

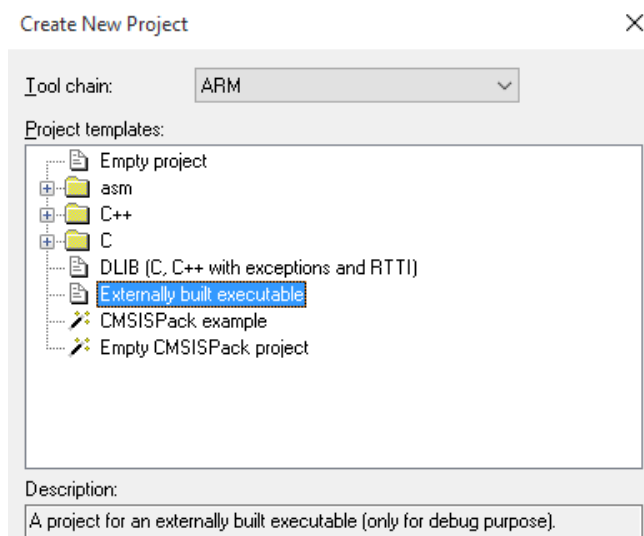
IAR Embedded Workbench for ARM 加载 ELF/Dwarf 文件调试

IAR EWARM 生成的带调试信息的可执行文件是 Elf/Dwarf 格式，可以加载到任意支持 Elf/Dwarf 格式的调试器中对程序进行调试，文件名后缀是(.out)。同样，其他编译器生成的带调试信息的 Elf/Dwarf 文件也可以加载到 IAR EWARM 进行调试。下面将说明如何操作。

首先要注意的是：

1. EWARM 的版本不能太老，必须是 2007 年之后发布的版本，在此之前的版本支持的可执行文件格式不是 Elf/Dwarf。
2. 要有工程的源代码，否则调试的时候只能查看到反汇编代码，不能查看 C/C++代码。编译调用的库文件只能查看反汇编代码。生成的 ELF 文件必须是标准的 Elf/Dwarf。

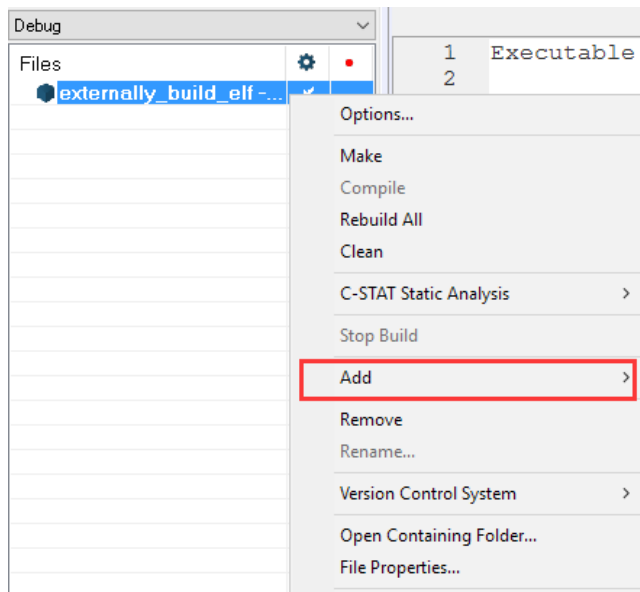
Step1. 在 IAR EWARM 中新建一个“Externally built executable”工程。单击 Project-> Create New Project，选中“Externally built executable”，单击 OK 确认。



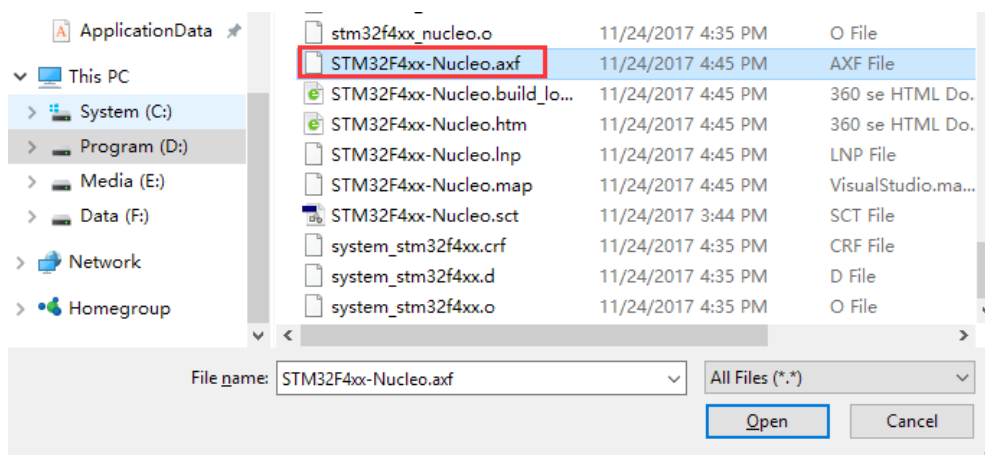
工程创建好之后会生成一个 Readme 文本，并自动打开，提示要做的一些操作。

```
readme.txt x
1 Executable files built outside of IAR Embedded Workbench
2
3 Follow these steps to load and debug an executable file that was built outside
4 IAR Embedded Workbench, one example is projects built on the command line.
5
6 1. Choose Project>Add Files to add the executable file (filename extension .out)
7 that you want to debug to the project.
8
9 2. Set up the debugger options according to you needs. The only project options
10 that are meaningful to set for this kind of project are options in the General
11 Options and Debugger categories. Make sure to set up the general project
12 options in the same way as when the executable file was built.
13
14 3. To start the debugger, select the project in the workspace window and
15 click the Debug button. The project can be reused whenever you rebuild your
16 executable file. The debugger will notice if the executable file has been
17 rebuilt and ask you if you want to reload it.
```

