



# Gowin FPGA 开发板 RISC-V 编程 快速应用手册

IPUG546-1.1,2022-11-11

版权所有 © 2022 广东高云半导体科技股份有限公司

 GOWIN高云、Gowin以及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
2019/04/29	1.0	初始版本。
2022/11/11	1.1	<ul style="list-style-type: none"><li>● 更新 AndeSight RDS v311 软件；</li><li>● 更新参考设计；</li><li>● 补充描述嵌入式工程编译结果 SPI Flash 下载启动方法。</li></ul>

# 目录

目录 .....	i
图目录 .....	ii
表目录 .....	iii
<b>1 前言 .....</b>	<b>1</b>
1.1 AE250 简介 .....	1
1.2 准备工作 .....	2
1.3 开发/调试步骤 .....	3
<b>2 Debug Cable 连接说明 .....</b>	<b>4</b>
<b>3 RDS 使用说明 .....</b>	<b>6</b>
3.1 RDS 安装 .....	6
3.2 新建工程 .....	6
3.3 导入/导出工程 .....	8
3.4 下载程序到 Flash .....	10
3.5 片上调试 .....	13
3.6 RDS 内置的串口终端使用方法 .....	15
<b>4 参考设计 .....</b>	<b>16</b>
4.1 工程代码 .....	16
4.2 参考设计 .....	17

# 图目录

图 1-1 AE250 结构框图 .....	1
图 1-2 开发/调试系统结构框图 .....	2
图 2-1 AICE-MINI+ Debug Cable 及其引脚示意图 .....	4
图 3-1 新建工程 .....	7
图 3-2 Import/Export a Project .....	8
图 3-3 导入工程界面 .....	9
图 3-4 导出工程界面 .....	9
图 3-5 设置 System Reset Vector Default .....	10
图 3-6 设置 SPI1 Configuration .....	11
图 3-7 设置 MSPI 接口为普通 IO .....	12
图 3-8 ae250.sag 中 bootloader 参数设置 .....	12
图 3-9 Debug 参数设置界面 .....	13
图 3-10 Debug 按钮介绍 .....	14
图 3-11 汇编指令代码窗口 .....	14
图 3-12 RDS 内置串口终端 .....	15

# 表目录

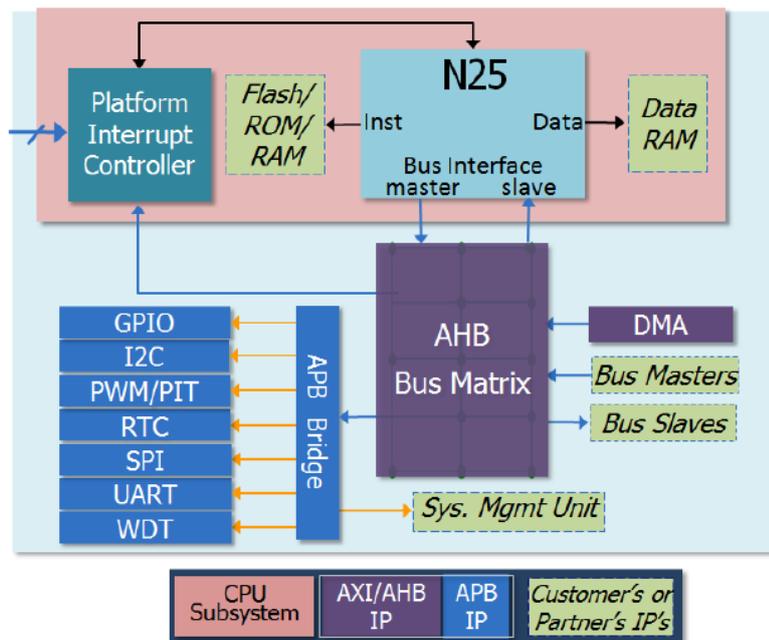
表 2-1 AICE-MINI+ Debug Cable 引脚定义 .....	4
表 3-1 SPI1 接口物理约束.....	11

# 1 前言

## 1.1 AE250 简介

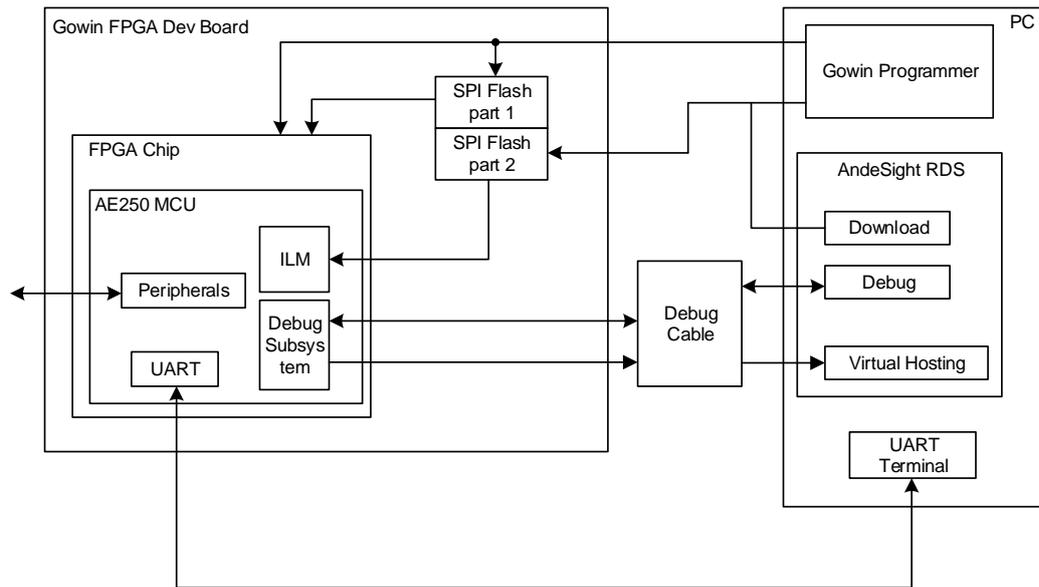
AE250 是一个 32 位 RISC-V MCU 系统，其主要结构如图 1-1 所示。

