to the Atom Team
Allow usage statistics and exception reports to be sent to the Atom team to help improve the product.
Do not send any telemetry data ▾

□ Use Proxy Settings When Calling APM
Use detected proxy settings when calling the `apm` command-line tool.

Version Pinned Packages
List of names of installed packages which are not automatically updated.

Platformio Tips

- Aktifkan nomor baris dan opsi wrap text -- **Informatif**



Show cursor while there is a selection.

☑ Show Indent Guide
Show indentation indicators in the editor.

□ Show Invisibles
Render placeholders for invisible characters, such as tabs, spaces and newlines.

☑ Show Line Numbers
Show line numbers in the editor's gutter.

☑ Soft Tabs
If the `Tab Type` config setting is set to "auto" and autodetection of tab type from buffer content fails, then this config setting determines whether a soft tab or a hard tab will be inserted when the Tab key is pressed.

☑ Soft Wrap
Wraps lines that exceed the width of the window. When `Soft Wrap At Preferred Line Length` is set, it will wrap to the number of characters defined by the `Preferred Line Length` setting.

☑ Soft Wrap At Preferred Line Length
Instead of wrapping lines to the window's width, wrap lines to the number of characters defined by the `Preferred Line Length` setting. This will only take effect when the soft wrap config setting is enabled globally or for the current language. **Note:** If you want to hide the wrap guide (the vertical line) you can disable the `wrap-guide` package.

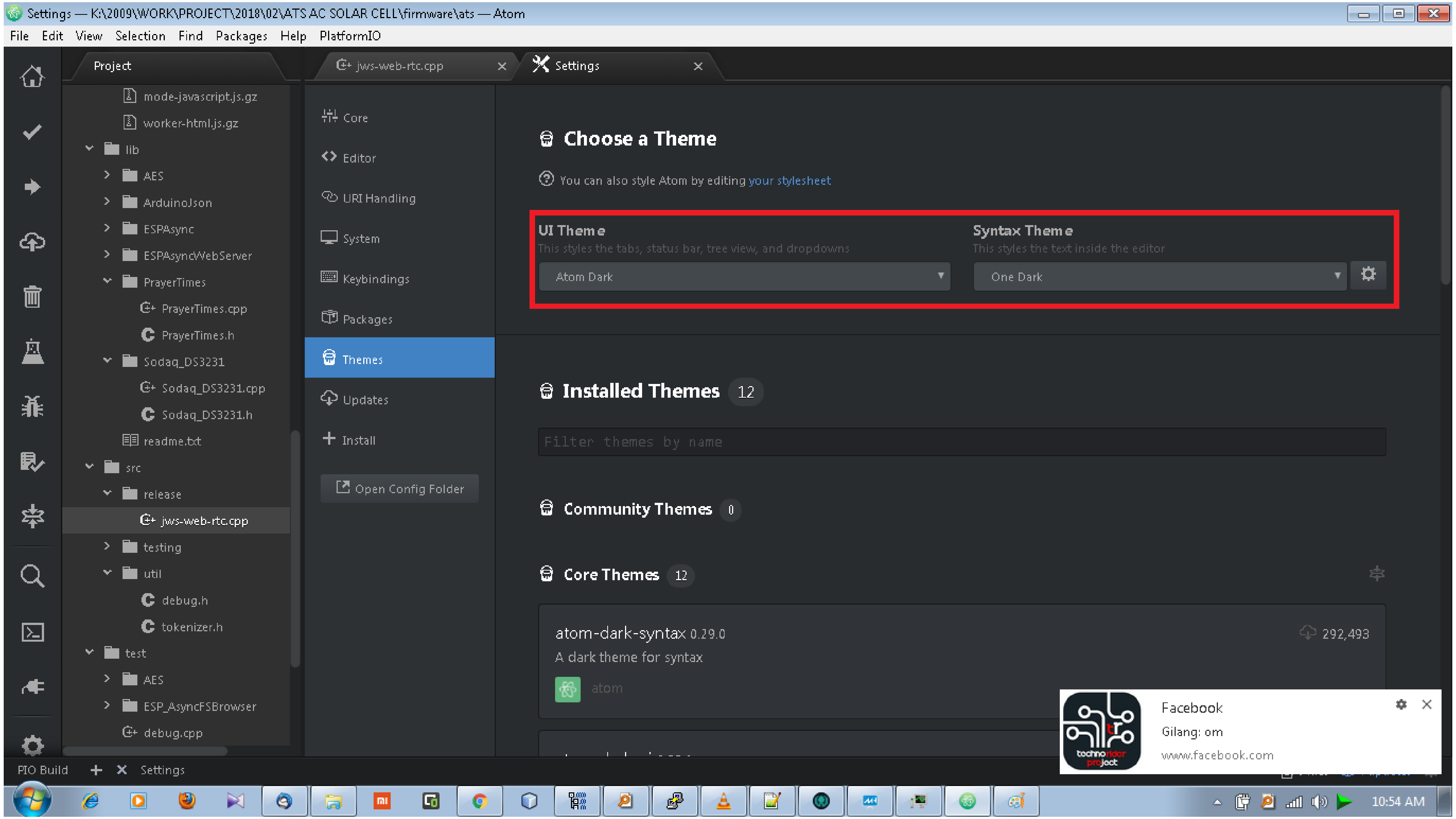
Soft Wrap Hanging Indent
When soft wrap is enabled, defines length of additional indentation applied to wrapped lines, in number of characters.
Default: 0

Tab Length
Number of spaces used to represent a tab.
Default: 2

Tab Type
Determine character inserted when Tab key is pressed. Possible values: "auto", "soft" and "hard". When set to "soft" or "hard", soft tabs (spaces) or hard tabs (tab characters) are used. When set to "auto", the editor auto-detects the tab type based on the contents of the buffer (it uses the first leading whitespace on a non-comment line), or uses the value of the Soft Tabs config setting if auto-detection fails.

Platformio Tips

- Pilih theme yang adem di mata -- **Bikin ga lelah mata**



Settings — K:\2009\WORK\PROJECT\2018\02\ATS AC SOLAR CELL\firmware\ats — Atom

Choose a Theme

You can also style Atom by editing your stylesheet

UI Theme
This styles the tabs, status bar, tree view, and dropdowns
Atom Dark

Syntax Theme
This styles the text inside the editor
One Dark

Installed Themes 12

Filter themes by name

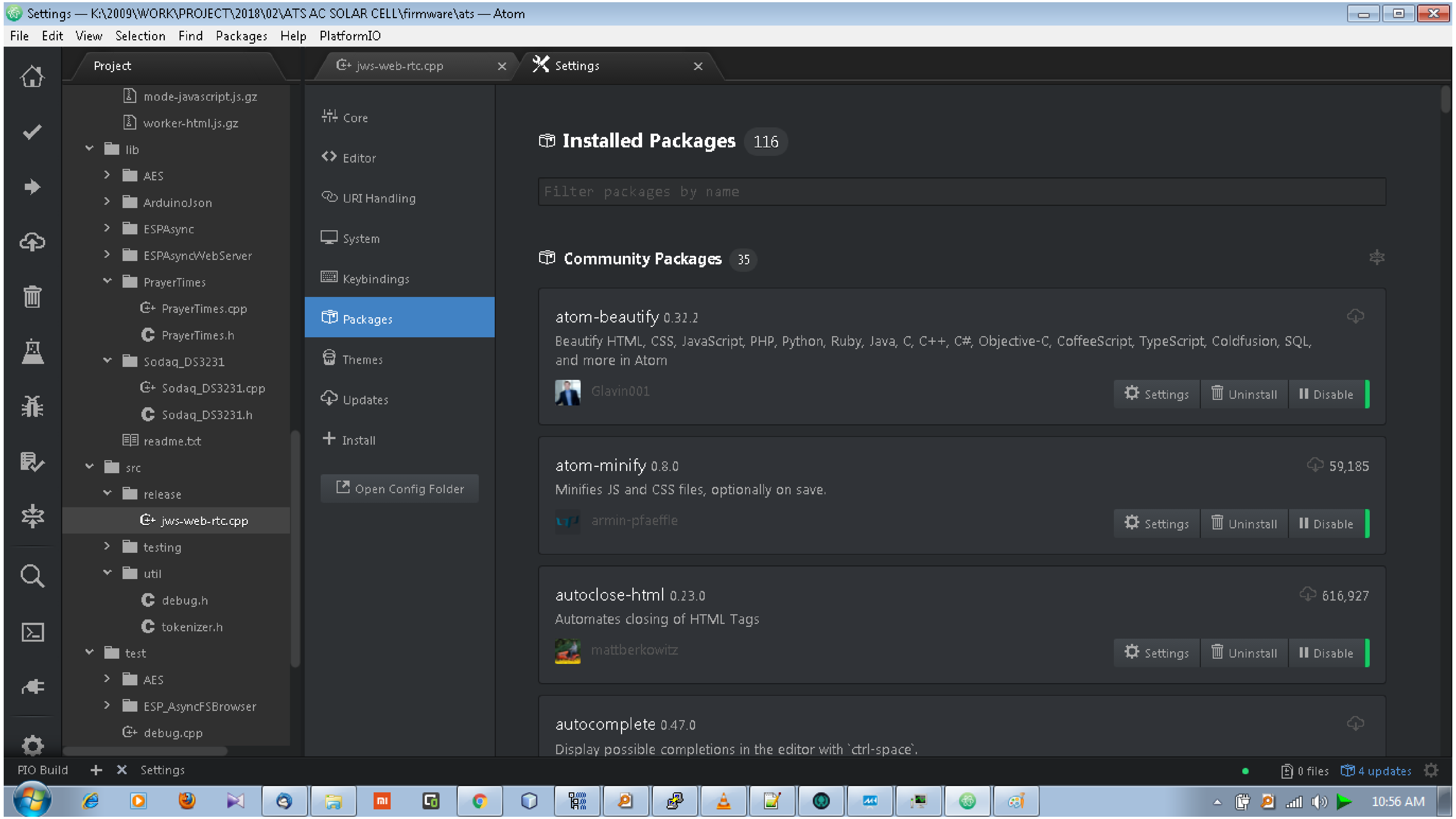
Community Themes 0

Core Themes 12

atom-dark-syntax 0.29.0
A dark theme for syntax
atom
292,493

Platformio Tips

- Tambahkan beberapa modul Atom -- **Lebih produktif**
  - **atom-beautify, dockblockr, minimap**



Installed Packages 116

Filter packages by name

Community Packages 35

atom-beautify 0.32.2
Beautify HTML, CSS, JavaScript, PHP, Python, Ruby, Java, C, C++, C#, Objective-C, CoffeeScript, TypeScript, Coldfusion, SQL, and more in Atom
Glavin001
Settings | Uninstall | Disable