Step2. 添加 Elf/Dwarf 文件到工程中。在工程名上右键单击，选择 Add -> Add Files...，找到其他编译器生成的 Elf/Dwarf 文件，单击选中后按 Open 按钮。

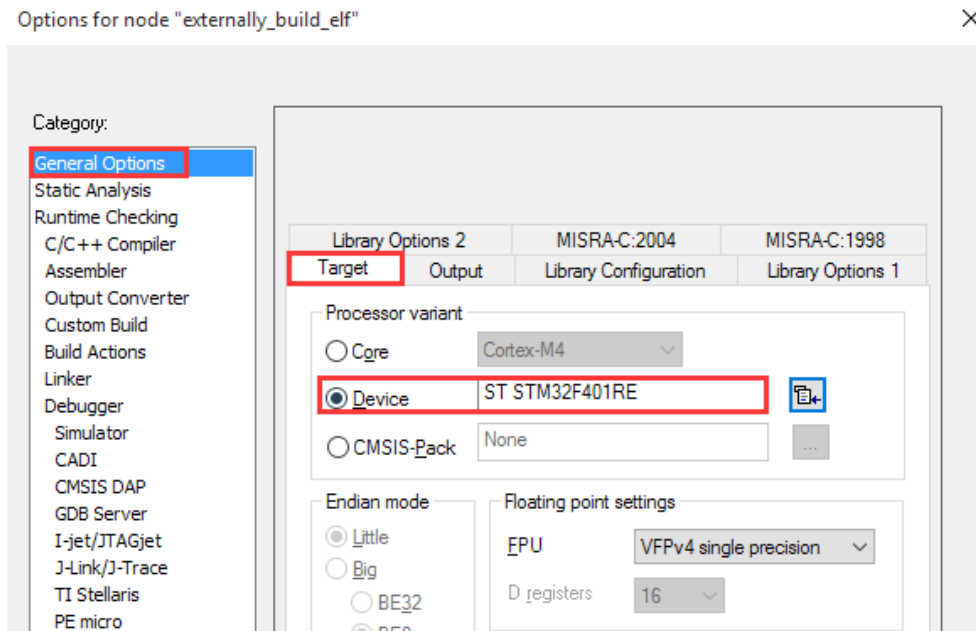


不同的编译器生成的 Elf/Dwarf 文件的后缀可能有区别，下图是 ARM MDK 生成的可执行文件，后缀是 (.axf)。

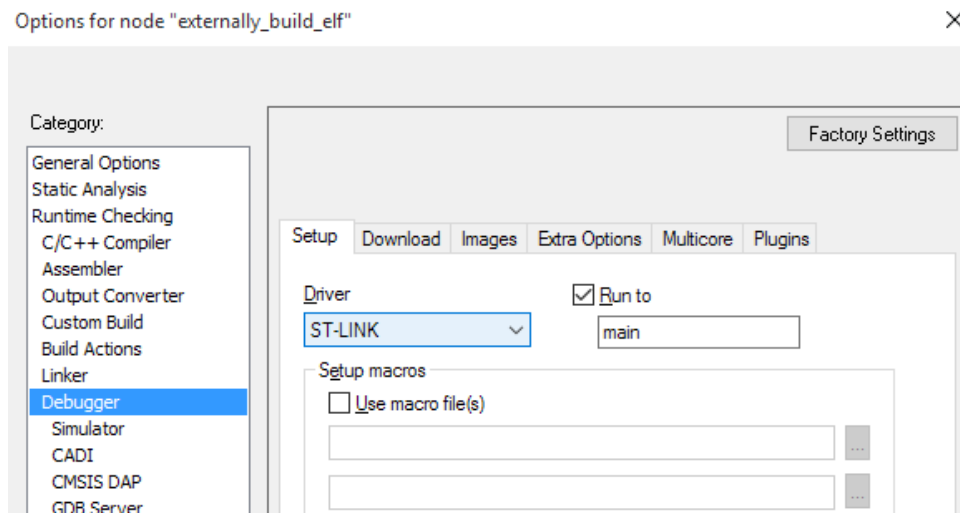


Note:源工程的目录和 C/C++ 源码文件都存在的情况下，即能够在 C 级别代码调试程序，推荐两种添加可执行文件的方式。(1)在直接定位源工程目录，找到可执行文件并添加。(2). 将源工程的目录复制一份到 IAR EWARM 新建工程的目录下，再从该目录添加。目的是保持源工程的目录结构不变，便于 IAR 识别源文件的路径。