图 1-1 AE250 结构框图



基于 Gowin FPGA 开发板的 RISC-V AE250 MCU 开发/调试系统如图 1-2 所示。

图 1-2 开发/调试系统结构框图



通过 PC 端的 Gowin Programmer 将开发板上的 FPGA 芯片配置为 AE250 MCU, 连接好 Debug Cable, 即可通过 PC 端的 AndeSight RDS v311 软件进行嵌入式程序的开发和调试。

## 1.2 准备工作

在使用 Gowin FPGA+AE250 进行开发调试之前, 需要准备以下工具:

1. Gowin GW2A 系列 FPGA 开发板。
2. 高云半导体云源®软件安装包, 用于配置和下载 FPGA 芯片。
3. AndeSight RDS v311 安装包, 用于开发和调试嵌入式程序。
4. Debug Cable 用于调试嵌入式程序, 当前默认选用 [AICE-MINI+](#), 请用户自行联系购买。

注!

- 如果需要通过 UART 输出信息, 则需准备一条 UART 转 USB 线;
- 其他要用的外设所需设备。

## 1.3 开发/调试步骤

使用基于 GW2A-55C 开发板的 RISC-V AE250 MCU 进行开发/调试操作的基本步骤如下：

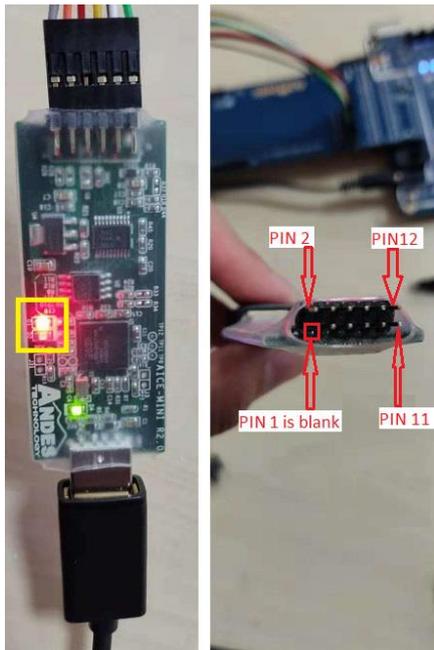
1. 安装软件。Gowin 云源软件用于配置和生成 AE250 的 RTL 设计并生成该设计的 Bitstream 文件；AndeSight RDS v311 软件用于用户进行嵌入式程序的开发和调试；以及其他调试所需的软件和驱动。
2. 配置开发板的电源和下载线。使用 Gowin Programmer 将 AE250 的 Bitstream 文件下载到开发板上的 FPGA 芯片中，此时 AE250 在开发板上运行。
3. 打开软件 RDS，新建嵌入式工程或者打开已有的工程，进行编码、编译等操作。连接 AE250 调试用的 Debug Cable，将工程的编译结果下载到 AE250 中的 ILM (指令存储器)中，开始片上调试。
4. 调试时，可以使用 UART 转 USB 的 Cable 将 AE250 的 UART 接口连接到 PC 端，使用 RDS 中内置的串口终端进行输入输出操作。可以使用 GPIO 连接到 LED 指示灯、按键或者外接插针上，以进行输入输出操作，另外还有 I2C、SPI 等外设可选择使用。
5. AE250 可以通过 SPI 外接一个 Flash，通过 Gowin Programmer 将嵌入式程序的编译结果下载到 Flash 中，在芯片上电后，AE250 自动从 SPI Flash 中读取嵌入式程序并启动。可以将保存 FPGA Bitstream 的 Flash 进行复用，一部分用来保存 FPGA Bitstream，另一部分保存嵌入式程序的编译结果。这是一种比较实用且经济的方法。

详细的基本步骤介绍请参见第二章 Debug Cable 连接说明，第三章 RDS 使用说明以及第四章 参考设计。

# 2 Debug Cable 连接说明

RDS + AE250 默认使用 AICE-MINI+ debug cable，其外观如图 2-1（左侧），图中右侧为引脚外观图，为 12 PIN 接口。注意图中空缺的 PIN 为 PIN 1。当连接线正确连接并打开 RDS 软件后，图中用黄色方框标注的红色 LED 灯会熄灭。

图 2-1 AICE-MINI+ Debug Cable 及其引脚示意图



AICE-MINI+ debug cable 的引脚定义如表 2-1，注意 PIN 1 的定义为 NC (No Connection)，对应空缺的 PIN 1。VREF 需连接一个 3.3V 电源引脚，3/5 脚的 GND 只需连接一个即可。

表 2-1 AICE-MINI+ Debug Cable 引脚定义

序号	AICE-MINI+ Debug Cable 引脚
1	NC
2	TSRST_N

序号	AICE-MINI+ Debug Cable 引脚
3	GND
4	TTMS
5	GND
6	TCK
7	VREF
8	NC
9	NC
10	TTRST_N
11	TTDO
12	TTDI

