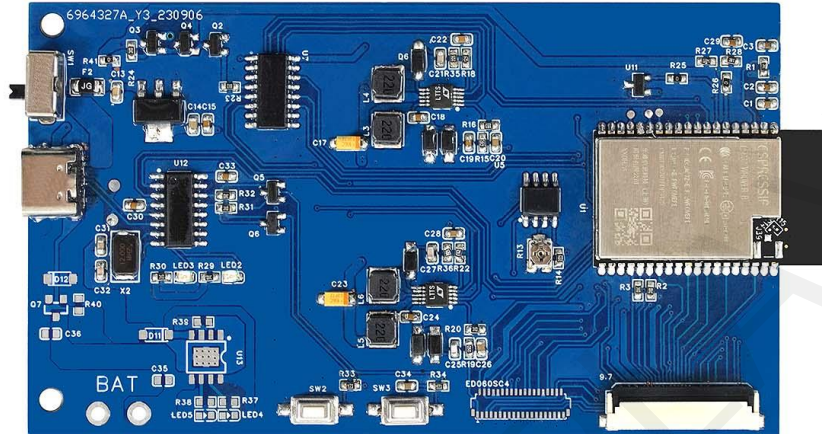




epdiy

并口电子纸显示屏驱动板说明书

产品规格



品类	标准品
描述	电子纸显示屏驱动板
品名	epdiy
日期	2023/10/18
版本	1.0

	设计团队		
	批准	校验	编写
			

大连市甘井子区中华西路 18 号中南大厦 A 座 1513

电话:0411-84619565

邮箱: sales@good-display.com

网址: www.good-display.cn

目录

一、概述	4
二、开发板的主要参数	4
三、主要功能模块	5
四、程序下载	7

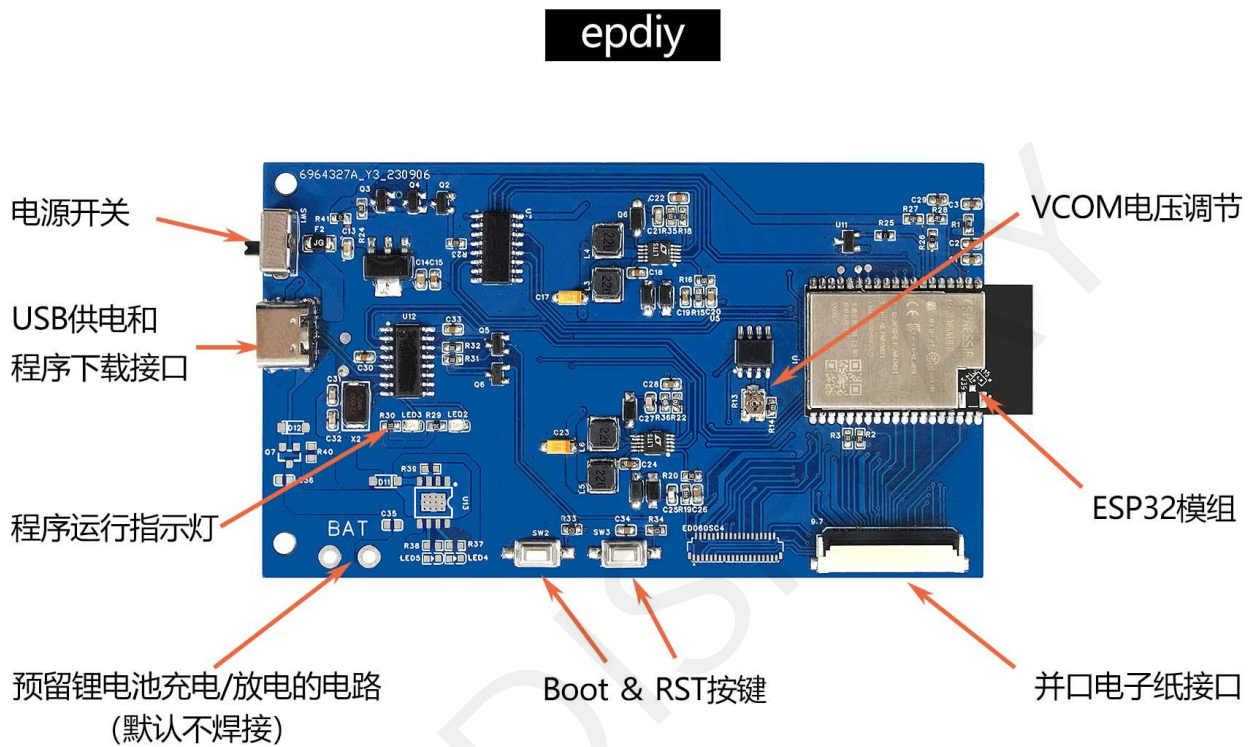
1. 概述

epdiy 开发套件可以辅助开发者更快更顺利地开发电子纸显示屏项目，专为并口电子纸显示屏而设计，支持16灰阶显示，支持ESP-IDF 和 Arduino 的驱动程序，可驱动包括6寸、9.7寸、13.3寸等多个尺寸的并口电子纸显示屏。较其他的并口驱动方案来说，EPDi y具有价格亲民、易于上手等优势，用户可根据项目的不同需求进行二次开发，是您驱动并口电子纸显示屏的一个很好的选择。

2. 开发板主要参数

参数	产品规格
型号	epdiy
使用平台	ESP-IDF
开发板外形尺寸	99.8mm x 57.8mm
电源	Type-C 接口供电
示例程序	可提供
工作温度	-20 ℃ ~ 70 ℃
主要功能	学习如何驱动电子纸显示屏； 测试和评估电子纸显示屏； 在此板的基础上进行二次开发。
辅助功能	USB转串口、指示灯、按键、电流检测等

3. 主要功能



3.1、字体、图像、波形

ESP32虽然功能相当强大，但仍然是一款微控制器。因此，由于内存和计算资源有限，执行尽可能多的处理是有用的。

epdiy附带脚本，可将字体、图像和波形转换为C标题，然后，您可以简单地#include您的项目中。

3.2、生成字体文件

字体只能由驱动程序以特殊的标头格式使用，需要从 TTF 字体生成。如果生成的具有默认字符的字体文件对于应用程序来说太大，您可以在intervals fontconvert.py中进行修改。

3.3、生成图像

转换图像的过程与转换字体非常相似。使用输入图像、图像名称和输出图像运行脚本。图像将转换为缩小的灰度，以匹配适合（默认为 1200x825）。为了获得准确的灰度，建议在转换图像之前使用专用工具对图像进行颜色分级和缩放。

3.4、转换波形

在商业应用中，显示器由所谓的波形文件中的信息驱动。这些指定，应用于像素的脉冲从一种灰色调过渡到另一种灰色调。

3.5、深度睡眠电流

在进入深度睡眠之前。这将取消初始化用于驱动 display 的 I2S 外设，并使 epdiy 使用的引脚进入低功耗状态。您应该能够实现小于 13 μ A 的深度睡眠电流。如果您的深度睡眠电流要高得多，请检查您连接的外围设备。对于某些模块，您必须在进入深度睡眠之前隔离 GPIO 12。

