

第一章 Windows 搭建 ESP32 开发环境 (vscode)

1. 乐鑫开发框架 esp-idf 介绍

ESP-IDF (Espressif IoT Development Framework) 是乐鑫 ESP32 开发的一套框架, 乐鑫所研发的软件开发环境 ESP-IDF 能够帮助用户快速开发物联网 (IoT) 应用, 满足用户对于 Wi-Fi、蓝牙、低功耗等性能的需求。框架里面包含很多例程, 是入门和进阶 ESP32 的优秀参考。

2. 乐鑫编译工具链介绍

开发一款芯片的应用程序需要有一套能够编译自己写的应用程序的工具链, 类似 STM32 单片机开发使用 keil, keil 内部已经集成包好了编译链, 不用设置即可使用, 写好应用程序, 点击编译按钮即可生成可下载到 STM32 单片机的文件。乐鑫的 ESP32 不同, 他没有类似 keil 这样的 IDE 工具, 但是官方提供了编译工具链, 用户可以使用任何编辑工具 (我们使用的是 Visual Studio Code), 写好应用程序, 使用官方提供的编译工具链进行交叉编译, 生成可以运行在 ESP32 上的文件。

3. Visual Studio Code 介绍

Visual Studio Code 是微软公司开发的一个编辑工具, 有以下特点:

- 轻量级

轻量级的意思是指开发工具占用的空间要小, 以及使用其强大的代码提示功能。

- 跨平台

Windows/Mac/Linux 全支持。

4. 搭建环境步骤

4.1. 如何下载

- 代码编辑工具 Visual Studio Code

Visual Studio Code 大家可以在微软的官网上根据自身的开发平台下载, 下载地址如下所示: [Visual Studio Code 下载地址](#)

- 开发框架 esp-idf

乐鑫的 esp-idf 包均放在 Github 上 (尽可能地选择 release 版本, 不要选择尝鲜版本), 具体下载地址如下所示: [esp32 esp-idf 开发包下载地址](#)

- 编译工具链

Visual Studio Code 毕竟只是一个轻量级的代码编辑工具, 本身是不自带编译工具链的, 而且乐鑫 ESP32 芯片使用的又不是主流的 ARM 内核, 所以还是要使用乐鑫官方指供的工具链才行, 具体的下载如下所示: [ESP32 编译工具链下载地址](#)

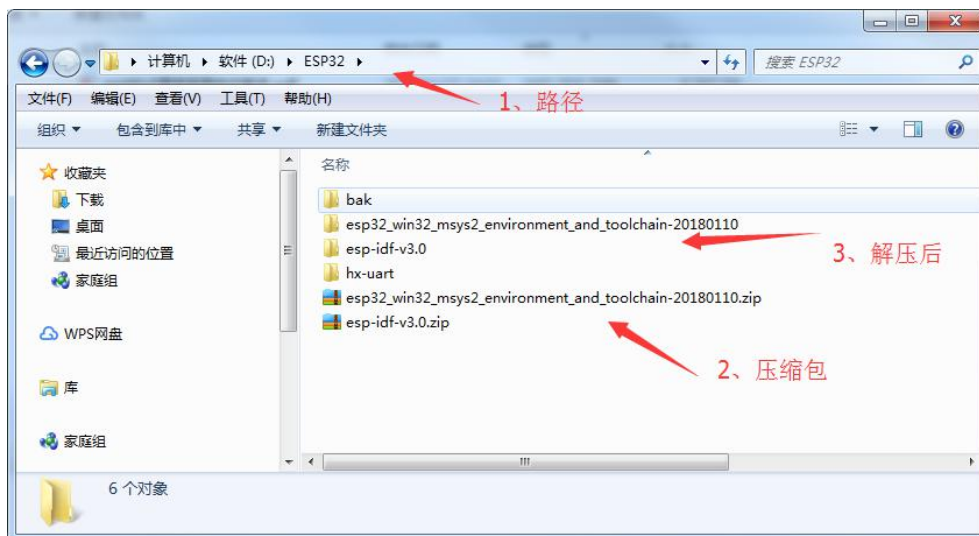
4.2. 如何安装使用

下载完 Visual Studio Code 和 ESP32 工具链, 那么如何搭建起来开发 ESP32 呢? 其实很简单, ESP32 工具链其实已经完成了大部分的设置了, 只需要解压就可以直接使用, 具体的使用步骤如下所示:

