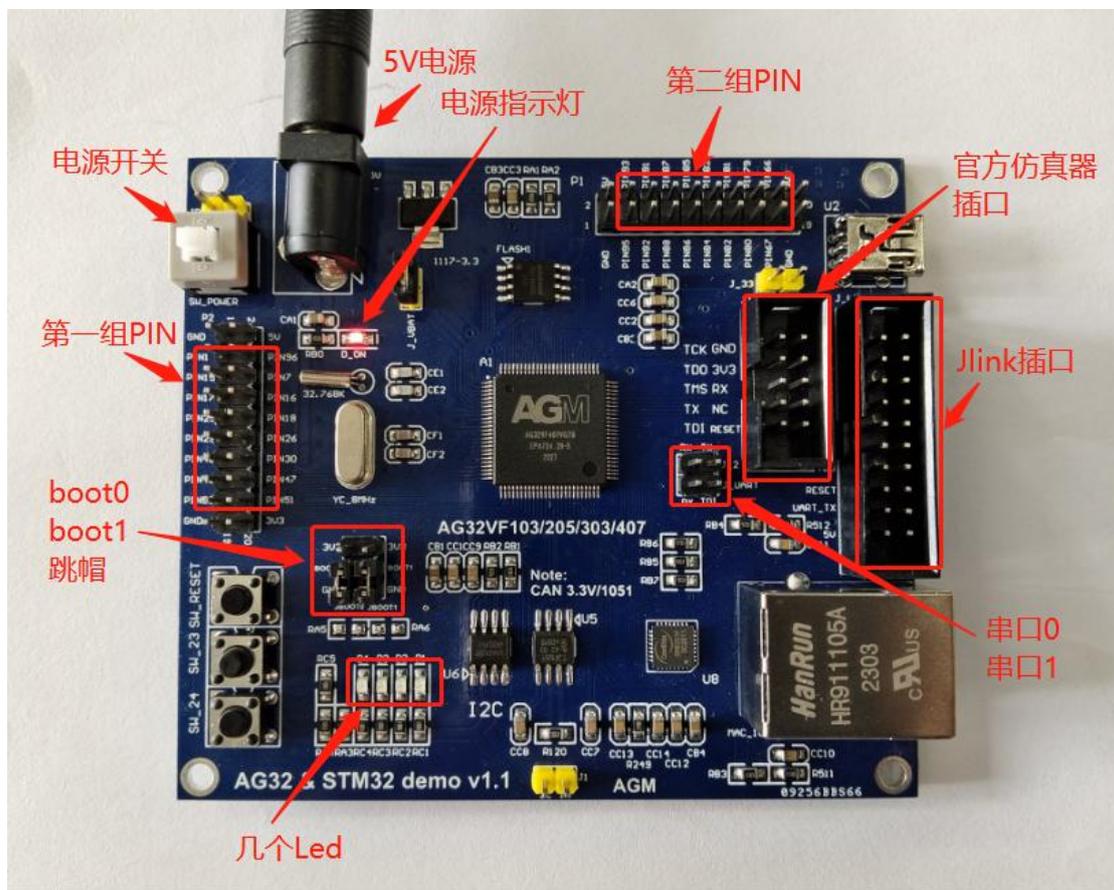


# AG32 开发板使用入门



在使用开发板前，请确认已经安装好开发环境。  
安装环境过程，请参考文档《AG32 开发环境搭建.pdf》

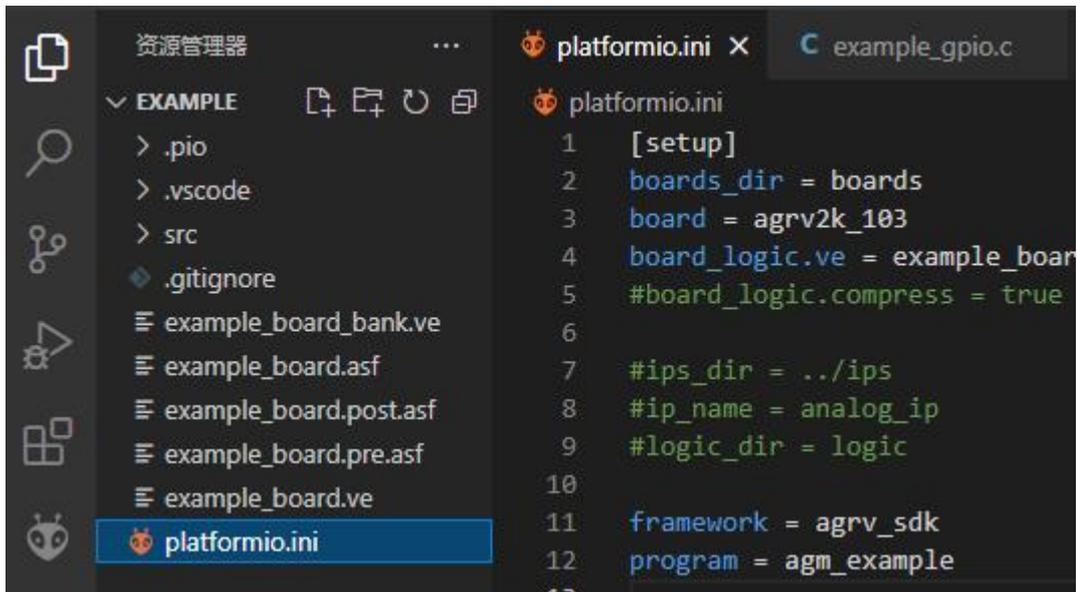
### 一、上电：

给开发板 5V 供电，打开开关，可以看到电源旁边的小红灯亮起。

### 二、使用 example 例程：

打开 example 例程，在 SDK 路径下：

D:\xxxxx\AgRV\_pio\platforms\AgRV\examples\example (注意这里的两重 example)



