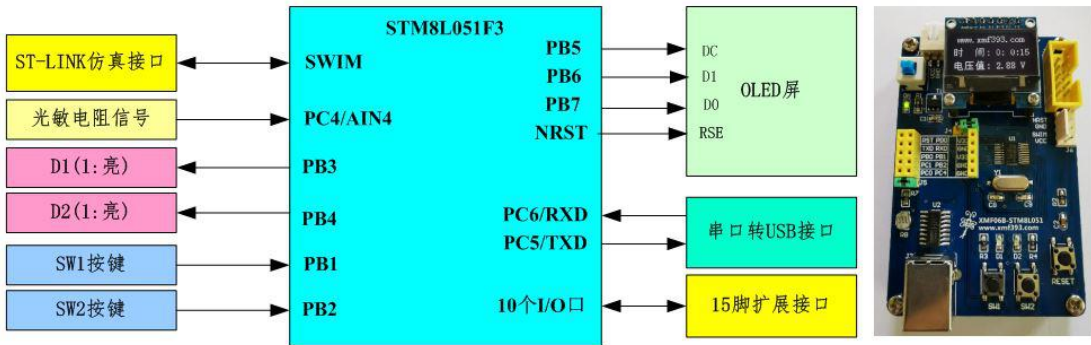


IAR 环境下 STM8 微控制器开发快速入门教程

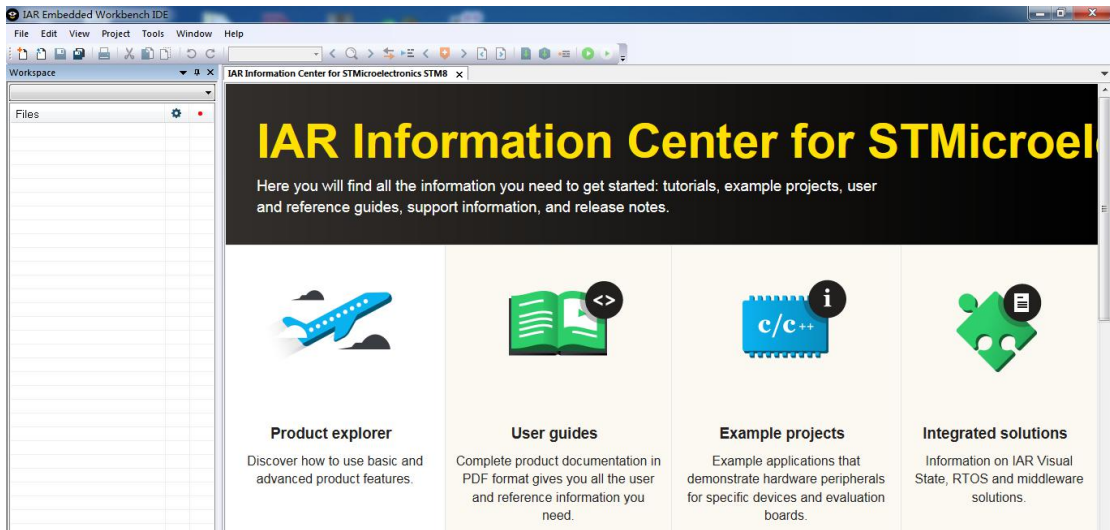
广东职业技术学院 www.xmf393.com 欧浩源 (ohy3686@qq.com)

● 在 XMF06B 开发板上，完成以下任务：

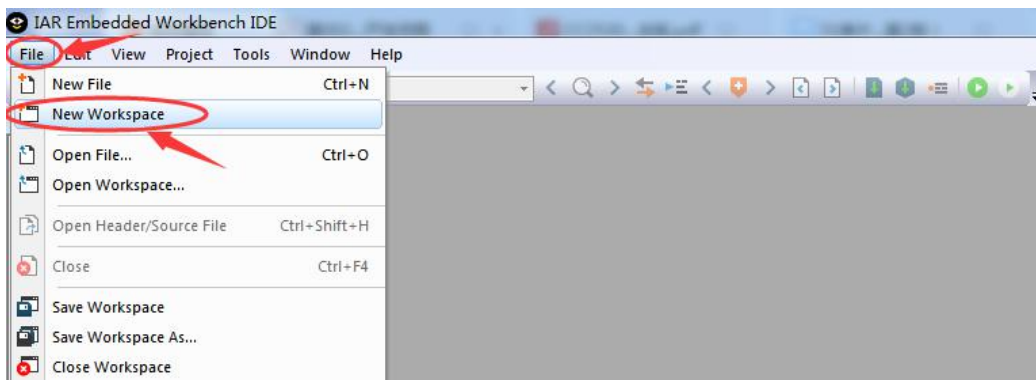
- 【1】设计初始化函数 `Init_Port()`，配置 D1 和 D2 灯的引脚为 **推挽输出** 模式，速度为 **2MHz**。
- 【2】设计跑马灯函数 `LED_Running()`，实现 D1 点亮，延时，D2 点亮，延时；D1 熄灭，延时，D2 熄灭，延时，(LED 灯高电平点亮)
- 【3】在 `main()` 函数中，反复调用 `LED_Running()`，实现跑马灯功能。



