




Gowin 云源软件 用户指南

SUG100-2.7, 2022-05-20

版权所有 © 2022 广东高云半导体科技股份有限公司

GOWIN高云、、Gowin、GOWIN、gowin、GowinSynthesis、高云、云源、小蜜蜂以及晨熙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，其所有权利属其所有者所有。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 |
|------------|-------|---|
| 2019/11/08 | 2.0 | <ul style="list-style-type: none">● 添加Options窗口的Synplify Pro配置选项说明；● 更新Synplify Pro属性及取值说明。 |
| 2019/11/28 | 2.1 | <ul style="list-style-type: none">● Place & Route新增General配置项：Promote Physical Constraint Warning to Error、Report Auto-Placed IO Information、Place Option、Route Option；● Hierarchy层级显示窗口增加资源信息显示；● Process窗口的Netlist文件支持查找所有匹配项；● IDE自带编辑器支持添加/删除注释功能。 |
| 2020/03/09 | 2.2 | <ul style="list-style-type: none">● Hierarchy窗口支持解析VHDL文件；● GowinSynthesis®支持VHDL及Verilog与VHDL混合语言的综合；● 新增工程文件未保存时执行综合或PnR流程弹出提示框的功能；● 添加信息输出窗口中使用Tcl命令的介绍；● 添加Synthesize和PnR配置选项的解释说明；● 添加User Flash初始化文件编辑器的使用介绍。 |
| 2020/05/13 | 2.3 | <ul style="list-style-type: none">● 新增Unused Pin配置功能；● Hierarchy窗口新增文件加密功能。 |
| 2020/09/01 | 2.4 | <ul style="list-style-type: none">● 新增Schematic Viewer功能；● 更新Hierarchy窗口资源信息显示功能；● Place & Route新增BitStream配置项：Enable Daisy Chain Bypass。 |
| 2020/10/21 | 2.4.1 | Synplify Pro相关描述更新。 |
| 2021/06/17 | 2.5 | <ul style="list-style-type: none">● 更新界面截图及部分描述；● 移除Synplify Pro相关的内容。 |
| 2021/11/02 | 2.6 | <ul style="list-style-type: none">● 更新SSPI和MSPI的复用管脚信息；● 更新-ireg_in_iob/-oreg_in_iob/-ioreg_in_iob的描述；● 移除MODE复用管脚的配置；● Place & Route新增BitStream配置项：Power On Reset；● 增加仿真文件介绍。 |
| 2022/05/20 | 2.7 | <ul style="list-style-type: none">● 更新Loading Rate的取值；● 新增8.2.9 source章节。 |

目录

| | |
|-------------------------------|-----------|
| 目录 | i |
| 图目录 | iv |
| 表目录 | vi |
| 1 关于本手册 | 1 |
| 1.1 手册内容 | 1 |
| 1.2 相关文档 | 1 |
| 1.3 术语、缩略语 | 2 |
| 1.4 技术支持与反馈 | 2 |
| 2 云源概述 | 3 |
| 2.1 简介 | 3 |
| 2.2 芯片支持 | 4 |
| 2.3 云源的安装与启动 | 4 |
| 3 云源用户界面 | 5 |
| 3.1 标题栏 | 7 |
| 3.2 菜单栏 | 7 |
| 3.2.1 File 菜单栏 | 7 |
| 3.2.2 Edit 菜单栏 | 7 |
| 3.2.3 Project 菜单栏 | 8 |
| 3.2.4 Tools 菜单栏 | 8 |
| 3.2.5 Window 菜单栏 | 9 |
| 3.2.6 Help 菜单栏 | 9 |
| 3.3 工具栏 | 9 |
| 3.4 工程管理区 (Design) | 10 |
| 3.5 过程管理区 (Process) | 10 |
| 3.6 设计层级显示区 (Hierarchy) | 10 |
| 3.6.1 右键菜单 | 10 |
| 3.6.2 资源显示 | 11 |
| 3.6.3 文件加密 | 12 |

| | |
|-------------------------------|-----------|
| 3.7 源文件编辑区 | 14 |
| 3.8 信息输出区 | 15 |
| 4 云源使用 | 18 |
| 4.1 新建工程 | 18 |
| 4.2 打开工程 | 20 |
| 4.3 编辑工程 | 21 |
| 4.3.1 编辑工程器件 | 22 |
| 4.3.2 编辑工程文件 | 23 |
| 4.3.3 编辑工程配置 | 29 |
| 4.4 管理工程过程 | 44 |
| 4.4.1 Design Summary | 45 |
| 4.4.2 User Constraints | 46 |
| 4.4.3 Synthesize | 46 |
| 4.4.4 Place & Route | 47 |
| 4.4.5 Program Device | 48 |
| 4.5 工程存档及恢复 | 49 |
| 4.5.1 工程存档 | 49 |
| 4.5.2 恢复存档工程 | 50 |
| 4.6 增量编译 | 50 |
| 4.7 退出软件 | 52 |
| 5 云源集成工具 | 53 |
| 5.1 物理约束编辑器 | 53 |
| 5.2 时序约束编辑器 | 55 |
| 5.3 IP Core 产生器 | 55 |
| 5.4 在线逻辑分析仪 | 57 |
| 5.5 功耗分析工具 | 58 |
| 5.6 块存储器初始化文件编辑器 | 59 |
| 5.7 User Flash 初始化文件编辑器 | 61 |
| 5.7.1 二进制格式 (Bin File) | 62 |
| 5.7.2 十六进制格式 (Hex File) | 62 |
| 5.8 原理图查看器 | 64 |
| 6 云源输出文件 | 66 |
| 6.1 综合报告 | 66 |
| 6.2 布局布线报告 | 67 |
| 6.3 端口属性报告 | 68 |
| 6.4 时序报告 | 69 |

| | |
|-----------------------------|-----------|
| 6.5 功耗分析报告 | 69 |
| 7 仿真文件 | 71 |
| 7.1 功能仿真文件 | 71 |
| 7.2 时序仿真文件 | 71 |
| 8 Tcl 命令说明 | 73 |
| 8.1 启动命令行模式 | 73 |
| 8.1.1 gw_sh.exe | 73 |
| 8.2 命令介绍 | 73 |
| 8.2.1 add_file | 73 |
| 8.2.2 rm_file | 74 |
| 8.2.3 set_device | 74 |
| 8.2.4 set_file_prop | 75 |
| 8.2.5 run | 75 |
| 8.2.6 set_file_enable | 75 |
| 8.2.7 saveto | 76 |
| 8.2.8 set_option | 76 |
| 8.2.9 source | 83 |

图目录

| | |
|--|----|
| 图 3-1 软件用户界面 | 6 |
| 图 3-2 Hierarchy 窗口右键菜单显示 | 11 |
| 图 3-3 Hierarchy 窗口资源信息显示 | 12 |
| 图 3-4 Pack User Design 对话框 | 12 |
| 图 3-5 Pack User Design 对话框打印信息 | 13 |
| 图 3-6 Pack User Design 对话框提取错误信息 | 14 |
| 图 3-7 Find & Replace 对话框 | 15 |
| 图 3-8 Search Result 窗口 | 15 |
| 图 3-9 信息输出窗口 | 16 |
| 图 3-10 Tcl 编辑窗口 | 17 |
| 图 4-1 新建工程 | 18 |
| 图 4-2 新建工程向导 | 19 |
| 图 4-3 设置 FPGA 器件信息 | 20 |
| 图 4-4 工程信息 | 20 |
| 图 4-5 打开文件 | 21 |
| 图 4-6 工程文件窗口 | 22 |
| 图 4-7 工程配置芯片型号 | 23 |
| 图 4-8 新建文件对话框 | 23 |
| 图 4-9 新建 Verilog File 对话框 | 24 |
| 图 4-10 新建配置文件对话框 | 24 |
| 图 4-11 GPA 配置文件窗口 | 25 |
| 图 4-12 Design 窗口右键菜单 | 25 |
| 图 4-13 工程文件编辑菜单 | 26 |
| 图 4-14 外部编辑器 | 27 |
| 图 4-15 Save Modified Files 对话框 | 27 |
| 图 4-16 工程文件属性对话框 | 28 |
| 图 4-17 选中多个同类型文件右键菜单列表 | 29 |
| 图 4-18 选中多个不同类型文件右键菜单列表 | 29 |
| 图 4-19 工程配置选项对话框 | 30 |

| | |
|--|----|
| 图 4-20 GowinSynthesis 参数配置项..... | 30 |
| 图 4-21 配置布局布线选项..... | 32 |
| 图 4-22 Unused Pin 选项..... | 33 |
| 图 4-23 配置复用管脚选项..... | 34 |
| 图 4-24 配置 Bitstream 文件选项..... | 35 |
| 图 4-25 Background Programming 设为 I2C..... | 35 |
| 图 4-26 Background Programming 设为 I2C/JTAG/SSPI/QSSPI | 35 |
| 图 4-27 工程 Process 窗口 | 45 |
| 图 4-28 工程信息显示 | 45 |
| 图 4-29 右击 Synthesize..... | 47 |
| 图 4-30 Programmer 界面 | 48 |
| 图 4-31 工程存档对话框 | 50 |
| 图 4-32 恢复存档工程对话框 | 50 |
| 图 4-33 增量编译配置窗口..... | 51 |
| 图 5-1 Chip Array 窗口 | 54 |
| 图 5-2 Package View 窗口 | 54 |
| 图 5-3 创建时序约束界面..... | 55 |
| 图 5-4 IP Core Generator 窗口 | 56 |
| 图 5-5 GAO 配置文件窗口 | 57 |
| 图 5-6 GAO 界面 | 58 |
| 图 5-7 GPA 配置文件窗口..... | 59 |
| 图 5-8 初始化文件新建 New 对话框 | 60 |
| 图 5-9 初始化文件新建 New File 对话框..... | 60 |
| 图 5-10 初始化文件配置窗口 | 61 |
| 图 5-11 列数配置 | 61 |
| 图 5-12 批量设置 | 61 |
| 图 5-13 初始化文件新建 New 对话框 | 63 |
| 图 5-14 初始化文件新建 New File 对话框 | 63 |
| 图 5-15 初始化文件配置窗口 | 64 |
| 图 5-16 批量设置 | 64 |
| 图 5-17 Schematic Viewer 窗口 | 65 |
| 图 6-1 GowinSynthesis 综合报告 | 66 |
| 图 6-2 Place & Route Report | 67 |
| 图 6-3 Ports & Pins Report..... | 68 |
| 图 6-4 时序报告内容 | 69 |
| 图 6-5 功耗分析报告 | 70 |
| 图 7-1 选项配置 | 72 |

表目录

| | |
|---------------------------------------|----|
| 表 1-1 术语、缩略语 | 2 |
| 表 3-1 常见 Warning 和 Error 信息 | 16 |
| 表 4-1 PnR 配置选项含义描述 | 31 |
| 表 4-2 Background Programming 取值 | 36 |
| 表 4-3 Loading Rate 值及计算方式（一） | 38 |
| 表 4-4 Loading Rate 值及计算方式（二） | 39 |
| 表 4-5 Loading Rate 值及计算方式（三） | 41 |
| 表 4-6 Loading Rate 值及计算方式（四） | 42 |
| 表 4-7 Loading Rate 值及计算方式（五） | 43 |

1 关于本手册

1.1 手册内容

本手册主要描述高云半导体云源[®]软件（以下简称云源）的安装及操作，旨在帮助用户快速熟悉云源的使用流程，提高设计效率。本手册中的软件界面截图参考的是 1.9.8.06 版本，因软件版本升级，部分信息可能会略有差异，具体以用户软件版本的信息为准。

1.2 相关文档

通过登录高云半导体网站 www.gowinsemi.com.cn 可下载、查看以下相关文档：

1. [SUG940, Gowin 设计时序约束指南](#)
2. [SUG935, Gowin 设计物理约束指南](#)
3. [SUG114, Gowin 在线逻辑分析仪用户指南](#)
4. [SUG282, Gowin 功耗分析工具用户指南](#)
5. [SUG502, Gowin Programmer 用户指南](#)
6. [UG285, Gowin 存储器\(BSRAM & SSRAM\)用户指南](#)
7. [SUG283, Gowin 原语用户指南](#)
8. [UG286, Gowin 时钟资源\(Clock\)用户指南](#)
9. [UG287, Gowin 数字信号处理器\(DSP\)用户指南](#)
10. [UG289, Gowin 可编程通用管脚\(GPIO\)用户指南](#)
11. [UG295, Gowin 闪存资源\(User Flash\)用户指南](#)

1.3 术语、缩略语

本手册中的相关术语、缩略语及相关释义如表 1-1 所示。

表 1-1 术语、缩略语

| 术语、缩略语 | 全称 | 含义 |
|---------------------------|-------------------------------|-----------|
| CRC | Cyclic Redundancy Check | 循环冗余校验 |
| FPGA | Field Programmable Gate Array | 现场可编程门阵列 |
| GowinSynthesis | GowinSynthesis | 高云半导体综合工具 |
| IP Core | Intellectual Property Core | 知识产权核 |
| PnR | Place & Route | 布局布线 |
| GAO | Gowin Analyzer Oscilloscope | 在线逻辑分析仪 |
| GPA | Gowin Power Analyzer | 功耗分析工具 |
| FloorPlanner | FloorPlanner | 物理约束编辑器 |
| Timing Constraints Editor | Timing Constraints Editor | 时序约束编辑器 |
| Schematic Viewer | Schematic Viewer | HDL原理图查看器 |

1.4 技术支持与反馈

高云半导体提供全方位技术支持，在使用过程中如有任何疑问或建议，可直接与公司联系：

网址：www.gowinsemi.com.cn

E-mail：support@gowinsemi.com

Tel: +86 755 8262 0391

2 云源概述

2.1 简介

云源设计系统是专为高云半导体 FPGA 芯片而配套的集成电路设计与实现工具。云源系统针对高云 FPGA 芯片构架的低功耗、低成本特点进行了全面的优化设计，覆盖了从 RTL 电路功能描述到生成 FPGA 码流文件的完整流程，包括了优化设计、自动设计、图形交互设计等功能，具有性能优越、容易使用等特点。

云源设计系统主要功能：

- 软件系统支持高云 FPGA 芯片所有功能，覆盖从 RTL 电路功能描述到生成 FPGA 码流文件的完整设计流程；
- 综合优化工具 GowinSynthesis®支持高性能的逻辑设计和综合；
- 自动设计和交互式图形设计并用；
- 支持 Centos6.8/7.0/7.5(64bits)、Ubuntu18.04/20.04LTS、Win7/8/10(32bits/64bits)、Win XP (32bits)系统；
- 亿万门级软件；
- 支持 VHDL、Verilog HDL 和 System Verilog 语言；
- 支持高云产品优化的芯片架构；
- 具有独创的快速、高性能算法的布局和布线系统；
- 精确的时序分析和时序报告；
- 时钟分析和控制保证了更好的时序性能；
- 支持各种时序约束和物理约束；
- 支持实时监测硬件电路信号并加以存储，同时以时序波形图直观显示；
- 资源共享技术可提高芯片利用率、降低成本。

云源设计系统主要特点：

- 一体化设计
 - 设计可分阶段完成，也可一揽子自动完成；

- 可选择命令行模式或图形交互模式完成设计；
- 利用脚本设计,可灵活地设计任何单一模块而不影响一体化设计全程。
- 优化设计
 - 网表优化设计；
 - 快速时序优化分析和设计；
 - 资源分析和优化。
- 分层设计和分析
 - 支持分层网表结构输入和输出；
 - 同时支持扁平化网表结构输入和输出；
 - 分层图形显示、追踪、分析网表。
- 方便灵活的交互图形设计
 - 用户界面简单清晰；
 - 包含项目、设计模块、工具和输出部分；
 - 设计约束输入、选择和更新；
 - 快速时序分析和报告；
 - Push button 设计技术。

2.2 芯片支持

云源目前支持小蜜蜂[®]家族芯片和晨熙[®]家族芯片，具体的芯片类型、芯片主要资源及封装方式等详见高云官方网站介绍。

- 小蜜蜂[®]家族芯片：www.gowinsemi.com.cn/prod_view.aspx
- 晨熙[®]家族芯片：www.gowinsemi.com.cn/prod_view.aspx

注！

软件版本不同，支持的芯片可能会略有差异，具体信息请参考所使用的软件版本。

2.3 云源的安装与启动

云源在 Windows 系统下的安装方式一致，双击云源安装包，根据提示进行安装，安装结束后会默认在 PC 桌面创建快捷方式。云源在 Linux 系统下解压安装包文件夹即可完成软件安装。

软件安装完成后在第一次启动时需要先对 License 进行配置，软件 License 是一种格式合同，由高云半导体与用户签订，用以规定和限制软件用户使用软件的权利，以及高云半导体应尽的义务。

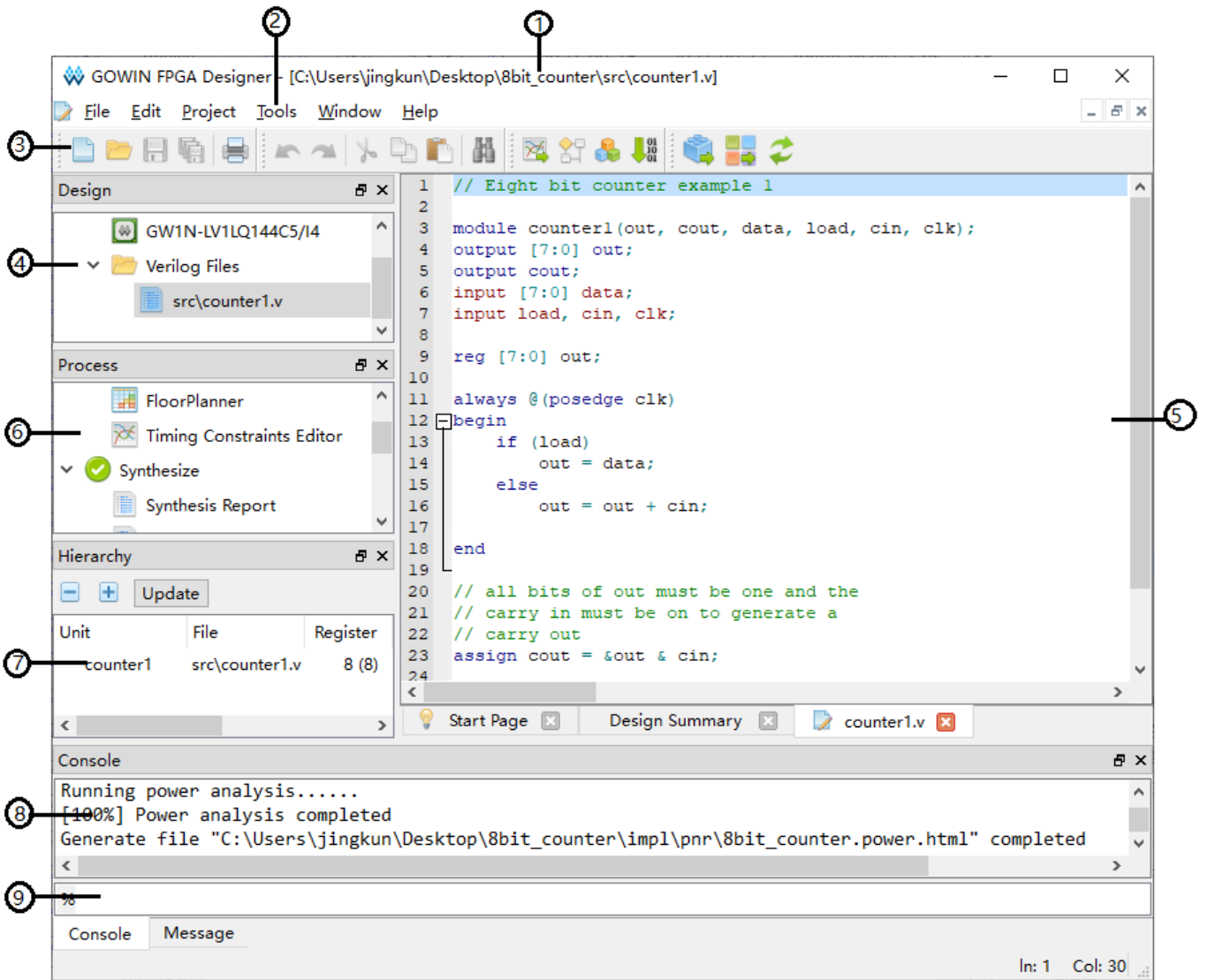
注！

云源详细的安装与启动指南请参考 [SUG501, Gowin 云源软件快速安装启动指南](#)。

3 云源用户界面

云源用户界面如图 3-1 所示，主要分为标题栏、菜单栏、工具栏、工程管理区 (**Design**)、过程管理区 (**Process**)、源文件编辑区、设计层级显示区 (**Hierarchy**)、信息输出区和 **Tcl** 命令编辑区。

图 3-1 软件用户界面



- ① 标题栏
- ② 菜单栏
- ③ 工具栏
- ④ 工程管理区
- ⑤ 源文件编辑区
- ⑥ 过程管理区
- ⑦ 设计层级显示区
- ⑧ 信息输出区
- ⑨ Tcl 命令编辑区

3.1 标题栏

主要显示当前工程的路径、名称及当前打开的文件名称。

3.2 菜单栏

主要提供一些常用菜单以及工程所需的启动工具, 包括 **File**、**Edit**、**Project**、**Tools**、**Window** 和 **Help** 选项, 详情如下:

3.2.1 File 菜单栏

- **New (Ctrl+N)**: 新建
- **Open (Ctrl+O)**: 打开
- **Save (Ctrl+S)**: 保存
- **Save As...**: 另存为
- **Save All (Ctrl+Shift+S)**: 保存所有文档
- **Close**: 关闭
- **Close All**: 关闭所有打开的文档
- **Close Project**: 关闭当前工程
- **Print Preview...**: 打印预览
- **Print... (Ctrl+P)**: 打印
- **Recent Files**: 最近打开的文件, 可以重新选择打开
- **Recent Projects**: 最近打开过的工程, 可以重新选择打开
- **Exit**: 软件退出

3.2.2 Edit 菜单栏

- **Undo (Ctrl+Z)**: 撤销
- **Redo (Ctrl+Y)**: 重做
- **Cut (Ctrl+X)**: 剪切
- **Copy (Ctrl+C)**: 复制
- **Paste (Ctrl+V)**: 粘贴
- **Select All (Ctrl+A)**: 选择全部
- **Find & Replace (Ctrl+F)**: 查找替换关键词
- **Toggle Comment Selection (Ctrl+ \backslash)**: 对所选内容添加注释
- **Macros**: 宏, 该选项下包含三个子选项:
 - **Start Record**, 单击 **Start Record** 后, IDE 中对可编辑文件进行的编辑操作会被录制下来;

- Stop Recording, 停止录制;
- Play Macro (Alt+R), 单击 Play Macro 会对可编辑文件执行所录制的操作。

3.2.3 Project 菜单栏

- Archive Project: 工程存档
- Restore Archived Project: 恢复存档工程
- Set Incremental: 设置增量编译
- Set Device: 设置当前工程的器件信息
- Configuration: 打开配置窗口
- Design Summary: 显示当前工程的具体信息

3.2.4 Tools 菜单栏

- Start Page: 开始页, 包含四部分内容, 即 Recent Projects、Quick Start、Tools 和 User Manuals。
 - Recent Projects: 最近打开过的工程列表, 最多会保留 10 个工程;
 - Quick Start 包含三项: New Project (新建工程)、Open Project (打开工程)、Open Example Project (打开示例工程);
 - Tools 包含三项: Floorplanner (启动物理约束编辑器)、Timing Constraints Editor (启动时序约束编辑器)、Programmer (启动编程器);
 - User Manuals 包含两项: Manual for LittleBee (小蜜蜂家族用户手册)、Manual for Arora (晨熙家族用户手册)。
- Gowin Analyzer Oscilloscope: 高云在线逻辑分析仪
- Schematic Viewer: HDL 设计原理图查看器
- IP Core Generator: IP Core 产生器
- Programmer: 编程器
- FloorPlanner: 物理约束编辑器
- Timing Constraints Editor: 时序约束编辑器
- Options: 包含三个选项: Environment、Text Editor、External Editor
 - Environment: 设置 IDE 参数, 包括语言 (Language)、工具栏图标大小 (Toolbar Icon Size) 以及默认新建工程的路径。设置语言后需要重新启动 IDE 后设置才会生效;
 - Text Editor: 设置文本编辑器属性, 包括字体、字号、配色方案、是否显示行号、是否可视化空白符、是否高亮显示当前行、是否高亮显示匹配的括号;

- **External Editor:** 设置第三方文本编辑器，可以选择是否始终使用第三方编辑器打开设计文件。

3.2.5 Window 菜单栏








- **Full Screen (F11):** 对 IDE 界面进行全屏显示
- **Tile:** 平铺显示
- **Cascade:** 覆盖显示
- **Reset Layout:** 恢复初始设置
- **Panels:** 对界面中各区域显示模块的选择，共有五个选项：
Design (工程管理区)、**Hierarchy** (设计层级显示区)、**Process** (过程管理区)、**Message** (输出信息分类汇总区)、**Console** (全部输出信息显示区)
- **Start Page:** 在源文件编辑区显示开始页
- **Design Summary:** 在源文件编辑区显示 **Design Page**, 页面包含 **General** 和 **Target Device** 两项内容，详细介绍如下：
 - **General:** 工程概述信息，包括工程路径信息和所使用的综合工具；
 - **Target Device:** 工程器件信息，包括封装信息、速度等级和内核电压。











3.2.6 Help 菜单栏

- **View Help:** 查看软件编译过程中的输出信息帮助文档
- **Contact Us:** 官网联系我们地址
- **Manage License:** 许可证管理，详细的许可证配置管理方法请参考 [SUG501, Gowin 云源软件快速安装启动指南](#)
- **About:** 软件版本信息

3.3 工具栏

提供一些常用功能的快速访问入口，从左至右依次为：

- “” (Ctrl+N): 新建文件或工程 (New File or Project)
- “” (Ctrl+O): 打开文件或工程 (Open File or Project)
- “” (Ctrl+S): 保存文件 (Save)
- “” (Ctrl+Shift+S): 一键保存所有被打开编辑过的文件 (Save All)
- “” (Ctrl+P): 打印 (Print)
- “” (Ctrl+Z): 撤销 (Undo)
- “” (Ctrl+Y): 重做 (Redo)

- “” (Ctrl+X): 剪切 (Cut)
- “” (Ctrl+C): 复制 (Copy)
- “” (Ctrl+V): 粘贴 (Paste)
- “” (Ctrl+F): 查找 (Find)
- “” : 启动在线逻辑分析仪 (Gowin Analyzer Oscilloscope)
- “” : 启动设计 HDL 原理图查看器 (Schematic Viewer)
- “” : 启动 IP 核产生器 (IP Core Generator)
- “” : 启动编程器 (Programmer)
- “” : 执行综合 (Run Synthesis)
- “” : 执行布局和绕线 (Run Place Route)
- “” : 执行综合、布局和绕线 (Run All)

3.4 工程管理区 (Design)