由于开发板使用的是 407 芯片，需要先修改 platformio.ini 中的 board 类型：

**board = agrv2k\_103**

修改为：

**board = agrv2k\_407**

为了验证简单化，可以先把 example\_board.ve 中的配置暂时删除，只留下前两项：

**SYSCLK 100**

**HSECLK 8**

### 三、烧录 VE 文件和代码 bin：

烧录程序可选 jlink 和官方烧录器两种；（串口烧录这里不讨论）

#### 如果使用 jlink：

连线：用 jlink 线连接开发板的 jlink 接口。

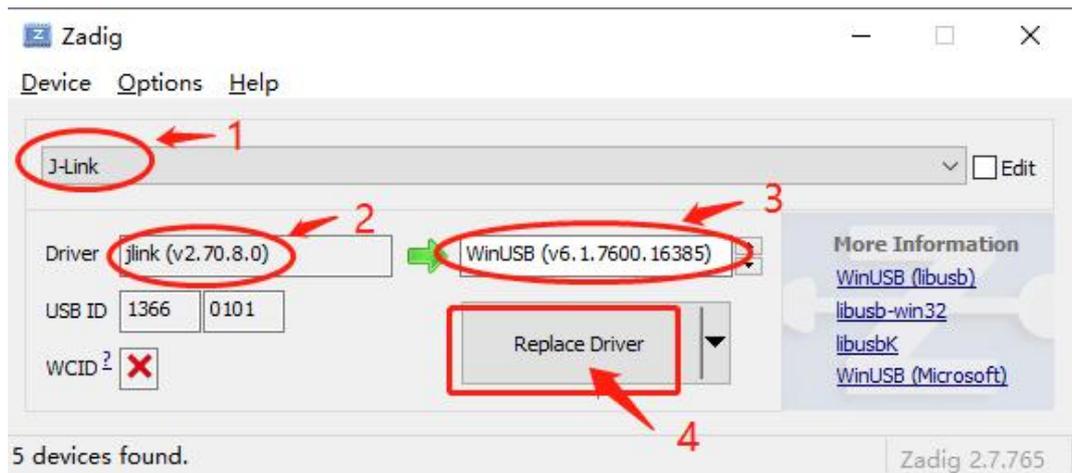
配置：确认 platformio.ini 中的以下两行配置是打开的：

**debug\_tool = jlink-openocd**

**upload\_protocol = jlink-openocd**

安装插件：第一次使用 jlink，需要先安装插件【zadig-2.8.exe】，安装参考下图：

