Application Note
ZHCAFM3 – July 2025

合适的 UCD3138 系列数字电源控制器的 CCS 配置

Bliss Zhou

摘要

UCD3138 系列产品是 TI 的高性能数字电源控制器产品。在使用 UCD3138 系列控制器时需要合适的 CCS 配置。当用户重命名一个 UCD3138 工程名称，升级 CCS 版本或者更换一台新的电脑，即便之前能很好工作的 Firmware 工程也可能出现不可预测的问题。而这些问题一般都是由于不恰当的 CCS 配置导致。本应用说明介绍了这些问题清单，同时提供了解决方案。

1 目录

1 不合适的 CCS 配置导致的问题清单 1
2 合适的配置避免这些问题 2
2.1 避免重命名工程后编译失败 2
2.2 避免由于升级 CCS 或者将工程移植到另外一台电脑后编译失败 3
2.3 避免设备锁死 6
2.4 解决没有.x0 文件生成的问题 9
2.5 解决编译成功后没有 .pp 文件的问题 9
3 参考文献 11

1 不合适的 CCS 配置导致的问题清单

- 重命名工程后编译失败
- 升级 CCS 版本或者将工程移植到另外一台电脑后编译失败
- 设备锁住，一个新的 image 无法再次烧录到 UCD
- 编译完成后没有生成 .x0 目标文件

- 编译完成后没有生成 .pp 文件

2 合适的配置避免这些问题

2.1 避免重命名工程后编译失败

通常，用户不会直接使用 TI EVM 代码工程名称作为自己的 CCS 项目名称。 用户通常需要一个自定义的工程名称。 但是，当用户重命名 CCS 工程名后可能遇到一些编译错误。 图 2-1 显示了常见的错误信息。

图 2-1: “No rule to make target” 错误

重命名工程名将使 *Linked Recourse* 自动更改为名为“*copy_PARENT*“的默认路径。但这是一个用户的本地计算机上不存在的路径。 一个 *invalid location* 错误会出现在 *project properties -> Resource -> Linked resource -> Linked resource,* 如图 2-2 所示。

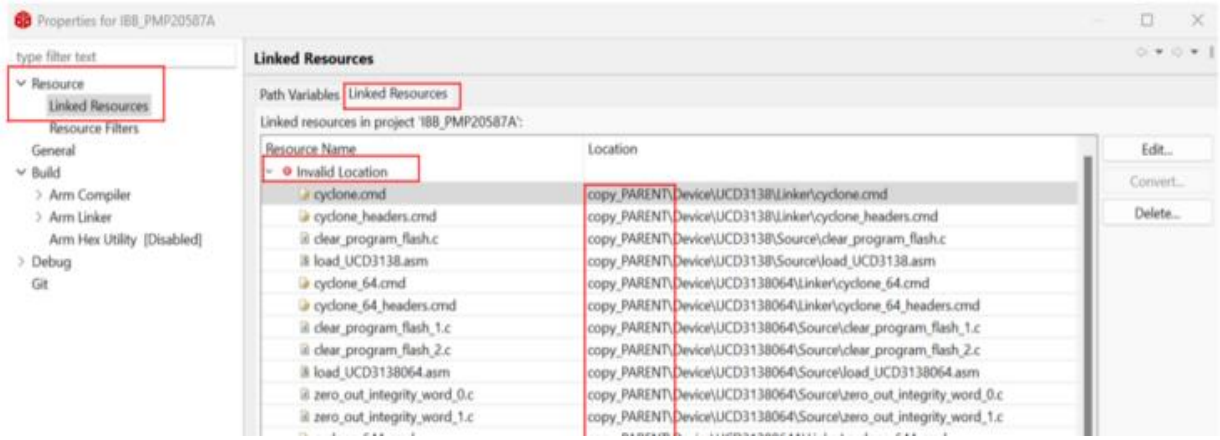

图 2-2: Invalid location for linked Resources

用户需要做的是将所有的 *“copy_PARENT\”* 使用一个 *“PROJECT_LOC\..\..\”* 进行替代以解决这个错误。一个正确的路径设置如图 2-3。