# 3 RDS 使用说明

## 3.1 RDS 安装

安装包解压后进入 Windows/Disk1，双击 `setup.exe` 进行安装。安装时不需要做特殊设置，在安装过程中有弹出对话框询问是否安装驱动，请选择安装。具体安装步骤请参照文档：

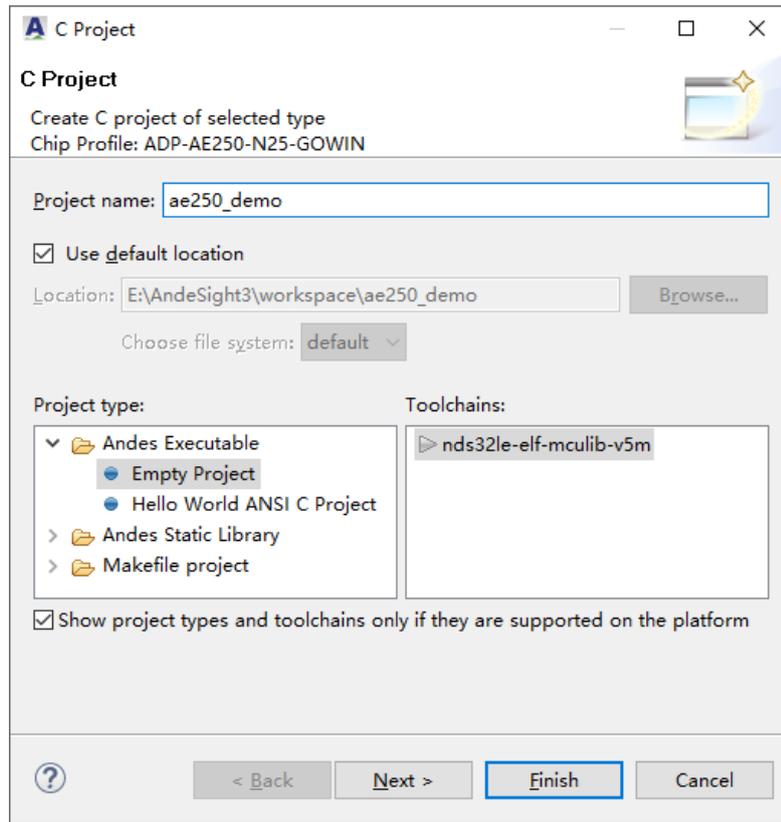
`AndeSight_RDS_v3.2_Installation_Guide_UM207_V1.0.pdf`，可在安装包中找到。

1. 设置安装路径和工作空间(`workspace`)路径时，请不要包含汉字或空格，否则运行时会出现错误。
2. 当前版本的 RDS 默认支持的 Debug Cable 为 AICE-MINI+ Cable。
3. 安装 RDS 之后，可能会出现 Gowin Programmer 无法连接到开发板的问题，此时重新安装 Gowin Programmer 驱动即可修复此问题。
4. RDS 的序列号和证书文件，请联系 Gowin Semiconductor Corp. 申请。

## 3.2 新建工程

在 RDS 界面上点击“File > New > Project > Andes C project > Next”，进入新建 C 工程的配置界面，如图 3-1 所示。

图 3-1 新建工程



新建 C 工程时，需要配置的参数如下：

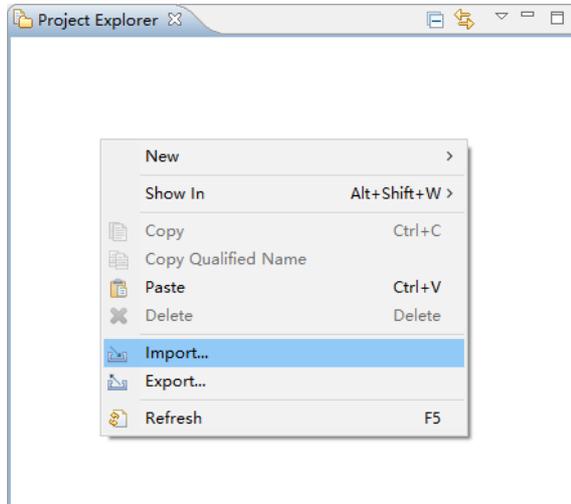
1. 工程名。
2. 保存位置，默认位置是当前的 workspace。
3. Connection Configuration 设置为 ICE，表示使用 ICE debug cable 连接开发板。如果使用仿真器进行作为测试平台，请选择 SID。
4. Chip Profile 选择 ADP-AE250-N25-GOWIN，这是根据 Gowin FPGA 进行优化配置了的 Chip Profile。
5. Project Type 包括空工程和 Hello World ANSI C Project 工程模板。
6. Toolchains，AE250 默认为 nds32le-elf-mculib-v5m。

新建工程后，在“Project Explorer”工程名上单击鼠标右键，在下拉菜单中选择“Build Project”或者点击工具栏上的“”对工程进行编译、链接等操作。在下拉菜单中选择“Clean Project”可进行 make clean 的操作。