（该插件在 sdk 路径的根目录下）

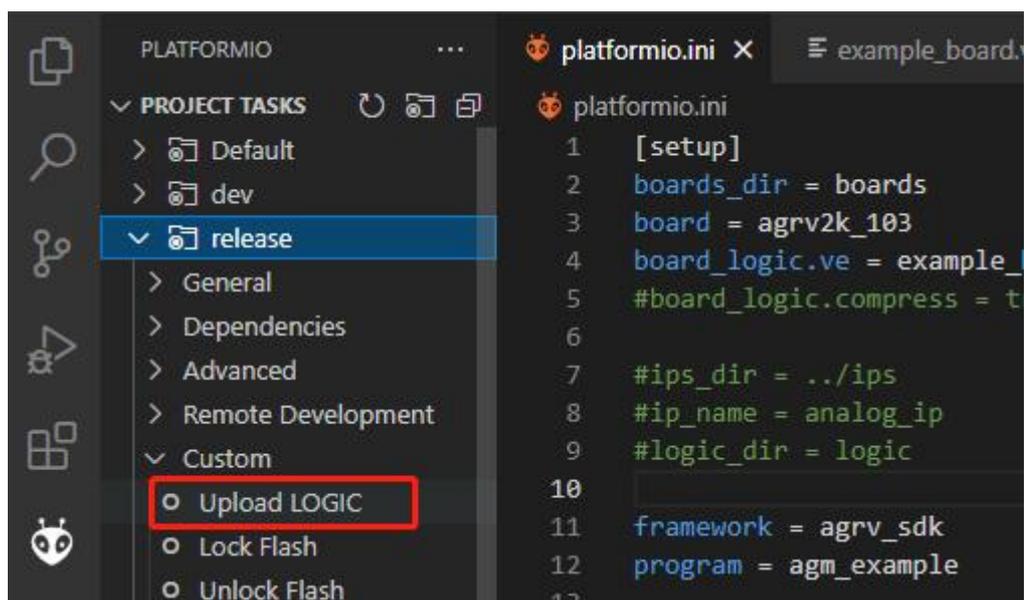


注：如果第一步从下拉列表中找到【J-Link】项，可以把下拉列表打开，插拔 Jlink 几次，找列表中的变化项。列表中的那个变化项，就是要更新驱动项的。

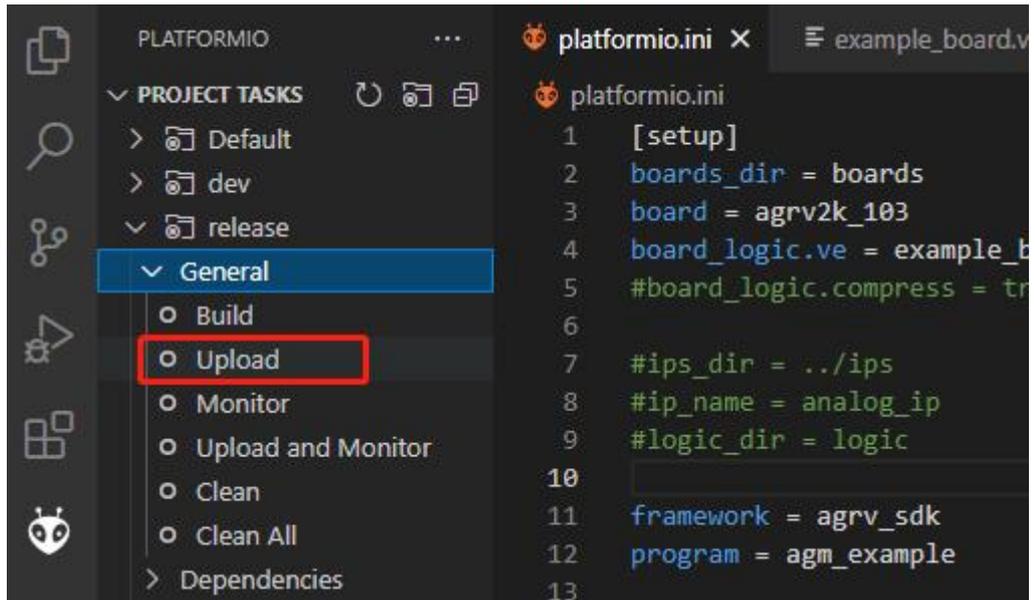
烧录：

新开发板第一次使用，要先烧录 VE 配置。（不烧录 VE 而先烧录程序 bin，会报错）

烧录 VE：

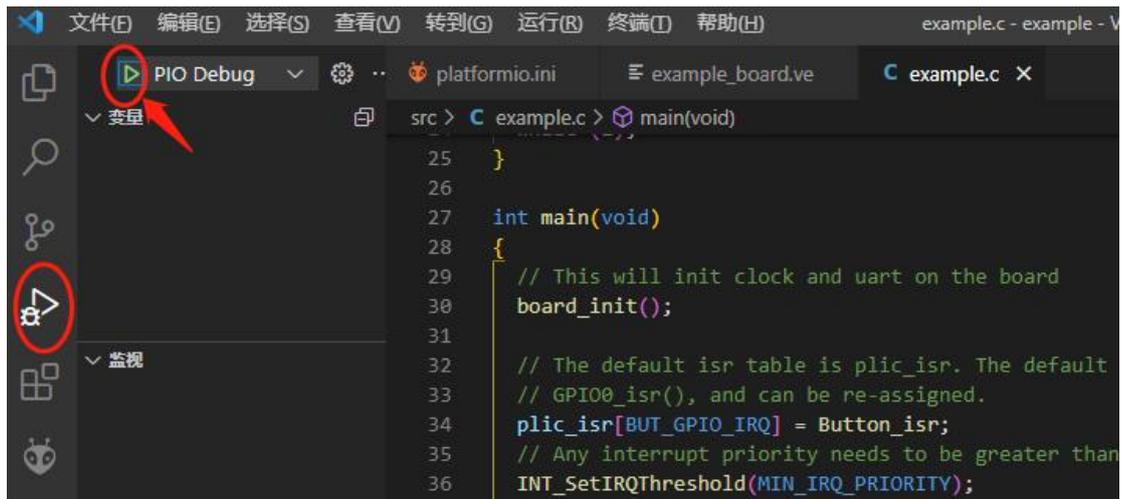


烧录程序：

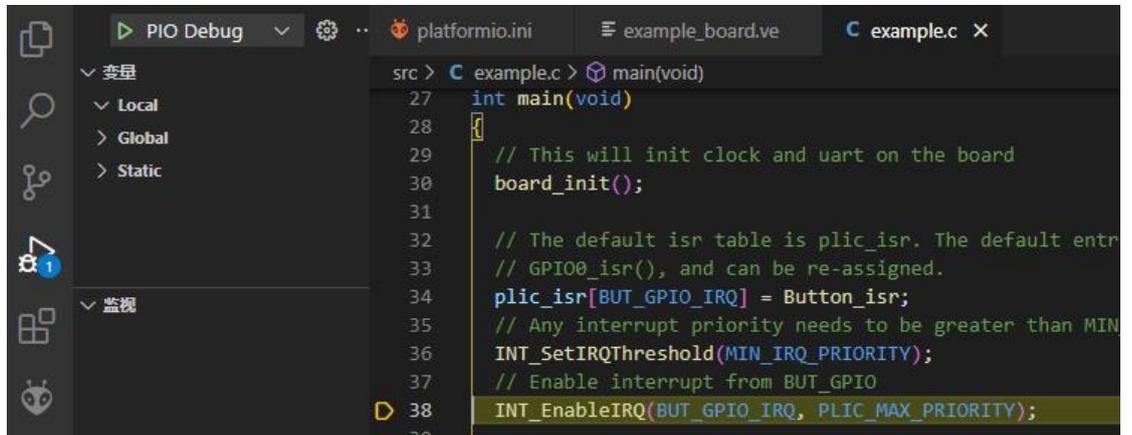


仿真：

点击仿真按钮，可以进入仿真调试。可单步运行到 main 函数的结尾。



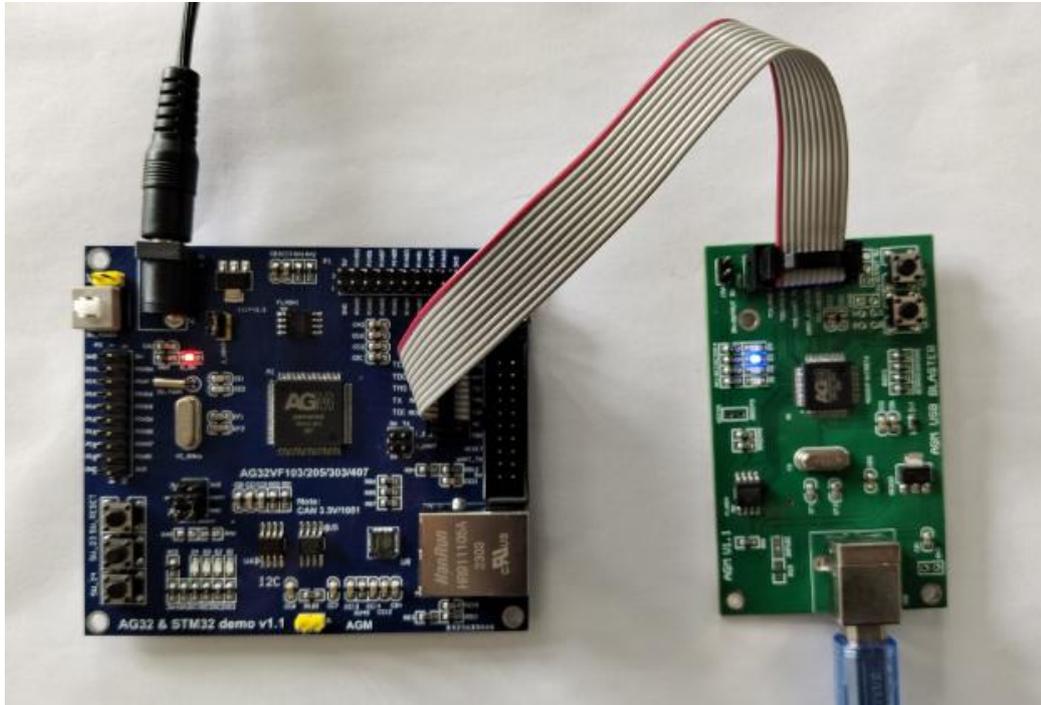
单步状态下如图：



如果使用官方烧录器：

连线：

连线后的样子如下图：



配置：

确认 platformio.ini 中的以下两行配置是否打开

**`debug_tool = cmsis-dap-openocd`**

**`upload_protocol = cmsis-dap-openocd`**

使用官方烧录器，PC 是免驱动的。

烧录：

烧录 VE 和烧录程序 bin，同上边的 Jlink 的烧录过程。

不再重复，请参考上边。

调试：

调试的启动和过程，也同上边的 Jlink 的调试过程。

不再重复，请参考上边。

#### 四、查看 led 灯：

在 example 样例程序的 main 函数中，最后是调用函数 TestGpio()。

进入 TestGpio()函数，里边是对 LED 灯的闪灯操作。

由于我们在第三步时，删除了 example\_board.ve 里的全部 IO 配置。现在把 led 的 pin 脚映射恢复回来。copy 下边的设置到 ve 文件：

