

CYT4BB7 开发资 料汇总

| | | |
|-------|--|----|
| 第 1 章 | 前言 | 3 |
| 第 2 章 | IAR 环境配置及基本使用 | 4 |
| 2.1 | IAR 环境配置 | 4 |
| 2.2 | IAR 基本使用说明 | 5 |
| 2.2.1 | 界面说明 | 5 |
| 2.2.2 | 常用快捷键释义、绑定、添加 | 5 |
| 2.2.3 | 工程目录说明 | 8 |
| 2.2.4 | 多工程说明 | 8 |
| 2.2.5 | 函数、变量跳转功能说明 | 9 |
| 2.3 | IAR 下载、调试说明 | 11 |
| 2.3.1 | IAR 下载说明 | 11 |
| 2.3.2 | 如何进入调试模式 | 11 |
| 2.3.3 | 调试界面说明 | 12 |
| 2.3.4 | 多核同步调试说明 | 15 |
| 第 3 章 | CYT4BB7 开发注意事项 | 18 |
| 第 4 章 | 编译、下载、调试报错及解决办法汇总 | 19 |
| 4.1 | 编译报错 | 19 |
| 4.1.1 | An exception occurred when building the project: @J@L@@@@@x@Y@ù@ Unicode @@@@@@y@磐@@@@@ | 19 |
| 4.1.2 | '...drivers/adc/cy_adc.c', needed by '.../cy_adc.o', Missing and no known rule to make it | 20 |
| 4.1.3 | Cannot open source file“xxxx.h” | 20 |
| 4.1.4 | Error[Lt039]: Incompatible architecture attributes | 23 |
| 4.2 | 下载及调试报错 | 24 |
| 4.2.1 | Fatal error: Probe not found | 24 |
| 4.2.2 | Fatal error: Failed to connect to CPU | 24 |

| | | |
|--------------|--|-----------|
| 4.2.3 | No emulators connected via USB, Do you want to connect to a J-Link via TCP/IP instead ? | 25 |
| 4.2.4 | Failed to load flash loader: caprogram filesIAR Systems Embedded Workbench 9.2\arm/config/flashloader/infineon/flashCYT4 4M CFLASH.flash | 26 |
| 4.2.5 | 下载或者调试在加载界面卡死 | 27 |
| 4.2.6 | Could not stop the CPU (possibly it is in sleep mode). Press OK to retry | 28 |
| 第 5 章 | 问题反馈注意事项及途径 | 30 |

第1章 前言

该汇总内容针对基于 IAR 9.40.1 版本的 CYT4BB7 三核芯片开发所产生的软件各类报错、错误使用方法等内容进行归纳总结并给出解决方法，对于 C 语言的语法编写类报错则不做提及，需自行解决。如果不熟悉 IAR 的基本使用方法，可参考本文档的第二章内容。

如有遇到未见到过的报错类型并且确认是软件本身的问题，可以加入逐飞科技交流群并联系群管理员**布丁**协助解决问题，如果该问题可稳定复现并解决，那么将记录在此汇总中以便后人学习参考。

在 IAR 使用过程中，有小部分原因是因为各类电脑管家、杀毒软件、权限不足等情况导致 IAR 无法正常打开或者编译，建议使用前先完整阅读教程并完全关闭所有类似软件，以获得良好的编程体验。

第2章 IAR 环境配置及基本使用

2.1 IAR 环境配置

下载 IAR 安装包，链接如下：

https://pan.baidu.com/s/1BNu17F-bBur_uXkNxwesOw



（安装路径以及文件路径不要有任何中文字符）

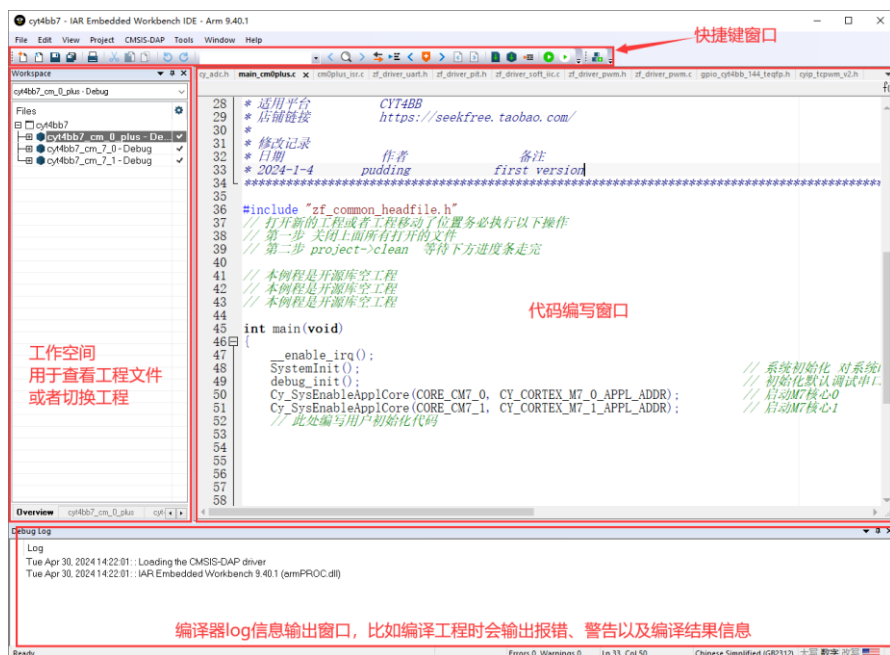
（安装路径以及文件路径不要有任何中文字符）

（安装路径以及文件路径不要有任何中文字符）

下载“IAR9.40.1”文件夹内的所有内容，确保关闭所有安全管家并且当前 windows 账户有完全管理员权限的情况下，双击“EWARM-9401-63915.exe”开始安装 IAR，**安装路径不要有中文字符**，建议默认安装所有驱动，安装期间同意 IAR 所有请求，软件安装完成后开始安装驱动，也全部同意。安装完成后按照注册教程进行软件本地注册，注册后才可以正常使用 IAR 软件。

2.2 IAR 基本使用说明

2.2.1 界面说明



2.2.2 常用快捷键释义、绑定、添加

I. 常用快捷键释义如下:

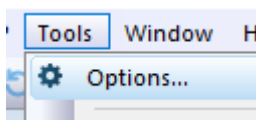


- 1.新建文件，可用于保存为不同文件类型，比如.C 文件或者.H 文件。
- 2.打开文件，用于打开外部文件，也可将文件直接拖动至代码编写窗口打开。
- 3.保存当前文件，也可使用快捷键 Ctrl+S。
- 4.保存所有文件。
- 5.搜索当前文件内容，也可使用快捷键 Ctrl+F。