提供工程及其相关文件的管理和显示功能，显示或编辑工程所用器件信息、用户设计文件、用户设计约束文件以及配置文件等。

3.5 过程管理区 (Process)

提供用户 FPGA 设计流程，包括综合 (Synthesize)、布局布线 (Place & Route) 以及下载码流文件 (Program Device)，同时可双击启动时序约束和物理约束工具进行约束文件编辑。

3.6 设计层级显示区 (Hierarchy)

加载设计文件后，软件会先对设计文件进行解析，Hierarchy 窗口会显示当前工程的设计层级关系。通过 Hierarchy 窗口可以定位某个 module 的定义以及实例化在设计文件中的位置，还可以将某个 module 设为 top module。在 Hierarchy 窗口中，Unit 列显示设计文件的 module 层级结构，Files 列显示 module 定义所在的文件。目前 Hierarchy 已支持对 Verilog 语言、VHDL 语言以及 System Verilog 语言的解析。

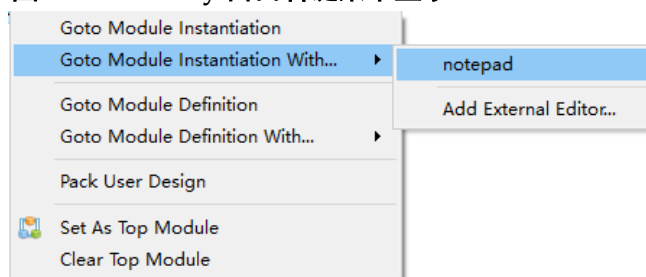
3.6.1 右键菜单

Hierarchy 窗口中 module 处的右键菜单中目前已支持的功能：

- **Goto Module Instantiation**: 跳转到该 module 在源文件中的实例化位置，默认用云源自带的编辑器打开。如果在菜单栏 Tools > Options > External Editor 中配置了第三方编辑器且勾选了 “Always Use External Editor”，Goto Module Instantiation 会默认用第三方编辑器打开源文件；

- **Goto Module Instantiation With...**: 跳转到该 module 在源文件中的实例化位置，在右键菜单中提供已设置的第三方编辑器和“Add External Editor”，如图 3-2 所示。如果选择“Add External Editor”，则弹出“External Editor”的 Options 选项对话框，可以对第三方编辑器进行设置；
- **Goto Module Definition**: 跳转到该 module 在源文件中的定义位置，默认用云源自带的编辑器打开。如果在菜单栏 Tools > Options > External Editor 中配置了第三方编辑器且勾选了“Always Use External Editor”，Goto Module Definition 会默认用第三方编辑器打开源文件；
- **Goto Module Definition With...**: 跳转到该 module 在源文件中的定义位置，在右键菜单中提供已设置的第三方编辑器和“Add External Editor”，如图 3-2 所示。如果选择“Add External Editor”，则弹出“External Editor”的 Options 选项对话框，可以对第三方编辑器进行设置；
- **Pack User Design**: 加密用户设计，可对该 module 及其子 module 进行加密；
- **Set As Top Module**: 将该 module 设为 top module，被设置为 top 的 module 会添加标记“🏠”表明当前 module 是 top module，且原有的 hierarchy 层级结构不变；
- **Clear Top Module**: 清除该 module 的 top module 设置。

图 3-2 Hierarchy 窗口右键菜单显示

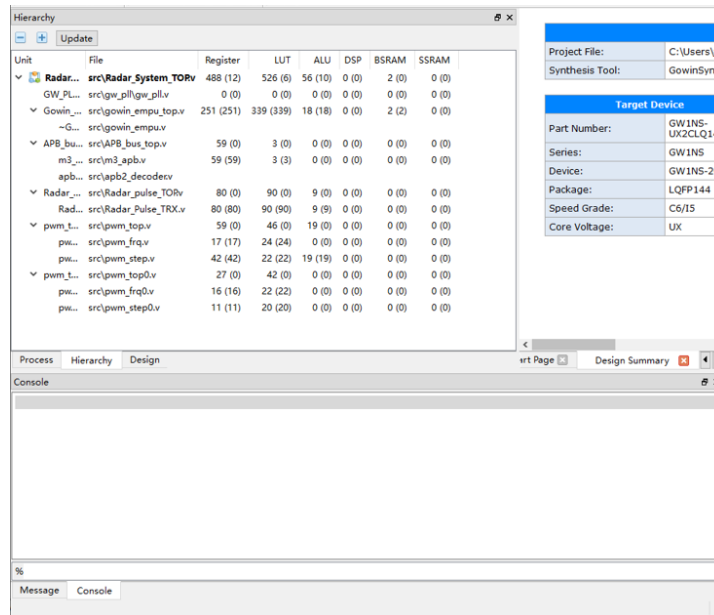


如果对工程文件进行 hierarchy 解析时存在 error 信息，在 Hierarchy 窗口右上方会显示红色字体标识的“RTL Analysis Error”提示，单击该提示会弹出提示框，提示框内为具体的 Error 信息。

3.6.2 资源显示

综合完成后 Hierarchy 窗口会自动显示当前工程设计中的资源信息，如图 3-3 所示。如果设计中 module 的定义为加密模块，则不会显示它的资源信息，该加密模块的资源信息会被统计到其上层模块中。每个 module 使用的各类资源会显示两个数据，如图 3-3 中 module Radar_System_TOP 的 Register 资源数据 488 (12)，其中，12 为该 module 自身所使用的 Register 个数，488 为该 module 及其子 module 所使用的 Register 的总数。

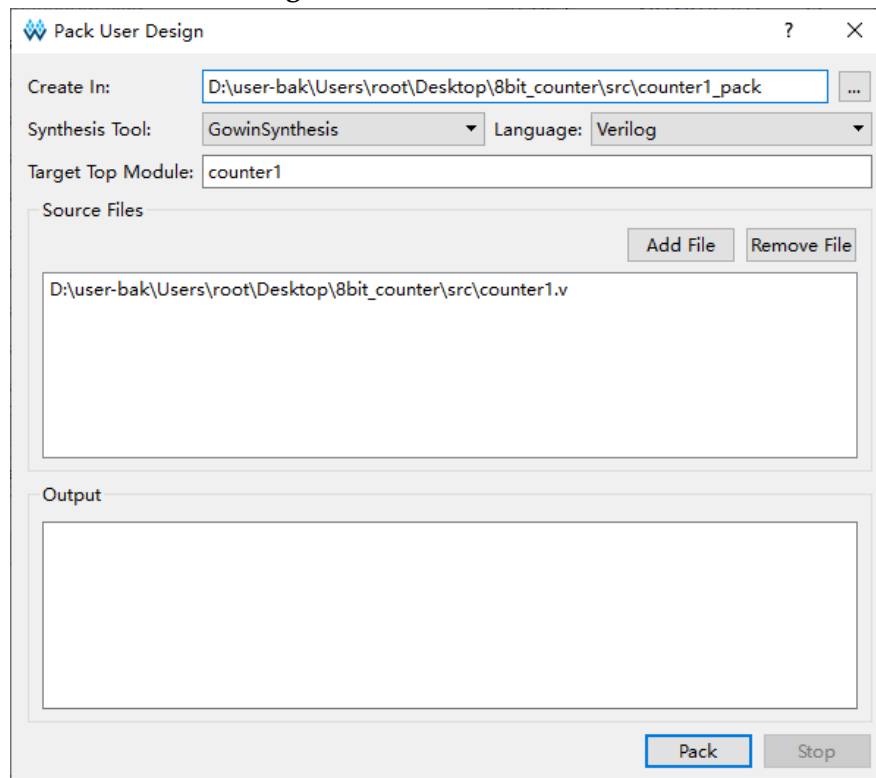
图 3-3 Hierarchy 窗口资源信息显示



3.6.3 文件加密

在打开工程的情况下，当用户需要对 FPGA 设计中全部源文件或者部分源文件进行加密操作时，可以在 Hierarchy 窗口中右击所要加密的 module，通过右键菜单中的“Pack User Design”生成综合后的加密文件。“Pack User Design”对话框如图 3-4 所示。

图 3-4 Pack User Design 对话框



Pack User Design 对话框各项配置含义如下：

- **Create In:** 生成加密文件的目标路径，仅支持绝对路径的形式，默认为工程路径\src\- **Synthesis Tool:** 指定综合工具;
- **Language:** 选择硬件语言，下拉列表支持 Verilog 和 VHDL，默认选择 Verilog;
- **Target Top Module:** 想要进行加密的 top module。默认为在 Hierarchy 窗口选择的 module，允许用户进行修改;
- **Source Files:** 列出 Hierarchy 窗口选择的 module 及其 sub module 所在的源文件;
- **Add File:** 添加待加密的设计文件;
- **Remove File:** 移除无需加密的设计文件;
- **Output 窗口:** 执行信息窗口;
- **Pack:** 执行加密;
- **Stop:** 终止加密。

开始加密及加密成功后会在 **Output** 窗口打印相关信息，如图 3-5 所示。加密过程中如果有 **error** 信息，会把 **error** 信息显示到 **Output** 窗口，同时会打印加密失败的信息，如图 3-6 所示。

图 3-5 Pack User Design 对话框打印信息

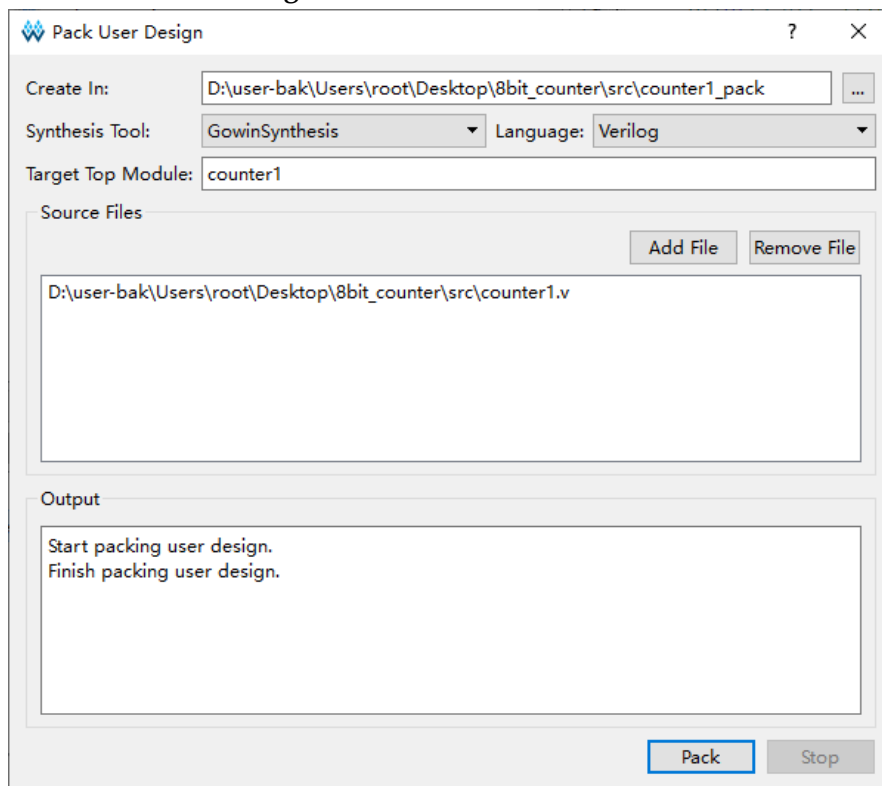
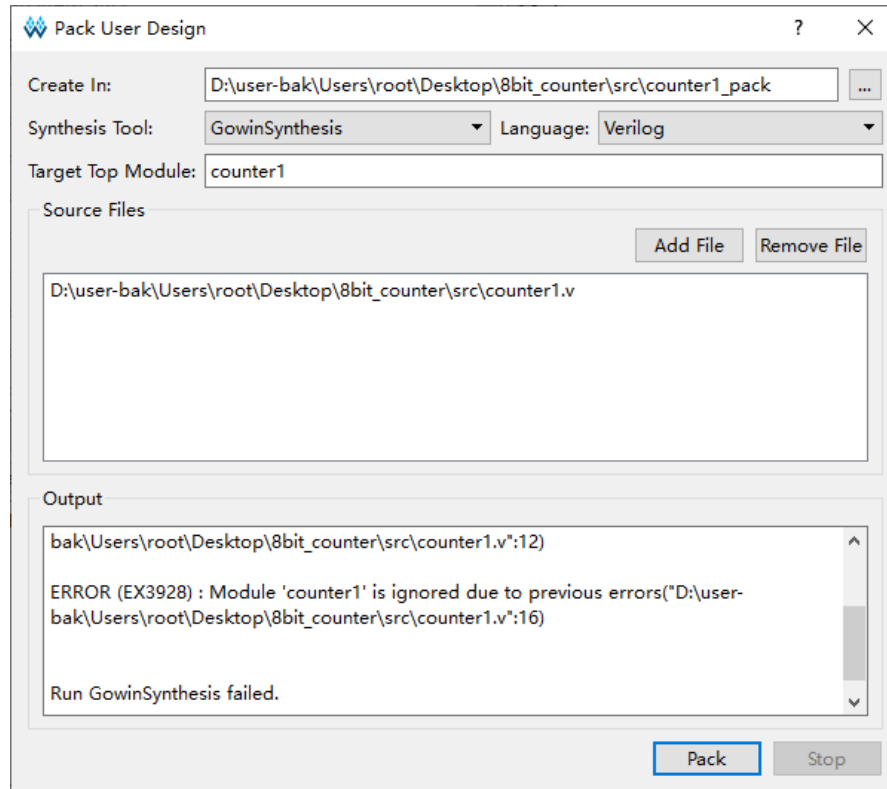


图 3-6 Pack User Design 对话框提取错误信息



加密完成后会在目标路径下生成两个文件，如果 Language 选择的是 Verilog，则生成的文件为<topmodule_name>_gowin.vp 和 <topmodule_name>_sim.v，如果 Language 选择的是 VHDL，则生成的文件为<topmodule_name>_gowin.vhdp 和<topmodule_name>_sim.v。<topmodule_name>_gowin.vp 和<topmodule_name>_gowin.vhdp 是用于保密的加密文件，可以提供给他人使用。<topmodule_name>_sim.v 是打平的综合后明文网表文件，可以用于加密模块的功能仿真。

注！

如果工程中有多个 module 都实例化了同一个 sub module，分别 pack 这几个 module 后产生的文件中都会有该 sub module 的定义，如果将产生的这几个文件用于同一个工程，会报出 sub module 被重复定义的错误，该使用方式需要规避。


3.7 源文件编辑区

提供基本的文件编辑、查看及选中关键词高亮功能。

新建或打开的文件、综合后生成的文件以及 Place & Route 后生成的文件均可显示在文本编辑区，另外“Start Page”及工程的“Design Summary”同样会显示在文本编辑区。

如文件显示在文本编辑区内，在外部对文件进行了修改操作，则在文本编辑区内会弹出“File Changed”提示，选择“Reload”则会重新加载该文件。

单击“File”菜单栏中“Close”选项或文本编辑区当前显示文件名右

侧“”，会关闭文本编辑区内当前显示的文件。

单击“File”菜单栏中“Close All”选项，则会关闭文本编辑区内显示的所有文件。

打开某个文件后，可以通过快捷键 **Ctrl+F** 或单击工具栏中的 **Find & Replace** 来打开“Find & Replace”对话框。对话框中的 **Find All** 选项可以选择三个查找范围：**Current File**、**Open Files** 以及 **Current Project**，如图 3-7 所示。单击 **Find All** 后，软件下方会弹出“Search Result”窗口，搜索内容会高亮显示，匹配结果的总数会显示在首行的行尾处，如图 3-8 所示。

图 3-7 Find & Replace 对话框

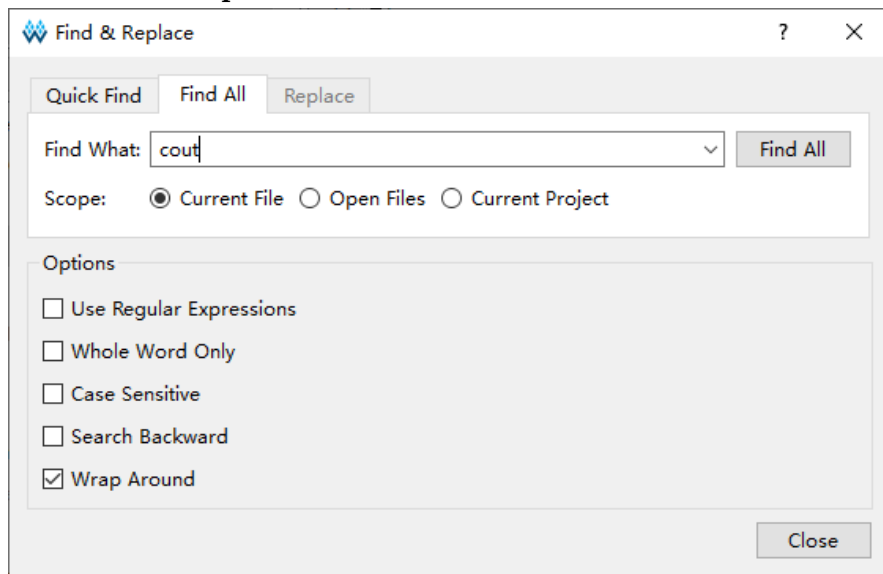
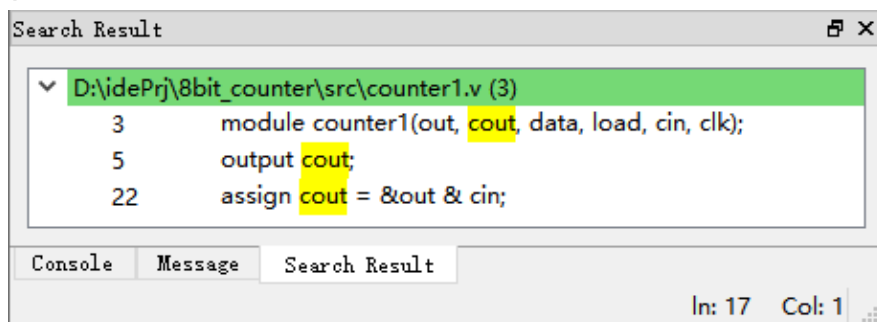


图 3-8 Search Result 窗口



3.8 信息输出区

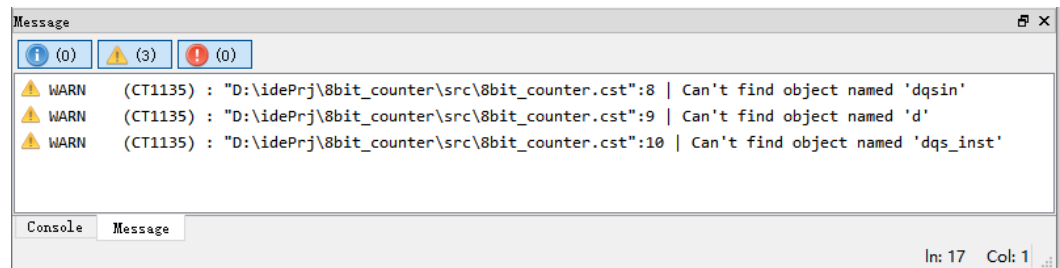
显示软件在运行过程中的处理信息，可手动切换标签页查看不同类型的输出信息：

- 全部信息（Console 窗口），包括 Tcl 命令窗口、警告信息、错误信息等；
- 信息汇总（Message 窗口），包括 Note 信息、警告信息、错误信息。

在 Console 窗口右击选择“Clear”，可清空所有页的信息。Message 窗口包含 Note、Warning 和 Error 三个子选项，可控制 Message 窗口只显

示 Note 信息、Warning 信息或 Error 信息，每个子选项会在相关的信息页标签处记录出现信息的个数，如图 3-9 所示。在 Message 窗口右击选择“Clear”，可清空当前页面的信息。

图 3-9 信息输出窗口



选中 PnR 报出的某条 Error 或 Warning 信息后右击选择“Help”，或者按快捷键“F1”，会弹出该条 Error 或 Warning 的“GOWIN Help”帮助文档，文档中会详细描述该条 Error 或 Warning 的帮助信息。一些常见的 Warning 或 Error 信息如表 3-1 所示，详细的帮助文档可通过菜单栏 Help > View Help 查看，该帮助文档支持中英文显示。

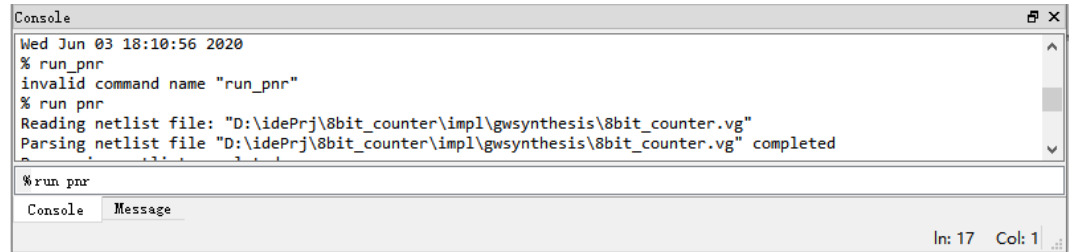
表 3-1 常见 Warning 和 Error 信息

| 名称 | 代码 | 描述 |
|---------|--|----------------------|
| Warning | WARN (PA1002): <file>:<line> Invalid parameterized value <value>(<parameter>) specified for instance <instanceName> | 指定位置处的器件设置了错误的参数值。 |
| | WARN (PA1008): <file>:<line> Object <name> is already defined | 存在连线或接口的重复定义。 |
| | WARN (PA1001) : Dangling net <netName>(source:<instanceName>) in module <moduleName> has no destination | 指定模块中的连线没有连接目标。 |
| | WARN (CT1098) : <file>:<line> Group name <name> is already defined | 约束文件中存在约束组的重复定义。 |
| | WARN (CT1101) : <file>:<line> Location column <number> is out of chip range(<maxColumn>) | 约束位置信息中的列超出了芯片的范围。 |
| Error | ERROR (PA2000): <file>:<line> Syntax error near token <name> | 指定位置处存在语法错误。 |
| | ERROR (PA2001): <file>:<line> Module <moduleName> is already defined | 存在模块的重复定义。 |
| | ERROR (PA2017): The number(<value>) of <instType> in the design exceeds the resource limit(<maxValue>) of current device | 设计文件中模块的数量超出了该器件的总数。 |
| | ERROR (PA2025): No <instType> resource in current device | 设计文件中含有本芯片不支持的器件资源。 |
| | ERROR (PA2054): <file>:<line> <name> is already declared | 设计文件中存在器件名称重复。 |

在 Console 信息输出窗口的最下方是 Tcl 命令编辑窗口，可以在窗口中输入 Tcl 命令后通过回车键来执行相关命令，如 [8_Tcl 命令说明](#)。

图 3-10 所示。详细的 Tcl 命令使用方法见本文档 [8_Tcl 命令说明](#)。

图 3-10 Tcl 编辑窗口



4 云源使用

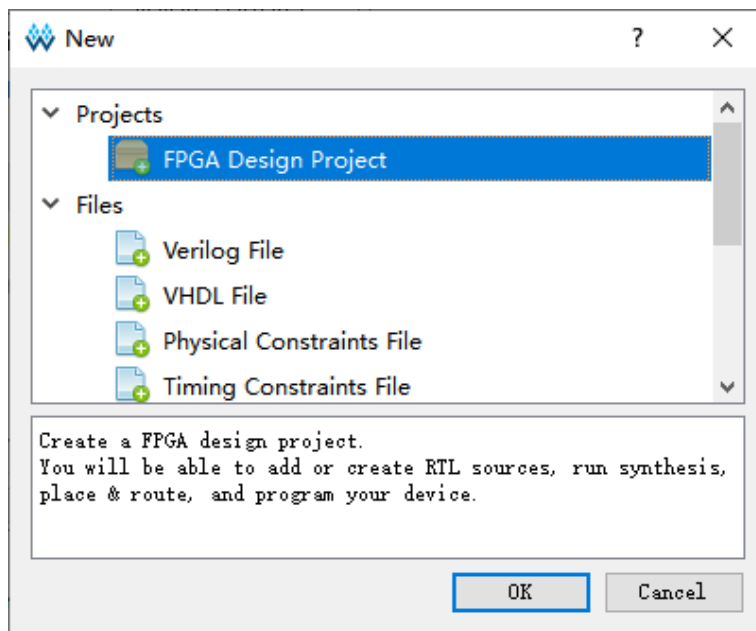
云源支持界面模式和命令行模式运行，命令行模式的使用可参考本文档的 8 Tcl 命令说明。

以 Windows10 版本云源界面运行为例，介绍云源的使用方法。

4.1 新建工程

1. 单击“File > New...”，打开“New”对话框，如图 4-1 所示。

图 4-1 新建工程

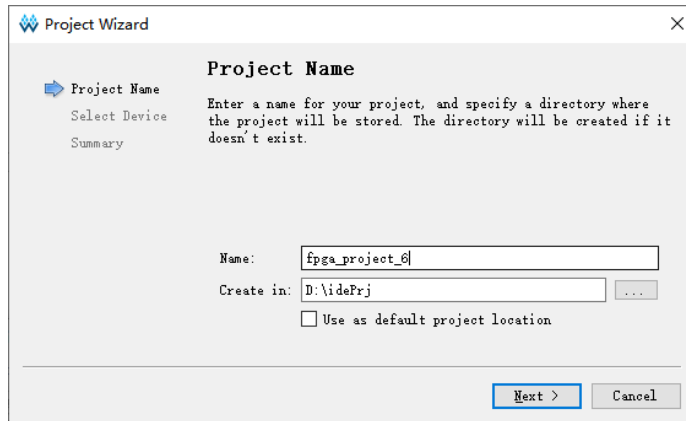


注！

也可通过以下三种方式打开“New”对话框：

- 使用快捷键 **Ctrl+N**；
 - 单击工具栏上“New File or Project”图标；
 - 单击 Start Page 窗口上“Quick Start > New Project”。
2. 打开“New”对话框后默认已选择“FPGA Design Project”，单击“OK”，打开“Project Wizard”对话框，如图 4-2 所示。