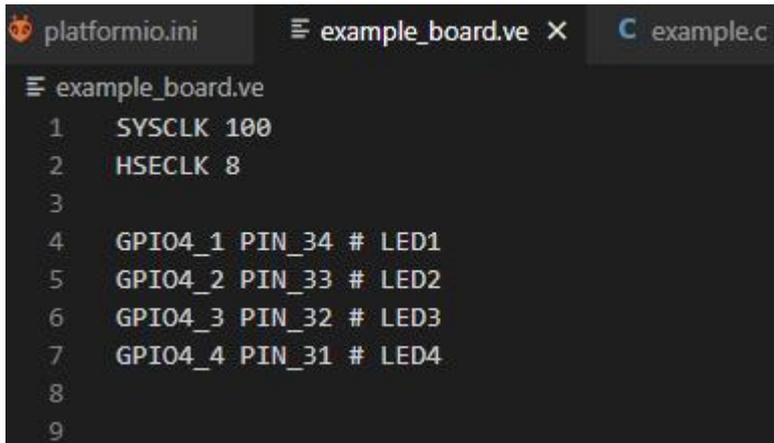
**`GPIO4_1 PIN_34 # LED1`**

**`GPIO4_2 PIN_33 # LED2`**

**`GPIO4_3 PIN_32 # LED3`**

**`GPIO4_4 PIN_31 # LED4`**

copy 后效果如下图:



```
platformio.ini  example_board.ve X  example.c
example_board.ve
1  SYSCLK 100
2  HSECLK 8
3
4  GPIO4_1 PIN_34 # LED1
5  GPIO4_2 PIN_33 # LED2
6  GPIO4_3 PIN_32 # LED3
7  GPIO4_4 PIN_31 # LED4
8
9
```

此时，再次烧录 ve 文件。

(烧录 VE 的方式，如上边第三步)

烧录成功后，可以看到左下角的几个 led 闪烁起来。

## 五、查看 log 输出:

在以上的基础上，修改以下三项:

### 1. Platformio.ini 中:

确认 logger\_if 配置是打开的:

**logger\_if = UART0**

**build\_flags = -DBAUD\_RATE=115200**

以上两项分别设置: log 输出通过 uart0 输出、输出的波特率是 115200.

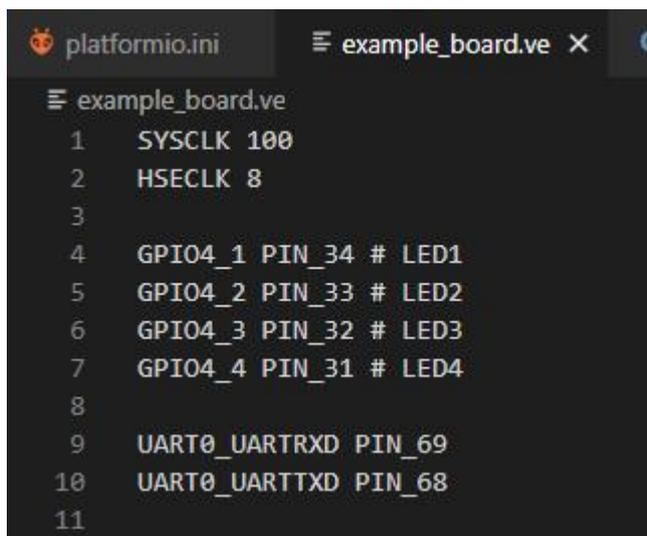
### 2. Example\_board.ve 中:

Copy 以下的串口 IO 配置到 ve 中去:

**UART0\_UARTRXD PIN\_69**

**UART0\_UARTTXD PIN\_68**

修改后图示如下:



```
platformio.ini  example_board.ve X  C
example_board.ve
1  SYSCLK 100
2  HSECLK 8
3
4  GPIO4_1 PIN_34 # LED1
5  GPIO4_2 PIN_33 # LED2
6  GPIO4_3 PIN_32 # LED3
7  GPIO4_4 PIN_31 # LED4
8
9  UART0_UARTRXD PIN_69
10  UART0_UARTTXD PIN_68
11
```

### 3. 在 example\_gpio.c 中的 TestGpio()函数中，while(1)里增加一句 log:

```
platformio.ini example_board.ve example.c
src > C example_gpio.c > TestGpio()
1  #include "example.h"
2
3  void TestGpio()
4  {
5      printf("Testing gpio\n");
6      int counter = 0;
7      while (1) {
8          printf("Test log\r\n");
9          UTIL_IdleUs(100e3);
10         GPIO_Toggle(EXT_GPIO, EXT_GPIO_BITS);
11     }
12 }
13
```

最后，烧录 VE，烧录程序 bin。

然后，用串口线，接到开发板的串口 0（上边的那组）上，在 PC 端的串口工具（波特率 115200）上可以看到 log 的输出信息。

以上，只是展示了拿到开发板后验证 LED 灯和 log 通过串口 0 输出的样例。更多的驱动使用，请参考文档《AG32 驱动的使用.pdf》