




Gowin_EMPU_M3 IDE 软件 参考手册

IPUG919-1.1, 2021-07-16

版权所有 © 2021 广东高云半导体科技股份有限公司

GOWIN高云, , Gowin, 高云均为广东高云半导体科技股份有限公司注册商标, 本手册中提到的其他任何商标, 其所有权利属其拥有者所有。未经本公司书面许可, 任何单位和个人都不得擅自摘抄、复制、翻译本档内容的部分或全部, 并不得以任何形式传播。

免责声明

本档并未授予任何知识产权的许可, 并未以明示或暗示, 或以禁止发言或其它方式授予任何知识产权许可。除高云半导体在其产品的销售条款和条件中声明的责任之外, 高云半导体概不承担任何法律或非法律责任。高云半导体对高云半导体产品的销售和 / 或使用不作任何明示或暗示的担保, 包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等, 均不作担保。高云半导体对档中包含的文字、图片及其它内容的准确性和完整性不承担任何法律或非法律责任, 高云半导体保留修改档中任何内容的权利, 恕不另行通知。高云半导体不承诺对这些档进行适时的更新。

版本信息

| 日期 | 版本 | 说明 |
|------------|-----|--------------|
| 2020/04/03 | 1.0 | 初始版本。 |
| 2021/07/16 | 1.1 | 更新 MCU 软件版本。 |

目录

目录 i

图目录 ii

1 ARM Keil MDK 软件 1

 1.1 软件安装 1

 1.2 工程模板 1

 1.2.1 创建工程 1

 1.2.2 配置选项 2

 1.2.3 编译 7

 1.2.4 下载 8

 1.2.5 软件调试 9

 1.3 参考设计 9

2 GOWIN MCU Designer 10

 2.1 软件安装与配置 10

 2.2 工程模板 10

 2.2.1 创建工程 10

 2.2.2 配置选项 12

 2.2.3 编译 16

 2.2.4 下载 17

 2.2.5 调试 18

 2.3 参考设计 20

图目录

| | |
|---|----|
| 图 1-1 创建工程 | 1 |
| 图 1-2 配置器件 | 2 |
| 图 1-3 配置 ROM 和 RAM..... | 3 |
| 图 1-4 配置输出文件格式..... | 3 |
| 图 1-5 配置 C 头文件路径..... | 4 |
| 图 1-6 配置仿真器类型 | 5 |
| 图 1-7 配置调试接口类型..... | 6 |
| 图 1-8 配置 Flash 选项 | 6 |
| 图 1-9 配置调试初始化文件 | 7 |
| 图 1-10 编译 | 7 |
| 图 1-11 下载..... | 8 |
| 图 1-12 启动调试 | 9 |
| 图 2-1 创建工程 | 11 |
| 图 2-2 选择平台配置类型..... | 11 |
| 图 2-3 选择编译工具链和路径 | 12 |
| 图 2-4 配置 Target Processor | 13 |
| 图 2-5 配置 GNU ARM Cross Assembler | 13 |
| 图 2-6 配置 GNU ARM Cross C Compiler..... | 14 |
| 图 2-7 配置 GNU ARM Cross C Linker..... | 15 |
| 图 2-8 配置 GNU ARM Cross Create Flash Image | 15 |
| 图 2-9 配置 Devices..... | 16 |
| 图 2-10 编译 | 16 |
| 图 2-11 下载..... | 17 |
| 图 2-12 创建调试配置选项..... | 18 |
| 图 2-13 配置 Main 选项 | 18 |
| 图 2-14 配置 Debugger 选项 | 19 |
| 图 2-15 启动调试 | 19 |