## 3.3 导入/导出工程

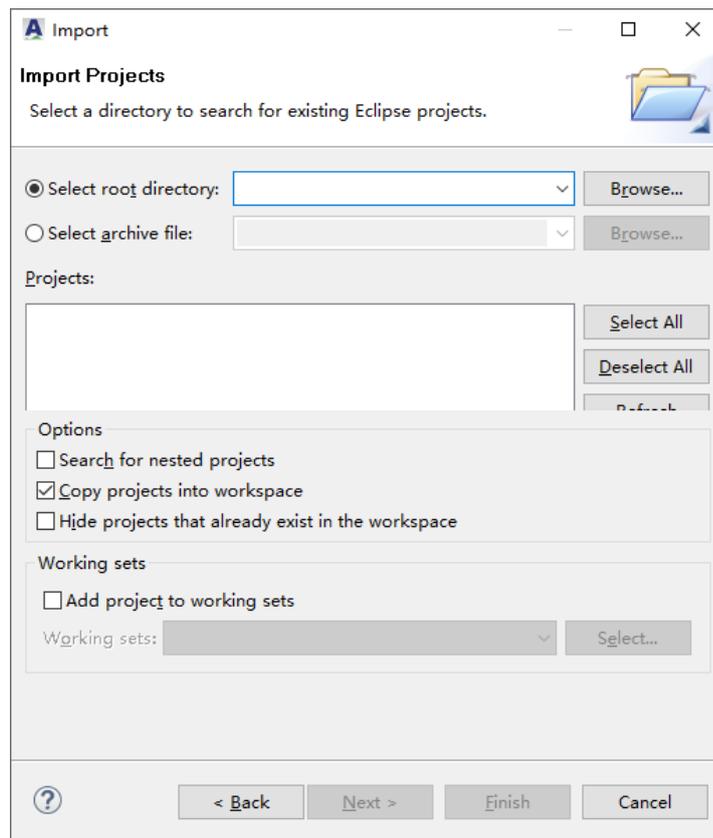
在 Project Explorer 的空白处单击鼠标右键，即可选择 **Import**（导入）或者 **Export**（导出），进行工程的导入/导出操作，如图 3-2 所示。

图 3-2 Import/Export a Project



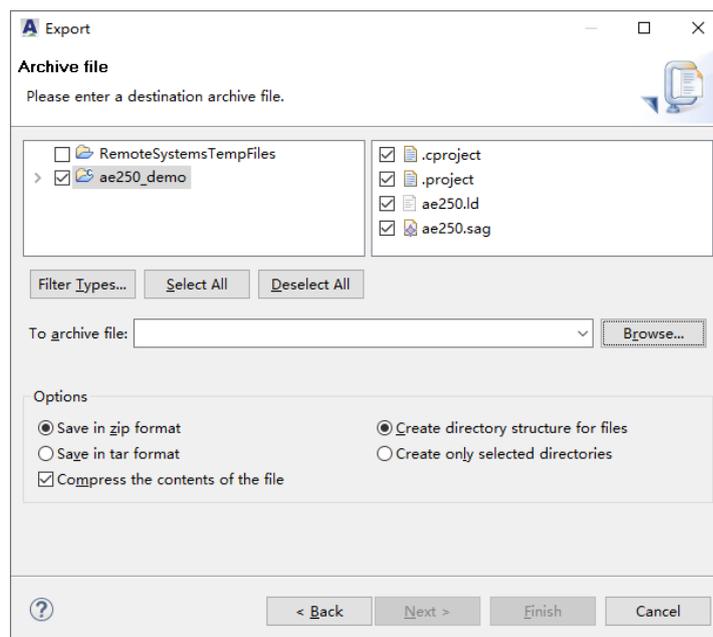
导入工程的界面如图 3-3 所示，点击 “**Import > General > Existing Project into workspace**”即可进入导入工程界面。选择“**Select root directory**”时，导入文件夹形式的工程；选择“**Select archive file**”时，导入压缩包格式的工程。

图 3-3 导入工程界面



导出工程的界面如图 3-4 所示，选择“Export... > Archive File”，即可打开导出工程界面，选择要导出的工程、压缩格式、保存路径等参数后，即可完成导出。