图 4-2 新建工程向导



3. 创建工程名和路径，如图 4-2 所示：

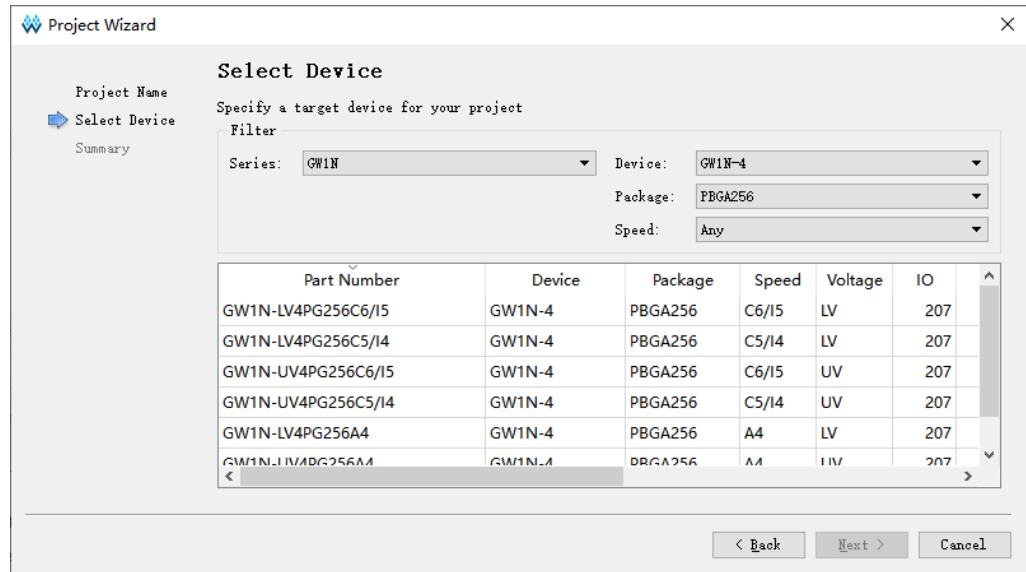
- a) 在“Name”文本框中输入工程名；
- b) 单击“...”选择工程路径。

若选中“Use as default project location”，即会将该工程路径设置为默认路径，下次新建工程时会默认创建在该路径。

注！

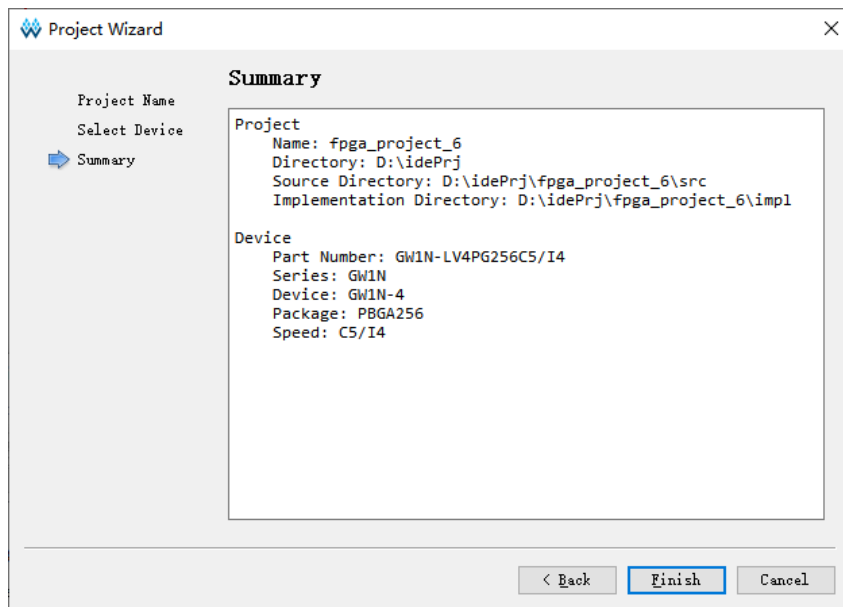
- Windows 和 Linux 下的文件路径长度均有限制，Windows 系统路径长度限制 260 个字符，Linux 系统限制 4096 个字符。在路径下存在文件长度超出系统限制的情况下，删除或者拷贝路径都会失败；
 - 与 Linux 不同，在 Windows 中路径分隔符为“\”，如 E:\Gowin\ide。
4. 单击“Next”，设置 FPGA 的器件信息，共有四个过滤选项 Series、Device、Package 和 Speed。
- 在 Series 中过滤芯片系列
 - 在 Device 中过滤器件
 - 在 Package 中过滤封装类型
 - 在 Speed 中过滤速度等级
 - 在“Part Number”栏中选择芯片型号，并且该栏中会显示所选择器件的详细资源信息，如图 4-3 所示。

图 4-3 设置 FPGA 器件信息



5. 单击“Next”，在 Summary 中核对新建 FPGA 工程的工程信息及器件信息，如图 4-4 所示。

图 4-4 工程信息




6. 单击“Finish”，工程创建完成。

4.2 打开工程

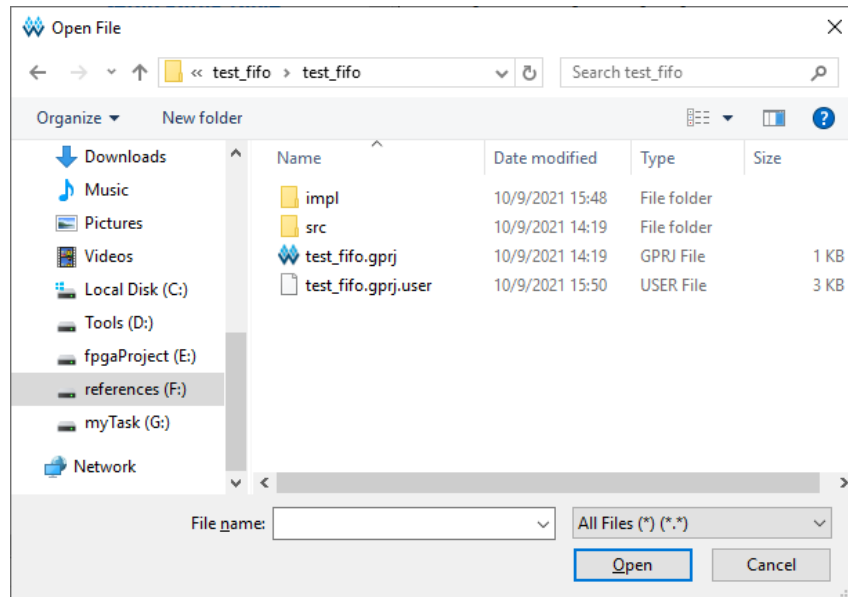
用户可直接通过云源界面打开已创建的高云软件工程，有以下五种方式打开工程：

工具栏打开

1. 单击工具栏上的“”图标，打开“Open File”对话框，如图 4-5 所示；

2. 选择工程文件*.gprj，单击“Open”，打开工程。

图 4-5 打开文件



菜单栏打开

1. 菜单栏中，选择“File > Open ...”打开“Open File”对话框，如图 4-5 所示；
2. 选择工程文件*.gprj，单击“Open”，打开工程。

Start Page 打开

1. 在“Start Page”页面中，单击“Open Project...”，打开“Open Project”对话框；
2. 选择工程文件*.gprj，单击“Open”，打开工程。

Recent Projects 打开

菜单栏中，单击“File > Recent Projects”，选择需打开的工程。

注！

- 亦可在“Start Page > Recent Projects”列表中，选择需打开的工程；
- “Recent Projects”为最近打开过的工程列表，最多会保留 10 个工程；
- 若该工程已被删除，会弹出“Open Project”提示框，提示找不到指定路径。

工程文件打开

双击工程文件*.gprj，自动使用最近一次安装的云源打开该工程。

4.3 编辑工程

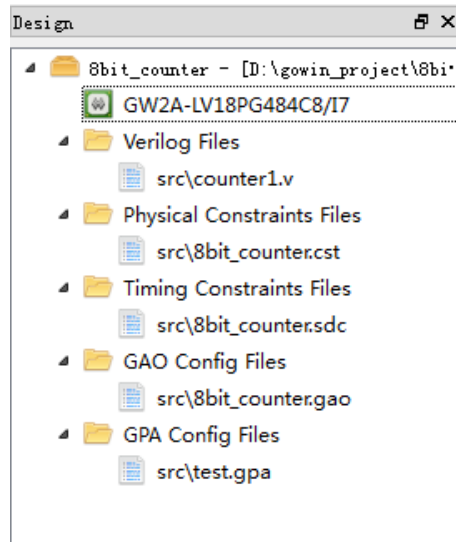
新建或打开工程后，可在工程管理窗口对 FPGA 工程器件信息及相关文件等进行编辑，如图 4-6 所示。

工程管理区主要包含以下信息：

- FPGA 工程的路径信息；

- 使用的芯片型号；
- 当前工程包含的文件信息，包括用户设计文件（**Source Files**）、约束文件和配置文件。其中，约束文件包括物理约束文件（.cst）、时序约束文件（.sdc），配置文件包括 GAO 配置文件（.gao、.rao）、功耗分析配置文件（.gpa）等。

图 4-6 工程文件窗口



4.3.1 编辑工程器件

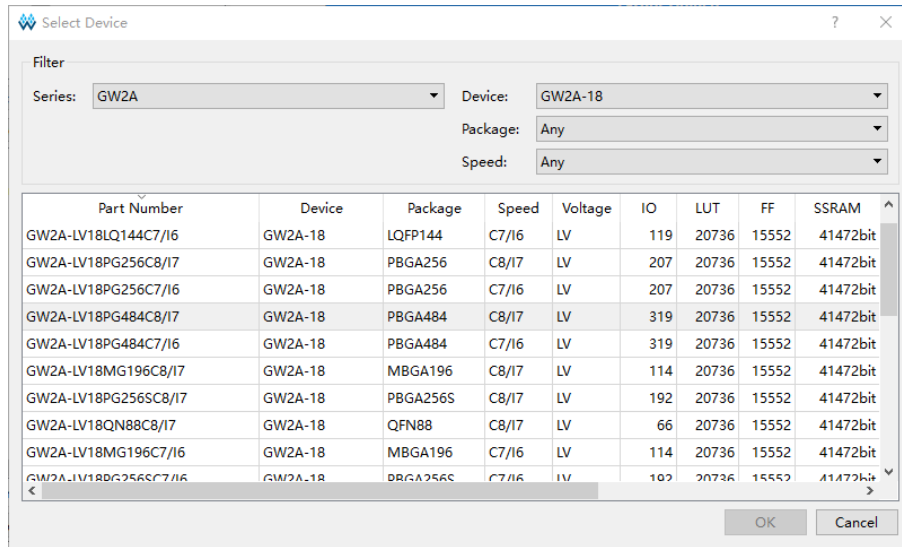
可在工程管理窗口内编辑当前 FPGA 工程所使用的芯片型号：

1. 如图 4-6 所示，双击“GW2A-LV18PG484C8/I7”，打开“Select Device”对话框，也可以通过菜单栏 Project 下拉列表中的 Set Device 项打开，如图 4-7 所示；
2. 在“Select Device”对话框中设置 FPGA 的器件信息，在“Part Number”栏中选择芯片型号，即可编辑当前工程所用的器件。

注！

“Part Number”栏中会显示所选芯片的详细资源信息且要与所使用的芯片保持一致。

图 4-7 工程配置芯片型号



4.3.2 编辑工程文件

工程中需添加的文件类型包括用户 RTL 设计文件（Source Files）、约束文件和配置文件。下面介绍如何编辑工程所需文件。

新建设计及约束文件


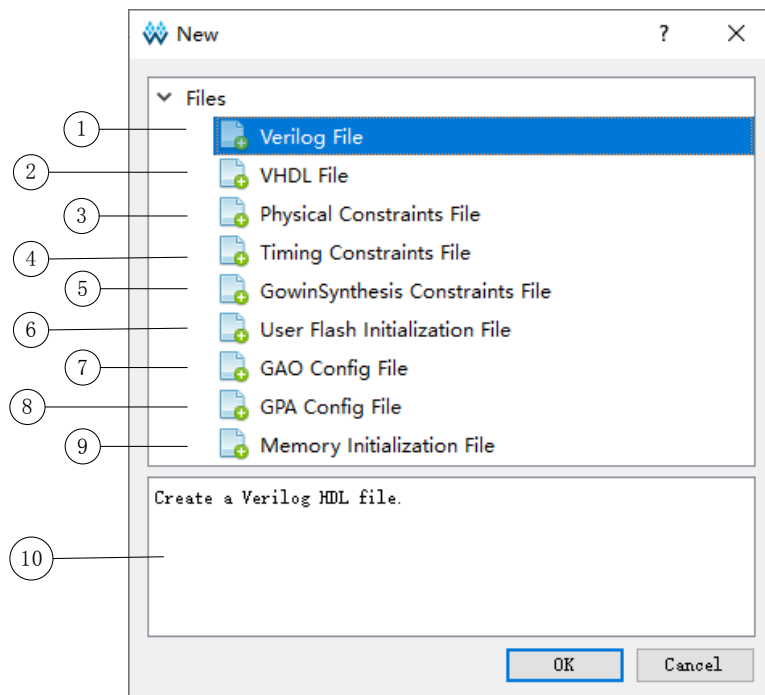
1. 单击工具栏中的新建工程或文件图标“”，或通过菜单栏 File 下拉列表中的“New...”项打开 New 对话框；
2. 选择需新建的文件类型，如图 4-8 所示；

图 4-8 新建文件对话框

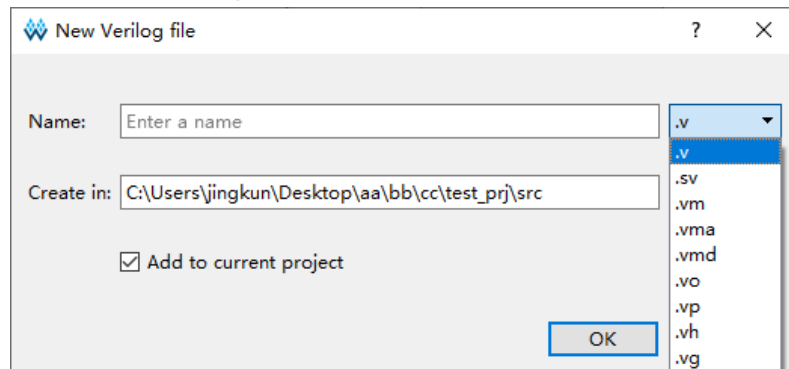


- ① 用户 Verilog 设计文件 ② 用户 VHDL 设计文件
 ③ 物理约束文件 ④ 时序约束文件

- | | |
|---------------|--------------------|
| ⑤ 综合优化约束文件 | ⑥ User Flash 初始化文件 |
| ⑦ 在线逻辑分析仪配置文件 | ⑧ 功耗分析配置文件 |
| ⑨ 块存储器初始化文件 | ⑩ 文件类型解释区 |

- 以新建 Verilog File 为例，选中“Verilog File”打开新建 Verilog File 对话框，可选择文件扩展格式，如图 4-9 所示。默认勾选“Add to current project”，即新建的设计文件默认会加到当前工程中；

图 4-9 新建 Verilog File 对话框



- 填写文件名，单击“OK”。

新建配置文件


- 单击工具栏中的新建工程或文件图标“”，或通过菜单栏 File 下拉列表中的“New...”项打开 New 对话框，如图 4-8 所示；
- 在图 4-8 中，选择需新建的配置文件类型。以新建功耗分析配置文件为例，选中“GPA Config File”打开新建 New GPA Config File 对话框，如图 4-10 所示，输入文件名，单击“OK”，新建的 GPA 配置文件会自动添加到工程管理区；
- 在工程管理区双击配置文件，以窗口形式打开空白配置文件，进行编辑，如图 4-11 所示。

图 4-10 新建配置文件对话框

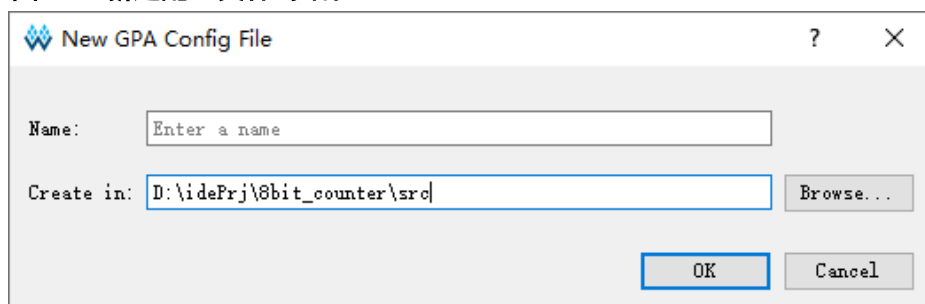
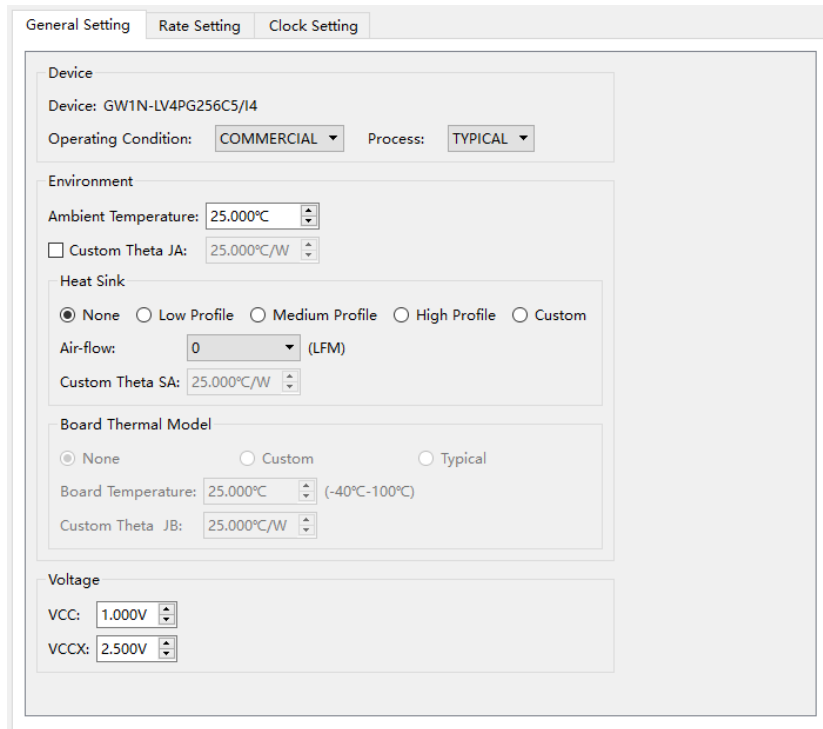


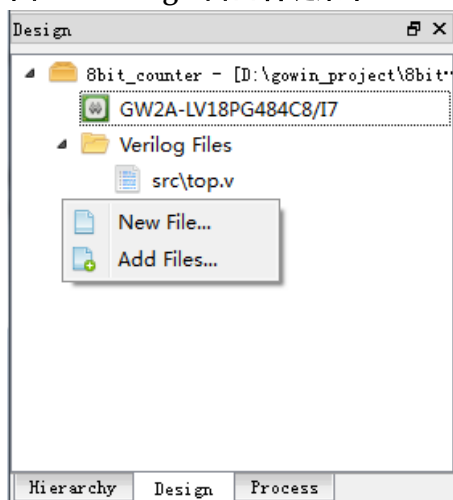
图 4-11 GPA 配置文件窗口



添加工程文件

1. 如图 4-12 所示，在工程管理区空白处右击选择“Add Files...”，打开“Select Files”对话框；
2. 选择工程文件，可同时选择多个文件或单个文件添加，完成添加用户设计文件。添加后，软件会自动在工程文件管理区将文件进行分类，如果添加的文件不是 RTL 设计文件、网表文件、约束文件、GPA 配置文件、GAO 配置文件，则在工程管理区会增加一个文件分类“Other Files”。

图 4-12 Design 窗口右键菜单

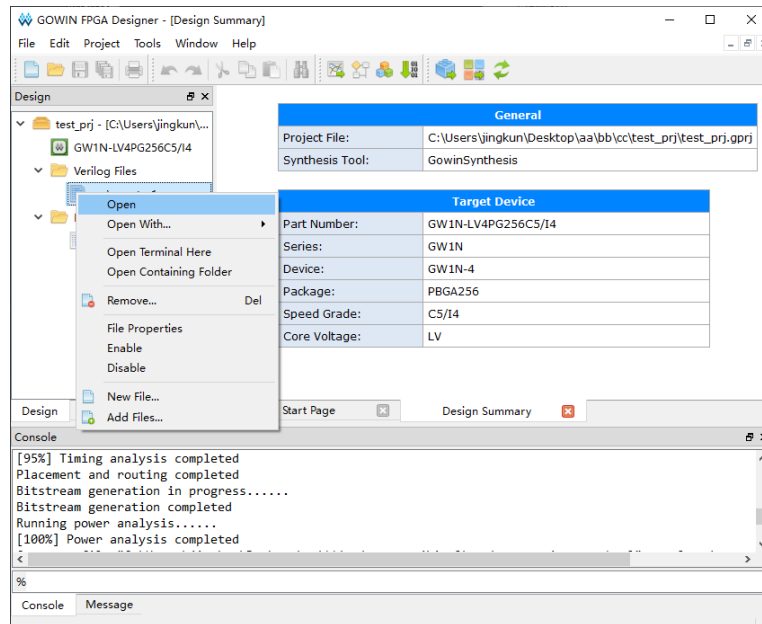


修改工程文件

如图 4-13 所示，可通过以下两种方式打开文件：

1. 双击工程管理区内的任一文件，文件即会显示在源文件编辑区；
2. 右击需要修改的文件，单击“Open”。

图 4-13 工程文件编辑菜单

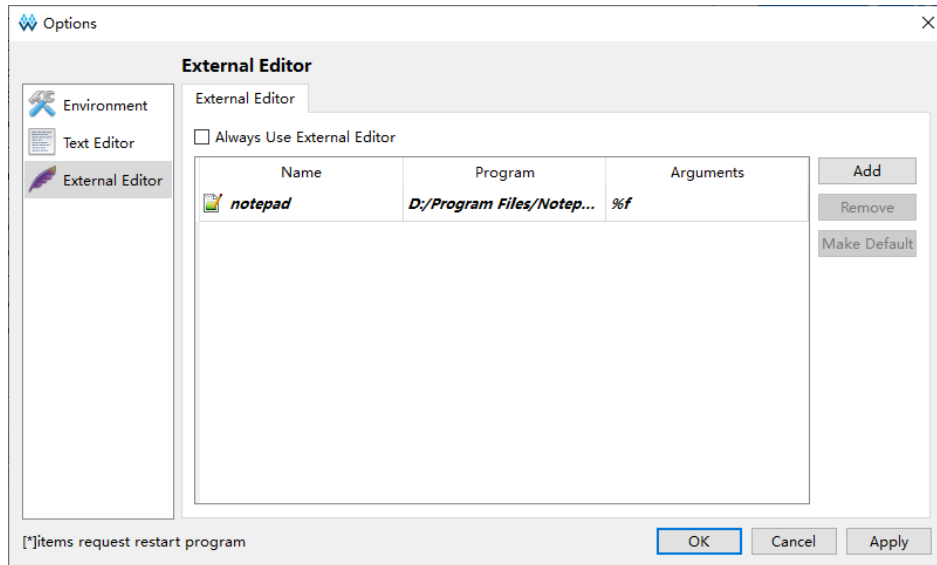


如果用户已经通过菜单栏 **Tools > Options** 配置了第三方文本编辑软件，选择“**Open With...**”可以用第三方的文本编辑软件打开设计文件。如果选择的是“**Add External Editor**”，用户打开 **Options** 对话框添加其他外部编辑器，如图 4-14 所示。如果勾选了“**Always Use External Editor**”，则会始终使用外部编辑器打开设计文件。在右键菜单中选择“**Open Containing Folder**”可打开文件所在的文件夹，选择“**Open Terminal Here**”可打开命令行运行窗口，用户可通过命令行模式运行。

如果用户使用外部编辑器对已经在云源编辑区打开的文件进行修改并保存，云源会弹出文件已变更是否重新加载的提示。

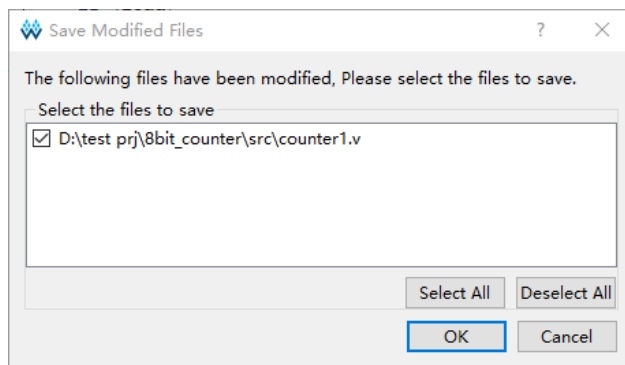
如果用户关闭编辑后未保存的文件，云源会弹出是否保存更改文件的提示框。

图 4-14 外部编辑器



在工程文件修改后,如果忘记保存直接执行 Synthesize 或 Place & Route, 会弹出“Save Modified Files”的对话框,如图 4-15 所示。

图 4-15 Save Modified Files 对话框



单击“OK”后,文件会被保存且自动执行弹出对话框之前想要执行的流程。如果单击“Cancel”,则文件不会被保存,且不会再执行 Synthesize 或 Place & Route。

删除工程文件

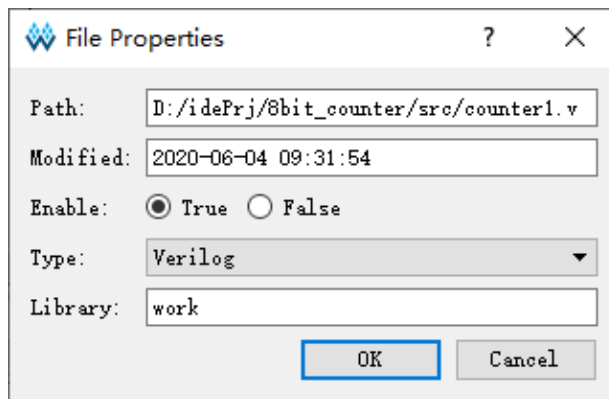
1. 选中工程管理区内的文件;
2. 右击选择“Remove”或直接按键盘“Delete”键,弹出“Delete File”对话框,若选中“Delete Permanently”复选框,则该文件从当前工程删除且在磁盘上删除,否则,该文件仅从当前工程删除。

编辑工程文件属性

在工程管理区内的任一文件处右击选择“File Properties”,会弹出该文件的文件属性对话框,如图 4-16 所示。对话框中包含该文件的路径信息、修改时间、是否 Enable、Type 以及 Library 信息。通过 Type 下拉菜单可以选择编辑文件的 Type 信息,单击“OK”后,该文件会在 Design 窗口中自动

移动到所选的 **Type** 类型下。**Library** 用于指定综合 VHDL 文件时所用到的编译库，默认为 **work**，如果有多个库，需要用分号分隔。

图 4-16 工程文件属性对话框



使能工程文件

在工程管理区内的任一文件处右击，弹出的右键菜单中有“**Enable**”和“**Disable**”选项，如图 4-13。文件被 **Enable** 时，参与工程编译过程，被 **Disable** 时不参与工程编译过程。

1. 通过右键菜单选项设置文件的 **Enable/Disable** (包括单独文件的设置及文件的批量设置)；
2. 选中多个设计文件 (**Verilog Files**、**VHDL Files** 或 **Netlist Files**) 时，**Enable** 和 **Disable** 都可用；
3. 对于同一种约束类文件或配置类文件，选中多个时，**Enable** 不可用，**Disable** 可用，如图 4-17 所示；
4. 对于同一种约束类文件或配置类文件，工程中只有一个会处于 **Enable** 状态，再次新建或添加时，会将前一个约束文件或配置文件 **Disable**；当选中多个不同类型的文件时，**Enable** 和 **Disable** 功能不可用，如图 4-18 所示。