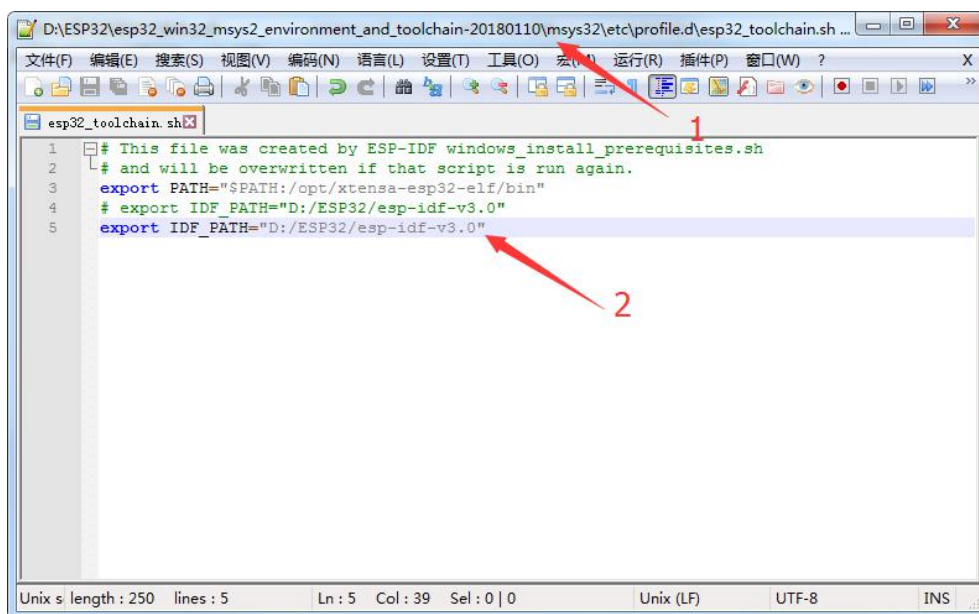
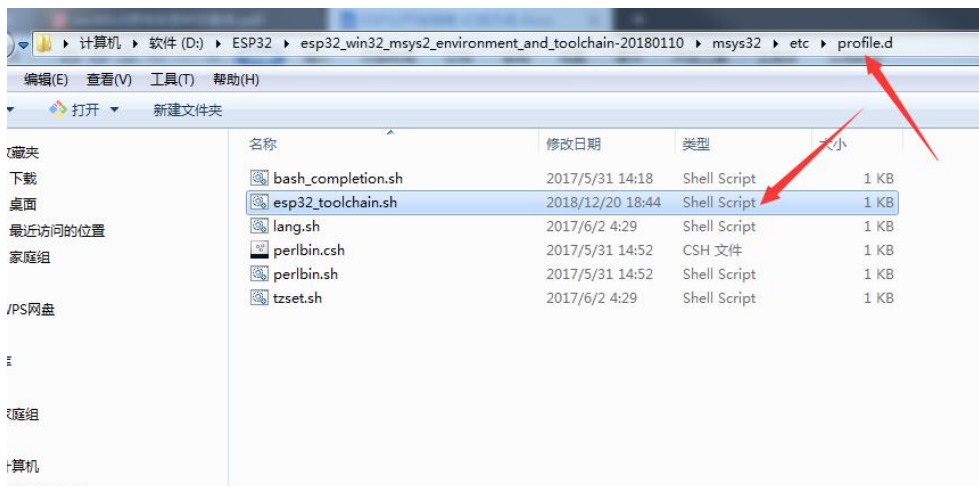
- 安装 Visual Studio Code 全程 Next 到底。

- 在自己选择适当的路径下解压 ESP32 编译工具链 (不要有中文路径, 路径尽可能地短)。

- 在自己选择适当的路径下解压 esp-idf 开发包 (不要有中文路径, 路径尽可能地短)。



- 设置 esp-idf 开发包的路径: 在 esp32_toolchain.sh 增加 esp-idf 开发包的根目录路径 (export IDF_PATH="D:/ESP32/esp-idf-v3.0") , 路径因不同解压路径不同而不同。 .sh 文件, 右击用记事本打开。