1 ARM Keil MDK 软件

1.1 软件安装

请参考 ARM Keil MDK 网站提供的 [Getting Started with MDK](#)。

注！

建议使用 ARM Keil MDK V5.26 及以上版本。

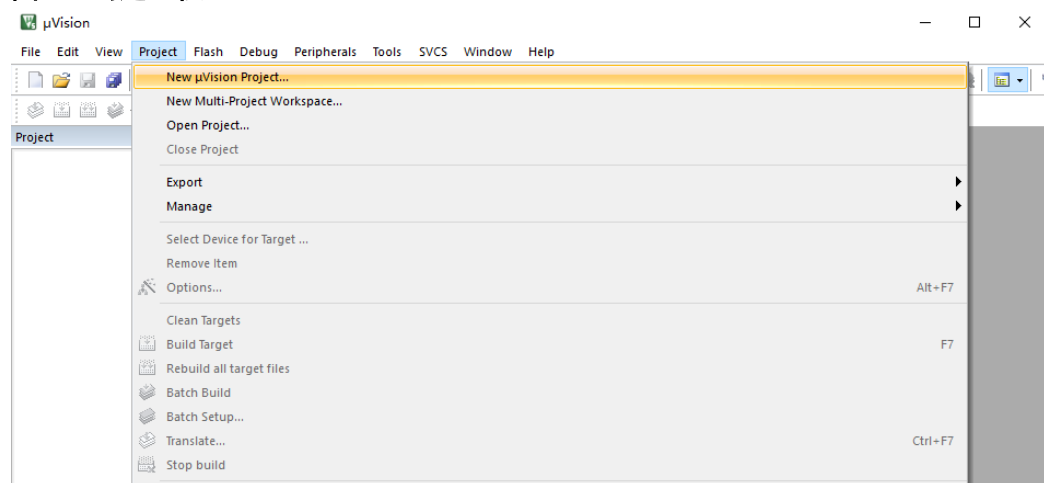
1.2 工程模板

使用 ARM Keil MDK 软件进行 Gowin_EMPU_M3 软件编程设计，需要创建工程、配置选项、编写代码、编译、下载和调试。

1.2.1 创建工程

打开 ARM Keil MDK 软件，选择菜单栏“Project > New uVision Project...”，创建工程，如图 1-1 所示。

图 1-1 创建工程

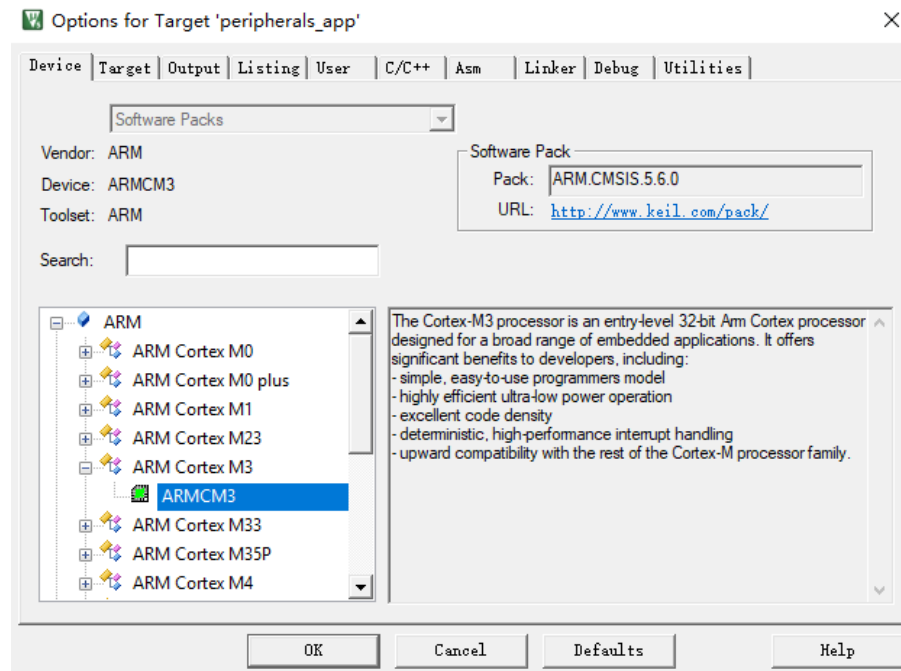


1.2.2 配置选项

配置器件

Gowin_EMPU_M3 内置 ARM Cortex-M3 内核，配置器件为“ARM Cortex M3 > ARMCM3”，如图 1-2 所示。

图 1-2 配置器件



配置 ROM 和 RAM

Gowin_EMPU_M3 的 Instruction Memory 作为 ROM。

Gowin_EMPU_M3 的 Data Memory 作为 RAM。

配置 ROM（Instruction Memory）和 RAM（Data Memory）的起始地址和 Size，如图 1-3 所示。

ROM 配置：

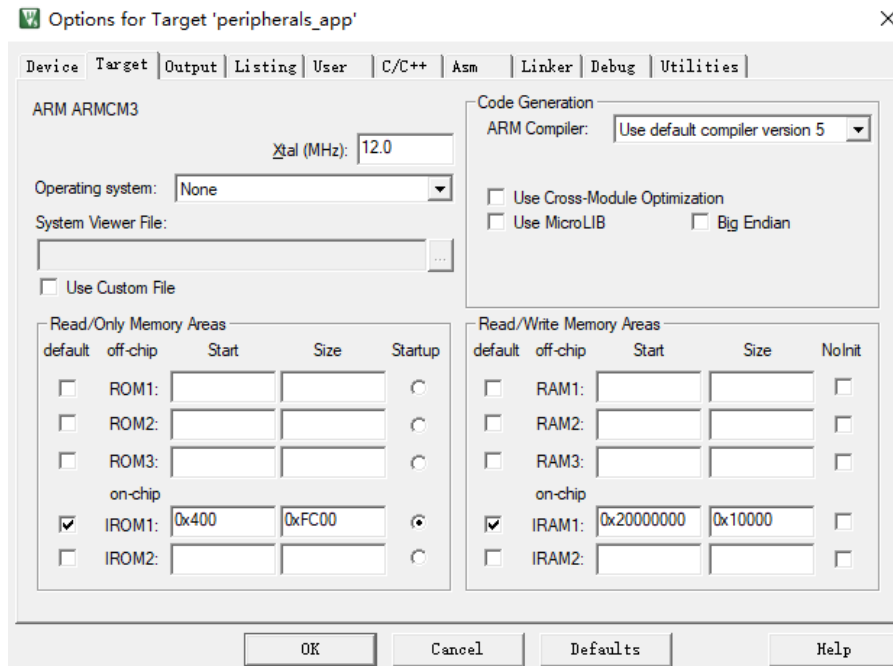
- 片外 SPI-Flash 下载启动方式，ROM 起始地址为 0x400
- Size 根据 Gowin 云源软件的 IP Core Generator > Gowin_EMPU_M3 > Instruction Memory Size 实际配置设定，可以配置为 16KB、32KB、64KB 或 128KB
- 以软件开发工具包 Keil_RefDesign 参考设计为例，ROM Size 配置为 0xFC00（Gowin 云源软件的 IP Core Generator > Gowin_EMPU_M3 > Instruction Memory Size 配置为 64KB）

RAM 配置：

- RAM 起始地址为 0x20000000
- Size 根据 Gowin 云源软件的 IP Core Generator > Gowin_EMPU_M3 > Data Memory Size 实际配置设定，可以配置为 16KB、32KB、64KB 或 128KB

- 以软件开发工具包 Keil_RefDesign 参考设计为例，RAM Size 配置为 0x10000（Gowin 云源软件的 IP Core Generator > Gowin_EMPU_M3 > Data Memory Size 配置为 64KB）

图 1-3 配置 ROM 和 RAM

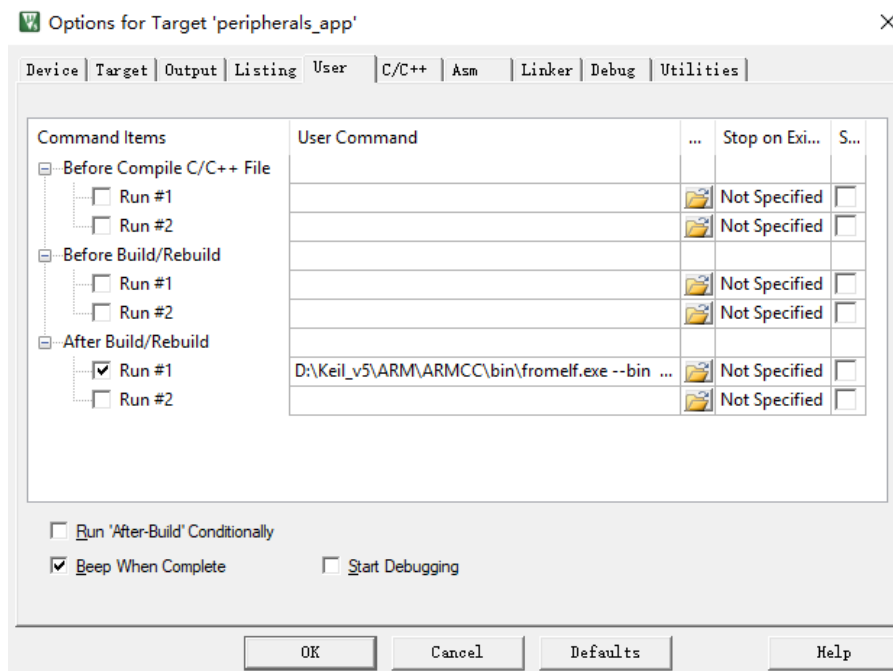


配置输出文件格式

Gowin 下载软件 Programmer，支持 BIN 二进制文件格式，配置 Gowin_EMPU_M3 软件输出文件格式为 BIN。

- User 命令行选项中文件格式转换工具调用方法，如图 1-4 所示。
- 格式转换命令为：Run #1: fromelf.exe --bin -o bin-file axf-file

图 1-4 配置输出文件格式



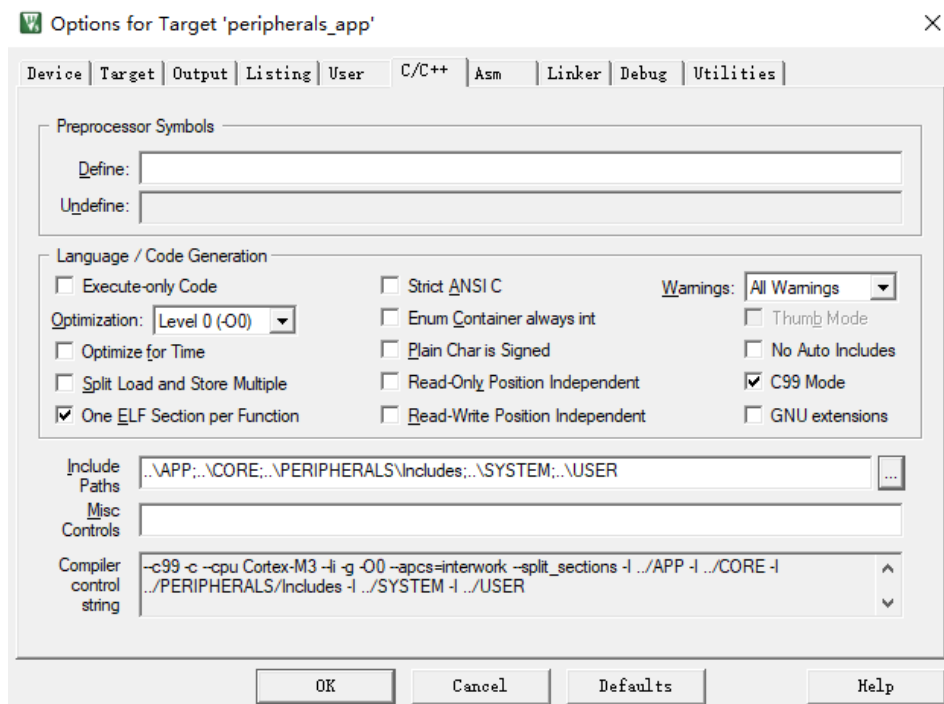
配置头文件路径

配置 C 头文件路径，编译过程中用来调用不同路径的 C 头文件，配置如图 1-5 所示。

以软件开发工具包 Keil_RefDesign 参考设计为例，C 头文件路径配置，如下所示。

- "..\CORE"
- "..\PERIPHERALS\Includes"
- "..\SYSTEM"
- "..\USER"
- "..\APP"

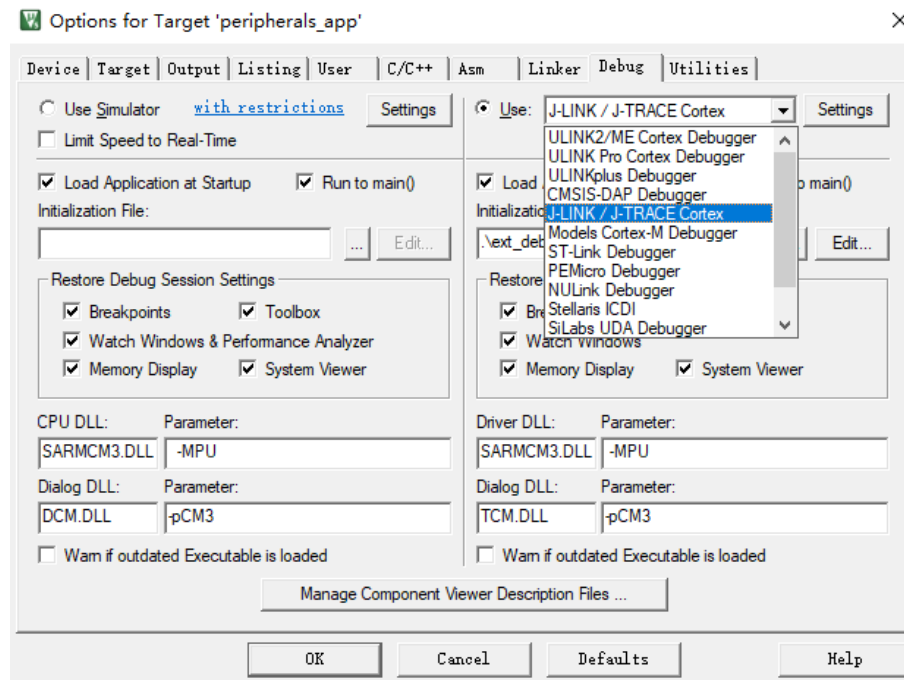
图 1-5 配置 C 头文件路径



配置调试选项

- 配置仿真器类型
 - 单击 **Debug** 仿真器类型下拉列表，选择所用仿真器类型，如**错误!未找到引用源。**所示。
 - U-LINK 仿真器
 - 如果选择使用 U-LINK 仿真器，则 **Debug** 选项配置为 **ULNK2/ME Cortex Debugger**。
 - J-LINK 仿真器
 - 如果选择使用 J-LINK 仿真器，则 **Debug** 选项配置为 **J-LINK/J-TRACE Cortex**。

