Application Note

C2000 IDE Assist 迁移工具指南

Shashank Madineni, Nima Eskandari, Delaney Woodward, and Aishwarya Rajesh

摘要

C2000 IDE Assist（也称为 C2000 IDEA）是一款集成开发工具，旨在增强德州仪器 (TI) C2000™ 微控制器的开发。这在 Visual Studio Code™ 和 Code Composer Studio™ 中提供了一个集中的环境，提供项目检测、定向配套资料交付和开发人员效率工具等功能。关键功能之一是迁移支持，它可帮助开发人员在不同的 C2000 器件之间高效转换应用代码。

本应用手册将指导用户完成迁移过程的每个步骤，从检测和设置项目到执行迁移检查和解决兼容性问题。本文档介绍了常见迁移问题的快速修复、用于分析代码差异的迁移报告以及重要限制和注意事项，以确保在器件架构之间平稳可靠过渡。

内容

1 简介....3
1.1 入门....3
2 概述....4
3 迁移支持功能....5
3.1 项目检测....6
3.2 “设定迁移设置”页面....6
3.3 迁移执行....8
3.4 快速修复....12
3.5 迁移报告....15
3.6 位域迁移....16
4 总结....17
5 参考资料....17

插图清单

图 2-1. “C2000 IDEA - 功能”面板....4
图 2-2. 任意版本间的器件 Driverlib 迁移 HTML 页面....4
图 3-1. C2000 IDEA - 迁移流程....5
图 3-2. C2000 IDEA - 项目检测....6
图 3-3. “迁移设置设定”页面....7
图 3-4. 手动更新项目的当前器件....8
图 3-5. 文件迁移执行（扩展树和编辑器视图）....9
图 3-6. 文件迁移进度条和完成通知....9
图 3-7. 对当前文件启用持续迁移检查....10
图 3-8. 对当前文件禁用持续迁移检查....10
图 3-9. 项目迁移执行（扩展树）....11
图 3-10. 项目迁移进度条和完成通知....12
图 3-11. 项目迁移后的 CCS 输出控制台....12
图 3-12. 迁移执行后的文件视图....12
图 3-13. 迁移问题 — 查看问题和快速修复....13
图 3-14. 迁移配套资料 HTML 页面....13
图 3-15. 打包 #IFDEF 快速修复（F28x 至 F28x）....14
图 3-16. 打包 #IFDEF 快速修复（F28x 至 F29x）....14
图 3-17. 所有枚举打包 #IFDEF 快速修复（F28x 至 F29x）....14

图 3-18. 导出迁移报告（扩展树）.....15
图 3-19. 迁移报告.....16

表格清单

表 3-1. 对 C2000 器件的迁移支持.....5
表 3-2. 对各种文件格式的迁移支持.....5

商标

C2000™ and Code Composer Studio™ are trademarks of Texas Instruments.
Visual Studio Code™ is a trademark of Microsoft Corporation.
所有商标均为其各自所有者的财产。

1 简介

德州仪器 (TI) C2000 软件开发套件 (SDK) 提供了一套完整的软件和文档，旨在加速实时控制应用程序的开发。这些资源包括器件专用驱动程序、库和外设示例、可帮助开发人员简化项目。这些 SDK 旨在适应经验水平各异的用户，并会提供适合初学者的功能以指导首次使用的用户。SDK 中包含的 C2000 SysConfig 工具通过直观的可视化界面简化了器件和外设的复杂性，从而完善了设置过程。为了进一步改善初始化、运行时和调试的用户体验，C2000 IDEA 将工具和配套资料集中在单一环境中，以实现更高效的工作流程。

本文档介绍了如何开始使用 C2000 IDE 辅助迁移功能并有效地利用该功能来简化和加速 C2000 器件之间的迁移。

1.1 入门

本部分介绍了 C2000 IDEA 工具的基础知识，例如如何设置开发环境以使用扩展。

- 下载并安装 Code Composer Studio (CCS) 版本 20.0.0 或更高版本的最新离线安装程序。
- 启动 CCS 并导航至位于左侧边栏中的*扩展*选项卡。
- 搜索并安装 C2000 IDEA 扩展。
- 重新启动 CCS 以启用扩展。用户现在可以在左侧边栏中看到 C2000 IDEA 图标。
- 在 CCS 中正常打开新的或现有工作区。

有关详细的设置说明和其他指导，请参阅 C2000 IDE Assist 工具功能指南应用手册。

2 概述

此部分概述了 C2000 IDEA 界面并介绍了如何轻松浏览该工具。位于屏幕左侧的“C2000 IDEA - 功能”面板可用于访问该工具的所有功能。下图使用一个 C2000WARE 示例来展示此面板，以帮助用户熟悉布局和选项。

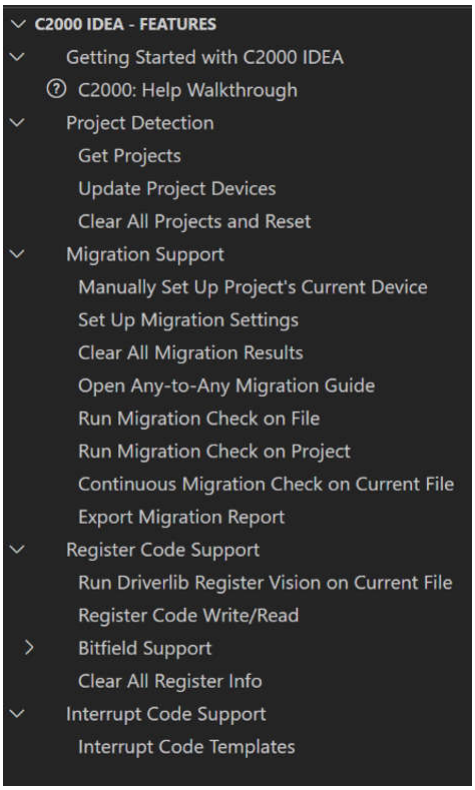

图 2-1. “C2000 IDEA - 功能”面板

在此面板中，有一个专门的*迁移支持*部分可让您快速轻松地访问该工具提供的所有与迁移相关的功能。用户可以参阅*打开任意版本间的迁移指南* HTML 页面（如图 2-2 所示），并为当前器件和目标器件选择合适的 C2000ware 版本。这样，用户便能够查看 driverlib 代码差异并探索不同 C2000 器件之间的一对一映射。