图 3-4 导出工程界面

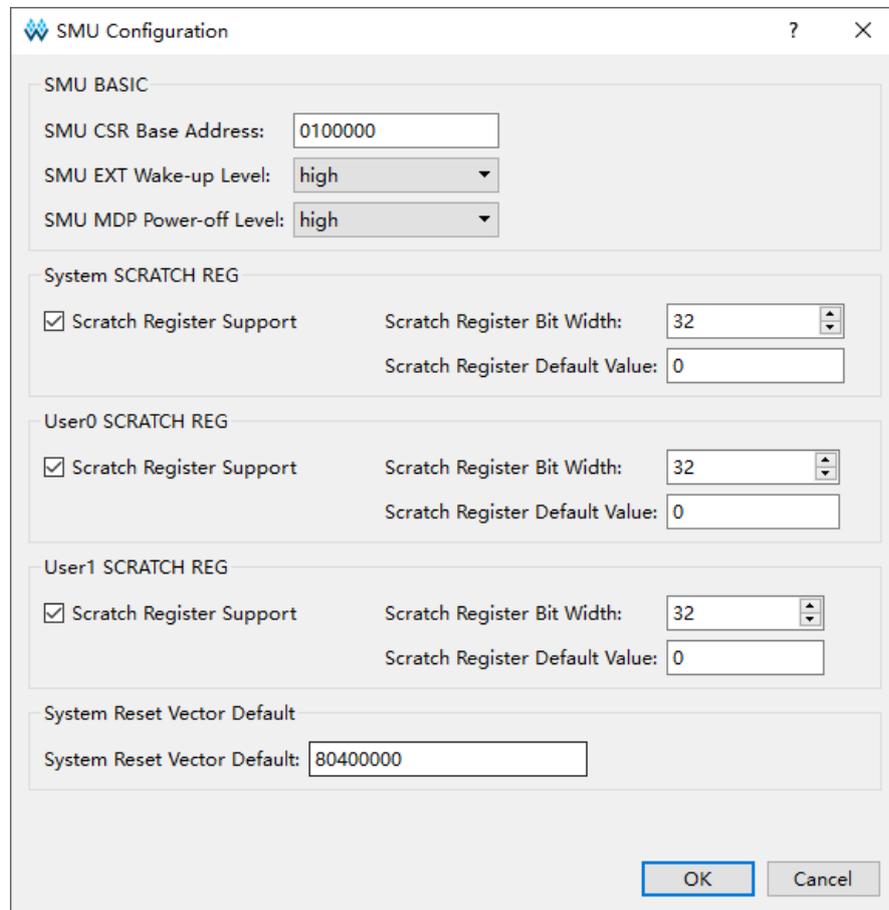


## 3.4 下载程序到 Flash

AE250 支持从 Flash 启动，启动时先通过 SPI 接口从 Flash 中读取嵌入式程序并存储到指令存储器 ILM 中，然后开始执行嵌入式程序。推荐的方法是复用保存 FPGA Bitstream 的 SPI Flash，使用 Flash 的前半部分保存 FPGA 的 Bitstream，后半部分保存嵌入式程序的二进制文件。

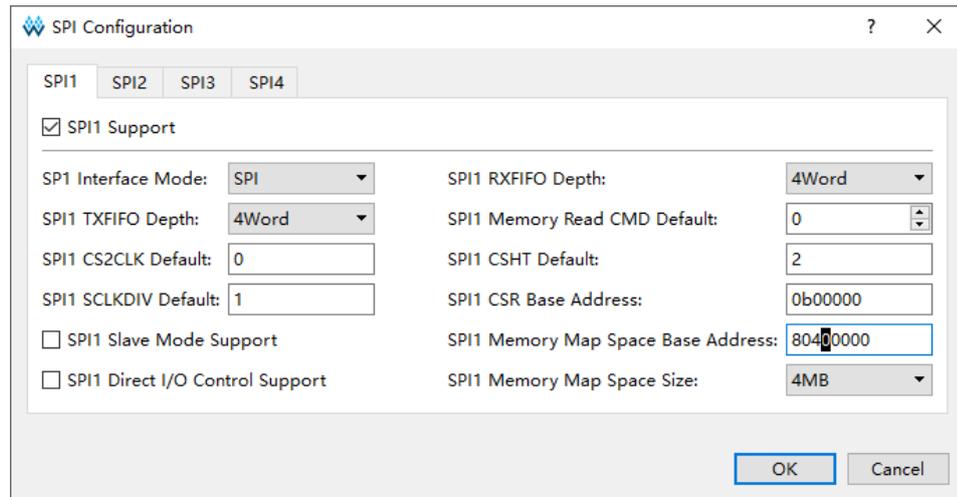
1. 打开 Gowin 云源软件 IP Core Generator，并调出当前 AE250 的 RTL 参数设置。双击 SMU，打开 SMU 的设置界面，将“System Reset Vector Default”设置为“0x80400000”，如图 3-5 所示。令 SPI Flash 的 0~0x400000 共 4M 字节的空间作为 Bitstream 的保存地址，从 0x400000 开始作为嵌入式程序的二进制文件保存地址。

图 3-5 设置 System Reset Vector Default



2. 双击 SPI1，打开 SPI1 的设置界面，勾选“SPI1 Support”，将“SPI1 Memory Map Space Base Address”设置为“0x80400000”，如图 3-6 所示。

图 3-6 设置 SPI1 Configuration



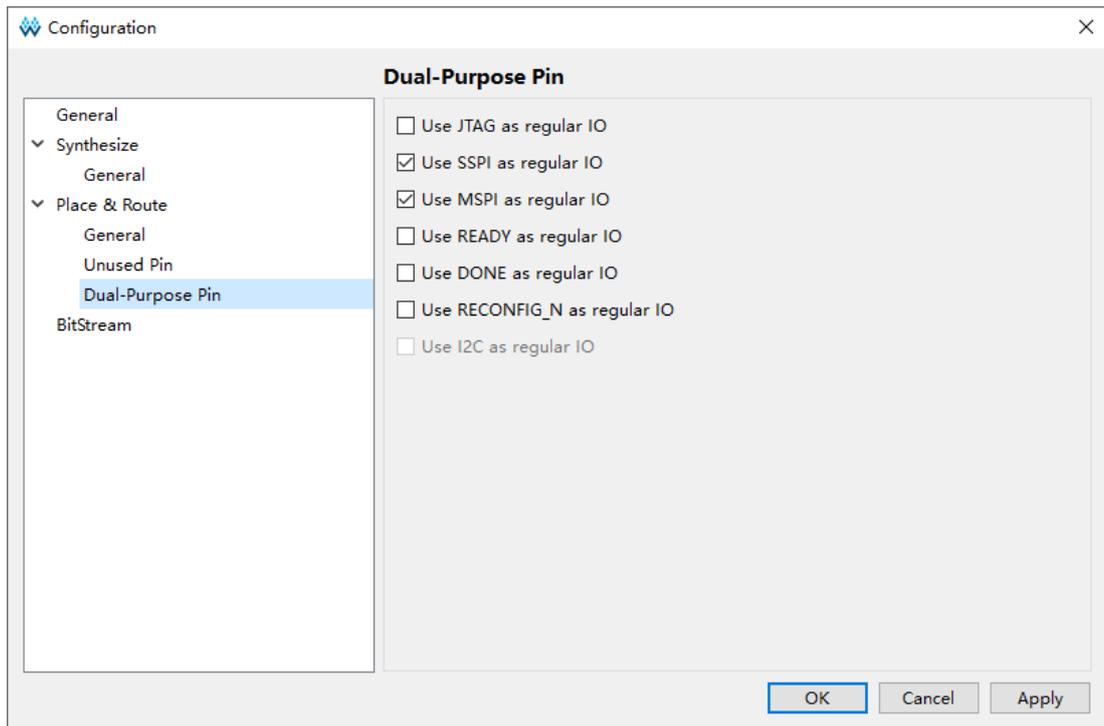
- 需要在 RTL 设计的物理约束中，将 SPI1 的接口接到 SPI Flash 上，将 SPI1 的接口按照下表进行物理约束，针对不同的 FPGA 芯片，MSPI 接口的位置也是不同的，需针对具体情况进行具体的约束。

