Gowin Software

User Guide

SUG100-4.4.2E, 06/27/2025

Disclaimer

GOWINSEMI assumes no liability and provides no warranty (either expressed or implied) and is not responsible for any damage incurred to your hardware, software, data, or property resulting from usage of the materials or intellectual property except as outlined in the GOWINSEMI Terms and Conditions of Sale. GOWINSEMI may make changes to this document at any time without prior notice. Anyone relying on this documentation should contact GOWINSEMI for the current documentation and errata.

Revision History

Date	Version	Description
06/17/2021	2.5E	● Screenshots and their descriptions updated. ● Synplify Pro removed.
11/02/2021	2.6E	● SSPI and MSPI dual-purpose pins updated. ● The descriptions of -ireg_in_iob/-oreg_in_iob/-ioreg_in_iob updated. ● MODE dual-purpose configuration removed. ● Place & Route BitStream: Power On Reset added. ● The description of simulation files added.
05/20/2022	2.7E	The description of Loading Rate updated.
07/28/2022	2.8E	● Route Maxfan option added in Place & Route. ● The use of Library added.
10/28/2022	2.9E	● power_on_reset name updated. ● Turn Off Bandgap option added in Bitstream. ● DSim added. ● Chapter 9 Appendix added.
12/16/2022	3.0E	● Device Version information added. ● TclPre added in Synthesize. ● The value Internal of Background Programming modified to GoConfig/UserLogic. ● Generate Post-PnR VHDL Simulation Model File added in Place & Route.
03/31/2023	3.1E	● The value GoConfig/UserLogic of Background Programming divided into GoConfig and UserLogic, and the related descriptions updated. ● FloorPlanner and Timing Constraints Editor added to the toolbar. ● The Tcl command -clock_route_order added.
04/20/2023	3.2E	● Place & Route and Bitstream configurations updated. ● Multi Boot and MSPI JUMP added in Bitstream configuration.
05/25/2023	3.3E	● Enable External Master Config Clock and Enable CMSER added in Bitstream configuration. ● Global configuration added to set the VCCX value.
06/30/2023	3.4E	The default value of Ram R/W Check option in Synthesis Configuration updated to unchecked.
08/18/2023	3.5E	● For GW5A-25-MBGA121N, Use SSPI as regular IO option is checked and cannot be modified. ● For GW5AT-138/GW5AST-138/GW5A-138 devices, the default values of Place input register to IOB, Place output register to IOB and Place inout register to IOB are modified to False. ● For GW5AT-138/GW5AST-138/GW5A-138 devices, Replicate Resources option added to Place & Route configuration.
09/28/2023	3.6E	● The Loading Rate values of GW5A(S)(T)-138 and GW5A(R)-25 updated. ● The configuration option CMSER updated. ● The MSPI JUMP option under the configuration option Feature sysControl updated.
10/31/2023	3.7E	● Figure 4-16 File Properties Dialog Box updated. ● Tcl commands create_project and import_files added.

Date	Version	Description
11/30/2023	3.8E	● Program Device updated to Programmer. ● A value of 2 added in Place option for Place & Route configuration. ● Loading Rate for LittleBee and Arora devices updated. ● Tcl command run close added. ● Output Base Name option added.
02/02/2024	3.9E	● GW2AN-9X and GW2AN-18X Loading Rate updated. ● "Enable CMSER" updated to "Enable SEU Handler". ● "Constraints" option added for configuring Frequency.
03/29/2024	4.0E	● GW5AT-60 Loading Rate values added. ● Configuration option "Enable CTP" for Bitstream added.
06/28/2024	4.1E	● Tcl command open_project added. ● Configuration option "VCC" added in Place & Route. ● Virtual input/output debugging tool added.
08/09/2024	4.1.1E	● The configuration option Place Option in Place & Route now includes a new value of 3. ● Descriptions of the open_project command updated.
10/25/2024	4.2E	● The configuration option Place Option in Place & Route now includes a new value of 4. ● A new value "GoConfig Mode1" added to the Background Programming configuration option for the devices Version C GW1N-2/GW1NR-2/GW1N-1P5. ● A new configuration option "Incremental PnR" added to Place & Route.
12/31/2024	4.3E	● A search function added to the Hierarchy window. ● GoBert eye diagram analysis tool supported.
03/07/2025	4.4E	● The SPI Flash address access mode "Normal" updated to "Single" in BitStream configuration option. ● Descriptions of Chapter 8 Tcl Commands updated.
04/30/2025	4.4.1E	● The interface dark mode added. ● The search function added in the device selection window.
06/27/2025	4.4.2E	A new configuration option and Tcl command added to specify the CSR file.

