

STM32CubeIDE 实用技巧之 STM32H7 双核调试配置

关键字: STM32CubeIDE, H7, 双核, 调试

1. 前言

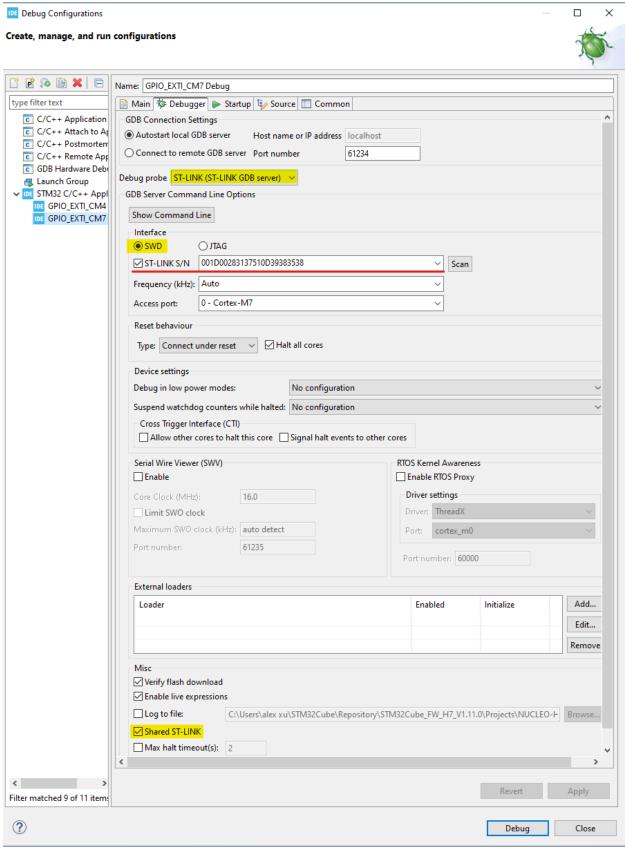
在 STM32CubelDE 中进入 Debug 模式来开发 STM32,是非常重要的调试手段,相对于单核来说,双核的 Debug 配置项就会多一些注意事项,这里给出一些关键的配置说明,确保用户顺利进入双核的调试模式。

2. 调试配置

首先我们在调试器中,选择 "ST-LINK GDB server" 然后选择 SWD 接口,并扫描当前插入电脑上的开发板(调试器),其他选项都保持默认配置即可,见图 1.



图1. M7 工程的调试配置选项-Debugger



LAT1396 - Rev 1.0 page 2/12



然后在 Startup 选项卡中,点击"Add"按钮,按照下图所示新增 M4 工程的下载,这样我们就一次性将 M7 和 M4 的执行文件下载到芯片中,如图 2.

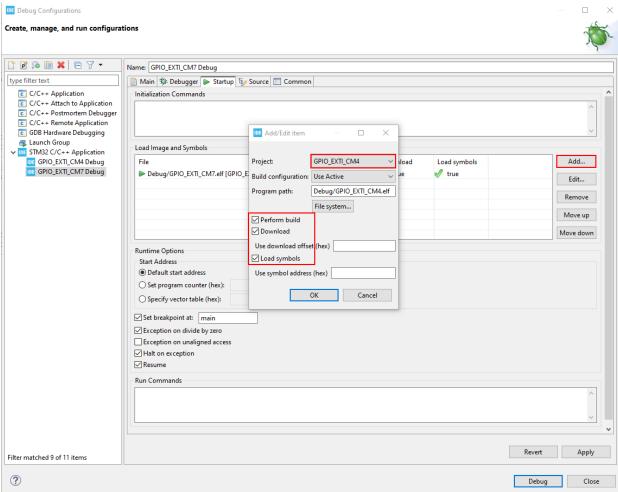


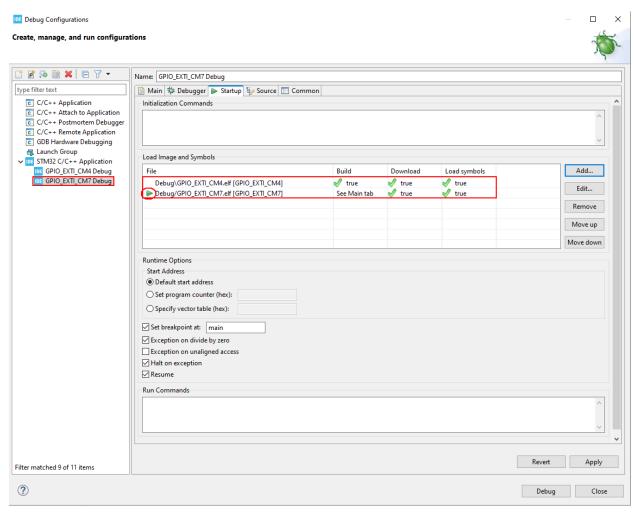
图2. M7 工程的调试配置选项-Startup

注意, M7 和 M4 的镜像文件的加载顺序是非常重要的, 我们要保持程序是从 M7 开始启动 (例程是配置的从 M7 内核启动的), 要让下图中的绿色箭头指向 M7 的 elf 文件, 以保证 PC 指针指向对应的镜像文件, 如下图 3 所示.

