

ET-Studio 使用说明

文档版本 2.1
发布日期 2026-08-4

版权所有 © 成都翌创微电子有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



和其他翌创微商标均为成都翌创微电子有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受成都翌创微电子有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，成都翌创微电子有限公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

成都翌创微电子有限公司

地址：四川省成都市高新区高朋大道 3 号东方希望大厦 9 楼 901 号

电话：028-83350520

网址：<https://www.etmcu.com>

邮箱：support@etmcu.com

前言

概述

本文档是关于针对 ET-Studio 使用相关的内容。

读者对象

本文档主要适用于以下工程师：

- 电子产品硬件设计人员
- 电子产品软件设计人员

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	用于警示紧急的危险情形，若不可避免，将会导致人员死亡或严重的人身伤害。
 警告	用于警示潜在的危险情形，若不可避免，可能会导致人员死亡，严重或中轻度的人身伤害。
 注意	用于传递设备或环境安全警示信息，若不可避免，可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “注意”不涉及人身伤害。
 说明	用于突出重要/关键信息，是针对正文内容做的附加说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害。

修订记录

修订记录累积了本文档每次修订的具体信息。

序号	版本	修改日期	修改说明
1	1.0	2025/09/23	初始发布
2	2.0	2026/06/05	新增 ET6025/ET6039/ET6039C/ET6015X 工程结构调整 在线文档接入 新增驱动版本管理
2	2.1	2026/08/04	ET6039C/ET6015X 驱动更新 增加示例导入说明 增加 JLINK 驱动添加说明

说明：以下内容均以 ET6002 为例讲解，其余目标芯片配置过程中参考 ET6002 配置修

改即可。

目录

前言	2
目录	3
1 简介	5
1.1 软件定位	5
1.2 核心功能	5
1.3 支持芯片型号	5
2 安装	6
2.1 软件包获取	6
2.2 安装流程	6
2.3 PACK 芯片驱动包安装	6
3 功能简介	7
3.1 选择工作空间	7
3.2 Welcome 界面介绍	7
3.3 在线文档查看	8
3.4 Gitee 仓库访问	8
4 工程配置	9
4.1 新建工程	9
4.2 导入工程	10
4.3 源文件添加	11
4.4 头文件路径添加	12
4.5 全局宏添加	13
5 编译、调试、下载	14
5.1 编译	14
5.2 调试	14
5.2.1 JLINK 调试	15
5.2.1.1 驱动安装	15
5.2.1.2 调试配置	15
5.2.1.3 各调试按键功能描述:	17
5.2.1.4 添加断点	17
5.2.1.5 查看寄存器值	18
5.2.1.6 查看变量值	18
5.2.1.7 返回工程界面	19
5.2.2 CMSIS-DAP 调试	19
5.3 下载	19
6 ITCM 中运行	20

6.1 修改连接脚本文件	20
6.2 修改调试器配置	20
7 驱动更新	22
7.1 自动安装	22
7.2 手动安装	22
7.3 旧工程更新驱动	22

1 简介

1.1 软件定位

ET Studio 是基于 Eclipse 平台深度定制开发的嵌入式集成开发环境（IDE），专为 ETMCU 系列微控制器打造，旨在为开发者提供一站式、高效便捷的嵌入式开发解决方案。

从项目创建到代码编写、调试测试，ET Studio 覆盖嵌入式开发全流程：依托 Eclipse 成熟的代码编辑框架，支持语法高亮、代码补全、跳转查错等功能，大幅降低 ETMCU 相关代码的编写难度；内置针对 ETMCU 的编译工具链与调试适配模块，可直接对接硬件调试器，实现程序下载、在线调试、断点设置等核心调试操作，无需额外配置第三方工具；同时兼容 ETMCU 系列不同型号芯片的硬件特性，能自动匹配对应的芯片参数与外设驱动模板，帮助开发者快速搭建项目基础框架，减少重复开发工作，显著提升基于 ETMCU 的嵌入式项目开发效率与稳定性。

1.2 核心功能

1. 内置全套编译链、芯片驱动模板，开箱即用，无需手动配置交叉编译器；
2. 双仿真器原生支持：SEGGER J-Link、CMSIS-DAP；
3. 支持一键创建工程模版、导入工程；
4. 可视化调试：外设寄存器、内存、变量实时观测，源码 / 汇编双调试视图；
5. 内置在线芯片手册、Gitee 官方 SDK 仓库、多版本 PACK 驱动管理。