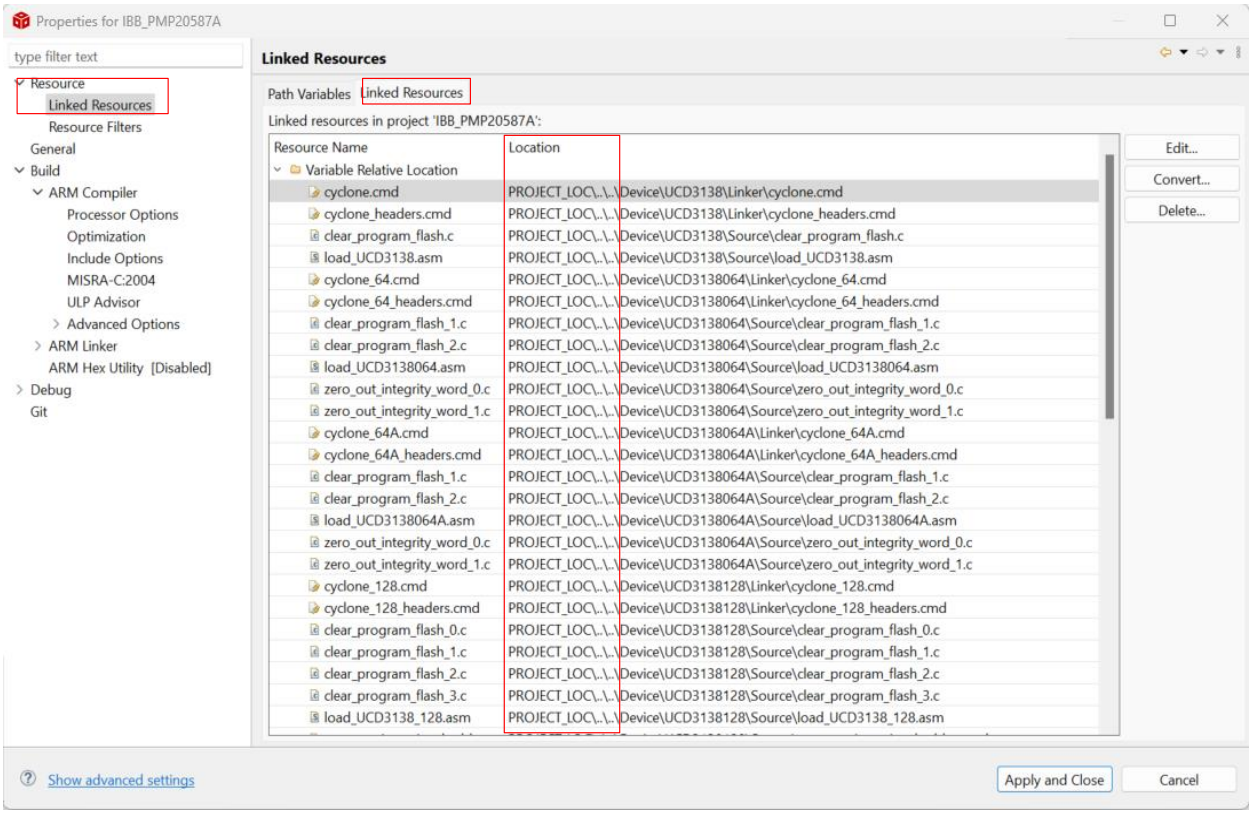

图 2-3: Valid location for linked Resources

2.2 避免由于升级 CCS 或者将工程移植到另外一台电脑后编译失败

将 CCS 升级到新版本或设置新计算机后，先前编译成功的工程可能也会遇到编译失败。常见如图 2-4 错误提示 *"#1559: Interrupt handlers must be compiled in ARM mode"* ， 并指向于文件 *'standard_interrupt.c'*， *'software_interrupt.c'* 和 *'interrupts.c'*。此错误的根本原因是中断服务函数必须在 ARM 模式(而非 Thumb 模式) 下进行编译。 升级 CCS 版本时，设置为 ARM 模式的原始配置可能不会延续，而被自动改为 Thumb 模式，从而导致此问题（请注意，.C 文件通常以 Thumb 模式编译，以减小代码大小）。

