

使用 PlatformIO 烧录 BPI:bit

1. 首先打开 PlatformIO Home 页面，然后点击`Open Project`选择争取的路径打开工程，（可以选择测试代码）。如果使用自己的编写的代码烧录，那么 platformio.ini 文件应写入如下代码

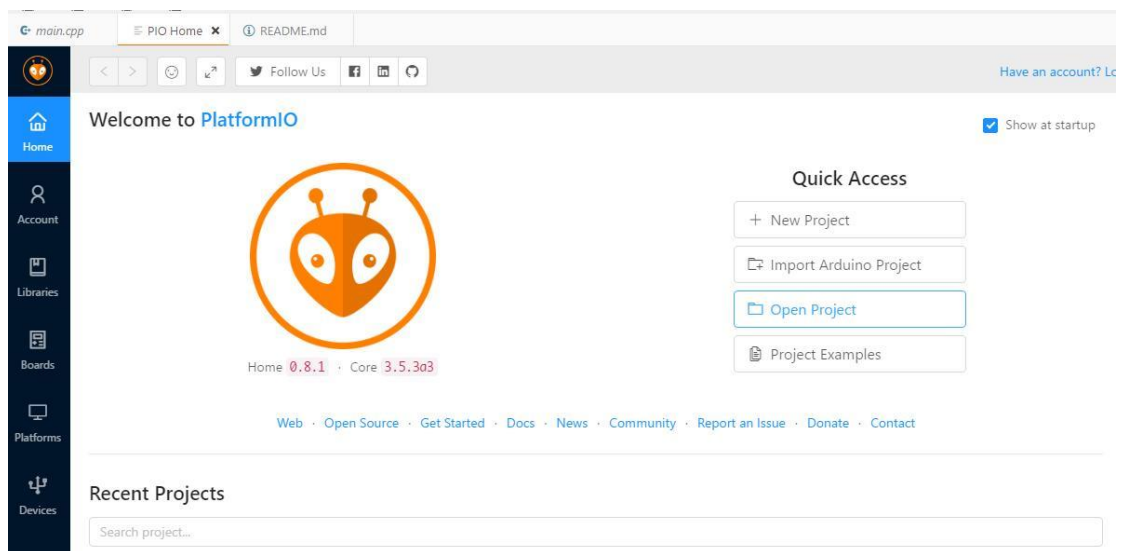
```
[env:BPI-BIT]
```

```
platform = espressif32
```

