# PlatformIO.org

## Embedded Development made easy

## Pi and More 9

Nico Maas

# Details

**Nico Maas**

**IT Systemelektroniker**

**Bachelor of Science**

**mail@nico-maas.de**

**www.nico-maas.de**

**@nmaas87**

# Agenda

- **I. Introduction**
  - What is PlatformIO?
  - Which hardware / frameworks are supported?
  - Which special features do exist?
- **II. Demo**
  - Installation
  - Demo with Arduino Uno / Arduino Due
  - Demo with ESP8266
- **III. End**

# Introduction

**PlatformIO is an open source ecosystem for IoT development.**

Cross-platform code builder.

Pure Python.

Windows, Linux, Mac.

Continuous Integration and IDE integration.

Arduino and mbed compatible (and a lot more…)

# Introduction

200+ Boards

15+ Platforms:

Atmel AVR, Atmel SAM, Espressif, Freescale Kinetis, Nordic nRF51,

NXP LPC, Silicon Labs EFM32, ST STM32, Teensy, TI MSP430, TI TIVA

10+ Frameworks:

Arduino, CMSIS, WiringPi, libOpenCM3, Energia, SPL, mbed

CLI and IDE (Atom.io)

# Introduction

Easy Development for (nearly) every platform without the need to install additional tools and IDEs.

Crossplatform development.

Build one project for different MCUs.

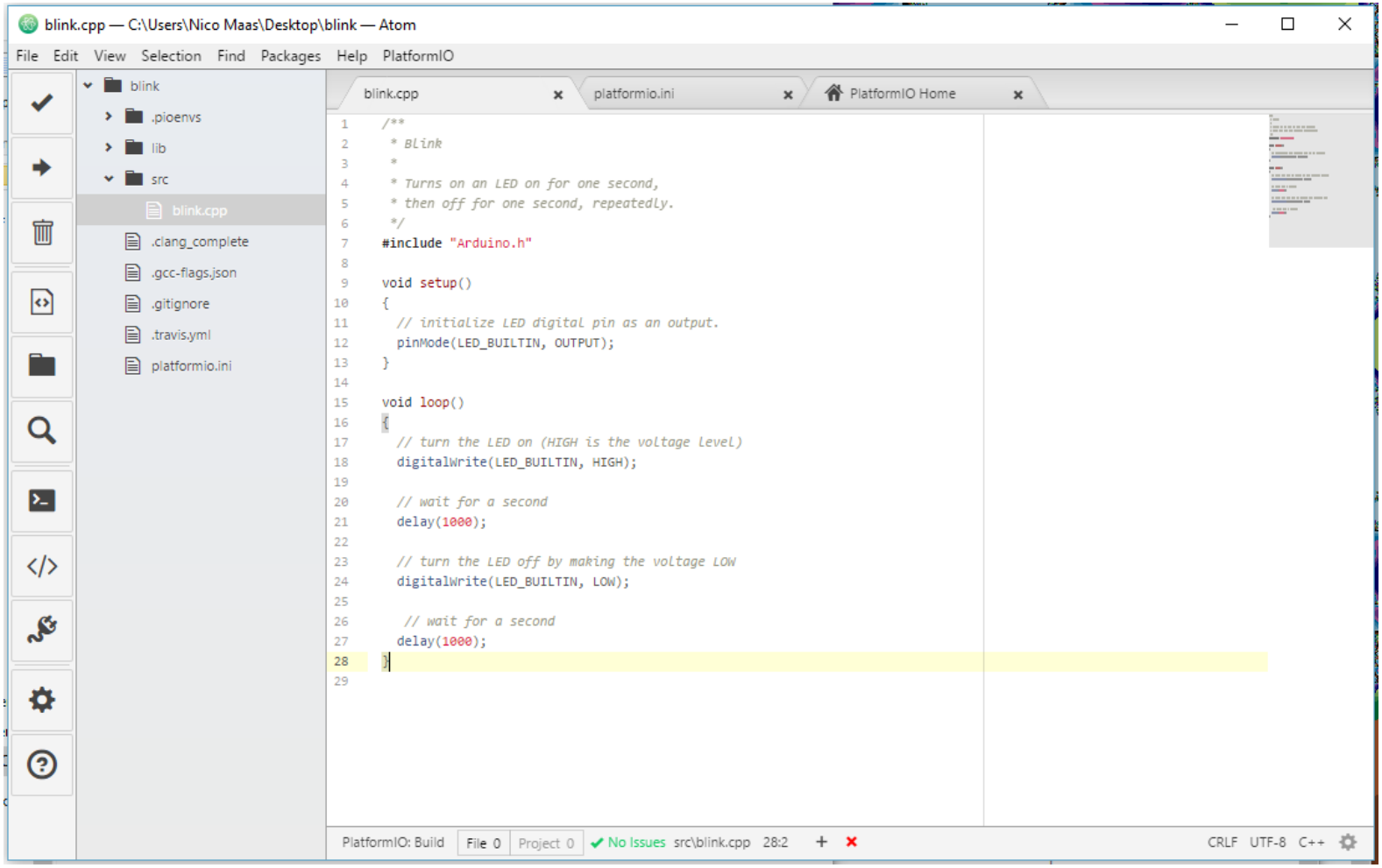
Library Manager (CLI, Web)