图 1-6 配置仿真器类型



- 配置调试接口类型

例如，选择 J-LINK/J-TRACE Cortex，单击 Settings，打开 Cortex JLink/JTrace Target Driver Setup，选择所用调试接口类型。

- JTAG 接口

如果 Gowin 云源软件的 IP CoreGenerator > Gowin_EMPU_M3 > Debug Interface 使能 JTAG 接口，则配置 SWJ Port 为 JTAG 调试接口。

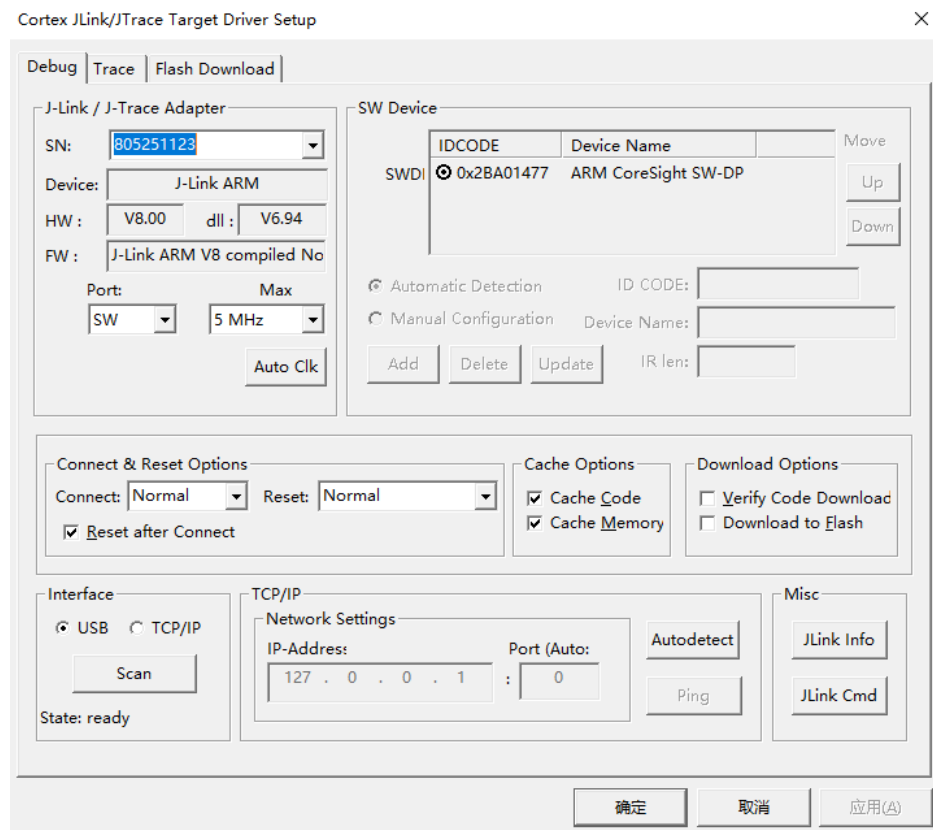
- SW 调试接口

如果 Gowin 云源软件的 IP CoreGenerator > Gowin_EMPU_M3 > Debug Interface 关闭 JTAG 接口，则配置 SWJ Port 为 SW 调试接口。

Download Options 配置选项，请不要勾选“Verify Code Download”选项和“Download to Flash”选项。

如果调试选项配置成功，连接 J-LINK 或 U-LINK 仿真器后，JTAG Device Chain 可以正确显示 Gowin_EMPU_M3 的 IDCODE、Device Name 等信息，如图 1-7 所示。

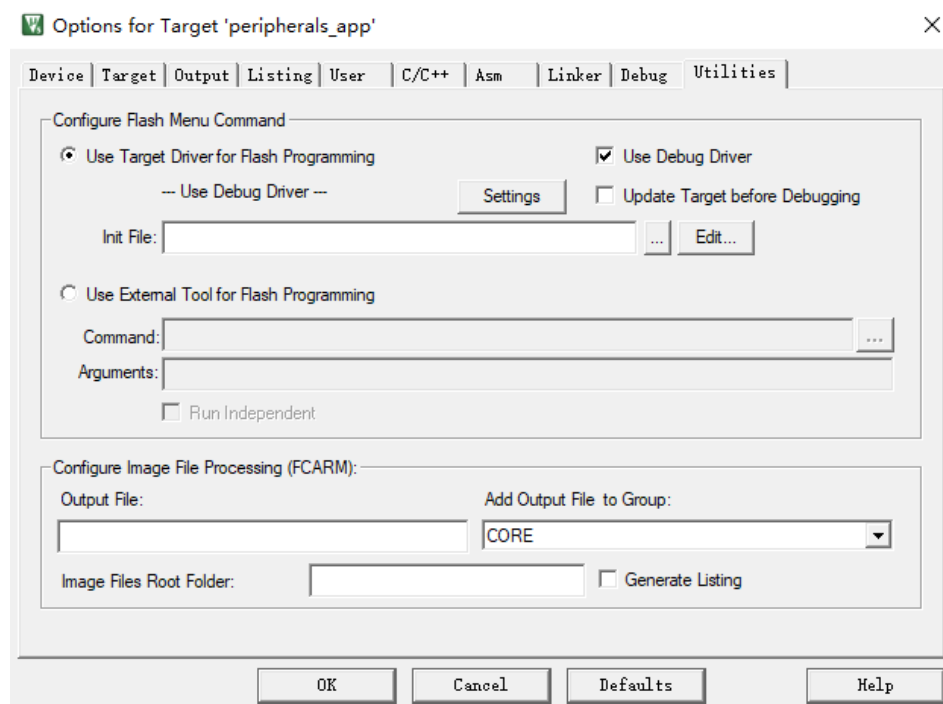
图 1-7 配置调试接口类型



配置 Flash 选项

请不要勾选 “Utilities > Update Target before Debugging” 选项，如图 1-8 所示。

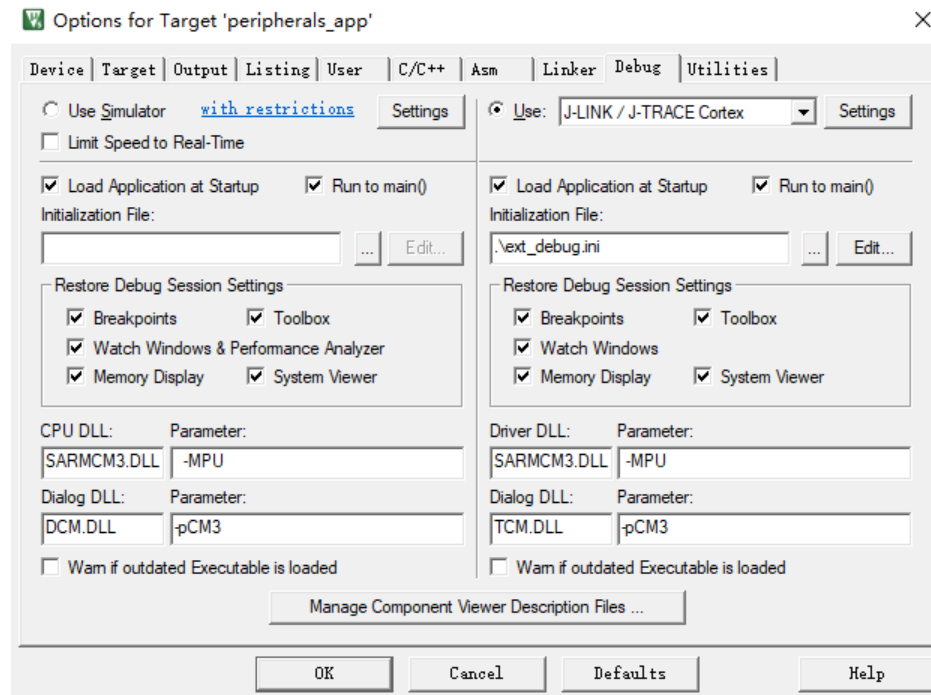
图 1-8 配置 Flash 选项



配置调试初始化文件

Gowin_EMPU_M3 软件调试时，请加载调试初始化文件，“Debug > Initialization File”选项，加载调试初始化文件“ext_debug.ini”，如图 1-9 所示。