图 4-17 选中多个同类型文件右键菜单列表

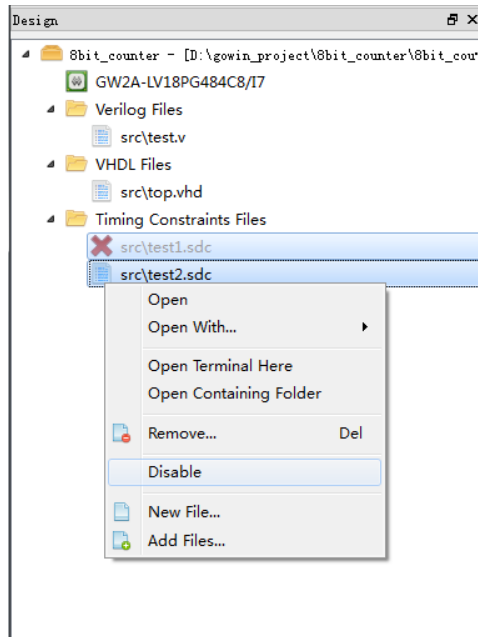
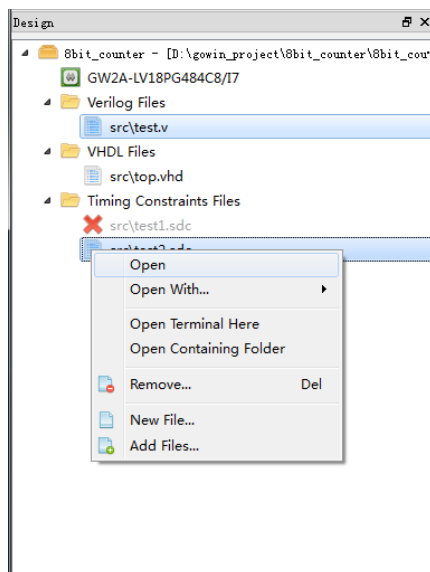


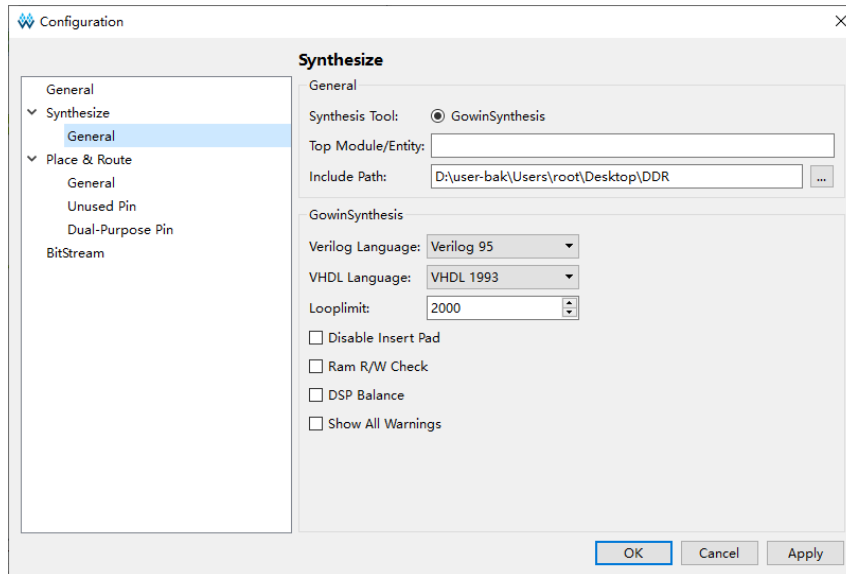
图 4-18 选中多个不同类型文件右键菜单列表



4.3.3 编辑工程配置

在工程管理区内的 Synthesize 处或 Place & Route 处右击选择“Configuration”，打开工程配置选项对话框，如图 4-19 所示。

图 4-19 工程配置选项对话框



如图 4-19 所示，可配置的工程选项包含 **General**、**Synthesize**、**Place & Route** 和 **BitStream**。有关可配置的各个工程选项的详细介绍如下：

General

工程基本选项配置，指定输出文件的基本名称，默认为当前工程的名称。

Synthesize

Synthesis Tool 为 GowinSynthesis[®]，可以对综合选项进行配置，如图 4-20 所示，将鼠标悬浮在部分选项处时会显示其解释。

图 4-20 GowinSynthesis 参数配置项

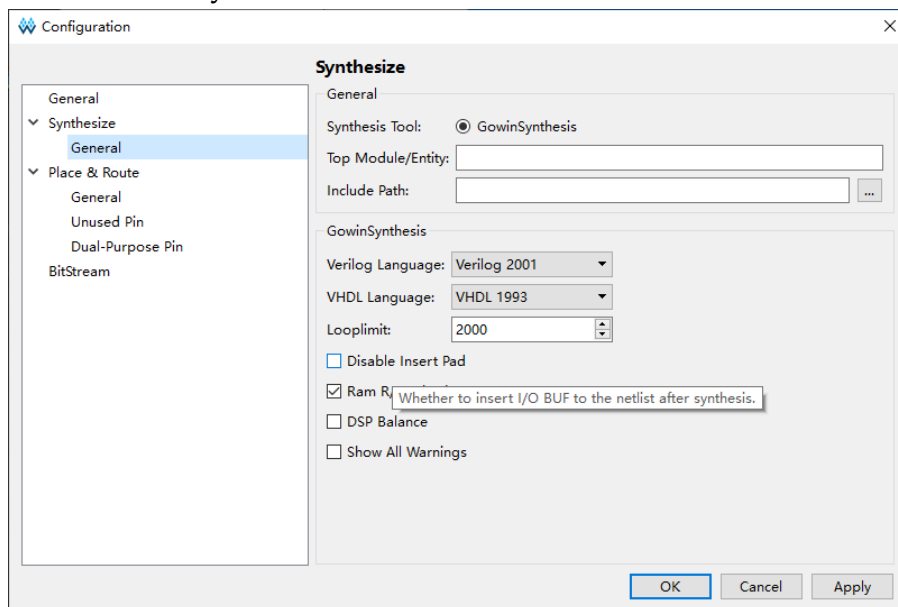


图 4-20 中的各项参数释义如下所示：

- **Top Module/Entity:** 指定顶层模块;
- **Include Path:** 指定文件包含路径;
- **Verilog Language:** 指定 verilog 语言, 支持 System Verilog 2017、Verilog 2001 和 Verilog 95, 默认值为 Verilog 2001;
- **VHDL Language:** 指定 VHDL 语言, 支持 VHDL1993 和 VHDL 2008, 默认值为 VHDL1993;
- **Looplimit:** 设置 RTL 中默认的编译器循环限制值, 默认值为 2000;
- **Disable Insert Pad:** 综合后的网表中是否插入 I/O Buffer, 默认不勾选;
- **Ram R/W Check:** 如果 RAM 存在读或写冲突, 勾选该选项后会在 RAM 周围插入旁路逻辑以防止仿真不匹配, 禁用该选项不会生成旁路逻辑, 默认为勾选;
- **DSP Balance:** 如果对设计进行综合时 DSP 资源超过了限制, 勾选该选项后将超出的 DSP 综合为 registers, 默认不勾选;
- **Show All Warnings:** 勾选该选项后综合过程中会打印出所有的 Warning 信息, 默认不勾选。

注!

关于所有选项配置的详细用法, 具体可参考 [SUG550, GowinSynthesis 用户指南](#)。

Place & Route

Place & Route 选项又包括 General、Unused Pin 和 Dual-Purpose Pin, 各选项含义如表 4-1 所示。

表 4-1 PnR 配置选项含义描述

| 工程选项 | 描述 |
|------------------|---|
| General | 用于配置运行布局布线时的基本参数 |
| Unused Pin | 用于对 unused pin (除去复用管脚的未使用管脚) 设置不同 IO 类型和属性 |
| Dual-Purpose Pin | 用于配置所选器件信息中封装方式对应的 I/O 信息, 主要用于配置复用管脚。 |

单击 **Reset all to default**, 当前配置页面会全部恢复为默认设置。**General** 选项配置如图 4-21 所示。

图 4-21 配置布局布线选项

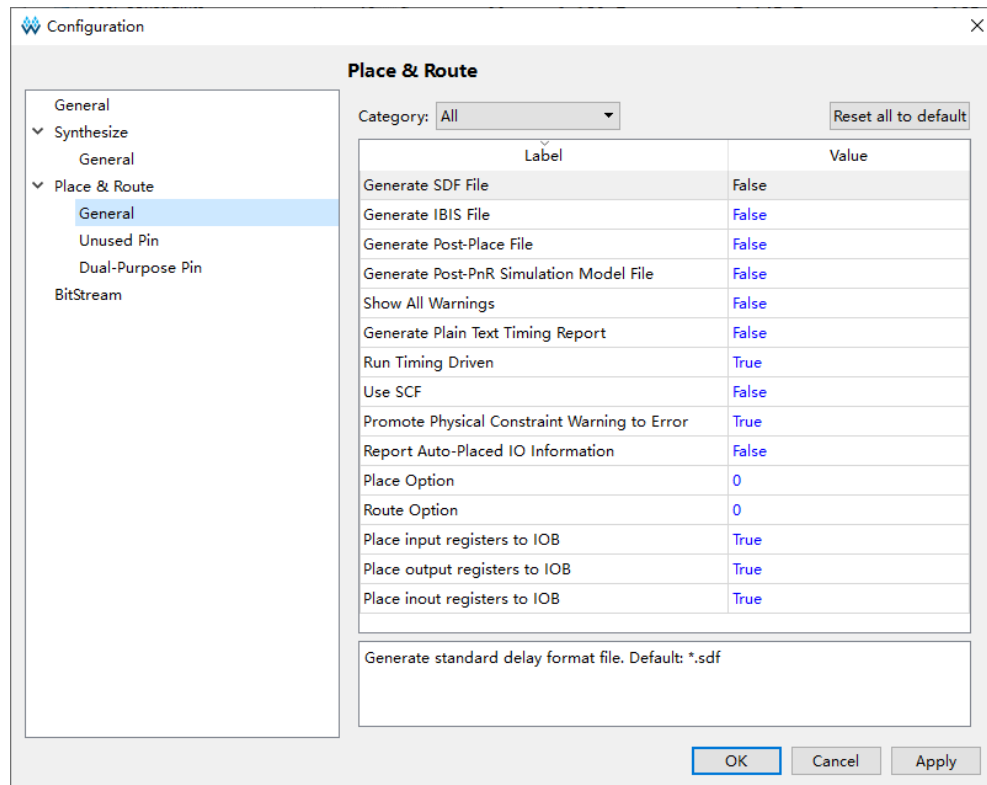


图 4-21 中的各选项使用说明如下：

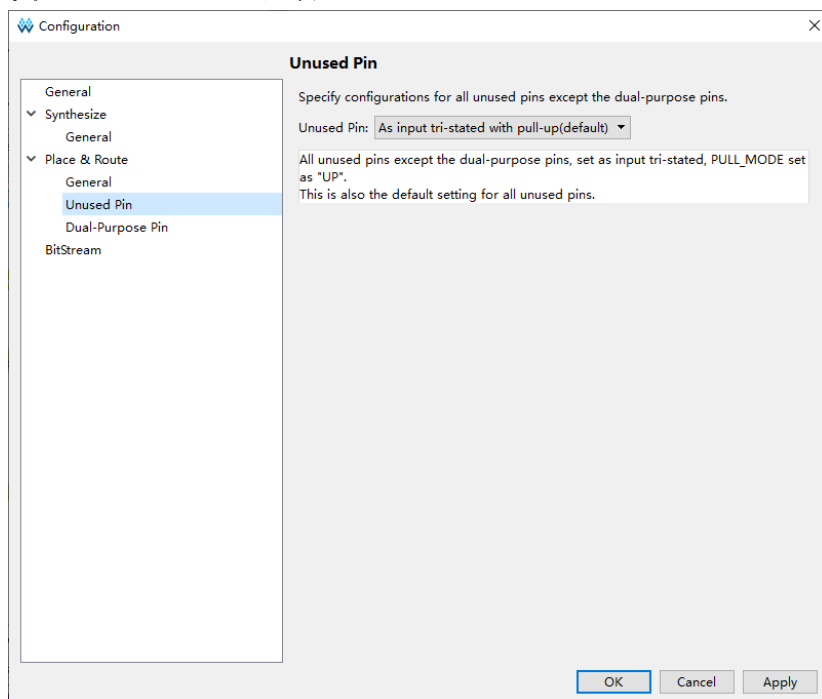
- **Generate SDF File:** 产生标准延迟格式文件，扩展名为.sdf，用于布局布线后的网表时序仿真，默认值为 **False**。具体使用方法参考本文档 7 仿真文件；
- **Generate IBIS File:** 产生输入/输出缓冲区信息指定文件，扩展名为.ibs，默认为 **False**；
- **Generate Post-Place File:** 产生只含有 BSRAM 布局信息的文件，扩展名为.posp，默认值为 **False**；
- **Generate Post-PNR Simulation Model File:** 产生时序仿真模型文件用于时序仿真，扩展名为.vo，默认值为 **False**；
- **Show All Warnings:** PNR 运行时输出所有的 **Warning** 信息，默认值为 **False**；
- **Generate Plain Text Timing Report:** 产生文本格式的时序报告，扩展名为.tr，默认值为 **False**；
- **Run Timing Driven:** 时序驱动优化布局布线结果，默认值为 **True**；
- **Promote Physical Constraint Warning to Error:** 将物理约束警告提升为错误信息，默认值为 **True**；
- **Report Auto-Placed IO Information:** 报告自动布局 IO 的位置信息，默认值为 **False**；

- **Place Option:** 布局算法选项，可选项有 0 和 1，默认值为 0:
 - 为 0 时，使用默认布局算法;
 - 为 1 时，在算法 0 的基础上，牺牲一些时间效率来尝试找到更优的布局结果。
- **Route Option:** 布线算法选项，可选项有 0、1 和 2，默认值为 0:
 - 为 0 时，采用默认布线算法，根据拥塞调整布线;
 - 为 1 时，根据时序数据调整布线;
 - 为 2 时，布线速度会相对快一点。
 如果算法 0 的布线算法效果不好时建议尝试 1 或 2 算法;
- **Place input register to IOB:** 布局输入 Buffer 驱动的寄存器到 IOB 上，默认值为 True;
- **Place output register to IOB:** 布局输出/三态 Buffer 驱动的寄存器到 IOB 上，默认值为 True;
- **Place inout register to IOB:** 布局双向 Buffer 驱动的寄存器到 IOB 上，默认值为 True。

Unused Pin

Unused Pin 选项可以对封装出的但未使用的管脚（不包括复用管脚）进行配置。有两个选项: As input tri-stated with pull-up (default) 、As open drain driving ground，如图 4-22 所示。

图 4-22 Unused Pin 选项



- **As input tri-stated with pull-up (default)** : 默认选项, 所有没有被用户设计用到的管脚 (不包括复用管脚) 会配置为输入三态弱上拉;
- **As open drain driving ground**: 所有没有被用户设计用到的管脚 (不包括复用管脚) 会配置为输出类型, 并且 **OPEN DRAIN** 配置为 **ON**。

Dual-Purpose Pin

Dual-Purpose Pin 选项是符合高云器件定制的复用管脚配置, 将鼠标悬浮在选项处会显示其解释。具体配置项目如图 4-23 所示。

图 4-23 配置复用管脚选项

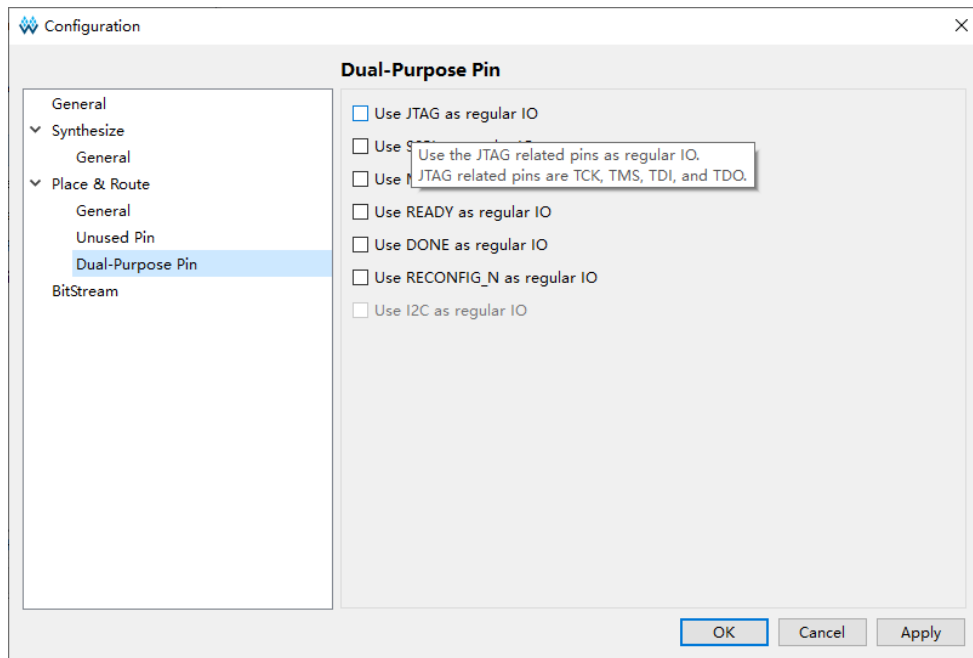


图 4-23 中的各项复用管脚说明如下:

- **Use JTAG as regular IO**: 将 JTAG 相关管脚复用为普通 IO 管脚, JTAG 相关的管脚为 TCK、TMS、TDI、TDO;
- **Use SSPI as regular IO**: 将 SSPI 相关管脚复用为普通 IO 管脚, SSPI 相关的管脚为 SCLK、CLKHOLD_N、SI、SO、SSPI_CS_N;
- **Use MSPI as regular IO**: 将 MSPI 相关管脚复用为普通 IO 管脚, MSPI 相关的管脚为 MCLK、MCS_N、MI、MO;
- **Use READY as regular IO**: 将 READY 相关管脚复用为普通 IO 管脚, READY 相关的管脚为 READY;
- **Use DONE as regular IO**: 将 DONE 相关管脚复用为普通 IO 管脚, DONE 相关的管脚为 DONE;
- **Use RECONFIG_N as regular IO**: 将 RECONFIG_N 相关管脚复用为普通 IO 管脚, RECONFIG_N 相关的管脚为 RECONFIG_N;
- **Use I2C as regular IO**: 将 I2C 相关管脚复用为普通 IO 管脚, I2C 相关的管脚为 SCL 和 SDA。支持的器件及封装有:

- GW1NR-2/GW1NR-2B/GW1NR-2C/GW2AN-9X/GW2AN-18X;
- GW1N-2 的封装 LQFP100X/LQFP144X/MBGA132X/QFN48E/MBGA132H/WLCSP42H/MBGA49;
- GW1N-2B 的封装 LQFP100X/LQFP144X/MBGA132X/MBGA132H/MBGA121X/GW138
- GW1N-2C 的封装 LQFP100X/LQFP144X/MBGA132X/MBGA132H/MBGA121X/GW138
- GW1N-1P5 的封装 LQFP100X
- GW1N-1P5B 的封装 LQFP100X/QFN48X
- GW1N-1P5C 的封装 LQFP100X/QFN48X

BitStream

BitStream 是符合高云芯片下载模式的码流文件格式选项以及下载频率选项等，将鼠标悬浮在选项处会显示其解释。具体配置如图 4-24 所示。

图 4-24 配置 Bitstream 文件选项

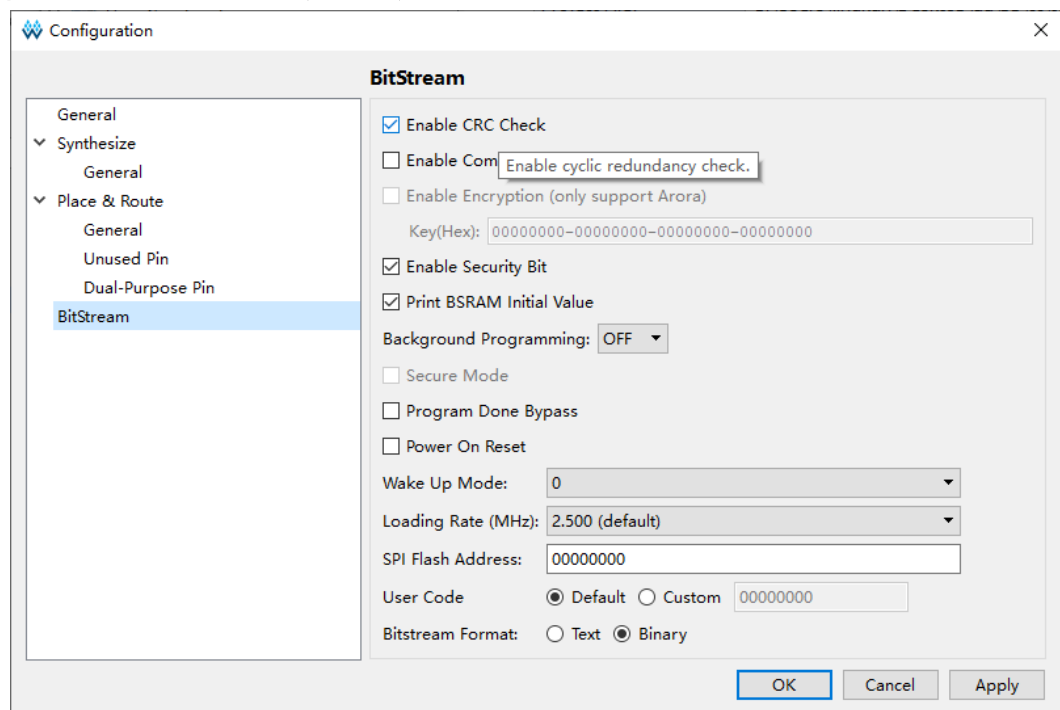


图 4-25 Background Programming 设为 I2C

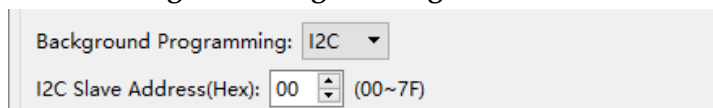


图 4-26 Background Programming 设为 I2C/JTAG/SSPI/QSSPI

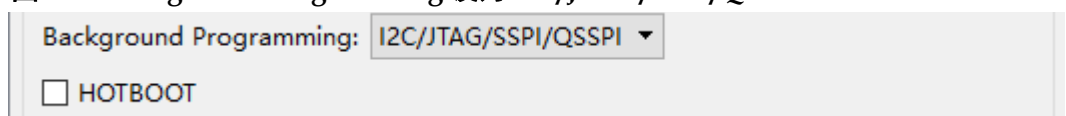


图 4-24 的各项参数配置使用说明如下：

- **Enable CRC Check:** 使能循环冗余校验，默认为勾选；
- **Enable Compress:** 使能码流文件压缩，默认为不勾选；
- **Enable Encryption (only support Arora):** 对码流文件进行加密处理，仅支持晨熙家族，默认为不勾选；
- **Key (Hex) :** 勾选“Enable Encryption(only support Arora)”后才可以对该项进行编辑，该项可以使用户对加密的密钥进行自定义，默认 key 全为 0；
- **Enable Security Bit:** 使能安全位控制，对码流文件添加安全位，添加之后码流无法再次回读，默认为勾选；
- **Print BSRAM Initial Value:** 打印 BSRAM 的初始值到码流文件中，默认为勾选；
- **Background Programming:** 背景升级功能，在不中断 FPGA 芯片当前正在执行的码流文件的前提下对 Flash 进行再次烧录。

支持背景升级的器件及其取值情况表 4-2 所示。

表 4-2 Background Programming 取值

| 器件类型 | Background Programming取值 |
|--|--|
| GW1N-1P5/GW1N-2/GW1NR-2、 GW1N-4B、 GW1NR-4B/GW1NR-4D/GW1NRF-4B、 GW1NS-4/GW1NSR-4、 GW1N-9/ GW1N-9C、 GW1NR-9/GW1NR-9C、 GW1NZ-1/GW1NZ-1C | OFF、JTAG 默认为OFF |
| GW1N-1P5B/GW1N-2B/GW1NR-2B、 GW1N-2C/GW1NR-2C/GW1N-1P5C | OFF、JTAG、I2C 默认为OFF |
| GW2AN-18X/GW2AN-9X | OFF、Internal、I2C/JTAG/SSPI/QSSPI 默认为OFF |

Background Programming 各取值功能介绍如下：

- **OFF:** 关闭 Background Programming 功能，如果器件为 GW2AN-18X 或 GW2AN-9X，“Dual-Purpose Pin”对话框中的“Use MSPI as regular IO”为不勾选且不可配置状态；
- **JTAG:** 使用 JTAG 模式进行背景升级；

- **I2C:** 使用 I2C 模式进行背景升级。对于器件 GW1N-1P5B/GW1N-2B/GW1NR-2B, 配置对话框中会出现选项 “I2C Slave Address(Hex)”, 对用户操作 I2C 设备的地址进行设置, 可设置值范围为 00~7F, 如图 4-25 所示。选择 I2C 后 “Dual-Purpose Pin” 对话框中的 “Use JTAG as regular IO” 为不勾选且不可配置状态。对于器件 GW1N-2C/GW1N-1P5C/GW1NR-2C, 使用 I2C 模式进行背景升级时, 配置对话框中不会出现选项 “I2C Slave Address(Hex)”, “Dual-Purpose Pin” 对话框中的 “Use RECONFIG as regular IO” 为不勾选且不可配置状态。
- **Internal:** 使用内部逻辑进行背景升级, “Dual-Purpose Pin” 对话框中的 “Use MSPI as regular IO” 为勾选且不可配置状态;
- **I2C/JTAG/SSPI/QSSPI:** 使用 I2C/JTAG/SSPI/QSSPI 模式进行背景升级, “Dual-Purpose Pin” 对话框中的 “Use MSPI as regular IO” 为不勾选且不可配置状态;
- 选择 **Internal** 或 **I2C/JTAG/SSPI/QSSPI** 时, 对话框中会出现配置选项 “HOTBOOT”, 如图 4-26 所示, 为可选状态, 默认不勾选;
- 配置项更改或 “HOTBOOT” 勾选状态更改后, **Place & Route** 的状态会变为过期状态。如果配置项切换前后包含 I2C, 则 **Synthesize** 和 **Place & Route** 的状态都会改为过期状态。

注!