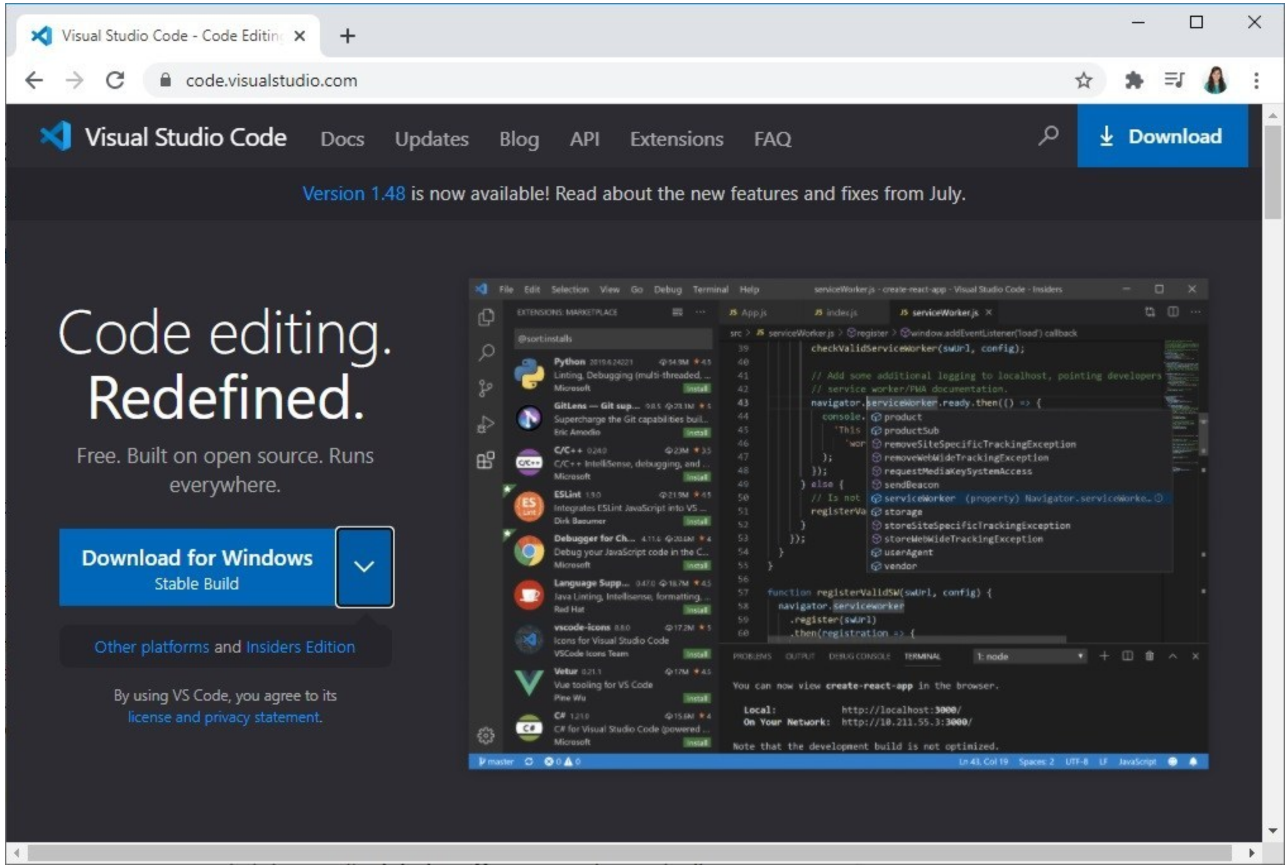
atom-minify 0.8.0 — 59,185
Minifies JS and CSS files, optionally on save.
armin-pfaeffle
Settings | Uninstall | Disable

autoclose-html 0.23.0 — 616,927
Automates closing of HTML Tags
mattberkowitz
Settings | Uninstall | Disable

autocomplete 0.47.0
Display possible completions in the editor with `ctrl-space`.

Visual Studio Code

- Download versi portable dan ekstrak ke folder tanpa spasi : https://code.visualstudio.com/download https://code.visualstudio.com/sha/download?build=stable&os=win32-archive
- Pastikan Python 3.5 ke atas sudah terinstall



Version 1.48 is now available! Read about the new features and fixes from July.

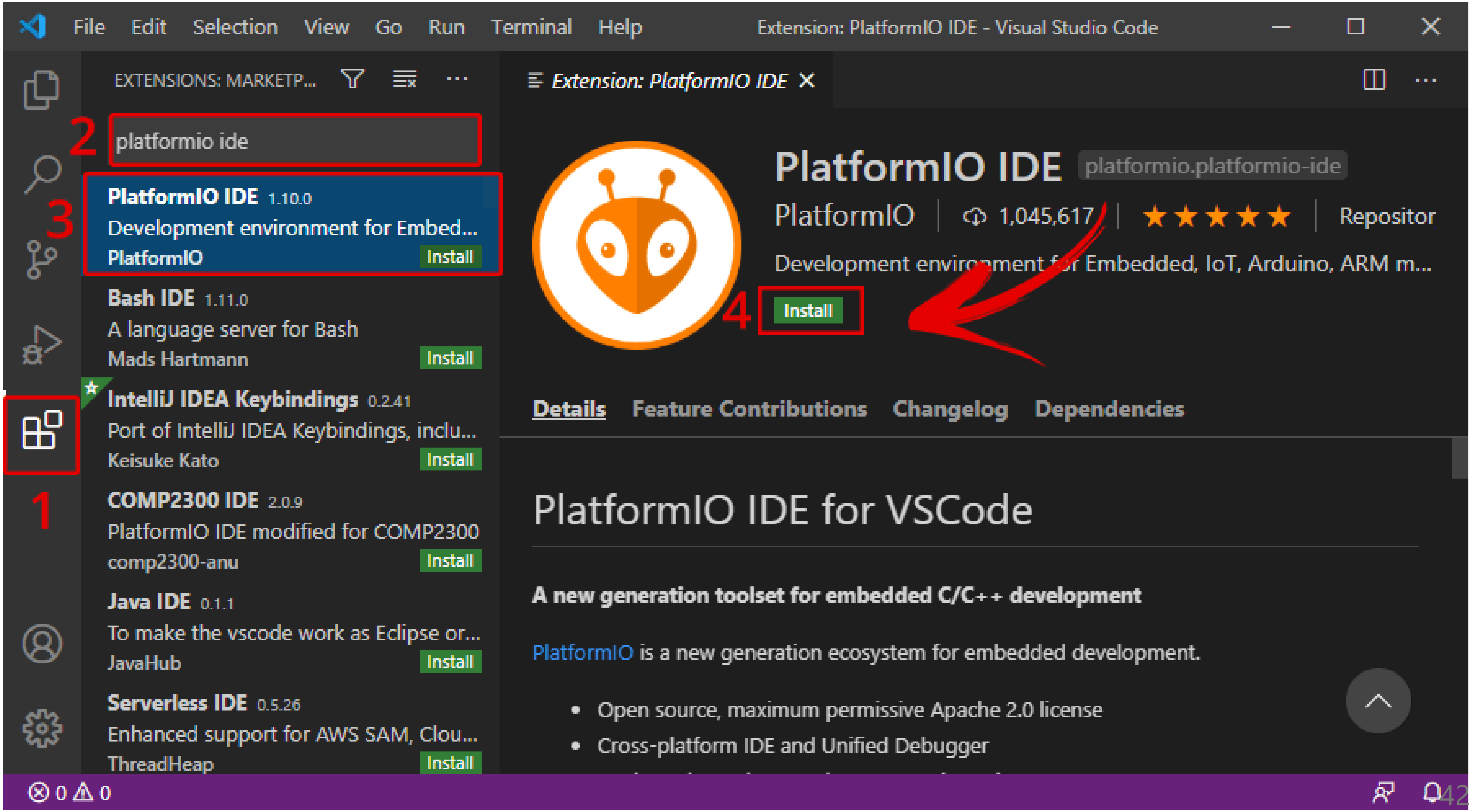
Code editing. Redefined.

Free. Built on open source. Runs everywhere.

By using VS Code, you agree to its license and privacy statement.

PlatformIO Extension

- Run VS Code As Adminitrator
- Klik Extensions icon atau tekan **Ctrl+Shift+X**
- Cari **PlatformIO IDE**
- Klik install dan silakan tunggu
- Setelah selesai, pastikan PlatformIO Extension sudah aktif



platformio ide

PlatformIO IDE 1.10.0
Development environment for Embed...
PlatformIO

Bash IDE 1.11.0
A language server for Bash
Mads Hartmann

IntelliJ IDEA Keybindings 0.2.41
Port of IntelliJ IDEA Keybindings, inclu...
Keisuke Kato

COMP2300 IDE 2.0.9
PlatformIO IDE modified for COMP2300
comp2300-anu

Java IDE 0.1.1
To make the vscode work as Eclipse or...
JavaHub

Serverless IDE 0.5.26
Enhanced support for AWS SAM, Clou...
ThreadHeap

PlatformIO IDE platformio.platformio-ide

PlatformIO | 1,045,617 | ★★★★★ | Repositor

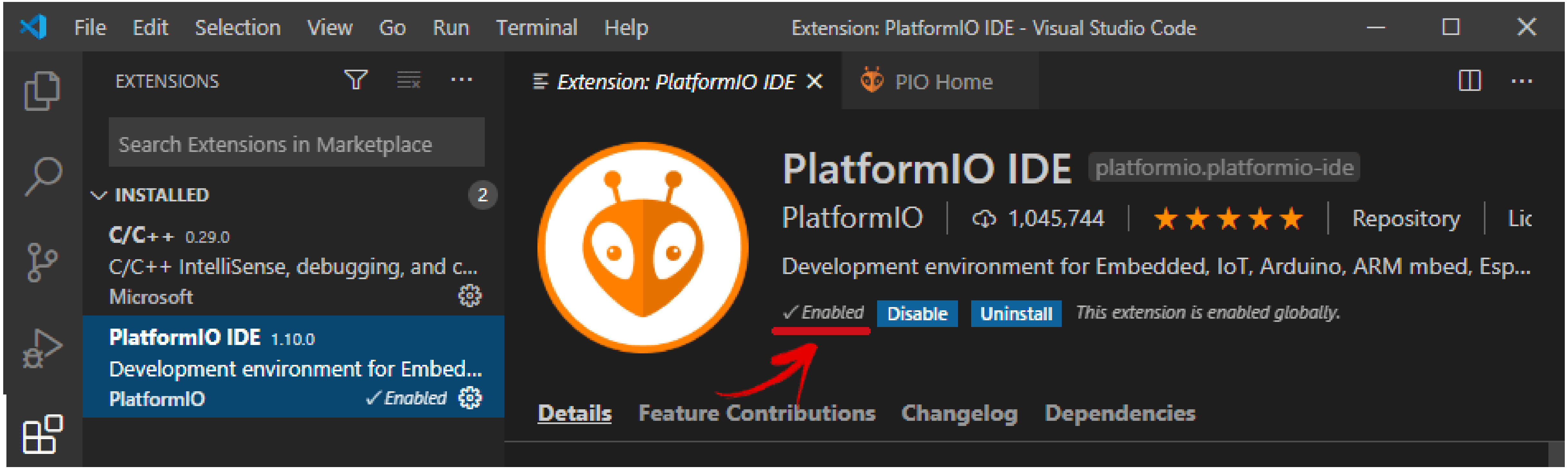
Development environment for Embedded, IoT, Arduino, ARM m...