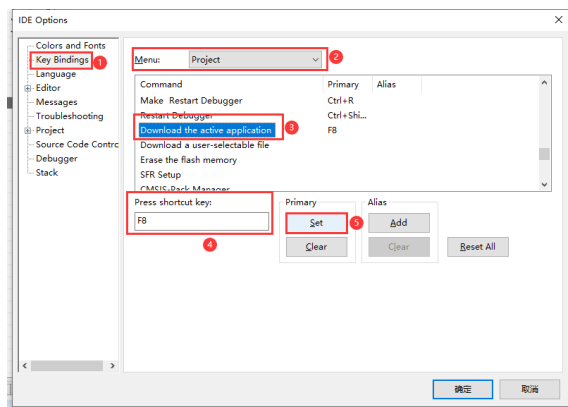
6. 替换当前文件下的内容，也可使用快捷键 Ctrl+H。
7. 跳转到指定行。
8. 工程局部编译，也叫增量编译，在之前的编译文件的基础上继续编译修改过的文件，提高编译效率。
9. 工程全局编译，清除之前的编译文件，从头开始编译一次，耗时会较长，一般仅第一次编译时需要使用。
10. 下载并调试，在成功编译工程后(无报错)，将程序烧录到单片机并且进入调试模式，可在调试模式中查看单片机运行状态、内存数据、变量信息等。
11. 调试但不下载，仅让单片机进入调试模式并执行当前工程的程序，不将程序写入到单片机 flash，调试结束后单片机不会运行调试的程序段。

II. 在 IAR 的设置中可以设置不同功能对应的快捷键，下面简单演示设置“仅下载”功能的快捷键至 F8。

1. 打开 IAR 设置

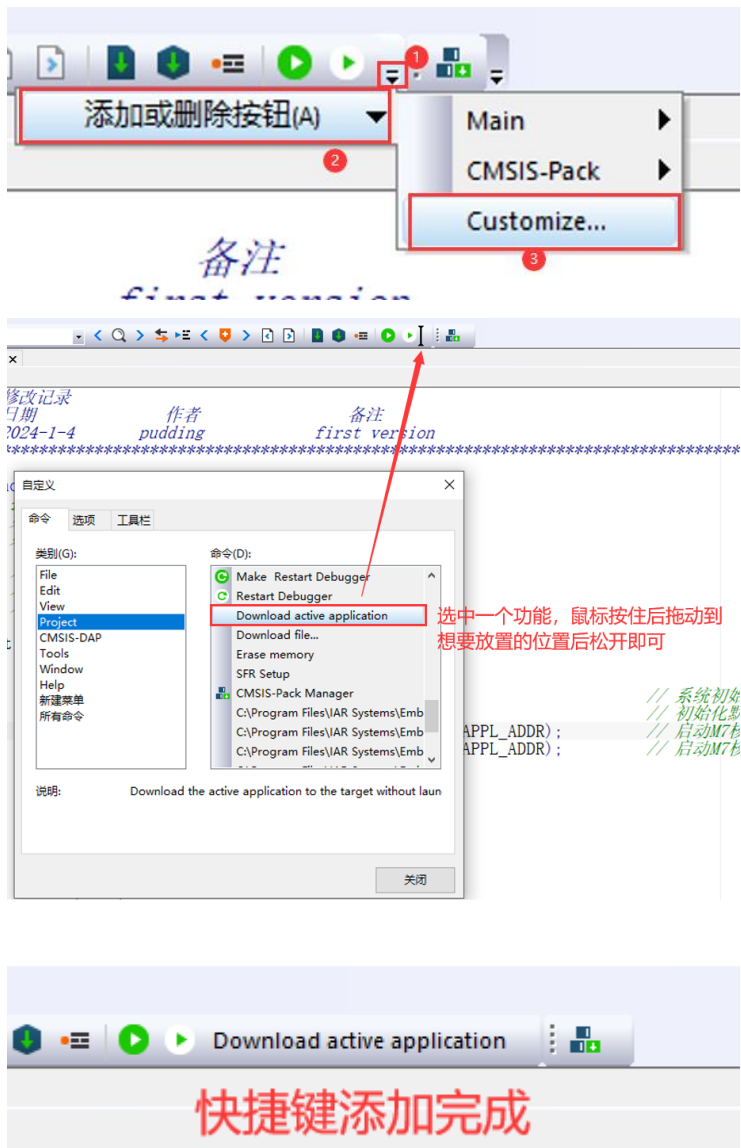


2. 在设置中，选择按键绑定设置，在 project 菜单下，找到“Download the active application”，在 press shortcut key 输入框中输入 F8，然后点击 set 按钮。



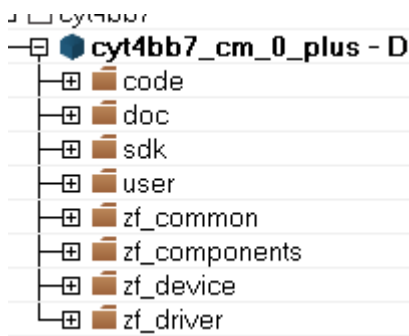
- 3.若是提示按键绑定冲突则选择取消之前的绑定，将当前按键作为该功能的唯一快捷键。
- 4.最后点击确定即可完成快捷键绑定。

III.快捷键添加功能，这里演示单独增加一个仅下载的快捷键：



2.2.3 工程目录说明

双击工作空间的工程可以打开工程目录，目录下有多个不同的文件



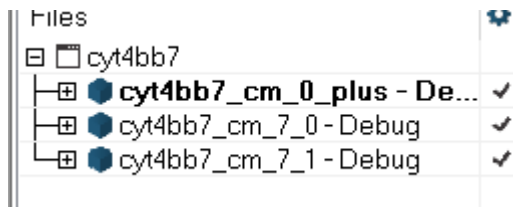
各个文件存放内容如下：

- 1.**code 文件夹**，用于方便用户自行添加程序文件，该目录已经添加到编译路径下，用户使用时若将自己的文件放置在 **code** 文件夹下并且在工程目录下添加即可直接编译。
- 2.**doc 文件夹**，用于存放文档信息，比如开源协议声明、当前库版本号等。
- 3.**sdk 文件夹**，存放官方的 SDK 文件，逐飞开源库是基于官方的 SDK 进行二次开发，意在以更加简洁方便的方式使用单片机的各种功能。
- 4.**user 文件夹**，存放中断文件以及主函数文件。
- 5.**zf_common 文件夹**，逐飞开源库的公共层。
- 6.**zf_components 文件夹**，逐飞开源库的组件层。
- 7.**zf_device 文件夹**，逐飞开源库的设备层。
- 8.**zf_driver 文件夹**，逐飞开源库的驱动层。