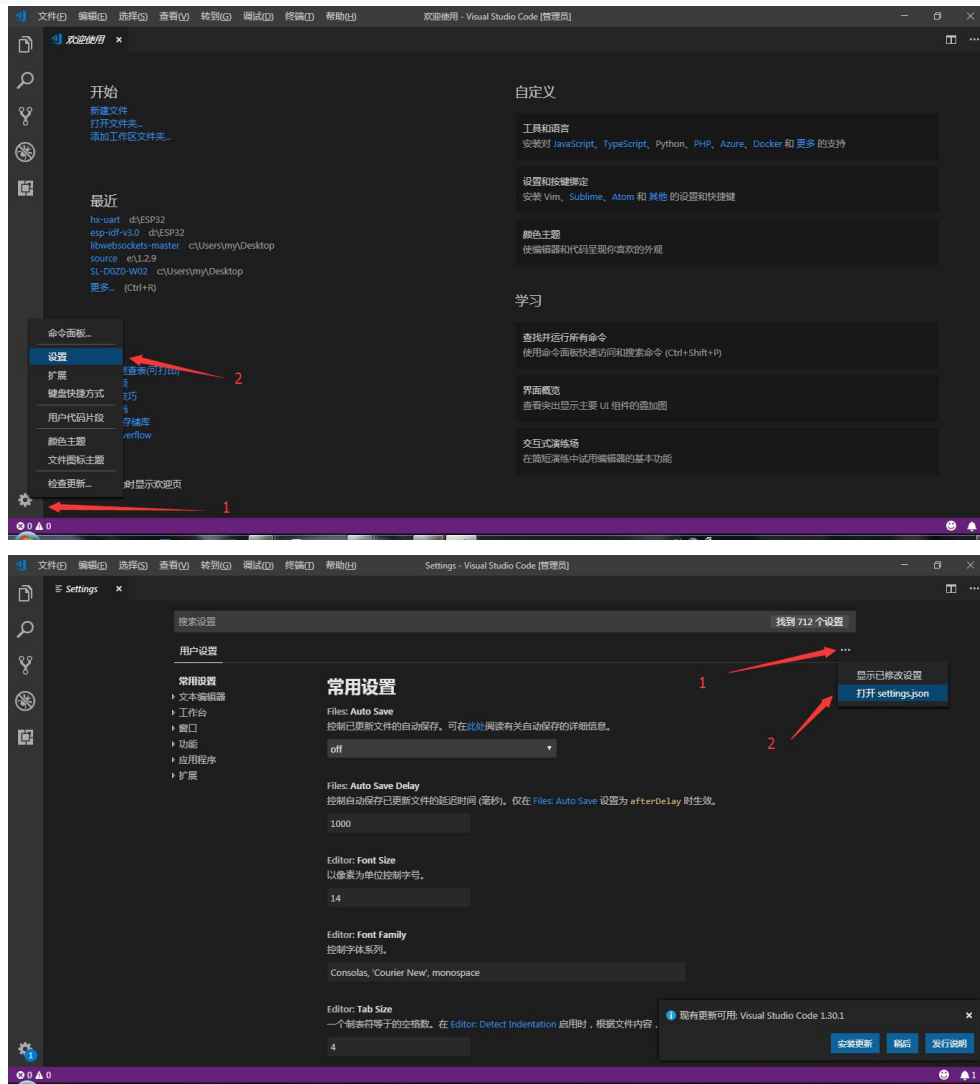


- 打开 Visual Studio Code 并将编译工具链的 shell 嵌入至 Visual Studio Code 的

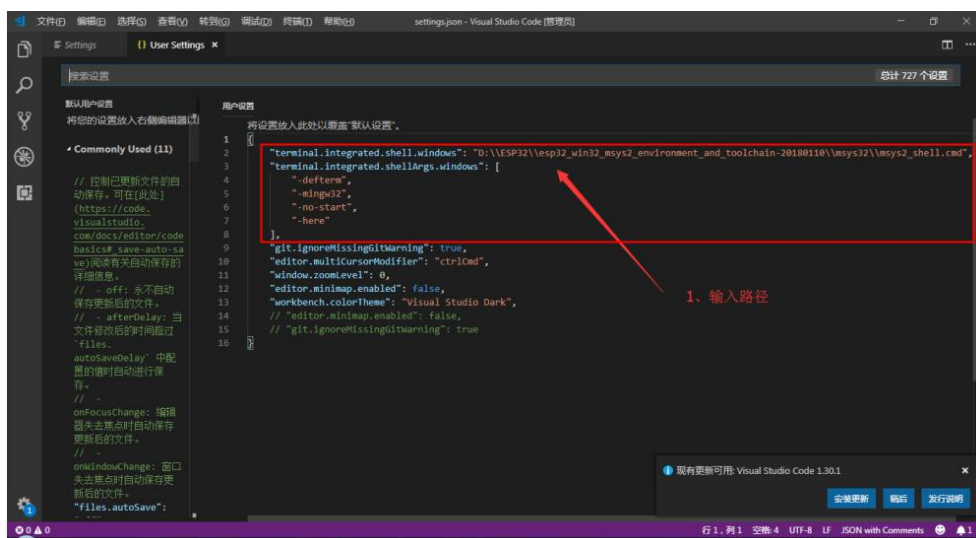
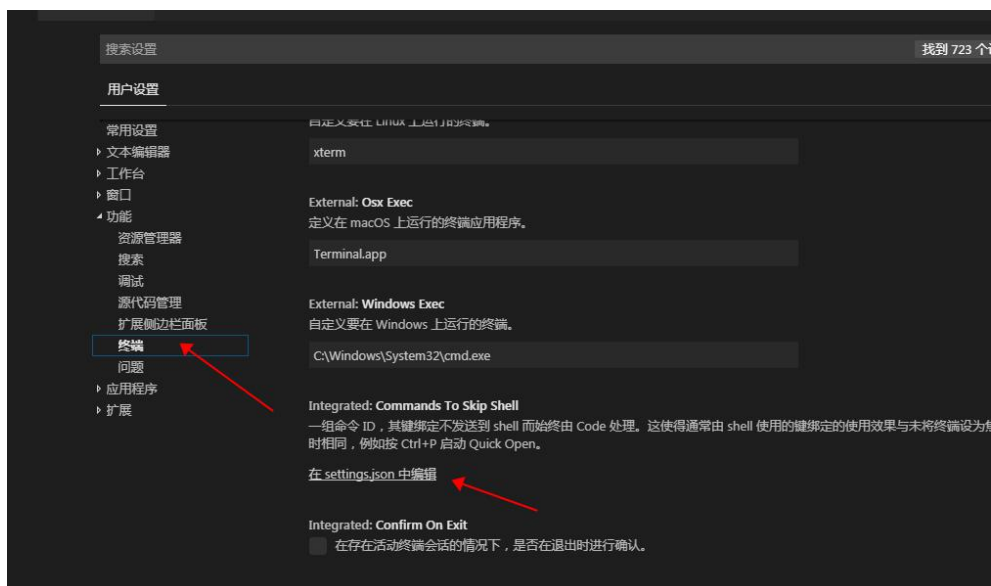
集成终端，设置的参数 (其中 **shell** 的路径因不同解压路径不同而不同), 如下所示:

```
"terminal.integrated.shell.windows":