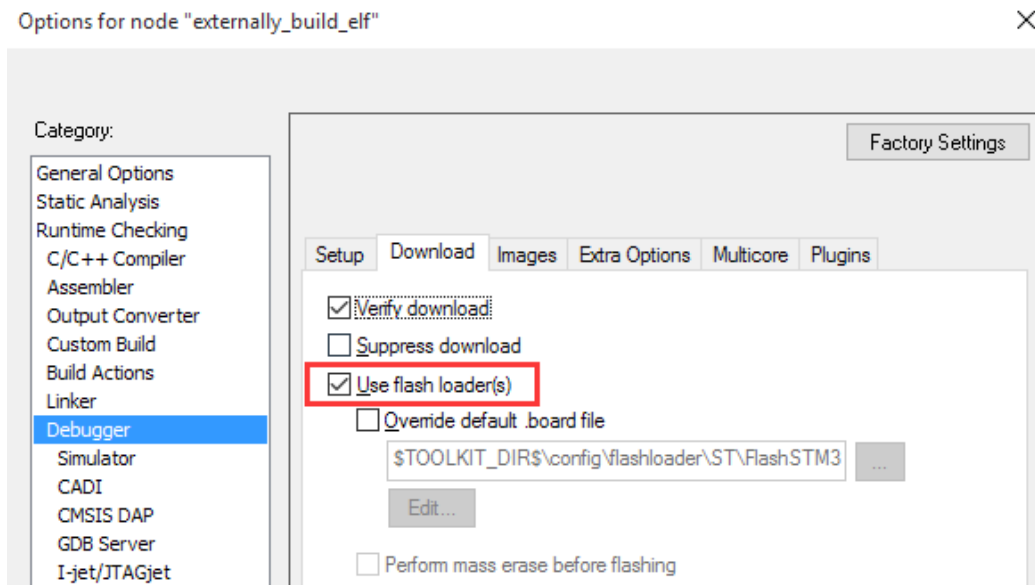
Step3. 在工程 Options 里面选择所使用的 MCU 型号。工程名上右键单击选择 Options.. -> General Options -> Target -> Device, 从列表中选择所使用的 MCU 型号。



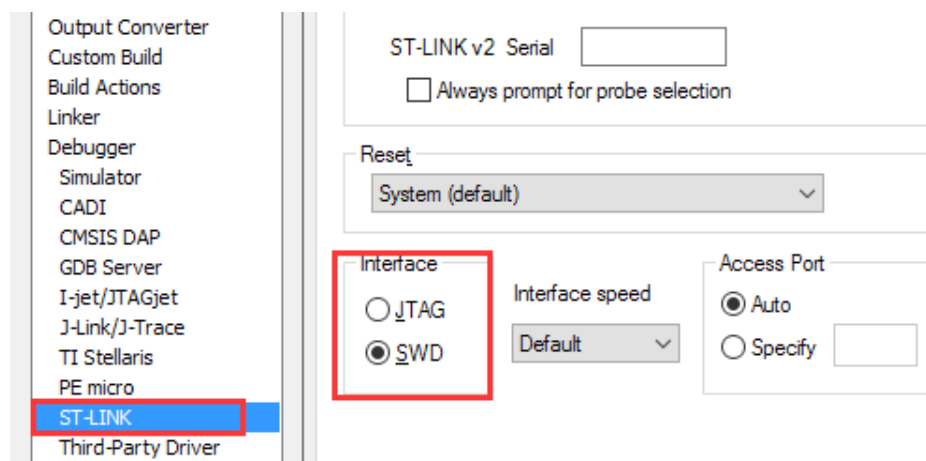
Step4. 根据使用的仿真器设置 Debugger 选项。在工程 Options ->Debugger 选择仿真器型号, 示例使用的是 ST-Link。



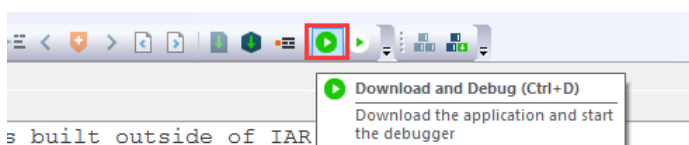
Step5. 设置 Flash Loader。在 Debugger ->Download 勾选 “Use flash loader(s)”，以便在下载调试的时候能够将程序下载到内部的 Flash。