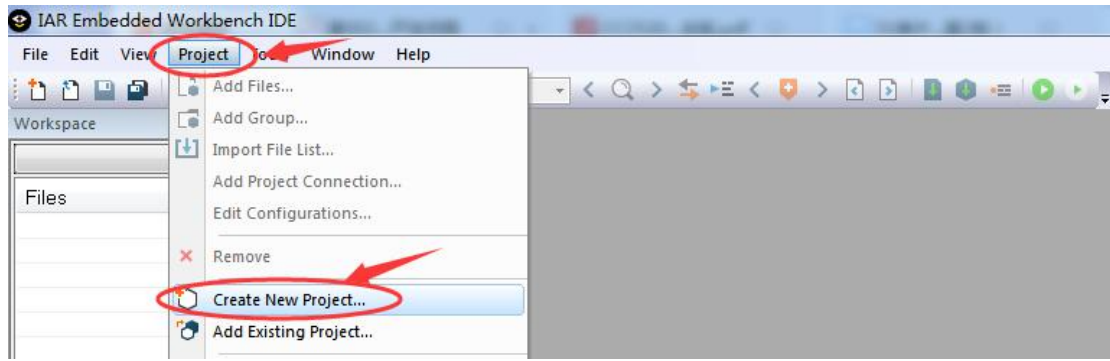
【1】首先在电脑新建一个空的文件夹“跑马灯工程”，然后打开 IAR for STM8 3.10。



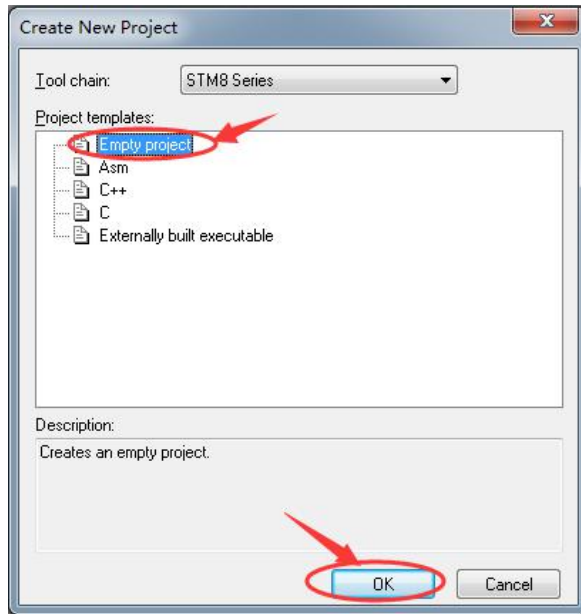
【2】点击“File”菜单，在下拉列表中点击“New Workspace”创建一个新的工作区。