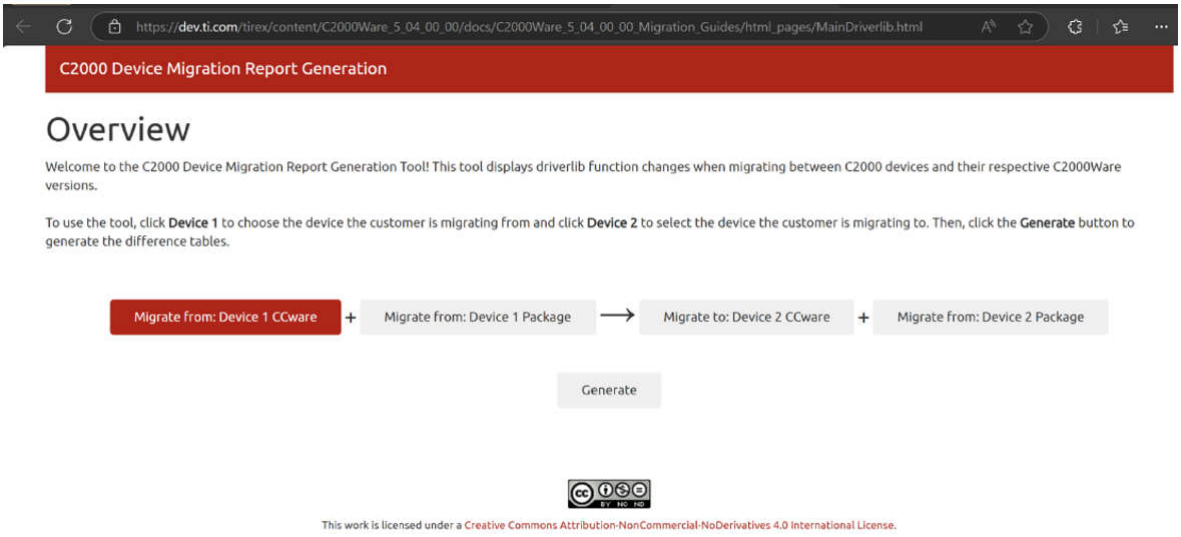

图 2-2. 任意版本间的器件 Driverlib 迁移 HTML 页面

3 迁移支持功能

C2000 IDEA 工具通过自动识别和突出显示 driverlib 代码的更改（包括添加、删除或修改的应用外设接口 (API)、寄存器、字段、枚举和宏）来简化 C2000 器件之间的代码迁移。该自动化功能可减少人工操作，更大程度地降低错误风险，并验证迁移过程是否顺利高效。该自动化功能支持针对完整项目和独立 driverlib 文件进行迁移，可灵活地满足各种用户需求。对于特定器件，该工具还支持位域代码迁移。C2000 IDEA 通过检测架构差异并提供有针对性的建议以实现更可靠的过渡，简化了 F28x 至 F28x 以及 F28x 至 F29x 器件之间的迁移。表 3-1 提供了 C2000 器件之间迁移支持的清晰摘要，帮助用户快速评估 Driverlib 和位域实现方案的兼容性和可用支持。

表 3-1. 对 C2000 器件的迁移支持

器件系列	Driverlib 迁移支持	位域迁移支持
F29H85x	是	否
F28P55x	是	否
F28P65x	是	否
F28002x	是	否
F28004x	是	否
F28003x	是	否
F280013x	是	是（目标）
F280015x	是	否
F2838x	是	否
F2837x	是	否
F2807x	是	否
F2803x	否	是（来源）

此工具支持特定文件格式的源文件，如表 3-2 中所示。

表 3-2. 对各种文件格式的迁移支持

源文件格式	迁移支持
*.c	是
*.h	是
*.asm	否
*.cmd	否
*.lib	否
链接器文件	否

图 3-1 概述了迁移独立文件或整个项目所需的关键步骤，可确保实现无缝高效的迁移过程

图 3-1. C2000 IDEA - 迁移流程

3.1 项目检测

只需点击*获取项目*元素即可检测到所有使用 C2000 器件的项目。通过此步骤，扩展可以识别每个项目的器件并跟踪每个文件所属的项目。项目检测实时可实现扩展的完整功能，例如跨器件的 Driverlib 迁移。某些扩展功能仅在项目检测（或默认器件设置）时可用，而其他功能可以通过要求输入当前器件来在单个文件上运行。

如要检测项目，请按照以下步骤操作：

1. 如要检测工作区中的所有项目、请使用以下两个选项之一：
   a. 输入 CTRL+SHIFT+P，并点击 *C2000：Get Projects*。
   b. 点击扩展树的 *C2000 IDEA - 功能*窗格中的*项目检测* > *获取项目*。
2. 运行后，扩展检测到的工作区中的所有项目都会出现在扩展树的 *C2000 IDEA - 项目*窗格中。查看项目并验证器件型号和当前器件详细信息是否符合预期。

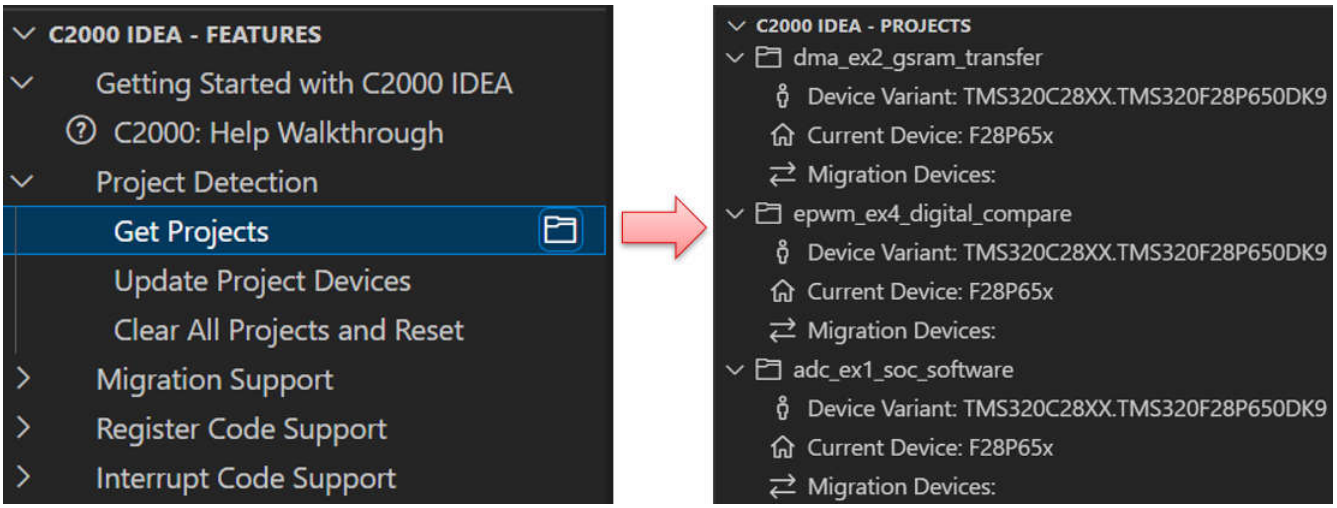

图 3-2. C2000 IDEA - 项目检测

3.2 “设定迁移设置”页面

此页面提供了一个用于跨整个项目自定义迁移过程的综合界面，便于用户根据特定要求配置迁移设置。

要访问此页面，用户可以使用以下方法之一：

1. 输入 CTRL+SHIFT+P，并点击 *C2000：Set Up Migration Settings*。
2. 扩展树方法：
   a. 导航至 *C2000 IDEA - 功能* 窗格中的“迁移支持”>“设定迁移设置”。
   b. 单击 *C2000 IDEA - 项目* 窗格中检测到的项目下的迁移器件旁边的图标，如图 3-3 所示。