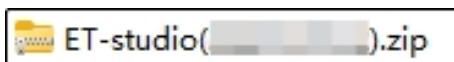
1.3 支持芯片型号

ET6002、ET6025、ET6039、ET6015X、ET6039C

2 安装

2.1 软件包获取

获取到 ET-Studio 压缩包后，对其进行解压缩。



解压后有两个文件，分别为：ET-Studio 软件压缩包及其使用说明。



2.2 安装流程

提取 ET-Studio.zip 将其解压缩到您想要安装的指定路径（需纯英文路径）。

建议：不要直接在原文件夹下解压，否则可能显示文件路径过长，导致解压失败。

解压完成后，双击 ET-Studio 即可运行



2.3 PACK 芯片驱动包安装

PACK 包存储各型号芯片底层驱动、链接脚本、寄存器定义等，ET Studio 中已经安装有默认版本的 pack 包。若要更新 pack 包，详见第 7 章节。

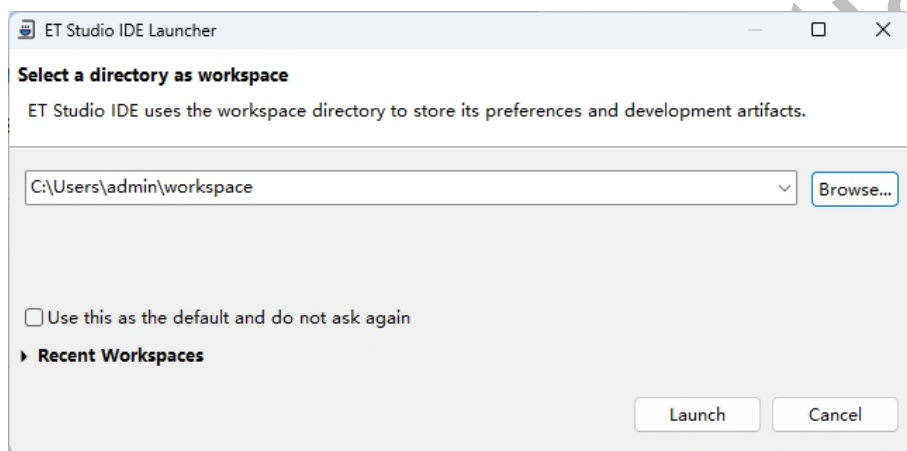
3

功能简介

3.1 选择工作空间

运行 ET-Studio 时，需要配置工作空间，可根据自身喜好进行设置。默认路径在 C 盘，建议修改。

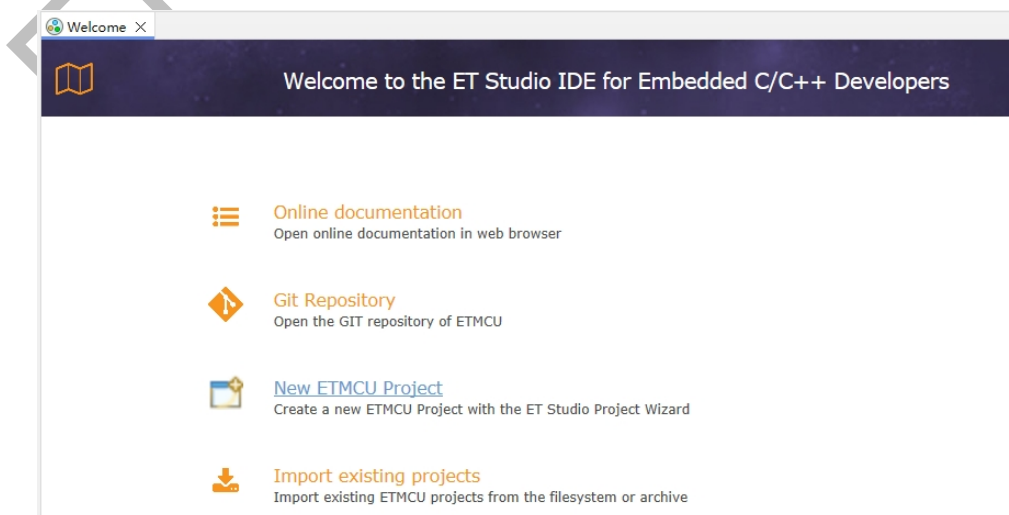
若确定工作空间后，不想每次启动都出现此弹窗，可勾选下方 “Use this as the default and do not ask again”



3.2 Welcome 界面介绍

首次打开 ET Studio，会自动打开 Welcome 界面，界面布局如下：

- Online documentation: ETMC 的在线文档。
- Git Repository: ETMCU 的 gittee 仓库。
- New ETMCU Project: 创建一个 ETMCU 的工程模版。
- Import existing projects: 导入现有 ETMCU 工程



3.3 在线文档查看

在线文档链接内嵌在 Welcome 界面中，首次启动时会自动打开 Welcome 界面。

Welcome 在 Help 选择卡下，点击 Help->Welcome->Online documentation，然后选择目标芯片，即可查看相关在线文档。



3.4 Gitee 仓库访问

Gitee 仓库链接内嵌在 Welcome 界面中，首次启动时会自动打开 Welcome 界面。

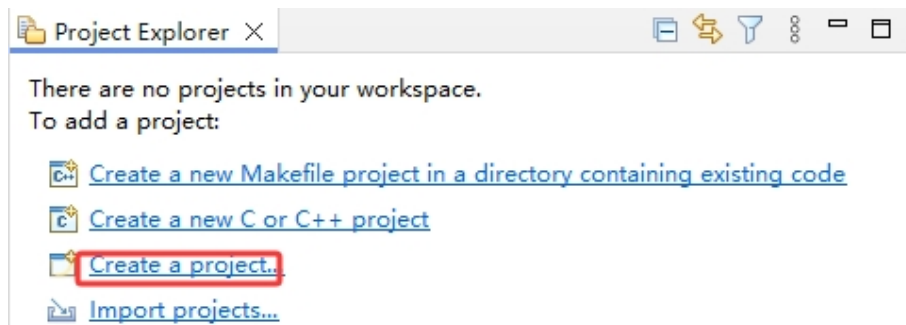
Welcome 在 Help 选择卡下，点击 Help->Welcome->Git Repository 即可访问。

4

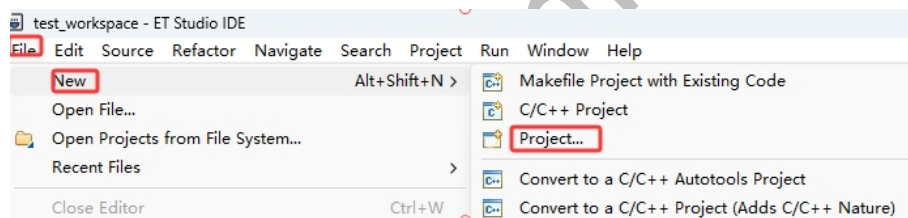
工程配置

4.1 新建工程

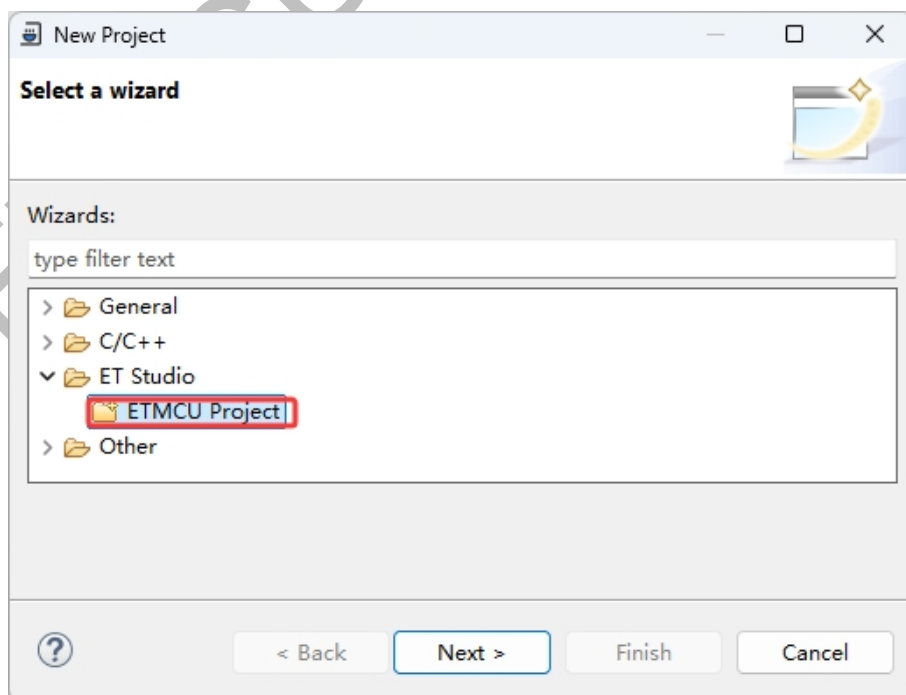
工作空间中没有工程时，左侧会出现一个快捷菜单，点击“Create a project...”即可创建 ETMCU 工程。