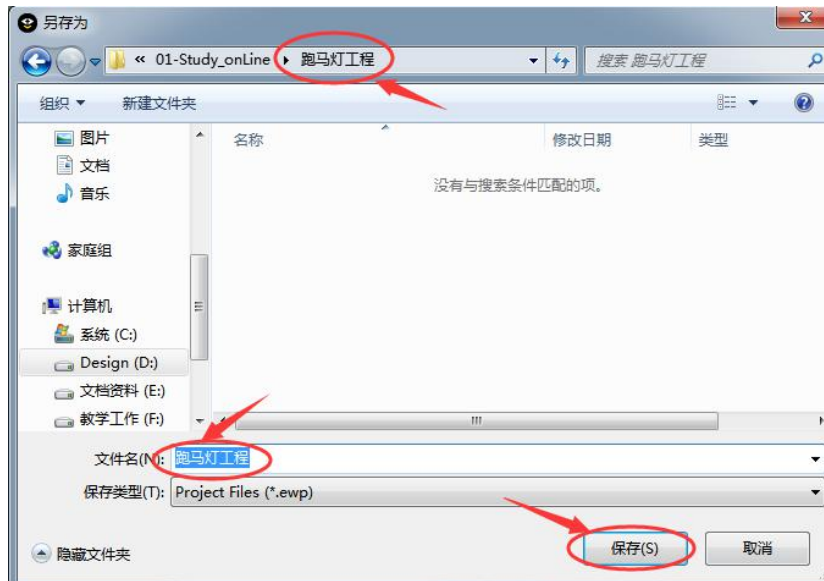
【3】点击“Project”菜单，在下拉列表中点击“Create New Project...”。



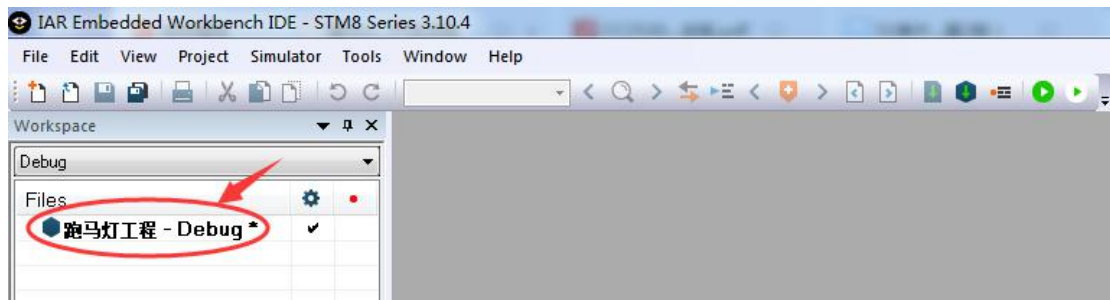
【4】在弹出的新建工程对话框中，选择“Empty Project”，然后点击“OK”按钮。



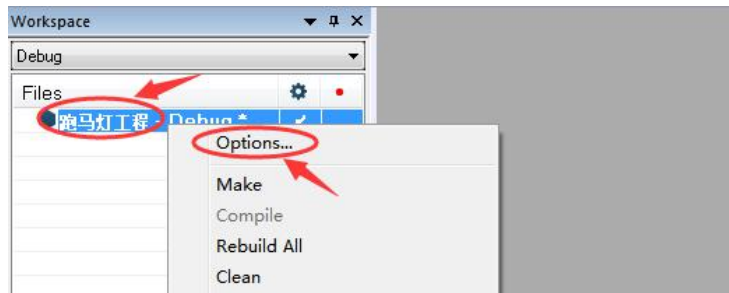
【5】在弹出的文件对话框中，将工程保存到刚才新建的“跑马灯工程”空文件夹中，并为该新建工程命名为“跑马灯工程”，然后点击“保存”按钮。