Contents

List of Figures

List of Tables

1 About This Guide

1.1 Purpose

This manual describes Gowin Software installation and operation, and it aims to help you to be familiar with the using flow and improve design efficiency. The software screenshots in this manual are based on V1.9.11.03. As the software is subject to change without notice, some information may not remain relevant and may need to be adjusted according to the software that is in use.

1.2 Related Documents

The latest user guides are available on GOWINSEMI Website. You can find the related documents at www.gowinsemi.com:

- SUG940, Gowin Design Timing Constraints Guide
- SUG935, Gowin Design Physical Constraints Guide
- SUG114, Gowin Analyzer Oscilloscope User Guide
- SUG282, Gowin Power Analyzer User Guide
- SUG502, Gowin Programmer User Guide
- UG285, Gowin BSRAM & SSRAM User Guide
- SUG283, Gowin Primitives User Guide
- UG286, Gowin Clock User Guide
- UG287, Gowin Digital Signal Processing (DSP) User Guide
- UG289, Gowin Programmable IO (GPIO) User Guide
- UG295, Gowin User Flash User Guide
- SUG1018, Arora V Design Physical Constraints User Guide
- UG299, Arora V Series Analog to Digital Converter (ADC) User Guide
- UG300, Arora V BSRAM & SSRAM User Guide
- UG304, Arora V Programmable IO (GPIO) User Guide
- UG305, Arora V Digital Signal Processing (DSP) User Guide
- UG306, Arora V Clock User Guide

- SUG1189, Gowin Virtual Input Output User Guide
- SUG1198, Gowin GoBert User Guide

1.3 Terminology and Abbreviations

Table 1-1 shows the abbreviations and terminology used in this manual.

Table 1-1 Terminology and Abbreviations

Terminology and Abbreviations	Meaning
CRC	Cyclic Redundancy Check
FloorPlanner	Physical Constraints Editor
FPGA	Field Programmable Gate Array
GAO	Gowin Analyzer Oscilloscope
GowinSynthesis	GowinSynthesis
GPA	Gowin Power Analyzer
GVIO	Gowin Virtual Input/Output
IP Core	Intellectual Property Core
PCIe	Peripheral Component Interconnect Express
PnR	Place & Route
Schematic Viewer	HDL Schematic Viewer
SEU Handler	Single-Event Upsets Handler
Tcl	Tool Command Language

1.4 Support and Feedback

Gowin Semiconductor provides customers with comprehensive technical support. If you have any questions, comments, or suggestions, please feel free to contact us directly by the following ways.

Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

2 Overview

2.1 Introduction

Gowin design system is an integrated circuit design and implementation tool specifically for GOWIN FPGA chips, and it has superior performance and easy to use. Gowin design system provides a comprehensive and optimized design for Gowin FPGA chips with the low-power and low-cost architecture, and it integrates the flow from RTL description to FPGA bitstream file generation, including design optimization, automatic design, and graphical interaction, etc.

Functions

- The software system supports all functions of Gowin FPGA chips, covering the complete design flow from the functional description of the RTL circuit to the generation of FPGA bitstream file
- GowinSynthesis supports high-performance logic design and synthesis
- Supports automatic design and interactive graphical design in parallel
- Supports Centos6.8/7.0/7.3/7.5/8.2 (64 bits), Ubuntu18.04/20.04LTS, Win7/8/10/11 (32 bits/64 bits), Win XP (32 bits) systems
- Millions of gate-level software
- Supports VHDL, Verilog HDL and System Verilog languages
- Supports optimized architecture of Gowin products
- Supports original and high-performance algorithm PnR
- Precise timing analysis and timing report
- Clock analysis and control to ensure better timing performance
- Supports various timing and physical constraints
- Supports real-time monitoring of hardware circuit signals and storing them, along with timing waveform diagrams display
- Resource sharing can improve chip utilization and reduce cost

Features

- Integrated design

  - The design can be completed in stages or automatically as a package
  - Supports command line mode or GUI mode
  - You can use scripted design, and design any single module flexibly without affecting the integrated design throughout
- Can optimize design
  - Netlist optimization
  - Quick timing optimization analysis and design
  - Resource analysis and optimization
- Hierarchy design and analysis
  - Supports hierarchical netlist input and output
  - Supports flattened netlist input and output
  - Can hierarchically display, trace, and analyze netlist
- Flexible interactive graphic design
  - Simple and clear user interface
  - Displays projects, modules, tools and output
  - Design constraint input, selection and update
  - Quick timing analysis and report
  - Push button design

2.2 Supported Devices

Gowin Software supports the LittleBee family and Arora family chips. For the details of chip types, resources and packages, etc., you can visit Gowin official website.

- LittleBee family: www.gowinsemi.com/en
- Arora family: www.gowinsemi.com/en
- Arora V: https://www.gowinsemi.com/en

Note!

The supported devices may vary according to the software version in use. Please refer to the software you use for more detailed device information.