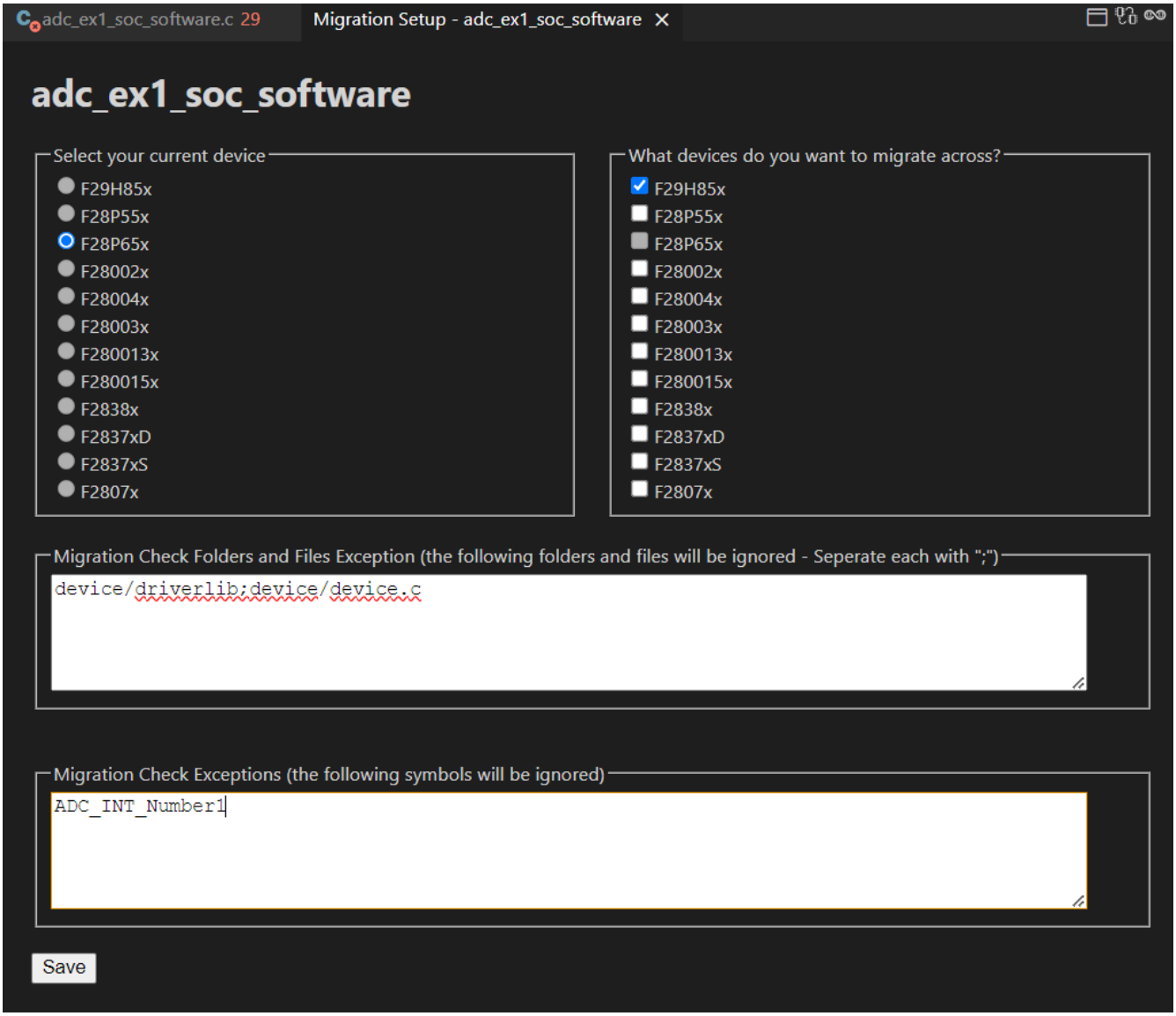

图 3-3. “迁移设置设定”页面

“迁移设置”页面会根据在上一步中完成的项目检测流程自动检测并识别当前器件。由于器件是预先确定的，因此用户不能直接在“迁移设置”页面中进行修改。

如果需要更改项目的当前器件，则用户可以通过以下方法进行更改：

1. 输入 CTRL+SHIFT+P，并点击 *C2000：手动设置项目的当前器件*。
2. 单击 *C2000 IDEA - 项目* 窗格中检测到的项目下的*当前器件* 旁边的图标。

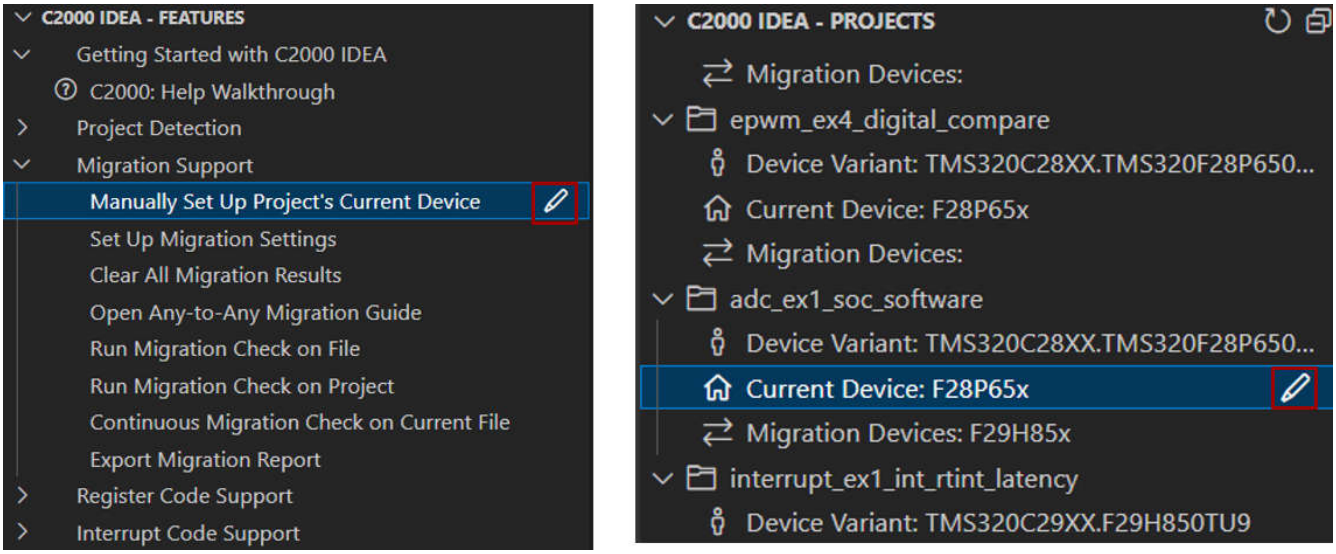

图 3-4. 手动更新项目的当前器件

在此页面中，用户可以通过选择器件并指定文件、文件夹和代码更改的例外情况来自定义迁移过程。

- **选择迁移器件**：
  - 用户可以通过选中“迁移设置”页面中的选择框来选择一个或多个迁移器件。
- **排除文件和文件夹**：
  - 在*迁移文件*和*文件夹例外*部分中指定迁移期间要忽略的文件和文件夹。
  - 各条目必须用分号 (;) 分隔。
- **排除代码更改**：
  - 在*迁移检查例外*部分中指定要在迁移期间忽略的代码修改。
  - 各条目必须用分号 (;) 分隔。

输入所需的详细信息后，保存迁移设置页面。在扩展树中的“C2000 IDEA - 项目”窗格中相应地更新迁移器件信息。

3.3 迁移执行

C2000 IDEA 扩展可用于在使用 driverlib 样式代码编写的项目或文件上运行 F28x 到 F28x 或 F28x 到 F29x 迁移检查。这种代码样式的特点是使用对 driverlib 源文件中定义的函数的调用（如：Device_init ()) 和/或包含 _o_ 语法的寄存器访问。

3.3.1 对独立文件运行迁移检查

借助此扩展，用户能够对一个独立文件执行迁移检查。要执行迁移检查，请完成以下步骤：

- 打开独立文件，并确保在编辑器中将该文件设置为活动文件。
- 使用以下方法之一运行迁移检查：
  1. 输入 CTRL+SHIFT+P，并点击 *C2000：对文件运行迁移检查*。
  2. 扩展树方法：
     a. 导航至 *C2000 IDEA - 功能* 窗格中的“迁移支持”> *对文件运行迁移检查*。
     b. 点击活动文件编辑器右上方的图标，如图 3-5 所示。