3.6、VCOM值的调节

通过调节板子上的R13，来调节VCOM值，VCOM值一定要严格按照屏幕上的值调节

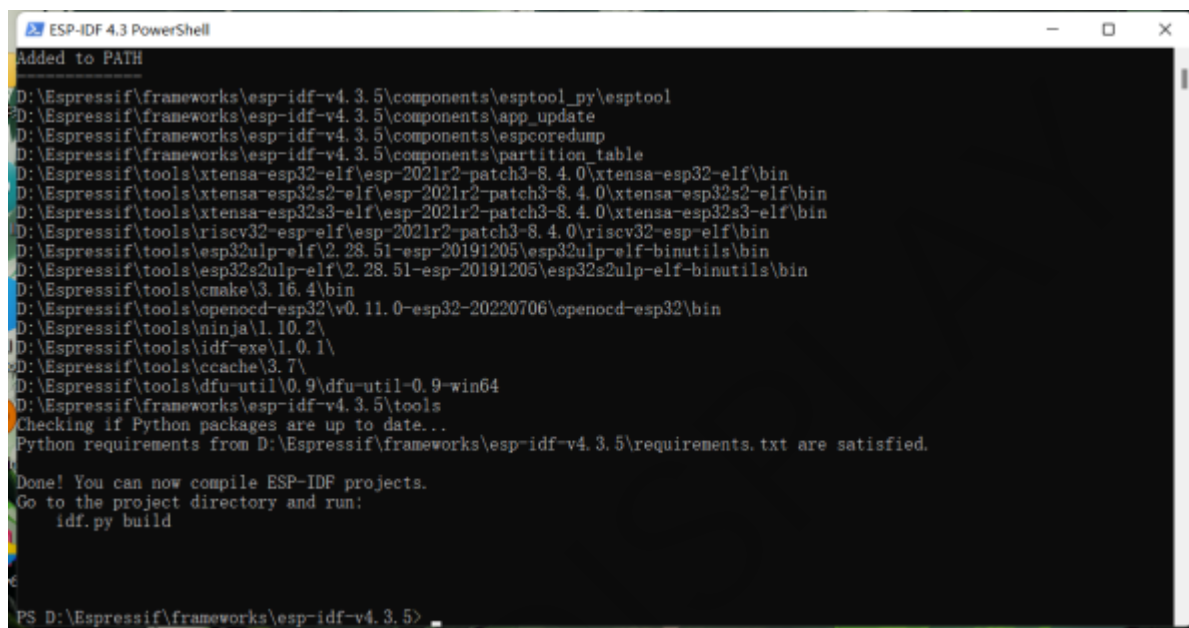


4. 程序下载

程序来源GitHub vrol and/epdiy <https://github.com/vrol and/epdiy>

4.1、下载软件（ESP-IDF 4.3 PowerShell）（乐鑫官网提供）

以下操作按epdiy-master123\examples\dragon为例



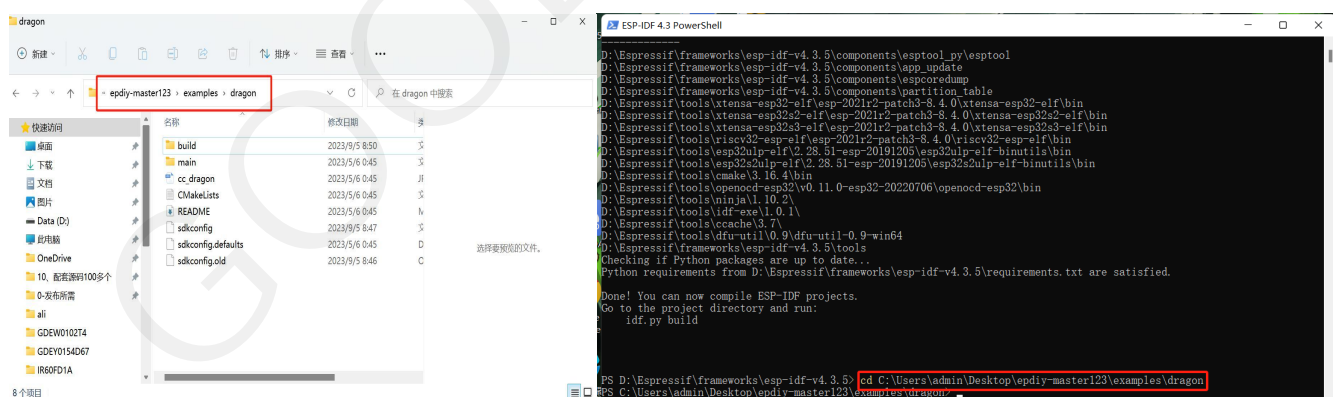
```

ESP-IDF 4.3 PowerShell
Added to PATH
D:\Espressif\frameworks\esp-idf-v4.3.5\components\esptool_py\esptool
D:\Espressif\frameworks\esp-idf-v4.3.5\components\app_update
D:\Espressif\frameworks\esp-idf-v4.3.5\components\espcoredump
D:\Espressif\frameworks\esp-idf-v4.3.5\components\partition_table
D:\Espressif\tools\xtensa-esp32-elf\esp-2021r2-patch3-8.4.0\xtensa-esp32-elf\bin
D:\Espressif\tools\xtensa-esp32s2-elf\esp-2021r2-patch3-8.4.0\xtensa-esp32s2-elf\bin
D:\Espressif\tools\xtensa-esp32s3-elf\esp-2021r2-patch3-8.4.0\xtensa-esp32s3-elf\bin