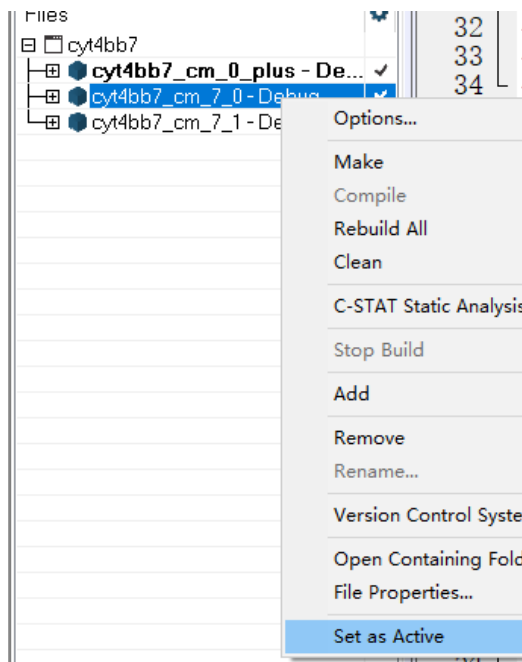
2.2.4 多工程说明

在使用某些多核心芯片时，可能需要多个工程来满足对多个核心的程序编写、编译下、载。以 CYT4BB7 单片机举例，该单片机有三个核心，因此可以在 IAR 的工作空间看

到有三个工程：



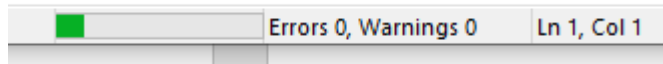
由于 IAR 功能限制，同一时刻仅能对其中一个工程进行编译，观察工程目录发现其中一个工程名字加粗，若此时点击编译按键，则是对名字加粗的工程进行编译，若要切换工程看右击工程后点击“Set as active”。设置后会发现被设置为活动工程的名称将加粗。



2.2.5 函数、变量跳转功能说明

IAR 支持函数、变量的定义跳转功能，但仅可以跳转当前的活动工程的函数或者变量定义，当初次打开工程时，IAR 主界面右下角会出现一个进度条，等待进度条加载完毕即

可跳转：

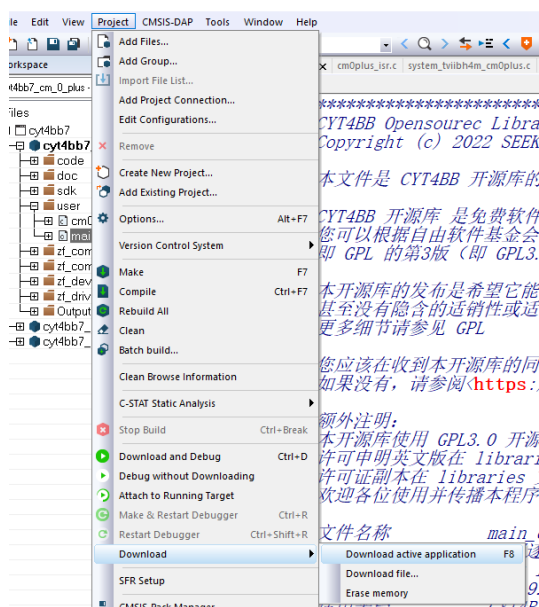


跳转可使用快捷键，选中需要跳转的内容后，单击键盘上的 F12 按键即可直接跳转。

2.3 IAR 下载、调试说明

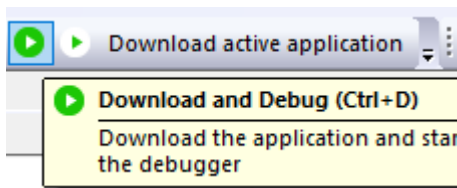
2.3.1 IAR 下载说明

当工程正确编译且无报错后，单击“Download active application”按钮即可将当前工程下载到单片机，如果使用的单片机是多核芯片，那么仅会下载到对应核心的 flash 区域，不会影响其他核心的 flash 内容。如果增加了快捷键即可直接点击快捷键，如果没有增加快捷键也可以通过以下按钮下载：



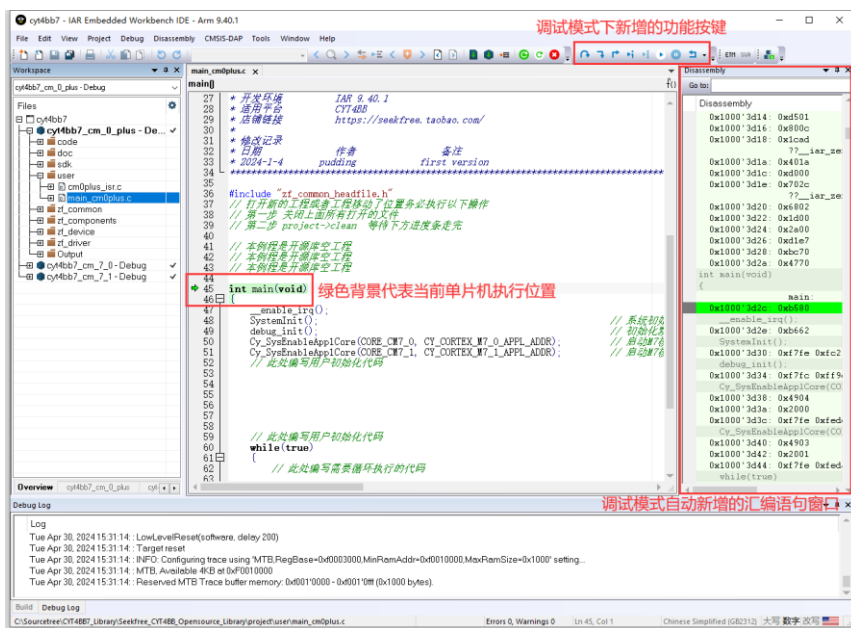
2.3.2 如何进入调试模式

一般使用调试模式时都是选择“下载并调试”的方式，当工程编译无报错并且正确使用调试器连接电脑和单片机后，可以点击“Download and Debug”按钮，也可以使用快捷键 Ctrl+D 以进入调试界面。



2.3.3 调试界面说明

以 CYT4BB7 的 M0 核心调试界面作为演示：



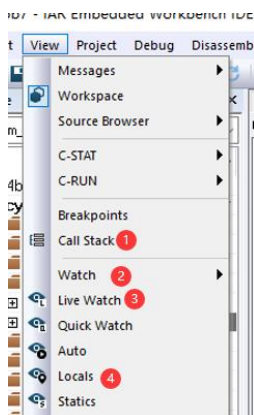
新增的汇编语句窗口一般并不会关心，但 IAR 默认会打开这个窗口，所以忽略不管即可，新增或者修改的功能按键释义如下：