2.3 Install and Start

The installation method in Windows system is the same; double-click the Gowin Software installation package and install according to the prompt. After installation, the shortcut will be created on the PC desktop by default. The installation method in the Linux system is to decompress the installation file.

You need to configure the license when you start Gowin Software for the first time after installation. The software license is a format contract

between the users and GOWINSEMI to define and limit the rights of users and the obligations of GOWINSEMI.

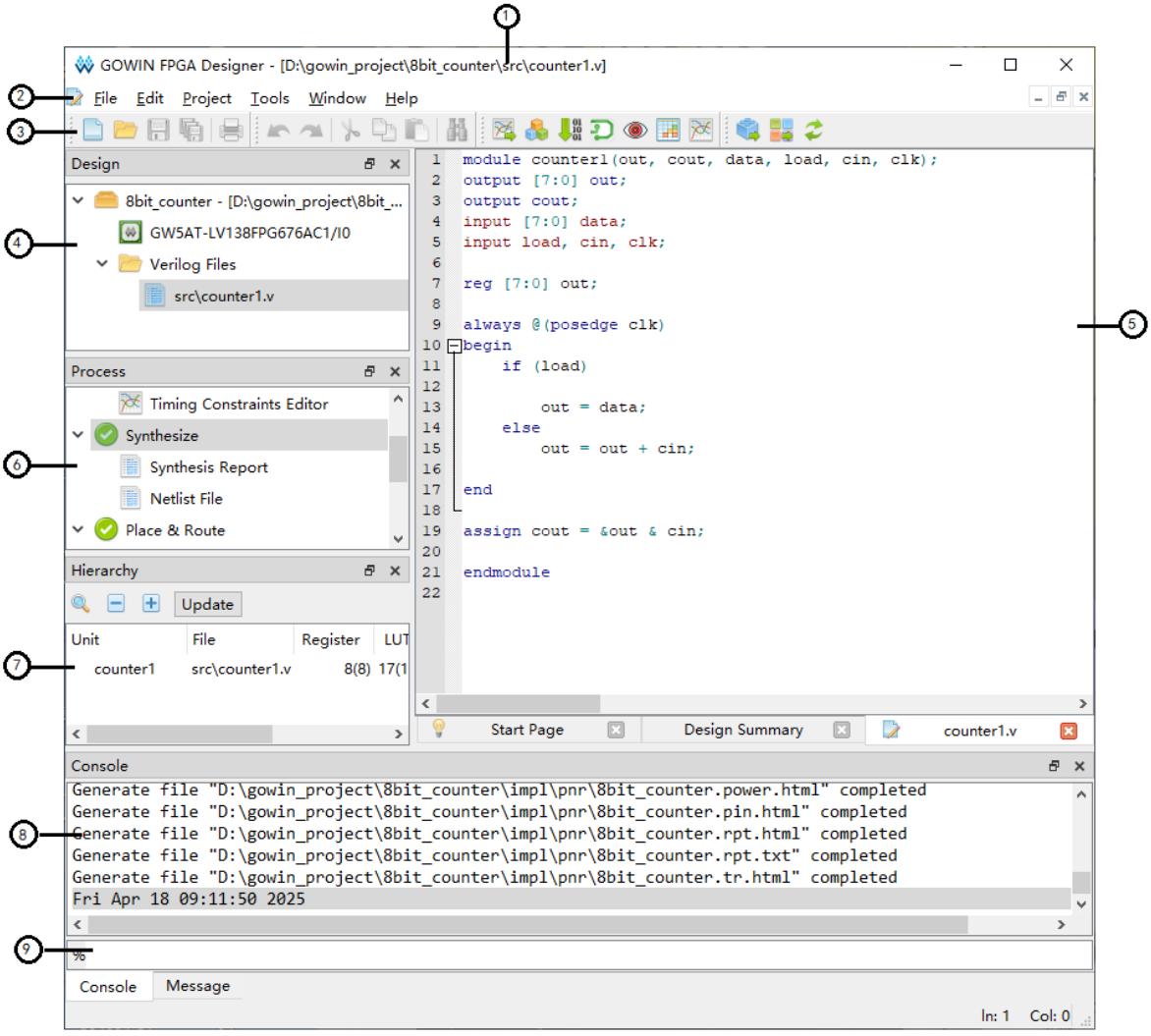
Note!

The installation address of Gowin Software does not support paths containing Chinese. For the details, see *SUG501, Gowin Software Quick Installation Guide*.

3 Gowin Software GUI

Gowin Software GUI is as shown in Figure 3-1. It consists of the title bar, menu bar, tool bar, project area (Design), process area (Process), source file editing area, design hierarchy (Hierarchy), information output area and Tcl command editing area.