图3. M7 工程的调试配置选项-配置后的 Startup

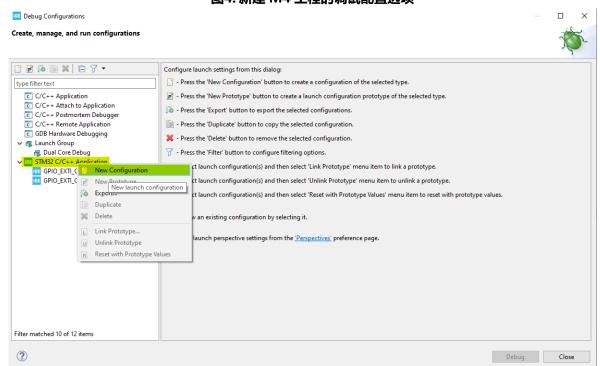
LAT1396 - Rev 1.0 page 3/12





用户可以通过如下图 4,来新建调试配置选项.

图4. 新建 M4 工程的调试配置选项



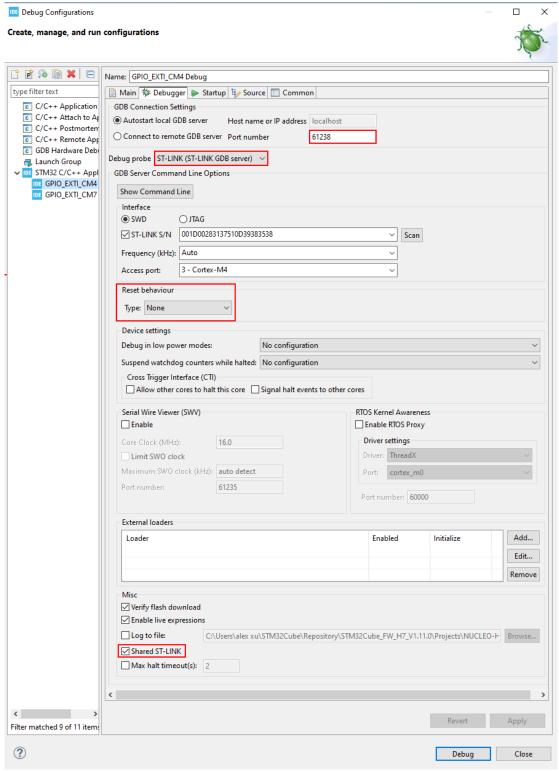
LAT1396 - Rev 1.0 page 4/12



在 M4 的 Debugger 选项表中,基于默认的配置,以下几点是重要检查项目,如图 5.

- 1. 确保 Port number 的值比刚刚 M7 中的这个值高出至少 3,这里设置的是 61238.
- 2. 确保 Reset behaviour 为 None.
- 3. ST-LINK 是 share 的选项.

图5. M4 工程的调试配置选项-Debugger



LAT1396 - Rev 1.0 page 5/12



在 M4 工程的 Startup 选项卡中,我们要调整默认的配置,通过"Edit"按钮,将镜像文件的下载选项取消,因为在 M7 中已经设置了下载,配置结果如图 6 所示.

Debug Configurations Create, manage, and run configurations Name: GPIO_EXTI_CM4 Debug 🗎 Main 🕸 Debugg 🕟 Startup 🔊 Source 🔲 Common type filter text C/C++ Application Initialization Commands © C/C++ Attach to Application
C C/C++ Postmortem Debugger
C C/C++ Remote Application
G GDB Hardware Debugging DE Add/Edit item Some properties of this item can only be changed in the "Main" tab. Launch Group Load Image and Symbols GPIO_EXTI_CM4 Project: STM32 C/C++ Application IDE GPIO_EXTI_CM4 Debug Build Load symbols Add... Debug/GPIO_EXTI_CM4.elf [GPIO_EXTI_CM4] See Main tab 🔞 false v true Edit... Remove Move up Download Move down Use download offset (hex) Load symbols Start Address Use symbol address (hex) Default start address OK Cancel O Set program counter (hex): O Specify vector table (hex): Set breakpoint at: main Exception on divide by zero Exception on unaligned access Halt on exception Resume Run Commands Re<u>v</u>ert Apply Filter matched 9 of 11 items ? <u>D</u>ebug Close