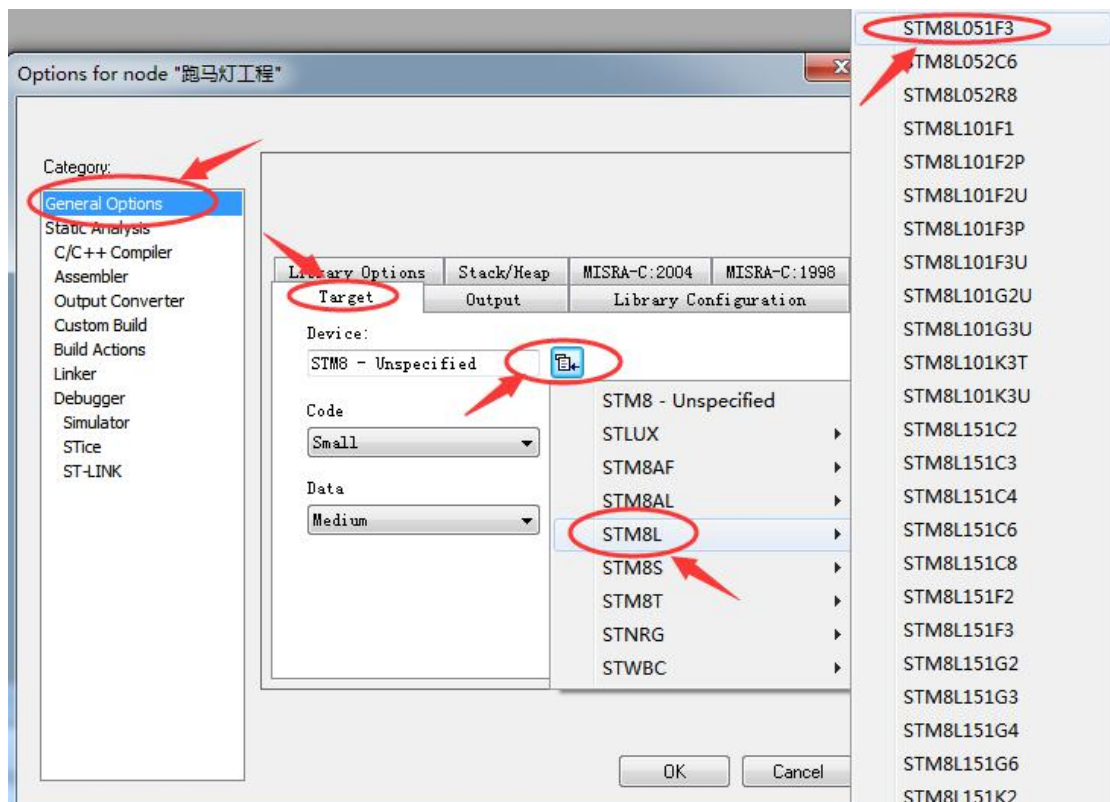
【6】此时，一个名字为“跑马灯工程”的新工程就创建好了。



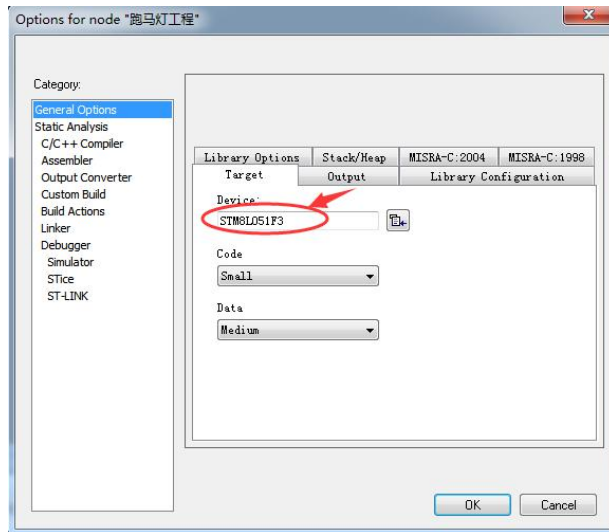
【7】鼠标右键单击“Workspace”工作区中“跑马灯工程”的工程名，在弹出的下拉菜单中，点击“Options”选项。



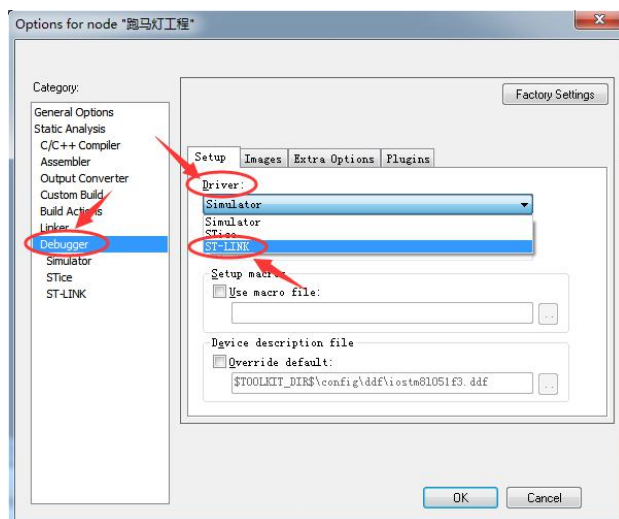
【8】在弹出的参数设置对话框中，在左侧的属性列表中选择“General Options”，点击右侧“Target”选项卡中“Device”编辑框右侧的按钮，在弹出的下拉列表中选择“STM8L”，其右侧的黑色三角符号将会展开该系列所有的芯片型号，你可以根据项目的实际情况进行选择，在本教程这种选择“STM8L051F3”这个型号。



