- 1.简述
- 2.安装Visual Studio Code PlatformIO
- 3.测试
- 4.问题与高级功能
- 5.注意点
- 6.总结

1.简述

介绍如何使用VScode直接开发Arduino程序,避免使用Arduino IDE时的没有代码提示功能,文件关系不清 晰、头文件打开不方便等问题及使用Visual Stdio集成插件的庞大安装工程;同时Visual Studio Code插件 PlatformIO IDE开发Arduino 跨平台无论你是用的windows,ubuntu或者mac都可以玩转。

2.安装Visual Studio Code PlatformIO

- https://code.visualstudio.com/页面下载安装vscode
- 安装完成vscode启动,扩展页面下搜索platformio即可找到,选择第一个Platformio IDE,安装即可 (这里需要耐心等待一会)



• 安装完成,重新加载后,左下角会多一个小房子图标,点击后即可显示Platformio IDE主页

3.测试

- 选择New Project创建工程,选择相应的Board,我这里使用Mega2560,输入2560找到对应的Board
- 修改main.cpp

```
#include <Arduino.h>
void setup() {
    pinMode(13, OUTPUT);
}
void loop() {
    digitalWrite(13, HIGH);
    delayMicroseconds(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```

- 编译与下载

4.问题与高级功能

Arduino IDE有库管理功能可以,下载到需要的库。这里还要方便,例如我们想使用TimerOne输出PWM,

```
#include <Arduino.h>
#include <TimerOne.h>
void setup() {
    Timer1.initialize(40);
}
void loop() {
    Timer1.pwm(11, 512);
}
```

Arduino IDE我们会这样写,然库管理搜索下载TimerOne库,在这里我们只需要在配置文件platformio.ini加上下面一句即可

lib_deps = TimerOne

选择编译按钮编译,我们可以看到输出信息

, plat	formio.ini - 171122-123034-megaatmega2560 - Visua	l Studio Code			↔	—		×
文件(E)	编辑(E) 选择(S) 查看(V) 转到(G) 调试(D) 任务(T)	帮助(<u>H</u>)						
n	资源管理器	💐 欢迎使用	≣ PIO Home	G main.cpp	≣ platformio.ini ×			
	资源管理器 ◆ 打开的编辑器 ▲ 打开的编辑器 ▲ 打开的编辑器 ■ PIO Home G main.cpp src ■ platformio.ini ▲ 171122-123034-MEGAATMEGA2560 ▶ .pioenvs ▲ .piolibdeps ▲ TimerOne_ID131 ▶ config ▶ examples () .library.json ≣ keywords.txt () library.json ፪ keywords.txt () library.json ፪ keywords.txt () library.properties () README.md G TimerOne.cpp C TimerOne.h ▶ .vscode ▶ lib ▲ src G main.cpp ◊ .gitignore ! .travis.yml E platformio.ini	<pre>SQUEER PIO Home C main.cpp E platformio.ini × 1 ; PlatformIO Project Configuration File 2 ; 3 ; Build options: build flags, source filter 4 ; Upload options: custom upload port, speed and extra flags 5 ; Library options: dependencies, extra library storages 6 ; Advanced options: extra scripting 7 ; 8 ; Please visit documentation for the other options and examples 9 ; http://docs.platformio.org/page/projectconf.html 10 11 [env:megaatmega2560] 12 platform = atmelavr 13 board = megaatmega2560 14 framework = arduino 15 lib_deps = TimerOne</pre>						
÷ ₩ • • • • • • • • • • • • • • • • • • •		问题 输出 预试验验 终端 1:任务-Build ▲ 台 ▲ 白 ▲ □ ★ > Executing task: platformio.exe run < [11/22/17 12:53:20] Processing megaatmega2560 (platform: atmelavr; lib_deps: TimerOne; board: megaatmega2560; framework: arduino)						

找到了TimerOne库并下载至.piolibdeps文件夹下

5.注意点

- 接上面我们也可以把下载好的TimerOne库直接放置在lib目录下,也就无需添加lib_deps。
- 我们不想在main里面直接使用TimerOne的pwm,我们想自己写一个motor库,motor库会使用到 TimerOne

motor.h

```
#ifndef TEST_MOTOR_H_
#define TEST_MOTOR_H_

class Motor{
    public:
        void init(unsigned char fre);
        void pwm(unsigned short);
};
#endif
```

```
#include "motor.h"
#include "TimerOne.h"

void Motor::init(unsigned char fre) {
    Timer1.initialize(1000.0/fre);
}
void Motor::pwm(unsigned short val) {
    Timer1.pwm(11,val);
}
```

main.cpp

```
#include <Arduino.h>
#include <motor.h>
Motor motor1;
void setup() {
    motor1.init(15);
}
void loop() {
    motor1.pwm(512);
}
```

• 编译完成,提示找不到TimerOne.h头文件



- ,可以看到Library Dependency Graph没有TimerOne
- 两种解决方法

1. main.cpp include头文件TimerOne.h,这个比较Iow,英文main中根本就没有使用到TimerOne

2. 之前的办法添加lib_deps = TimerOne

6.总结

至此可以看到,使用VSCode集成的PlatformIO IDE插件开发与查看Arduino的代码都非常方便