图6. M4 工程的调试配置选项-Startup

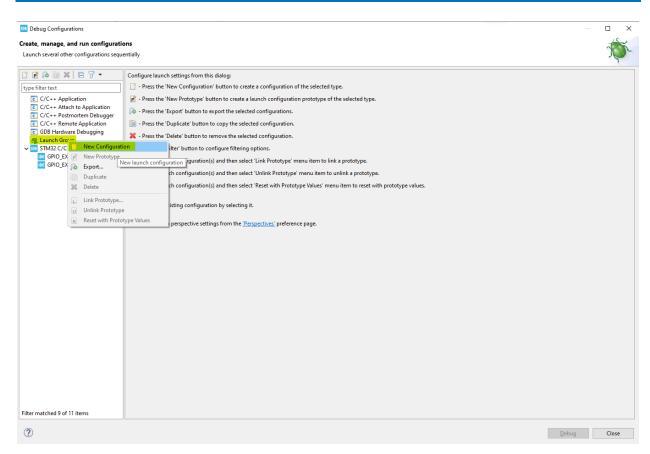
分别配置完 M7 和 M4 工程的调试配置后,我们还需要通过"Launch Group Configuration"选项,将两个工程组合在一起来启动双核的调试,如果没有配置这个项,那就会进入单核的调试。

我们按照图 7 所示,新建 Launch Group。

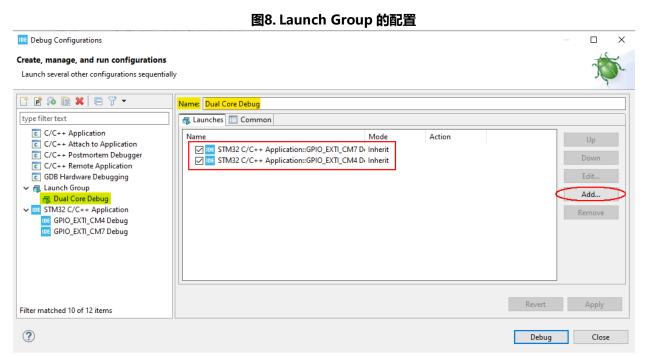
图7. 新建 Launch Group Configuration 选项

LAT1396 - Rev 1.0 page 6/12





然后通过 "Add" 按钮,为该组分别添加 M7 和 M4 的调试配置,并将其命名为 "Daul Core Debug",如下图 8 所示,这个名字是为了用户方便区分单核的调试,也可以使用默认的名称。



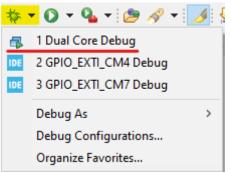
LAT1396 - Rev 1.0 page 7/12



3. 进入调试界面

通过上述的配置,我们就可以在调试的快捷按钮下方,看到刚才配置的启动选项了,如图 9,通过点击这里就可以快速进入双核的调试界面。

图9. 工程调试的快捷键显示的配置项



当然,我们也可以在工程的调试配置界面,通过点击如下图 10 的黄色部分,进入调试界面,效果是一样的。

Debug Configurations Create, manage, and run configurations Launch several other configurations sequentially Name: Dual Core Debug type filter text Launches Common C/C++ Application Mode Action C/C++ Attach to Application STM32 C/C++ Application::GPIO_EXTI_CM7 Di Inherit C/C++ Postmortem Debugger STM32 C/C++ Application::GPIO_EXTI_CM4 Dr Inherit C/C++ Remote Application C GDB Hardware Debugging ➤ ■ Launch Group

□ Dual Core Debug

□ STM32 C/C++ Application Add... Remove GPIO_EXTI_CM4 Debug
GPIO_EXTI_CM7 Debug Revert Apply Filter matched 10 of 12 items ? Debug Close