当设计中含有 User Flash 时不能配置 Background Programming 选项, 否则会报出 Error 提示。

- **Secure Mode (device can be programmed only once)** : 启用安全模式, 此时 JTAG 管脚为 GPIO, 码流文件只能对设备编程一次。只有器件 GW1NSE-2C 和 GW1NSER-4C 支持该功能, 默认为不勾选状态;
- **Program Done Bypass:** 在 Done Final 内部信号生效时, 同时外部 Done Pin 保持低电平, 使码流加载完成后可以转发新的码流数据;
- **Power On Reset:** 使能上电复位控制, 默认为不勾选状态。勾选该选项后, 上电工作时会先进入复位模式, 将所有 RAM 位清除, 并通过内部弱上拉电阻将使用的 I/O 置为三态, 然后依次完成配置、初始化工作;
- **Wake Up Mode:** 使能芯片唤醒模式, 可选值有 0 和 1, 默认值为 0;
 - Wake Up Mode 为 0: DONE 管脚拉高或拉低对 Wake Up 没有影响;
 - Wake Up Mode 为 1:
 - a) 如果 DONE 管脚处于拉高状态, 可以正常下载且芯片正常工作;
 - b) 如果 DONE 管脚处于拉低状态, 可以正常下载, 下载完后 DONE 管脚需要拉高且同时保持 TCK 连接脉冲信号芯片才能 Wake Up。

- **Loading Rate:** AutoBoot 配置模式和 MSPI 配置模式下，码流文件从 Flash 到 SRAM 的加载速度。
GW1N-4/GW1N-4B/GW1N-4D/GW1NRF-4B/GW1NR-4/GW1NR-4B/GW1NR-4D 默认为 2.100MHz。
GW1NS-4/GW1NS-4C/GW1NSR-4/GW1NSR-4C/GW1NSER-4C 的速度等级为 C7/I6 的芯片型号默认为 2.6MHz，其他器件默认为 2.500MHz。
AutoBoot 配置模式和 MSPI 配置模式参考文档 [UG290, Gowin FPGA 产品编程配置手册](#)。不同器件 Loading Rate 的取值及其计算方式不同；
 - 以下器件封装的 Loading Rate 取值只支持 2.500MHz。
 - a) GW1N-2: LQFP100X/LQFP144X/MBGA132X/WLCSP42H/MBGA49
 - b) GW1N-2B: LQFP100X/LQFP144X/MBGA132X/MBGA121X
 - c) GW1N-2C: LQFP100X/LQFP144X/MBGA132X/MBGA121X/MBGA49/QFN32X
 - d) GW1NR-2: MBGA49P/MBGA49PG/MBGA49G
 - e) GW1NR-2B: MBGA49P/MBGA49PG/MBGA49G
 - f) GW1NR-2C: MBGA49P/MBGA49PG/MBGA49G
 - g) GW1N-1P5: LQFP100X
 - h) GW1N-1P5B: LQFP100X/QFN48X
 - i) GW1N-1P5C: LQFP100X/QFN48X
 - 以下器件的 Loading Rate 取值及计算方式如表 4-3 所示。
 - a) GW1NZ-1/GW1NZ-1C
 - b) GW1N-2/GW1N-2B/GW1N-2C/GW1N-1P5/GW1N-1P5B/GW1N-1P5C/GW1NR-2/GW1NR-2B/GW1NR-2C 除上述仅支持 2.500MHz 封装外的其他封装
 - c) GW1NSER-4C/GW1NS-4/GW1NSR-4/GW1NS-4C/GW1NSR-4C 速度等级为非 C7/I6 的芯片型号
 - d) GW1N-9/GW1N-9C/GW1NR-9/GW1NR-9C
 - e) GW2A-18/GW2A-18C/GW2AR-18/GW2AR-18C/GW2ANR-18C
 - f) GW2A-55/GW2A-55C/GW2AN-55C

表 4-3 Loading Rate 值及计算方式（一）

| Loading Rate (MHz) | 分数表示法 |
|--------------------|-----------|
| 2.500 (default) | 250 / 100 |
| 5.435 | 250 / 46 |
| 5.682 | 250 / 44 |
| 5.952 | 250 / 42 |
| 6.250 | 250 / 40 |

| Loading Rate (MHz) | 分数表示法 |
|--------------------|----------|
| 6.579 | 250 / 38 |
| 6.944 | 250 / 36 |
| 7.353 | 250 / 34 |
| 7.812 | 250 / 32 |
| 8.333 | 250 / 30 |
| 8.929 | 250 / 28 |
| 9.615 | 250 / 26 |
| 10.417 | 250 / 24 |
| 11.364 | 250 / 22 |
| 12.500 | 250 / 20 |
| 13.889 | 250 / 18 |
| 15.625 | 250 / 16 |
| 17.857 | 250 / 14 |
| 20.833 | 250 / 12 |
| 25.000 | 250 / 10 |
| 31.250 | 250 / 8 |
| 41.667 | 250 / 6 |
| 62.500 | 250 / 4 |
| 125.000 | 250 / 2 |

– 以下器件的 Loading Rate 取值及计算方式如表 4-4 所示。

- a) GW1N-1/GW1N-1S/GW1NR-1
- b) GW1NS-2/GW1NSR-2/GW1NS-2C/GW1NSR-2C/
GW1NSE-2C

表 4-4 Loading Rate 值及计算方式（二）

| Loading Rate(MHz) | 分数表示法 |
|-------------------|----------|
| 2.500 (default) | 240 / 96 |
| 2.553 | 240 / 94 |
| 2.609 | 240 / 92 |
| 2.667 | 240 / 90 |
| 2.727 | 240 / 88 |
| 2.791 | 240 / 86 |
| 2.857 | 240 / 84 |
| 2.927 | 240 / 82 |
| 3.000 | 240 / 80 |
| 3.077 | 240 / 78 |
| 3.158 | 240 / 76 |
| 3.243 | 240 / 74 |

| Loading Rate(MHz) | 分数表示法 |
|-------------------|----------|
| 3.333 | 240 / 72 |
| 3.429 | 240 / 70 |
| 3.529 | 240 / 68 |
| 3.636 | 240 / 66 |
| 3.750 | 240 / 64 |
| 3.871 | 240 / 62 |
| 4.000 | 240 / 60 |
| 4.138 | 240 / 58 |
| 4.286 | 240 / 56 |
| 4.444 | 240 / 54 |
| 4.615 | 240 / 52 |
| 4.800 | 240 / 50 |
| 5.000 | 240 / 48 |
| 5.217 | 240 / 46 |
| 5.455 | 240 / 44 |
| 5.714 | 240 / 42 |
| 6.000 | 240 / 40 |
| 6.316 | 240 / 38 |
| 6.667 | 240 / 36 |
| 7.059 | 240 / 34 |
| 7.500 | 240 / 32 |
| 8.000 | 240 / 30 |
| 8.571 | 240 / 28 |
| 9.231 | 240 / 26 |
| 10.000 | 240 / 24 |
| 10.909 | 240 / 22 |
| 12.000 | 240 / 20 |
| 13.333 | 240 / 18 |
| 15.000 | 240 / 16 |
| 17.143 | 240 / 14 |
| 20.000 | 240 / 12 |
| 24.000 | 240 / 10 |
| 30.000 | 240 / 8 |
| 40.000 | 240 / 6 |
| 60.000 | 240 / 4 |
| 120.000 | 240 / 2 |

- 以下器件的 Loading Rate 取值及计算方式如表 4-5 所示。

- a) GW2AN-9X
b) GW2AN-18X

表 4-5 Loading Rate 值及计算方式 (三)

| Loading Rate(MHz) | 分数表示法 |
|-------------------|-----------|
| 2.500 (default) | 200 / 80 |
| 1.562 | 200 / 128 |
| 1.587 | 200 / 126 |
| 1.613 | 200 / 124 |
| 1.639 | 200 / 122 |
| 1.667 | 200 / 120 |
| 1.695 | 200 / 118 |
| 1.724 | 200 / 116 |
| 1.754 | 200 / 114 |
| 1.786 | 200 / 112 |
| 1.818 | 200 / 110 |
| 1.852 | 200 / 108 |
| 1.887 | 200 / 106 |
| 1.923 | 200 / 104 |
| 1.961 | 200 / 102 |
| 2.000 | 200 / 100 |
| 2.041 | 200 / 98 |
| 2.083 | 200 / 96 |
| 2.128 | 200 / 94 |
| 2.174 | 200 / 92 |
| 2.222 | 200 / 90 |
| 2.273 | 200 / 88 |
| 2.326 | 200 / 86 |
| 2.381 | 200 / 84 |
| 2.439 | 200 / 82 |
| 2.564 | 200 / 78 |
| 2.632 | 200 / 76 |
| 2.703 | 200 / 74 |
| 2.778 | 200 / 72 |
| 2.857 | 200 / 70 |
| 2.941 | 200 / 68 |
| 3.030 | 200 / 66 |
| 3.125 | 200 / 64 |
| 3.226 | 200 / 62 |
| 3.333 | 200 / 60 |

| Loading Rate(MHz) | 分数表示法 |
|-------------------|----------|
| 3.448 | 200 / 58 |
| 3.571 | 200 / 56 |
| 3.704 | 200 / 54 |
| 3.846 | 200 / 52 |
| 4.000 | 200 / 50 |
| 4.167 | 200 / 48 |
| 4.348 | 200 / 46 |
| 4.545 | 200 / 44 |
| 4.762 | 200 / 42 |
| 5.000 | 200 / 40 |
| 5.263 | 200 / 38 |
| 5.556 | 200 / 36 |
| 5.882 | 200 / 34 |
| 6.250 | 200 / 32 |
| 6.667 | 200 / 30 |
| 7.143 | 200 / 28 |
| 7.692 | 200 / 26 |
| 8.333 | 200 / 24 |
| 9.091 | 200 / 22 |
| 10.000 | 200 / 20 |
| 11.111 | 200 / 18 |
| 12.500 | 200 / 16 |
| 14.286 | 200 / 14 |
| 16.667 | 200 / 12 |
| 20.000 | 200 / 10 |
| 25.000 | 200 / 8 |
| 33.333 | 200 / 6 |
| 50.000 | 200 / 4 |
| 100.000 | 200 / 2 |

- 以下器件的 Loading Rate 取值及计算方式如表 4-6 所示。

a) GW1N-4/GW1N-4B/GW1N-4D/GW1NRF-4B/GW1NR-4/GW1NR-4B/GW1NR-4D

表 4-6 Loading Rate 值及计算方式（四）

| Loading Rate (MHz) | 分数表示法 |
|--------------------|-----------|
| 2.100 (default) | 210 / 100 |
| 4.565 | 210 / 46 |
| 4.773 | 210 / 44 |

| Loading Rate (MHz) | 分数表示法 |
|--------------------|----------|
| 5.000 | 210 / 42 |
| 5.250 | 210 / 40 |
| 5.526 | 210 / 38 |
| 5.833 | 210 / 36 |
| 6.176 | 210 / 34 |
| 6.563 | 210 / 32 |
| 7.000 | 210 / 30 |
| 7.500 | 210 / 28 |
| 8.077 | 210 / 26 |
| 8.750 | 210 / 24 |
| 9.545 | 210 / 22 |
| 10.500 | 210 / 20 |
| 11.667 | 210 / 18 |
| 13.125 | 210 / 16 |
| 15.000 | 210 / 14 |
| 17.500 | 210 / 12 |
| 21.000 | 210 / 10 |
| 26.250 | 210 / 8 |
| 35.000 | 210 / 6 |
| 52.500 | 210 / 4 |
| 105.000 | 210 / 2 |

– 以下器件的 Loading Rate 取值及计算方式如表 4-7 所示。

a) GW1NSER-4C/GW1NS-4/GW1NSR-4/GW1NS-4C/GW1NSR-4C 速度等级为 C7/I6 的芯片型号

表 4-7 Loading Rate 值及计算方式（五）

| Loading Rate (MHz) | 分数表示法 |
|--------------------|-----------|
| 2.600 | 260 / 100 |
| 5.652 | 260 / 46 |
| 5.909 | 260 / 44 |
| 6.190 | 260 / 42 |
| 6.500 | 260 / 40 |
| 6.842 | 260 / 38 |
| 7.222 | 260 / 36 |
| 7.647 | 260 / 34 |
| 8.125 | 260 / 32 |
| 8.667 | 260 / 30 |
| 9.286 | 260 / 28 |

| Loading Rate (MHz) | 分数表示法 |
|--------------------|----------|
| 10.000 | 260 / 26 |
| 10.833 | 260 / 24 |
| 11.818 | 260 / 22 |
| 13.000 | 260 / 20 |
| 14.444 | 260 / 18 |
| 16.250 | 260 / 16 |
| 18.571 | 250 / 14 |
| 21.667 | 260 / 12 |
| 26.000 | 260 / 10 |
| 32.500 | 260 / 8 |
| 43.333 | 260 / 6 |
| 65.000 | 260 / 4 |
| 130.000 | 260 / 2 |

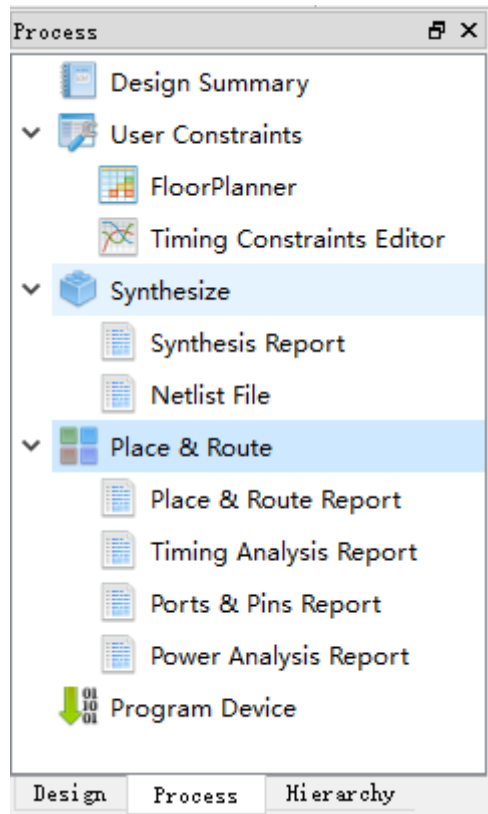
- **SPI Flash Address:** 指定 SPI Flash 地址。SPI Flash 地址是指下一次 multiboot 时，加载码流文件的起始地址，默认为 00000000。具体可参考 [SUG502, Gowin Programmer 用户指南](#);
- **User Code:** 用户可以自定义 User Code，定义的值会体现在产生的码流文件中，通过编程器下载码流文件时会对 User Code 进行校验。默认为 Default(00000000);
- **Bitstream Format:** 用于指定生成的码流文件内容的格式，可选项有 Text 和 Binary，默认为 Binary。当选择 Text 选项时，会生成纯文本格式的*.fs 文件；选择 Binary 选项时会生成*.fs、*.bin 和*.binx 格式的码流文件。*.bin 和*.binx 是二进制格式的码流文件，*.binx 文件含有头部注释信息，*.bin 没有头部注释信息。

4.4 管理工程过程

在过程管理区，列出了云源的过程管理，如图 4-27 所示。在该窗口中可进行以下操作：

- 查看 Design 概述；
- 启动物理约束编辑器；
- 启动时序约束编辑器；
- 执行综合；
- 查看综合设计报告；
- 执行布局布线；
- 查看 Place&Route 后生成的报告；
- 启动编程器等功能。

图 4-27 工程 Process 窗口



4.4.1 Design Summary

新建工程时，软件会分析工程并提供一份报告 Design Summary，报告中包括工程文件路径、综合工具的信息以及器件信息，如图 4-28 所示，有三种方法打开 Design Summary：

- 在 GOWIN FPGA Designer 菜单栏上，选择“Window > Design Summary”；
- 在 Process 窗口中，双击“Design Summary”；
- 在 Process 窗口中，右击“Design Summary”选择“Open”。

图 4-28 工程信息显示

| General | |
|-----------------|---|
| Project File: | D:\user-bak\Users\root\Desktop\8bit_counter\8bit_counter.gprj |
| Synthesis Tool: | GowinSynthesis |

| Target Device | |
|---------------|-----------------------|
| Part Number: | GW2AN-LV18XUG400C8/I7 |
| Series: | GW2AN |
| Device: | GW2AN-18X |
| Package: | UBGA400 |
| Speed Grade: | C8/I7 |
| Core Voltage: | LV |

4.4.2 User Constraints

User Constraints 提供了快速打开和创建约束文件的方法。User Constraints 分为物理约束和时序约束。

有关约束编辑器的详细使用方法，请参考 [SUG940, Gowin 设计时序约束指南](#)；[SUG935, Gowin 设计物理约束指南](#)。

4.4.3 Synthesize

GowinSynthesis[®]是高云研发的综合软件，支持高云半导体的库文件及其实现，目前 Verilog 语言支持 System Verilog 2017、Verilog 2001 和 Verilog 95，VHDL 语言支持 VHDL1993 和 VHDL 2008。

通过在工程管理区的 Synthesize 处右击选择“Configuration”，打开综合选项对话框，选择综合工具，如图 4-20 所示。


Synthesize 提供了运行综合、设置综合选项参数及管理网表文件 (Netlist File) 和综合报告 (Synthesis Report) 的功能，综合报告的介绍请参考 6.1 综合报告。

参考以下步骤运行 Synthesize:

1. 配置 Synthesis 选项, 有关 Synthesis 选项, 请参考 4.3.3 编辑工程配置;
2. 运行 Synthesize;
3. 在过程管理区, 双击“Synthesize”或右击“Synthesize > Run”, 启动综合工具对源文件进行综合。若综合成功, 则 Synthesize 栏前会出现图标“

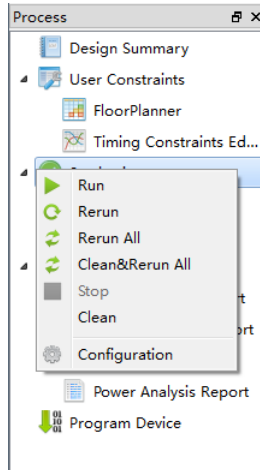
若综合之前 (Synthesize 图标为“

右击“Synthesize”, 如图 4-29 可进行的操作包含:

- Run: 只有 Synthesize 栏前图标为初始状态“SUG100-2.7

- Stop: 停止运行 Synthesize 过程;
- Clean: 清除综合后产生的文件夹 (GowinSynthesis 产生的文件夹为 gwsynthesis), 单击该选项会弹出提示框;
- Configuration: 可对 Synthesize 中参数进行设置。

图 4-29 右击 Synthesize



4.4.4 Place & Route

Place & Route 提供运行布局布线、设置布局布线参数及管理布局布线后生成文件的功能。

注!

Place & Route 依赖于综合过程, 执行该步骤时, 如其依赖项 (Synthesize) 未执行, 则会先执行 Synthesize, 再执行该步骤。

参考以下步骤运行 Place & Route:




1. 配置 Place & Route 选项, 关于 Place & Route 选项的介绍请参考 4.3.3 编辑工程配置;
2. 运行 Place & Route, 双击“Place & Route”或右击 Place & Route 选择“Run”执行布局布线产生码流文件和相关报告文件。如运行成功, 则 Place & Route 栏前会出现图标“✔”, 否则出现图标“!";
3. Place & Route 运行成功后, 在 Place & Route 下方双击文件或右击选择“Open”可在文本编辑区浏览报告文件;
4. 可查看生成的四种报告文件, 包括布局布线报告 (Place & Route Report)、时序分析报告 (Timing Analysis Report)、端口属性报告 (Ports & Pins Report) 及功耗分析报告 (Power Analysis Report), 这四种文件均不可编辑。具体请参考 6.2 布局布线报告、6.3 端口属性报告、6.4 时序报告、6.5 功耗分析报告。

注!

- 若当前已经打开报告文件, 再运行 Place & Route 重新生成报告文件后会提示是否更新文件;

- 若运行 Place & Route 之前（Place & Route 栏前图标为“”），双击报告文件或在报告文件右击选择“Open”，则会先运行 Place & Route，运行成功后打开报告文件。

右击 Place & Route，可进行的操作包含：

- Run: 只有 Place & Route 栏前图标为初始状态“”或失败状态“”或过期状态“”时，选择该选项会运行 Place & Route；
- Rerun: 无论 Place & Route 是何种状态，选择该选项，重新运行 Place & Route；
- Rerun All: 无论 Synthesize 和 Place & Route 是何种状态，选择该选项，都会重新对源文件进行 Synthesize 及 Place & Route；
- Clean& Rerun All: 清除工程文件夹 impl 下的 gwsynthesis 和 pnr 文件夹并重新执行 Synthesize 及 Place & Route；
- Stop: 停止运行 Place & Route 过程；
- Clean: 清除运行 Place & Route 后产生的文件夹（pnr），单击该选项会弹出是否要删除文件夹的提示框。如果删除文件夹失败会报出 Warning 提示；
- Configuration: 可对 Place & Route 参数进行设置。

4.4.5 Program Device

云源在布局布线运行成功后，会生成码流文件，需启动编程器才可将码流文件下载到芯片，实现用户所需的功能。

注！

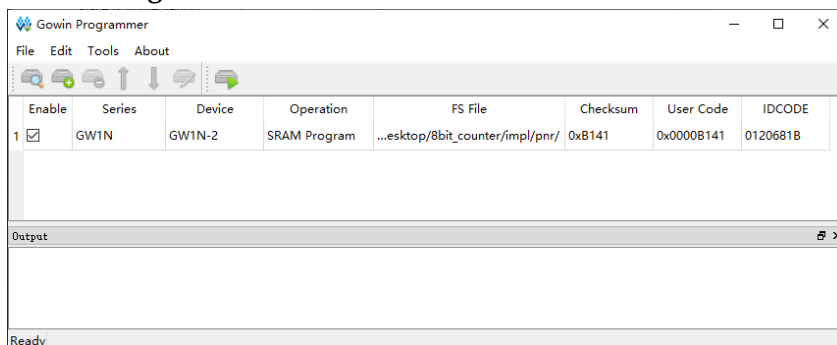
Program Device 依赖于 Synthesize 及 Place & Route 步骤，执行该步骤时，如其依赖项（Synthesize 及 Place & Route）未执行，会弹出警告提示。

双击“Program Device”或在该项右击“Run”选项，打开编程器，如图 4-30 所示。

注！

Linux 安装包中的编程器不适用于 Linux 版本 Red Hat 5.10，只支持 Red Hat 6 以上的版本，且 Linux 内核版本需要在 2.18 及以上。

图 4-30 Programmer 界面



有关编程器的详细使用方法，请参考 [SUG502, Gowin Programmer 用户指南](#)。

4.5 工程存档及恢复

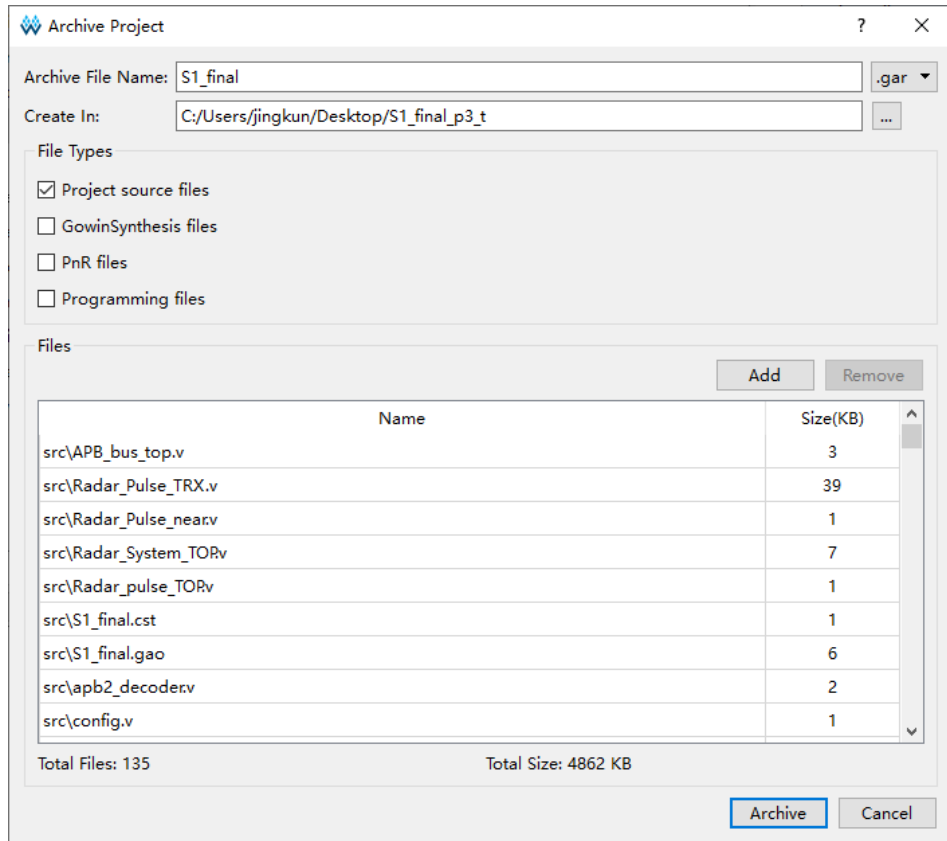
云源支持对当前工程进行存档以及恢复被存档的工程。通过菜单栏 Project 中的“Archive Project”和“Restore Archived Project”对工程进行存档或恢复。

4.5.1 工程存档

单击菜单栏 Project 中的“Archive Project”会弹出存档工程对话框，如图 4-31 所示，将鼠标悬浮在选项处时会显示其解释。

- **Archive File Name** 是存档后的文件名，默认和当前要存档的工程名一致，扩展名是.gar;
- **Create In** 为存档后的文件要存放的路径，默认为当前工程路径;
- 存档工程项包括 **Project source files**(默认勾选)、**GowinSynthesis files**、**PnR files**、**Programming files**;
 - **Project source files**: 包含工程所在路径/src 下的所有文件;
 - **GowinSynthesis files**: 包含工程所在路径/impl/gwsynthesis 下运行综合产生的工程文件 (*.prj)、网表文件 (*.vg)、综合报告 (*_syn.rpt.html)、资源统计文件 (*_syn_rsc.xml);
 - **PnR files**: 包含工程所在路径/impl/pnr 下运行布局布线产生的文件;
 - **Programming files**: 包含工程所在路径/impl/pnr 下运行布局布线产生的码流文件*.fs 、*.bin 和*.binx。
- 当勾选了某一存档工程项后，下方会显示该项在当前工程下的源文件、所在路径以及大小;
- **Add** 和 **Remove** 可以用来自行增删存档的文件;
- 单击 **Archive** 后，如果工程中的文件未保存，则会弹出警告提示框，提示先保存文件;
- 存档完成后会弹出提示框，提示存档成功或失败;
- 存档完成后在 **Create In** 路径下会生成两个文件: 存档工程*.gar 和与 gar 同名的存档文件*.garlog。扩展名为.gar 的文件压缩存储所有被存档的文件，日志文件*.garlog 供参考使用，可以查看哪些文件被存档以及存档是否成功。