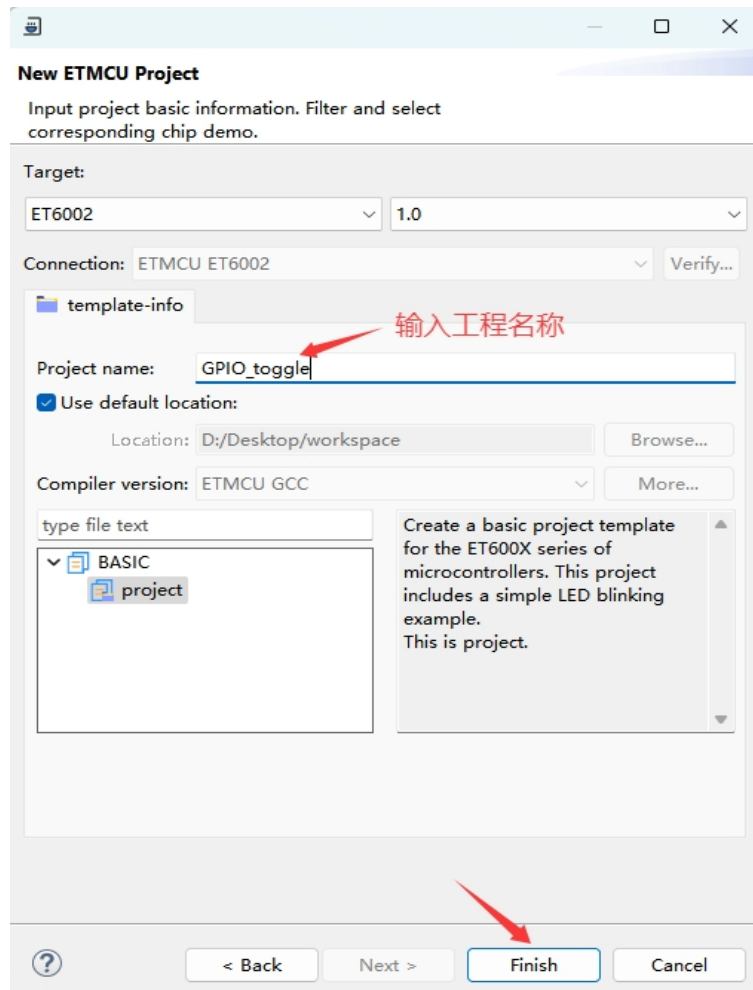
或者点击左上角，File->New->Project



选择 ETMCU Project，点击 Next

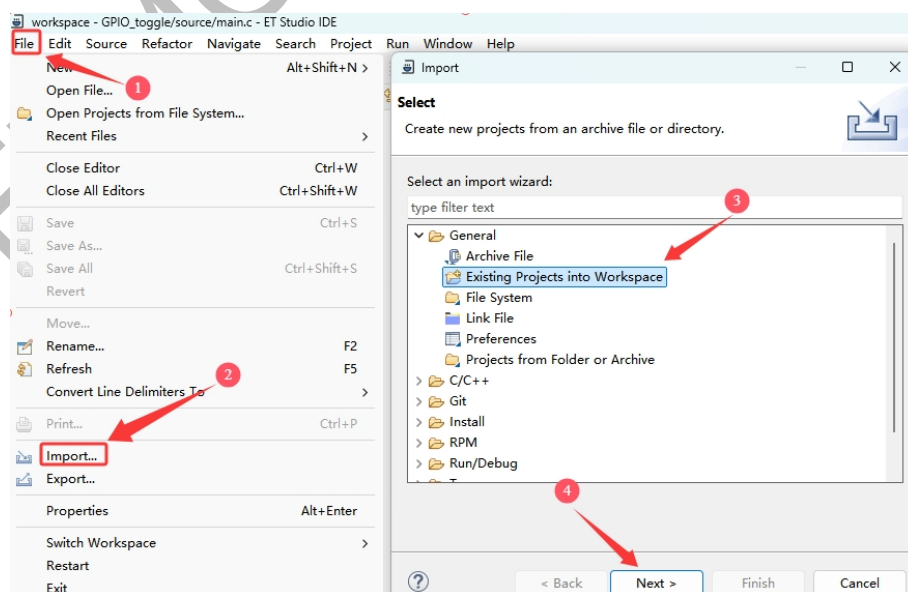


输入工程名称，点击 Finish

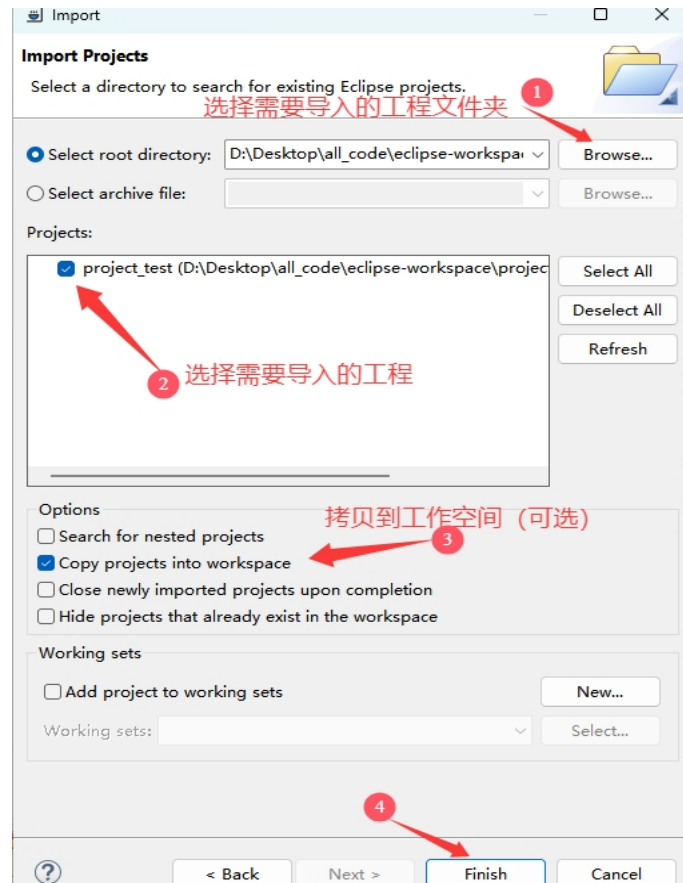


4.2 导入工程

若以往有已经创建好的工程，可以直接导入。

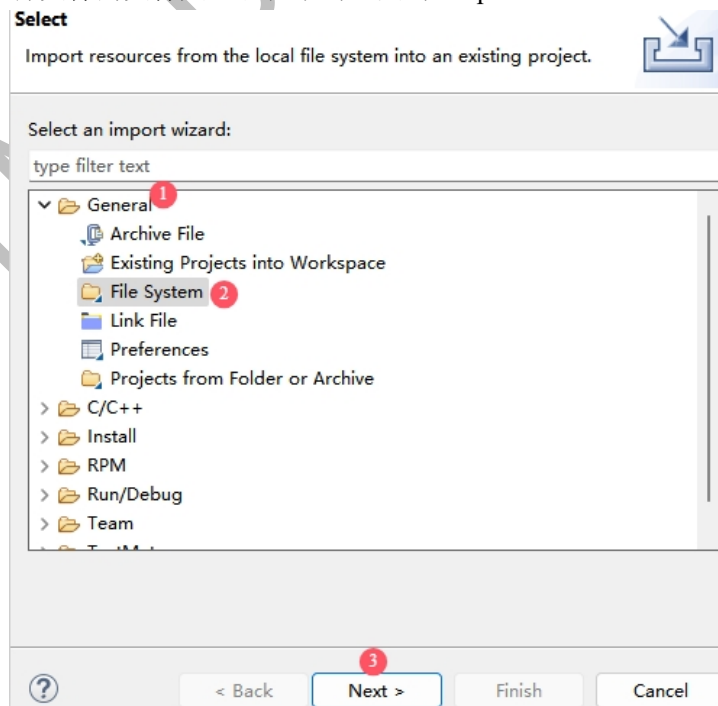


选择需要导入的工程



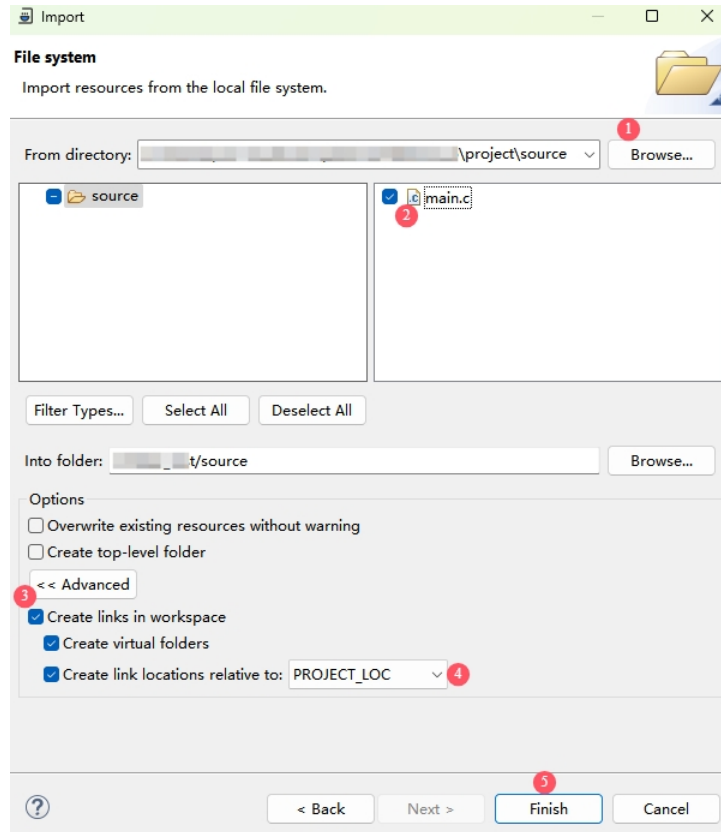
4.3 源文件添加

在需要添加源文件的文件夹上右键单击，点击 Import->General->File System



找到源文件所在位置，勾选需要添加的文件，然后勾选 create links in workspace，并选

择相对位置关系。

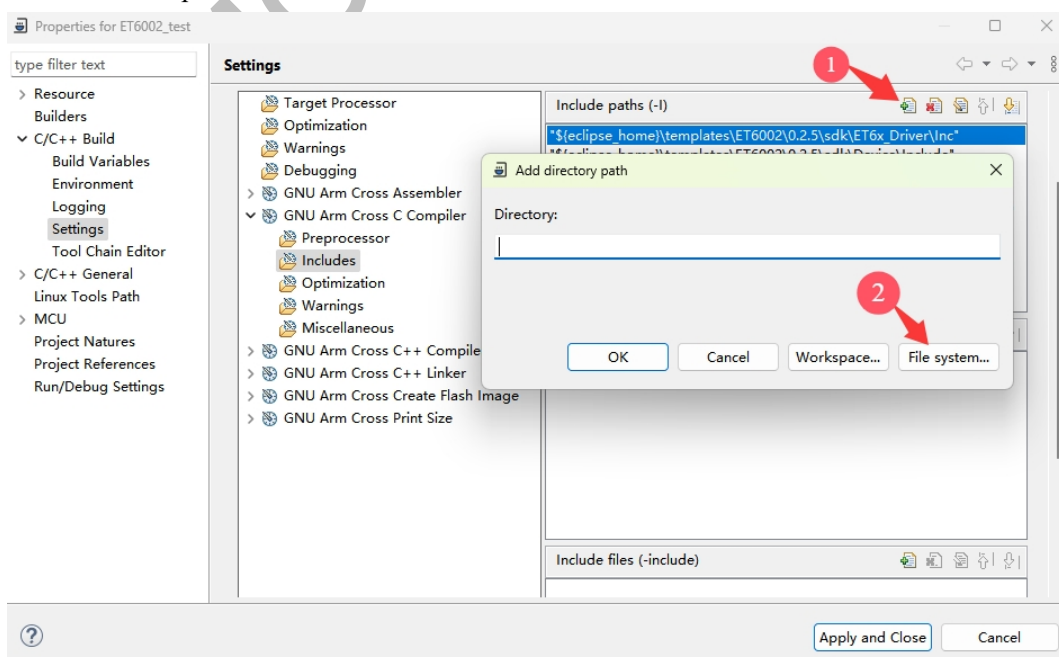


4.4 头文件路径添加