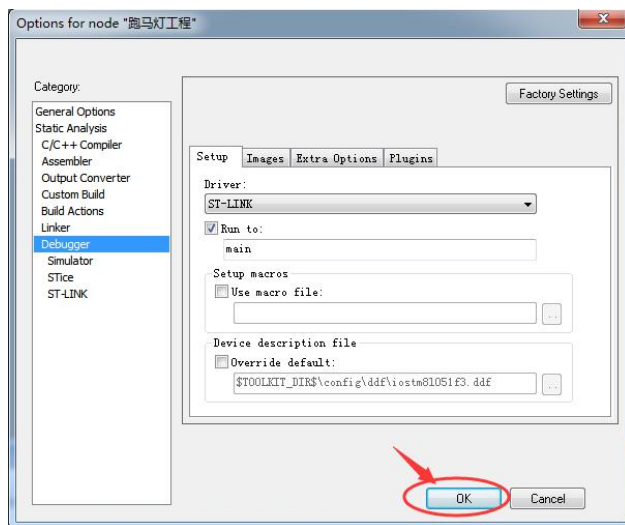
【9】在参数设置对话框“General Options”属性“Target”选项卡中，芯片类型“Device”参数编辑框中出现了刚才选择的芯片型号。



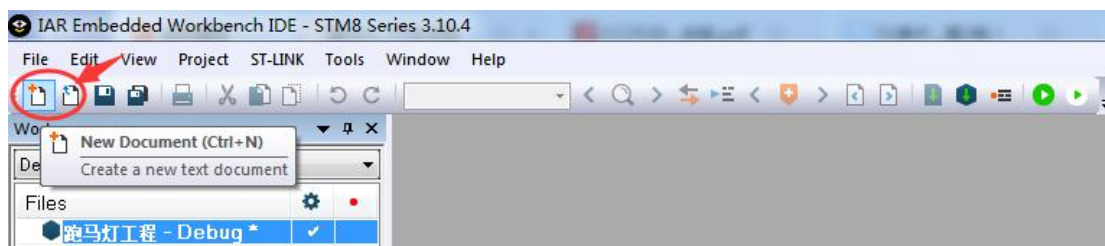
【10】在参数设置对话框“Debugger”属性“Setup”选项卡中，点击“Driver”编辑框右侧的三角符号，在其下拉列表中选择“ST-LINK”。



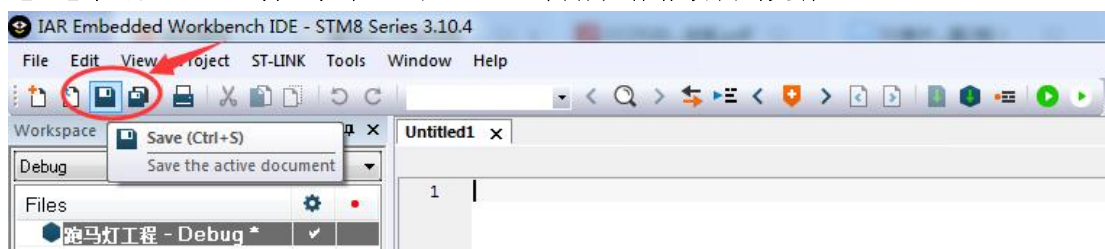
【11】此时，工程的仿真器设置为 ST-LINK，点击右下方的“OK”按钮，即可完成参数配置。



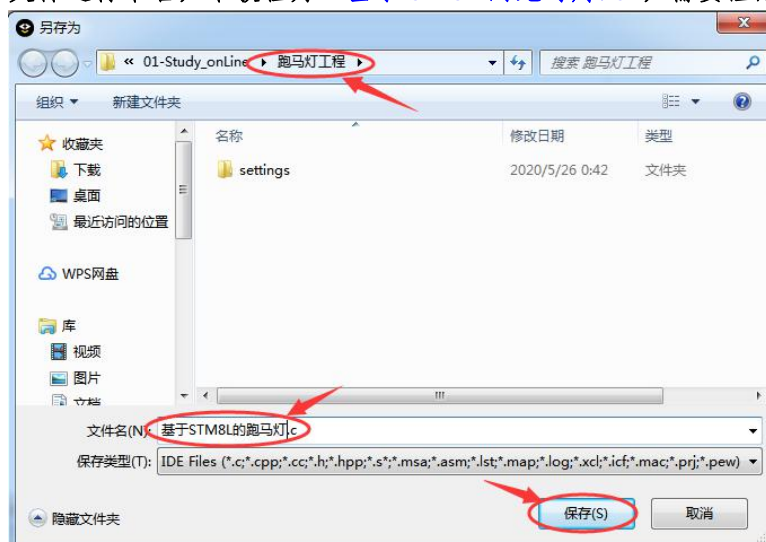
【12】点击 IAR 左上方工具栏上的“New Document”图标，新建一个代码文件。