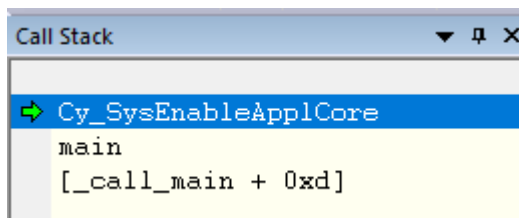
1.重新编译、下载、调试，适合在调试下找到问题并且修改程序后一键重新调试。

- 2.重新进入调试，一般不会使用。
- 3.关闭本次调试。
- 4.跳过当前语句，执行完绿色背景覆盖的语句并在下一句时暂停。
- 5.进入执行，可理解为下一步，若是函数则跳转到函数内逐句执行。
- 6.跳出执行，可理解为跳出循环、函数。
- 7.下一个语句，执行当前这段以分号结尾的语句。
- 8.运行程序，直接运行当前程序段，若无断点则相当于单片机自己跑程序，若遇到断点则暂停。
- 9.暂停执行，使单片机停止执行程序内容，并且跳转到停止位置。
- 10.复位单片机。

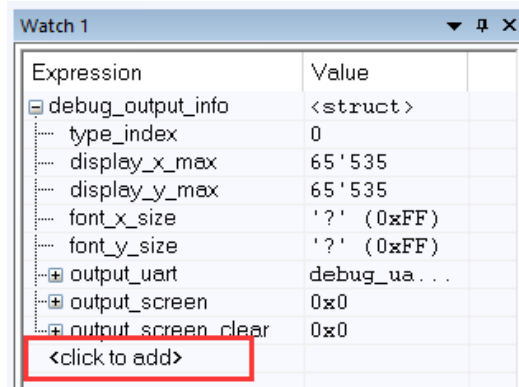
以上按键为调试功能需要掌握的基本操作，可以控制单片机的程序执行，配合各种界面查看窗口即可分析程序中存在的各种 bug，一般经常使用的窗口如下：



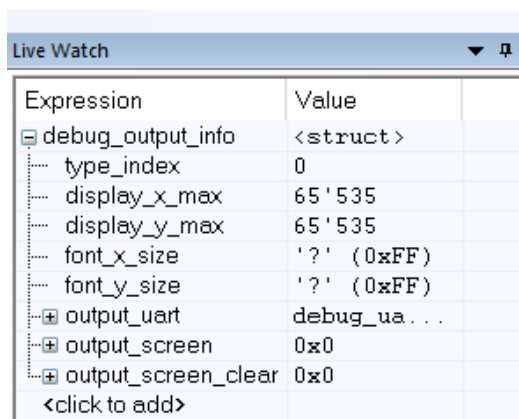
1.call stack 窗口，一般叫调用栈或者程序调用栈，可以查看单片机的程序嵌套调用情况，双击对应行可以跳转到嵌套进入位置，常用于单片机莫名跑飞或者卡死的情况下，跳出问题之前执行的程序内容来分析问题原因。



2. watch 窗口，用于查看表达式、变量数据，将变量复制后粘贴到 watch 窗口即可，变量数据需要手动刷新，比如暂停后或者主动点击变量数据。



3. live watch 窗口，与 watch 窗口使用方式一致，但 live watch 窗口的变量会自动刷新，单片机运行时若修改变量值，live watch 窗口将同步更新，比 watch 窗口更加方便，但也会占用单片机性能，看个人喜好使用。

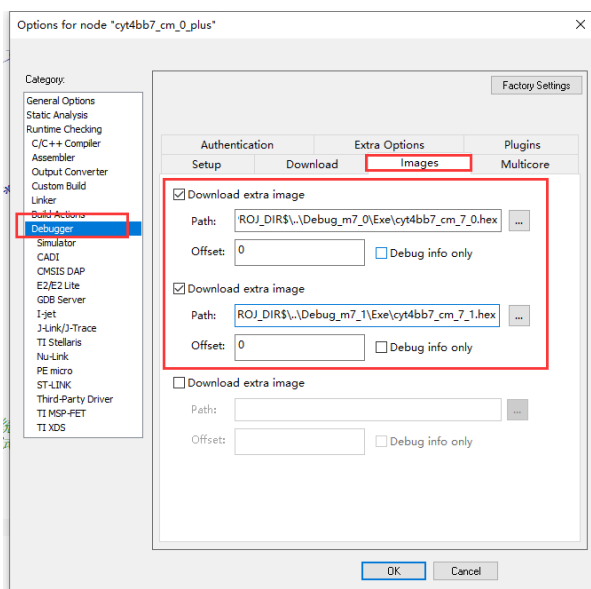


4. Locals 窗口，用于查看当前函数下的临时变量数据，不能自行添加。

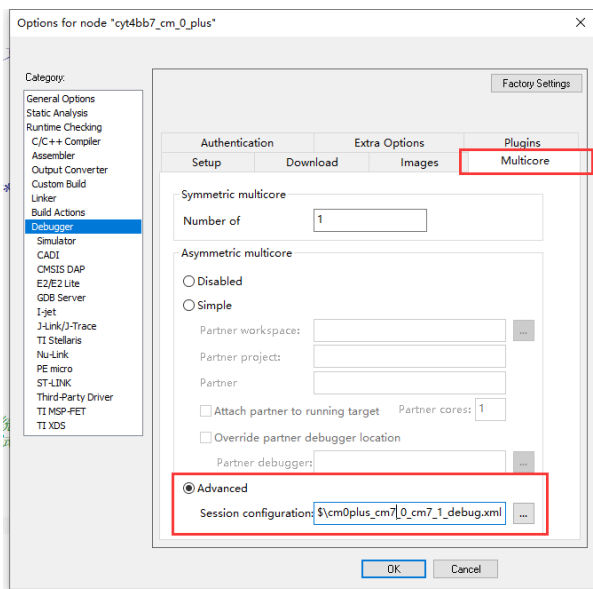
| Locals | |
|-------------------|--------------|
| Variable | Value |
| core | '\0' (0x00) |
| vectorTableOffset | 268'959'744 |
| cmStatus | <unavail...> |
| interruptState | <unavail...> |
| tPwrCtl0 | <union> |
| tPwrCtl1 | <union> |

2.3.4 多核同步调试说明

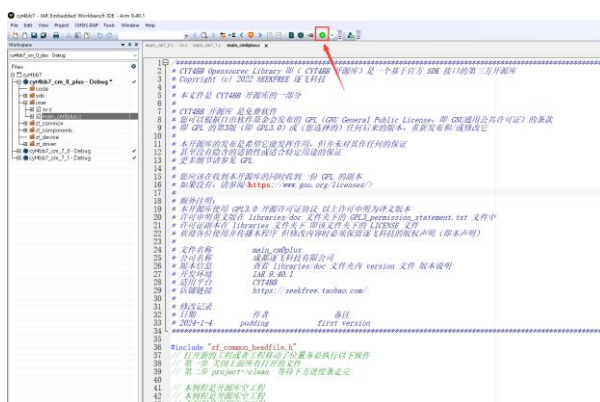
对于多核心的单片机来说, IAR 提供了支持多核同步调试的方法, 但需要用户自己提前将三个核心编译完成, 以 CYT4BB7 芯片为例, 打开 M0 的工程设置, 找到 Debugger-Images 窗口, 分别勾选两个 M7 核心的 hex 镜像, 如果没有路径可以自行添加。



