

# Необходимое программное обеспечение для работы с лабораторным комплексом Atmega32 (сенсорные устройства)

---

Для работы с лабораторным комплексом необходимы следующие программы:

- [Visual Studio Code](#)
- [Расширение PlatformIO](#)
- [Утилита AVRDUDE](#)
- [Система Git](#)

## Установка редактора Visual Studio Code

Visual Studio Code (VSCode) - кроссплатформенный редактор кода с возможностью кастомизации и расширения, в частности, под разработку ПО для микроконтроллеров. В нашем случае это МК ATmega32, с помощью которого осуществляется работы с сенсорами и сенсорными устройствами.

Visual Studio Code можно установить с [сайта](#), скачав установщик и проделав все стандартные процедуры установки.

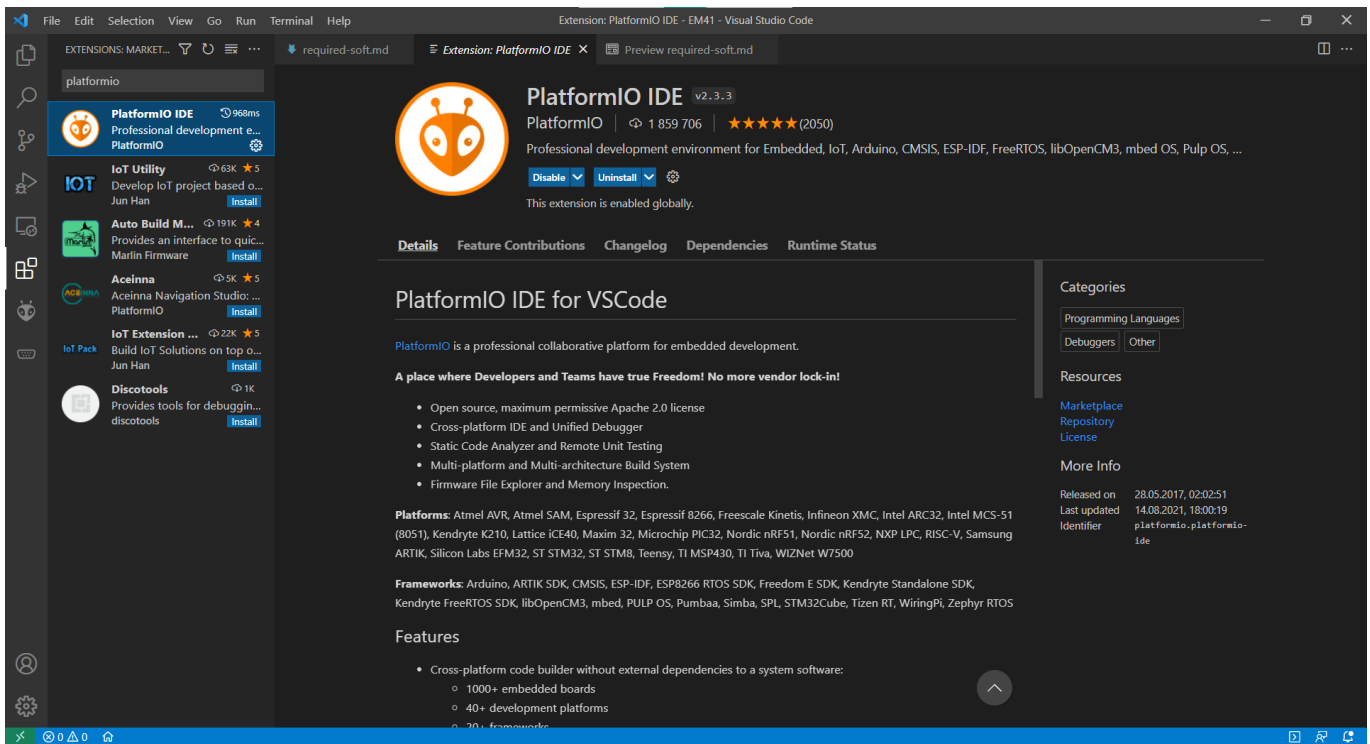
Теперь нам доступен модный и стильный редактор кода, в котором мы продолжим свою работы.