Figure 3-1 GUI



①

① Title Bar
② Menu Bar
③ Tool Bar
④ Project Area
⑤ Source File Editing Area
⑥ Process Area
⑦ Design Hierarchy Area
⑧ Information Output Area
⑨ Tcl Command Editing Area

3.1 Title Bar

Title bar shows the name of Gowin Software and the name of the currently opened file.

3.2 Menu Bar

The menu bar contains links to the tools and functionality that are commonly used in projects, including the File, Edit, Project, Tools, Window, and Help options. See the following for details.

3.2.1 File Bar

- Open Example Project…: Open an example project
- New (Ctrl+N): Newly create
- Open (Ctrl+O): Open an item
- Save (Ctrl+S): Save the currently active item
- Save As…: Save the active item using a different file name
- Save All (Ctrl+Shift+S): Save all changed documents
- Close: Close an item
- Close All: Close all changed documents
- Close Project: Close current project
- Print Preview…: Print preview
- Print… (Ctrl+P): Print
- Recent Files: Show the files opened. You can click on the names of these files to re-open.
- Recent Projects: Show the projects opened. You can click on the names of these projects to re-open.
- Exit: Exit and close Gowin Software

3.2.2 Edit Bar

- Undo (Ctrl+Z): Undo your last operation
- Redo (Ctrl+Y): Redo your last operation
- Cut (Ctrl+X): Cut
- Copy (Ctrl+C): Copy
- Paste (Ctrl+V): Paste

- Select All (Ctrl+A): Select all
- Find & Replace (Ctrl+F): Find or replace key words
- Toggle Comment Selection (Ctrl+/): Add comments to the selected
- Increase Indent (Tab): Increase indent
- Decrease Indent (Shift+Tab): Decrease indent
- Macros
  - Start Record: Click Start Record, and the editing operations performed on editable files in the IDE will be recorded.
  - Stop Recording: Stop recording
  - Play Macro (Alt+R): Click Play Macro, and it will perform the recorded operations on the editable files.

3.2.3 Project Bar

- Archive Project: Archive project
- Restore Archived Project: Restore archived project
- Set Device: Set the device information of current project
- Configuration: Open configuration window
- Design Summary: Show details of current project

3.2.4 Tools Bar

- Start Page: Include Recent Projects, Quick Start, Tools, and User Manuals.
  - Recent Projects: Shows the recently opened projects, and up to 10 projects will be kept.
  - Quick Start: Include New Project, Open Project, and Open Example Project.
  - Tools: Include Floorplanner, Timing Constraints Editor, and Programmer.
  - User Manuals: Manuals for LittleBee and Manual for Arora
- Gowin Analyzer Oscilloscope: Gowin analyzer oscilloscope
- Schematic Viewer: HDL design schematic viewer
- IP Core Generator: IP core generator
- Programmer: Programmer
- FloorPlanner: Physical constraints editor
- Timing Constraints Editor: Timing constraints editor
- DSim: Simulation and Verification Cloud Platform
- GoBert: SerDes analysis tool
- Options: Includes Environment, Text Editor, and External Editor.

- Environment: Used to set IDE parameters, including Theme, Language, Toolbar Icon Size, and the default path of new project. here are two options for the theme: Classic and Dark. After setting the theme and language, you need to restart the IDE, then the settings will take effect.
- Text Editor: Used to set the text editor attributes, including font, font size, color, line numbers display, blank characters visualization, the current line and matching parentheses highlight.
- External Editor: Used to set the third-party text editor, and you can choose whether to always use the third-party editor to open the design file.

3.2.5 Window Bar

- Full Screen (F11): Display the IDE GUI in full screen
- Tile: Tile display
- Cascade: Cascade display
- Reset Layout: Restore initial settings
- Panels: Select whether to display the five panels:

  Design, Hierarchy, Process, Message, and Console
- Start Page: Display start page in source file editing
- Design Summary: Display design page in source file editing area, including General and Target Device.
  - General: Project information, including project path and the synthesis tool used.
  - Target Device: Engineering device information, including package, speed grade, and core voltage.

3.2.6 Help Bar

- View Help: View help documents of output information during compilation.
- Contact Us: Click to contact us
- Manage License: Manage license, and you can refer to *SUG501, Gowin Software Quick Installation Guide*.
- About: Show software version and copyright information

3.3 Tool Bar

It provides quick access to some commonly used functions, and from left to right are:

- "  " (Ctrl+N): Create a new file or project
- "  "(Ctrl+O) : Open a file or project
- "  "(Ctrl+S) : Save a file or project