右键单击项目，点击 Properties->C/C++ Build->Setting->GNU Arm Cross C Compiler->Include,

单击右上角 +，然后在弹窗中选择需要添加的文件路径即可完成添加。

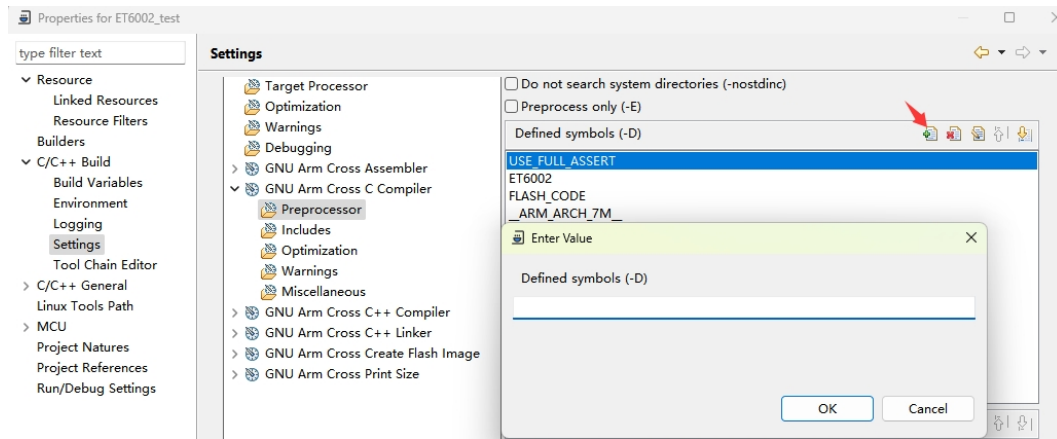
点击 Workspace 添加的路径则为工程所在位置的相对路径。



4.5 全局宏添加

右键单击项目，点击 Properties->C/C++ Build->Setting->GNU Arm Cross C Compiler->Preprocessor，

单击右上角 + ，然后在弹窗中输入需要添加的宏即可。

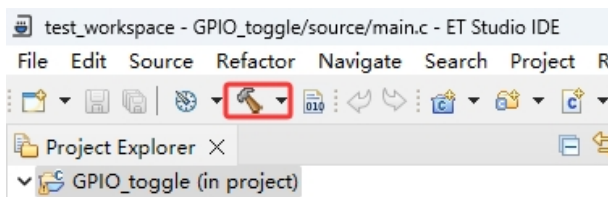


5

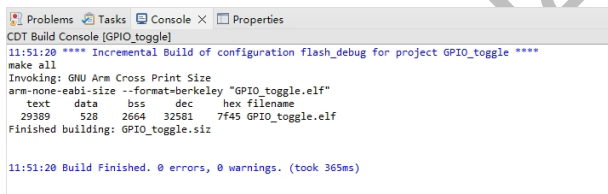
编译、调试、下载

5.1 编译

选中工程，然后点击 build project 进行工程编译

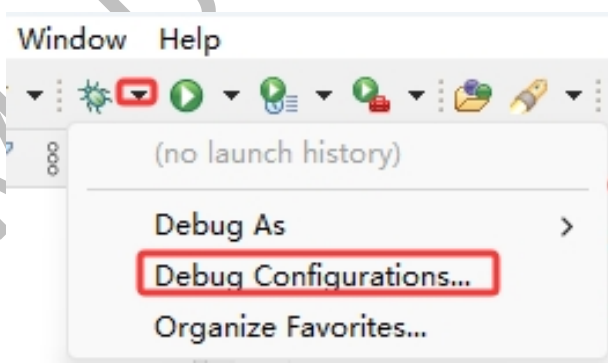


编译结果

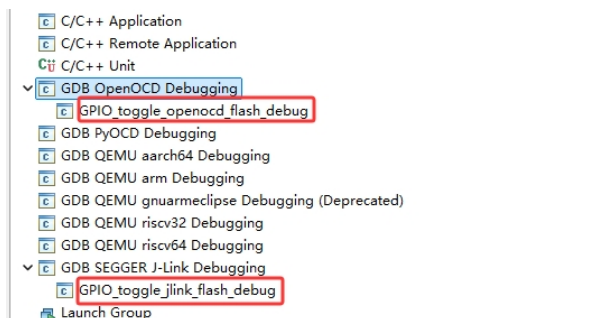


5.2 调试