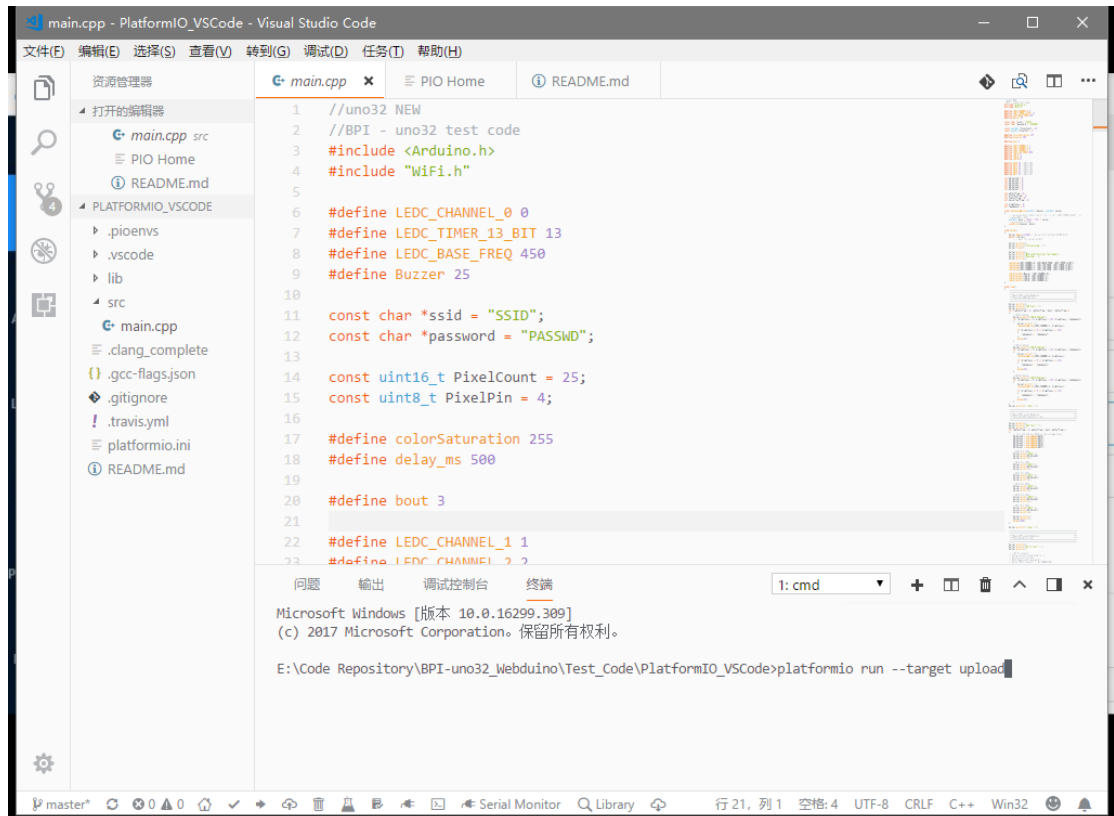
```
board = nodemcu-32s
```

```
framework = arduino
```

```
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:BPI-BIT]
12 platform = espressif32
13 board = nodemcu-32s
14 framework = arduino
```



- PlatformIO 工程下的代码路径一般为 src/main.cpp，代开代码然后使用 Ctrl+` 打开终端调试器，在里面输入 platformio run --target upload



The screenshot shows the Visual Studio Code interface with the PlatformIO_VSCODE project open. The file explorer on the left shows the project structure, including the src directory and the main.cpp file. The main editor displays the contents of main.cpp, which is a C++ program for an Arduino Uno32. The program includes headers for Arduino and WiFi, defines LED and buzzer pins, and sets up a serial connection. The terminal window at the bottom shows the command 'platformio run --target upload' being executed.