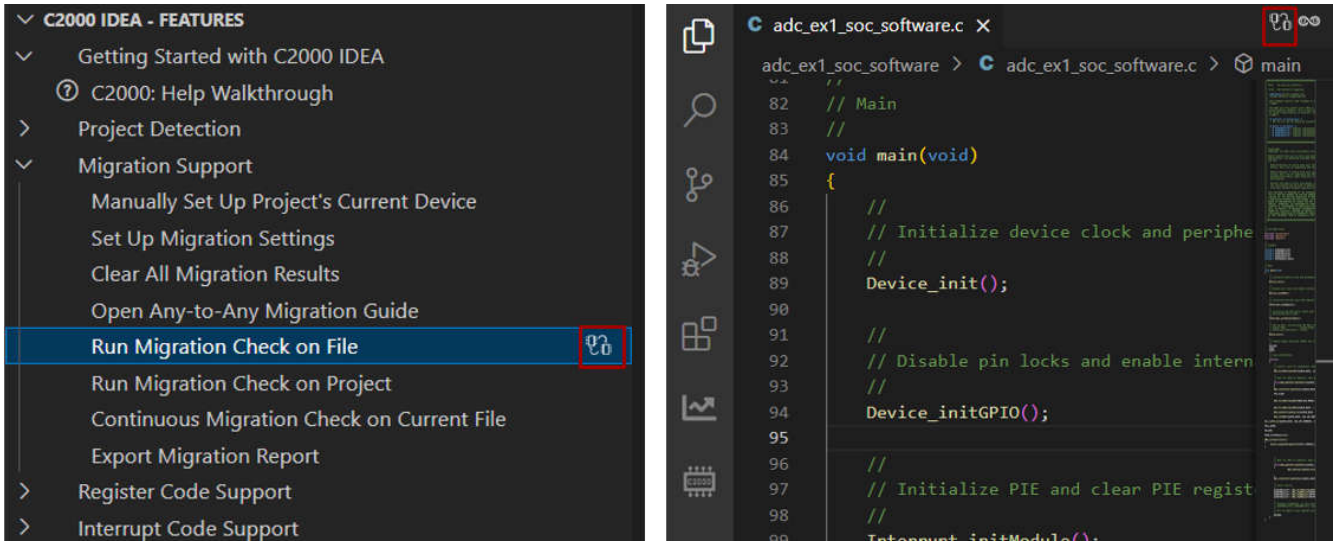

图 3-5. 文件迁移执行（扩展树和编辑器视图）

对活动文件运行迁移检查后，屏幕的右下角会显示一个进度条。此进度条提供实时更新，显示当前正在处理的文件路径。迁移过程完成后，进度条将在同一位置替换为一条状态消息，指示*已完成迁移检查*。

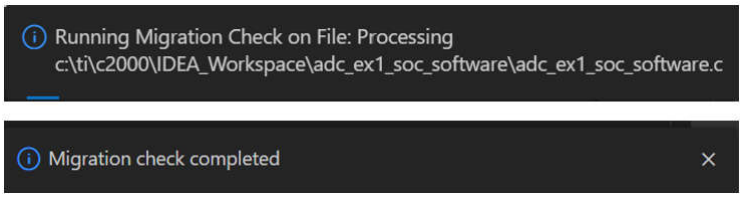

图 3-6. 文件迁移进度条和完成通知

对当前文件运行持续迁移检查：每当检测到更改时，此功能会定期自动对活动文件运行迁移检查。这可确保用户在每次修改后无需手动重新执行检查即可迅速识别和解决迁移问题，从而提高效率并改进工作流程。

备注

当该工具中启用了 *C2000：对当前文件运行持续迁移检查*功能时，请勿使用任何迁移功能。

- 用户可以使用以下方法之一对活动文件启用持续迁移检查：
  1. 输入 CTRL+SHIFT+P，并点击 *C2000：对当前文件启用持续迁移检查*。
  2. 扩展树方法：
     a. 导航至 *C2000 IDEA - 功能* 窗格中的“迁移支持”> *对当前文件运行持续迁移检查*。
     b. 点击活动文件编辑器右上角的图标。

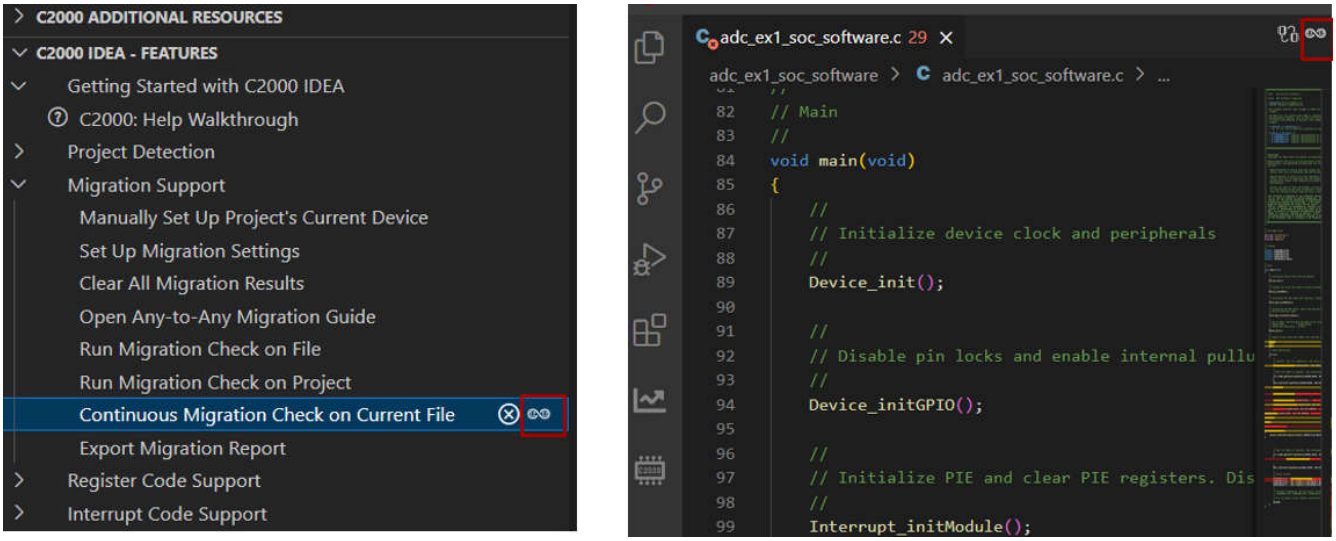

图 3-7. 对当前文件启用持续迁移检查

- 用户可以使用以下方法之一对活动文件禁用持续迁移检查：
  1. 输入 CTRL+SHIFT+P，并点击 *C2000：对当前文件禁用持续迁移检查。*
  2. 扩展树方法：
     a. 导航至 *C2000 IDEA - 功能* 窗格中的“迁移支持”> *对当前文件运行持续迁移检查。*
     b. 点击活动文件编辑器右上角的图标。

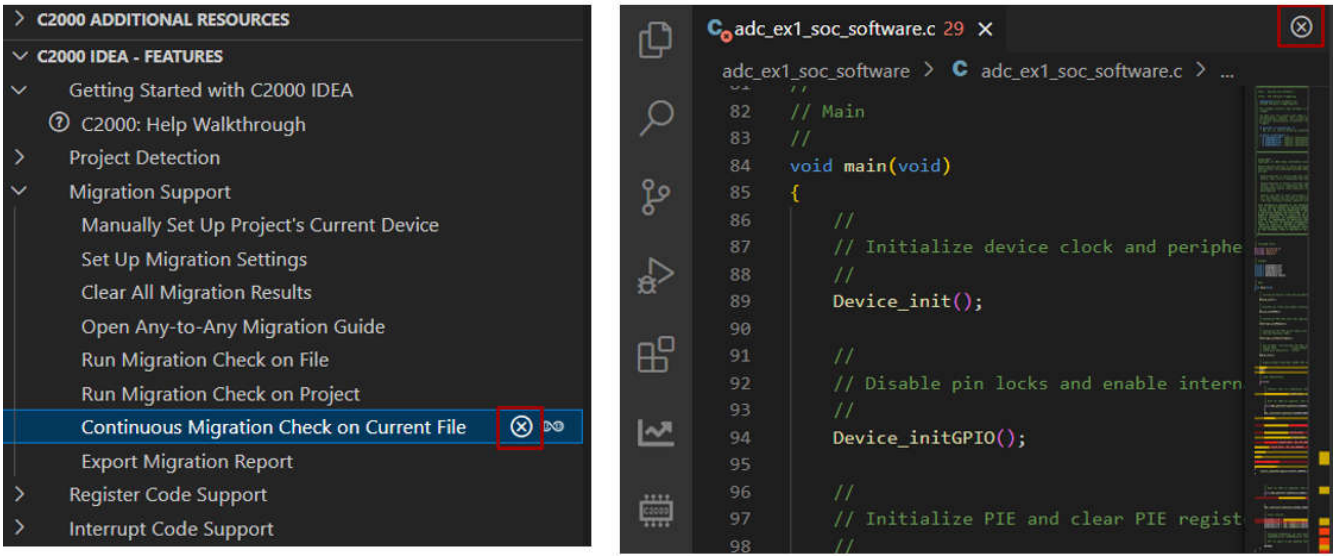

图 3-8. 对当前文件禁用持续迁移检查

3.3.2 对项目运行迁移检查

借助此扩展，用户能够对一个整个 C2000 器件项目执行迁移检查。要执行迁移检查，请完成以下步骤：

- 确保已根据要求自定义*迁移设置设定*页面。
- 要对项目运行迁移检查，请使用以下方法之一：
  1. 输入 CTRL+SHIFT+P，并点击 *C2000: 对项目运行迁移检查*。
  2. 扩展树方法：
     a. 导航至 *C2000 IDEA - 功能*窗格中的“迁移支持”> *对项目运行迁移检查*
     b. 单击 *C2000 IDEA - 项目* 窗格中检测到的项目下的迁移器件旁边的图标，如图 3-9 所示。