"D:\\Project\\ESP32\\toolchain\\esp32_win32_msys2_environment_and_toolchain-2
0180110\\msys32\\msys2_shell.cmd",

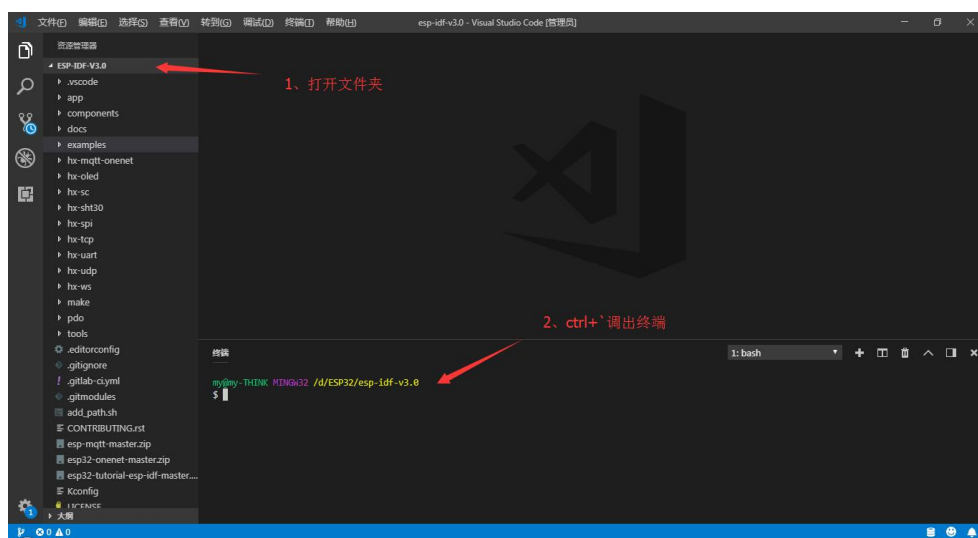
"terminal.integrated.shellArgs.windows": ["-defterm", "-mingw32",
"-no-start", "-here"]
```