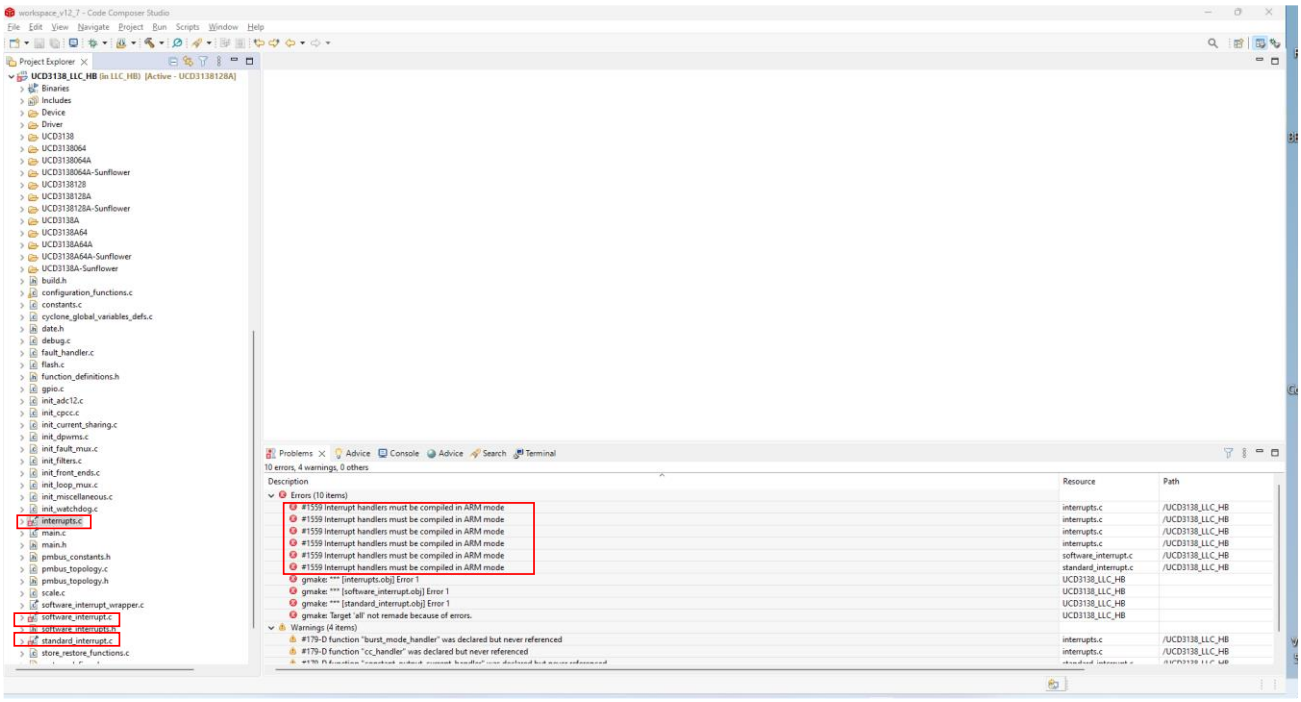

图 2-4: Interrupt handlers must be compiled in ARM mode

这里通常有两处理方式：

1, 设置 *'standard_interrupt.c'*，*'software_interrupt.c'* 和 *'interrupts.c'* 整个文件在 ARM 模式下编译。 比如配置整个 *'software_interrupt.c'* 在 ARM 模式下编译，如图 2-5 所示,遵循以下步骤以完成配置。点击 *project explorer pane -> 右击 'standard_interrupt.c' -> click properties -> Build -> ARM Compiler -> Processor Options.* 然后选择“32”。

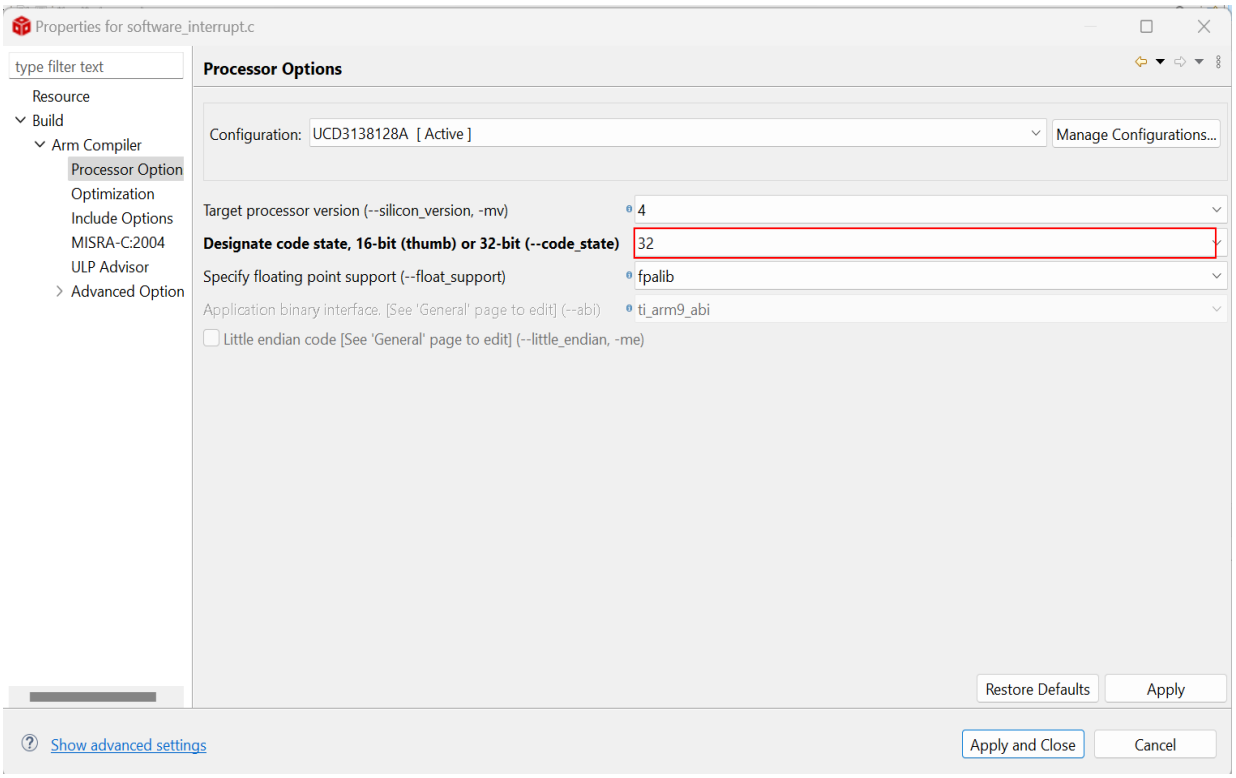

图 2-5: Setting the Designate code state to ARM 32-bit Mode for Interrupt Handler Compilation

2, 放置一个 *"#pragma CODE_STATE (function_name,32)"* 在中断函数之前以只设置此函数为 ARM 模式编译。例如在 *'standard_interrupt.c'* 中， *'void standard_interrupt(void)'* 必须在 ARM 模式编译。 使用 *"#pragma CODE_STATE (standard_interrupt, 32)"* 让 *'standard_interrupt(void)'* 在 ARM 模式编译。如 snippet1 所示。

```
#pragma INTERRUPT (standard_interrupt, IRQ)
#pragma CODE_STATE (standard_interrupt, 32)
void standard_interrupt(void)
{
      handle_standard_interrupt_global_tasks();
      switch (supply_state)
```

snippet 1: Enforce 'standard_interrupt(void)' to be built in ARM mode

2.3 避免设备锁死

同样，用户也可能在升级 CCS 版本后遭遇设备锁死无法再次烧录 image 的问题。一个 *"Device does not appear to be in ROM mode after ENABLE_ROM_command"* 的错误提示如图 2-6 所示。