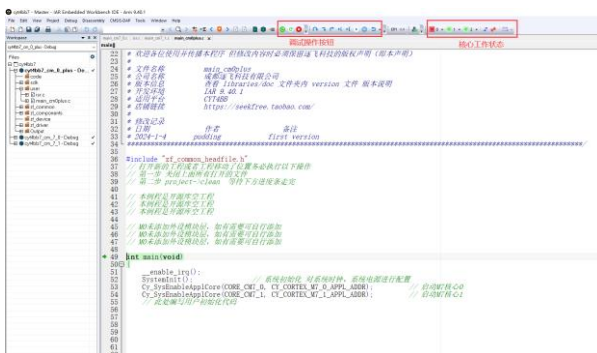
在 Images 旁边的 Multicore 窗口选择 Advanced, 这步的作用是通过加载我们编写好的 xml 文件内容, 从而同步启动另外两个工程。



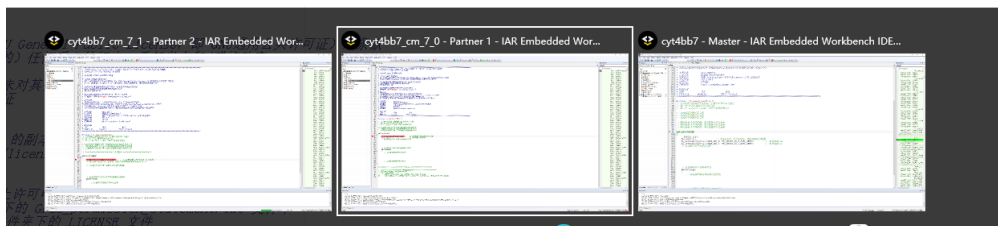
配置完成后点击调试按钮，稍加等待即可进入三核同步调试的界面



进入调试后可以看到单核心的调试按钮以及所有核心的工作状态，切换 IAR 界面可以调试不同的核心。



除了多核下载以及调试外，单核操作与普通 ARM 芯片完全一致，并且 CYT4BB7 的每个核心都支持单独的下载和调试



第3章 CYT4BB7 开发注意事项

1. CYT4BB7 开源库地址：https://gitee.com/seekfree/CYT4BB7_Library
2. CYT4BB7 为多核芯片，其中包括三个核心：M0、M7_0、M7_1，因此在 IAR 中有三个工程分别对应其三个核心，在第一次使用时为了保证三个核心程序统一，需要对每一个核心的工程都编译下载一遍。使用时如果不确定其他核心的程序是否和当前工程一致，也可以尝试先给三个核心依次下载程序。
3. CYT4BB7 的外设功能是共用的，相当于一个身体里面长了三个脑子，在多核使用时要注意不要多个核心同时控制一个外设，这样大概率会导致外设工作异常。
4. CYT4BB7 的 M0 核心运行频率为 80Mhz，两个 M7 核心运行频率为 250Mhz，如果不习惯多核使用，仅一颗 M7 核心的性能也是完全足够的。
5. CYT4BB7 的 M0 核心不支持硬件浮点单元，因此不能用于快速傅里叶变换(FFT)的计算，可以使用任意 M7 核心计算 FFT。
6. CYT4BB7 的 M0 核心将对 M7 核心的工作频率、供电电压进行配置，如果硬件的最小系统电路的供电部分与 M0 的配置内容不一样，可能导致 M7 核心无法运行，推荐使用逐飞科技的核心板配套逐飞科技 CYT4BB7 开源库使用。
7. 在对 CYT4BB7 的某一个核心进行“仅下载”后，需要单击复位按键或者重新上电后才会运行程序。
8. 新买的 CYT4BB7 核心板上电后会有闪灯效果，其内部程序为 M0 的全 IO 功能测试程序，用于核心板出售前对所有引脚以及引脚功能的测试，此测试程序会关闭两个 M7 核心，如果新买的 CYT4BB7 核心板下载 M7 核心程序后发现不运行并且闪灯，可以查看第 2 条注意事项。
9. CYT4BB7 手册中描述 flash 可擦写次数为 1000 次，虽然并未实际测试是否下载 1000 次就无法正常下载，但建议在下载程序之前多加考虑，避免频繁下载程序的情况发生。

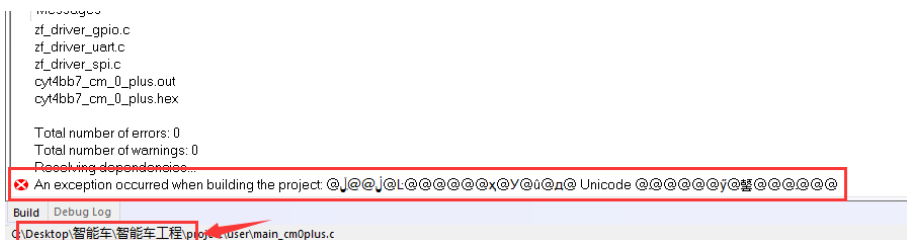
第4章 编译、下载、调试报错及解决办法汇总

4.1 编译报错

4.1.1 An exception occurred when building the project:

@J@J@L@@@@@x@Y@u@ Unicode @@@@@y@
@@@_@@@

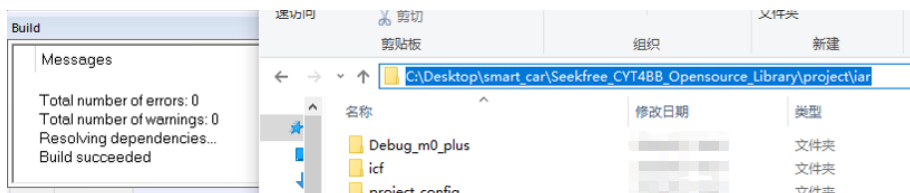
编译结束后的一堆@符号的乱码报错：



报错原因：当前工程的文件路径有中文字符，以上图为例，其中“智能车\智能车工程”这段中文字符在 IAR9.40.1 中是无法正常编码的，因此会给出一个乱码的报错提示。

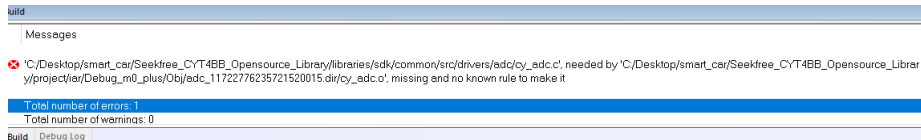
解决办法：修改工程路径，去除中文即可。

正确演示：



4.1.2 '....drivers/adc/cy_adc.c', needed by '....cy_adc.o',Missing and no known rule to make it