上面方法找不到可以这样

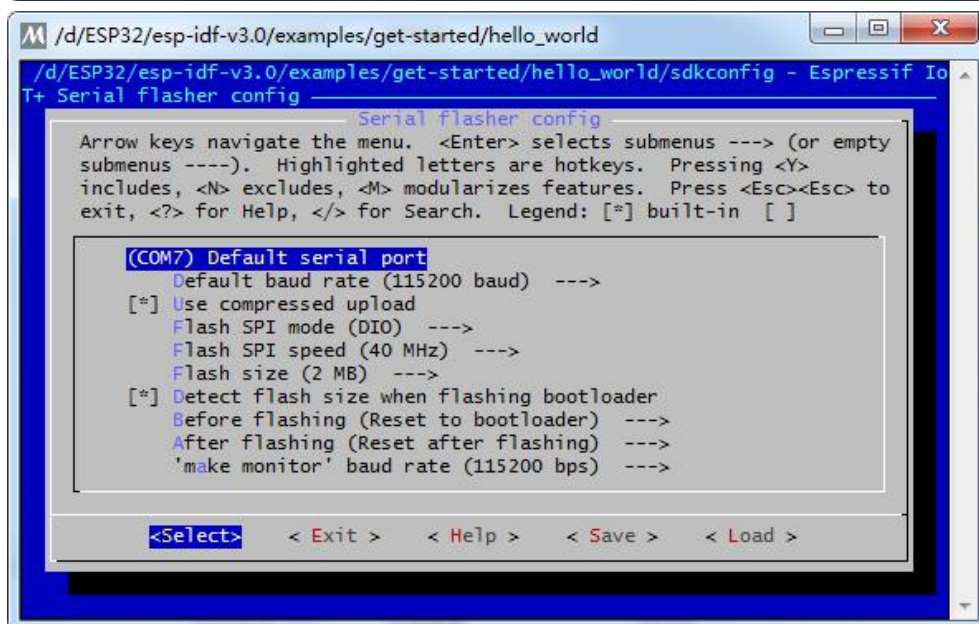
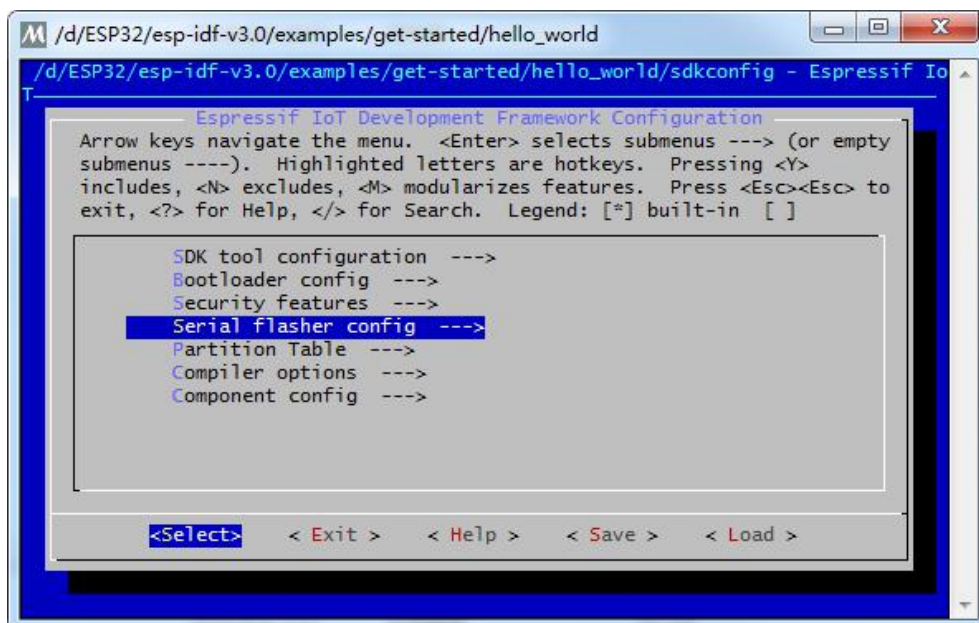
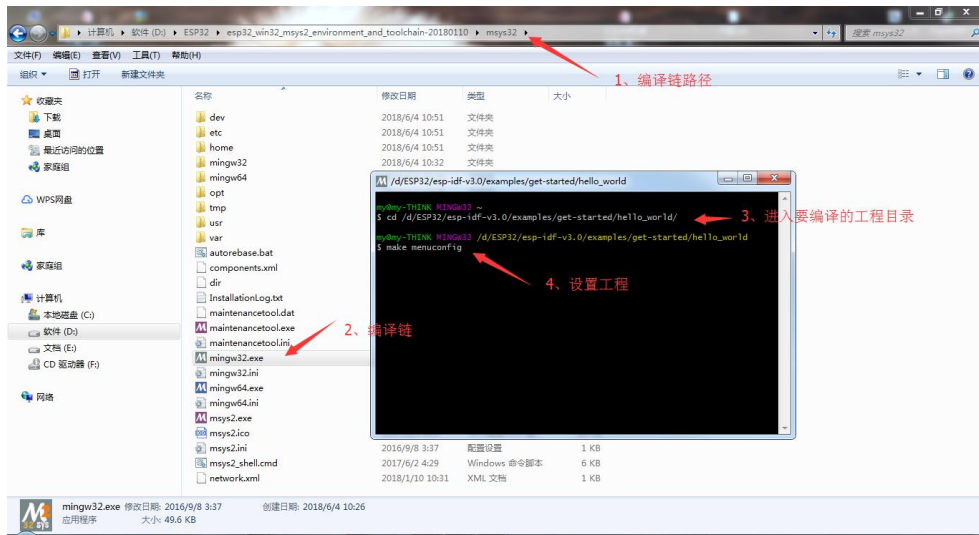