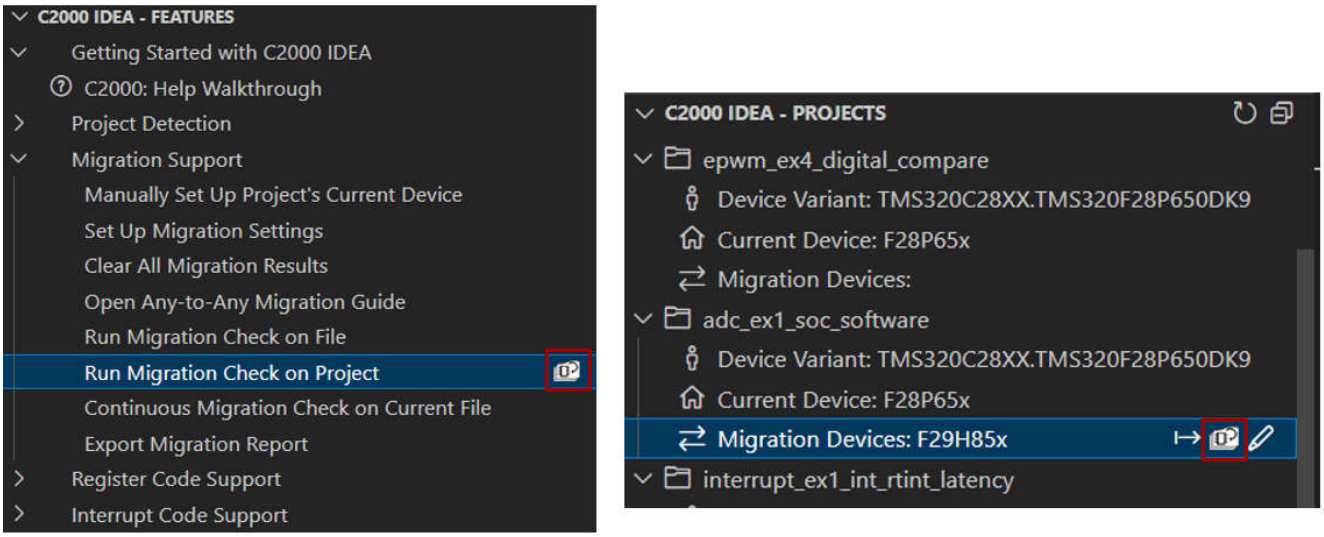

图 3-9. 项目迁移执行（扩展树）

备注

如果迁移文件夹中的文件夹或文件与迁移设置页面中要忽略的文件信息存在重叠，则扩展会抛出错误。

备注

当工具已经在对项目运行迁移检查时，请勿使用这些迁移功能。等待屏幕右下角的 *[项目名称] 迁移检查完成* 显示，然后再启用其他迁移功能。运行检查所需的时间完全取决于项目中存在多少个文件、多少行和多少代码更改。迁移报告包含每个文件所用的时间。

对整个项目运行迁移检查后，屏幕的右下角会显示一个带有百分比指示器的进度条。此进度条提供实时更新，显示迁移期间当前正在处理的文件路径。

迁移过程完成后，进度条将在同一位置替换为一条状态消息，指示*已完成 [项目名称] 的迁移检查*。

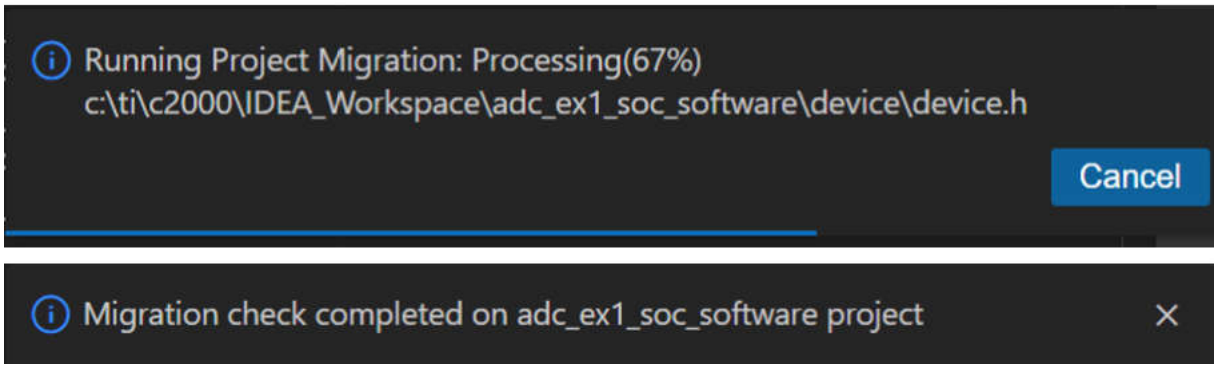

图 3-10. 项目迁移进度条和完成通知

可以在 Code Composer Studio (CCS) 的*输出控制台* 中查看项目迁移摘要。此摘要提供有关迁移过程的信息，包括被忽略的文件和文件夹。

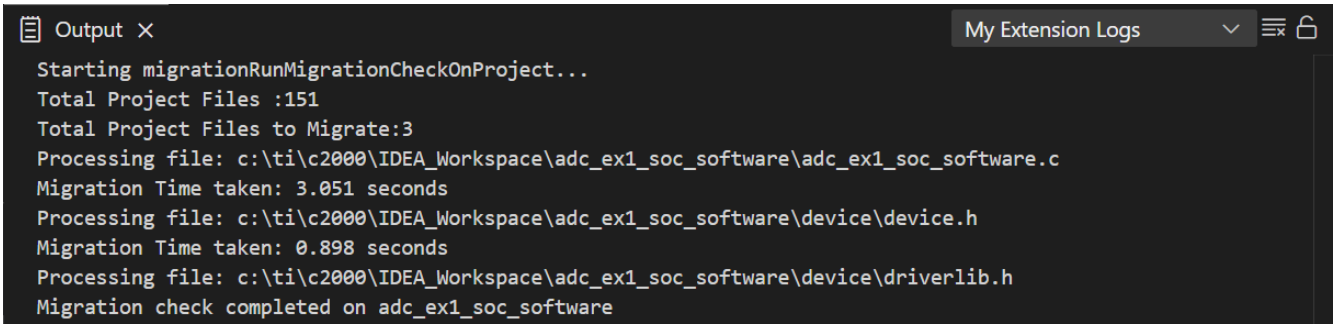

图 3-11. 项目迁移后的 CCS 输出控制台

3.4 快速修复

对独立文件或整个项目执行迁移运行时，C2000 IDEA 扩展可检测并突出显示迁移问题。文件中的所有主要迁移问题都用红色波浪下划线标出，文件中的所有其他迁移警告都用黄色波浪下划线标出。