图 1-9 配置调试初始化文件



1.2.3 编译



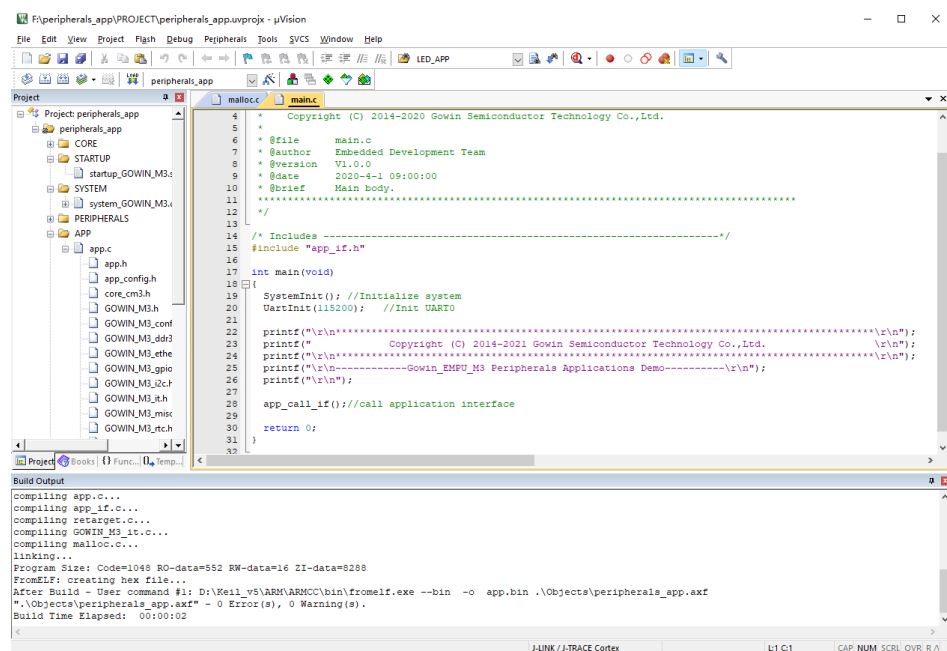
完成代码编写和选项配置后，单击工具栏 Build “”或 Rebuild “”，编译生成软件编程二进制 BIN 文件，如图 1-10 所示。

图 1-10 编译



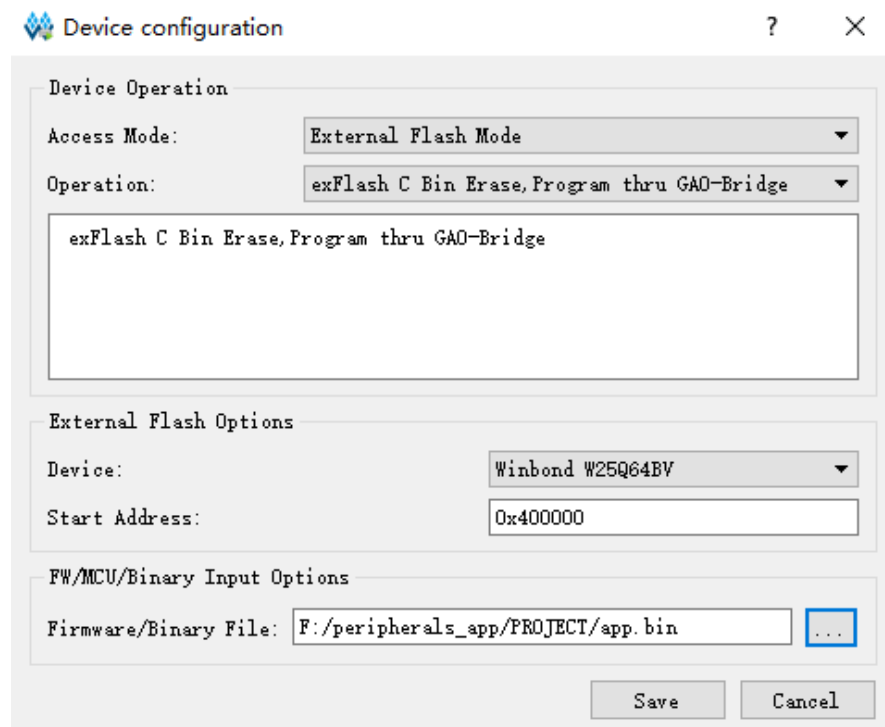
1.2.4 下载


完成 Gowin_EMPU_M3 软件程序设计编译后，使用 Gowin 下载软件 Programmer，下载软件编程二进制 BIN 文件。

由 Gowin 云源软件中或软件安装路径下打开 Programmer，单击 Programmer 菜单栏“Edit > Configure Device”或工具栏 Configure Device “

- Access Mode 下拉列表，选择“External Flash Mode”选项。
- Operation 下拉列表，选择“exFlash C Bin Erase, Program thru GAO-Bridge”或“exFlash C Bin Erase, Program, Verify thru GAO-Bridge”选项。
- FW/MCU/Binary Input Options > Firmware/Binary File 选项，导入需要下载的 Gowin_EMPU_M3 软件编程二进制 BIN 文件。
- External Flash Options > Device 选项，请根据开发板板载 Flash 芯片类型选择（如高云开发板板载 Winbond W25Q64BV）。
- External Flash Options > Start Address 选项，下载起始地址设置为“0x400000”。
- 单击“Save”，如图 1-11 所示。

图 1-11 下载



完成 Device configuration 后，单击 Programmer 工具栏 Program/Configure “IPUG919-1.1

1.2.5 软件调试

完成 Gowin_EMPU_M3 软件编程设计下载后，如果用户软件设计出现问题，可以连接 U-LINK 或 J-LINK 仿真器在线调试。

1. 连接仿真器

按照硬件设计中约束到 FPGA IO 的 Debug Access Port 的位置，连接 J-LINK 或 U-LINK 仿真器。

2. 启动调试


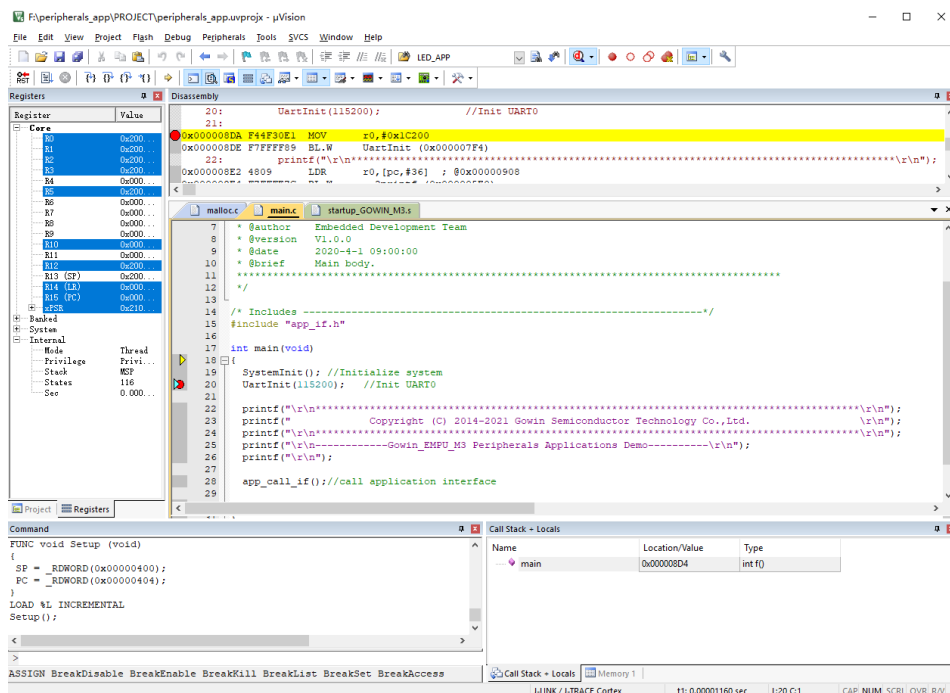
连接 U-LINK 或 J-LINK 仿真器后，单击工具栏 Debug “”，进入调试状态，可以进行断点设置、单步调试、复位和运行等操作，如图 1-12 所示。

图 1-12 启动调试



1.3 参考设计

Gowin_EMPU_M3 支持 ARM Keil MDK (V5.26 及以上版本) 软件环境的参考设计，通过链接获取如下参考设计：

Gowin_EMPU_M3\ref_design\MCU_RefDesign\Keil_RefDesign

2 GOWIN MCU Designer

2.1 软件安装与配置

高云半导体网站提供 GOWIN MCU Designer 软件安装包下载
<http://www.gowinsemi.com.cn/prodshow.aspx>。