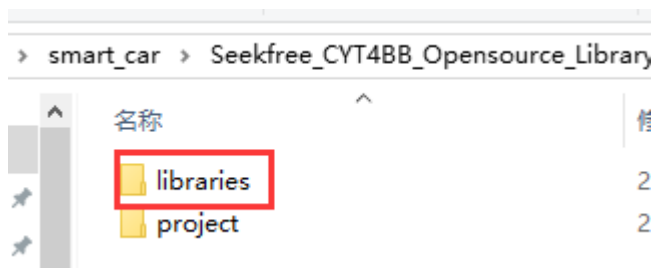
编译时直接对一个 adc 的文件报错：



报错原因：移动了工程文件夹但是并没有把 libraries 文件夹一起移动，导致编译时找不到任何底层库文件。

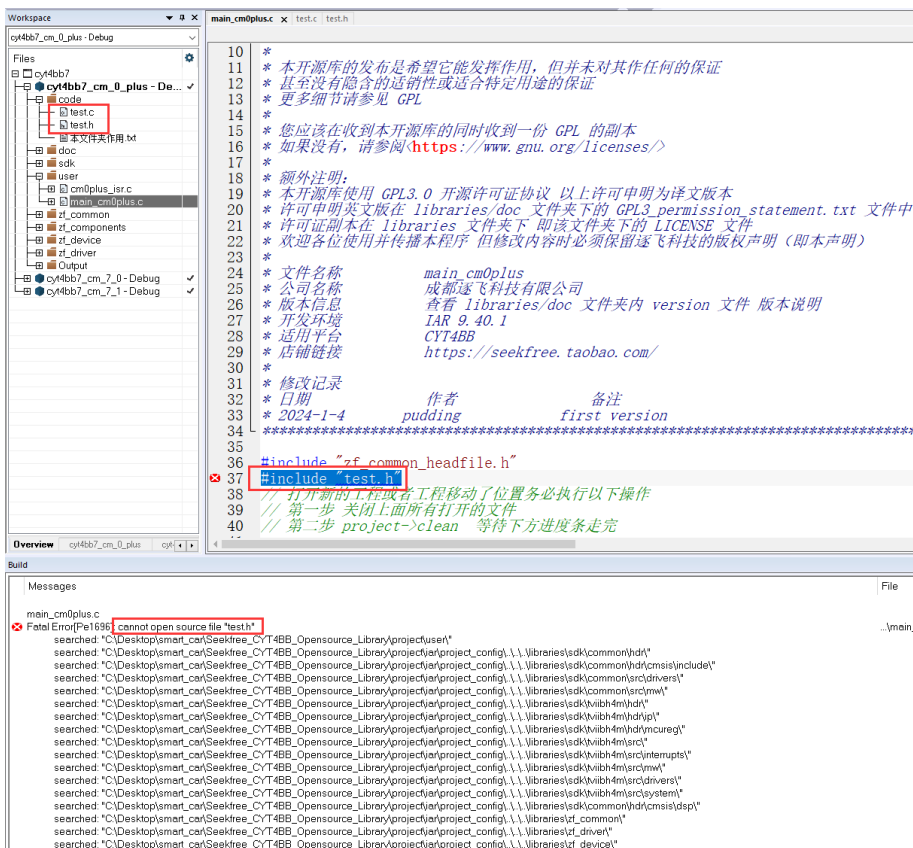
解决办法：将 libraries 文件夹放置在工程文件夹同路径下。

正确演示：



4.1.3 Cannot open source file “xxxx.h”

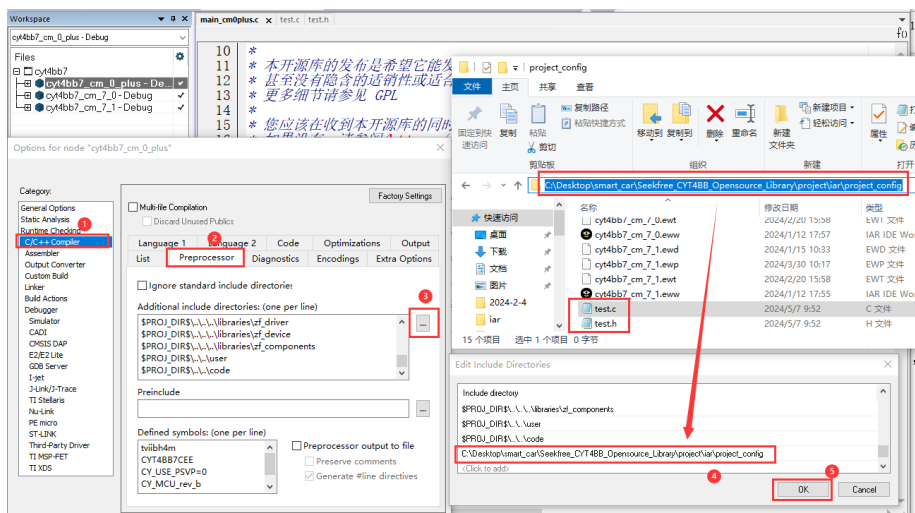
在使用 IAR 开发时，明明添加了某个头文件到 code 文件夹，但使用 include 包含时却提示找不到这个文件：



报错原因: 文件实际路径并不在当前工程的编译路径下，导致编译器找不到当前包含的文件。

解决办法:

1. 增加工程编译路径，将需要包含的文件的路径添加到工程编译路径中：



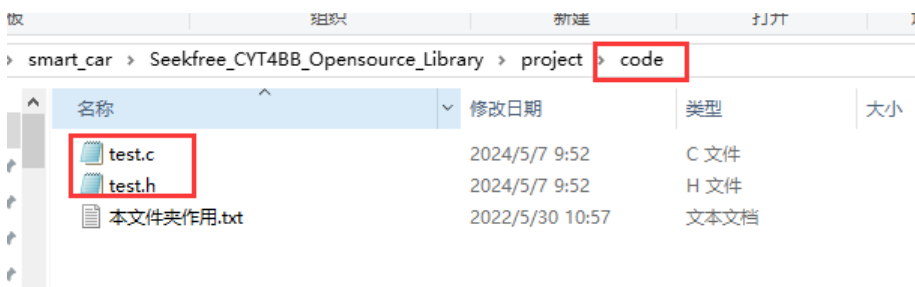
```

zf_driver_timer.c
zf_driver_pwm.c
zf_driver_uart.c
zf_driver_spi.c
cyt4bb7_cm_0_plus.out
cyt4bb7_cm_0_plus.hex
    
```

```

Total number of errors: 0
Total number of warnings: 0
Resolving dependencies...
Build succeeded
    
```