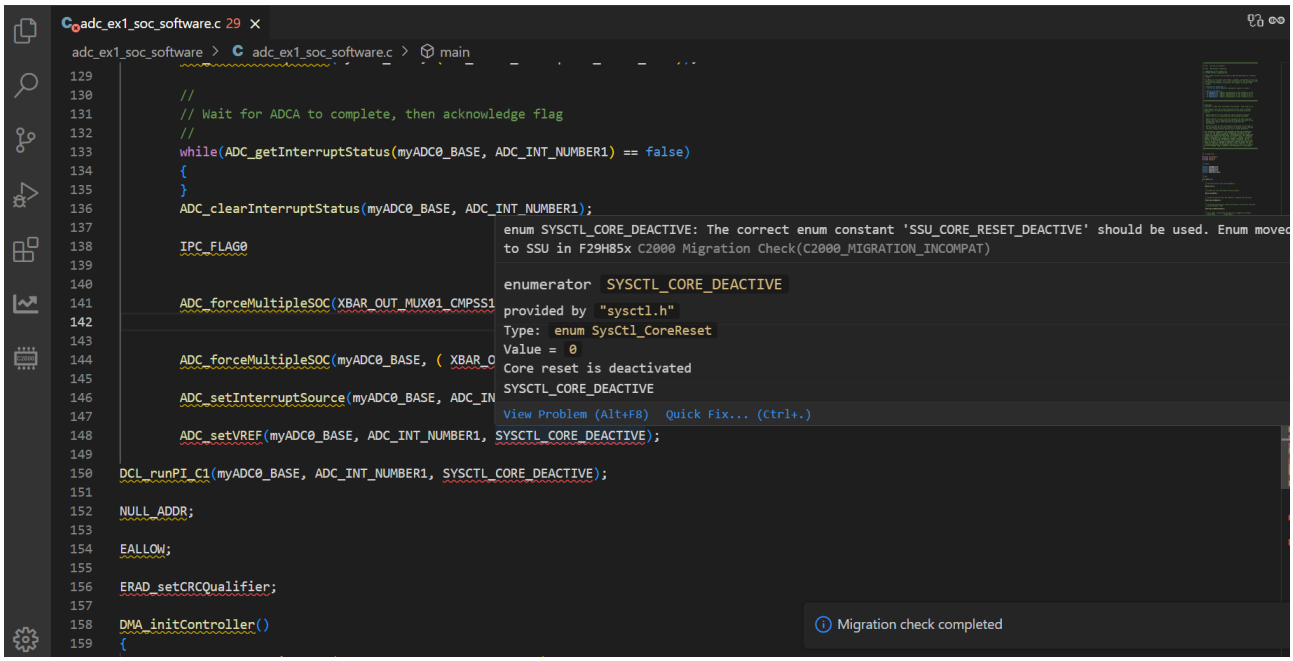

图 3-12. 迁移执行后的文件视图

每个突出显示的迁移问题都包含一条提示消息以及*查看问题*和*快速修复*选项：

- 查看问题：显示有关迁移问题的详细说明，说明当前和迁移器件之间的差异。
- 快速修复：针对可用于高效解决迁移问题的答案给出建议。

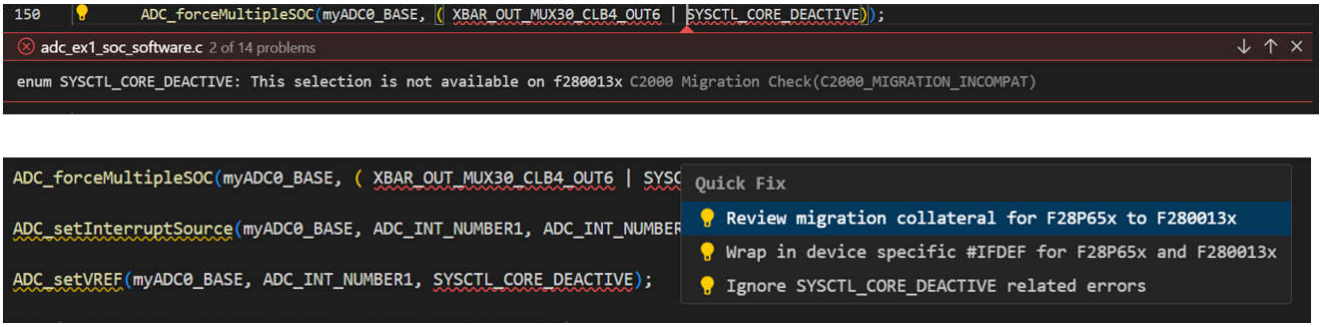

图 3-13. 迁移问题 — 查看问题和快速修复

此扩展提供了多种选项，可帮助用户高效解决迁移问题，并确保器件之间的平稳过渡。用户可以在以下解决方案中进行选择：

- *查看从 [当前器件] 到 [迁移器件] 的迁移配套资料*：
  - 打开指向 C2000 器件迁移报告生成 的链接，获取最新的 C2000WARE 在线迁移配套资料，为特定迁移代码更改提供详细指导。

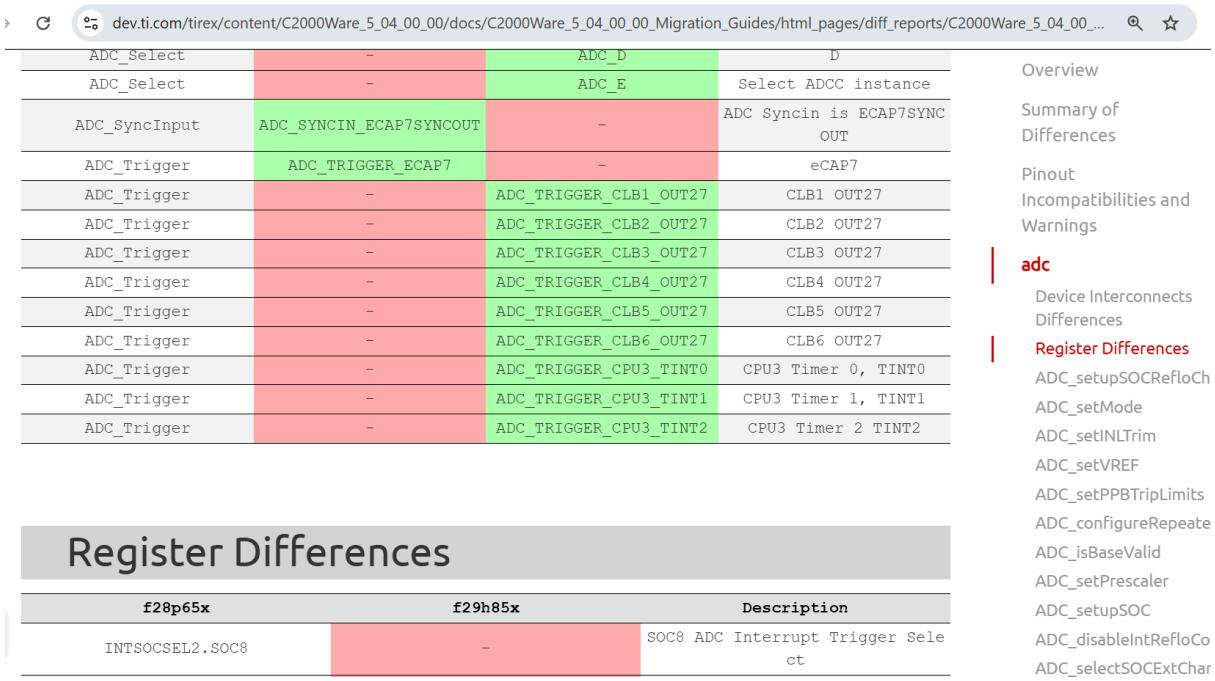

图 3-14. 迁移配套资料 HTML 页面

- *为 [当前器件] 和 [迁移器件] 打包在器件专属 #IFDEF 中*
  - 在代码行周围自动生成预处理器条件 (#IFDEF)，以便为新器件编译代码的更新版本，并在文件的对应位置为当前器件添加一个 #define。
  - 允许用户定义一个为新器件量身定制的替代代码实现方案。在需要进行修改的地方插入占位符注释 (// 输入替代代码) 。
  - 该扩展可以为大多数 F28x 至 F29x 迁移问题提供代码替换建议。

```
Start of device specific migration code - F28P65x
#if F28P65x //_DEVICE_MIGRATION_
        ADC_forceMultipleSOC(XBAR_OUT_MUX01_CMPSS1_CTRIPOUTL, ( XBAR_OUT_MUX30_CLB4_OUT6 | SYSCTL_CORE_DEACTIVE));
Change of device specific migration code - F280013x
#elif F280013x //_DEVICE_MIGRATION_
        // Enter alternate code
End of device specific migration code - F28P65x/F280013x
#endif //_DEVICE_MIGRATION_
```