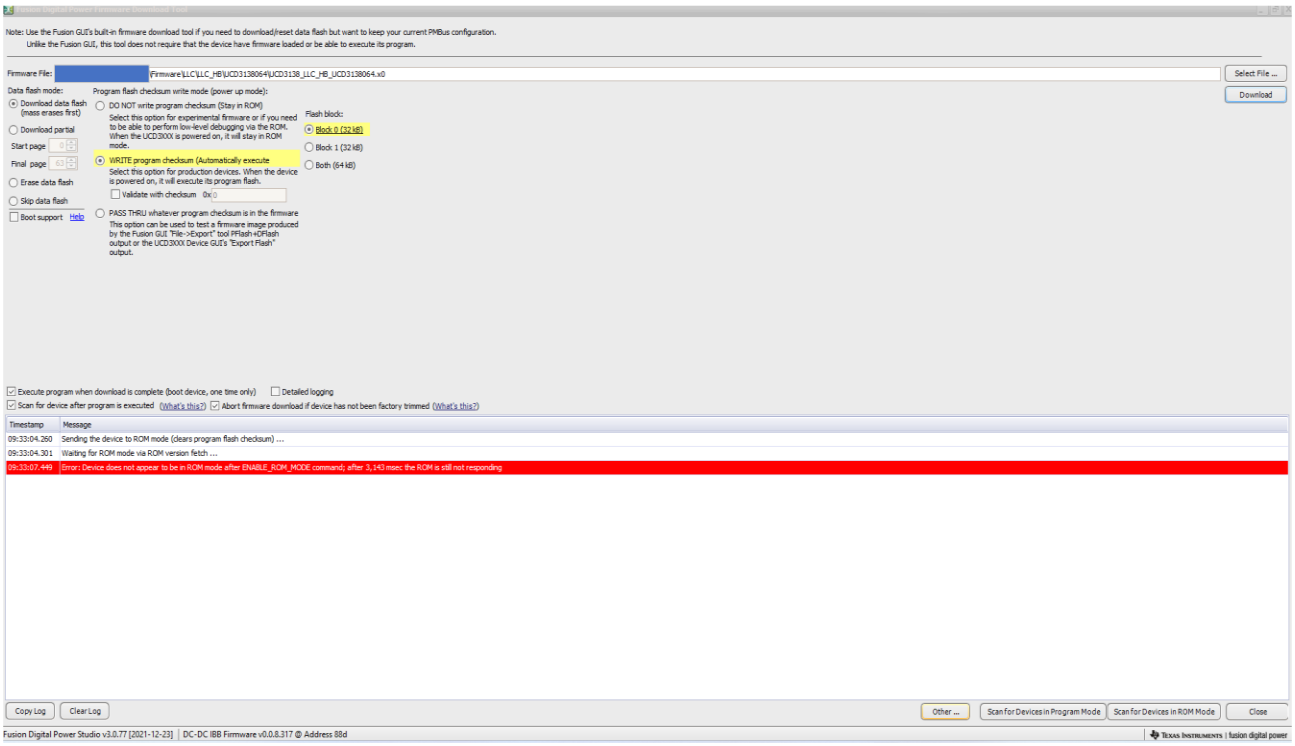

图 2-6: Device lockup

应用手册 Lockup Avoidance for UCD3138 Devices 对关于设备锁死原因和处理方法的进行了详细的分析。除了此应用手册中提到的原因，这里补充两个由于编译器版本导致的潜在陷阱。

- 第一个潜在原因是用户正在使用一个不兼容的编译器版本。UCD 的 EVM code 都是基于编译器版本 TI V5.2.2~V5.2.5 开发的。一部分 EVM code 并不能和其他编译器版本（如 V5.2.9） 很好的工作。如果用户遭遇芯片锁死现象，请先检查是否正在使用一个兼容的编译器版本。如果不是，一个兼容的编译器被要求重新安装。安装步骤如下：

  Step1：点击 *Help -> Install Code Generation Compiler Tools-> TI Complier Updates* 。

  Step2: 点击取消 *show only the latest versions of available software -> ARM Compiler Tools 5.2.5 -> Next* ，如图 2-7 所示。