【13】点击 IAR 左上方工具栏上的“Save”图标，保存该代码文件。



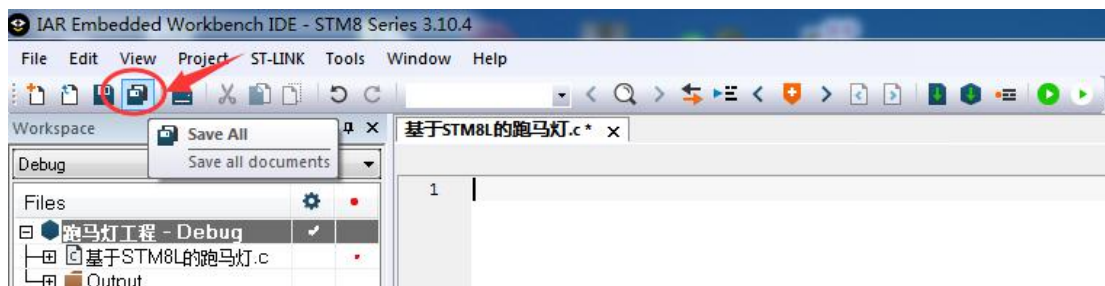
【14】在弹出的文件对话框中，将其保存路径选择为“跑马灯工程”文件夹，并为该代码文件进行命名，本教程为“基于STM8L的跑马灯.c”，需要注意的是后缀需要是.c。



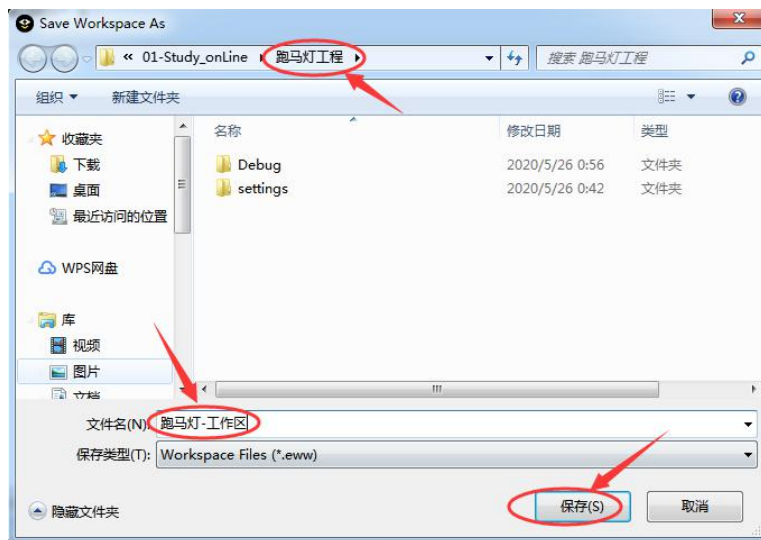
【15】鼠标右键单击“Workspace”工作区中“跑马灯工程”的工程名，在弹出的下拉菜单中，点击“Add”选项，在其展开的列表中，将需要的代码文件添加到工程中。



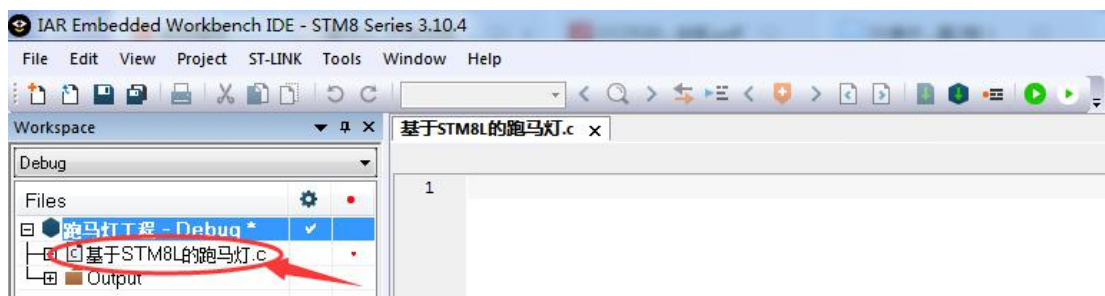
【16】点击 IAR 左上方工具栏上的“Save All”图标，保存该代码文件。



