

概述

本文主要介绍移植 Keil 工程至 IAR 的操作流程。

注意： 本文档主要以 SPC1068 为例进行介绍。

目录

1	移植 Keil 工程至 IAR	6
---	-----------------------	---

SPIN TROL

图片列表

图 1-1: 文件缺失	6
图 1-2: 删除示例工程原文件	6
图 1-3: 添加.c 文件	7
图 1-4: 添加启动文件	7
图 1-5: 添加.h 文件	8
图 1-6: 库函数编译报错	8
图 1-7: 编译成功	8
图 1-8: 加入测试程序	9
图 1-9: 移植成功	9

版本历史

版本	日期	作者	状态	变更
C/0	2024-02-27	周佳莉	Released	首次发布。

SPIN TROL

术语或缩写

术语或缩写	描述
/	/

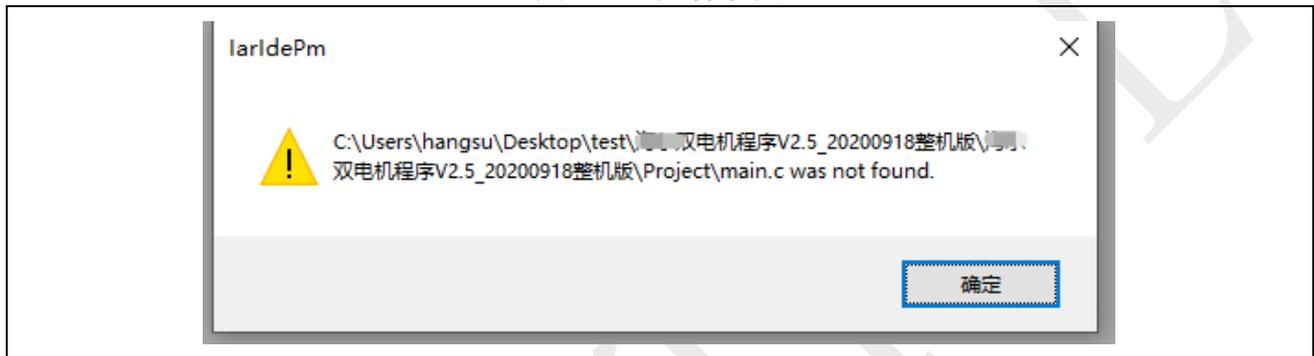
SPIN TROL

1 移植 Keil 工程至 IAR

1. 复制 EWARM（包含 IAR 工程示例文件）文件到 Project 目录下。
2. 复制 IDE_Support（包含 IAR 下载支持文件）到 Libraries 同级目录
3. 复制 iar（包含 IAR 启动文件）到 Libraries\CMSIS\device\startup 目录下

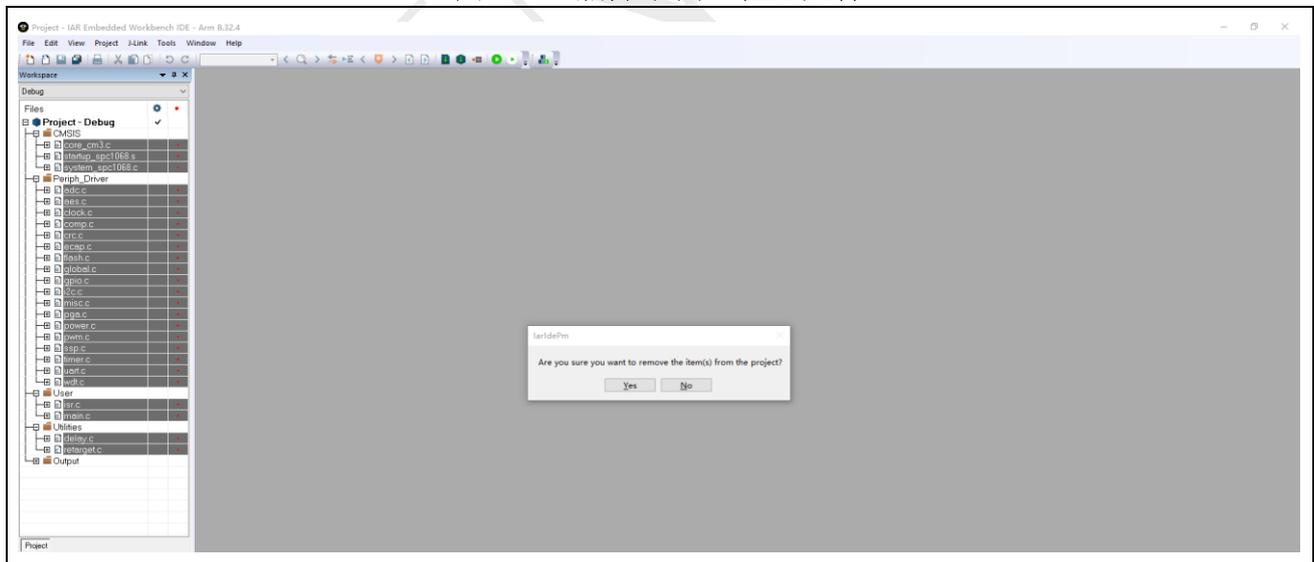
打开 IAR 工程 Project.eww，会显示文件缺失，如图 1-1 所示。这里一路点击确定即可，之后需要将这些文件重新导入。