GOWIN MCU Designer 软件安装与配置，请参考 [SUG549](#), GOWIN MCU Designer 用户指南。

注！

建议使用 GOWIN MCU Designer V1.1 及以上版本。

2.2 工程模板

使用 GOWIN MCU Designer 软件进行 Gowin_EMPU_M3 软件编程设计，需要创建工程、配置选项、编写代码、编译、下载和调试。

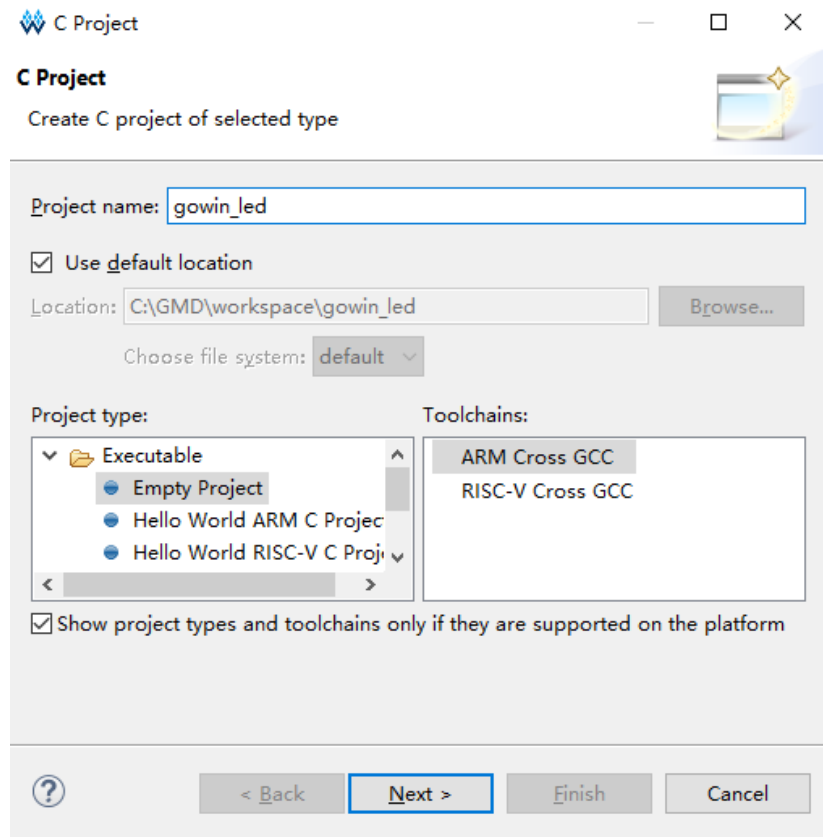
2.2.1 创建工程

创建工程

选择菜单栏“File > New > C Project”，如图 2-1 所示。

1. 建立项目名称和位置；
2. 选择项目类型 Empty Project；
3. 选择工具链 ARM Cross GCC。

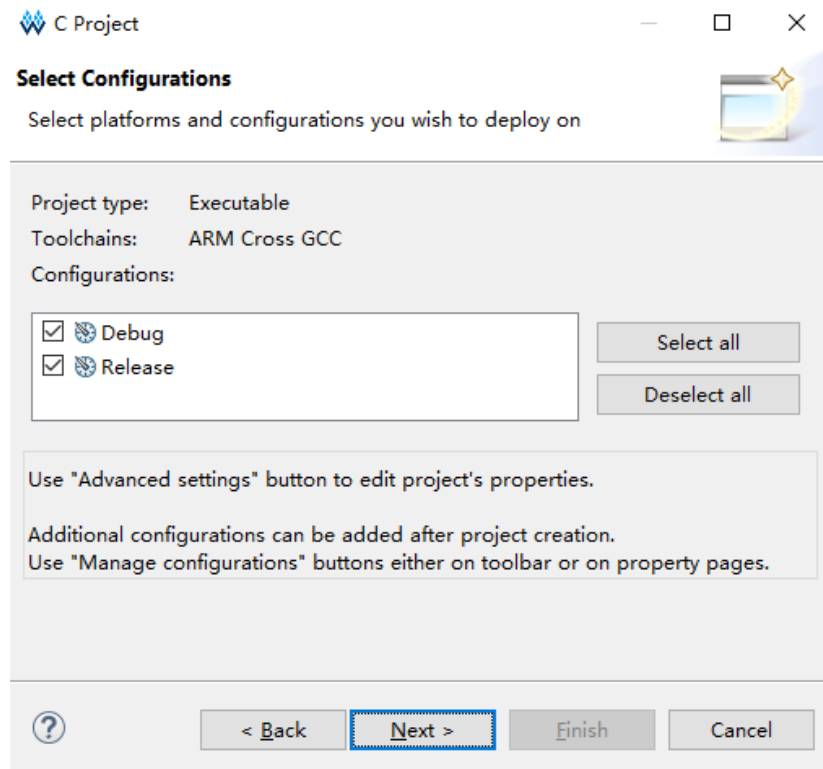
图 2-1 创建工程



选择平台配置类型

选择平台配置类型“Debug”和“Release”，如图 2-2 所示。

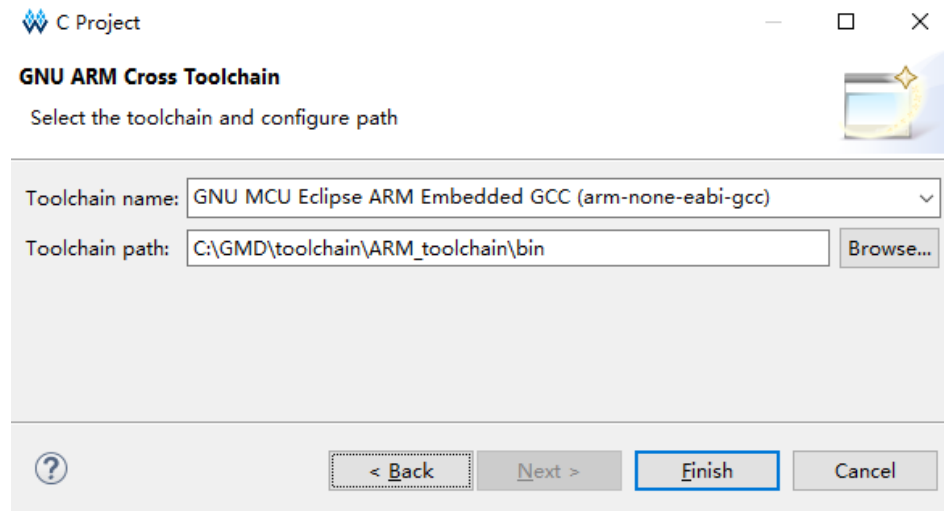
图 2-2 选择平台配置类型



选择编译工具链和路径

选择交叉编译工具链 `arm-none-eabi-gcc` 和其所在路径，默认配置，如图 2-3 所示。

图 2-3 选择编译工具链和路径



建立项目工程

完成工程创建后，选择 GOWIN MCU Designer 工作空间 `workspace` 下新建的项目工程，添加工程结构和代码，导入软件编程设计。

以软件开发工具包 `GMD_RefDesign` 参考设计为例，软件编程设计目录及代码如下所示。

- CORE: ARM Cortex-M3 内核寄存器定义
- PERIPHERALS: 外部设备驱动函数库
- STARTUP: 引导启动文件
- SYSTEM: 寄存器定义、系统初始化和系统时钟定义
- USER: 用户应用设计
- Script/GOWIN_M3_flash.Id: Flash 链接器

完成工程结构创建后，在 GOWIN MCU Designer 的 Project Explorer 视图中选择当前工程，右键选择“Refresh”选项，自动更新当前工程的结构和代码。

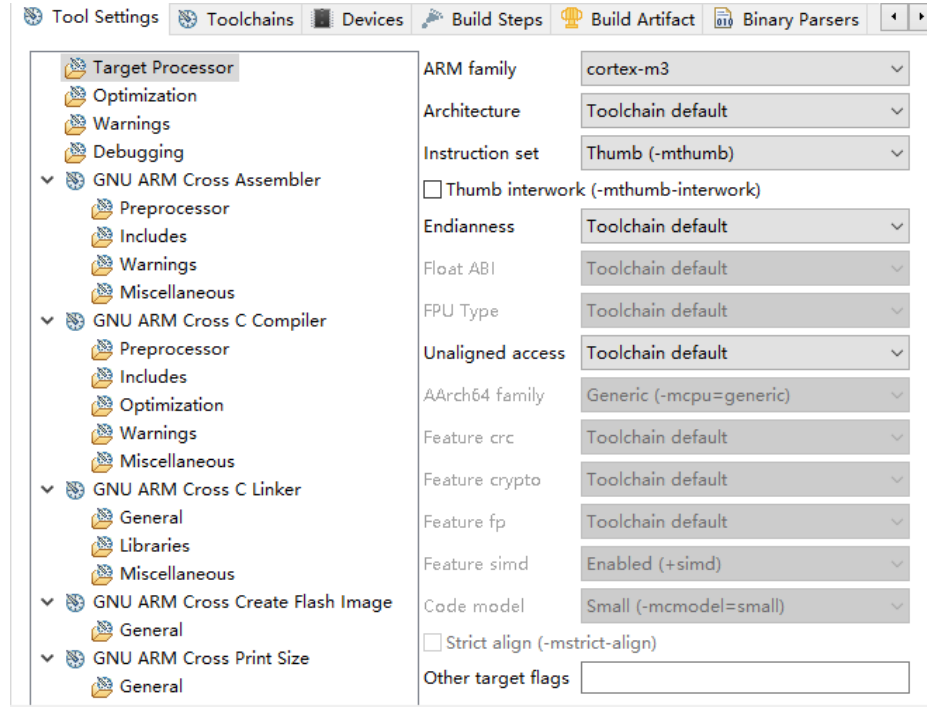
2.2.2 配置选项

GOWIN MCU Designer 的 Project Explorer 视图中，选择当前工程，右键选择“Properties > C/C++ Build > Settings”选项，配置当前工程的参数选项。