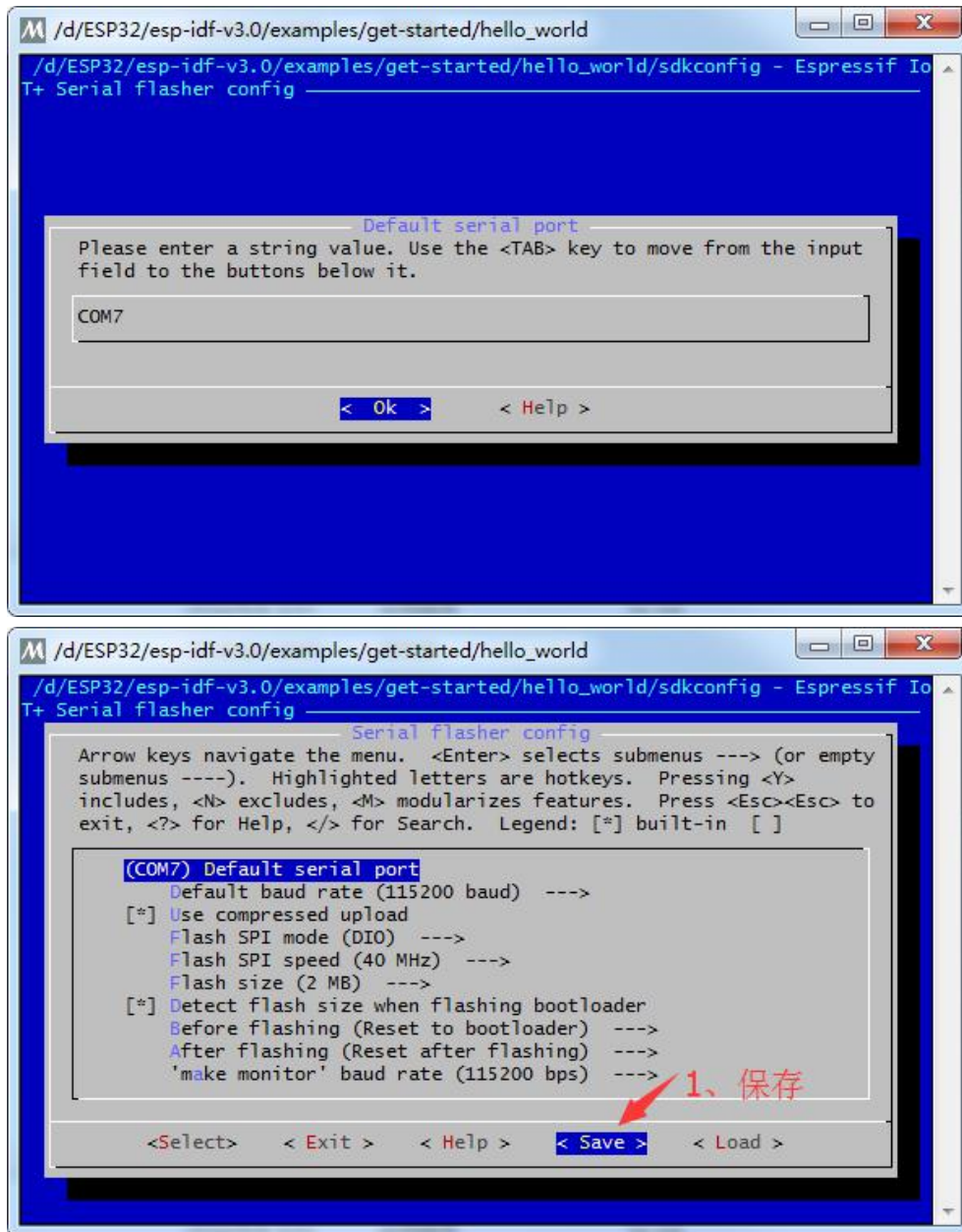
- 使用 Visual Studio Code 打开 esp-idf 开发包中的工程示例文件, 打开之后, 然后同时按下 Ctrl+` 就可以打开嵌入了 shell 的终端, 此时就可以开发您的第一个 esp32 应用了。