表 3-1 SPI1 接口物理约束

AE250 SPI1 接口	FPGA MSPI 接口
CSN	MCSN
CLK	MCLK
MISO	MSO
MOSI	MSI

- 将 MSPI 接口复用为普通 IO。在云源软件的界面中点击“Process”窗口中，右键单击“Place & Route”，在弹出菜单中选择“Configuration”，选择“Dual-Purpose Pin”标签页，勾选“Use MSPI as regular IO”选项并点击确定即可进行正确的布局布线。

图 3-7 设置 MSPI 接口为普通 IO



5. 修改嵌入式程序的参数设置。首先修改链接器脚本中 `bootloader` 的参数，由于 AE250 的嵌入式程序中的链接器脚本是使用 SAG 文件自动生成的，因此在 SAG 文件中进行修改。打开 `ae250.sag`，找到 `BOOTLOADER`，将其修改为 RTL 设计中 `System Reset Vector Default` 的值，如图 3-8 所示。然后修改 `config.h`，打开 `src/bsp/config/config.h`，找到宏定义 `BUILD_MODE`，将其修改为 `BUILD_BURN`。

图 3-8 ae250.sag 中 bootloader 参数设置

```

1 USER_SECTIONS .bootloader
2 USER_SECTIONS .loader
3
4 HEAD 0x00000000
5 {
6     BOOTLOADER 0x80400000
7     {
8         ADDR __flash_start
9         * KEEP (.bootloader)
10        LOADADDR __bootloader_lmaend
11    }
12 }
13 }
14

```

**注!**

- 参数应当与 RTL 参数中 `System Reset Vector Default` 的值保持一致；
- 修改编译设置，右键单击嵌入式工程名，选择 `Build Settings`，选择 `Objcopy > General` 选项卡，将第一条 `Disable. (Do not auto-generate output file.)` 这一选项取消勾选。

重新编译嵌入式程序，生成嵌入式工程的二进制文件，使用 `Gowin Programmer external Flash C Bin` 模式下载到 SPI Flash `0x400000` 地址。

将修改后的 RTL 设计进行重新综合和布局布线，使用 Gowin Programmer external Flash 模式下载到 SPI Flash 0x000000 地址。