- "  " (Ctrl+Shift+S): Save all files or projects
- "  " (Ctrl+P): Print
- "  " (Ctrl+Z): Undo your last operation
- "  " (Ctrl+Y): Redo your last operation
- "  " (Ctrl+X): Cut
- "  " (Ctrl+C): Copy
- "  " (Ctrl+V): Paste
- "  " (Ctrl+F) : Find
- "  ": Start Gowin Analyzer Oscilloscope; for the details, you can see *SUG114, Gowin Analyzer Oscilloscope User Guide*.
- "  ": Start IP core Generator
- "  ": Start Programmer; for the details, you can see *SUG502, Gowin Programmer User Guide*.
- "  ": Open DSim
- "  ": Start Eye Diagram analysis tool
- "  ": Start FloorPlanner; for the details, you can see *SUG935, Gowin Design Physical Constraints Guide*.
- "  ": Start Timing Constraints Editor; for the details, you can see *SUG940, Gowin Design Timing Constraints Guide*.
- "  ": Run Synthesis
- "  ": Run Place & Route
- "  ": Run Synthesis and Place & Route

3.4 Project Area (Design)

The project area shows projects and the related files. You can check or edit the project device information, user design files, user constraints files, and configuration files, etc.

3.5 Process Area (Process)

The process area provides FPGA design flow, including synthesis, place & route, and Programmer. You can also double-click timing constraints editor and physical constraints editor to edit the constraints files.

3.6 Hierarchy Area (Hierarchy)

After loading the design files, software will parse the design files first.

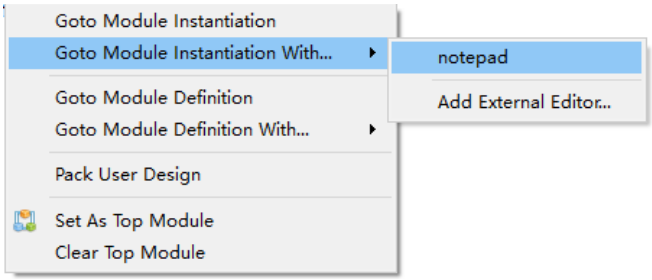
The hierarchy view shows the hierarchy of current project. In Hierarchy view, you can locate the definition and instance of a module in a design file. You can also set a module to top module. The Unit column shows module hierarchy of the design files, and the Files column shows the file where the module definition is. Additionally, the Hierarchy window allows you to search project design files by module name or file name. Currently, the Hierarchy supports parsing Verilog, VHDL, and System Verilog languages.

3.6.1 Right-click Menu

Functions supported in the right-click menu in the Hierarchy view are as follows:

- Goto Module Instantiation: Go to the module instance in the source file and open it with the editor built in Gowin Software. If you configure a third-party editor in "Tools > Options > External Editor" and "Always Use External Editor" is checked, you can open the source with a third-party editor by default via Goto Module Instantiation.
- Goto Module Instantiation With…: Go to the module instance in the source file. As shown in Figure 3-2, you can choose the "notepad" or "Add External Editor". If you select "Add External Editor", "External Editor" dialog box pops up for setting.
- Goto Module Definition: Go to the module in the source file and open it with the editor built in Gowin Software. If you configure a third-party editor in "Tools > Options > External Editor" and "Always Use External Editor" is checked, you can open the source with a third-party editor by default via Goto Module Instantiation.
- Goto Module Definition With…: Go to the module in the source file. As shown in Figure 3-2, you can choose the set editor or "Add External Editor". If you select "Add External Editor", "External Editor" dialog box pops up.
- Pack User Design: Pack the module and its sub-module.
- Set As Top Module: Set the module as top module; the module set as the top will be marked "  " to indicate that the current module is the top module, and the original hierarchy remains.
- Clear Top Module: Clear the top module setting.

Figure 3-2 Right-click Menu of Hierarchy



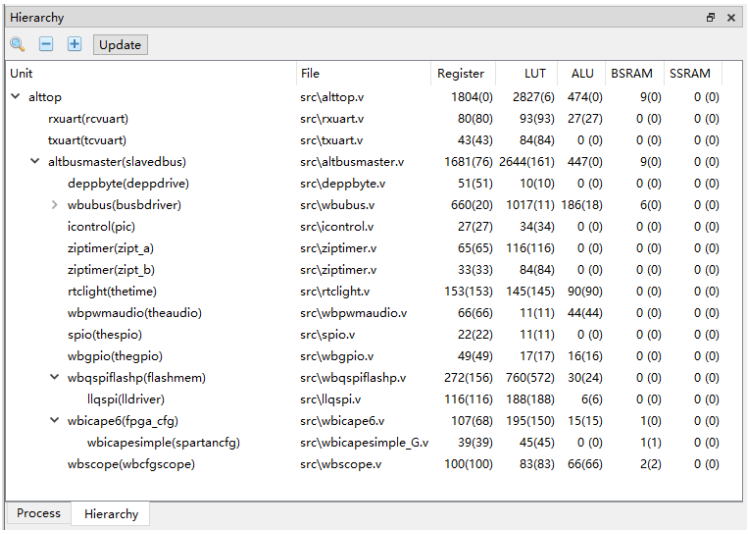
If there is an error during hierarchy parse of the project files, the prompt " RTL Analysis Error" marked in red will pop up at the top right of

the hierarchy view.

