一.程序编译

这个部分我们来学习下整个ESP32在VSCODE中是怎样编译的！

我们先不学习创建工程，先使用esp-idf example中提供的最简单的一个工程hello world(路径在esp-idf\examples\get-started\hello\_world)

步骤一: 使用VSCODE打开菜单: 文件->打开文件夹(选择你的idf目录,选择C:\Users\86139\Documents\esp32\esp-idf\examples\get-started\hello\_world),打开后就是这个样子

步骤二: 点击底部红色编译按钮，等待编译完成

漫长等待，终于编译好了，如图，会统计出来各个字段的信息

电脑萤幕的截图

描述已自动生成

二.程序烧录

我们的板子是使用UART直接烧录程序，所以不需要额外的接下载器

步骤一：选择要烧录的UART COM口

备注：此时COM口不能被占用

步骤二：选择开发板为ESP32

步骤三：在VSCODE的底部点击烧录按钮烧录

文本

描述已自动生成

在这里你可能会提出一个问题，那我也大概猜到大家有什么问题，就是如果我在产线中，或者想给别人验证一下程序，却不能给源码，那么我要怎么烧录？

我们使用的是这个工具flash\_download\_tool\_

步骤一：我们打开这个工具，勾选如下：

然后打开工程的build路径，勾选如下，然后点击start

C:\Users\86139\Documents\esp32\esp-idf\examples\get-started\hello\_world\build\hello\_world.bin

C:\Users\86139\Documents\esp32\esp-idf\examples\get-started\hello\_world\build\bootloader\bootloader.bin

C:\Users\86139\Documents\esp32\esp-idf\examples\get-started\hello\_world\build\partition\_table\partition-table.bin

完成之后点击stop

图形用户界面, 文本

描述已自动生成

备注：并不是所有工程这么配置，这个要根据不同的项目不同的地址以及配置，所以要询问给你这写bin的人，那这个工程的信息我是从哪看到的人？是VSCODE中下载的时候会提示如下信息：

> Executing task: C:\Users\12475\.espressif\python\_env\idf4.3\_py3.8\_env\Scripts\python.exe C:\Users\12475\esp\esp-idf\components\esptool\_py\esptool\esptool.py -p COM5 -b 460800 --before default\_reset --after hard\_reset --chip esp32 write\_flash --flash\_mode dio --flash\_freq 40m --flash\_size detect

0x10000 hello-world.bin

0x1000 bootloader/bootloader.bin

0x8000 partition\_table/partition-table.bin

三.程序运行

我们打开板子的串口，会看到输出以下日志！

图形用户界面, 文本, 应用程序

描述已自动生成

注意：监视器如果出现乱码，修改波特率频率即可。

通过Arduino烧录

图形用户界面

中度可信度描述已自动生成