---

## Установка PlatformIO extensions

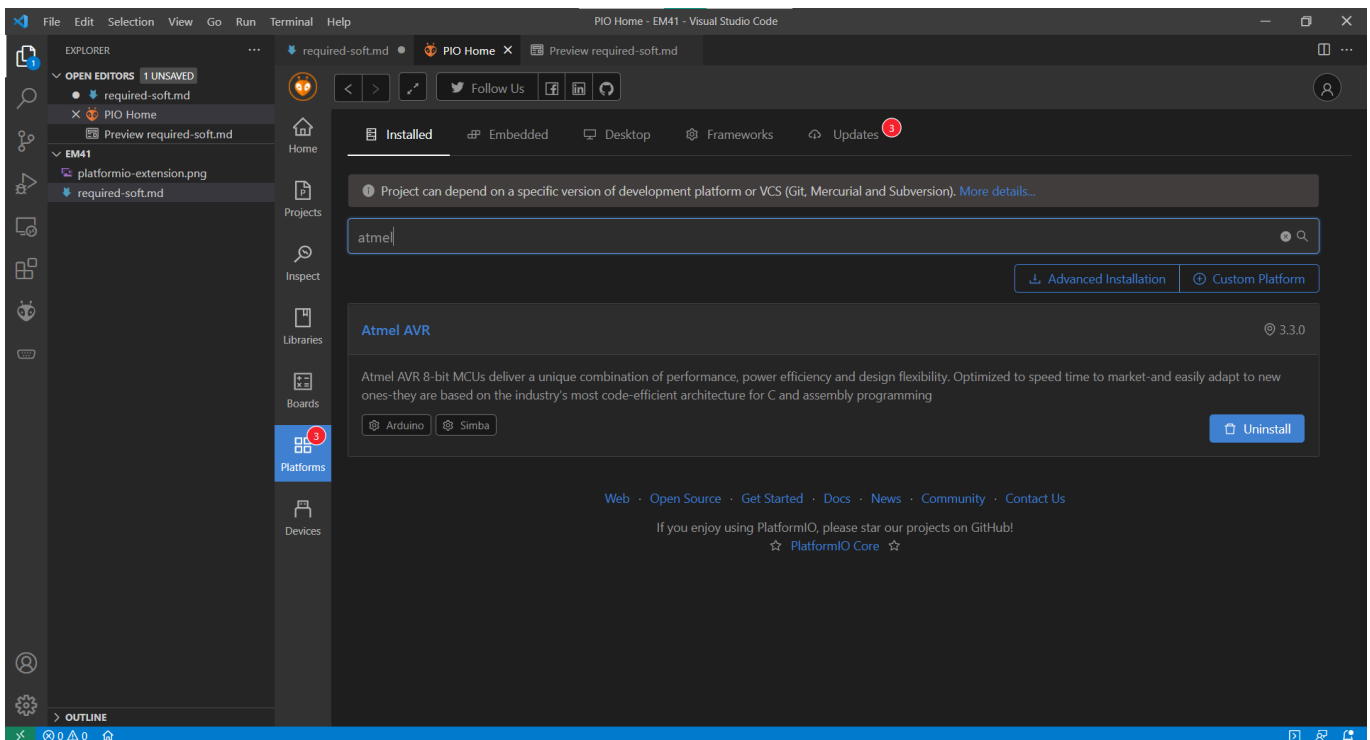
PlatformIO - экосистема для разработки ПО для встраиваемых систем. Включается в себя систему сборки под множество платформ и архитектур. С помощью PlatformIO мы будем собирать и загружать код в МК ATmega32.

Из VSCode PlatformIO доступно в виде расширения, которое можно [установить](#) внутри VSCode:



Вместе с собой это расширение поставить ещё вспомогательное - не мешать ему в этом!

После установки расширения PlatformIO переходим на домашнюю страницу расширения чтобы установить поддержку платформы Atmel AVR:



Всё, теперь можно создавать и редактировать код для лабораторных работ.

## Установка утилиты AVRDUDE

AVRDUDE - утилита для загрузки/считывания/редактирования в память микроконтроллеров AVR. Нам эта утилита будет необходима для загрузки файла прошивки в память МК.

Скачиваем архив с программой со [страницы](#) загрузки. Выбираем файл с названием `avrdude-6.3-mingw32.zip`. Затем распаковываем этот файл куда-нибудь в диск в отдельную папку. Затем **(обязательный этап!)** добавить путь к папке `\bin` в системную переменную PATH.

---

## Система контроля версий Git

Git - это распределённая система контроля версия, предназначена для сопровождения проекта, облегчая работы с изменяющимися файлами. То есть, Git позволяет хранить в себе разные версии файлов проекта (**репозитория**) и при необходимости перемещаться между версиями, видеть изменения в файлах и многое другой. Git также работает с веб-сервисами (например Github), где удалённо хранятся репозитории.

Git понадобится для работы с репозиторием ы на Github, на котором размещены лабораторные работы. Установить Git [официального сайта](#). Процесс установки можно посмотреть [здесь](#).

Также будет полезно завести аккаунт на [Github](#).