3.6.2 Resources Display

The Hierarchy view will automatically display the resources of the current project after synthesis, as shown in Figure 3-3. If a module is defined as a pack module in the design, its resource is not displayed, and its resource will be counted in its upper module. The resources used by each module will be displayed with two data. As shown in Figure 3-3, the LUT resource for module alttop is 2827 (6), where 6 represents the number of LUTs used by the module itself, and 2827 represents the total number of LUTs used by the module and its submodules.

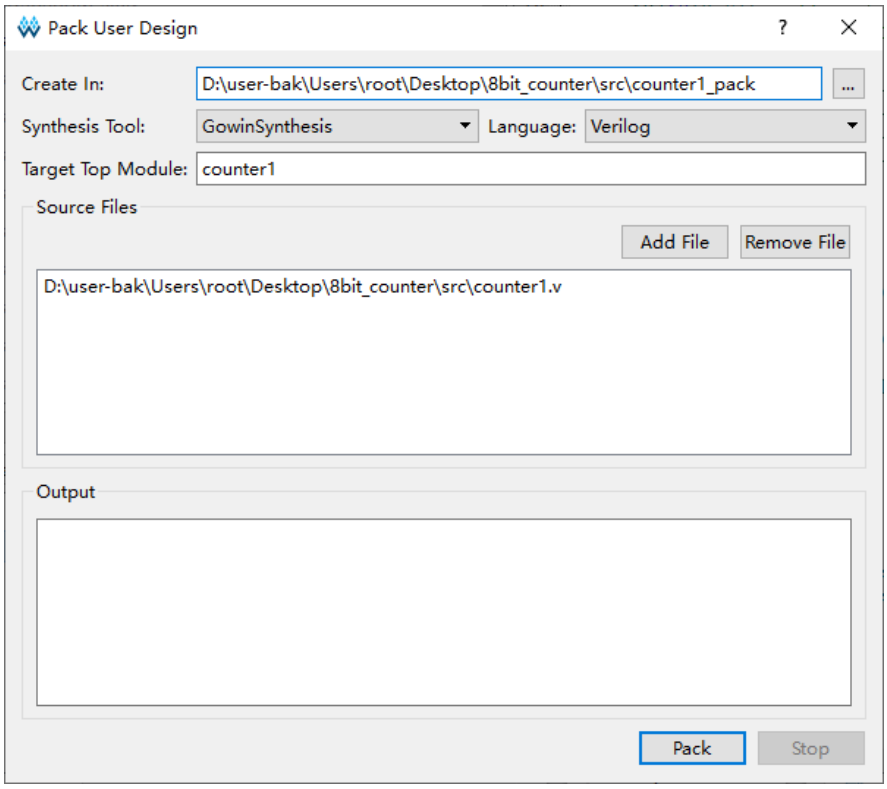
Figure 3-3 Resources Display in Hierarchy View



3.6.3 Pack File

When you open a project, if you need to pack all/some source files, you can right-click the module to be packed in the Hierarchy view, and choose "Pack User Design" to generate a post-synthesis pack file. "Pack User Design" dialog box is as shown in Figure 3-4.

Figure 3-4 Pack User Design Dialog Box



The Pack User Design configurations are as follows:

- Create In: The path of the generated pack files, it supports absolute path only, and the default path is \src\<topmodule_name>_pack.
- Language: Verilog and VHDL, and the default is Verilog.
- Target Top Module: Top module to be packed. The default is the module selected in the Hierarchy view. You can change it.
- Source Files: List the source files of module and sub module selected in the Hierarchy view.
- Add File: Add a file to be packed
- Remove File: Remove a file that do not need to be packed
- Output window: Output information
- Pack: Run Pack
- Stop: Stop Pack

Information will be printed in the Output view, as shown in Figure 3-5. When pack starts, if there is an error, the error will be reported in the Output view, and the information of pack failure will be printed, as shown in Figure 3-6.