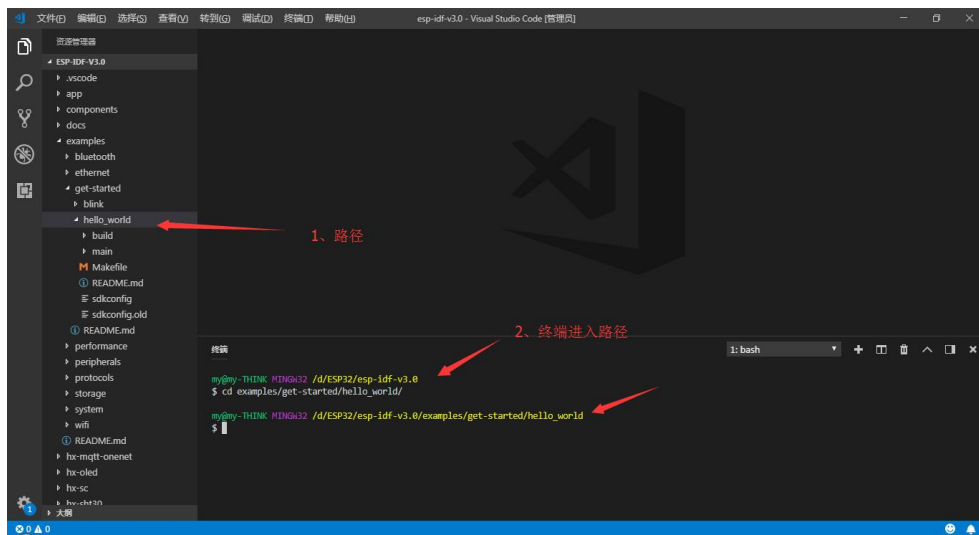
- 接上开发板, 使用官方编译链, make menuconfig 命令设置工程, 需要设置串口下载的串口号 (不同人串口号不同)。也可以在 Visual Studio Code 中 make menuconfig