图 3-15. 打包 #IFDEF 快速修复（F28x 至 F28x）

```
Start of device specific migration code - F28P65x
#if F28P65x //_DEVICE_MIGRATION_
        ADC_forceMultipleSOC1(XBAR_OUT_MUX01_CMPSS1_CTRIPOUTL, ( XBAR_OUT_MUX30_CLB4_OUT6 | SYSCTL_CORE_DEACTIVE));
Change of device specific migration code - F29H85x
#elif F29H85x //_DEVICE_MIGRATION_
        // Suggested replacement: ADC_forceMultipleSOC1(XBAR_OUT_MUX01_CMPSS1_CTRIPOUTL, ( XBAR_OUT_CLB4_OUT6 | SYSCTL_CORE_DEACTIVE));
End of device specific migration code - F28P65x/F29H85x
#endif //_DEVICE_MIGRATION_
```

图 3-16. 打包 #IFDEF 快速修复（F28x 至 F29x）

- *忽略与代码相关的错误*
  - 通过在 *设定迁移设置* 页面的“迁移检查例外”部分中添加来抑制迁移问题。
  - 通常在用户实施了不同方法以手动处理迁移问题时使用。
- *所有枚举修复为 [当前器件] 和 [迁移器件] 打包在器件专属 #IFDEF 中*
  - 功能类似于 #IFDEF 之前的快速修复，但专门针对基于枚举的迁移问题而设计。
  - 仅在迁移到 F29H85x 时以及在同一行上存在多个与枚举相关的更改时才会出现。
  - 自动为该行中所有相关的枚举迁移问题应用 #IFDEF 包装器。

```
Start of device specific migration code - F28P65x
#if F28P65x //_DEVICE_MIGRATION_
        ADC_forceMultipleSOC1(XBAR_OUT_MUX01_CMPSS1_CTRIPOUTL, ( XBAR_OUT_MUX30_CLB4_OUT6 | SYSCTL_CORE_DEACTIVE));
Change of device specific migration code - F29H85x
#elif F29H85x //_DEVICE_MIGRATION_
        // Suggested replacement:ADC_forceMultipleSOC1(XBAR_OUT_CMPSS1_CTRIPOUTL, ( XBAR_OUT_CLB4_OUT6 | SSU_CORE_RESET_DEACTIVE));
End of device specific migration code - F28P65x/F29H85x
#endif //_DEVICE_MIGRATION_
```

图 3-17. 所有枚举打包 #IFDEF 快速修复（F28x 至 F29x）

通过利用这些快速修复选项，用户可以显著减少手动工作，简化迁移过程并确保器件之间的代码兼容性。

3.5 迁移报告

当在 C2000 器件之间转换时，迁移报告在分析代码变化方面发挥着至关重要的作用。其中提供了修改的结构化概述，可帮助用户评估迁移复杂性、识别潜在问题并选择最佳目标器件。借助详细的诊断和导出功能，迁移报告可确保迁移流程既简洁又高效。

可以在执行迁移后为整个项目或特定的活动文件生成此报告。这种灵活性使开发人员能够专注于关键部分，同时全面了解迁移所带来的影响。

有多种方法可以生成和导出迁移报告：

1. 输入 CTRL+SHIFT+P，并点击 *C2000：导出迁移报告*。
2. 扩展树方法：
   a. 导航至 *C2000 IDEA - 功能* 窗格中的“迁移支持”> *导出迁移报告*。
   b. 单击 *C2000 IDEA - 项目* 窗格中检测到的项目下的迁移器件旁边的图标，如图 3-18 所示。

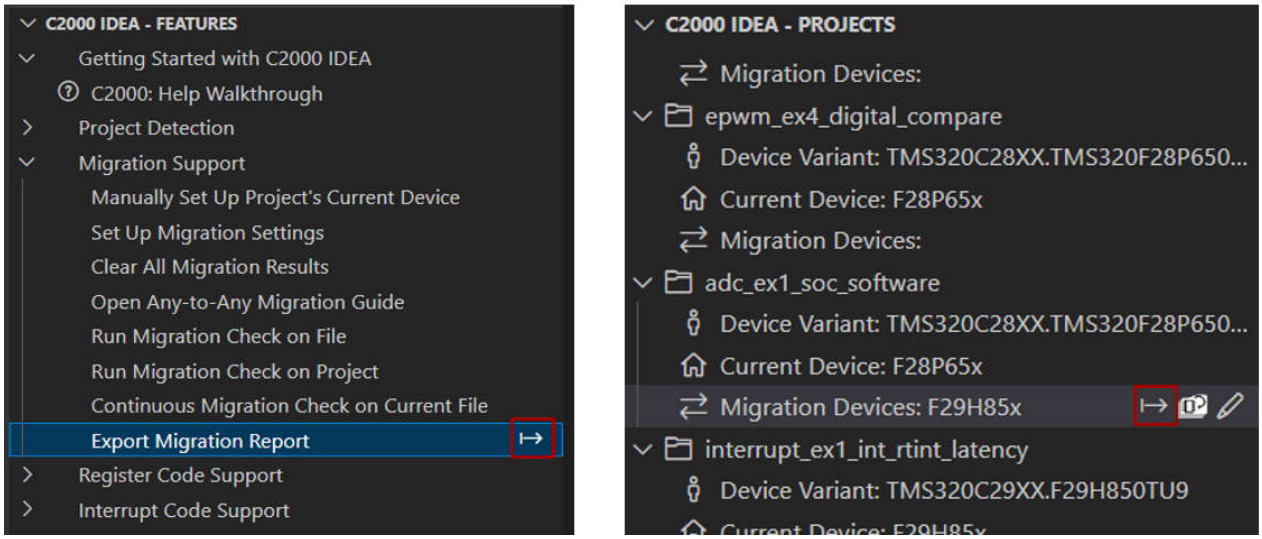

图 3-18. 导出迁移报告（扩展树）

生成后，可以使用文件浏览器将报告导出并保存到所需位置，验证迁移数据是否可供进一步分析和记录。

迁移报告中的关键部分

1. 迁移信息 — 概述了迁移过程
   a. *源器件和目标器件*：显示原始器件和选定的一个或多个迁移器件。
   b. *忽略的符号、文件和文件夹*：列出在迁移过程中未考虑的已排除符号、文件和目录。
   c. *已处理的文件*：指定经过迁移的已分析文件。
   d. *迁移时间*：表示每个文件迁移所花费的时间，帮助深入了解性能和复杂性。
2. *迁移诊断* — 突出显示迁移过程中检测到的更改和潜在问题
   a. *分类*：每个检测到的更改都根据严重性分类为警告或错误。
   b. *详细代码更改分析*：提供源器件和目标器件之间差异的清晰说明。
   c. *位置*：指定受影响代码的精确行 (Ln) 和列 (CoL)，提高问题解决效率。