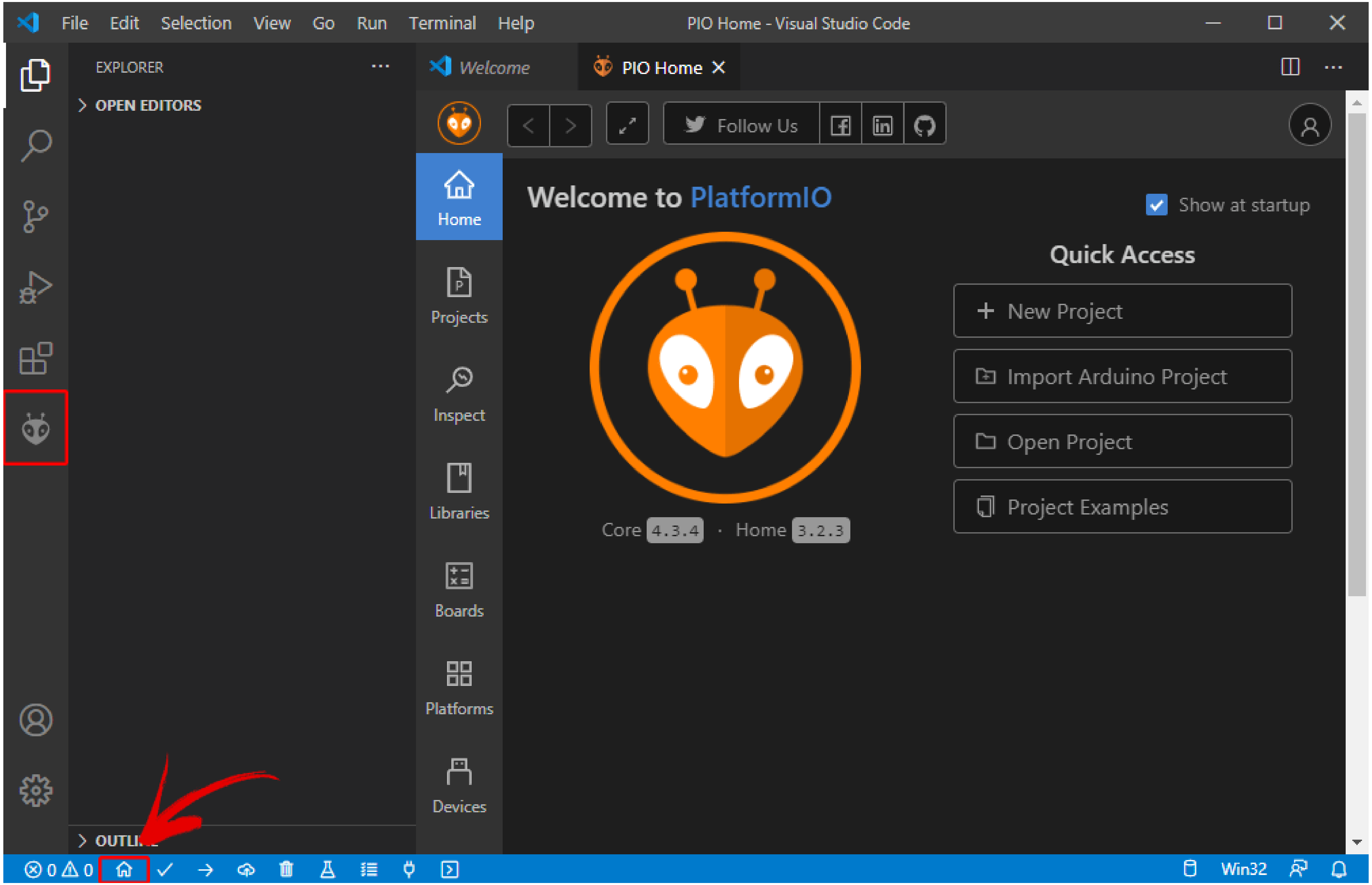
PlatformIO IDE for VSCode

A new generation toolset for embedded C/C++ development

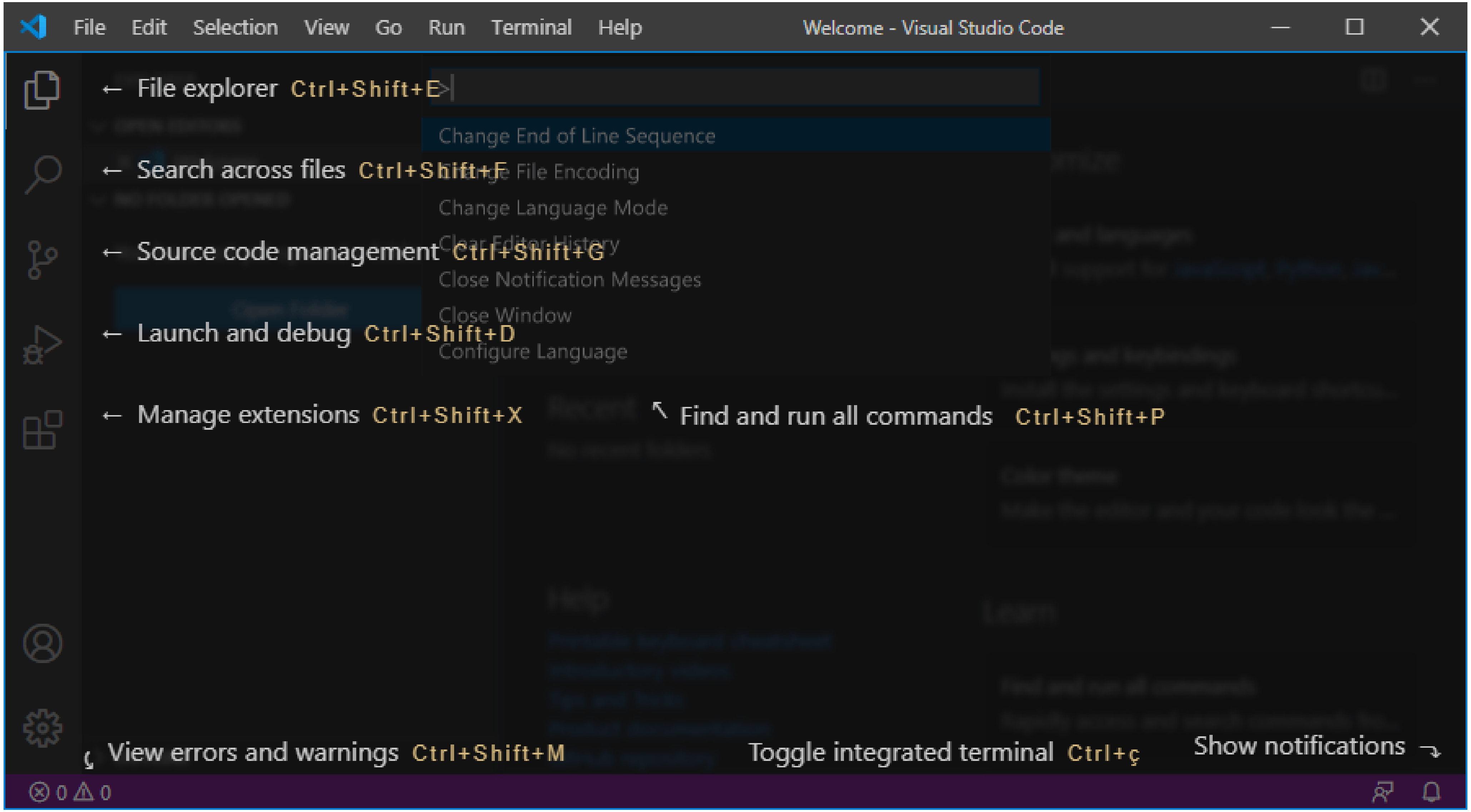
PlatformIO is a new generation ecosystem for embedded development.

- Open source, maximum permissive Apache 2.0 license
- Cross-platform IDE and Unified Debugger


File
Edit
Selection
View
Go
Run
Terminal
Help
Extension: PlatformIO IDE - Visual Studio Code
EXTENSIONS
Search Extensions in Marketplace
INSTALLED
2
C/C++ 0.29.0
C/C++ IntelliSense, debugging, and c...
Microsoft
PlatformIO IDE 1.10.0
Development environment for Embed...
PlatformIO
✓ Enabled
Extension: PlatformIO IDE
PIO Home
PlatformIO IDE
platformio.platformio-ide
PlatformIO
1,045,744
★★★★★
Repository
Lic
Development environment for Embedded, IoT, Arduino, ARM mbed, Esp...
✓ Enabled
Disable
Uninstall
This extension is enabled globally.
Details
Feature Contributions
Changelog
Dependencies


File
Edit
Selection
View
Go
Run
Terminal
Help
PIO Home - Visual Studio Code
EXPLORER
OPEN EDITORS
OUTLINE
Welcome
PIO Home
Follow Us
Home
Projects
Inspect
Libraries
Boards
Platforms
Devices
Welcome to PlatformIO
Show at startup
Core 4.3.4 · Home 3.2.3
Quick Access
New Project
Import Arduino Project
Open Project
Project Examples
0
0
Win32