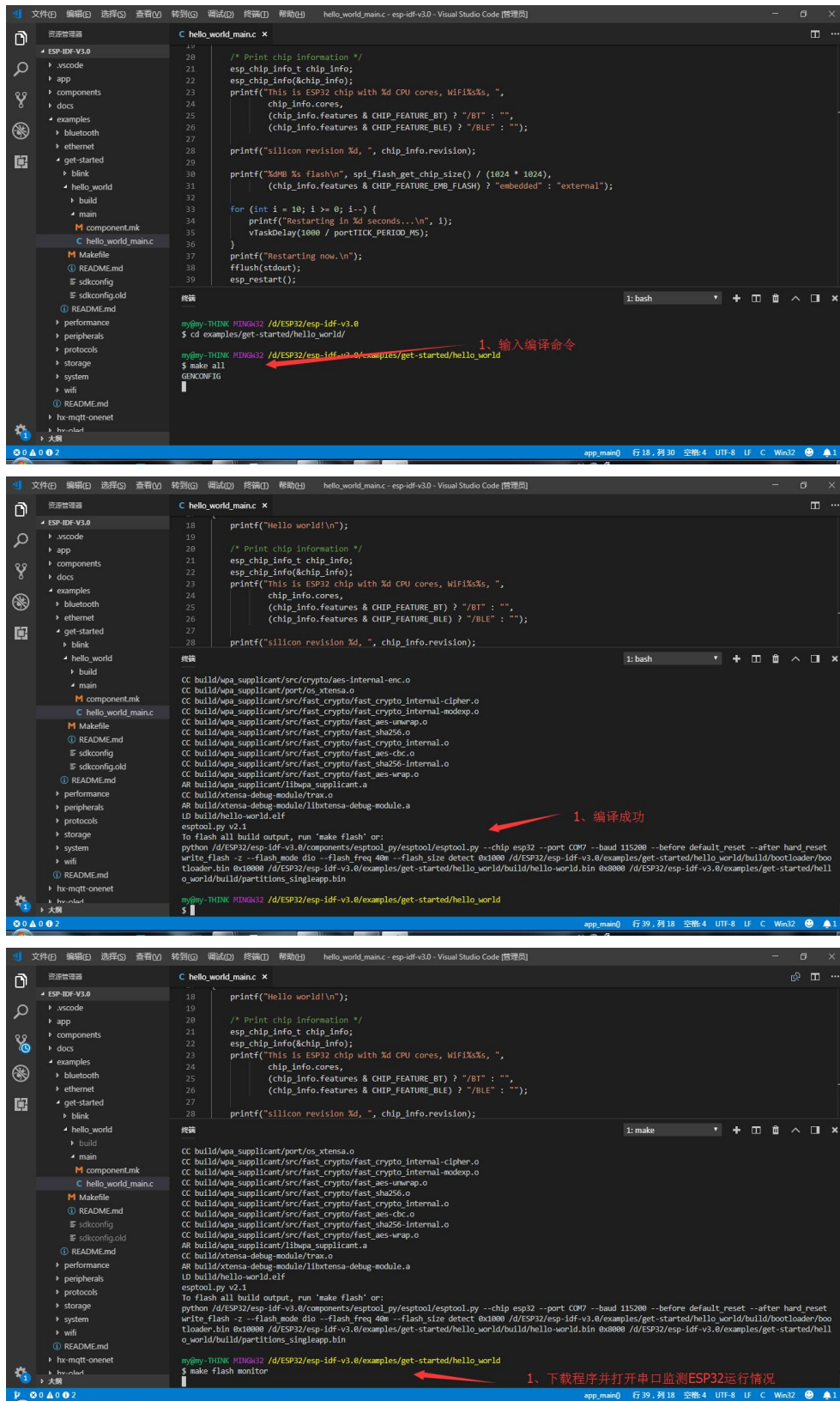
设置, 不过 Visual Studio Code 中显示乱码, 有解决办法。

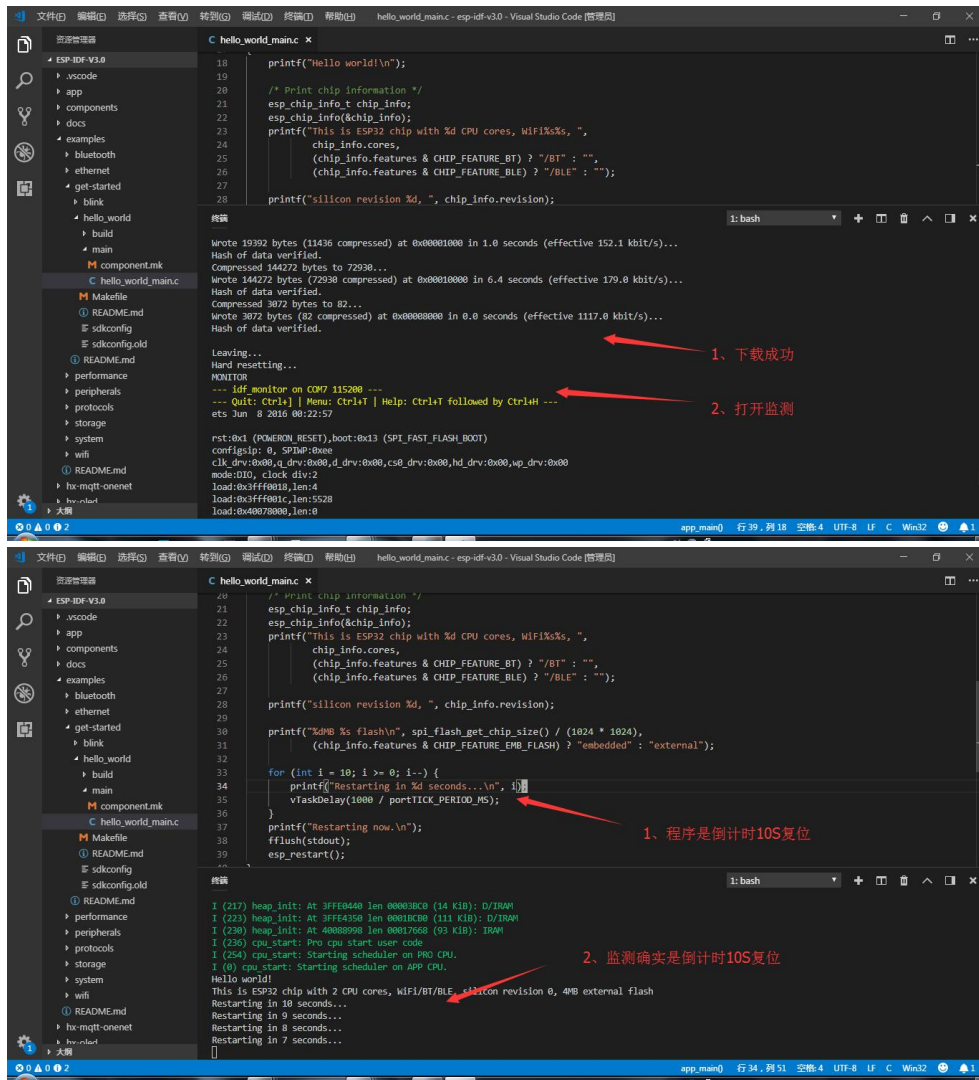




➤ 打开、编译官方 hello_world 工程







- 经过上面几个步骤，完成 ESP32 环境搭建工作，美中不足是 VScode 的 make menuconfig 乱码和 Windows 编译速度慢，有解决办法。

4.3. 更多 make 指令

终端中输入 make help，查看 ESP32 编译链中的所有 make 指令。



4.4. 搭建环境总结

- Windows 缺点是编译 ESP32 工程速度慢 (>2 分钟), 优点是熟悉, 软件多。
- Linux 和 MAC 缺点就是开发不习惯, 优点是速度快 5-10 倍, 自己考量。
- 我选择 Windows, 因为编译的时候我要看份报纸喝杯茶, 软件多, 方便些教程。
- 另外附上[优化 Windows 下编译 ESP32 的速度](#)的教程, 可以折腾下。