D:\Espressif\tools\riscv32-esp-elf\esp-2021r2-patch3-8.4.0\riscv32-esp-elf\bin
D:\Espressif\tools\esp32ulp-elf\2.28.51-esp-20191205\esp32ulp-elf-binutils\bin
D:\Espressif\tools\esp32s2ulp-elf\2.28.51-esp-20191205\esp32s2ulp-elf-binutils\bin
D:\Espressif\tools\cmake\3.16.4\bin
D:\Espressif\tools\openocd-esp32\v0.11.0-esp32-20220706\openocd-esp32\bin
D:\Espressif\tools\ninja\1.10.2\
D:\Espressif\tools\idf-exe\1.0.1\
D:\Espressif\tools\ccache\3.7\
D:\Espressif\tools\dfu-util\0.9\dfu-util-0.9-win64
D:\Espressif\frameworks\esp-idf-v4.3.5\tools
Checking if Python packages are up to date...
Python requirements from D:\Espressif\frameworks\esp-idf-v4.3.5\requirements.txt are satisfied.

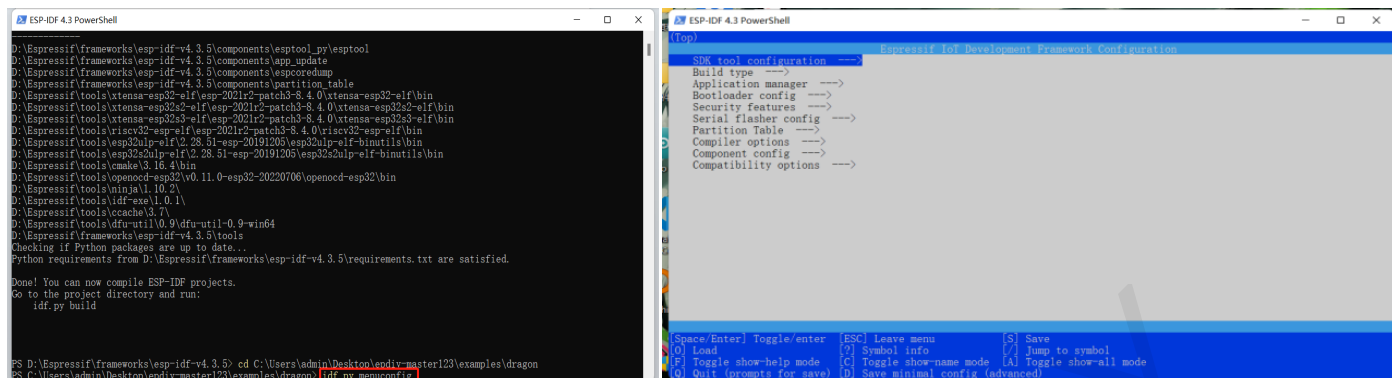
Done! You can now compile ESP-IDF projects.
Go to the project directory and run:
idf.py build

PS D:\Espressif\frameworks\esp-idf-v4.3.5>
    
```