Interface



← File explorer Ctrl+Shift+E

← Search across files Ctrl+Shift+F

← Source code management Ctrl+Shift+G

← Launch and debug Ctrl+Shift+D

← Manage extensions Ctrl+Shift+X

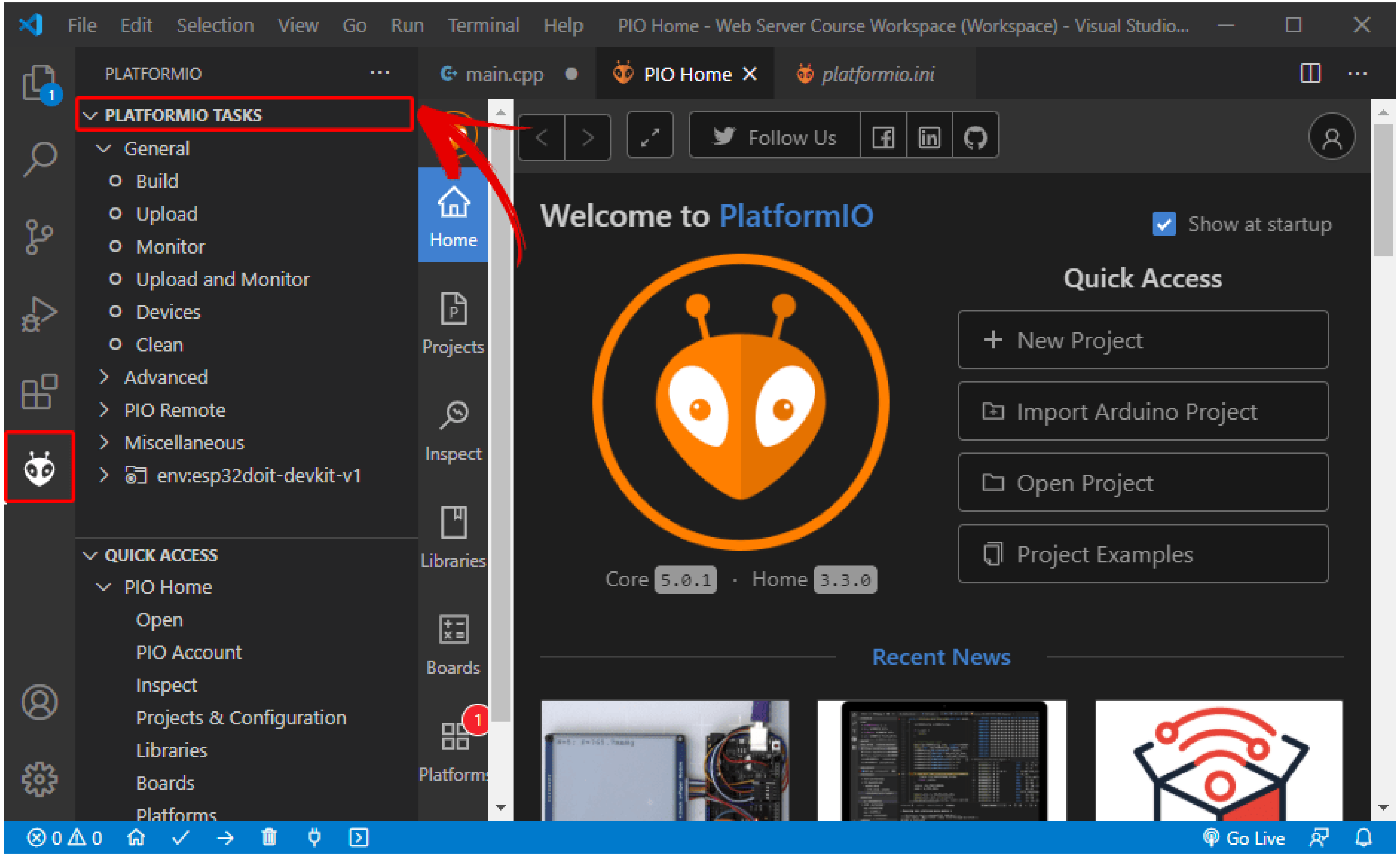
↖ Find and run all commands Ctrl+Shift+P

View errors and warnings Ctrl+Shift+M

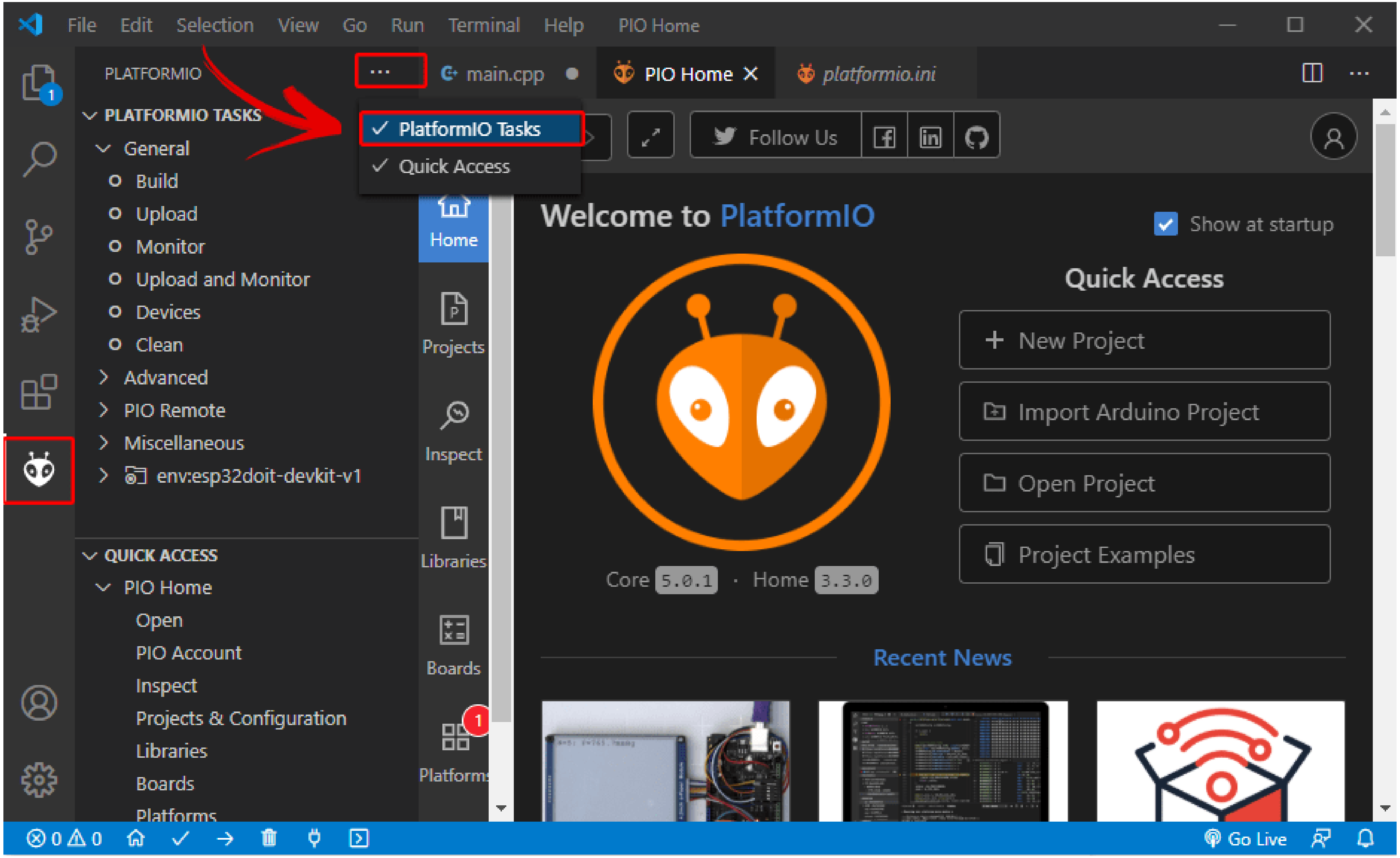
Toggle integrated terminal Ctrl+ç

Show notifications

PIO Task

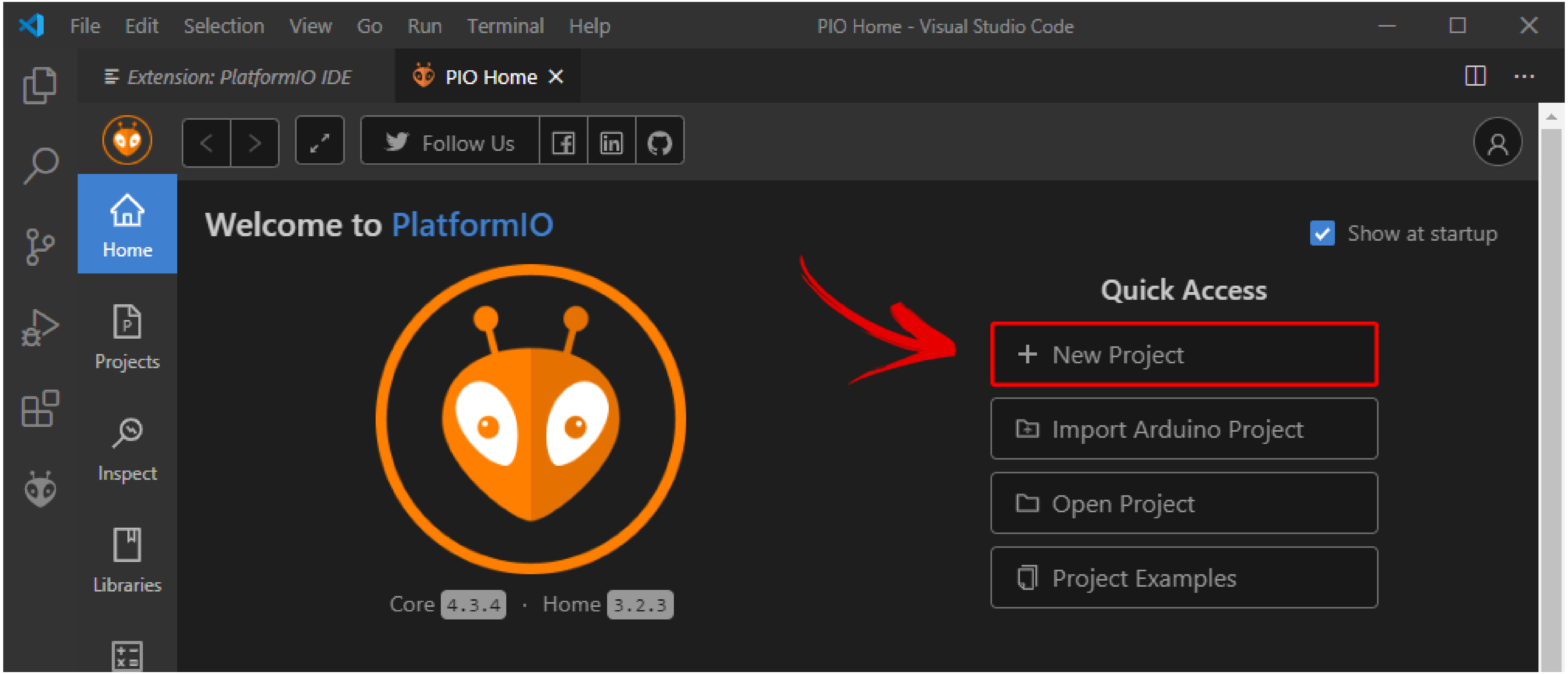


File Edit Selection View Go Run Terminal Help PIO Home - Web Server Course Workspace (Workspace) - Visual Studio...
PLATFORMIO
PLATFORMIO TASKS
General
Build
Upload
Monitor
Upload and Monitor
Devices
Clean
Advanced
PIO Remote
Miscellaneous
env:esp32doit-devkit-v1
QUICK ACCESS
PIO Home
Open
PIO Account
Inspect
Projects & Configuration
Libraries
Boards
Platforms
main.cpp
PIO Home
platformio.ini
Home
Projects
Inspect
Libraries
Boards
Platforms
Follow Us
Welcome to PlatformIO
Show at startup
Quick Access
New Project
Import Arduino Project
Open Project
Project Examples
Core 5.0.1 · Home 3.3.0
Recent News
Go Live


File
Edit
Selection
View
Go
Run
Terminal
Help
PIO Home
PLATFORMIO
...
main.cpp
PIO Home
platformio.ini
✓ PlatformIO Tasks
✓ Quick Access
PLATFORMIO TASKS
General
Build
Upload
Monitor
Upload and Monitor
Devices
Clean
Advanced
PIO Remote
Miscellaneous
env:esp32doit-devkit-v1
QUICK ACCESS
PIO Home
Open
PIO Account
Inspect
Projects & Configuration
Libraries
Boards
Platforms
Home
Projects
Inspect
Libraries
Boards
Platforms
Follow Us
Welcome to PlatformIO
Show at startup
Quick Access
New Project
Import Arduino Project
Open Project
Project Examples
Core 5.0.1 · Home 3.3.0
Recent News
Go Live

Project

1. Klik +New Project dari PIO Home
2. Pilih board Anda, misal Wemos D1 Mini
3. Jangan centang **Use default location**. Pilih Folder Anda sendiri
4. Klik **Finish*