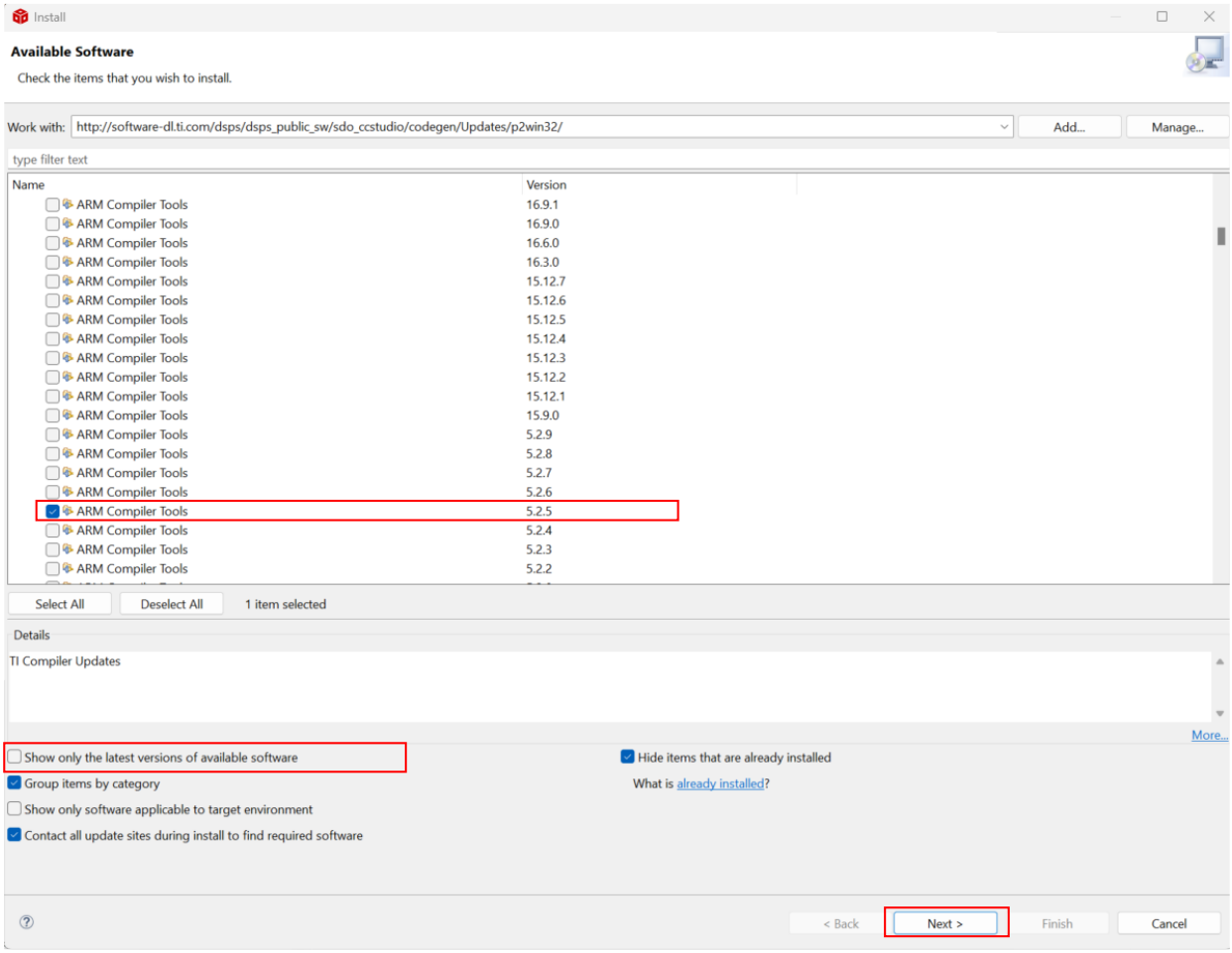

图 2-7: Install a compatible compiler version

Step3: 完成安装后重启电脑。

Step4: 切换编译器到 v5.2.5 如图 2-8 所示。

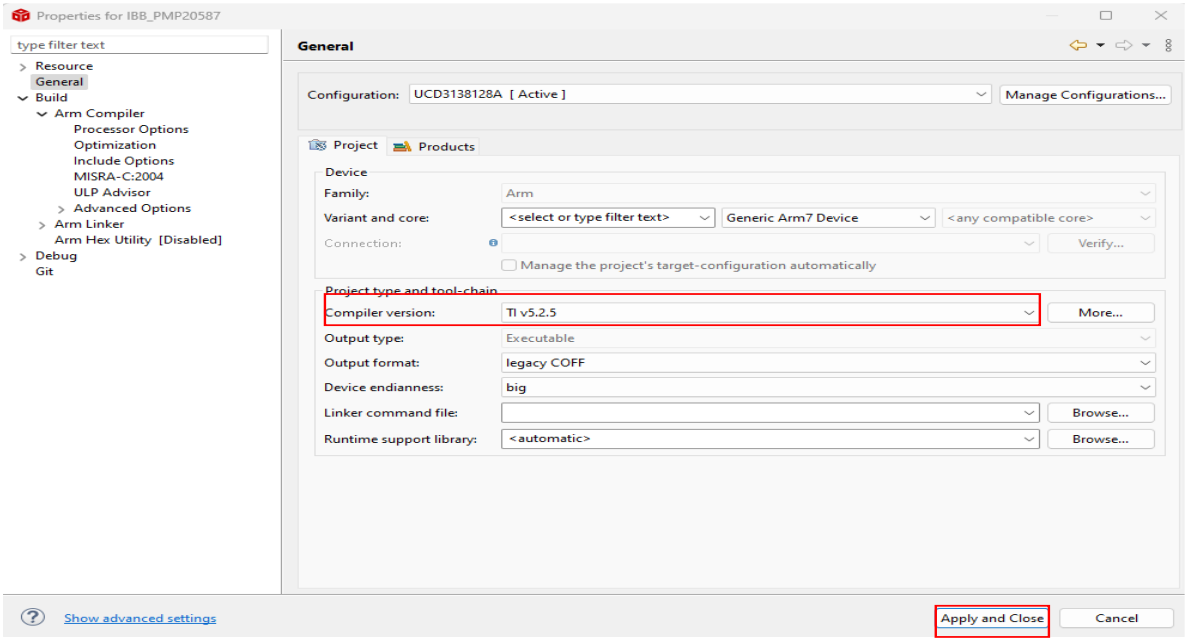

图 2-8: Switch to the compatible compiler version

- 另外一个潜在原因可能是文件 *'software_interrupt.c'* 被设置了一个太激进的 *code optimization level* (如 *"2-Global Optimizations"*) 导致 *checksum* 不能被成功擦除。使用 *"2-Global Optimizations"*，事实上一部分 UCD3138 系列设备是可以成功擦除 checksum 的。但一部分 UCD3138 系列设备的 *checksum* 却不能成功擦除。为了解决此问题， *'software_interrupt.c'* 被要求设置为一个较低的 *optimization level* 如 *"1-Local Optimizations"* 。设置步骤如图 2-9 所示。点击 *project explorer pane -> 右键单击 'software_interrupt.c' -> click properties -> Build -> ARM Compiler -> Optimization-> 1-Local Optimizations* 。