首次使用调试运行功能时，点击 Debug configuration...



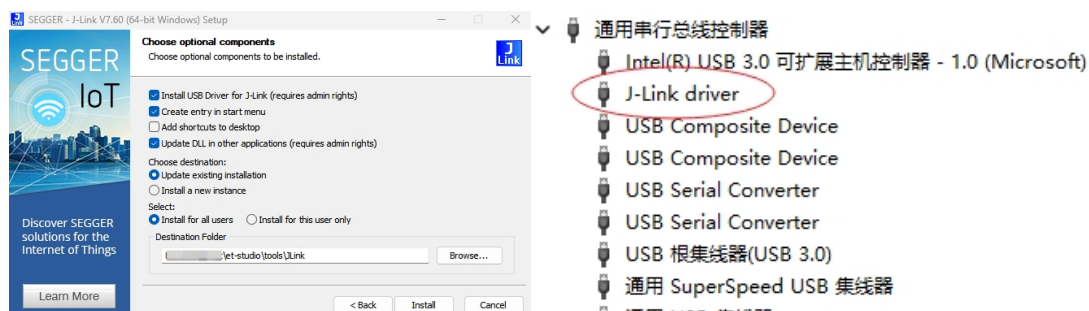
默认已生成 openocd、jlink 调试配置文件



5.2.1 JLINK 调试

5.2.1.1 驱动安装

到 [SEGGER 官网](#) 下载（建议下载 **V7.86** 以下版本） 下载完成后直接安装到 et-studio\tools\JLink 路径下，安装完成驱动也相应的安装完成。如安装成功，接入 J-link 调试器，打开设备管理器会显示如下：

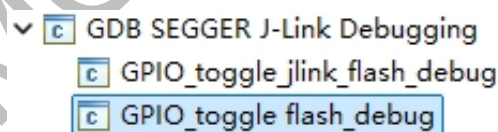


在 [gitee 仓库](#) 或 [百度网盘](#) 下载 JLINK 补丁文件，根据 readme.txt 说明，选择 JLink 版本对应的文件夹，将文件复制到对应路径中。

注意：路径必须是 et-studio 安装路径下的 \tools\JLink，不能有其他多余字符，若安装时强行添加了路径后缀，需手动删除。也可安装到其他路径下，安装完成后复制安装后的全部内容到 et-studio 安装路径下的 \tools\JLink。