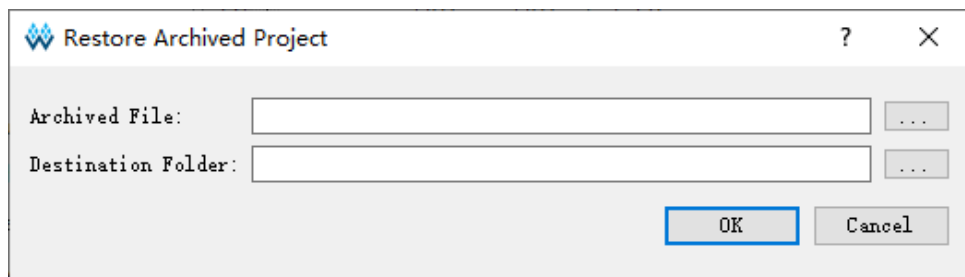
图 4-31 工程存档对话框



4.5.2 恢复存档工程

单击菜单栏 Project 中的“Restore Archived Project”会弹出恢复存档工程对话框，如图 4-32 所示。

图 4-32 恢复存档工程对话框

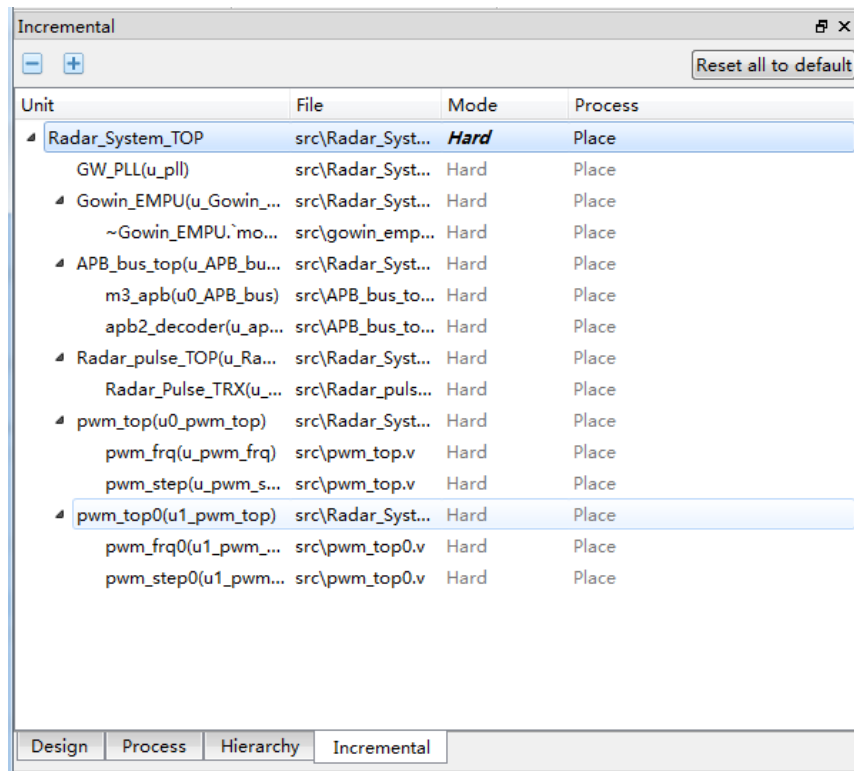


单击“Archived File”右面的路径选择按钮，选择要恢复的存档文件。选择要恢复的存档文件后，“Destination Folder”会自动更新为存档文件所在的路径。单击“OK”后会弹出恢复成功的对话框。

4.6 增量编译

云源的增量编译功能目前只支持对上一次布局的结果全部保留，提高运行效率。通过菜单栏 Project 中的“Set Incremental”打开增量编译配置窗口，如图 4-33 所示。

图 4-33 增量编译配置窗口



在 Incremental 窗口中，Unit 列显示设计文件的 module 层级结构，File 列显示 module 定义所在的文件，Mode 列可以配置增量编译模式，Process 列指定增量编译的有效流程。通过左上角的“-”和“+”可以对设计中所有的层级进行折叠或展开。单击右上角的“Reset all to default”后，当前配置页面全部恢复成默认设置。

在 Mode 列中可以通过双击对增量编译模式进行配置，目前支持如下三种配置：

- **Unset**: 用于还原未指定的增量编译状态；
- **Ignore**: 忽略上一次的综合、布局和布线结果，重新执行综合、布局和布线；
- **Hard**: 全部使用上一次的布局结果。


Process 列的取值目前只支持 Place，即综合和布局结果保持不变，只增量编译布线。

增量编译模式 Mode 默认显示为 Ignore。当 Mode 选择 Unset 或 Ignore 时，Process 显示为空。当 Mode 选择 Hard 时，Process 显示为 Place。目前仅支持对 top module 进行 Mode 的设置，sub module 的相应取值会跟随 top module 的配置进行同步更新。

增量编译模式选择 Hard 后会在工程路径/impl 下生成配置文件 incremental.cfg，文件内容为“top module name,hard,place”。

4.7 退出软件

通过以下方式退出云源：

1. 单击“**File > Exit**”选项；
2. 单击软件界面右上角图标“”。

注！

- 若有未保存的文件，则会先提示是否对文件进行保存；
- 软件提供的保存（**Save**）、保存所有（**Save All**）和另存为（**Save As...**）功能只针对文本编辑动作的保存；
- 软件对于工程配置（**Configuration**）信息的修改或对工程中文件的增删等操作不会即时保存到工程配置文件中，在软件关闭时自动保存；
- 如果软件在运行中，无法通过单击退出软件。

5 云源集成工具

5.1 物理约束编辑器

FloorPlanner 是高云半导体面向市场自主研发的物理约束编辑器，支持对 I/O、Primitive（原语）、Block（BSRAM、DSP）、Group 等的属性及位置信息的读取与编辑功能，同时可根据用户的配置生成新的布局与约束文件，文件中规定了 I/O 的属性信息，原语、模块的位置信息等。FloorPlanner 提供了快捷的布局与约束编辑功能，有效地提高编写物理约束文件的效率，同时可以根据器件布局和时序路径进行时序优化，可支持高云半导体的各款 FPGA 器件产品。

启动 FloorPlanner 有两种方式：

1. 未建立 FGPA 工程时，可直接在软件菜单栏的“Tools”下拉列表中选择“FloorPlanner”，此时需通过“File > New...”加载网表及所需器件信息；
2. 建立 FPGA 工程时，直接在过程管理区运行 Synthesize 后双击“FloorPlanner”，此时，FloorPlanner 会直接加载工程文件并显示在 FloorPlanner 界面。FloorPlanner 分为 Summary、Netlist、Chip Array、Package View 以及各项约束窗口，如图 5-1 和图 5-2 所示。

注！

- 该工具的详细使用方法请参考 [SUG935, Gowin 设计物理约束指南](#)；
- 除此之外 FloorPlanner 还可以进行时序优化。

图 5-1 Chip Array 窗口

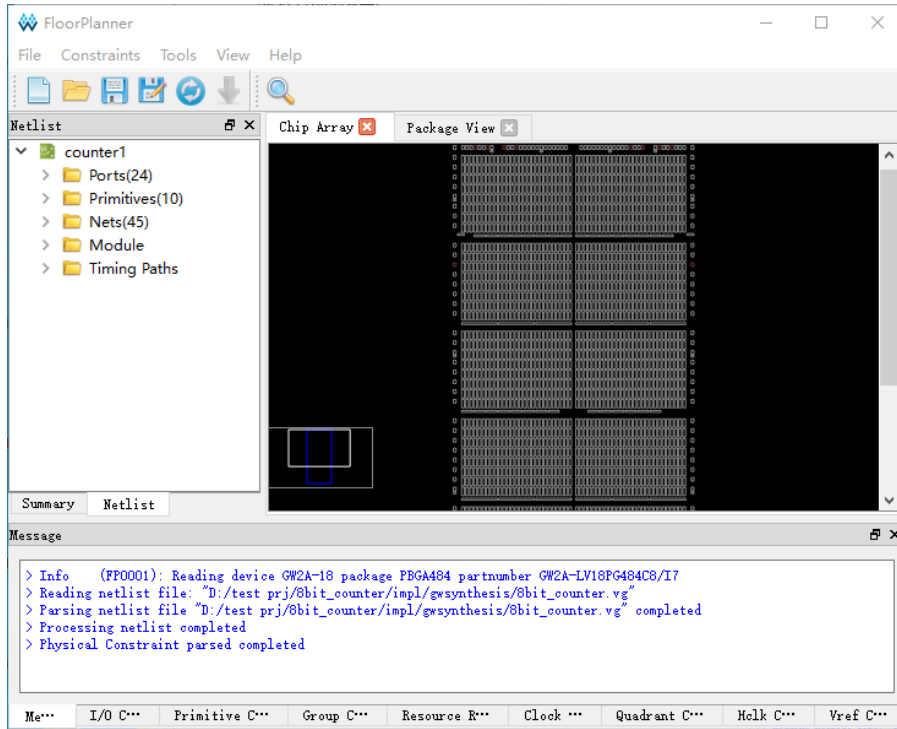
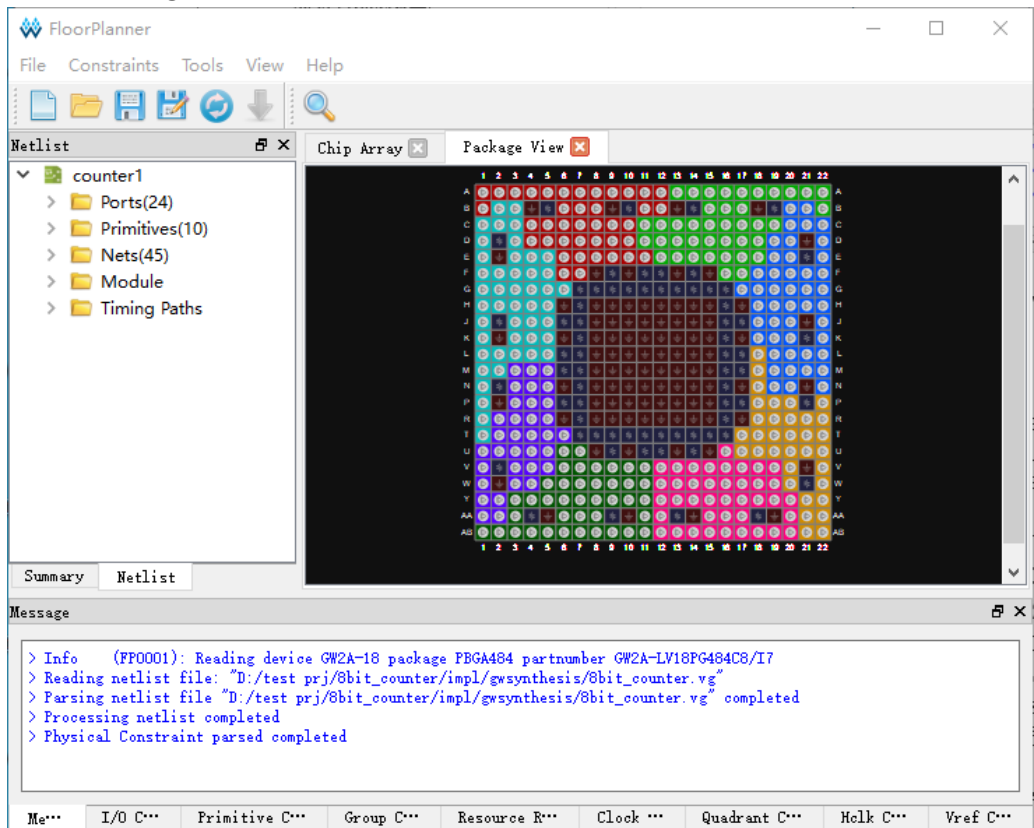


图 5-2 Package View 窗口



5.2 时序约束编辑器

时序约束编辑器（Timing Constraints Editor）是高云半导体面向市场自主研发的时序约束编辑工具，支持多种时序约束命令的编辑，包括时钟约束、输入输出约束、路径约束和时钟报告等约束编辑。Timing Constraints Editor 提供了简单快捷的时序约束编辑功能，可支持高云半导体的各 FPGA 器件产品。

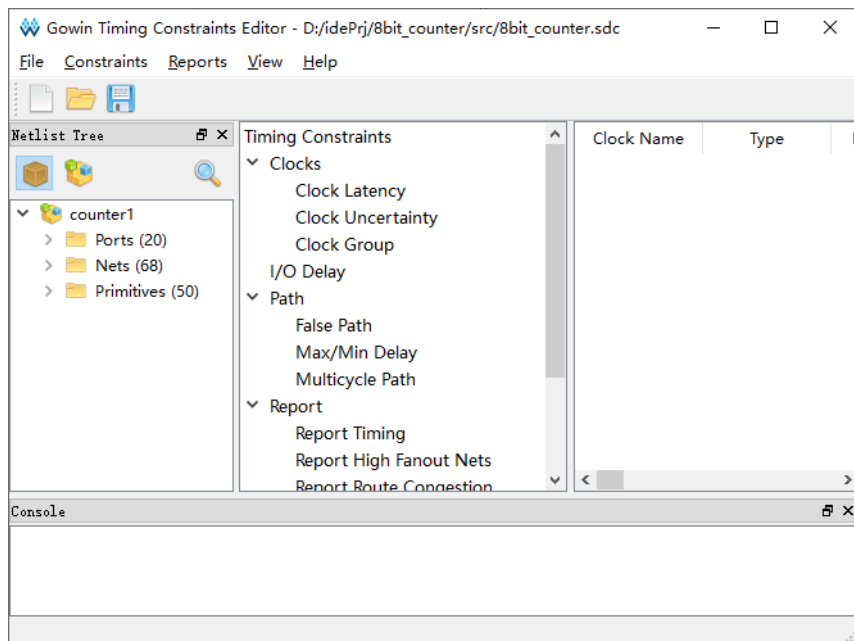
启动 Timing Constraints Editor 的方式有两种：

1. 如未建立 FPGA 工程，在菜单栏中，选择“Tools > Timing Constraints Editor”，此时，需要通过“File > New...”加载网表文件；
2. 如已建立 FPGA 工程，在过程管理区运行 Synthesize 后，双击“Timing Constraints Editor”，则 Timing Constraints Editor 会直接加载工程文件并显示在 Gowin Timing Constraints Editor 界面，如图 5-3 所示。

注！

有关时序约束编辑器的详细使用方法，请参考 [SUG940, Gowin 设计时序约束指南](#)。

图 5-3 创建时序约束界面



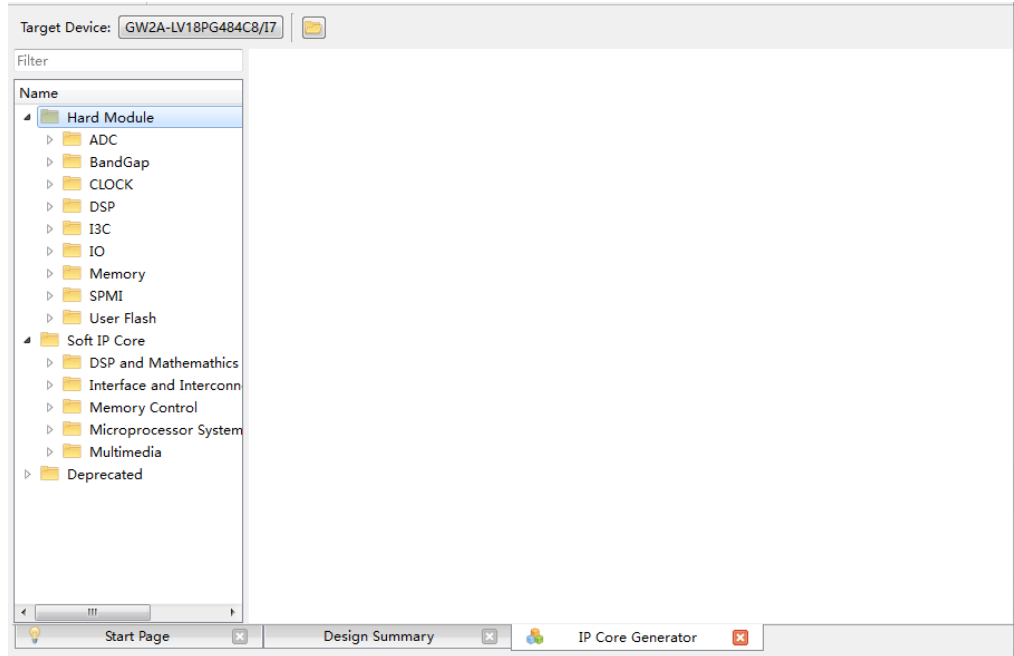
5.3 IP Core 产生器

IP Core Generator 是基于高云 FPGA 的 IP 产生工具，包括硬核和软核两部分，通过工具产生实例化的设计之后，用户可调用该实例化模块实现设计所需的功能，帮助用户快速实现复杂设计。如图 5-4 所示，主要功能有：

- 支持 Soft IP core、Hard module 的信息预览；
- 支持 Soft IP core、Hard module 定制生成；
- 支持 Hard module 实例化示例案例生成；
- 支持自动保存用户配置；

- 支持 IP 生成代码语言选择；
- 部分 Soft IP 支持自动产生激励文件；
- 支持器件信息自动过滤显示可用 IP。

图 5-4 IP Core Generator 窗口



通过单击菜单栏中 Tools 下拉列表中“IP Core Generator”启动 IP Core Generator 工具，进行 IP 调用。相关参考文档如下：

- 模数转换器（ADC）的产生参考 [SUG283, Gowin 原语用户指南](#)；
- BANDGAP 的产生参考 [SUG283, Gowin 原语用户指南](#)；
- 时钟资源（CLOCK）的产生参考 [UG286, Gowin 时钟资源（CLOCK）用户指南](#)；
- 数字信号处理器（DSP）的产生参考 [UG287, Gowin 数字信号处理器（DSP）用户指南](#)；
- I3C 的产生参考 [SUG283, Gowin 原语用户指南](#)；
- 输入输出逻辑（IO Logic）的产生参考 [UG289, Gowin 可编程通用管脚（GPIO）用户指南](#)；
- 存储器（BSRAM & SSRAM）的产生参考 [UG285, Gowin 存储器（BSRAM & SSRAM）用户指南](#)；
- 闪存资源（User Flash）的产生参考 [UG295, Gowin 闪存资源（User Flash）用户指南](#)；
- 双线串行接口（SPMI）的产生参考 [SUG283, Gowin 原语用户指南](#)；
- 软核相关的文档请参考官网的 IP 参考设计：
<http://www.gowinsemi.com.cn/enrollment.aspx?FId=n27:27:4>。

注!

置灰的 Hard Module 或者 Soft IP Core 为当前 device 所不支持的。

5.4 在线逻辑分析仪

高云在线逻辑分析仪（GAO）是高云半导体自主研发的一款数字信号分析工具，旨在帮助用户更加简便地分析设计中信号之间的时序关系，快速进行系统分析和故障定位，提高设计效率。

GAO 支持 RTL 级信号、综合后网表级信号捕获的标准版（Standard）和精简版（Lite）两个版本。标准版 GAO 最多可以支持 16 个功能内核，每个内核可配置一个或多个触发端口，支持多级静态或动态触发表达式。精简版 GAO 配置简便，无需设置触发条件，精简版 GAO 还可以捕获信号的初始值，方便用户分析上电瞬间的工作状态。信号捕获后可以将其波形导出，支持 *.csv、*.vcd 和 *.prn 三种导出文件格式，*.csv 和 *.prn 两种类型的文件可直接用于 Matlab 等第三方仿真工具使用，*.vcd 类型文件可用 ModelSim 工具使用。

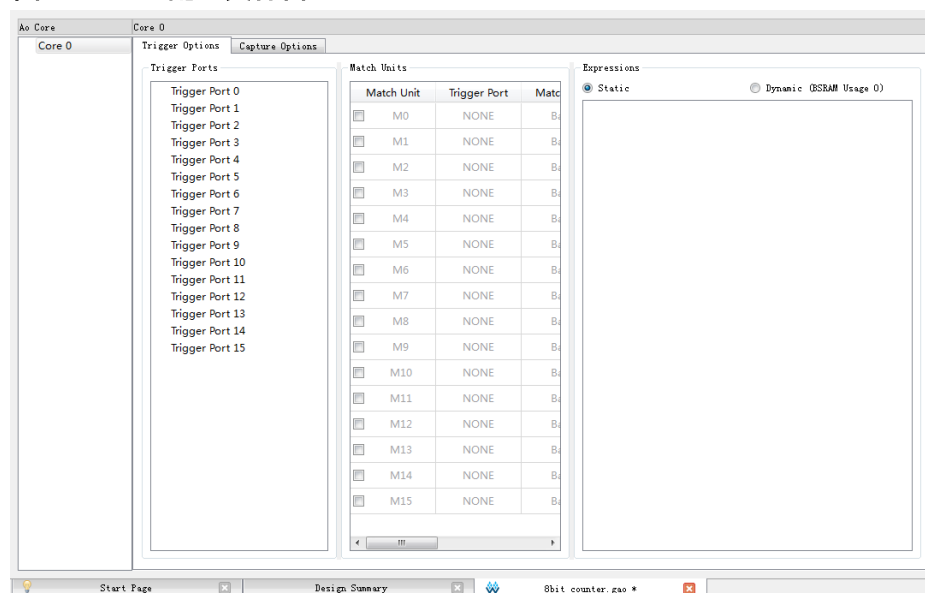
注!

Matlab、ModelSim 工具的使用需要取得第三方授权。

GAO 包括 Gowin GAO 配置和 Gowin Analyzer Oscilloscope 两个工具。Gowin GAO 配置主要用于把定位信息配置到设计中，这些定位信息主要基于采样时钟、触发单元和触发表达式；Gowin Analyzer Oscilloscope 通过 JTAG 接口连接软件和目标硬件，将 GAO 配置文件设置的采样信号的数据直观地通过波形显示出来。

启动 GAO 配置文件窗口之前，需在工程管理区新建 GAO 配置文件，打开配置文件窗口，以下是以标准版 GAO 为例，如图 5-5 所示。

图 5-5 GAO 配置文件窗口

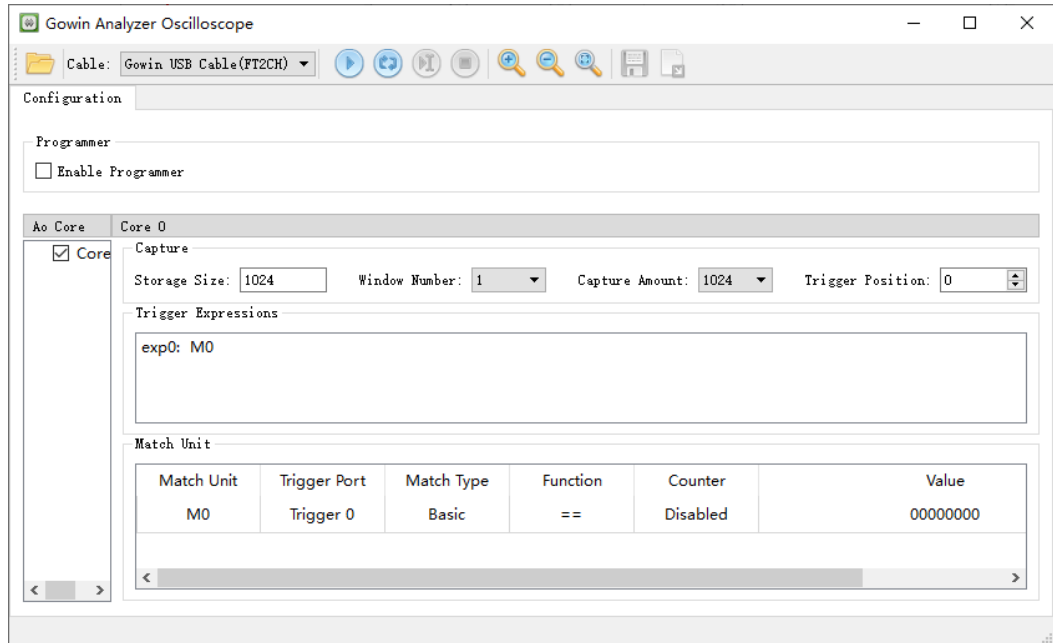


配置文件建立之后，在菜单栏中，选择“Tools > Gowin Analyzer Oscilloscope”，打开在线逻辑分析仪工具，如图 5-6 所示。

注！

在线逻辑分析仪的配置和使用方法请参考 [SUG114, Gowin 在线逻辑分析仪用户指南](#)。

图 5-6 GAO 界面



5.5 功耗分析工具

高云功耗分析工具（GPA）为用户提供尽可能准确的功耗分析，提供了丰富的用户设置选项。用户可根据设计工程的实际情况，设置影响功耗的芯片型号、工作环境以及信号翻转率等参数。功耗分析工具根据用户设置的参数，自动估算用户设计的功耗，产生功耗分析报告。

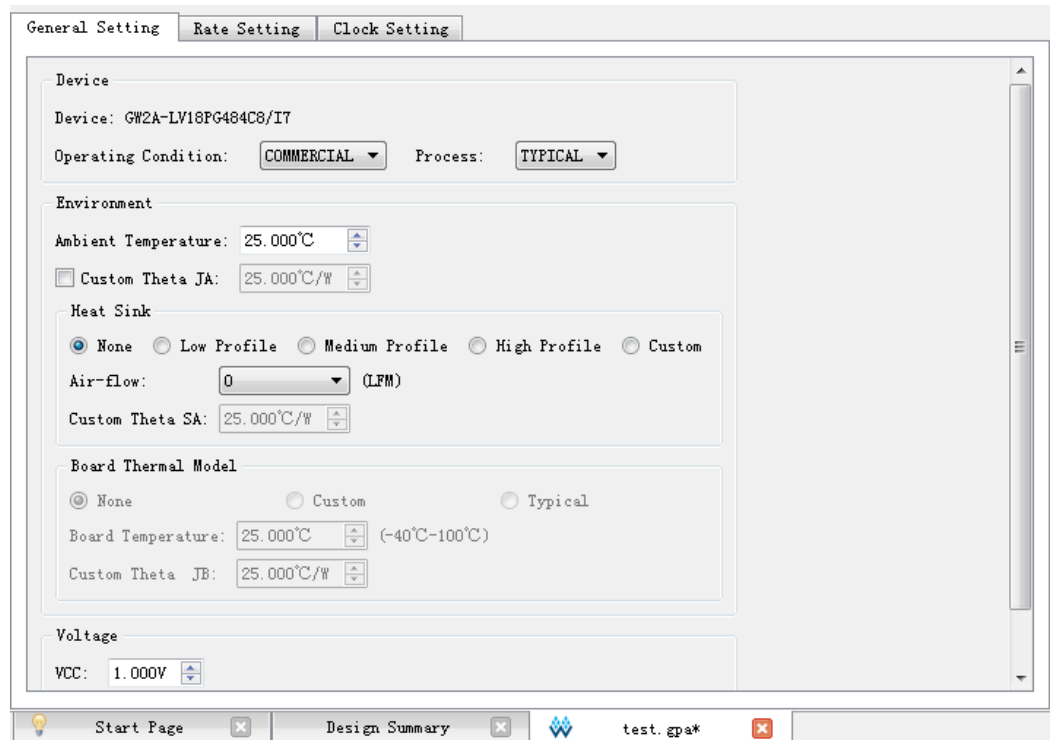
目前，GPA 工具的启动首先是以新建配置文件（.gpa）为基础，步骤如下所示：

1. 在软件工程管理区（Design），单击“File > New...”，打开“New”对话框；
2. 选择“GPA Config File”，在弹出的对话框中填写“Name”；
3. 单击“OK”，即可在“Design”窗口看到新建的 GPA Config File；
4. 双击文件名，在源文件编辑区对 GPA 配置文件进行配置，如图 5-7 所示。