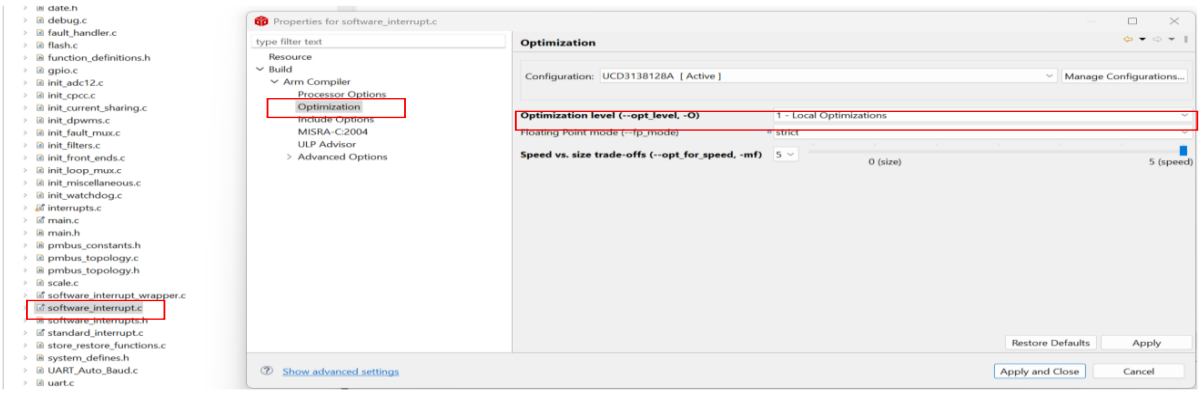

图 2-9: - Select 1-Local Optimizations

2.4 解决没有.x0 文件生成的问题

UCD 设备下载的 image 一般是基于.x0 的格式。但有时用户在成功编译工程后无法找到.x0 文件。 通常在编译完成后会直接生成的是一个.out 文件。要生成.x0 文件需要执行一个 post-build 命令行 *"${CG_TOOL_HEX}.exe" -x "${BuildArtifactFileName}" -o "${BuildArtifactFileBaseName}.x0" -memwidth 8* 。如图 2-10 所示, 检查此命令行是否成功设置。