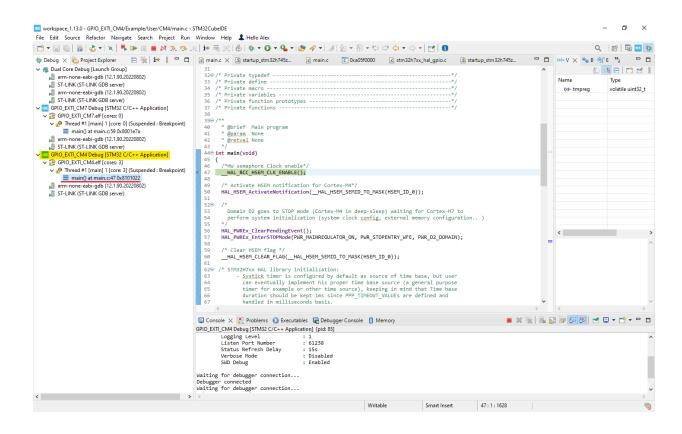
图10. 工程的调试配置

进入双核调试界面后,我们要在 M4 的进程中,让 M4 先运行起来,如图 11,然后在 M7 的工程中,让 M7 运行起来;否则,调试会出现异常,因为当前例程中的 M7 代码会等待 M4 boot 成功并进入 stop 模式,如果没有让 M4 先运行,那么 M7 的代码就会超时进入错误处理函数中,这点大家一定要理解和注意。

LAT1396 - Rev 1.0 page 8/12

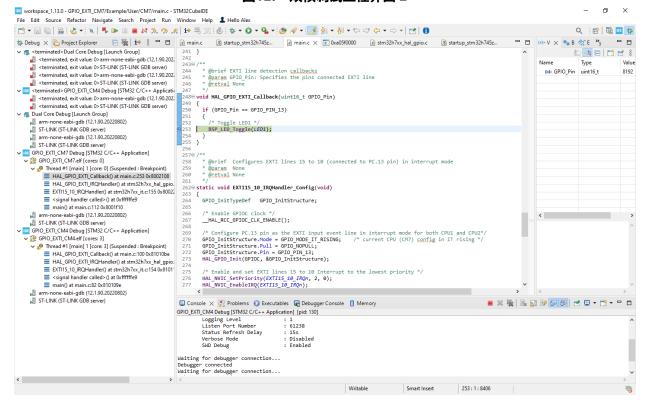


图11. 双核调试界面



在调试中, 双核是分开运行和控制的, 可以分别设置断点和运行状态, 如图 12 所示.

图12. 双核调试工程界面 2



LAT1396 - Rev 1.0 page 9/12



4. 注意事项

在使用时,大家遇到的最多的问题就是以下几点:

- 1. ST-LINK 要选择为 Share 属性;
- 2. M7 调试配置中要将 M4 的工程一同下载到芯片中,因此,在 M4 的调试配置中, 我们就不需要再下载镜像文件了(不勾选 Download)。
- 3. M4 的调试配置中,Debugger 的 port number 要比 M7 的至少高 3(见图 5).
- 4. M4 的 MCU 复位行为要选择为 "None".
- 5. 进入调试界面,要通过 Launch Group(本例中重命名为"Dual Core Debug")进入。
- 6. 在调试时,要让 M4 先运行起来(这取决于硬件启动配置和代码的实现,绝大多数用于都是和本例的启动方式相同)。

5. 小结

在整个配置过程中,我们用到了单核调试中不会关注的几点配置,只要稍加注意,双核调试的配置就能很快完成。

LAT1396 - Rev 1.0 page 10/12



参考文献

| 文件编号 | 文件标题 | 版本号 | 发布日期 |
|--------|--|-------|---------------|
| AN5361 | Getting started with projects based on dual-core | Rev 4 | February 2022 |
| | STM32H7 microcontrollers in STM32CubeIDE | | |
| | | | |
| | | | |

文档中所用到的工具及版本

STM32CubeIDE V1.13

LAT 中的附件

LAT_STM32CubeIDE 实用技巧之 STM32H7 双核调试的配置.7z

版本历史

| 日期 | 版本 | 变更 |
|-------------|-----|------|
| 2024年02月21日 | 1.0 | 首版发布 |
| | | |
| | | |

LAT1396 - Rev 1.0 page 11/12



重要通知 - 请仔细阅读

意法半导体公司及其子公司 ("ST")保留随时对 ST 产品和/或本文档进行变更的权利,恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。 ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用, ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定,将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 徽标是 ST 的商标。若需 ST 商标的更多信息,请参考 www.st.com/trademarks。所有其他产品或服务名称均为其 各自所有者的财产。

本文档是 ST 中国本地团队的技术性文章,旨在交流与分享,并期望借此给予客户产品应用上足够的帮助或提醒。若文中内容存有局限或与 ST 官网资料不一致,请以实际应用验证结果和 ST 官网最新发布的内容为准。您拥有完全自主权是否采纳本文档(包括代码,电路图等)信息,我们也不承担因使用或采纳本文档内容而导致的任何风险。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2020 STMicroelectronics - 保留所有权利

LAT1396 - Rev 1.0 page 12/12