- 将新添加的文件放到工程路径下的 code 文件夹内，该文件夹已经被添加到编译路径，因此可以直接编译：



```
test.c
cyt4bb7_cm_0_plus.out
cyt4bb7_cm_0_plus.hex

Total number of errors: 0
Total number of warnings: 0
Resolving dependencies...
Build succeeded
```

4.1.4 Error[Lt039]: Incompatible architecture attributes

当使用 M0 核心的工程调用 DSP 的 FFT 函数时出现的静态库报错：

```
cyt4bb7_cm_0_plus - Debug
Reading project nodes...

main_cm0plus.c
cyt4bb7_cm_0_plus.out
❌ Error[Lt039]: Incompatible architecture attributes
   Command-line specifies ARM v6S-M
   cm0plus_isr.o and 107 other objects are compiled for ARM v6S-M
   arm_cfft_f32.o(iar_cortexM7lf_math.a) and 7 other objects in iar_cortexM7lf_math
❌ Error[Lt006]: Incompatible object(s): arm_cfft_f32.o(iar_cortexM7lf_math.a) and 5 other
v6S-M)
❌ Error[Lt006]: Incompatible object(s): arm_cfft_f32.o(iar_cortexM7lf_math.a) and 7 other
⚠ Warning[Lt049]: Potential floating point parameter passing conflict. The first module be
   AAPCS, base variant: main_cm0plus.o
   AAPCS, VFP variant: arm_sin_f32.o(iar_cortexM7lf_math.a)

Total number of errors: 3
Total number of warnings: 1
Resolving dependencies...
Build failed
```

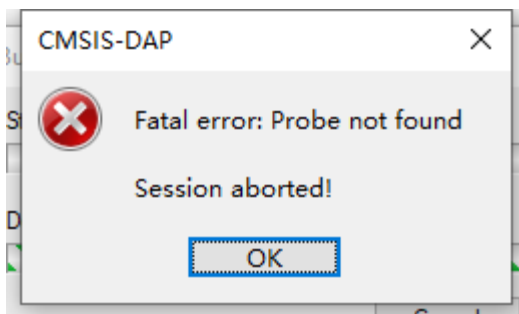
报错原因：M0 核心并不支持硬件浮点运算单元，因此也不支持 DSP 的数学库运算，报错提示内容是静态库编译方式与 M0 编译方式不同，因此也无法在 M0 中正常使用静态库的方式调用函数。

解决办法：不在 M0 核心工程使用任何调用静态库的函数。

4.2 下载及调试报错

4.2.1 Fatal error: Probe not found

下载或者调试时，弹出 DAP 调试器报错：



报错原因：未正确检测到 DAP 调试下载器，可能有以下几种情况：

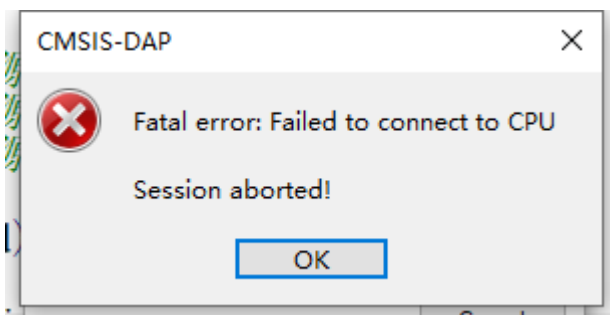
- 1) 数据线接触不良导致无法与 DAP 通讯
- 2) DAP 的 USB 驱动异常(设备管理器显示未知设备)，需要重新安装驱动
- 3) DAP 损坏导致无法正常通讯
- 4) 核心板短路导致 DAP 进入保护模式，从而无法与 IAR 正常通讯

解决办法：

- 1) 拔插数据线
- 2) 更换数据线
- 3) 更换调试下载器
- 4) 更换单片机

4.2.2 Fatal error: Failed to connect to CPU

下载或者调试时，弹出 DAP 调试器报错：未连接到 CPU



报错原因：无法连接到 CPU，可能有以下几种情况：

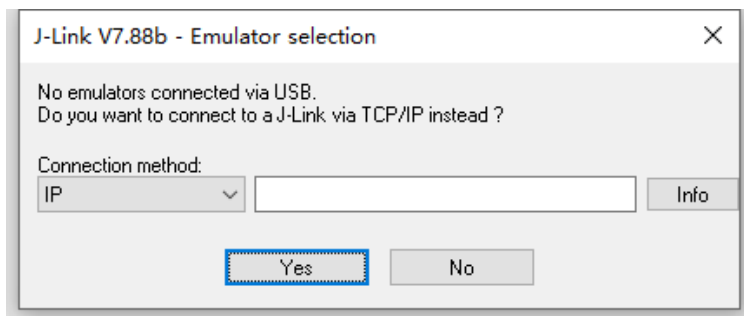
- 1) DAP 与单片机之间的连接有接触不良，导致无法正常与单片机通讯
- 2) 芯片下载接口损坏导致无法正常下载程序
- 3) 芯片最小系统电路工作异常导致芯片无法正常工作

解决办法：

- 1) 拔插灰排线
- 2) 更换灰排线
- 3) 更换核心板

4.2.3 No emulators connected via USB, Do you want to connect to a J-Link via TCP/IP instead ?

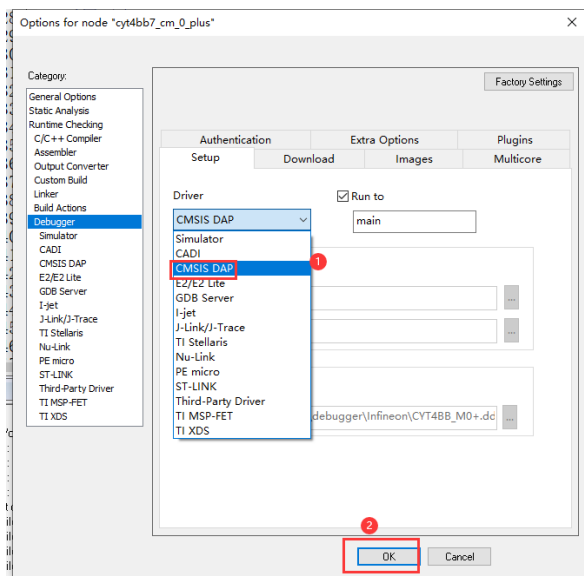
检测不到 J-link 下载器：



报错原因：未正确连接 J-link 调试下载器或者下载器选择错误

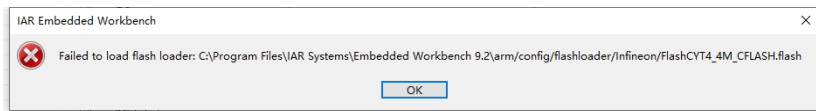
解决办法：

- 1) 正确连接 J-link 调试下载器
- 2) 如果使用的是 DAP 调试下载器，那么需要在工程设置中修改下载器的选择：



4.2.4 Failed to load flash loader: caprogram filesIAR Systems Embedded Workbench 9.2\arm/config/flashloader/infineon/flashCYT4 4M CFLASH.flash