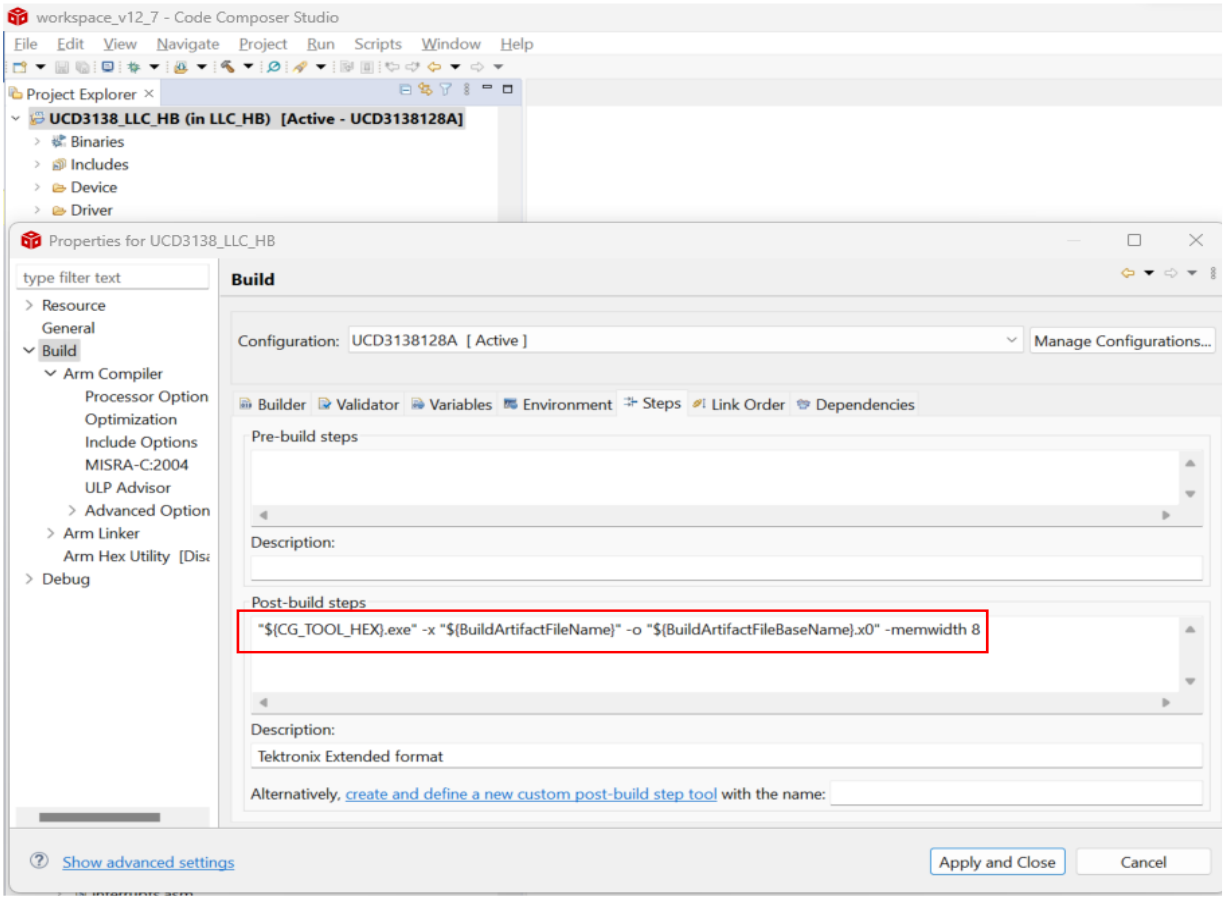

图 2-10: Post-build step for converting the .out file to a.x0 file

2.5 解决编译成功后没有 .pp 文件的问题

如果无法在工程文件中找到.pp 文件请按以下步骤完成配置。

- 正确的 *preprocessor* 文件路径。 点击 *project explorer pane -> 右击 project name -> properties -> build -> Arm Compiler-> Advanced Options -> Directory specifier->* 设置正确的工程路径 *"${PROJECT_ROOT}/UCD3138128A"* 到 *Preprocessor file directory.* 图 2-11 是一个 UCD3138128A 的例子。

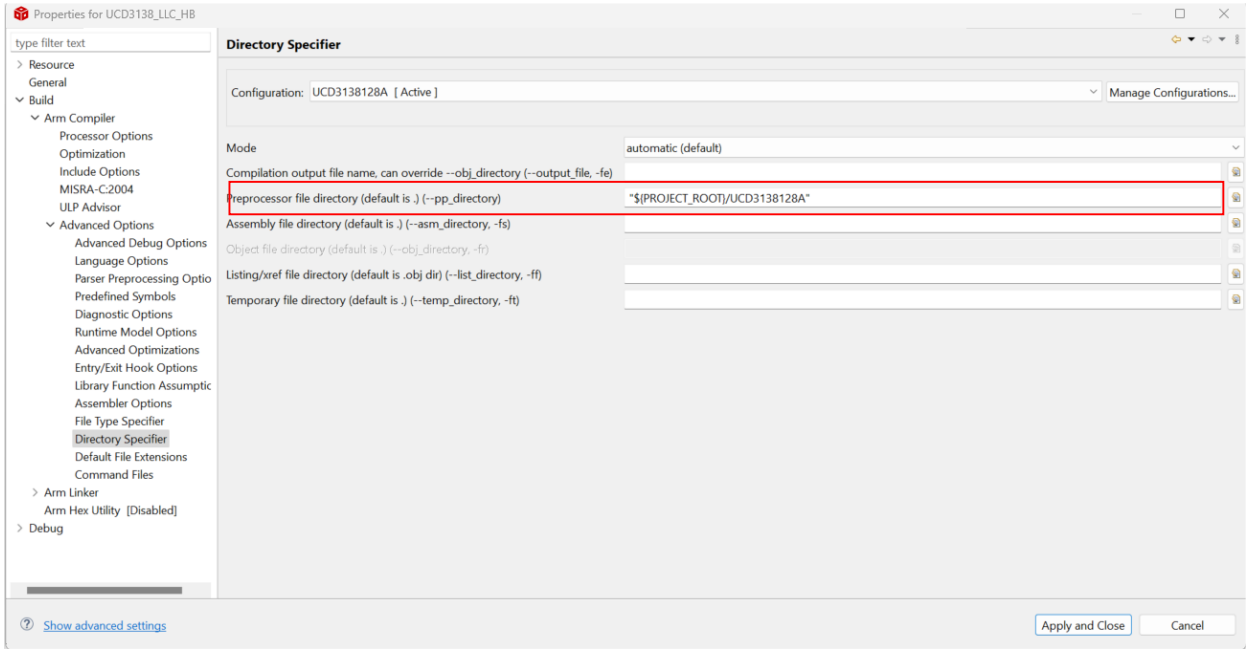

图 2-11: correct project path for Preprocessor file directory

- 正确的 Pre-define symbols。 点击 *project explorer pane -> 右击 project name -> properties -> build -> ARM Linker-> Advanced Options -> Command File Preprocessing -> Pre-define..,* 如图 2-12 设置 UCD3138128A=1。