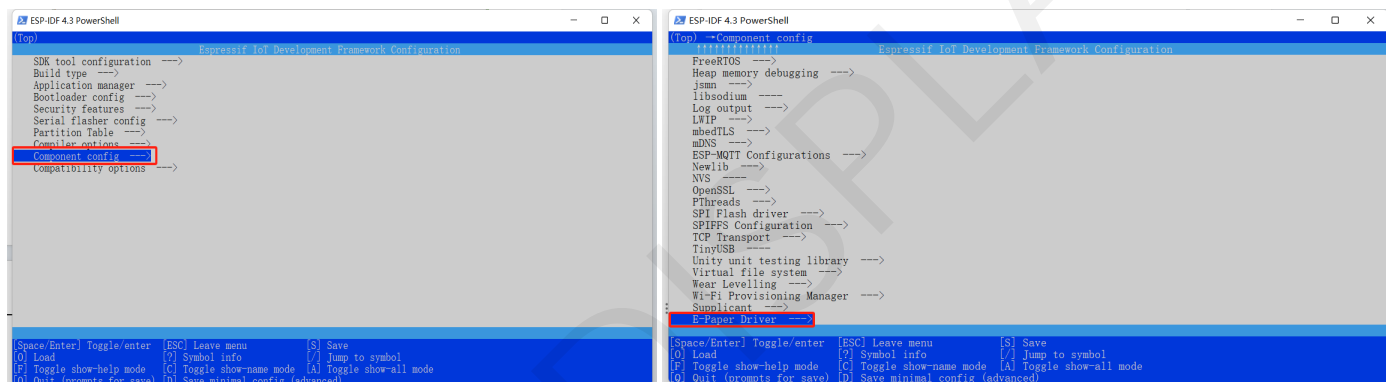
4.2、首先输入 cd 程序地址 按回车，使编译器跳转到该程序地址



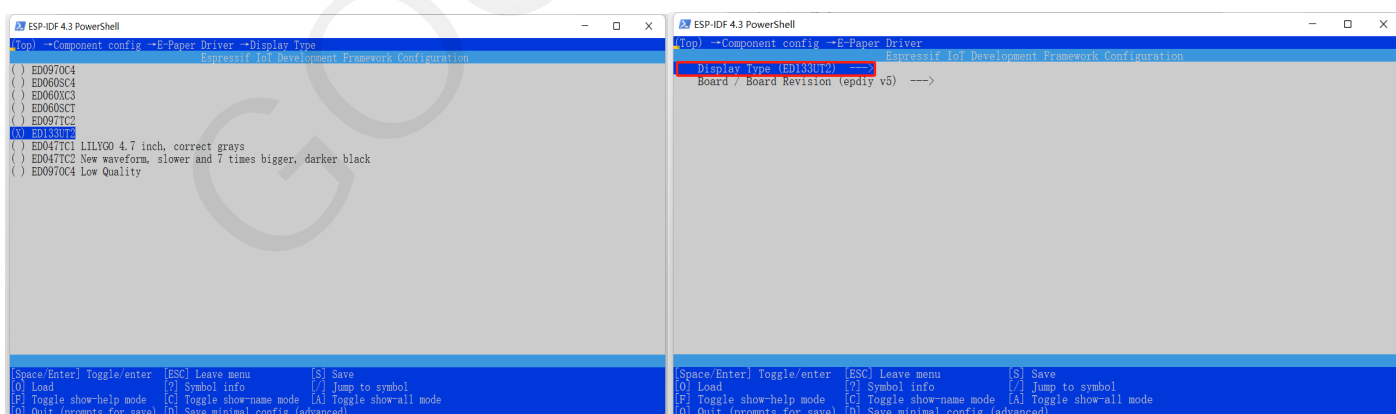
4.3、设置屏幕型号，输入idf.py menuconfig 并按回车 跳转到程序系统中