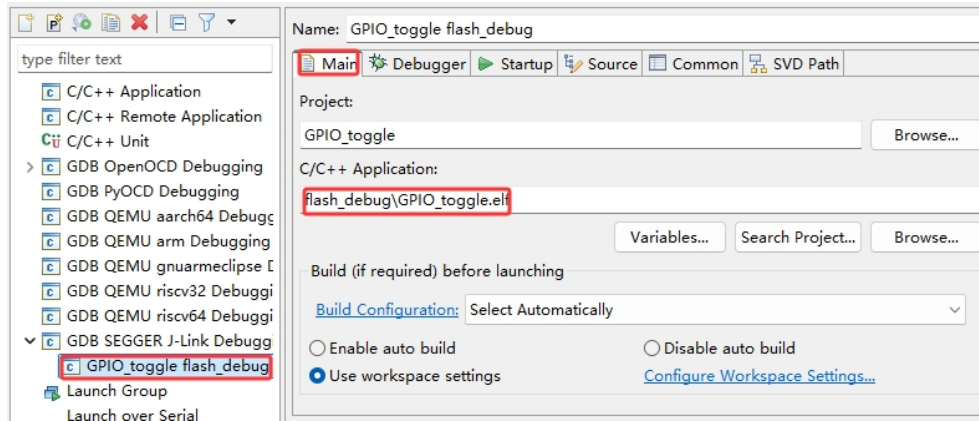
5.2.1.2 调试配置

使用生成的配置文件（推荐），或双击“GDB SEGGER J-link Debugging”，新建配置文件。

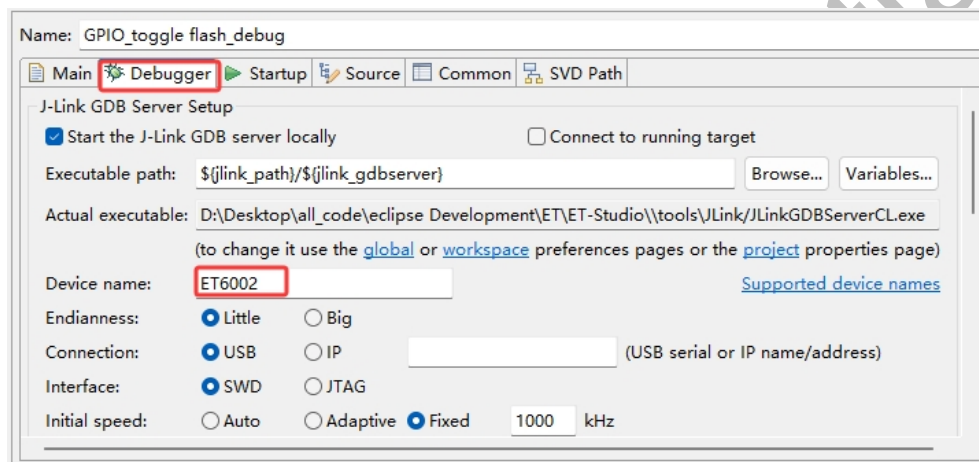


点击生成的配置文件，查看各配置参数是否已经被正确填入，若没有则手动填入（若使用自动生成的配置文件，通常参数已经被正确填入）。

若 C/C++ Application 中为空，则点击下方“Search Project”即可找到该工程的.elf 文件

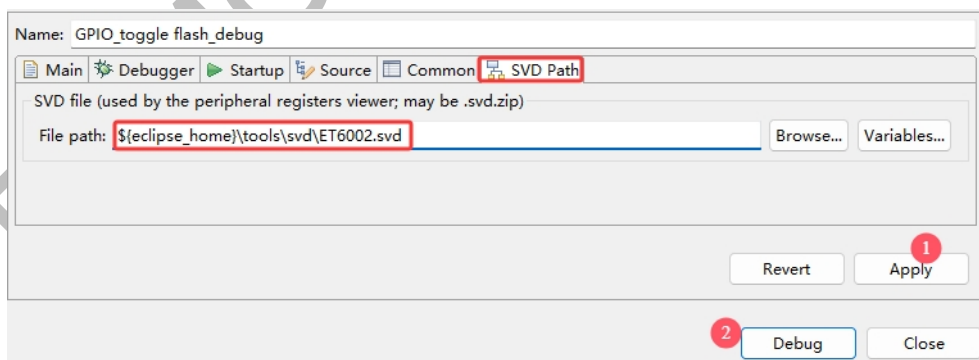


若 Device name 为空，则手动填入

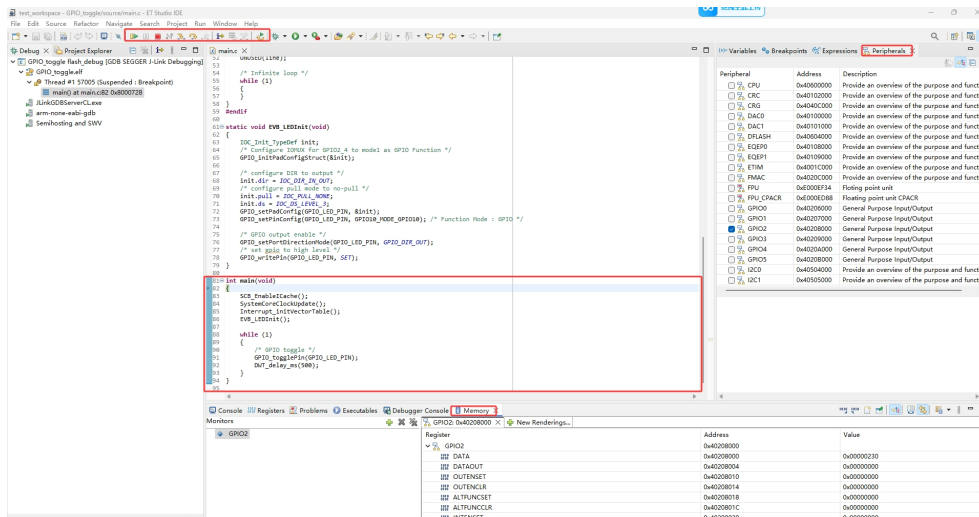


File path 中若为空，则手动填入：\${eclipse_home}\tools\svd\ET6002.svd，或者点击图中“Browse...”寻找样 tools 目录下的配置文件“ET6002.svd”