配置 Target Processor

配置“Target Processor > ARM family”选项，该选项配置为“cortex-m3”，如图 2-4 所示。

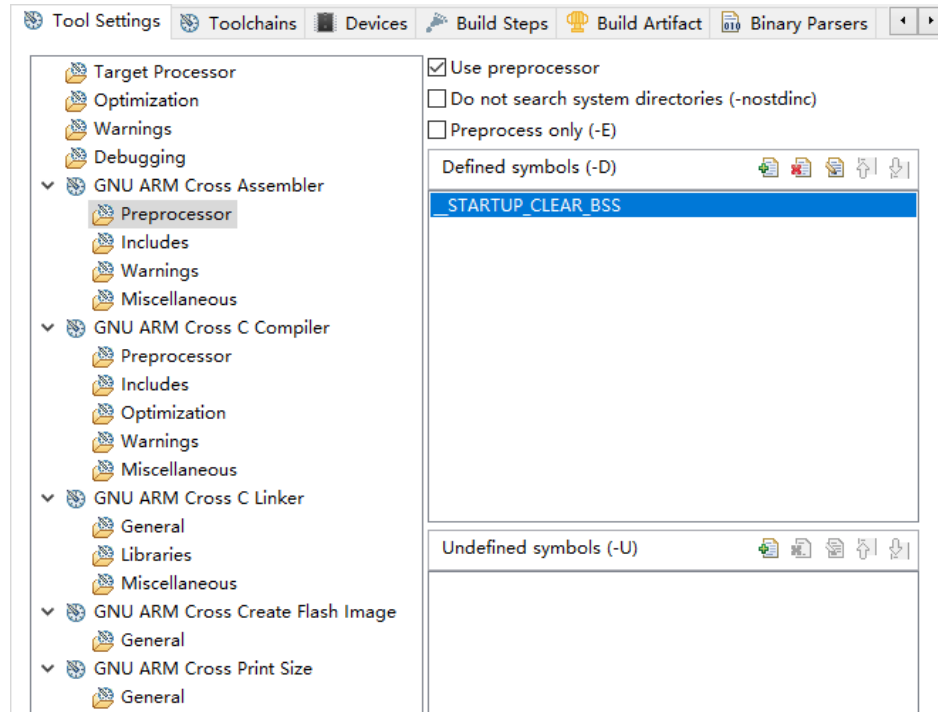
图 2-4 配置 Target Processor



配置 GNU ARM Cross Assembler

配置 “GNU ARM Cross Assembler > Preprocessor > Defined symbols (-D)” 选项，该选项配置为 “__STARTUP_CLEAR_BSS”，如图 2-5 所示。

图 2-5 配置 GNU ARM Cross Assembler



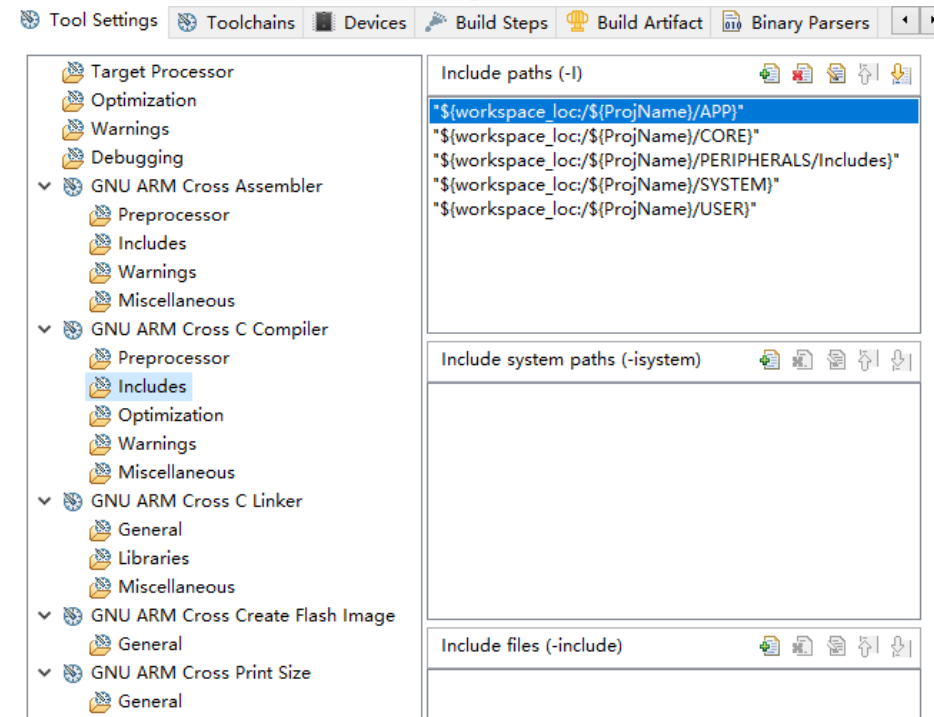
配置 GNU ARM Cross C Compiler

配置 “GNU ARM Cross C Compiler > Includes > Include paths (-I)” 选项，该选项配置为当前工程 C 头文件引用路径，如图 2-6 所示。

以软件开发工具包 GMD_RefDesign 参考设计为例,C 头文件路径配置,如下所示。

- "\${workspace_loc}/\${ProjName}/CORE}"
- "\${workspace_loc}/\${ProjName}/PERIPHERALS/Includes}"
- "\${workspace_loc}/\${ProjName}/SYSTEM}"
- "\${workspace_loc}/\${ProjName}/USER}"
- "\${workspace_loc}/\${ProjName}/APP}"

图 2-6 配置 GNU ARM Cross C Compiler



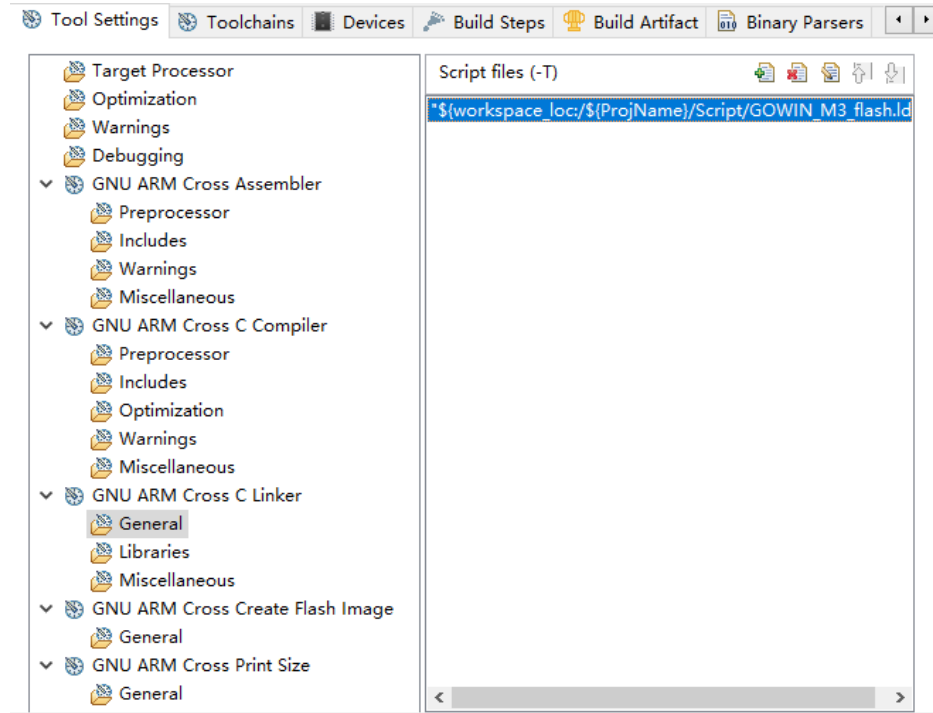
配置 GNU ARM Cross C Linker

配置“GNU ARM Cross C Linker > General > Script files (-T)”选项,该选项配置为 Flash 链接器 GOWIN_M3_flash.ld”,如图 2-7 所示。

以软件开发工具包 GMD_RefDesign 参考设计为例,Flash 链接器配置,如下所示。

```
"${workspace_loc}/${ProjName}/Script/GOWIN_M3_flash.ld}"
```

