

Содержание

- platformio** 3
 - Установка 3
 - CLI 3
- platformio.ini*** 4
 - Секция platformio 4
 - Common env 5
 - Working env 5
- Конфигурации сборки** 6

platformio

PlatformIO - профессиональный кросс-платформенный набор инструментов для инженеров и программистов встраиваемых систем.

Можно поставить в виде расширения (основанные на PlatformIO Core) на многие текстовые редакторы, а можно использовать напрямую PlatformIO Core CLI.

Установка

<https://docs.platformio.org/en/stable/core/installation/index.html>

Скачать ¹⁾ и запустить скрипт

```
# requirements: Python 3.6+

# using 'curl'
curl -fsSL -o get-platformio.py
https://raw.githubusercontent.com/platformio/platformio-core-
installer/master/get-platformio.py
# using 'wget'
wget -O get-platformio.py
https://raw.githubusercontent.com/platformio/platformio-core-
installer/master/get-platformio.py
# run
python3 get-platformio.py
```

CLI

```
Usage: pio [OPTIONS] COMMAND [ARGS]...
```

Options:

```
--version      Показывает версию и выходит
-c, --caller TEXT Caller ID (service)
--no-ansi      Не печатать управляющие символы ANSI
-h, --help     Покажет этот текст (справку)
```

Commands:

```
access      Manage resource access
account     Manage PlatformIO account
boards      Обзоратель плат
check       Static Code Analysis
ci          Continuous Integration
debug       Unified Debugger
device      Device manager & Serial/Socket monitor
home        веб-интерфейс для работы с PlatformIO
org         Manage organizations
pkg         Unified Package Manager
```

project	Project Manager
remote	Remote Development
run	Run project targets (build, upload, clean, etc.)
settings	Manage system settings
system	Miscellaneous system commands
team	Manage organization teams
test	Unit Testing
upgrade	Обновляет PlatformIO Core до последней версии

platformio.ini

... - конфигурационный ini файл проекта который находится в корневом каталоге проекта. Состоит из разделов ([header]) и пар ключ-значение. Все, что написано после ; игнорируется (типа комментариев). Задать несколько параметров можно или через «, », или разделяя переносом строки + отступ (2я пробела).

Секция platformio

Секция [platformio]²⁾ используется для перезаписи default configuration options для PlatformIO Core:

- Generic options:
 - name
 - description
 - default_envs
 - extra_configs
- Directory options:
 - core_dir
 - globallib_dir
 - platforms_dir
 - packages_dir
 - cache_dir
 - build_cache_dir
 - workspace_dir
 - build_dir
 - libdeps_dir
 - include_dir
 - src_dir
 - lib_dir
 - data_dir
 - test_dir
 - boards_dir
 - monitor_dir
 - shared_dir

Common env

Секция `[env]`³⁾ предустанавливает параметры для всех сред конфигурации (рабочих сред), типа `[env:NAME]`.

Пример:

```
[env]
platform = ststm32
framework = stm32cube
board = nucleo_l152re
lib_deps = Dep1, Dep2

[env:release]
build_flags = -D RELEASE
lib_deps =
    ${env.lib_deps}
    Dep3
.....
Равносильно:
.....ini
[env:release]
platform = ststm32
framework = stm32cube
board = nucleo_l152re
build_flags = -D RELEASE
lib_deps = Dep1, Dep2, Dep3
```

Working env

В каждом проекте должна быть хотя бы одна `_working environment_` (рабочая среда). Такая среда используется для выполнения команд (по типу `pio run`, `pio test`, `pio check`, `pio debug`) и выглядит примерно так: `[env:NAME]`.

Вы можете определить несколько рабочих сред в своем проекте, используя разные имена для раздела `[env:NAME]`. Важно отметить, что каждая среда должна иметь уникальное имя. Имя должно содержать только строчные буквы a-z, цифры 0-9 и специальные символы («_» и «-»). Например, `[env:hello_world]` - допустимое имя.

При помощи опции `-e/--environment` можно указать конкретную рабочую среду, в которой будет выполнена команда. Так же опция `default_envs` из `[platformio]` позволяет определить среды, для использования по-умолчанию.

```
Пример файла проекта <code ini> [platformio] default_envs = nodemcu2 ; custom common
options [common] build_flags = -D VERSION=1.2.3 -D DEBUG=1 lib_deps_builtin = SPI Wire
lib_deps_external = bblanchon/ArduinoJson @ ~5.6,!5.4 https://github.com/gioblu/PJON.git#v2.0
IRremoteESP8266=https://github.com/markszabo/IRremoteESP8266/archive/master.zip
[env:nodemcu2] platform = espressif8266 framework = arduino board = nodemcu2 ; Build options
build_flags = ${common.build_flags} -DSSID_NAME=HELLO -DSSID_PASSWORD=WORLD ; Library
```

```
options lib_deps = ${common.lib_deps_builtin} ${common.lib_deps_external}
https://github.com/me-no-dev/ESPAsyncTCP.git knolleary/PubSubClient@^2.8 paulstoffregen/OneWire
; Serial Monitor options monitor_speed = 115200 monitor_flags = --encoding hexlify ; Unit Testing
options test_ignore = test_desktop [env:bluepill_f103c8] platform = ststm32 framework = arduino
board = bluepill_f103c8 ; Library options lib_deps = ${common.lib_deps_external} ; Debug options
debug_tool = custom debug_server = ${platformio.packages_dir}/tool-jlink/JLinkGDBServer -singlerun
-if SWD -select USB -port 2331 -device STM32F103C8 ; Unit Testing options test_ignore =
test_desktop </code>
```

Конфигурации сборки

В PlatformIO 3 типа сборочной конфигурации:

- `release` (используется по-умолчанию) - Релизная конфигурация прошивки/программы не содержит отладочной информации и оптимизирована под платформу разработки.
- `test` - включает макрос `PIO_UNIT_TESTING` и флаги фреймворка юнит-тестирования.
- `debug` - Отладочная конфигурация прошивки/программы компилируется с полной отладочной информацией и без оптимизации. (Оптимизация усложняет отладку, поскольку связь между исходным кодом и сгенерированными инструкциями весьма запутанная)

Задать конфигурацию сборки можно через `build_type` в `platformio.ini` или указать `debug` в качестве цели в `pio run --target`

https://docs.platformio.org/en/stable/projectconf/build_configurations.html

!Pasted image 20240427101451.png

1)
<https://raw.githubusercontent.com/platformio/platformio-core-installer/master/get-platformio.py>

2)
<https://docs.platformio.org/en/stable/projectconf/sections/platformio/index.html>

3)
<https://docs.platformio.org/en/stable/projectconf/sections/env/index.html>

From:
<https://wiki.radi0.cc/> - **radi0wiki**

Permanent link:
<https://wiki.radi0.cc/soft:platformio>

Last update: **2025/11/09 12:07**