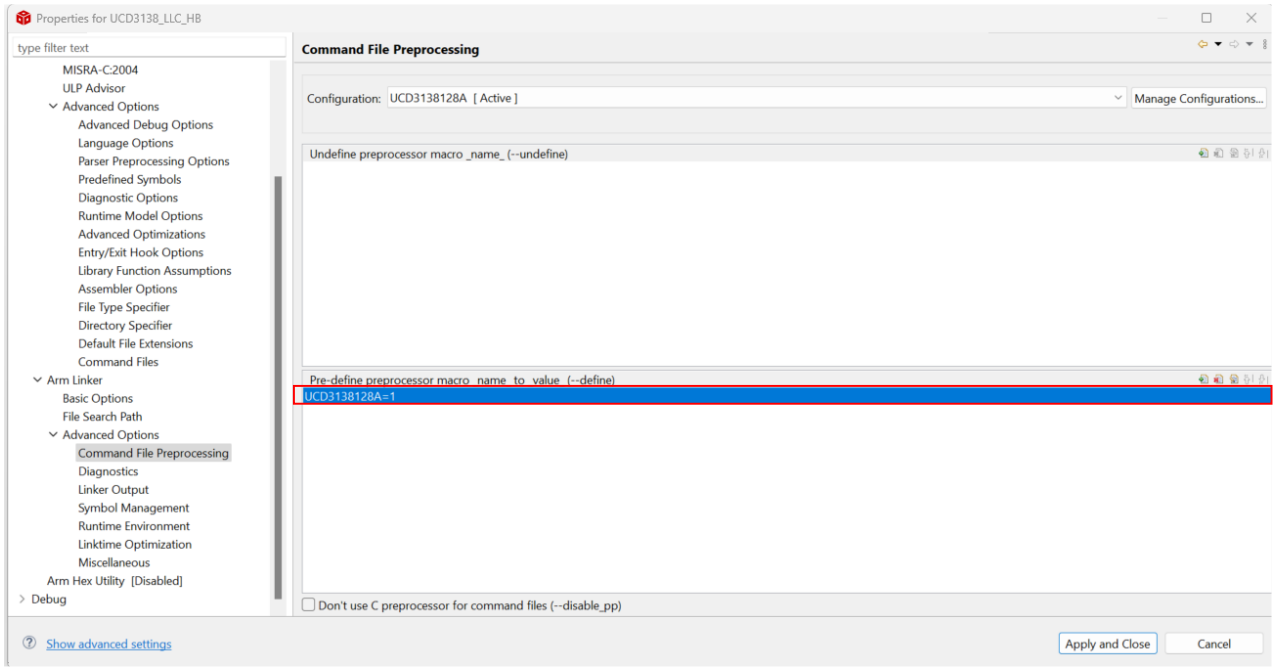

图 2-12: correct Pre-define preprocessor for Arm Linker

- *点击 explorer pane -> 右击 the project name -> properties ->Build -> Advanced Options -> Advanced Options -> Predefined Symbols -> Pre-define NAME,* 如图 2-13 设置 UCD3138128A=1。

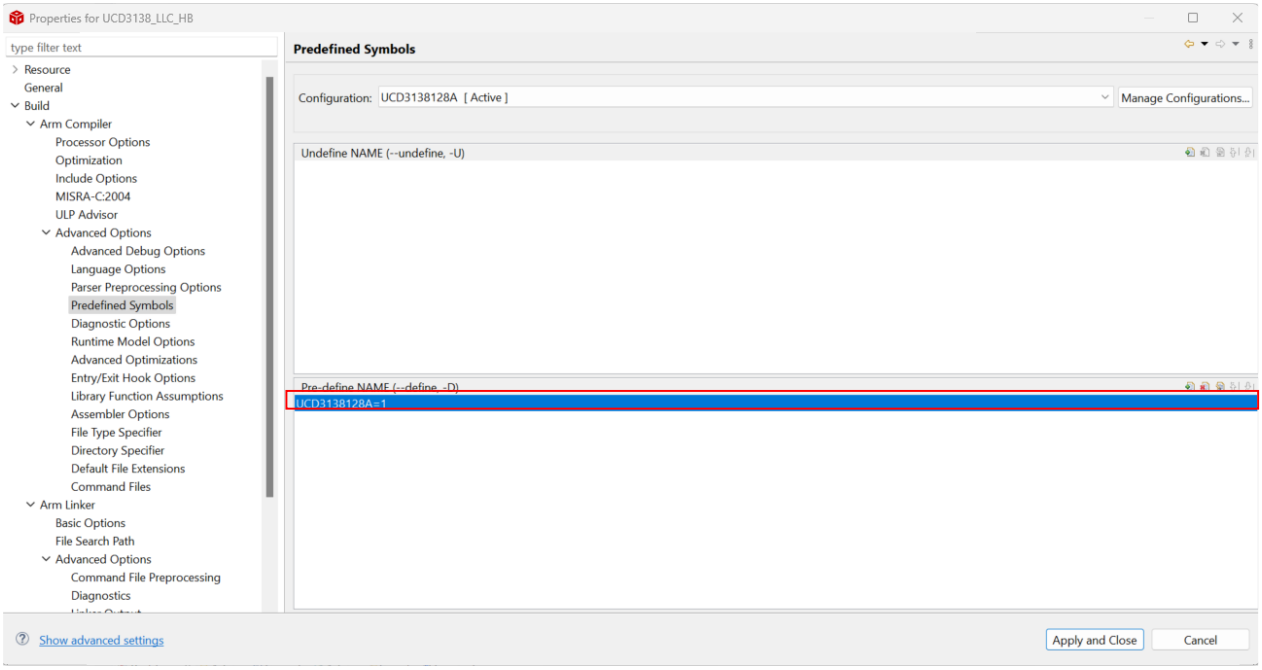

图 2-13: correct Pre-define preprocessor for Build

3 参考文献

- UCD31xx Technical Reference Manual (Rev. E)
- Lockup Avoidance for UCD3138 Devices

重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做出任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。 严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 TI 的销售条款或 ti.com 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
版权所有 © 2025，德州仪器 (TI) 公司