Welcome to PlatformIO

☑ Show at startup

Quick Access

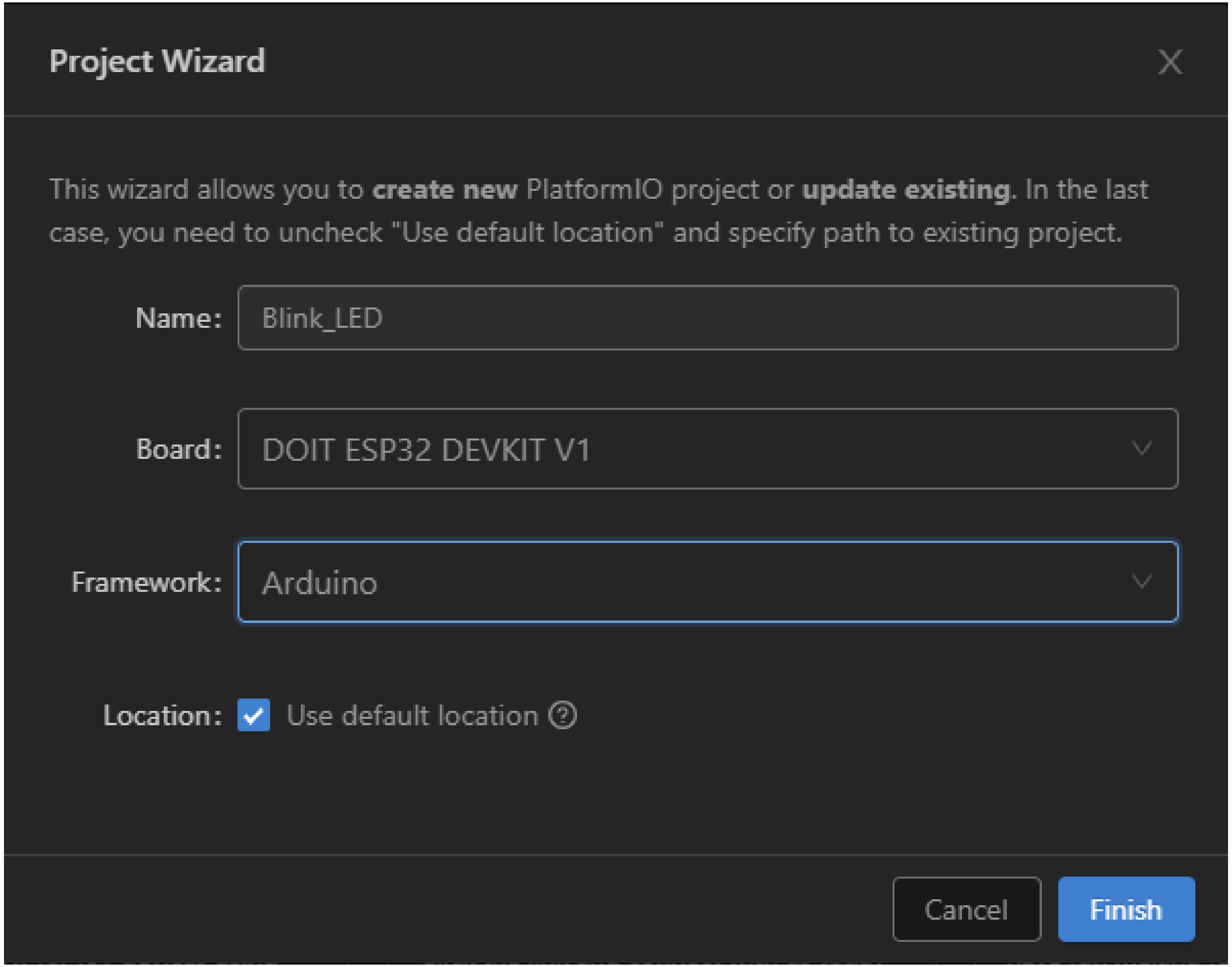
+ New Project

Import Arduino Project

Open Project

Project Examples

Core 4.3.4 · Home 3.2.3



Project Wizard

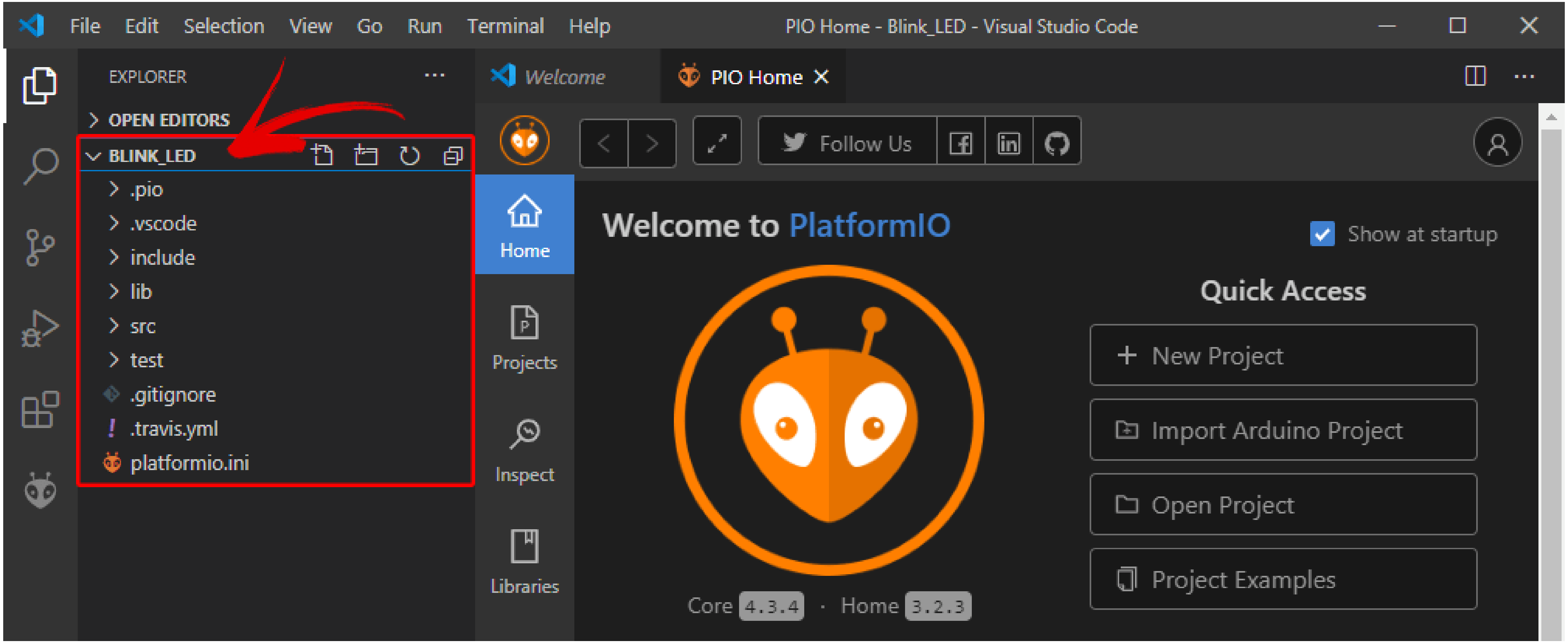
This wizard allows you to **create new** PlatformIO project or **update existing**. In the last case, you need to uncheck "Use default location" and specify path to existing project.

Name: Blink_LED

Board: DOIT ESP32 DEVKIT V1

Framework: Arduino

Location: ☑ Use default location


File
Edit
Selection
View
Go
Run
Terminal
Help
PIO Home - Blink_LED - Visual Studio Code
EXPLORER
OPEN EDITORS
BLINK_LED
.pio
.vscode
include
lib
src
test
.gitignore
.travis.yml
platformio.ini
Welcome
PIO Home
Follow Us
Home
Projects
Inspect
Libraries
Welcome to PlatformIO
Show at startup
Quick Access
New Project
Import Arduino Project
Open Project
Project Examples
Core 4.3.4 · Home 3.2.3

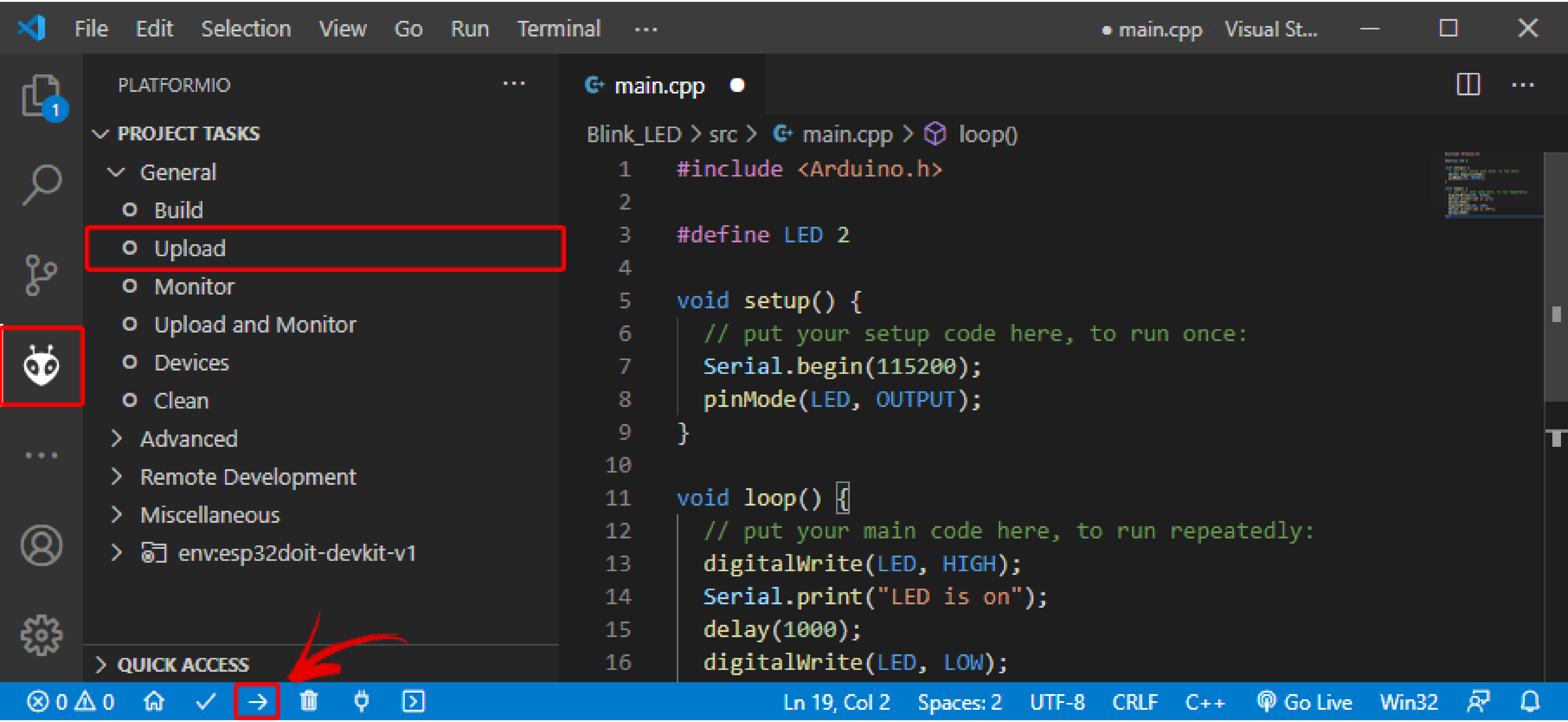
Kompilasi Program

Klik kanan folder **src** dan buat file baru misal **main.cpp**. Isikan dengan kode berikut ini :