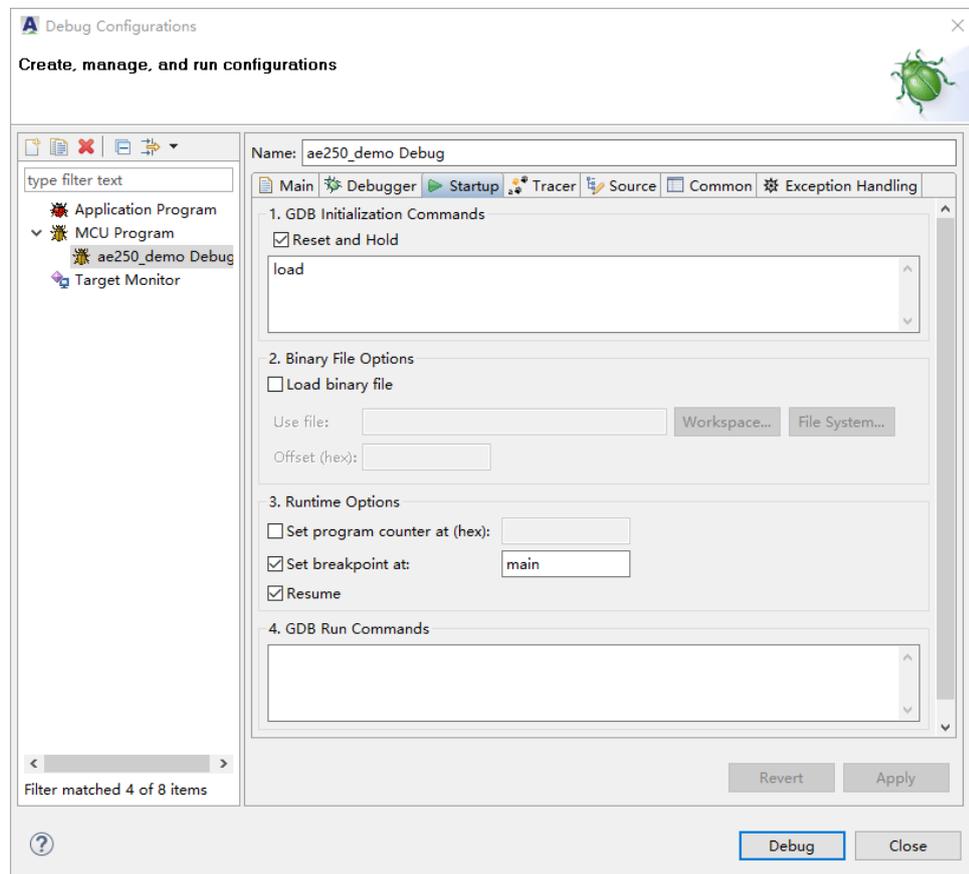
## 3.5 片上调试

编译成功后，可以将嵌入式工程的编译结果下载到开发板上进行片上调试。

修改 config.h, 打开 src/bsp/config/config.h, 找到宏定义 BUILD\_MODE, 将其修改为 BUILD\_LOAD, 重新编译嵌入式程序。

在 Project Explorer 工程名上单击鼠标右键，在下拉菜单中选择“Debug As > MCU Program”，第一次执行此操作时会弹出对话框，用于设置 Debug Configuration，如图 3-9 所示。

图 3-9 Debug 参数设置界面



在 Startup 标签页中，勾选“Reset and Hold”选项可以让程序在开始运行时在执行第一条指令之前停止。在该选项下方的参数框中输入 load，表示在进入片上调试之前先将嵌入式工程的编译结果下载到指令存储器 ILM 中。

在“Runtime Options”中勾选“Set breakpoint at”，在输入框中输入一个标签（比如 main），可以在 main 函数开始的地方设置一个断点，勾选

“Resume” 则会在进入片上调试后直接开始连续运行。

进入片上调试功能后，会自动进入到 debug 视图模式，可以看到如图 3-10 的一个区域。该区域为片上调试的操作区，红框中为一些调试用到的快捷按钮，从左到右分别是：重新开始 DEBUG、继续运行、暂停、结束、断开链接、链接到一个进程（图中为灰色，表示暂不可用）、进入子函数（step into）、执行下一条指令（step over）、跳出子函数（step return，此时为灰色，表示暂不可用）、单条指令运行模式（instruction stepping mode，在此模式下，每次运行一条 risc-v 汇编指令，否则每次运行一条 C 语言语句）。

在代码文本左侧行号上双击鼠标左键可快速设置断点或取消断点，在代码文本中右键单击鼠标可以在弹出菜单中选择运行到该行（run to line）。

图 3-10 Debug 按钮介绍

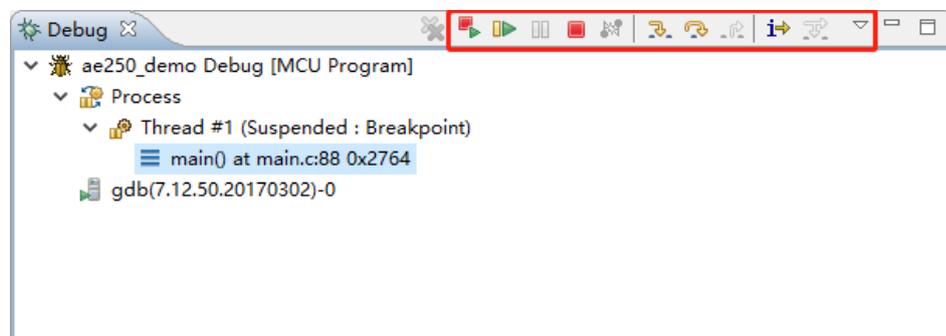
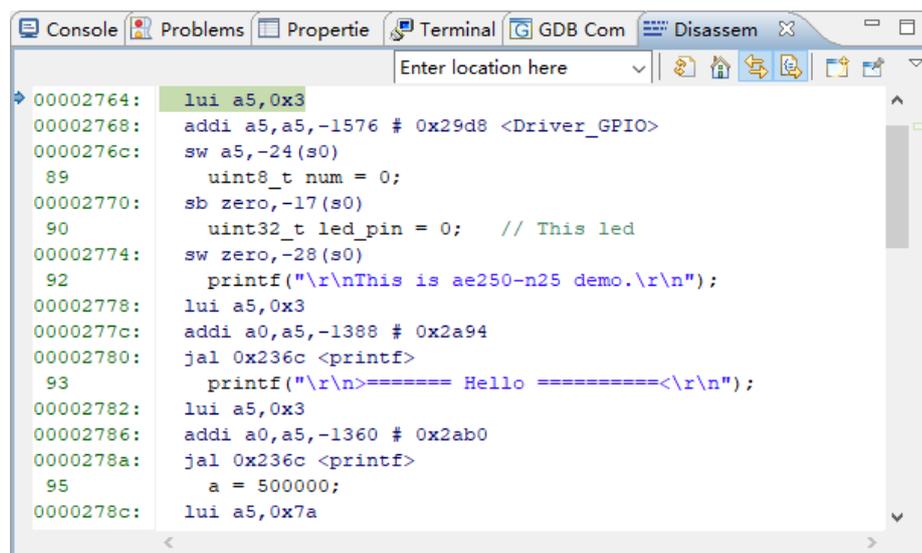