注！

有关高云功耗分析工具的配置和使用方法，请参考 [SUG282, Gowin 功耗分析工具用户指南](#)。

图 5-7 GPA 配置文件窗口



5.6 块存储器初始化文件编辑器

块存储器初始化文件是一个 ASCII 文件，其扩展名为.mi；用户可根据自身设计要求，生成相应格式的初始化文件，用以指定存储器中每个地址下的初始值。如果已有.mi 文件，可以在 IDE 中用块存储器初始化文件编辑器打开该.mi 文件，可以再次编辑后进行保存。

块存储器初始化文件的文件名为*.mi (file_name.mi)，文件中每一行代表一个存储单元，行数即为存储单元的个数，也代表存储器的地址深度 Address Depth；列数代表每个存储单元有多少位，即内存的数据宽度 Data Width。地址从上到下依次递增，每行数据高位在前，低位在后。

高云块存储器初始化文件的编辑是以新建配置文件 (.mi) 为基础，具体格式请参考 [UG285, Gowin 存储器\(BSRAM & SSRAM\)用户指南](#)。初始化文件编辑器具体使用步骤如下：

1. 在软件工程管理区 (Design)，单击 “File > New...”，打开 “New” 对话框；
2. 选择 “Memory Initialization File”，如图 5-8 所示，单击 “OK”，在弹出的 New File 对话框中填写初始化文件名字后单击“OK”，如图 5-9 所示；
3. 启动如图 5-10 所示的初始化文件配置窗口，窗口左侧部分表格填写初始值，右侧部分配置初始化文件大小和视图格式；
4. 在窗口的右侧配置初始化文件的 Depth 和 Width，以及左侧表格中地址和初始值的数值显示格式；

- Depth 和 Width 要与用户在 IP Core Generator 窗口上所选择 Block Memory 或 Shadow Memory 的 Address Depth 及 Data Width 一致，若初始化文件中 Address Depth 或 Data Width 大于窗口上所选择的值，IP Core Generator 将会提示错误信息；若 Address Depth 或 Data Width 小于各自窗口上所选择的值，则未指定的地址下的值默认初始化为 0，设置完单击“Update”；
 - 左侧表格中地址和数值的显示格式可以选择二进制，十六进制，带地址十六进制等格式。
5. 在配置窗口的左侧表格中进行初始值的写入，此外在左侧表格中可以对表格的视图格式进行设置；
- 通过表头右击可以配置列数的显示，有 1、8、16 三种选择，如图 5-11 所示；
 - 表格中的初始值既可以通过双击后手动写入，也可以通过右击进行设置，在要输入数值处右击，选择“Fill with 0”是指初始值每位都为 0，“Fill with 1”是指初始值每位都为 1，“Custom Fill”使用户可以根据需要进行数值写入，同样也可批量设置初始值如图 5-12 所示。
6. 保存文件。

图 5-8 初始化文件新建 New 对话框

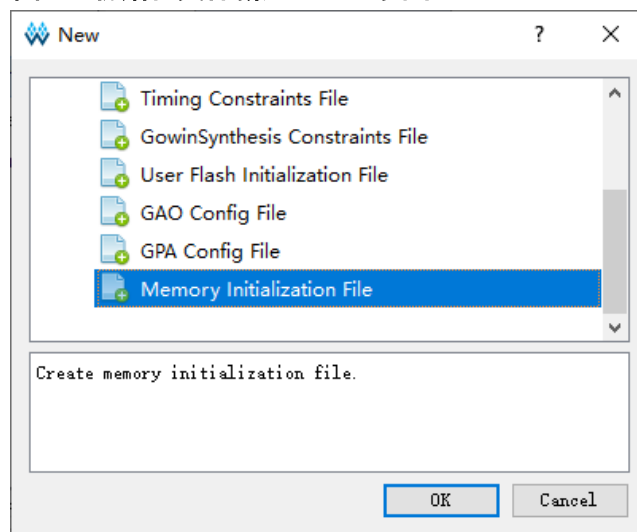


图 5-9 初始化文件新建 New File 对话框

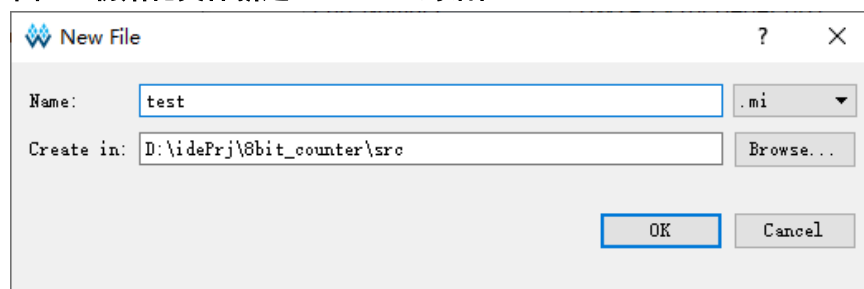


图 5-10 初始化文件配置窗口

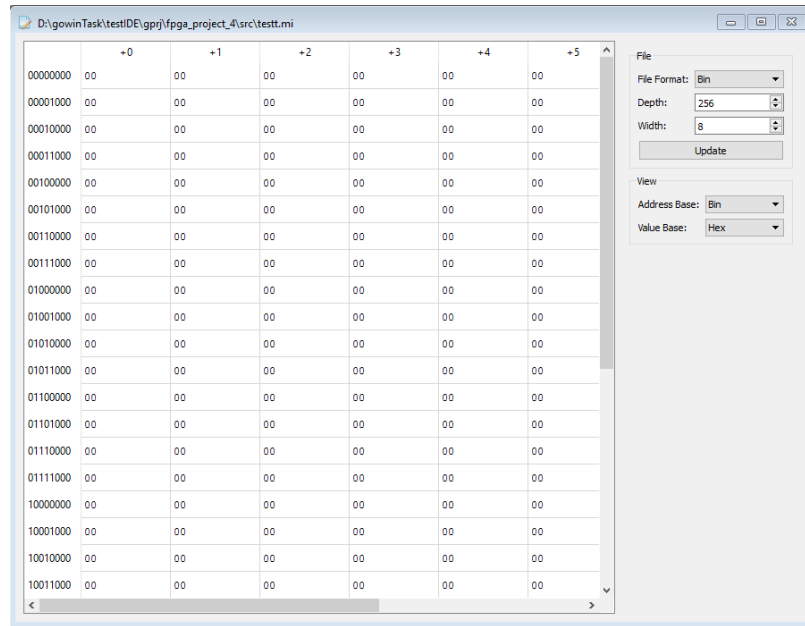


图 5-11 列数配置

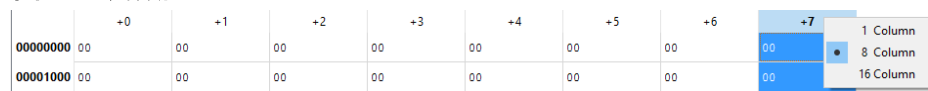
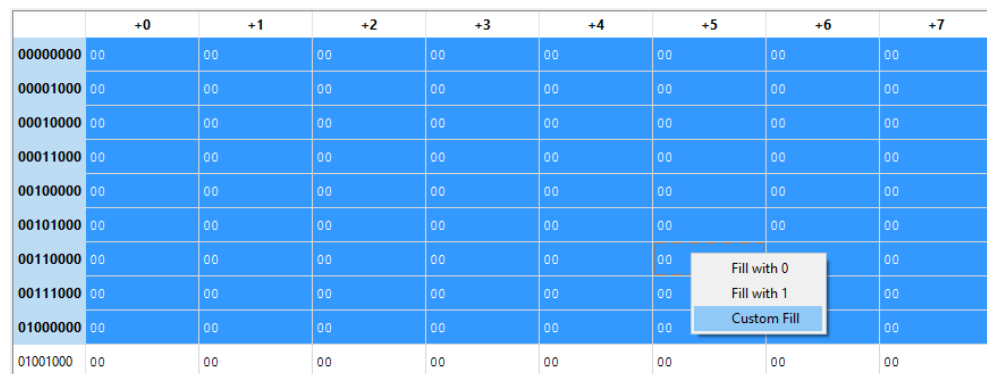


图 5-12 批量设置



5.7 User Flash 初始化文件编辑器

User Flash 初始化文件是一个 ASCII 文件，其扩展名为 .fi；用户可根据自身设计要求，生成相应格式的初始化文件，用以指定 User Flash 中每个地址下的初始值。如果已有 .fi 文件，可以在软件中用 User Flash 初始化文件编辑器打开该 fi 文件，可以再次编辑后进行保存。

User Flash 初始化文件的文件名为 *.fi(file_name.fi)，文件中每一行代表一个存储单元，行数即为需要初始化数据的存储单元的个数；行头中括号内容分别表示纵坐标地址、横坐标地址，使用分号隔开；每行中括号后的内容表示存储单元初始化的数据，数据支持二进制和十六进制，数据高位在前，低位在后。 .fi 文件内容格式举例如下：

5.7.1 二进制格式 (Bin File)

Bin 文件是存储数据由二进制数 0 和 1 组成的文本文件。

```
//Copyright (C)2014-2021 Gowin Semiconductor Corporation.
//All rights reserved.
//File Title: User Flash Initialization File
//GOWIN Version: V1.9.8.01
//Part Number: GW1N-LV4LQ144C6/I5
//Device-package: GW1N-4-LQFP144
//Flash Type: FLASH256K
//File Format: Bin
//Created Time: 2021-10-11 17:24:22
[0:0] 0000000100010000000010000000000000
[2:1] 00000010011101000000001001110100
[4:2] 0001000100000000000010000000000001
```

5.7.2 十六进制格式 (Hex File)

Hex 文件与 Bin 文件格式类似，存储数据由十六进制数 0~F 组成。

```
//Copyright (C)2014-2021 Gowin Semiconductor Corporation.
//All rights reserved.
//File Title: User Flash Initialization File
//GOWIN Version: V1.9.8.01
//Part Number: GW1N-LV4LQ144C6/I5
//Device-package: GW1N-4-LQFP144
//Flash Type: FLASH256K
//File Format: Hex
//Created Time: 2021-10-12 08:48:03
[0:0] 01101000
[2:1] 02740274
[4:2] 11001001
[5:1] 564a2bc3
[8:1] eadbe012
```

高云 User Flash 初始化文件的编辑是以新建配置文件 (.fi) 为基础，初始化文件编辑器具体使用步骤如下：

1. 在软件工程管理区 (Design)，单击 “File > New...”，打开 “New” 对话框；

2. 选择“User Flash Initialization File”，如图 5-13 所示单击“OK”，在弹出的 New File 对话框中填写初始化文件名字，默认.fi 文件生成的路径是当前工程下的 src 文件夹，选择所需器件后单击“OK”，如图 5-14 所示。目前 User Flash 初始化文件编辑器所支持的器件与 User Flash 原语所支持的器件信息一致，如果选择的器件不支持 User Flash，则单击“OK”后会在 New File 对话框的底部打印提示信息“Current device do not support flash”；

图 5-13 初始化文件新建 New 对话框

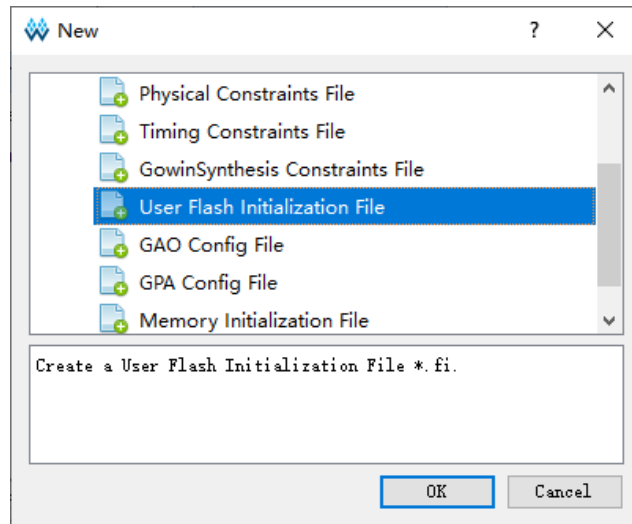
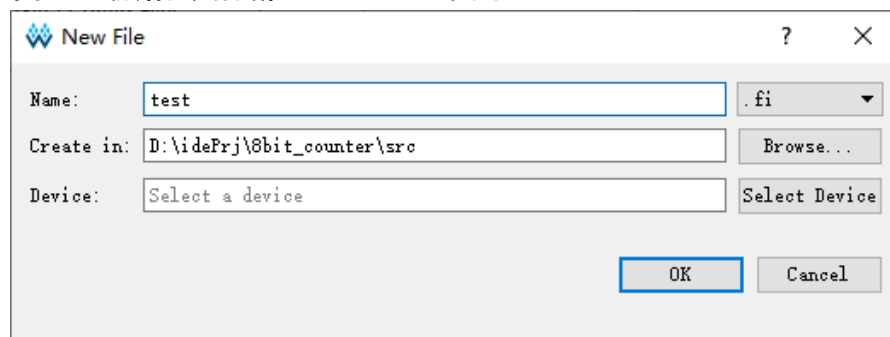
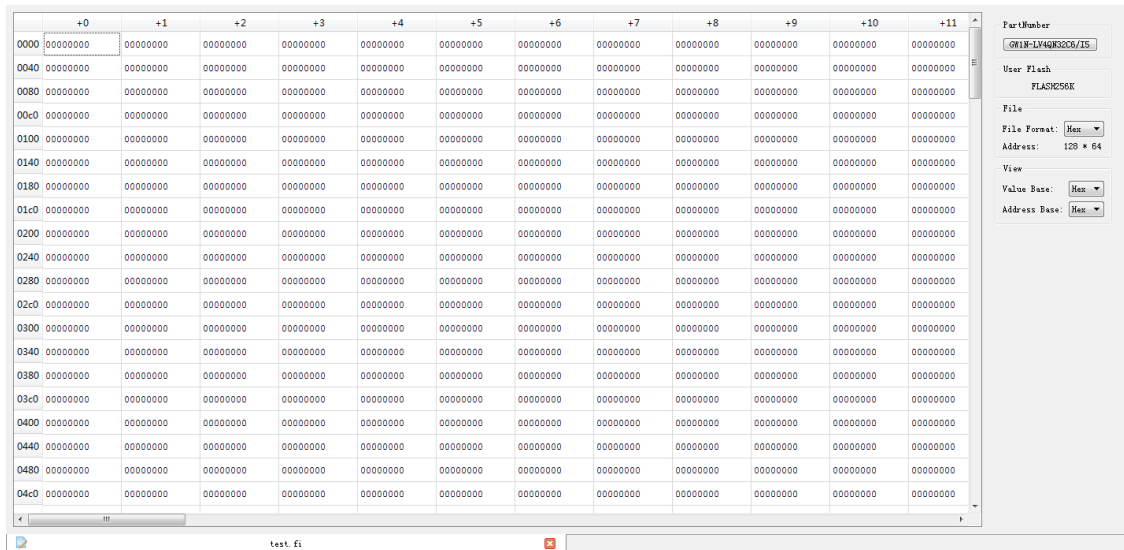


图 5-14 初始化文件新建 New File 对话框



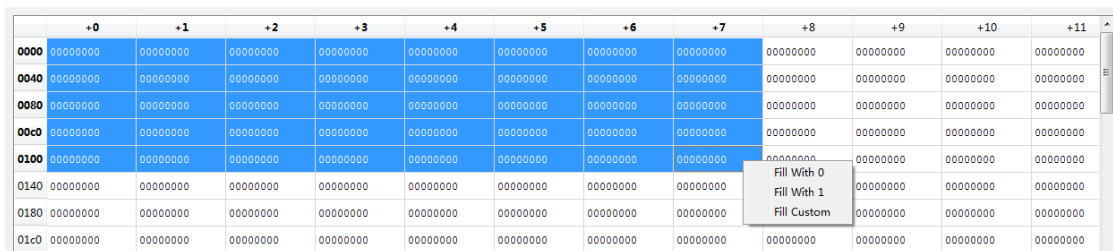
3. 启动如图 5-15 所示的初始化文件配置窗口，窗口左侧部分表格填写初始值，右侧部分配置初始化文件格式和视图格式，并且会显示芯片型号以及 User Flash 类型；

图 5-15 初始化文件配置窗口



4. 在配置窗口的右侧配置芯片型号、初始化文件的显示格式以及左侧表格中地址和初始值的数值显示格式；
 - 单击 Part Number 信息后会弹出“Select Device”对话框，可以重新选择其他芯片型号；
 - 左侧表格中地址和数值的显示格式可以选择二进制、八进制、十进制、十六进制等格式。
5. 在配置窗口的左侧表格中进行初始值的写入，此外在左侧表格中可以对表格的视图格式进行设置。表格中的初始值既可以通过双击后手动写入，也可以通过右击进行设置，在要输入数值处右击，选择“Fill with 0”是指初始值每位都为 0，“Fill with 1”是指初始值每位都为 1，“Fill Custom”使用户可以根据需要进行数值写入，同样也可批量设置初始值如图 5-16 所示；

图 5-16 批量设置



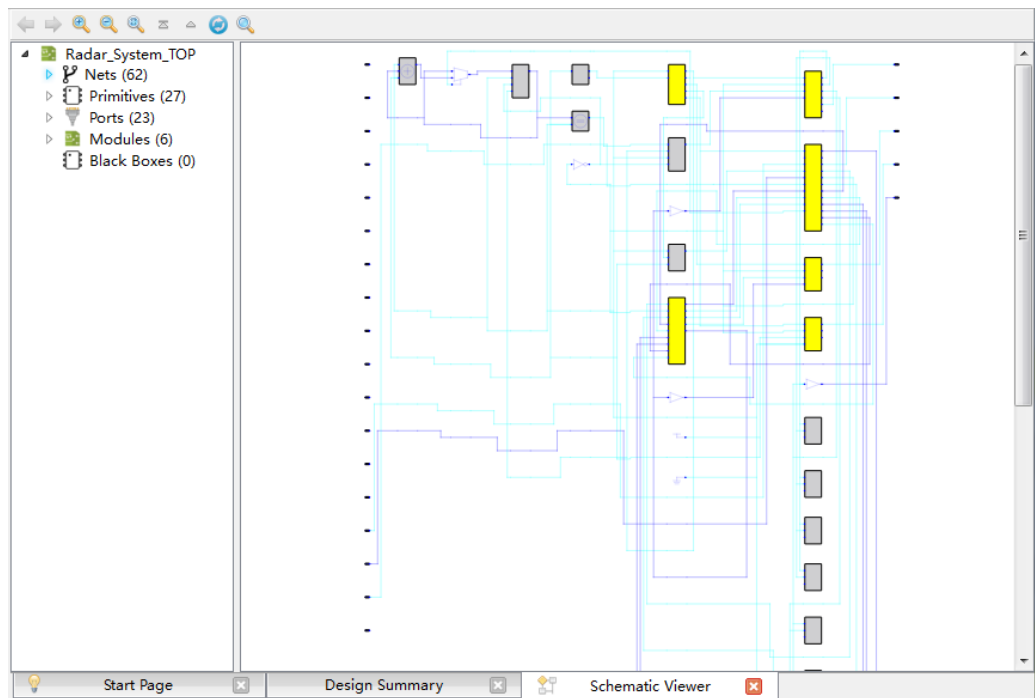
6. 保存文件。










5.8 原理图查看器

通过 HDL 原理图查看器 (Schematic Viewer) 可以直观的了解设计的逻辑构成，对后期修改设计有很好的帮助作用。Schematic Viewer 使用通用的元件符号构建电路，包括加法器、乘法器、寄存器、与门、非门和反向器等。

单击菜单栏中“Tools”下拉列表中“Schematic Viewer”，打开“Schematic Viewer”窗口，如图 5-17 所示。

图 5-17 Schematic Viewer 窗口



Schematic Viewer 窗口的工具栏中，含有后退“”、前进“”、放大“”、缩小“”、缩放“”、顶层视图“”、上一层级视图“”、重新加载“”和查找“”。在窗口的左侧会显示设计的层级结构，包括构成设计的各逻辑元件，如 Modules、Ports、Nets、Primitives、Black Boxes。

注！

Schematic Viewer 的详细使用方法请参考 [SUG755, Gowin HDL 设计原理图查看器用户指南](#)。

6 云源输出文件

高云云源在 FPGA 设计过程中，除生成码流文件以外，亦可通过使用不同的运行参数，生成多个云源报告供用户参考。默认生成的报告包括综合报告、布局布线报告、端口属性报告、时序报告和功耗分析报告等。此外，用户可通过右击 Place & Route，修改配置选项生成管脚约束文件、时序仿真模型文件等。

6.1 综合报告

GowinSynthesis®综合完成后会生成相应的综合报告及 Netlist 文件。

报告名为 *_syn.rpt.html，包含 Synthesis Message、Synthesis Details、Resource、Timing，如图 6-1 所示。

图 6-1 GowinSynthesis 综合报告

| Synthesis Details | |
|-------------------|---|
| Top Level Module | counter1 |
| | Running parser: CPU time = 0h 0m 0.109s, Elapsed time = 0h 0m 0.121s, Peak memory usage = 74.734MB Running netlist conversion: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 0MB Running device independent optimization: Optimizing Phase 0: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Optimizing Phase 1: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Optimizing Phase 2: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Running inference: Inferring Phase 0: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Inferring Phase 1: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Inferring Phase 2: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Inferring Phase 3: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Running technical mapping: Tech-Mapping Phase 0: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Tech-Mapping Phase 1: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Tech-Mapping Phase 2: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 74.734MB Tech-Mapping Phase 3: CPU time = 0h 0m 0.046s, Elapsed time = 0h 0m 0.095s, Peak memory usage = 87.988MB Tech-Mapping Phase 4: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s, Peak memory usage = 87.988MB |
| Synthesis Process | |

报告的左侧会显示具体的索引信息，详细介绍如下：

- **Synthesis Message:** 综合报告基本信息，主要包括综合的设计文件、当前 GowinSynthesis 版本、运行时间等信息；

- **Synthesis Details:** 设计文件的顶层模块、综合各子阶段的实际运行时间和 CPU 运行时间以及内存占用峰值情况、综合过程总的 CPU 运行时间和内存占用峰值情况；
- **Resource:** 资源信息，主要包括资源使用统计和器件占用统计；
- **Timing:** 时序分析报告，主要包括 Clock Summary、Max Frequency Summary、Detail Timing Paths Informations 等信息。

6.2 布局布线报告

布局布线报告，列出了用户设计占用的芯片资源信息、内存消耗信息、时间消耗信息等，方便用户了解设计的大小及与目标芯片是否匹配等内容。文件扩展名名为.rpt.html，具体信息可查看*.rpt.html 文件。

用户可在过程管理区的 Place & Route 区，双击“Place & Route Report”，打开 FPGA 工程对应的布局布线报告，如图 6-2 所示。

图 6-2 Place & Route Report

The screenshot displays the 'Place & Route Report' interface. On the left, there is a navigation tree with the following items:

- PnR Messages
- PnR Details
- Resource
 - Resource Usage Summary
 - I/O Bank Usage Summary
 - Global Clock Usage Summary
 - Global Clock Signals
 - Pinout by Port Name
 - All Package Pins

The main content area is divided into two sections:

PnR Details

Running placement:
 Placement Phase 0: CPU time = 0h 0m 0.004s, Elapsed time = 0h
 Placement Phase 1: CPU time = 0h 0m 0.263s, Elapsed time = 0h
 Placement Phase 2: CPU time = 0h 0m 0.002s, Elapsed time = 0h
 Placement Phase 3: CPU time = 0h 0m 0.8s, Elapsed time = 0h 0r
 Total Placement: CPU time = 0h 0m 1s, Elapsed time = 0h 0m 1s

Running routing:
 Routing Phase 0: CPU time = 0h 0m 0s, Elapsed time = 0h 0m 0s
 Routing Phase 1: CPU time = 0h 0m 0.189s, Elapsed time = 0h 0m 0s
 Routing Phase 2: CPU time = 0h 0m 0.082s, Elapsed time = 0h 0m 0s
 Total Routing: CPU time = 0h 0m 0.271s, Elapsed time = 0h 0m 0s

Generate output files:
 CPU time = 0h 0m 2s, Elapsed time = 0h 0m 2s

Place & Route Process

Total Time and Memory Usage CPU time = 0h 0m 3s, Elapsed time = 0h 0m 3s, Peak memory usag

Resource

Resource Usage Summary:

| Resource | Usage | Utilization |
|---------------------------|---------------------------|-------------|
| Logic | 10/20736 | 1% |
| --LUT,ALU,ROM16 | 10(3 LUT, 7 ALU, 0 ROM16) | - |
| --SSRAM(RAM16) | 0 | - |
| Register | 8/16683 | 1% |
| --Logic Register as Latch | 0/15552 | 0% |

布局布线报告的左侧会显示具体的索引信息，详细介绍如下：

- **PnR Messages:** 布局布线报告基本信息，包括报告名称、设计案例的路径及名称、物理约束文件、时序约束文件、软件版本号、器件信息、报告创建时间及法律声明；
- **PnR Details:**
 - place 各阶段所用时间及 place 总时间，如果工程中有 GAO，则包含 GAO place 的时间；
 - route 各阶段所用时间及 route 总时间，如果工程中有 GAO，则包含 GAO route 的时间；
 - 产生输出文件所用时间。

- Resource: 包括如下几项:
 - Resource Usage Summary: 用户设计所占用的器件资源信息;
 - I/O Bank Usage Summary: 用户设计所占用的 I/O bank 信息;
 - Global Clock Usage Summary: 所用全局时钟信息;
 - Global Clock Signals: 用户设计中所用的时钟信号;
 - Pinout by Port Name: 用户设计中的 port 所占用的引脚信息;
 - All Package Pins: 当前器件封装下所有引脚的详细信息。

如果工程中有 GAO, 则还包括 GAO Resource Usage Summary: 用户设计中 GAO 所占用的器件资源信息。

6.3 端口属性报告

端口属性报告, 是在布局之后输出的端口属性文件, 包括端口的类型、属性及端口位置信息等, 生成的文件扩展名名为 .pin.html, 具体信息可查看 *.pin.html 文件。

用户可在过程管理区的 Place & Route 区, 双击 “Ports & Pins Report”, 打开 FPGA 工程对应的端口属性报告, 如图 6-3 所示。