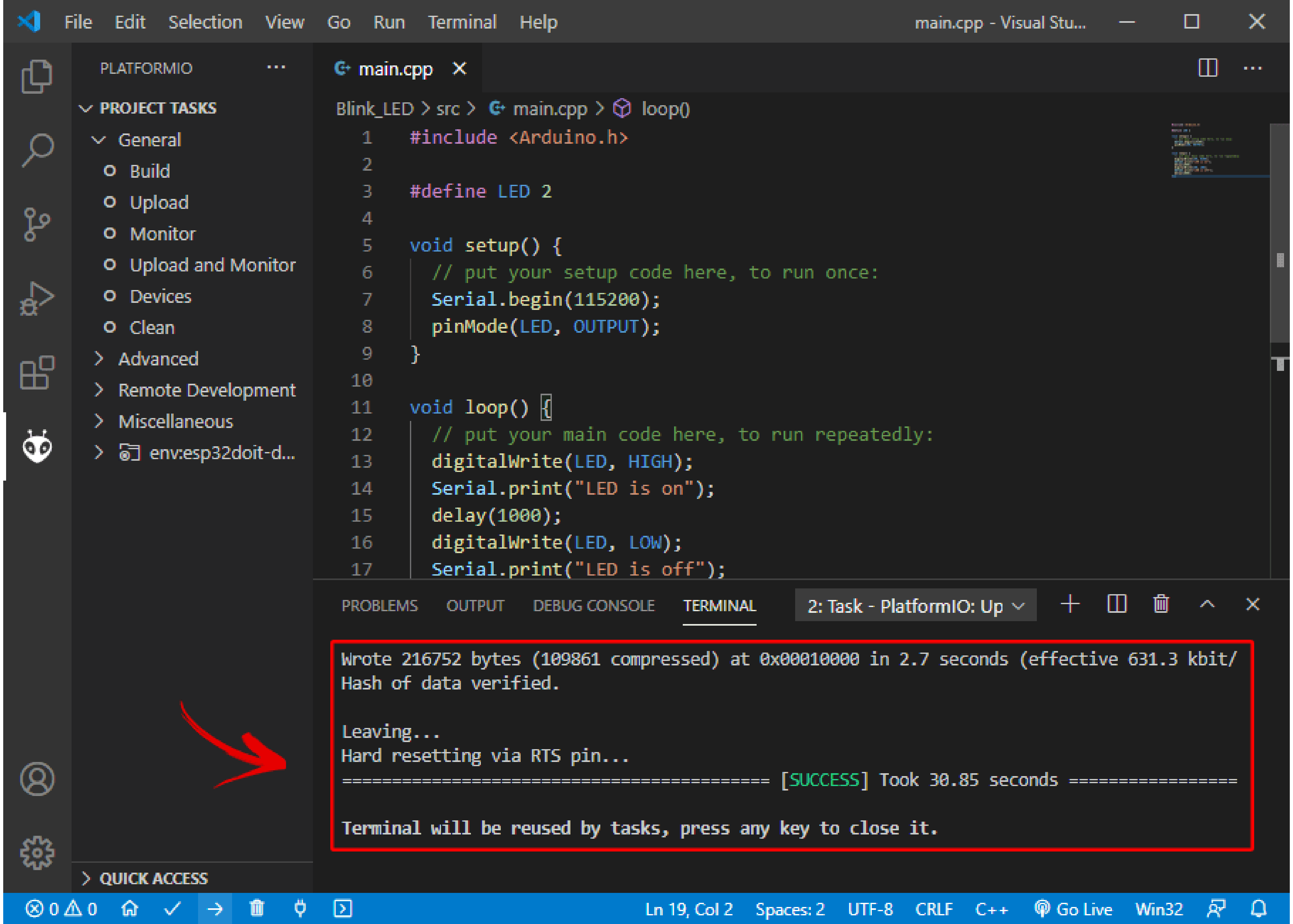
```
#include <Arduino.h>

#define LED LED_BUILTIN

void setup() {
  Serial.begin(115200);
  pinMode(LED, OUTPUT);
}

void loop() {
  digitalWrite(LED, HIGH);
  Serial.println("LED is on");
  delay(1000);
  digitalWrite(LED, LOW);
  Serial.println("LED is off");
  delay(1000);
}
```

Build dan Upload

1. Klik pilihan **Build** untuk melakukan kompilasi program Anda
2. Klik **Upload** untuk memprogram board Anda. Pastikan board Anda sudah dikenali di Device Manager

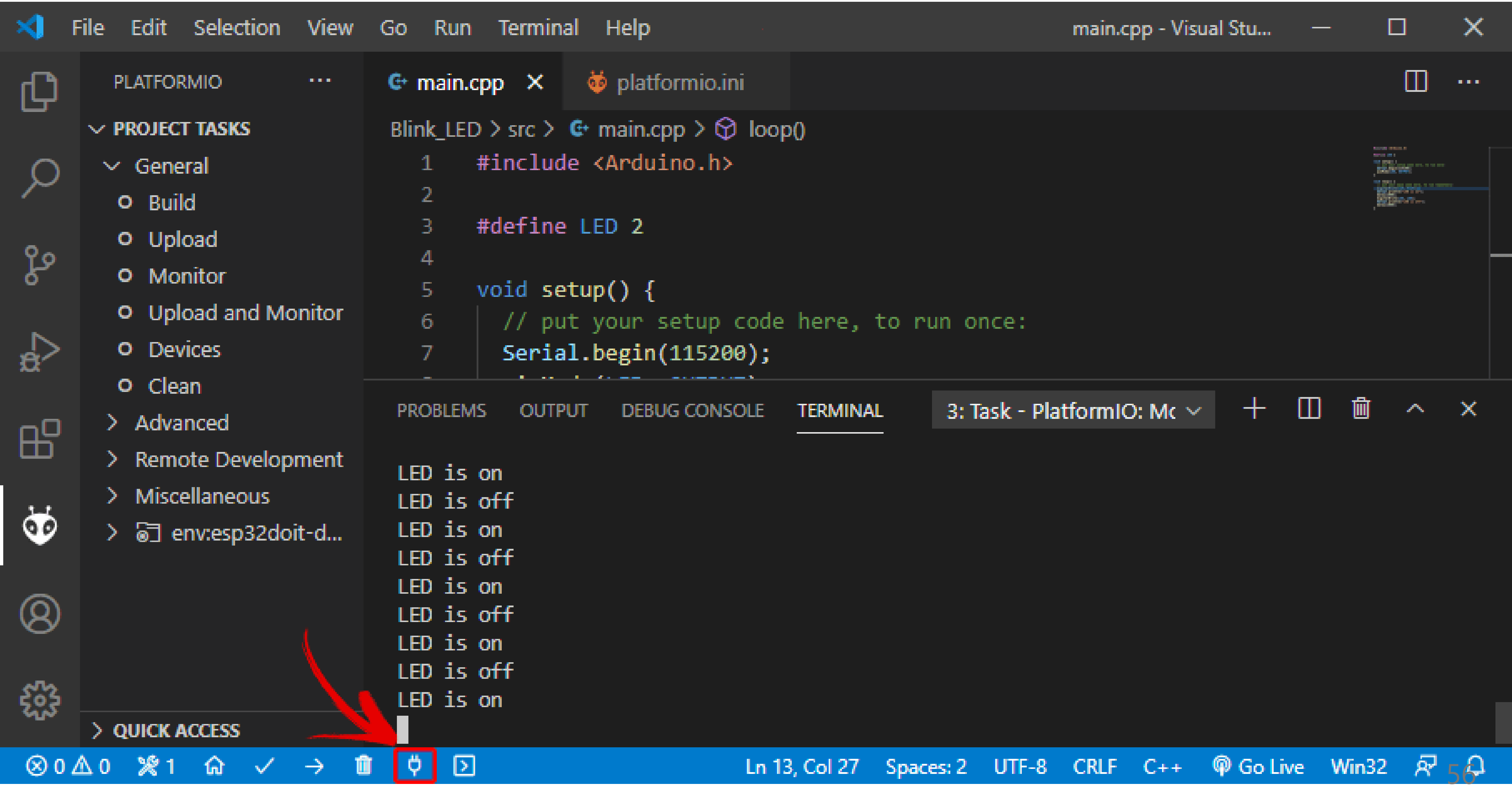


```
#include <Arduino.h>

#define LED 2

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(LED, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(LED, HIGH);
  Serial.print("LED is on");
  delay(1000);
  digitalWrite(LED, LOW);
```



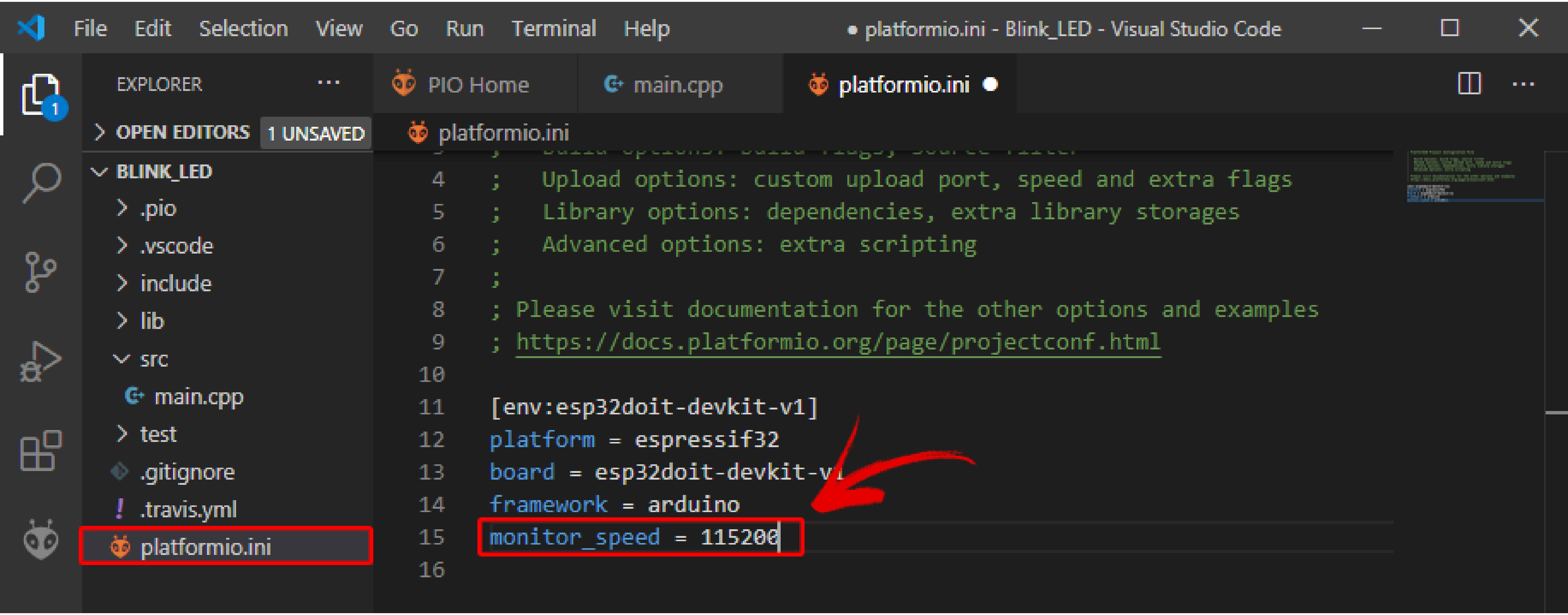
```cpp
#include <Arduino.h>

#define LED 2

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(LED, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(LED, HIGH);
  Serial.print("LED is on");
  delay(1000);
  digitalWrite(LED, LOW);
  Serial.print("LED is off");
```

```
Wrote 216752 bytes (109861 compressed) at 0x00010000 in 2.7 seconds (effective 631.3 kbit/
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
======================================= [SUCCESS] Took 30.85 seconds ================

Terminal will be reused by tasks, press any key to close it.
```



```
#include <Arduino.h>

#define LED 2

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
```

