

Simple PIOC

Introduction

This Python script processes a custom PlatformIO configuration file (`platformio.pioc`) and generates a standard `platformio.ini` file.

It extends PlatformIO's configuration capabilities by:

- Supporting **dynamic include paths** using glob patterns
- Extracting **C preprocessor defines** from board-specific Makefiles
- Expanding **custom syntax** into valid `build_flags`
- Resolving **absolute paths**.

Key Features

Custom `build_flags` Processing

Supports two types of entries:

- **+<pattern>**: Include directories
- **-<pattern>**: Exclude directories
- Other entries are treated as standard compiler flags

Include Path Resolution

Glob patterns are expanded into directory paths using recursive search.

Board-specific C Defines

Defines are extracted from:

```
components/<board>/firmware/Makefile
```

The script looks for a `C_DEFS` section and includes all compiler defines.

This is done, because cubeMX generates important definitions in the auto-generated makefile. These are not used if we don't copy them to the `.ini` file.

Environment Detection

```
[env:my_board]
```

This determines which board folder is used.

Get Absolute Path

For some functions, like `nanopb`, you might need the absolute path. There is no way to get in the default `platformio.ini` file, so that would mean that you would have to hard code it. We do not want that, so we have a placeholder for an absolute path.

The placeholder:

```
${{project_absolute_path}}$
```

is replaced with the absolute path of the project.

Workflow

1. Read `platformio.pio`
2. Detect environment
3. Parse `build_flags`
4. Resolve glob patterns
5. Extract C defines
6. Write `platformio.ini`
7. Replace placeholders

Example Input

```
[env:my_board]
build_flags =
  +<lib/**>
  -<lib/exclude/**>
  -DDEBUG
```

Example Output

```
[env:my_board]
build_flags =
  -I lib/module1
  -I lib/module2
  -DDEBUG
  -DDEFINE_FROM_MAKEFILE
```

Revision #2

Created 2026-04-16 13:07:00 UTC by Nikolaos Diamantopoulos

Updated 2026-04-17 13:02:32 UTC by Jaron Lendering