图 1-1: 文件缺失



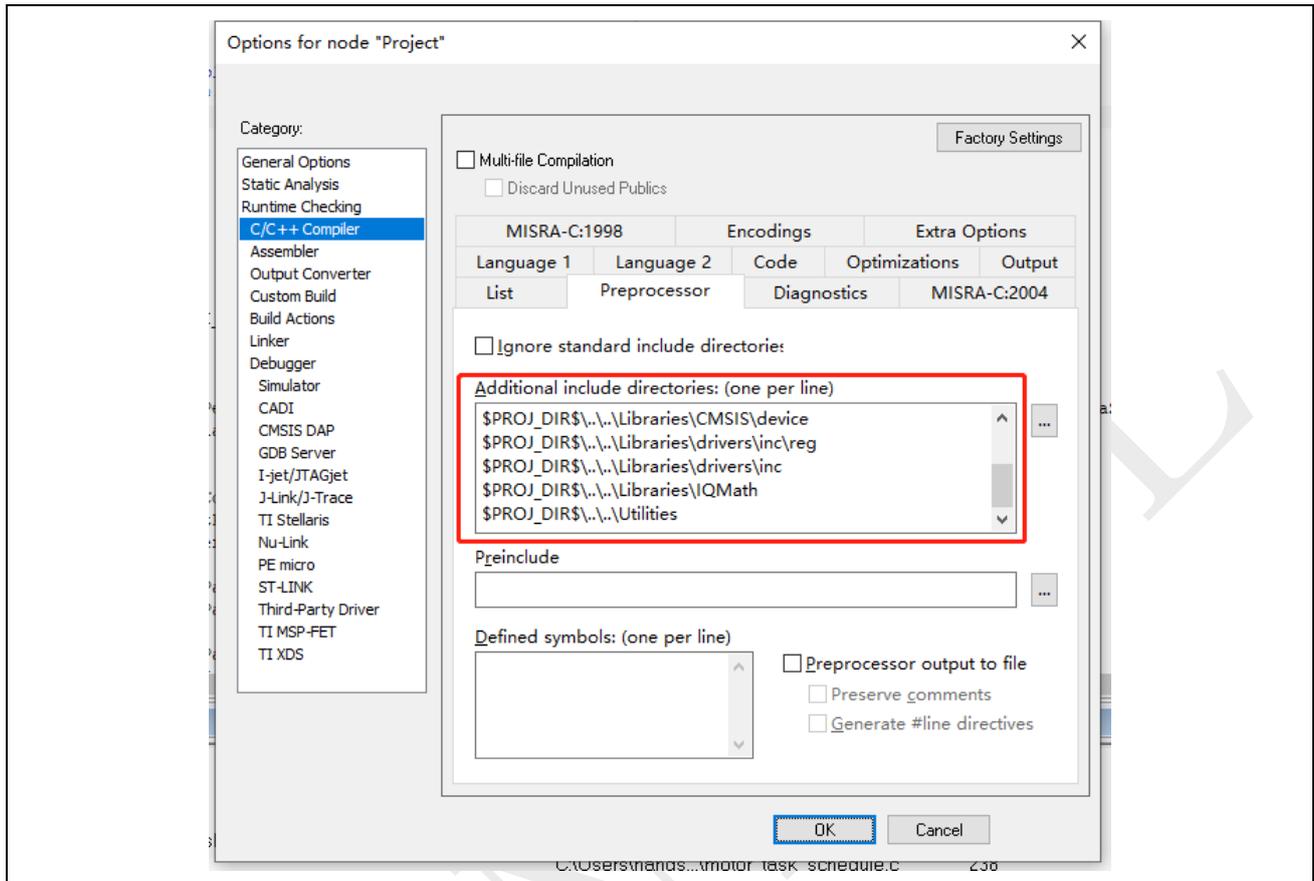
4. 将 IAR 工程示例文件中原有文件删除，如图 1-2 所示。

图 1-2: 删除示例工程原文件



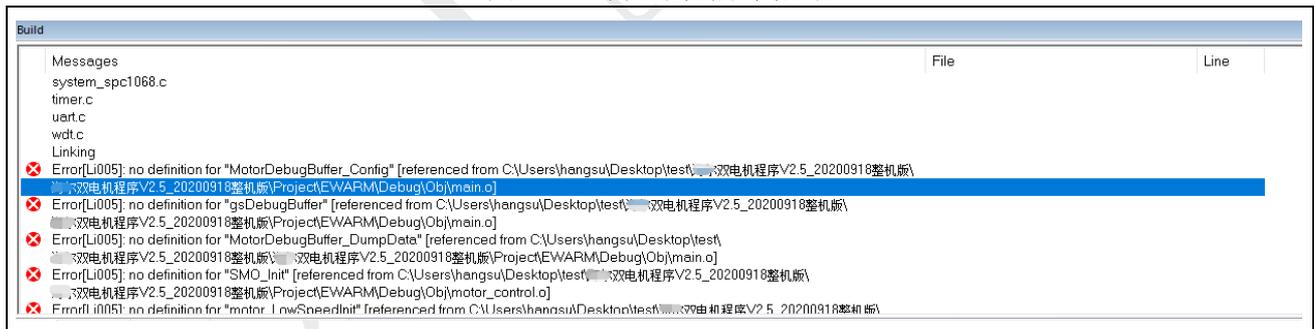
7. 添加 Keil 工程中所有的.h 文件，如图 1-5 所示。

图 1-5: 添加.h 文件



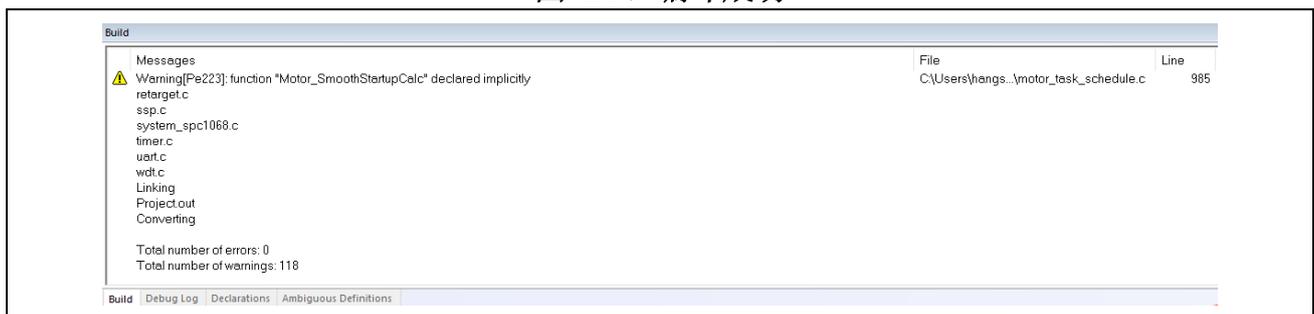
8. 编译工程，如图 1-6 所示，注释所有库函数，之后库函数需要依据 IAR 规则重新添加。

图 1-6: 库函数编译报错



9. 编译工程，直到成功，如图 1-7 所示，期间可根据需要修改代码优化等级。

图 1-7: 编译成功



10. 在主函数中加入测试程序，进行测试，如图 1-8 所示。

图 1-8: 加入测试程序

```
int main()
{
    int i;

    Sys_Init();

    CLOCK_Init(CLOCK_FROM_RC24MHZ,CLOCK_HCLK_100MHZ);

    GPIO_SetPinChannel(GPIO_34, GPIO34_UART_TXD);
    GPIO_SetPinChannel(GPIO_35, GPIO35_UART_RXD);
    UART_Init(UART, 38400);
    //无校验
    UART->UARTLCR.bit.PEN = UARTLCR_BIT_PEN_NO_PARITY;
    //打开串口
    UART->UARTDLH_UARTIER.UARTIER.bit.UUE = UARTIER_BIT_UUE_UART_ENABLE;
    printf("Test OK\n");

    while(1)
    {
    }

    Sys_Init();

    CLOCK_EnableModule(WDT0_MODULE);
    WDT_Init(WDT0,1000);
    WDT_Run(WDT0);
    #if (CHIP_SELECTION == SPC1068)
    CLOCK_Init(CLOCK_FROM_RC24MHZ,CLOCK_HCLK_100MHZ);
    #elif (CHIP_SELECTION == SPD1078)
```

11. 移植成功标志，如图 1-9 所示。

图 1-9: 移植成功