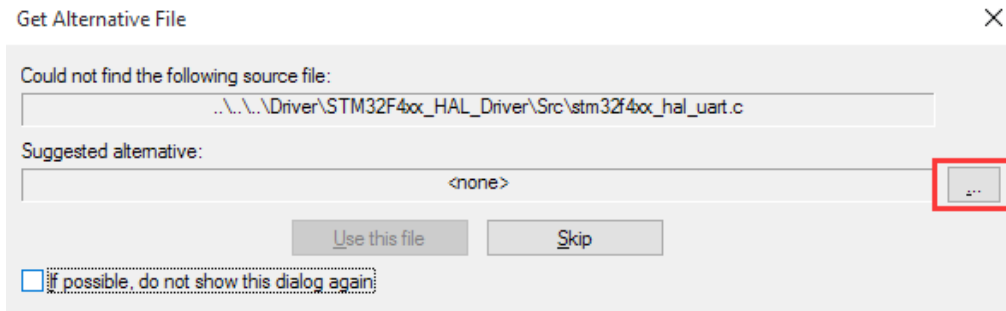
Step6. 设置硬件仿真器接口。在 Options 具体的硬件仿真器设置项中选择实际使用的调试仿真接口。



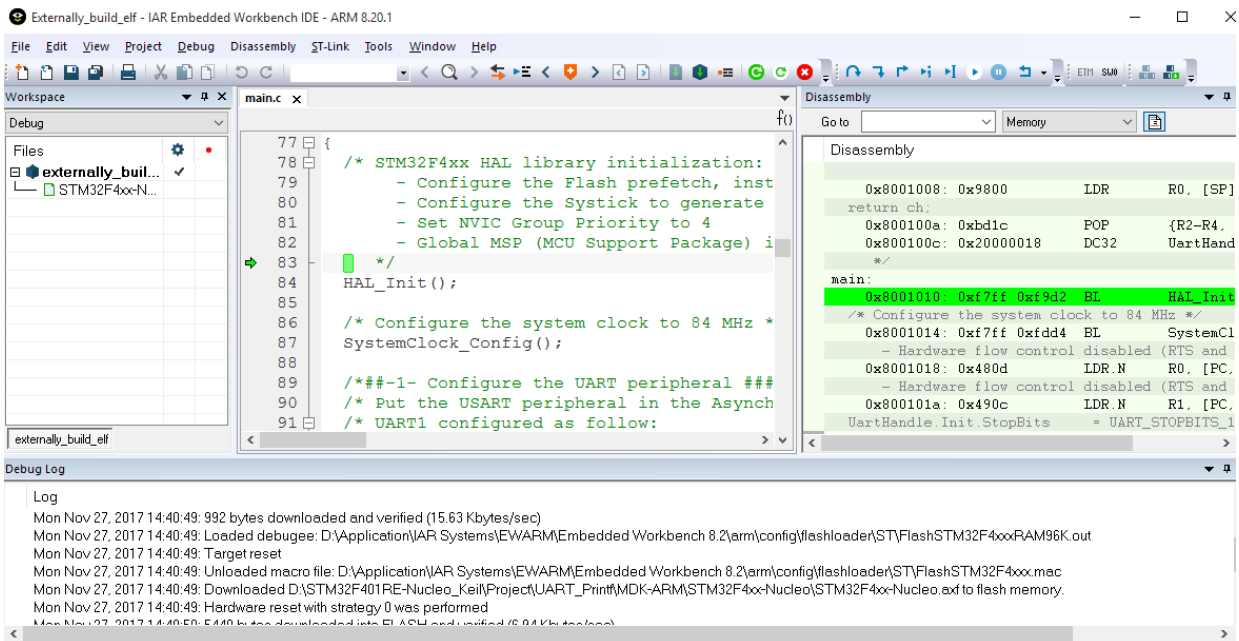
Step7. 进入调试环境。单击工具栏的 “下载和调试按钮”，程序会下载到 MCU，并进入调试界面。不同的版本 EWARM 的图标可能会有所不同，下图是 IAR EWARM 8.0 的图标样式。



Step8. 进入调试后，调试器可能会找不到源文件的目录。点击对话框上的按钮，找到所提示的源代码文件，然后选择“Use this file”。



手动指定未识别的源文件之后就可以正常调试了。



IAR 官方参考:

<https://www.iar.com/support/tech-notes/debugger/flashing-an-existing-image-without-risking-a-re-build-ewarm-5.x-6.x/>

版本	描述	作者	日期
1.0	创建	BMRTECH 付元斌	2017/11/27