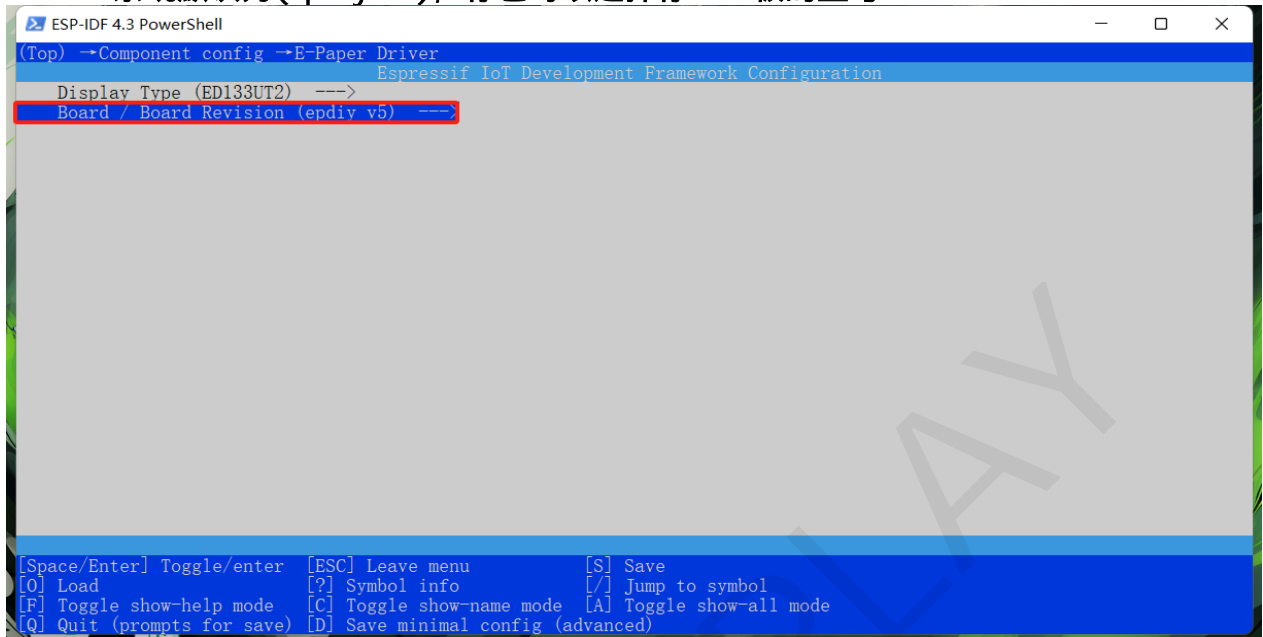
4.3.1 选定Componentconfig，找到E-Paperdriver



4.3.2 通过DisplayType，选择你需要的屏幕型号

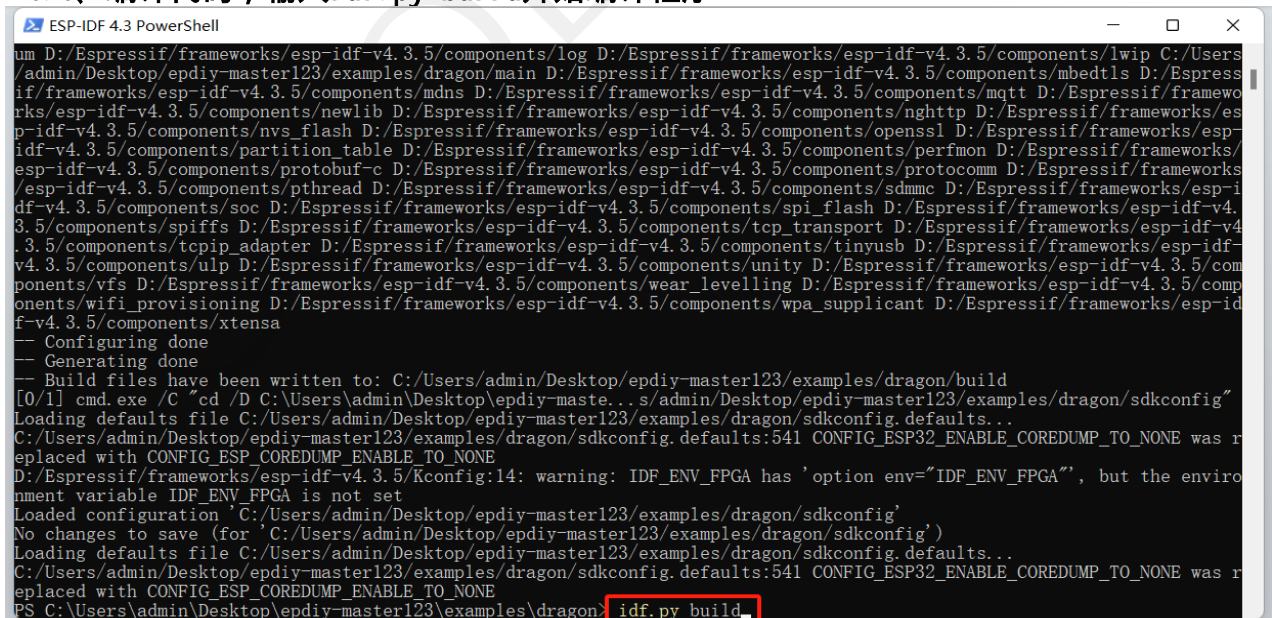


4.3.3 系统默认为(epdiy V5)，你也可以选择你demo板的型号



4.3.4 保存配置后，按快捷键Q退出系统

4.4、编译代码，输入idf.py build开始编译程序



4.5、程序烧录，输入idf.py flash monitor开始烧录程序

```
ESP-IDF 4.3 PowerShell
n/Desktop/epdiy-master123/examples/dragon/build/ldgen Libraries --objdump D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe"
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: cpu_util.
c.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: regi2c_ct
rl.c.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: rtc_clk.c
.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: rtc_clk_i
nit.c.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: rtc_init.
c.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: rtc_pm.c
.obj: file format not recognized
D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/bin/xtensa-esp32-elf-objdump.exe: rtc_sleep
.c.obj: file format not recognized
Traceback (most recent call last):
  File "D:\Espressif\frameworks\esp-idf-v4.3.5\tools\ldgen\ldgen.py", line 178, in <module>
    main()
  File "D:\Espressif\frameworks\esp-idf-v4.3.5\tools\ldgen\ldgen.py", line 134, in main
    dump = StringIO(subprocess.check_output([objdump, '-h', library]).decode())

  File "subprocess.py", line 466, in check_output
  File "subprocess.py", line 571, in run
subprocess.CalledProcessError: Command '['D:/Espressif/tools/xtensa-esp32-elf/esp-2021r2-patch3-8.4.0/xtensa-esp32-elf/b
in/xtensa-esp32-elf-objdump.exe', '-h', 'C:/Users/admin/Desktop/epdiy-master123/examples/dragon/build/esp-idf/esp_hw_sup
port/libesp_hw_support.a']' returned non-zero exit status 1.
ninja: build stopped: subcommand failed.
ninja failed with exit code 1
PS C:\Users\admin\Desktop\epdiy-master123\examples\dragon> idf.py flash monitor
```

4.6、效果展示