图 3-11 为汇编语句窗口，所显示的内容为 ILM 中实时运行的汇编指令的内容。

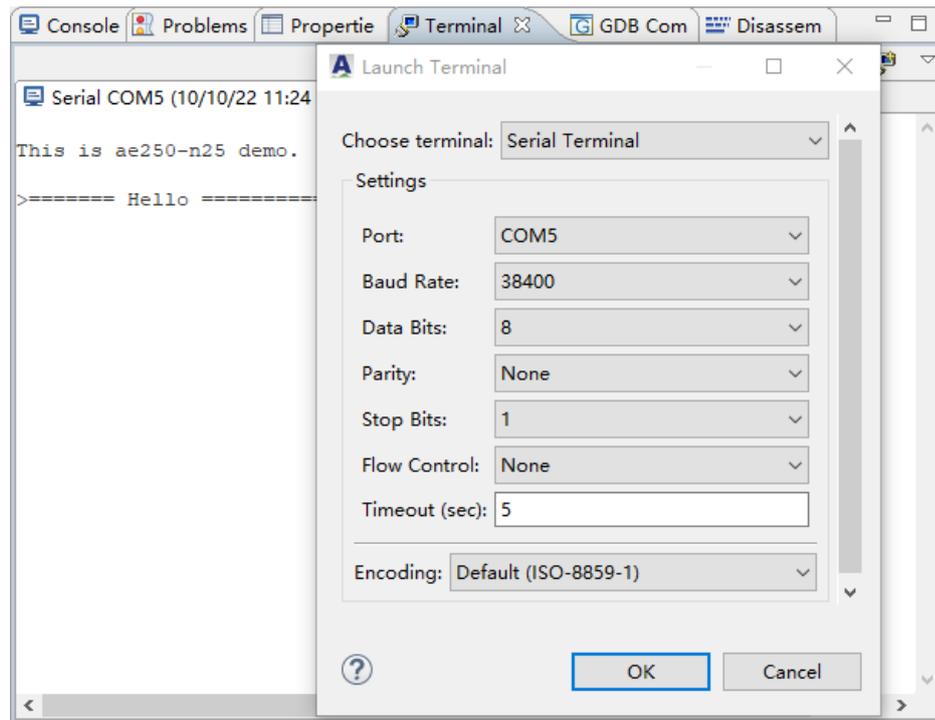
图 3-11 汇编指令代码窗口



## 3.6 RDS 内置的串口终端使用方法

图 3-12 为 RDS 内置的 UART Terminal 及设置界面，如果需要使用，请在顶部菜单中点击“Window > Show View > Terminal”，打开“Terminal”窗口，然后点击“open a terminal”按钮，新建一个串口终端。设置好端口号（可在硬件管理器中查看）、波特率等参数后即可点击“OK”开始使用。

图 3-12 RDS 内置串口终端



具体的使用方法，请参照文档：

AndeSight\_RDS\_v3.1\_User\_Manual\_UM170\_V1.0.pdf，可在安装目录中的 doc 路径下找到。

# 4 参考设计

## 4.1 工程代码

AE250 嵌入式工程模板中的关键文件如下：

1. `src/bsp/ae250/ae250.h`: 该文件中是系统时钟定义、外设寄存器定义、外设寄存器地址映射定义和中断源编号的定义。其中时钟定义须与 AE250 的参数配置保持一致。
2. `src/bsp/ae250/ae250.c`: `reset_handler` 函数是启动嵌入式程序的入口，入口中在执行 `main` 函数之前执行了 UART 初始化的操作，根据 AE250 的参数配置选择所需的 UART 端口并配置所需要的波特率。
3. `src/bsp/ae250/interrupt.c`: 该文件是 AE250 的中断处理函数定义。
4. `src/bsp/config/config.h`: 该文件中包含了控制编译方式的宏定义；其中 `#define BUILD_MODE` 可定义为 `BUILD_LOAD` 或 `BUILD_BURN`；`BUILD_LOAD` 表示程序直接加载到指令存储器 ILM 中，一般用于调试时；`BUILD_BURN` 表示程序下载到 SPI Flash 中，上电后先从 SPI Flash 中将程序读取到 ILM 中后运行，适用于发布版程序。
5. `src/bsp/start.S`: 汇编语言写成的起始文件。
6. `src/bsp/loader.c`: `bootloader` 文件，用于从 SPI Flash 启动。
7. `ae250.sag`: `sag` 是 `scattering-and-Gathering` 格式的脚本，用于生成链接器脚本。需要注意的是，`ae250.sag` 中的 `memory map` 参数需要与 AE250 的参数保持一致。
8. `src/bsp/driver`: 该目录下包含两个文件夹，`ae250` 为 AE250 的驱动代码，`include` 中为驱动函数的调用接口。
9. `src/bsp/lib`: 包含两个文件，`printf.c` 中通过重定义 C 标准库中子函数的形式，实现通过 UART 输出 `printf` 的信息。`read.c` 中为一个简单的通过 UART 读取输入信息的函数。

## 4.2 参考设计

安装完成后, 在安装目录的 **demo** 文件夹中或网站发布的参考设计包里, 可以找到几个基本的参考设计, 可以通过导入工程的方式加载到 **RDS** 中进行试用、调试以及二次开发。参考设计如下:

1. **ae250\_demo**: 演示的是 **AE250** 的 **UART** 输入、输出和 **GPIO** 输出功能。
2. **ae250\_plic**: 演示的是中断控制器对中断的响应, 并提供了 **machine timer** 和 **pit timer** 的演示。
3. **ae250\_freertos**: 演示的是 **AE250** 移植嵌入式实时操作系统 **FreeRTOS** 多线程运行程序。
4. **ae250\_ucosiii**: 演示的是 **AE250** 移植嵌入式实时操作系统 **uC/OS-III** 多线程运行程序。