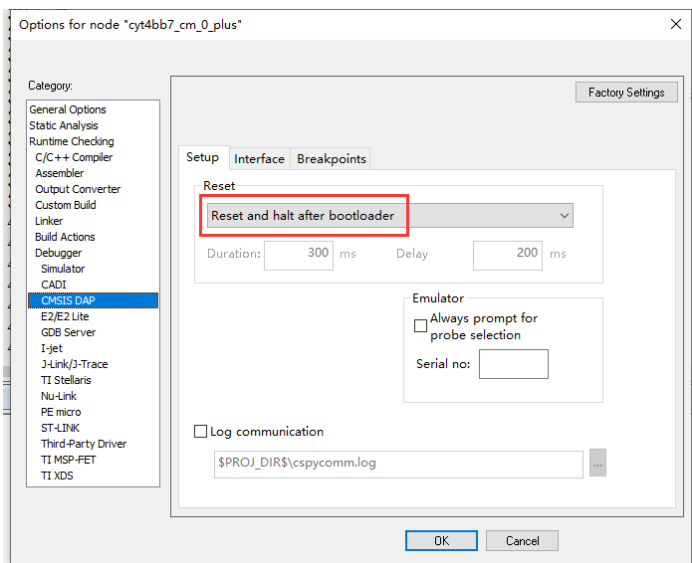
下载或者调试时提示加载 flash 失败



报错原因：单片机 flash 工作异常导致的无法正常向 flash 写入程序

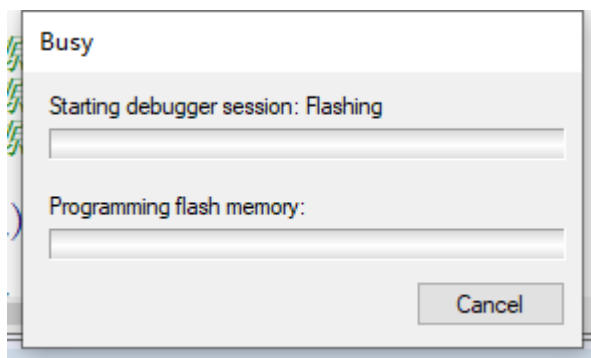
解决办法：

- 1) 核心板断电后重试
- 2) 更换调试下载器后重试
- 3) 更换下载配置为“Reset and halt after bootloader”后重试



- 4) 更换芯片后重试

4.2.5 下载或者调试在加载界面卡死

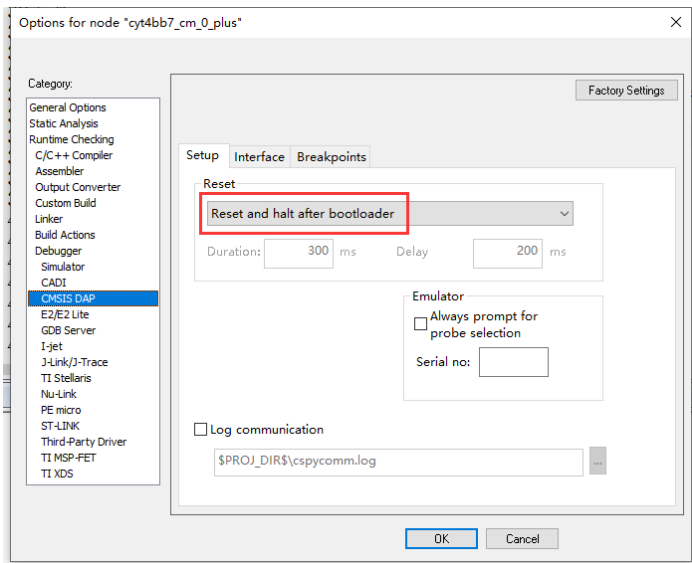


报错原因：单片机未正常执行下载流程导致下载中途卡死

解决办法:

卡死时 IAR 是无法正常操作的，需要拔掉下载器的数据线才能解除卡死状态，如果拔掉数据线依然没有解除卡死，那么可以直接通过任务管理器强制关闭 IAR

1) 更换下载配置为“Reset and halt after bootloader”后重试



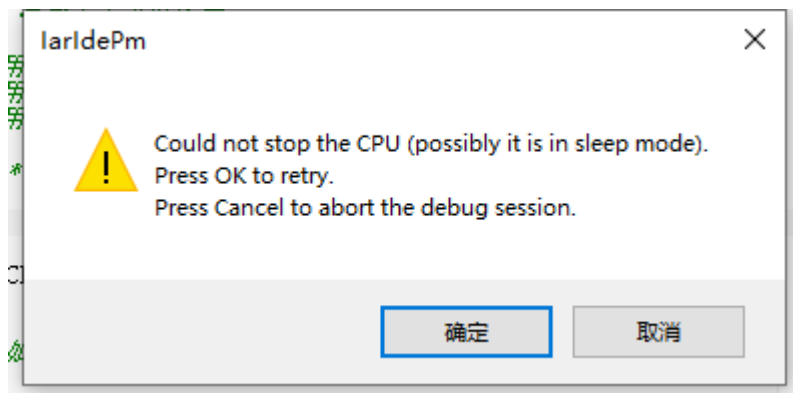
2) 核心板断电后重试

3) 更换调试下载器后重试

4) 更换芯片后重试

4.2.6 Could not stop the CPU (possibly it is in sleep mode). Press OK to retry

调试时弹出无法暂停 CPU 的界面:



报错原因：CPU 在调试时断开或者工作异常(比如供电、晶振)，导致无法继续调试

解决办法：

- 1) 检查调试器与单片机之间的灰排线是否接触不良
- 2) 重新进入 debug 调试

第5章 问题反馈注意事项及途径

使用逐飞科技发布的开源库以及配套核心板、主板等模块时，如果遇到自己不能解决的问题，可以先尝试阅读对应模块的使用手册并且结合配套例程测试，若还是无法解决则可以尝试联系逐飞科技淘宝店铺客服或者 QQ 交流群中的管理员，软件问题则联系软件技术支持，硬件问题则联系硬件技术支持。

问题反馈注意事项：

- 问题表达需要足够清晰，准确描述问题现象
- 涉及到程序则将程序部分完整截图，涉及到硬件则将接线部分完整拍照或者直接录制问题视频
- 不要一次性联系多位技术支持，只需要联系其中一位即可，技术支持会尽量协助解决问题。