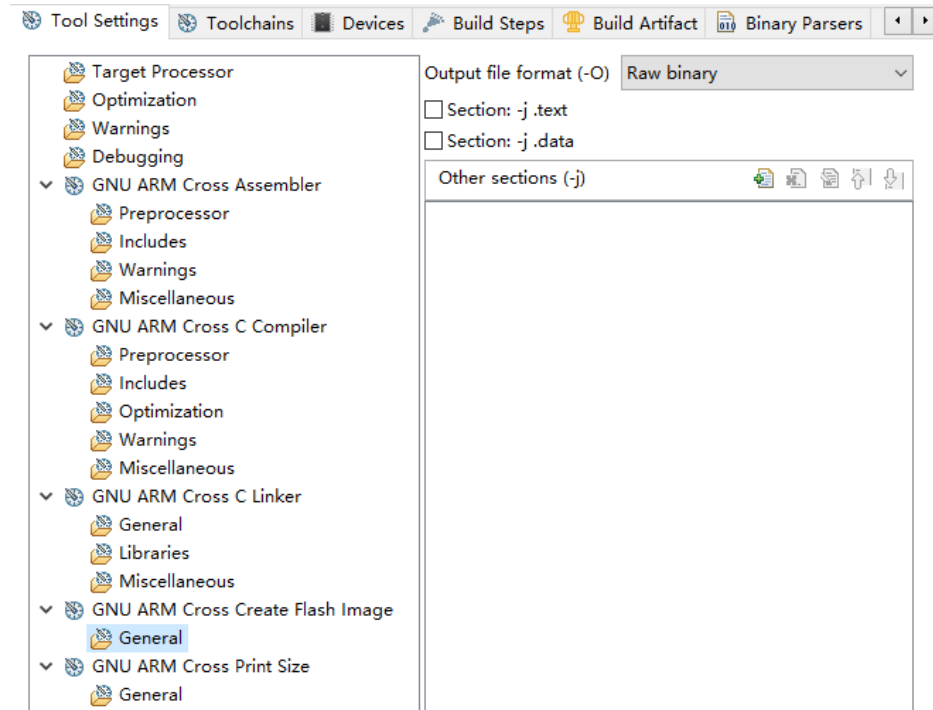
图 2-7 配置 GNU ARM Cross C Linker



配置 GNU ARM Cross Create Flash Image

配置 “GNU ARM Cross Create Flash Image > General > Output file format (-O)” 选项，该选项配置为 “Raw binary”，如图 2-8 所示。

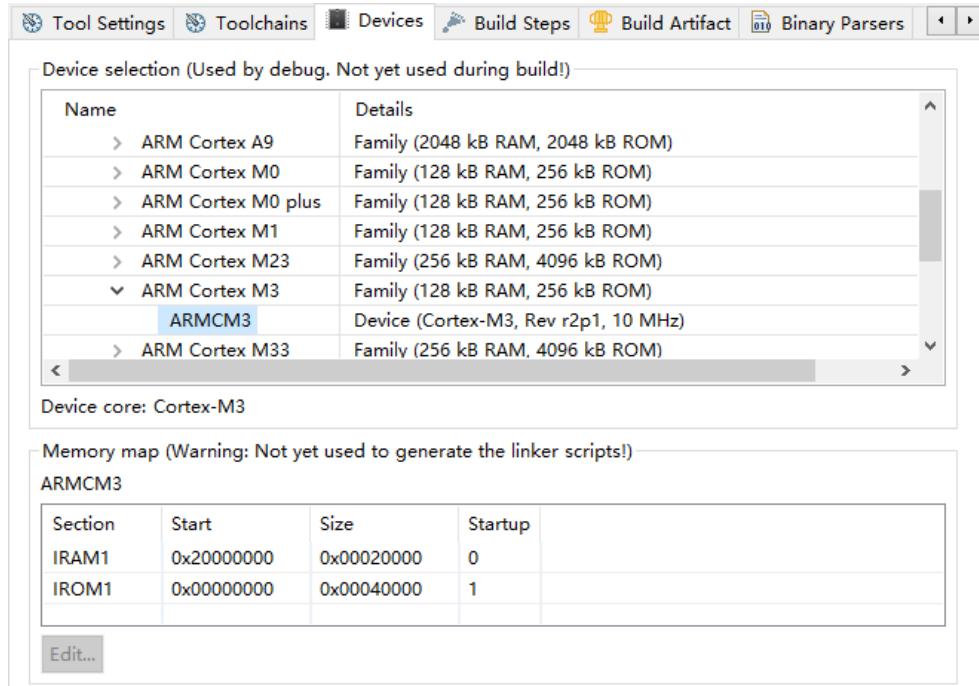
图 2-8 配置 GNU ARM Cross Create Flash Image



配置 Devices

配置“Devices > Device selection”选项，配置器件为“ARM Cortex M3 > ARMCM3”，如图 2-9 所示。

图 2-9 配置 Devices



2.2.3 编译


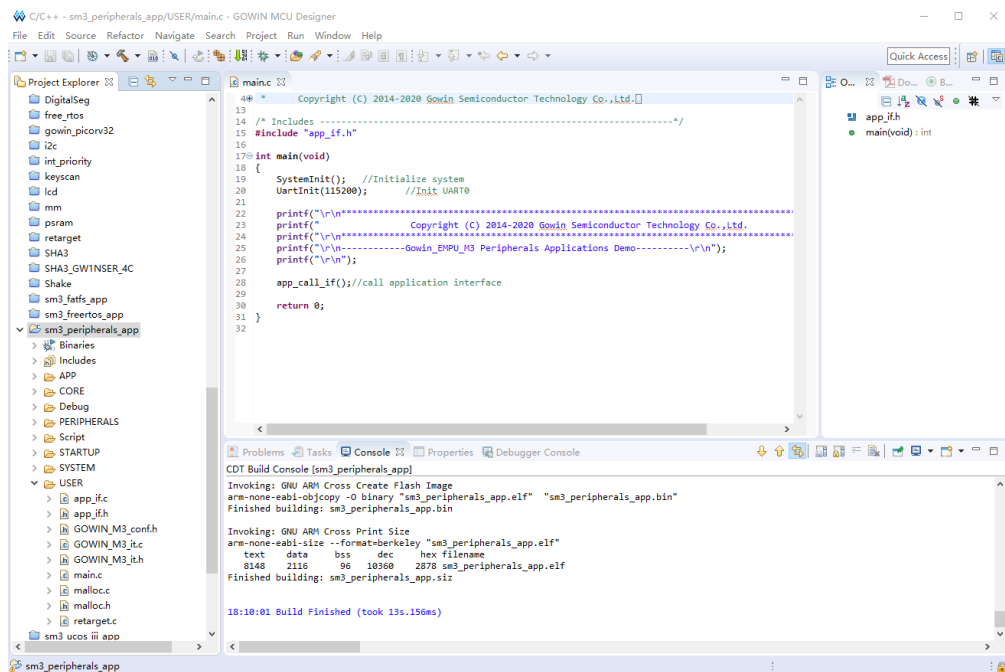

完成工程选项配置和代码编写后，编译当前工程，单击工具栏编译按钮“”，编译生成 Gowin_EMPU_M3 软件编程二进制 BIN 文件，如图 2-10 所示。


图 2-10 编译



2.2.4 下载

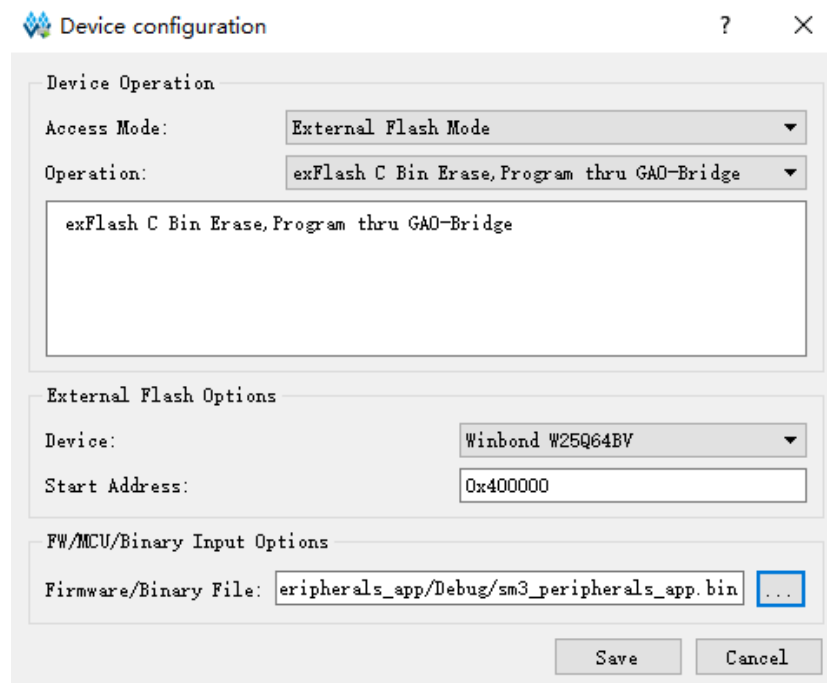
完成 Gowin_EMPU_M3 软件编程设计编译后，使用 Gowin 下载软件 Programmer 下载软件编程二进制 BIN 文件。


由 GOWIN MCU Designer 菜单栏“Run > Programmer”或工具栏“”，打开 Programmer。

单击 Programmer 菜单栏“Edit > Configure Device”或工具栏 Configure Device “”，打开 Device configuration。

- Access Mode 下拉列表，选择“External Flash Mode”选项。
- Operation 下拉列表，选择“exFlash C Bin Erase, Program thru GAO-Bridge”或“exFlash C Bin Erase, Program, Verify thru GAO-Bridge”选项。
- FW/MCU/Binary Input Options > Firmware/Binary File 选项，导入需要下载的 Gowin_EMPU_M3 软件编程二进制 BIN 文件。
- External Flash Options > Device 选项，请根据开发板板载 Flash 芯片类型选择（如高云开发板板载 Winbond W25Q64BV）。
- External Flash Options > Start Address 选项，下载起始地址设置为“0x400000”。
- 单击“Save”，如图 2-11 所示。