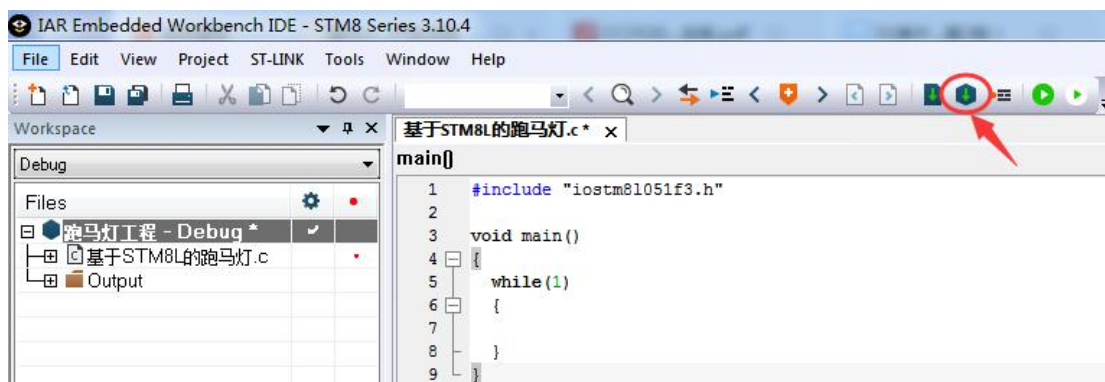
【17】在弹出的文件对话框中，将其保存路径选择为“跑马灯工程”文件夹，在这里要求保存的是工作区文件，在保存之前需要为该工作区进行命名，然后点击“保存”。



【18】此时，项目创建和参数设置等工作已全部完成，接下来就可以根据应用项目的需求，在项目的代码文件中进行程序设计和代码编写了。



【17】在代码文件中，首先引入芯片的头文件“iostm81051f5.h”，编写一个简单的框架，并点击 IAR 工具栏上的“Make”进行编译链接，正常来说应该是，0 错误和 0 警告。



【18】为两个LED灯引脚进行宏定义。

```
3 #define D1 PB_ODR_ODR3
4 #define D2 PB_ODR_ODR4
```

【19】编写一个普通的延时函数。

```
6 void Delay(unsigned int t)
7 {
8     while(t--);
9 }
```

【20】编写 GPIO 端口初始化函数。

```
11 void Init_Port()
12 {
13     PB_DDR_DDR3 = 1; //将PB3设置为输出模式
14     PB_DDR_DDR4 = 1; //将PB4设置为输出模式
15
16     PB_CR1_C13 = 1; //将PB3设置为推挽输出
17     PB_CR1_C14 = 1; //将PB4设置为推挽输出
18
19     PB_CR2_C23 = 1; //将PB3的速度设置为2MHz
20     PB_CR2_C24 = 1; //将PB4的速度设置为2MHz
21 }
```

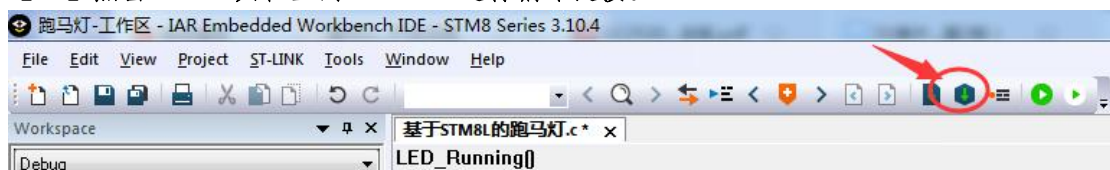
【21】编写 LED 灯跑马灯函数。

```
23 void LED_Running()
24 {
25     D1 = 1; //D1灯点亮
26     Delay(60000); //延时片刻
27     D2 = 1; //D2灯点亮
28     Delay(60000); //延时片刻
29
30     D1 = 0; //D1灯熄灭
31     Delay(60000); //延时片刻
32     D2 = 0; //D2灯熄灭
33     Delay(60000); //延时片刻
34 }
```