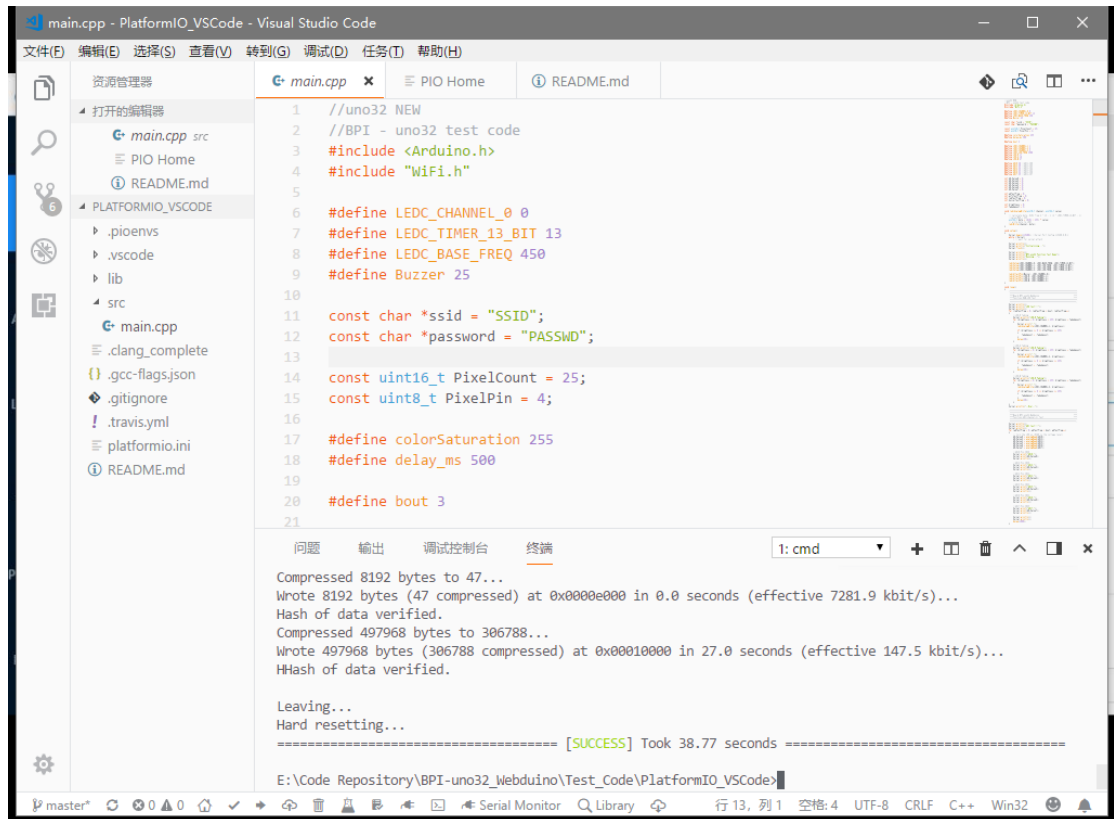
```
1 //uno32 NEW
2 //BPI - uno32 test code
3 #include <Arduino.h>
4 #include "Wifi.h"
5
6 #define LEDC_CHANNEL_0 0
7 #define LEDC_TIMER_13_BIT 13
8 #define LEDC_BASE_FREQ 450
9 #define Buzzer 25
10
11 const char *ssid = "SSID";
12 const char *password = "PASSWD";
13
14 const uint16_t PixelCount = 25;
15 const uint8_t PixelPin = 4;
16
17 #define colorSaturation 255
18 #define delay_ms 500
19
20 #define bout 3
21
22 #define LEDC_CHANNEL_1 1
23 #define LEDC_CHANNEL_2 2
```

问题 输出 调试控制台 终端 1: cmd

Microsoft Windows [版本 10.0.16299.309]
(c) 2017 Microsoft Corporation。保留所有权利。

E:\Code Repository\BPI-uno32_Webduino\Test_Code\PlatformIO_VSCODE>platformio run --target upload

3. 确保板子处于连接状态，程序会先编译，然后烧录进 BPI-uno32 板子，此时有几点需要注意，如果上面代码不指定端口的话，请尽量避免电脑上面有其他串口设备连接，指定串口号的话，请运行 `platformio --help` 查看。



```
1 //uno32 NEW
2 //BPI - uno32 test code
3 #include <Arduino.h>
4 #include "Wifi.h"
5
6 #define LEDC_CHANNEL_0 0
7 #define LEDC_TIMER_13_BIT 13
8 #define LEDC_BASE_FREQ 450
9 #define Buzzer 25
10
11 const char *ssid = "SSID";
12 const char *password = "PASSWD";
13
14 const uint16_t PixelCount = 25;
15 const uint8_t PixelPin = 4;
16
17 #define colorSaturation 255
18 #define delay_ms 500
19
20 #define bout 3
21
```

问题 输出 调试控制台 终端

1: cmd

Compressed 8192 bytes to 47...
Wrote 8192 bytes (47 compressed) at 0x0000e000 in 0.0 seconds (effective 7281.9 kbit/s)...
Hash of data verified.
Compressed 497968 bytes to 306788...
Wrote 497968 bytes (306788 compressed) at 0x00010000 in 27.0 seconds (effective 147.5 kbit/s)...
HHash of data verified.

Leaving...
Hard resetting...

===== [SUCCESS] Took 38.77 seconds =====

E:\Code Repository\BPI-uno32_Webduino\Test_Code\PlatformIO_VSCode>