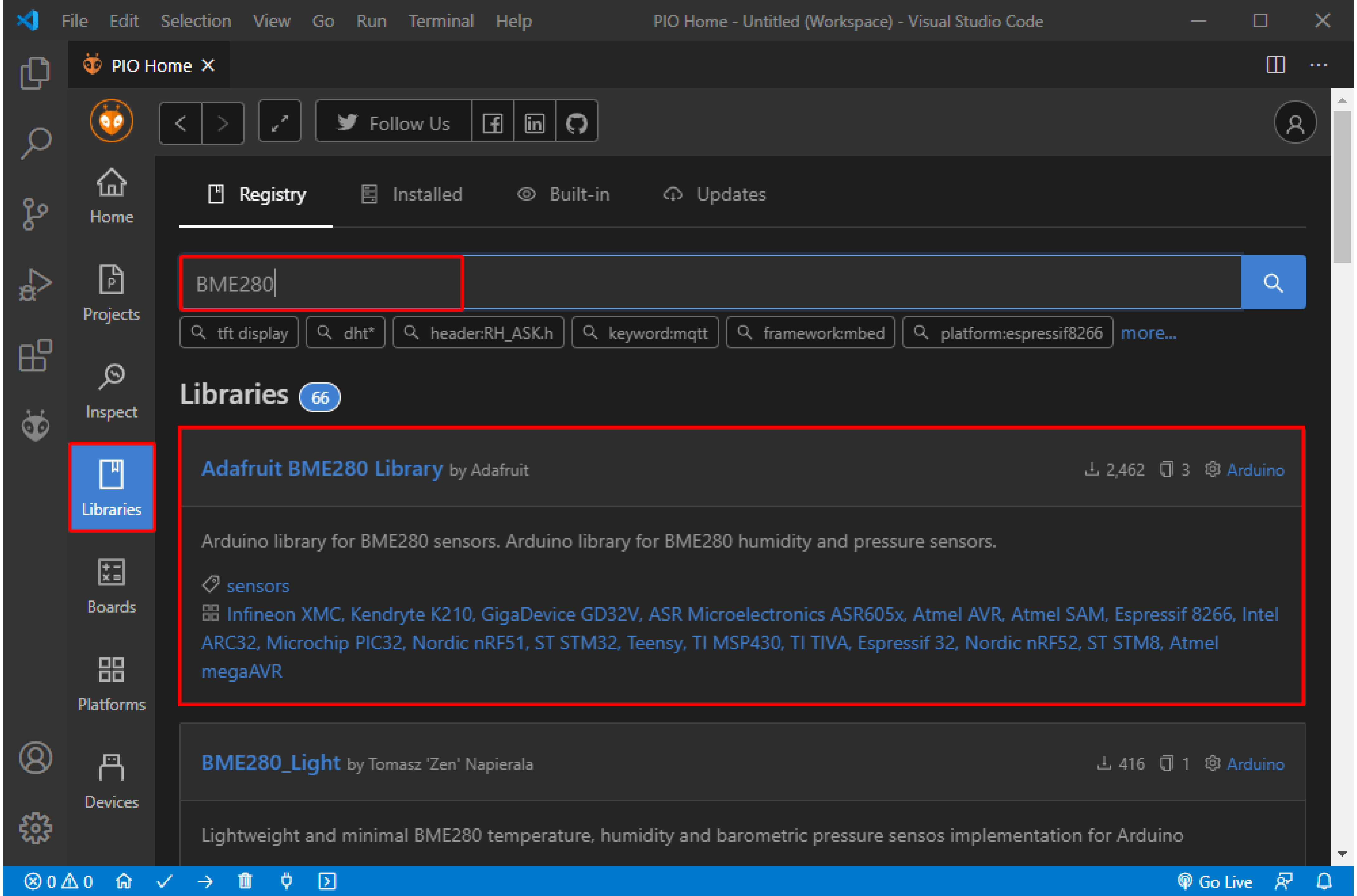
```
LED is on
LED is off
LED is on
LED is off
LED is on
LED is off
LED is on
LED is off
LED is on
```



```
;   Upload options: custom upload port, speed and extra flags
;   Library options: dependencies, extra library storages
;   Advanced options: extra scripting
;
; Please visit documentation for the other options and examples
; https://docs.platformio.org/page/projectconf.html

[env:esp32doit-devkit-v1]
platform = espressif32
board = esp32doit-devkit-v1
framework = arduino
monitor_speed = 115200
```

Instalasi Library

- Bisa secara manual (saya sarankan manual saja)
- Menggunakan library manager PIO



BME280

Libraries 66

Adafruit BME280 Library by Adafruit — 2,462 · 3 · Arduino

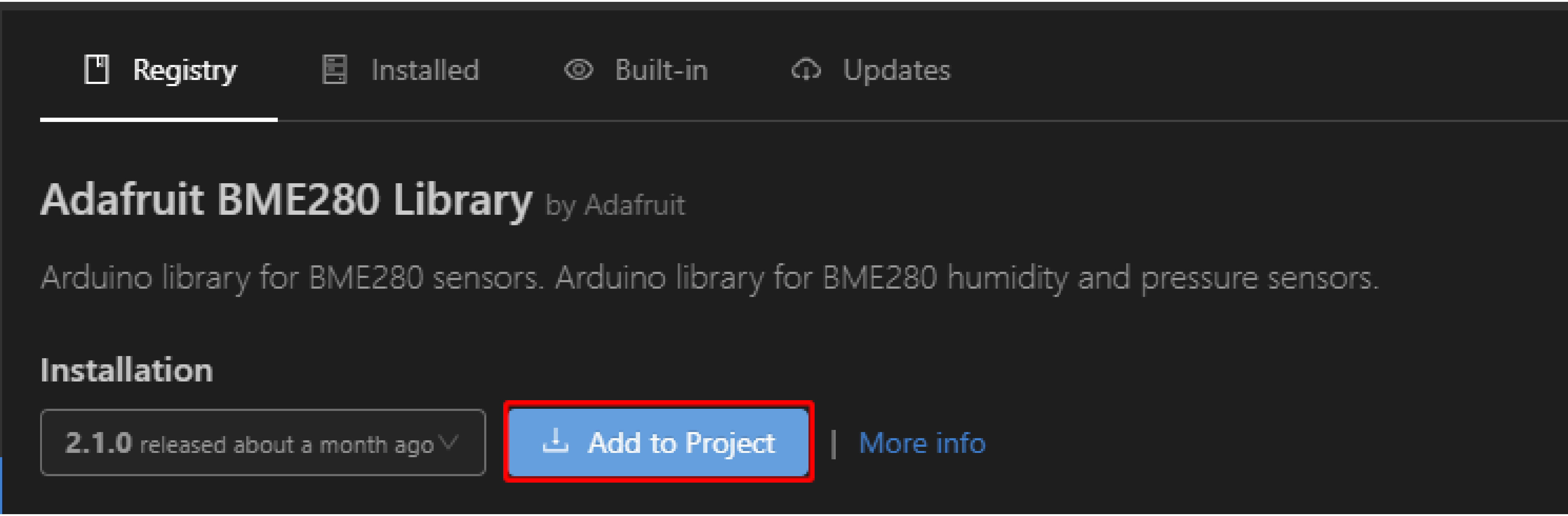
Arduino library for BME280 sensors. Arduino library for BME280 humidity and pressure sensors.

sensors

Infineon XMC, Kendryte K210, GigaDevice GD32V, ASR Microelectronics ASR605x, Atmel AVR, Atmel SAM, Espressif 8266, Intel ARC32, Microchip PIC32, Nordic nRF51, ST STM32, Teensy, TI MSP430, TI TIVA, Espressif 32, Nordic nRF52, ST STM8, Atmel megaAVR

BME280_Light by Tomasz 'Zen' Napierala — 416 · 1 · Arduino

Lightweight and minimal BME280 temperature, humidity and barometric pressure sensos implementation for Arduino