CI Integration (Drone, Travis,…)

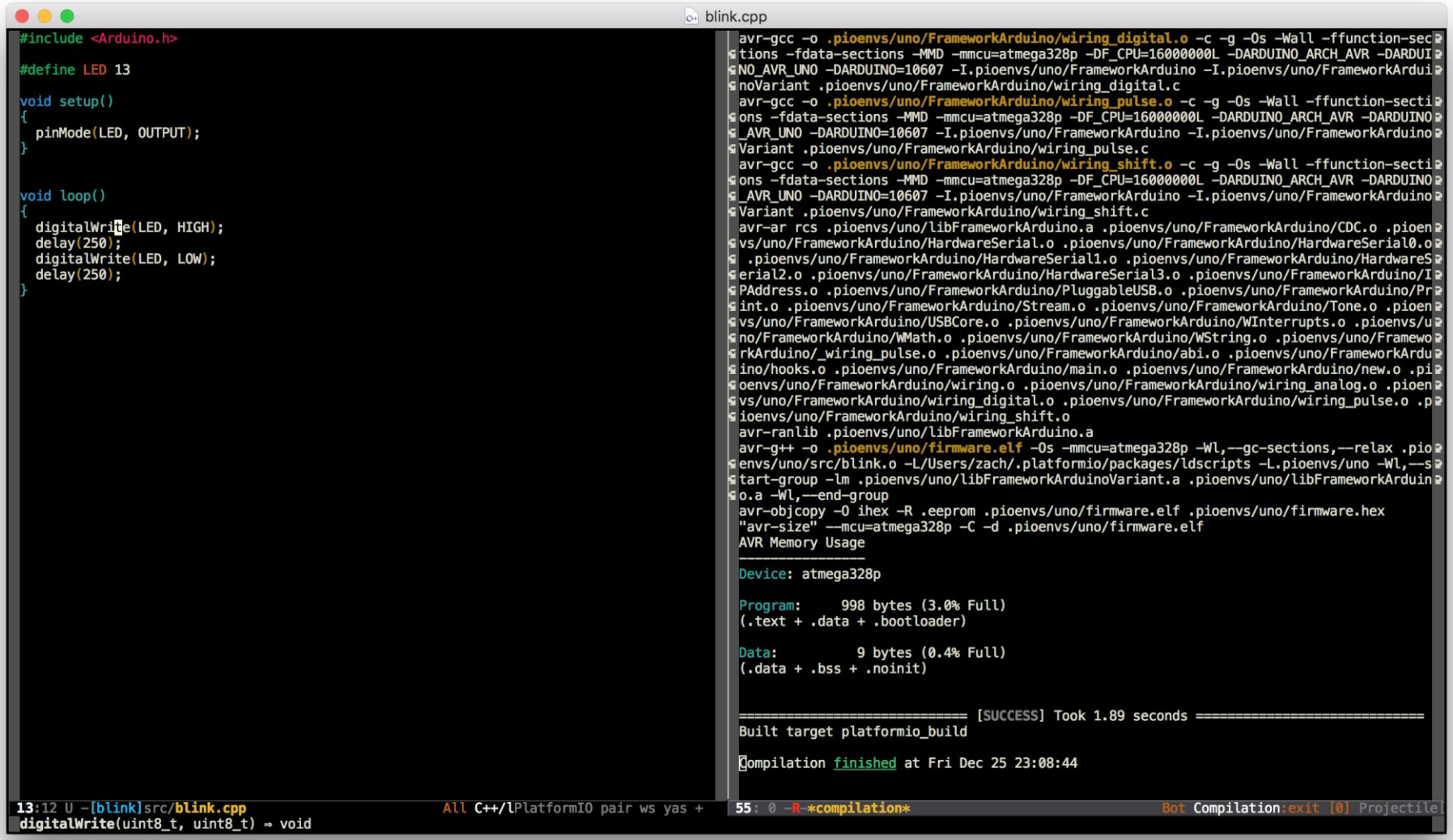
IDE Integration (Eclipse, Netbeans, EMACs, VIM, Sublime Text,…)