Figure 3-5 Output Information in Pack User Design View

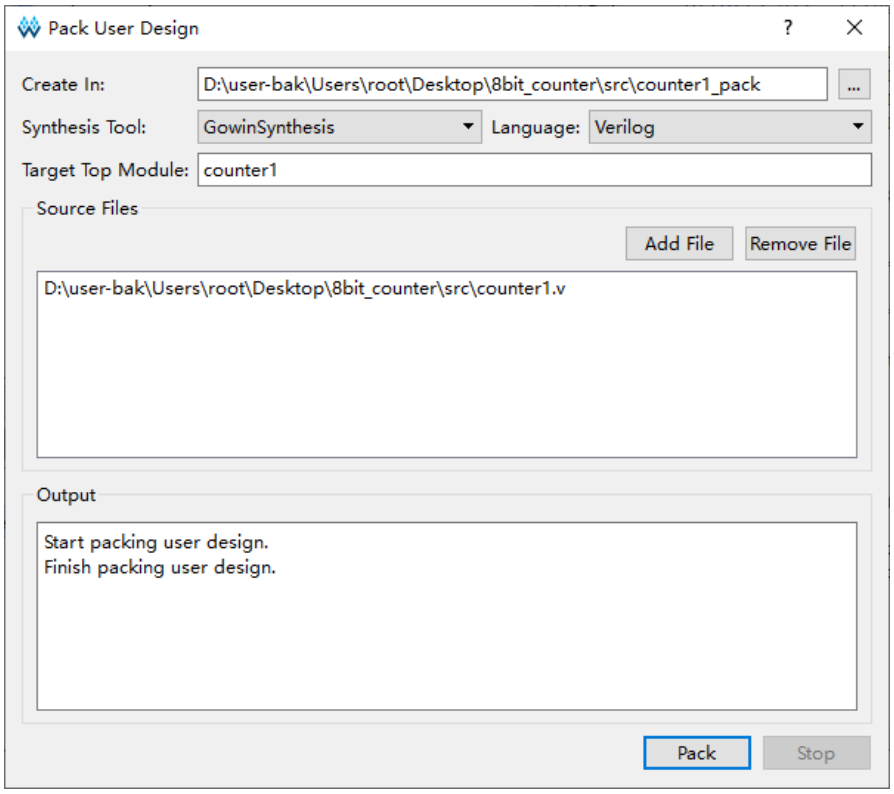
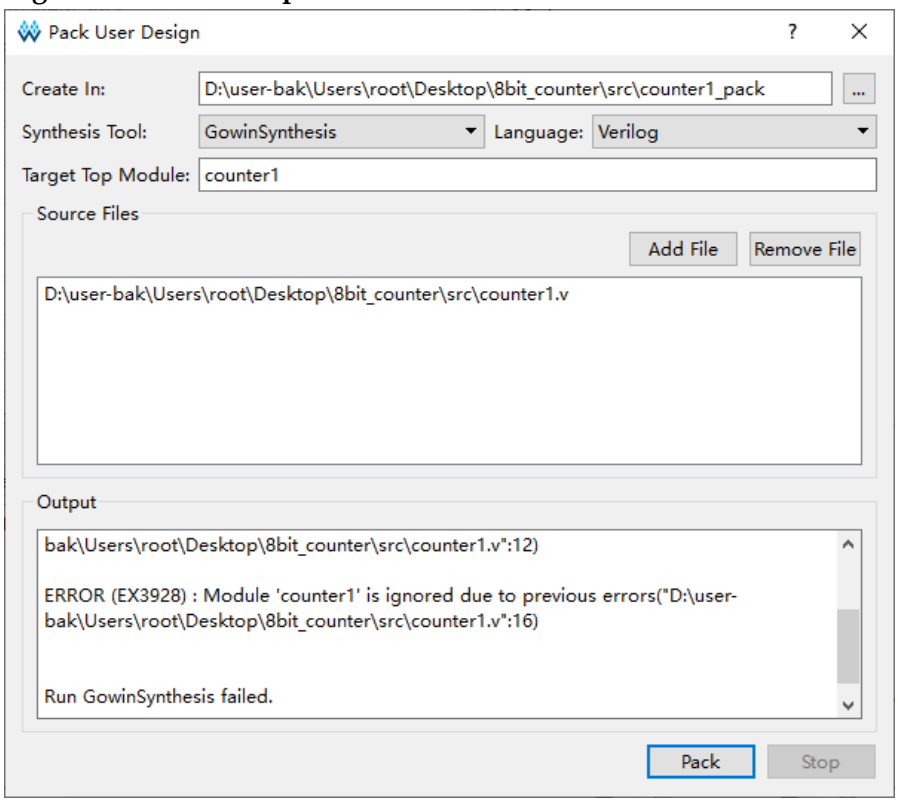


Figure 3-6 Error Prompt



After pack, two files are generated under the target path. If Verilog selected, the two files are <topmodule_name>_gowin.vp and

<topmodule_name>_sim.v. If VHDL selected, the two files are <topmodule_name>_gowin.vhdp and <topmodule_name>_sim.v. <topmodule_name>_gowin.vp and <topmodule_name>_gowin.vhdp are pack file and can be used by others. <topmodule_name>_sim.v is a flattened plain netlist that can be used for internal simulation. <topmodule_name>_sim.v is a flattened plain netlist that can be used for internal simulation.