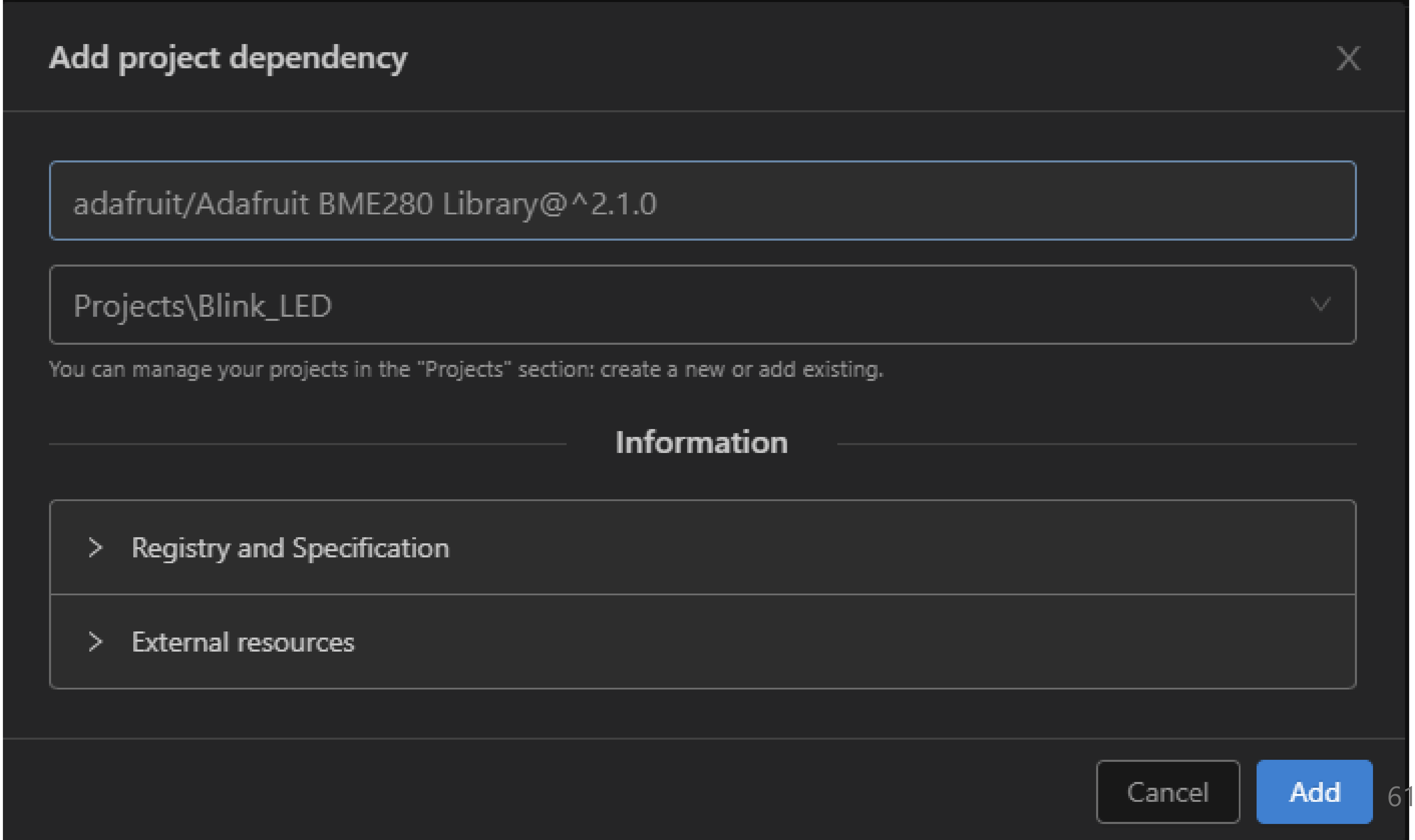


Adafruit BME280 Library by Adafruit

Arduino library for BME280 sensors. Arduino library for BME280 humidity and pressure sensors.

Installation

2.1.0 released about a month ago

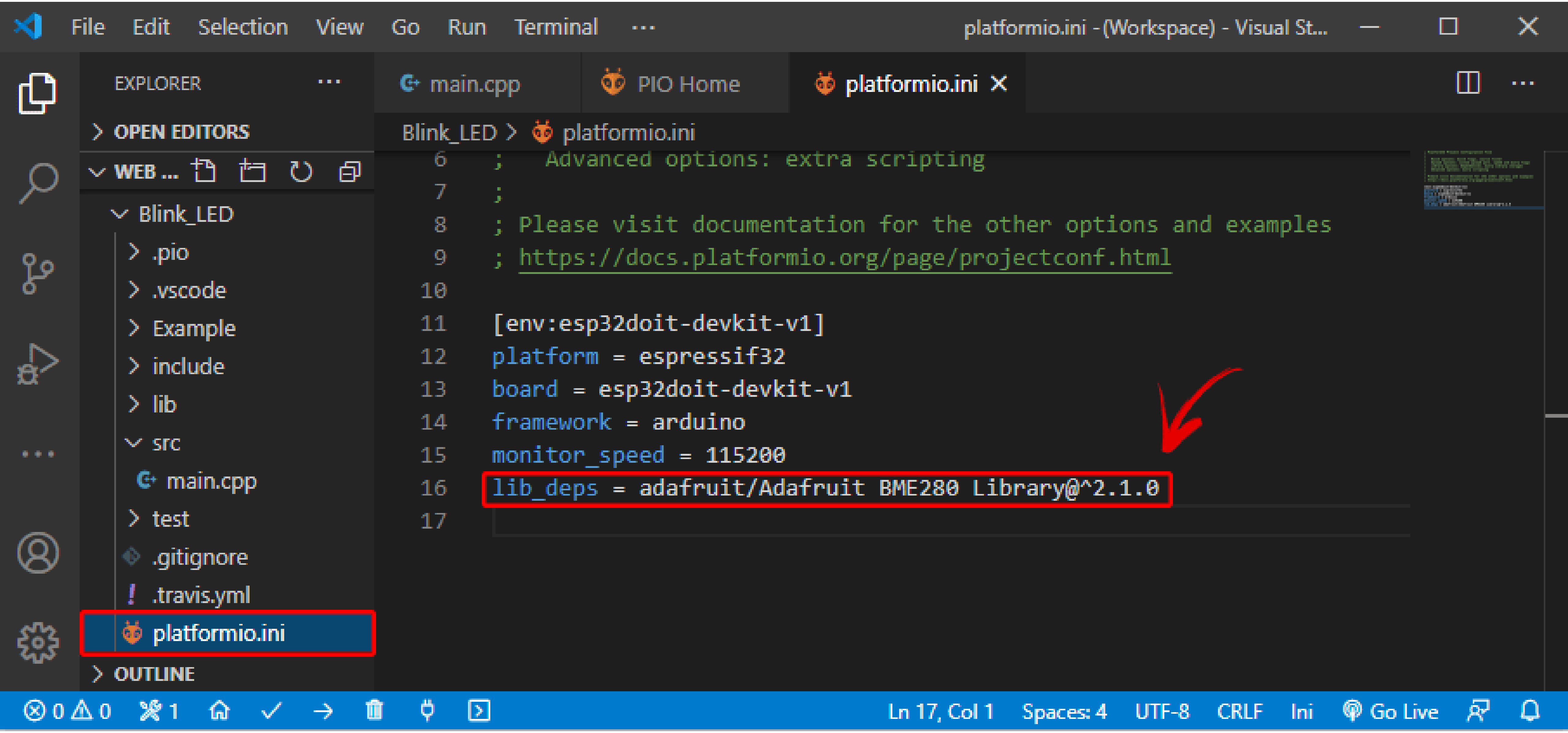


Add project dependency

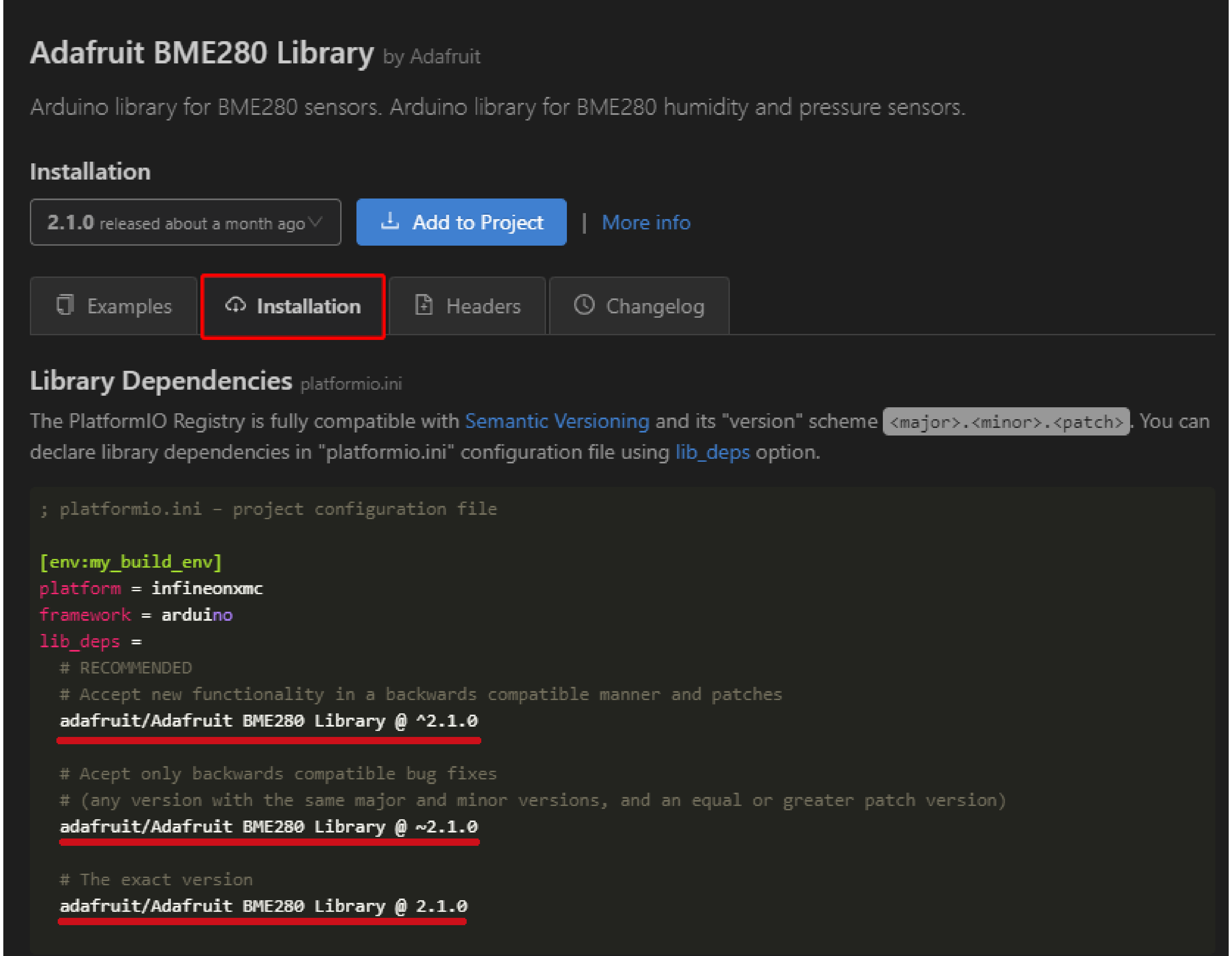
adafruit/Adafruit BME280 Library@^2.1.0

Projects\Blink_LED

You can manage your projects in the "Projects" section: create a new or add existing.



```ini
;   Advanced options: extra scripting
;
; Please visit documentation for the other options and examples
; https://docs.platformio.org/page/projectconf.html

[env:esp32doit-devkit-v1]
platform = espressif32
board = esp32doit-devkit-v1
framework = arduino
monitor_speed = 115200
lib_deps = adafruit/Adafruit BME280 Library@^2.1.0

```



Adafruit BME280 Library by Adafruit

Arduino library for BME280 sensors. Arduino library for BME280 humidity and pressure sensors.

Installation

Library Dependencies platformio.ini

The PlatformIO Registry is fully compatible with Semantic Versioning and its "version" scheme `<major>.<minor>.<patch>`. You can declare library dependencies in "platformio.ini" configuration file using lib_deps option.

```
; platformio.ini – project configuration file

[env:my_build_env]
platform = infineonxmc
framework = arduino
lib_deps =
  # RECOMMENDED
  # Accept new functionality in a backwards compatible manner and patches
  adafruit/Adafruit BME280 Library @ ^2.1.0

  # Acept only backwards compatible bug fixes
  # (any version with the same major and minor versions, and an equal or greater patch version)
  adafruit/Adafruit BME280 Library @ ~2.1.0

  # The exact version
  adafruit/Adafruit BME280 Library @ 2.1.0
```

Kenapa ATOM / VSCODE???

- Deteksi port serial otomatis
- Autocomplete
- Error Highlight
- Multiple tabs
- Code folding
- Advance code navigation

THANK YOU VERY MUCH