配置完成后点击 Apply，Debug



进入调试后，如下图：



5.2.1.3 各调试按键功能描述：

-  表示全速运行
-  表示暂停
-  表示停止调试
-  表示单步进入函数
-  表示单步跳过函数
-  表示跳出函数
-  表示汇编单步
-  表示复位

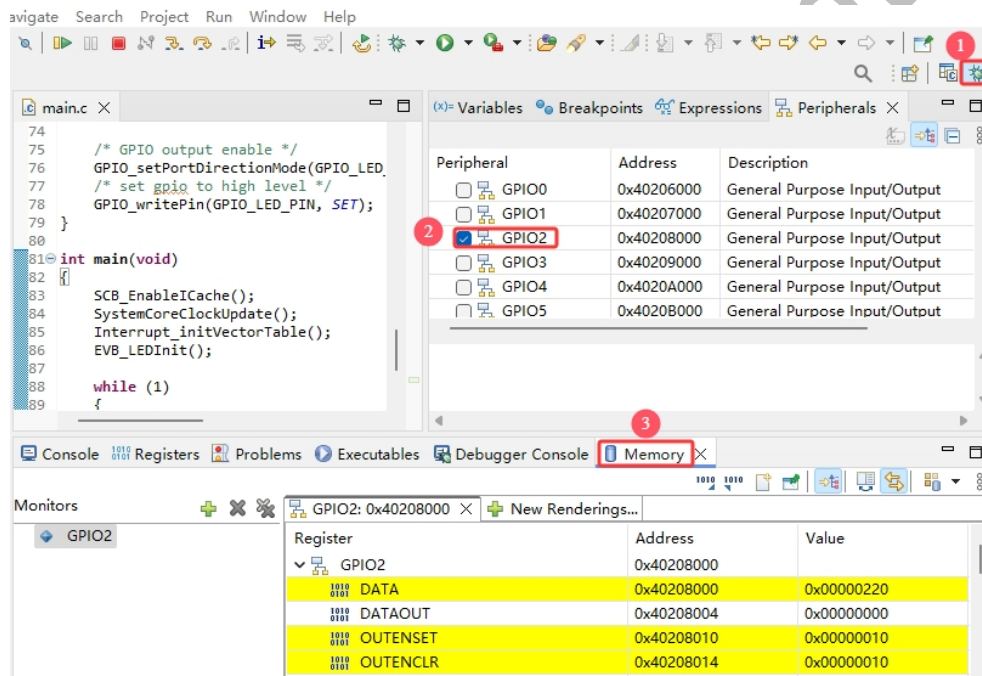
5.2.1.4 添加断点

双击需要添加断点所在行，添加断点，再次双击即可取消

```
81 int main(void)
82 {
83     SCB_EnableICache();
84     SystemCoreClockUpdate();
85     Interrupt_initVectorTable();
86     EVB_LEDInit();
87
88     while (1)
89     {
90         /* GPIO toggle */
91         GPIO_togglePin(GPIO_LED_PIN);
92         DWT_delay_ms(500);
93     }
94 }
95
```

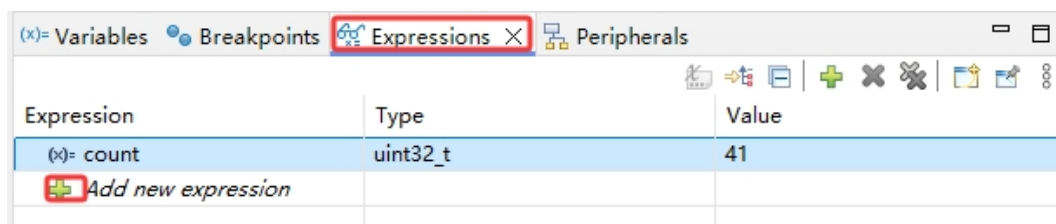
5.2.1.5 查看寄存器值

需要查看哪组寄存器，勾选后即可在下方查看。



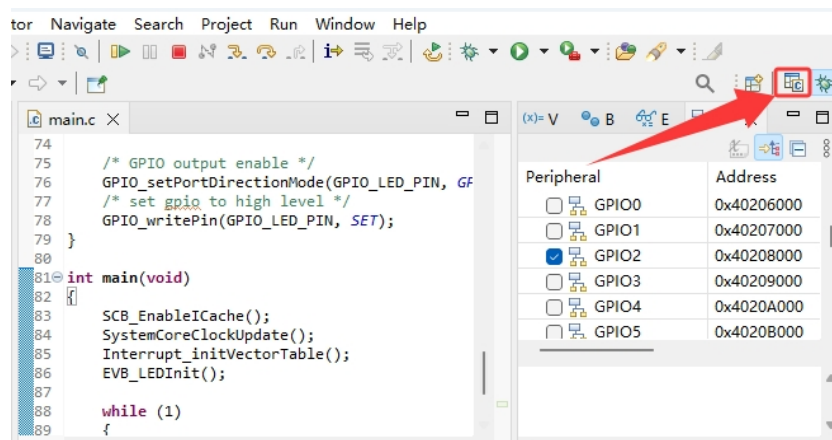
5.2.1.6 查看变量值

点击 Expressions，然后点击下方 +，输入需要观察的变量名或者表达式。



5.2.1.7 返回工程界面

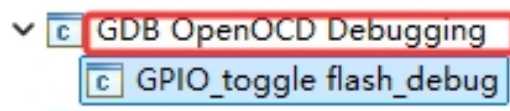
退出仿真后，想要回到工程管理界面，需要通过界面右上角图标切换回对应界面



5.2.2 CMSIS-DAP 调试

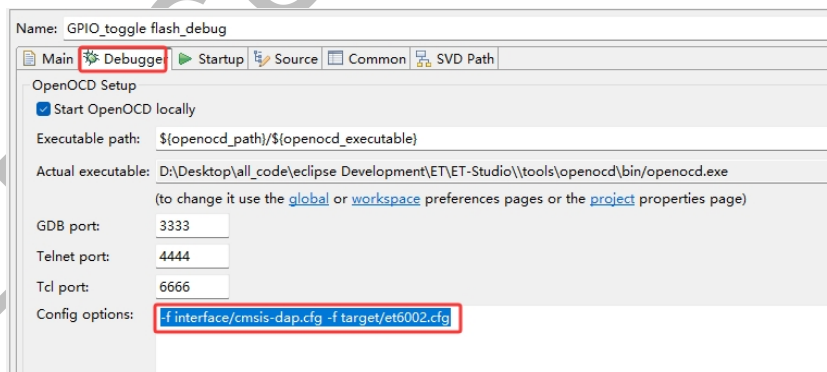
使用 CMSIS-DAP 调试无需安装驱动，内部已添加 openocd 支持。调试步骤与 JLINK 调试步骤基本一致，主要区别在于调试配置。

使用生成的配置文件（推荐），或双击“GDB OpenOCD Debugging”，新建配置文件。



点击配置文件，查看各配置参数是否已经被正确填入，若没有则手动填入。

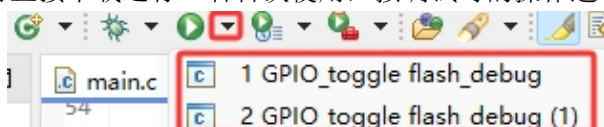
若 Config options 中为空，则手动填入：-f interface/cmsis-dap.cfg -f target/et6002.cfg



其余步骤与 JLINK 调试一致，参考 JLINK 调试步骤即可。

5.3 下载

若无需调试代码，可直接下载运行。若首次使用，按调试时的操作进行配置



6

ITCM 中运行

6.1 修改连接脚本文件

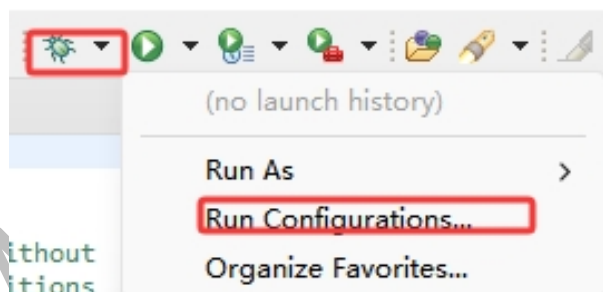
在工程中找到 Idscripts/sections.ld 双击打开，将 ROM1 的参数修改为 ITCM 的参数
ORIGIN = 0x00000000, LENGTH = 32K

```
3 MEMORY
4 {
5     ROM1 (rx) : ORIGIN = 0x08000000, LENGTH = 512K /* FLASH */
6     ROM2 (rwx) : ORIGIN = 0x00000000, LENGTH = 32K /* ITCM */
7     RAM1 (rwx) : ORIGIN = 0x20000000, LENGTH = 32K /* DTCM */
8     RAM2 (rwx) : ORIGIN = 0x20400000, LENGTH = 96K /* SRAM */
9 }
```