# PlatformIO IDE

The next generation integrated development environment for IoT



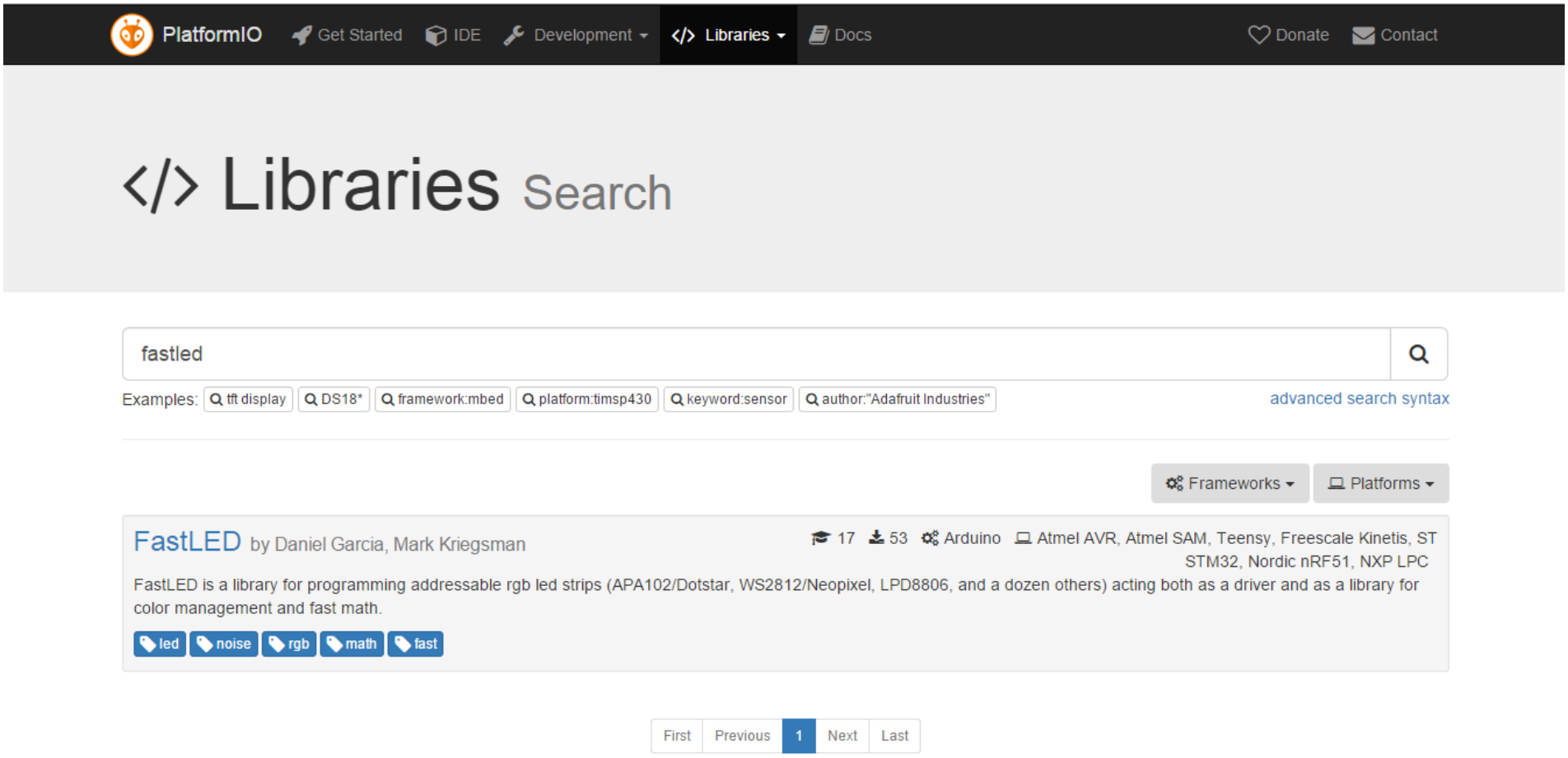
# IDE Integration Emacs



# PlatformIO CLI

```
(platformio-demo)➜ wiring-blink  platformio run
[Thu Aug 27 18:29:57 2015] Processing uno (platform: atmelavr, board: uno, framework: arduino)
--------------------------------------------------------------------------------------------------------------------------------
avr-g++ -o .pioenvs/uno/src/main.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffunction-sections -fdata-s
ections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=10605 -DPLATFORMIO=02020
2 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant src/main.cpp
avr-ar rcs .pioenvs/uno/libFrameworkArduinoVariant.a
avr-ranlib .pioenvs/uno/libFrameworkArduinoVariant.a
avr-g++ -o .pioenvs/uno/FrameworkArduino/CDC.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffunction-secti
ons -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=10605 -I.pio
envs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/CDC.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HID.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffunction-secti
ons -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=10605 -I.pio
envs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HID.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HardwareSerial.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffun
ction-sections -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=1
0605 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HardwareSeria
l.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HardwareSerial0.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffu
nction-sections -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=
10605 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HardwareSeri
al0.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HardwareSerial1.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffu
nction-sections -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=
10605 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HardwareSeri
al1.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HardwareSerial2.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffu
nction-sections -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=
10605 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HardwareSeri
al2.cpp
avr-g++ -o .pioenvs/uno/FrameworkArduino/HardwareSerial3.o -c -fno-exceptions -fno-threadsafe-statics -g -Os -Wall -ffu
nction-sections -fdata-sections -MMD -mmcu=atmega328p -DF_CPU=16000000L -DARDUINO_ARCH_AVR -DARDUINO_AVR_UNO -DARDUINO=
10605 -I.pioenvs/uno/FrameworkArduino -I.pioenvs/uno/FrameworkArduinoVariant .pioenvs/uno/FrameworkArduino/HardwareSeri
al3.cpp
```

# Library Manager



# Demo - Installation

Goto: http://platformio.org/#!/get-started

Download the version for your system

Install

Done

# Demo - Usage

**Step 1**

Initialize new *PlatformIO* based project using  button on the Toolbar or `Menu: PlatformIO > Initialize new PlatformIO Project or update existing`.

**Step 2**

Put your source code `*.h, *.c, *.cpp, *.S, *.ino, etc.` files to `src` directory.

**Step 3**

Process the project environments using `Menu: PlatformIO > Build / Upload / Clean` or buttons on the Toolbar:

- Build
- Upload
- Clean

# Demo - Demonstration

Demo Time ☺!

# Questions?

Thanks a lot for your time!

# Sources

- Logos, Icons and Screenshots
  - http://www.platformio.org/