图 2-11 下载



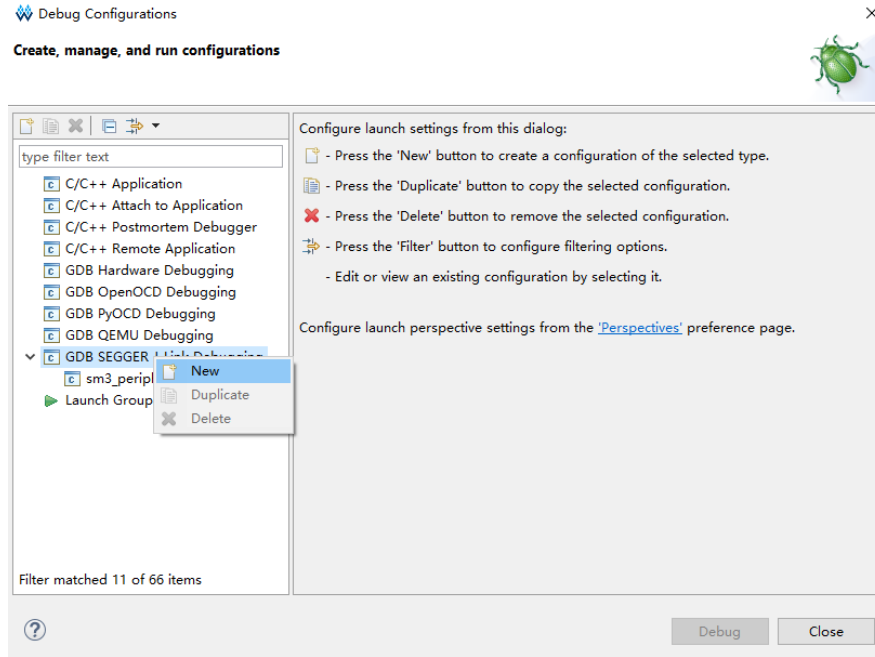
完成 Device configuration 后，单击 Programmer 工具栏 Program/Configure “”，完成 Gowin_EMPU_M3 软件编程二进制 BIN 文件下载。

2.2.5 调试

配置调试选项

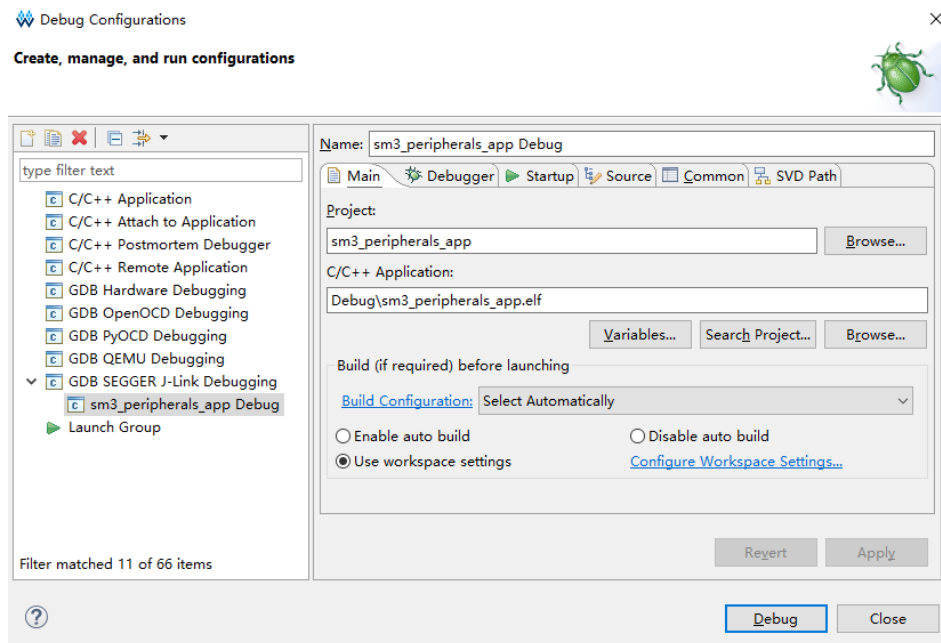
1. 选择菜单栏 “Run > Debug Configurations > GDB SEGGER J-Link Debugging”，右键选择 “New”，创建当前工程的调试配置选项，如图 2-12 所示。

图 2-12 创建调试配置选项



2. 选择 “Main” 选项，配置当前工程的输出映像文件等信息，如图 2-13 所示。

图 2-13 配置 Main 选项

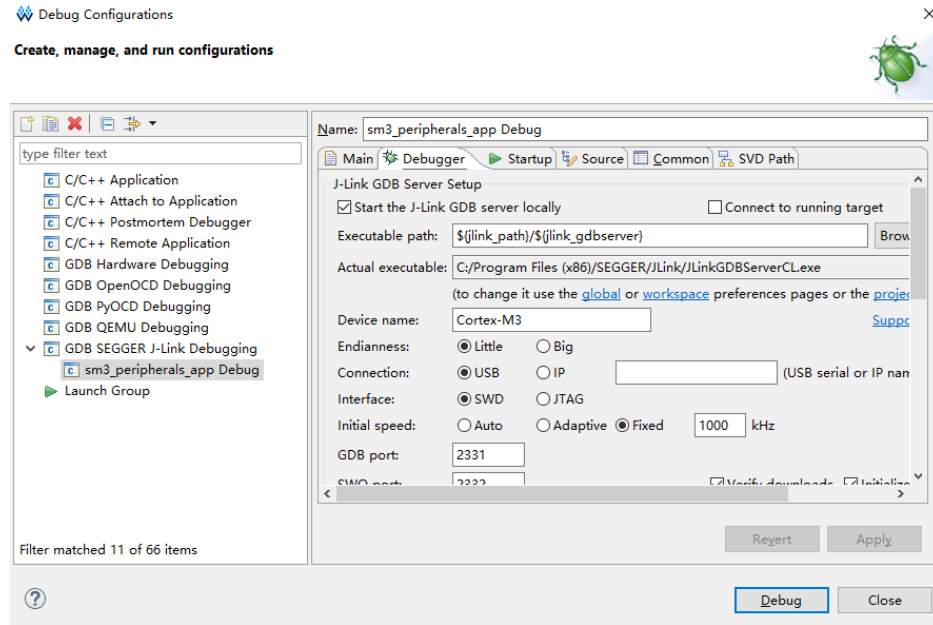


3. 选择 “Debugger” 选项，配置 Debugger 器件名称和调试接口等信息，

如图 2-14 所示。

- Device Name: Cortex-M3
- Interface: JTAG 或 SWD
- Endianness: Little
- Connection: USB

图 2-14 配置 Debugger 选项



软件调试


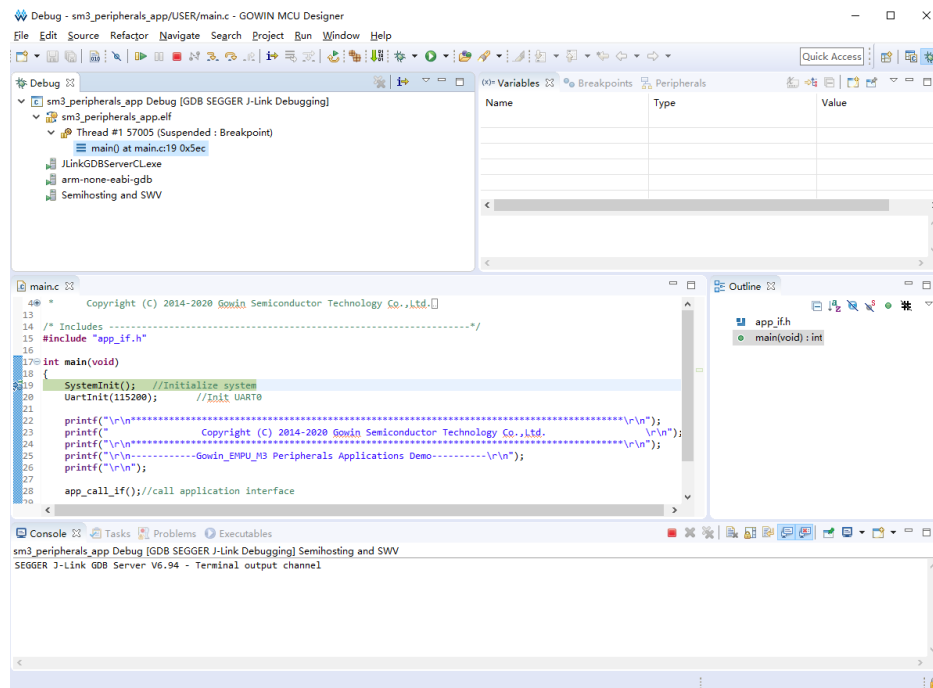
连接 J-LINK 仿真器与开发板,单击工具栏 Debug 按钮下拉列表“”,选择当前项目的 Debug 配置,单击进入调试状态,可以进行断点设置、单步调试、复位和运行等操作,如图 2-15 所示。

图 2-15 启动调试



2.3 参考设计

Gowin_EMPU_M3 支持 GOWIN MCU Designer (V1.1 及以上版本) 软件环境的参考设计，通过链接获取如下参考设计：

Gowin_MCU_M3\ref_design\MCU_RefDesign\GMD_RefDesign