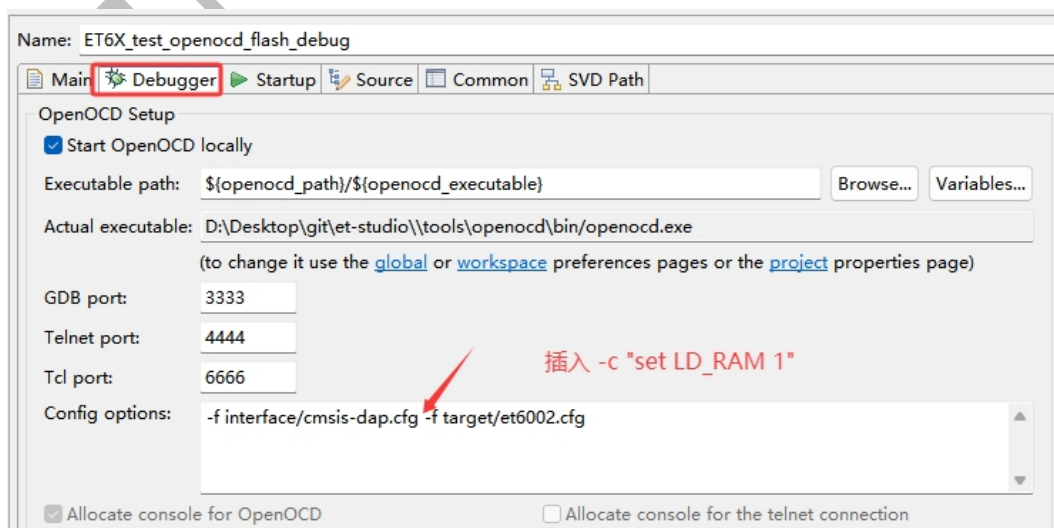
注意：修改连接脚本文件后，需要先 clean project，然后再重新 build project 才能生效。

6.2 修改调试器配置

打开调试器配置选项卡



修改 Config options 选项卡，在 -f target/et6002.cfg 前面插入：-c "set LD_RAM 1"



修改 Startup 配置选项：

取消勾选 Initial Reset 和 Load executable, Pre-run 选项卡；

在 Run/Restart Commands 下填入：set \$pc = 0x00000000

Name: ET6x_test_openocd_flash_debug

Main Debugger Startup Source Common SVD Path

Initialization Commands

☐ Initial Reset. Type: init

☒ Enable Arm semihosting

Load Symbols and Executable

☒ Load symbols

☒ Use project binary: ET6x_test.elf

☐ Use file:

Symbols offset (hex):

☐ Load executable

☒ Use project binary: ET6x_test.elf

☐ Use file:

Executable offset (hex):

Runtime Options

☐ Debug in RAM

Run/Restart Commands

☐ Pre-run/Restart reset Type: halt (always executed at Restart)

set \$pc = 0x00000000

☐ Set program counter at (hex):

☒ Set breakpoint at: main

☒ Continue

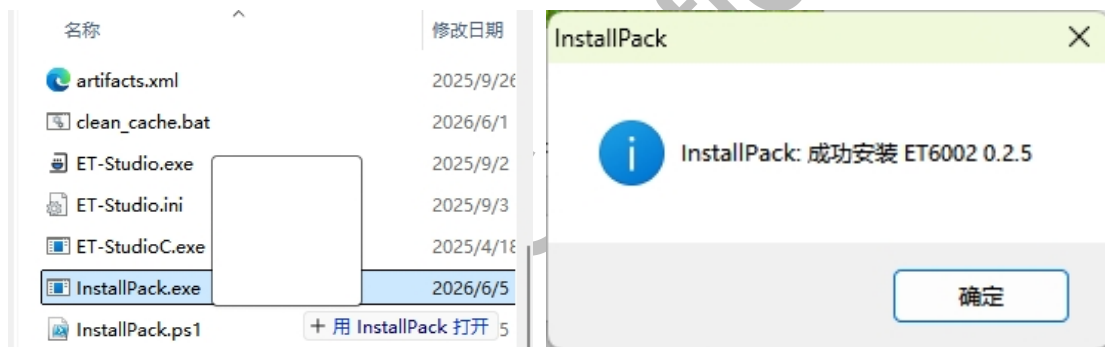
完成上述配置后，点击 Apply,然后点击 Debug 即可实现在 ITCM 区域进行代码调试。

7 驱动更新

PACK 包存放芯片底层驱动、寄存器、链接脚本，版本迭代后需更新，每次更新后会在 [ETMCU 官网](#) 发布 pack 包。ET Studio 芯片驱动可以通过以下方式更新。

7.1 自动安装

- 找到 ET-Studio 根目录下的 InstallPack.exe 程序，将下载完成的新版.pack 驱动文件，直接拖拽至该 exe 程序图标上；
- 程序将自动完成解压、注册驱动版本，安装结束后弹出窗口提示安装成功；
- 版本隔离机制：若新版 PACK 版本号与软件内现有驱动不一致，程序不会覆盖原有旧驱动，多版本驱动会同时保存在软件目录中。新建工程时，可在工程创建向导内自由选择需要使用的 PACK 驱动版本。



7.2 手动安装

- 右键本地的.pack 驱动压缩包，使用解压工具完整提取内部文件，解压后可获取 Device、ET6x_Driver 两个核心驱动文件夹；
- 进入 ET-Studio 安装路径：templates/目标芯片型号/驱动版本号/sdk/，将解压得到的 Device 与 ET6x_Driver 文件夹复制粘贴至该 sdk 目录，在系统弹窗中选择覆盖原有同名文件，完成底层驱动替换。

7.3 旧工程更新驱动

历史工程创建时会绑定对应 PACK 版本，更新驱动后不会自动切换至新版，可通过手动安装方式替换旧版本驱动，或修改新版 .pack 文件夹的版本号命名，使其与软件默认内置驱动版本号完全一致，再拖拽至 InstallPack.exe 执行自动安装，新版本会直接替换原有内置驱动。