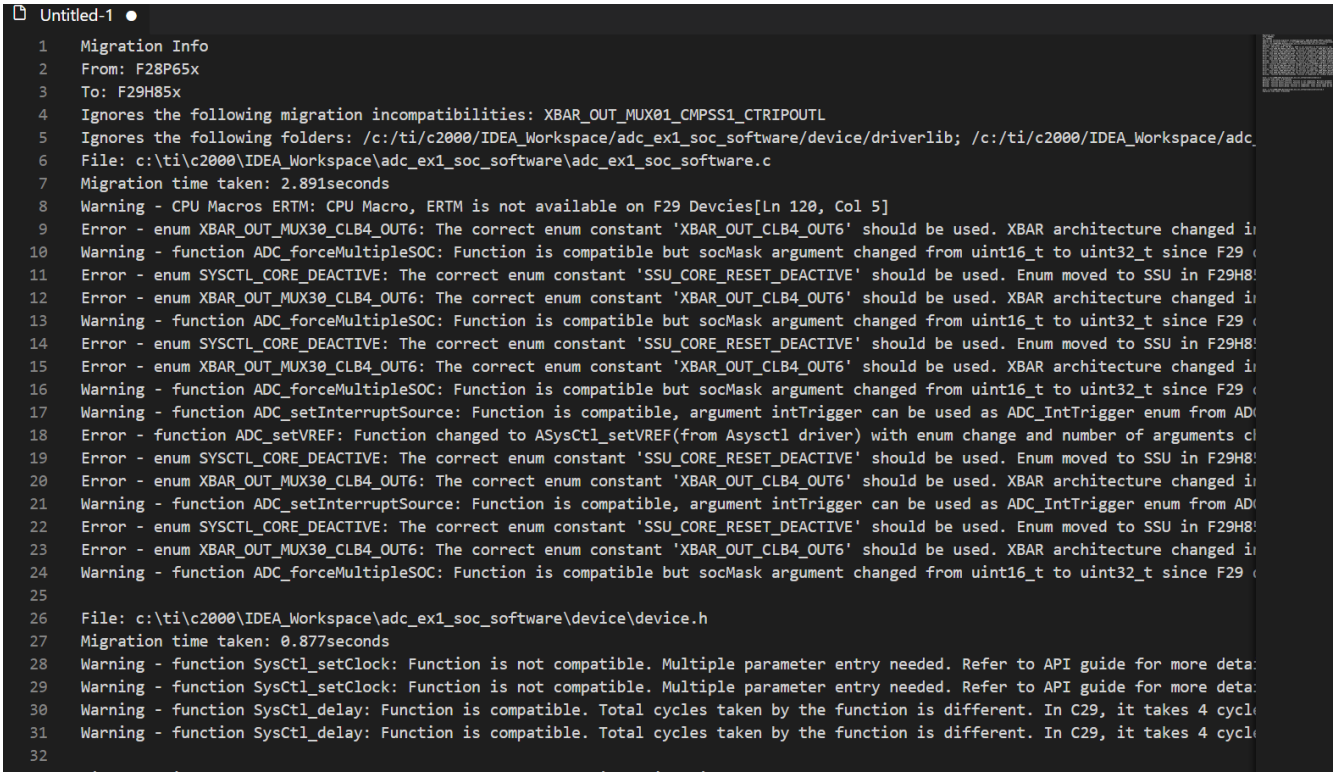

图 3-19. 迁移报告

通过导出报告并保存为文档，开发人员可以根据需要随时参考迁移详细信息。该报告对于分析多个目标器件之间的迁移复杂性、帮助开发人员评估潜在的挑战并确定最佳迁移方法特别有用。有了对代码修改和潜在障碍的清晰认识，开发人员可以在选择理想的迁移器件时做出明智的决策。

凭借结构化诊断、导出功能和详细见解，迁移报告在验证 C2000 器件之间是否无缝高效过渡方面是不可或缺的资源。

3.6 位域迁移

C2000 IDEA 扩展可用于在使用 bitfield 样式代码编写的文件上运行 F28x 到 F28x 或 F28x 到 F29x 迁移检查。这种代码样式的特点是使用对 bitfield 源文件中定义的函数的调用 (如：InitSysCtrl()) 和/或包含 [base name].[register name].all or [base name].[register name].bit.[field name] 语法的寄存器访问。

如要启用位域迁移，请按照以下步骤操作：

1. 打开 C2000 应用 C-Code 文件。
2. 通过按 CTRL+SHIFT+P，输入并选择“*C2000: 对文件运行 Bitfield 迁移检查*
3. 选择将文件中的代码应用于哪一个现有的 C2000 器件。
4. 选择要将文件迁移至哪一个 C2000 器件。
5. 完成后，屏幕右下角的状态栏会显示“*已完成从 [当前器件] 到 [迁移器件] 的位场迁移*”。文件中所有迁移问题均已用红色波浪下划线标出。
6. 查看并解决整个文件中的问题。当鼠标悬停在带下划线的代码上时，系统会提供以下选项：
   a. 选择*查看问题*，快速循环浏览文件中检测到的问题。
   b. 选择*快速修复*，缓解迁移问题。选择以下选项之一：
      i. *查看从 [当前器件] 到 [迁移器件] 的迁移配套资料* - 此选项会打开使用最新版本的 C2000WARE 的特定迁移路径的在线迁移资料的链接。
      ii. *为 [当前器件] 和 [迁移器件] 打包器件专属 #IFDEF* - 此选项会在代码行周围自动生成预处理器包装器，以便为新器件编译代码的更新版本。用修改后的代码填写带有“*//输入替换代码*”注释的行，并在文件的对应位置为当前器件添加一个 #define。

iii. *忽略与代码相关的错误* - 此选项会忽略此迁移问题。

4 总结

德州仪器 (TI) C2000 SDK 提供了驱动程序和库等基本工具，可简化实时控制应用程序开发。该生态系统具有用于简化代码初始化和外设配置的 C2000 SysConfig 等功能，旨在为各个级别的开发人员提供支持。C2000 IDEA 工具是在这些基础功能的基础上构建而成，用于集中开发环境，以便更高效地进行编码和调试。C2000 IDE Assist Extension (IDEA) 通过促进跨器件组合的无缝迁移以及将旧代码转换到新器件，进一步增强了这一工具。

作为 Code Composer Studio (CCS) 20 的强大扩展，IDEA 可自动分析客户代码、识别 TI MCU 之间的潜在迁移问题并通过上下文文档提供优化的解决方案，从而加快软件迁移。该工具会生成一个迁移报告，为代码修改提供结构化分析，突出显示兼容性问题并帮助选择最佳目标器件。IDEA 专为初学者和经验丰富的开发人员而设计，可简化迁移任务并确保器件之间的平稳过渡，从而提高工作效率。通过在每个开发阶段集成实时辅助功能，IDEA 工具显著减少了嵌入式软件迁移所需的时间和精力。凭借智能自动化和简化的接口，IDEA 工具成为嵌入式软件领域首个能够为实时微控制器实现高效、高性能开发的解决方案。

5 参考资料

工具与软件：

- 德州仪器 (TI)，*C2000 IDEA Open VSX (VSIX 下载)*
- 德州仪器 (TI)，*C2000 Idea GitHub 存储库 (VSIX 下载)*
- 德州仪器 (TI)，*Code Composer Studio (CCS) IDE*
- 德州仪器 (TI)，*C2000WARE (F28x SDK)*
- 德州仪器 (TI)，*F29X-SDK (F29x SDK)*

文档：

- 德州仪器 (TI)，*C28x Academy - 迁移资源*
- 德州仪器 (TI)，*C29x Academy - 迁移资源*
- 德州仪器 (TI)，*F28x 到 F29x 软件迁移指南*
- 德州仪器 (TI)，*将应用软件迁移到 C29 CPU 应用说明*
- 德州仪器 (TI)，*将 TMS320F2837x、TMS320F2838x、TMS320F28P65x 迁移至 TMS320F29H85x 用户指南*
- 德州仪器 (TI)，*C2000 设计与开发*

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