图 6-3 Ports & Pins Report

| Port Name | Diff Pair | Loc./Bank | Constraint | Dir. | Site | IO Type | Drive | Pull Mode | PCI |
|-----------|-----------|-----------|------------|------|----------|-----------|-------|-----------|-----|
| clk | | L1/7 | N | in | IOL25[A] | LVC MOS18 | OFF | DOWN | OFF |
| cout[0] | | M2/7 | N | out | IOL25[B] | LVC MOS18 | 8 | NONE | OFF |
| cout[1] | | F6/8 | N | out | IOL3[A] | LVC MOS18 | 8 | NONE | OFF |
| cout[2] | | G7/8 | N | out | IOL3[B] | LVC MOS18 | 8 | NONE | OFF |
| cout[3] | | D3/8 | N | out | IOL2[A] | LVC MOS18 | 8 | NONE | OFF |
| cout[4] | | D4/8 | N | out | IOL2[B] | LVC MOS18 | 8 | NONE | OFF |
| cout[5] | | A2/0 | N | out | IOT2[B] | LVC MOS18 | 8 | NONE | OFF |
| cout[6] | | E6/0 | N | out | IOT3[A] | LVC MOS18 | 8 | NONE | OFF |
| cout[7] | | F5/8 | N | out | IOL4[B] | LVC MOS18 | 8 | NONE | OFF |

| Loc./Bank | Signal | Dir. | Site | IO Type | Drive | Pull Mode | PCI Clamp | Hysteresis | Open Drain |
|-----------|---------|------|---------|-----------|-------|-----------|-----------|------------|------------|
| B1/0 | - | out | IOT2[A] | LVC MOS18 | 8 | NONE | OFF | OFF | ON |
| A2/0 | cout[5] | out | IOT2[B] | LVC MOS18 | 8 | NONE | OFF | OFF | OFF |
| E6/0 | cout[6] | out | IOT3[A] | LVC MOS18 | 8 | NONE | OFF | OFF | OFF |
| F7/0 | - | out | IOT3[B] | LVC MOS18 | 8 | NONE | OFF | OFF | ON |
| B2/0 | - | out | IOT4[A] | LVC MOS18 | 8 | NONE | OFF | OFF | ON |
| A3/0 | - | out | IOT4[B] | LVC MOS18 | 8 | NONE | OFF | OFF | ON |

端口属性报告的左侧会显示具体的索引信息, 详细介绍如下:

- Pin Messages: 端口属性报告基本信息, 包括报告名称、设计案例的路径及名称、物理约束文件、时序约束文件、软件版本号、器件信息、报告创建时间及法律声明;
- Pin Details: 包括如下几项:
 - Pinout by Port Name: 用户设计中的 port 所占用的引脚信息;

- All Package Pins: 当前器件封装下所有引脚的详细信息。

6.4 时序报告

时序报告会对电路网表中的时序模型进行全面的分析，计算电路中时序路径延迟，并判断其是否满足要求。时序报告包括建立时间检查、保持时间检查、恢复时间检查、移除时间检查、最小时钟脉冲检查、最大扇出路径、布线拥塞度报告等部分，默认情况下对上述所有检查进行报告，并提供最大频率的报告。

用户可在过程管理区的 Place & Route 区，双击“Timing Analysis Report”，打开 FPGA 工程对应的时序报告，如图 6-4 所示。

注！

时序报告的详细信息请参考 [SUG940, Gowin 设计时序约束指南](#)中的时序报告部分。

图 6-4 时序报告内容

The screenshot displays the 'Timing Summaries' section of a report. On the left is a navigation tree with categories like 'Timing Messages', 'Timing Summaries', 'Timing Details', and 'Timing Report By Analysis Type'. The main content area contains three tables:

STA Tool Run Summary:

| | |
|-------------------------------------|----------------|
| Setup Delay Model | Slow 1.14V 85C |
| Hold Delay Model | Fast 1.26V 0C |
| Numbers of Paths Analyzed | 8 |
| Numbers of Endpoints Analyzed | 8 |
| Numbers of Falling Endpoints | 2 |
| Numbers of Setup Violated Endpoints | 0 |
| Numbers of Hold Violated Endpoints | 0 |

Clock Summary:

| Clock Name | Type | Period | Frequency(MHz) | Rise | Fall | Source | Master | Objects |
|------------|------|--------|----------------|-------|--------|--------|--------|-------------|
| clk0 | Base | 20.000 | 50.000 | 0.000 | 10.000 | | | clk0_ibuf/I |
| clk1 | Base | 20.000 | 50.000 | 0.000 | 10.000 | | | clk1_ibuf/I |

Max Frequency Summary:

| NO. | Clock Name | Constraint | Actual Fmax | Logic Level | Entity |
|-----|------------|-------------|--------------|-------------|--------|
| 1 | clk0 | 50.000(MHz) | 437.197(MHz) | 1 | TOP |
| 2 | clk1 | 50.000(MHz) | 525.803(MHz) | 1 | TOP |

6.5 功耗分析报告

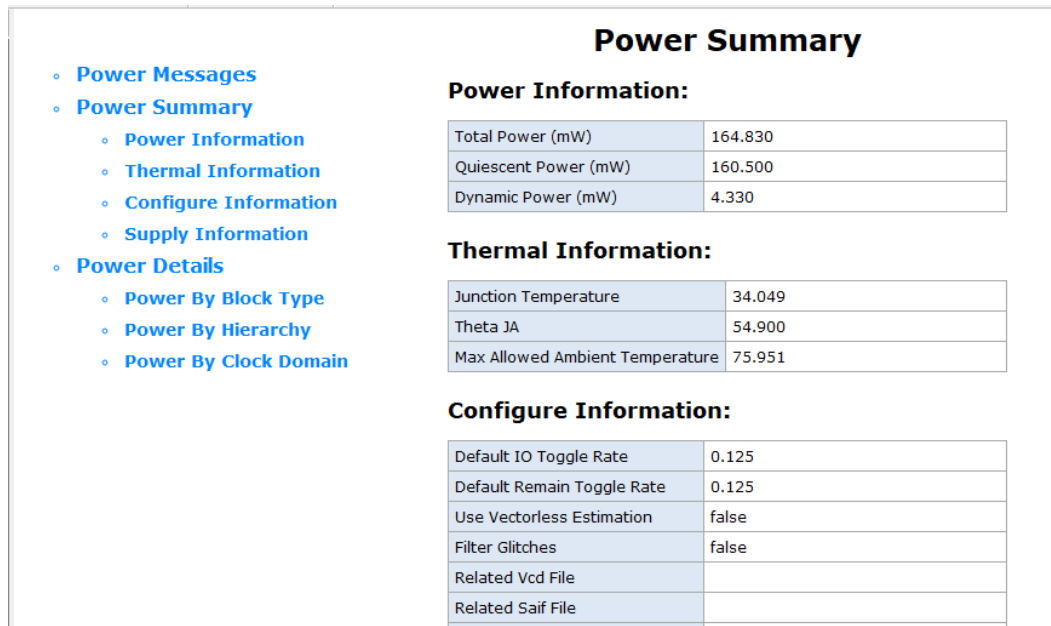
功耗分析报告主要是针对用户设计，根据器件的特性做一个预估的功耗计算，帮助用户评估设计的基本功耗值。

用户可在过程管理区的 Place & Route 区，双击“Power Analysis Report”，打开 FPGA 工程对应的功耗分析报告，如图 6-5 所示。

注！

有关功耗分析的影响因素及功耗分析报告的详细内容，请参考 [SUG282, Gowin 功耗分析工具用户指南](#)。

图 6-5 功耗分析报告



7 仿真文件

云源提供用于仿真的输入文件，仿真分为功能仿真和时序仿真，功能仿真也称为前仿真，主旨在于验证电路的功能是否符合设计要求，其特点是不考虑电路门延迟与线延迟，主要是验证电路与理想情况是否一致。

时序仿真也称为后仿真或者布局布线后仿真，是指电路已经映射到特定的工艺环境以后，综合考虑电路的路径延迟与门延迟的影响，验证电路能否在一定时序条件下满足设计构想的过程，是否存在时序违规。

7.1 功能仿真文件

功能仿真包括综合前的用户 RTL 设计功能仿真和综合后逻辑网表功能仿真，所需要的文件有：综合前用户设计 RTL 文件、综合后网表文件 (*.vg)、激励文件 (testbench) *tb.v 以及功能仿真库文件 prim_sim.v。

注！

- 仿真库文件所在目录：installPath\IDE\simlib；
- 由于产生的 IP 为密文，因此，设计中含有 IP 时，需要将 IP 产生后的 vo 文件作为功能仿真文件，vo 文件在当前工程目录 src 下产生的 IP 目录 src\ipName\ipName.vo。

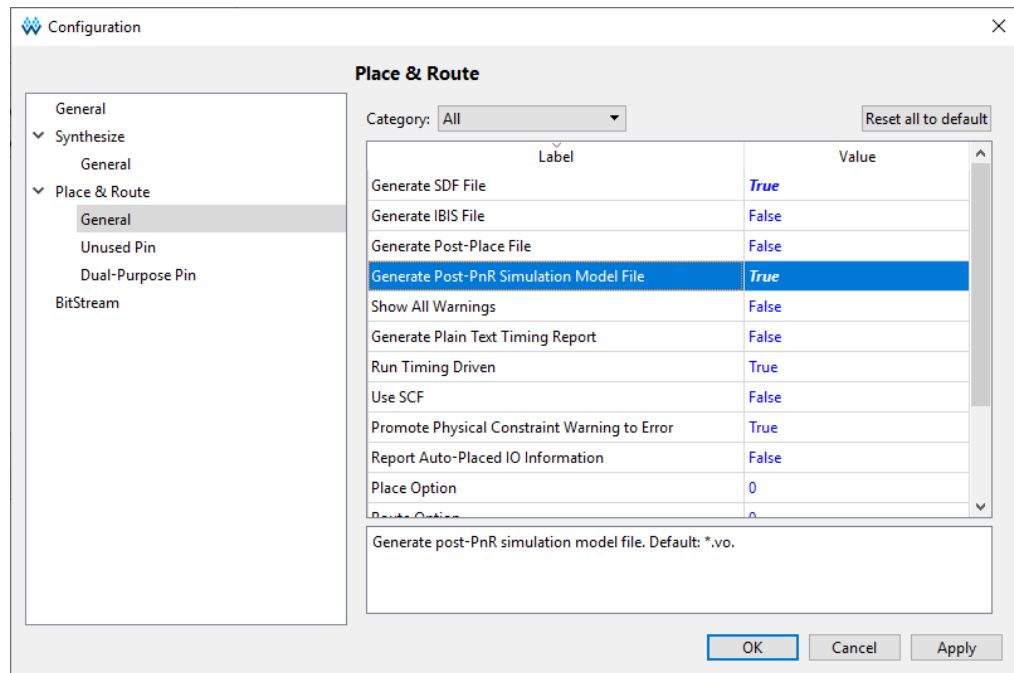
7.2 时序仿真文件

时序仿真需要的文件：软件产生的时序仿真逻辑网表文件*.vo、对应的延时文件*.sdf、对应的激励文件*tb.v 以及时序仿真库 prim_tsim.v。

时序仿真逻辑网表文件*.vo 和延时文件*.sdf 可以通过云源界面运行工程生成。具体步骤如下：

1. 建立/打开工程后，Configuration 进行设置，在 Place & Route 选项中把“Generate SDF File”和“Generate Post-PNR Simulation Model File”的 Value 设置成 True，单击“OK”，如图 7-1 所示。

图 7-1 选项配置



2. 运行 Place & Route，运行成功后在工程所在路径下的 impl/pnr/下可以找到所需要的 vo 和 sdf 文件。

8 Tcl 命令说明

云源支持命令行模式运行。在以下说明中，尖括号<>包含的内容为必须指定的内容，方括号[]包含的内容为可选的内容。

8.1 启动命令行模式

8.1.1 gw_sh.exe

语法

命令：以 Windows 系统为例，启动安装目录下*.x\IDE\bin\gw_sh.exe

参数：

[script file]

参数为空：直接进入命令行控制台模式；

script file：可选项，执行一个指定的脚本文件。

应用示例：

```
#启动命令行模式
```

```
gw_sh.exe
```

```
#执行脚本文件
```

```
gw_sh.exe script_file
```

8.2 命令介绍

8.2.1 add_file

```
add_file [-type] [- disable] [ -h/-help] <file...>
```

添加设计文件，Windows 系统和 Linux 系统下文件路径的分隔符支持两种书写形式：/或\。支持相对路径和绝对路径，相对路径形式在云源界面下相对的是当前工程的路径，在命令行模式下相对的是启动 gw_sh 时的路径。

参数：

<file...> 要添加的设计文件，可以指定多个，多个文件以空格分隔。

选项:

- `-type <type> add_file` 命令会根据文件扩展名自动判断文件类型，也可以使用该选项来明确指定文件类型，支持的文件类型有 `verilog`、`vhdl`、`netlist`、`cst`、`sdc`、`fdc`、`gao`、`gpa`、`gsc`;
- `-disable` 将添加的文件置为失效状态。处于失效状态的文件仅添加到文件列表中，不参与流程实现。相关命令：`set_file_enable`;
- `-h`，`--help` 显示帮助信息。

例:

```
add_file abc.v
add_file -type vhdl 1.vhd 2.vhdl 3.vhd
add_file D:/gowin_project/top.v
add_file D:\\gowin_project\\top.v
```

8.2.2 rm_file

```
rm_file [-h/-help ] <files...>
```

移除设计文件，文件路径形式请参考 8.2.1 `add_file`。

参数:

`<file...>` 要移除的设计文件，可以指定多个文件，以空格分隔。

选项:

`-h`，`--help` 显示帮助信息。

例:

```
rm_file a.v
rm_file a.v b.v c.v
rm_file D:/gowin_project/top.v
rm_file D:\\gowin_project\\top.v
```

8.2.3 set_device

```
set_device [-name] [-h/-help ] <part number>
```

设置芯片型号。

参数:

`<part number>` 指定目标器件的 Part Number，如 `GW1N-UV4LQ144C6/I5`。

选项:

- `-name <name>` 指定器件的名称，如 `GW1N-4`;
- `-h`，`--help` 显示帮助信息。

例:

```
set_device GW1N-LV1CS30C6/I5
set_device -name GW1N-4 GW1N-UV4LQ144C6/I5
```

8.2.4 set_file_prop

```
set_file_prop [-lib] [-h/-help] <file...>
```

设置文件属性，文件路径形式请参考 8.2.1 add_file。

参数：

<file...> 指定要设置的文件，可以指定多个，多个文件之间以空格分隔。

选项：

- -lib <name> 设置文件的 library name。该选项仅对 VHDL 类型的文件有效；
- -h, --help 显示帮助信息。

例：

```
set_file_prop -lib work top .vhd
set_file_prop -lib work D:/gowin_project/top.vhd
set_file_prop -lib work D:\\gowin_project\\top.vhd
```

8.2.5 run

```
run [-h/-help] <syn/pnr/all>
```

运行全部流程或某个流程。

参数：

<syn/pnr/all> 指定要运行的流程名称，可以被执行的流程名称有 syn 和 pnr，分别表示综合及布局布线。也可以指定 all，表示运行全部的流程。

选项：

-h, --help 显示帮助信息。

例：

```
run pnr
run all
```

8.2.6 set_file_enable

```
set_file_enable [-h/-help] <file> <true|false>
```

设置文件是否可以被使用，文件路径形式请参考 8.2.1 add_file。

参数：

- <file> 指定要设置的文件。
- <true|false> true 表示文件可以被使用，false 表示不可被使用。

选项：

-h, --help 显示帮助信息。

例:

```
set_file_enable top.v false
set_file_enable D:/gowin_project/top.v
set_file_enable D:\\gowin_project\\top.v
```

8.2.7 saveto

saveto [-all_options] [-h/-help] <file>

将当前工程设计数据保存到 tcl 脚本。文件路径形式请参考 8.2.1 add_file。

参数:

<file> 导出的文件名。

选项:

- -all_options saveto 命令默认只保存修改过的选项信息，即与默认值不同的选项。可以通过指定 -all_options 来保存所有的选项信息。
- -h, --help 显示帮助信息。

例:

```
saveto project.tcl
saveto -all_options project.tcl
saveto -all_options D:/gowin_project/project.tcl
saveto -all_options D:\\gowin_project\\project.tcl
```

8.2.8 set_option

set_option [options]

设置工程相关的属性配置及流程选项。

选项:

- -output_base_name <name>
指定输出文件的文件名。此选项仅指定文件的 base name，不同流程会根据输出文件的类型使用合适的扩展名。如 -output_base_name abc，则 gowinsynthesis 综合产生的网表文件名为 abc.vg;
- -synthesis_tool <tool>
指定综合工具 GowinSynthesis®;
- -top_module <name>
指定 Top Module/Entity;
- -include_path <path or path list>
指定包含路径。当指定多个包含路径时，路径之间需要使用分号进行分隔，并使用一对大括号 {} 包含所有的路径信息，如 -include_path

{/path1;/path2;/path3}。支持相对路径和绝对路径，相对路径为相对程序当前运行路径；

- **-inc <incremental.cfg >**

指定增量编译的配置文件 `incremental.cfg`。Windows 系统和 Linux 系统下文件路径的分隔符支持两种书写形式：/或\\，支持相对路径和绝对路径，相对路径形式在云源界面下相对的是当前工程的路径，在命令行模式下相对的是启动 `gw_sh` 时的路径。

在云源界面中将增量编译模式设置为 `Hard` 后会在工程路径/`impl` 下生成配置文件 `incremental.cfg`，文件内容为“`top module name,hard,place`”。

综合属性配置

- **-verilog_std <v1995|v2001|sysv2017>**

指定 Verilog 语言标准：Verilog 95/Verilog 2001/System Verilog 2017，默认为 Verilog 2001。

- **-vhdl_std <vhdl1993|vhdl2008>**

指定 VHDL 语言标准：VHDL 1993/VHDL 2008，默认为 VHDL1993。

- **-dsp_balance <0|1>**

指定综合时是否自动进行 DSP Balance，默认为 0。

0：不会自动进行 DSP Balance；

1：自动进行 DSP Balance。

- **-print_all_synthesis_warning <0|1>**

指定是否打印所有的综合 warning 信息，默认为 0。

0：不会打印所有的 warning 信息；

1：会打印所有的 warning 信息。

- **-disable_io_insertion <0|1>**

启用或禁用 I/O 插入的使能控制，默认为 0。

0：禁用 I/O 插入的使能控制；

1：启用 I/O 插入的使能控制。

- **-looplimit <value>**

设置 RTL 中默认的编译器循环限制值，默认值为 2000。

- **-maxfan <value>**

设置一个输入端口、net 或寄存器输出端的扇出值，默认为 10000。

- **-rw_check_on_ram <0|1>**

当需要在写入期间进行读取时防止仿真失败，启用该选项会自动插入旁路逻辑，默认为 1。

0: 不启用该选项;

1: 启用该选项。

Place & Route 属性配置

- **-gen_sdf <0|1>**

指定 Place & Route 是否产生 SDF 文件, 默认为 0。

0: 不产生 SDF 文件;

1: 产生 SDF 文件。

- **-gen_io_cst <0|1>**

指定 Place & Route 是否产生文件名为*.io.cst 的 port 端口的物理约束文件, 默认为 0。

0: 不产生*.io.cst 文件;

1: 产生*.io.cst 文件。

- **-gen_ibis <0|1>**

指定 Place & Route 是否产生文件名为*.ibs 的输入/输出缓冲区信息指定文件, 默认为 0。

0: 不产生*.ibs 文件;

1: 产生*.ibs 文件。

- **-gen_posp <0|1>**

指定 Place & Route 是否产生文件名为*.posp 的器件布局文件, 默认为 0。

0: 不产生*.posp 文件;

1: 产生*.posp 文件。

- **-gen_text_timing_rpt <0|1>**

指定 Place & Route 是否产生文件名为*.tr 的文本格式的时序报告, 默认为 0。

0: 不产生*.tr 文件;

1: 产生*.tr 文件。

- **-gen_sim_netlist <0|1>**

指定 Place & Route 是否产生文件名为*.vo 的时序仿真模型文件, 默认为 0。

0: 不产生*.vo 文件;

1: 产生*.vo 文件。

- **-show_init_in_vo <0|1>**

将默认初始值添加到生成的 Place & Route 时序仿真模型文件中的 instance 中, 默认为 0。

0: 不会将默认初始值添加到时序仿真模型文件中的 **instance** 中;

1: 将默认初始值添加到时序仿真模型文件中的 **instance** 中。

- **-show_all_warn <0|1>**

Place & Route 运行时输出所有的 **warning** 信息，默认为 0。

0: 不会输出 **Place & Route** 运行时所有的 **warning** 信息;

1: 输出 **Place & Route** 运行时所有的 **warning** 信息。

- **-timing_driven <0|1>**

启用该选项后会对布局布线进行时序驱动优化，默认为 1。

0: 不会进行布局布线时序驱动优化;

1: 进行布局布线时序驱动优化。

- **-use_scf <0|1>**

使用综合工具 **Synplify Pro** 生成的*.scf 文件作为附加的时序约束文件，默认为 0。

0: 不使用*.scf 文件;

1: 使用*.scf 文件;

- **-cst_warn_to_error <0|1>**

将 **Place & Route** 运行时的物理约束警告提升为错误信息，默认为 1。

0: 不会将物理约束警告提升为错误信息;

1: 将物理约束警告提升为错误信息。

- **-rpt_auto_place_io_info <0|1>**

报告自动 **place** 的 IO 位置信息，默认为 0。

0: 不会报告自动 **place** 的 IO 位置信息;

1: 报告自动 **place** 的 IO 位置信息。

- **-place_option <0|1>**

布局算法选项，默认为 0。

0: 采用默认布局算法;

1: 采用布局算法 1。

- **-route_option <0|1|2>**

布线算法选项，默认为 0。

0: 采用默认布线算法;

1: 采用布线算法 1;

2: 采用布线算法 2。

- **-ireg_in_job <0|1>**

启用该选项，Place & Route 会将输入 Buffer 连接的寄存器布局到 IOB 上，默认为 1。

0: 不使能;

1: 使能。

- **-oreg_in_iob <0|1>**

启用该选项，Place & Route 会将输出/三态 Buffer 连接的寄存器布局到 IOB 上，默认为 1。

0: 不使能;

1: 使能。

- **-ioreg_in_iob <0|1>**

启用该选项，Place & Route 会将双向 Buffer 连接的寄存器布局到 IOB 上，默认为 1。

0: 不使能;

1: 使能。

注!

Place & Route 相关运行选项的详细用法，请参考本文档 4.3.3 小节中的 Place & Route 部分。

复用管脚属性配置

- **-use_jtag_as_gpio <0|1>**

将 JTAG 相关管脚复用为普通 IO 管脚，JTAG 相关的管脚为 TCK、TMS、TDI、TDO，默认为 0。

0: 作为 JTAG 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_ssipi_as_gpio <0|1>**

将 SSPI 相关管脚复用为普通 IO 管脚，SSPI 相关的管脚为 SCLK、CLKHOLD_N、SI、SO、SSPI_CS_N，默认为 0。

0: 作为 SSPI 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_msipi_as_gpio <0|1>**

将 MSPI 相关管脚复用为普通 IO 管脚，MSPI 相关的管脚为 MCLK、MCS_N、MI、MO，默认为 0。

0: 作为 MSPI 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_ready_as_gpio <0|1>**

将 READY 相关管脚复用为普通 IO 管脚，READY 相关的管脚为 READY，

默认为 0。

0: 作为 READY 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_done_as_gpio <0|1>**

将 DONE 相关管脚复用为普通 IO 管脚, DONE 相关的管脚为 DONE, 默认为 0。

0: 作为 DONE 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_reconfig_as_gpio <0|1>**

将 RECONFIG_N 相关管脚复用为普通 IO 管脚, RECONFIG_N 相关的管脚为 RECONFIG_N, 默认为 0。

0: 作为 RECONFIG_N 专用管脚;

1: 复用为普通 IO 管脚。

- **-use_i2c_as_gpio <0|1>**

将 I2C 相关管脚复用为普通 IO 管脚, I2C 相关的管脚为 SCL 和 SDA, 默认为 0。

0: 作为 I2C 专用管脚;

1: 复用为普通 IO 管脚。

BitStream 属性配置

- **-bit_format <txt|bin>**

用于指定生成的码流文件内容的格式。

- **-bit_crc_check <0|1>**

对码流文件启用循环冗余校验。

0: 不启用循环冗余校验;

1: 启用循环冗余校验

- **-bit_compress <0|1>**

对生成的码流文件进行压缩。

0: 不对码流文件进行压缩;

1: 对码流文件进行压缩。

- **-bit_encrypt <0|1>**

对码流文件进行加密处理, 仅支持 GW2A 系列。

0: 不进行加密处理;

1: 进行加密处理。

- **-bit_encrypt_key <key>**

该项与“-bit_encrypt”配合使用，用户可以对加密的密钥进行自定义。

- **-bit_security <0|1>**
安全位使能控制。
0: 关闭安全位使能控制;
1: 使用安全位使能控制。
- **-bit_incl_bsram_init <0|1>**
将 BSRAM 的初始值打印进码流文件中。
0: 不将 BSRAM 的初始值打印进码流文件中;
1: 将 BSRAM 的初始值打印进码流文件中。
- **-bg_programming <off | jtag | i2c | internal | i2c_jtag_ssipi_qsspi>**
背景升级功能，在不中断 FPGA 芯片现有功能执行的前提下对 Flash 进行烧录。
off: 不使用背景升级功能;
jtag: 使用 JTAG 模式进行背景升级;
i2c: 使用 I2C 模式进行背景升级;
internal: 使用 FPGA 内部逻辑进行背景升级;
i2c_jtag_ssipi_qsspi: 使用 I2C/JTAG/SSPI/QSSPI 模式进行背景升级。
- **-hotboot <0|1>**
0: 不使用热启动模式;
1: 使用热启动模式。
- **-i2c_slave_addr <value>**
I2C Slave Address(Hex): 可配置值范围为 00~7F。
- **-secure_mode <0|1>**
启用安全模式,此时 JTAG 管脚为 GPIO,码流文件只能对设备编程一次。
0: 不启用安全模式;
1: 启用安全模式。
- **-loading_rate <value>**
AutoBoot 配置模式和 MSPI 配置模式下,码流数据从 Flash 到 SRAM 的加载速度,默认为 2.500MHz。
- **-spi_flash_addr <value>**
指定 SPI Flash 的地址。
- **-program_done_bypass <0|1>**
配置该选项后,在 Done Final 内部信号生效时,同时外部 Done Pin 保持低电平,使码流加载完成后可以转发新的码流数据,默认为 0。

0: 不启用该功能;

1: 启用该功能。

- `-power_on_reset <0|1>`

上电复位功能的使能控制。

0: 不启用上电复位功能;

1: 启用上电复位功能。

- `-wakeup_mode <0|1>`

Wake Up Mode 的使能控制。

0: 关闭 Wake Up Mode 的使能控制;

1: 打开 Wake Up Mode 的使能控制。

- `-user_code <default|value>`

用户可以自定义 User Code。

注!

BitStream 相关选项的详细解释, 请参考本文档 4.3.3 小节中的 BitStream 部分。

Unused Pin 属性配置

`-unused_pin <default|open_drain>`

对未使用的管脚(除去复用管脚)设置不同的 IO 类型和属性值。

注!

Unused Pin 相关选项的详细解释, 请参考本文档 4.3.3 小节中的 Unused Pin 部分。

8.2.9 source

`source <file>`

在云源的 tcl 命令编辑窗口中或启动命令行模式后, 使用该命令可以执行 tcl 脚本。文件路径形式请参考 8.2.1 add_file。

参数:

`<file>` 要被执行的 tcl 脚本文件。

例:

```
source project.tcl
```

```
source D:/gowin_project/project.tcl
```

```
source D:\\gowin_project\\project.tcl
```