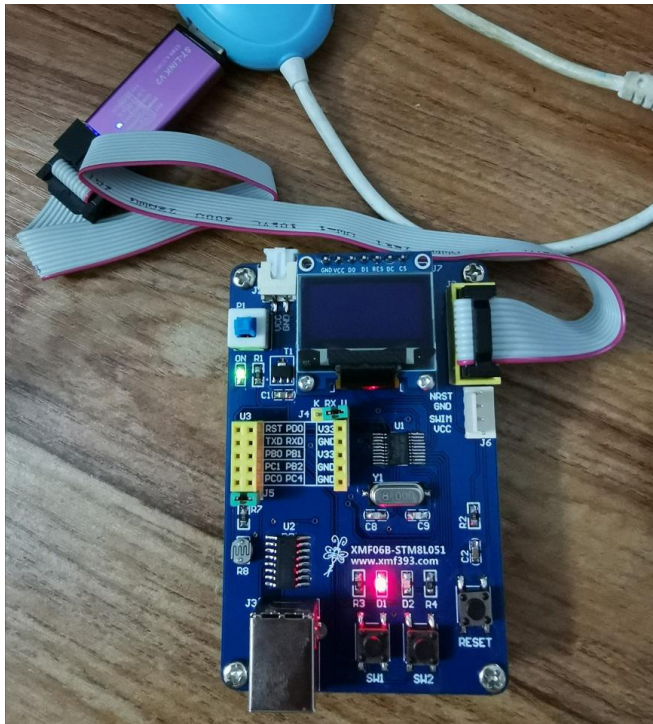
【22】编写 main() 函数。

```
36 void main()
37 {
38     Init_Port(); //初始化GPIO
39     while(1)
40     {
41         LED_Running(); //循环执行跑马灯函数
42     }
43 }
```

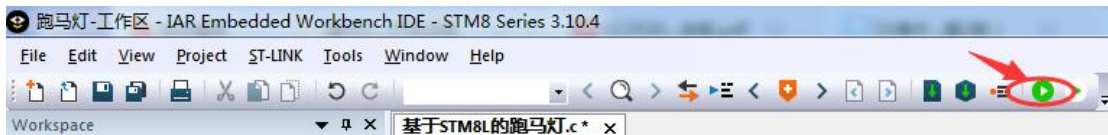
【23】点击 IAR 工具栏上的“Make”进行编译链接。



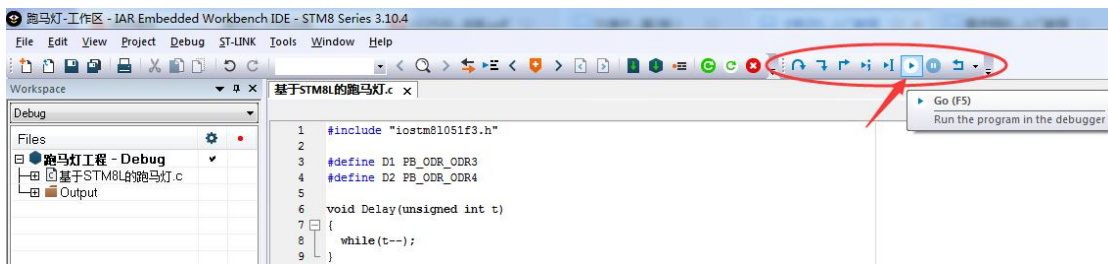
【24】如果编译链接没有错误和警告，则可将 ST-LINK 仿真器与目标板进行连接。



【25】点击 IAR 工具栏上的“Download and Debug”，进行程序下载和仿真调试。



【26】进入仿真调试界面后，在工具栏的左侧有一排跟运行调试相关的图标，点击一个背景为白色的蓝色三角符号，即“Go”图标，全速运行程序。此时，你会看到 XMF06B 目标板上 D1 和 D2 两个灯在循环闪烁。



【结束】IAR 环境下 STM8 应用开发流程全部完成，更多精彩教程详见：www.xmf393.com。

【特别提示】：关于 STM8 的外设功能与寄存器定义，可以参考 RM0031 手册。



RM0031 Reference manual

STM8L050J3, STM8L051F3, STM8L052C6, STM8L052R8 MCUs
and STM8L151/L152, STM8L162, STM8AL31, STM8AL3L lines