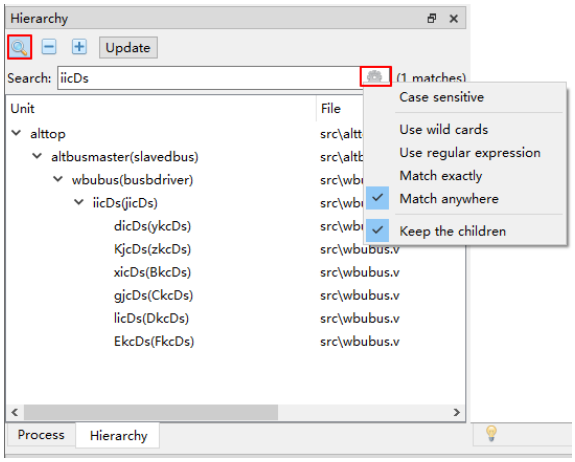
Note!

In the project, if there are multiple modules that instantiate the same sub module, the files generated after packing these modules separately will have the definition of the sub module. If the generated files are used in the same project, it reports an error that the sub module is repeatedly defined, which needs to be avoided.

3.6.4 Search Function

When a project is open and contains many design files, you can click the search icon in the Hierarchy window to search by module name or file name. Various options are available during the search, such as "Use wild cards" or "Match exactly". The dialog box is as shown in Figure 3-7.

Figure 3-7 Search Function



3.7 Source File Editing Area

You can view, edit and highlight source files in the source file edit area.

The source file editing area shows different files, including new files or opened files, the generated files after synthesis or Place & Route, and it also shows "Start Page" and "Design Summary".

If the file is displayed in the editing area and a modification is performed on the file externally, "File Changed" will pop up in the file editing area. Select "Reload" to reload the file.

Click "File > Close", or click the icon " " in the file editing area to close current files.

Click "File > Close All" to close all the files in the file editing area.

Open a file, and you can open "Find & Replace" dialog box by shortcut Ctrl+F or clicking "Find & Replace" in the toolbar. There are three options in "Find All": Current File, Open Files, and Current Project, as shown in Figure

3-8. After clicking "Find All", "Search Result" view will pop up and the search content will be highlighted, and the total number of matches is displayed at the end of the first line, as shown in Figure 3-9.

Figure 3-8 Find & Replace Dialog Box

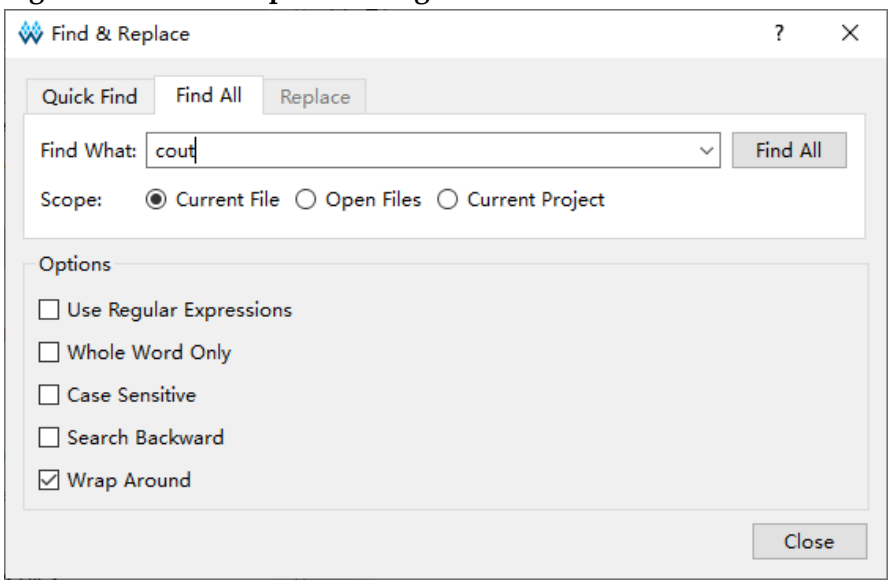
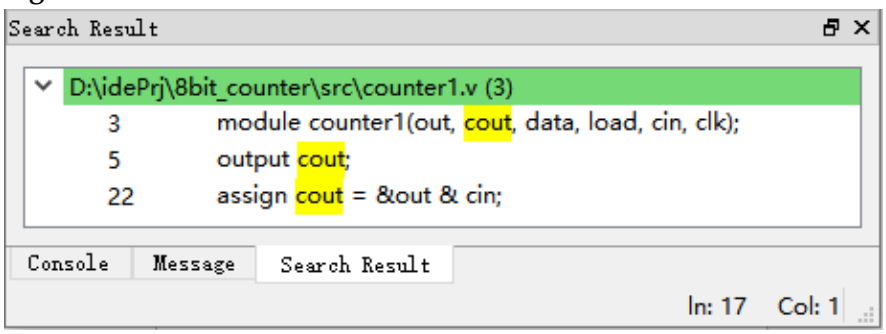


Figure 3-9 Search Result View



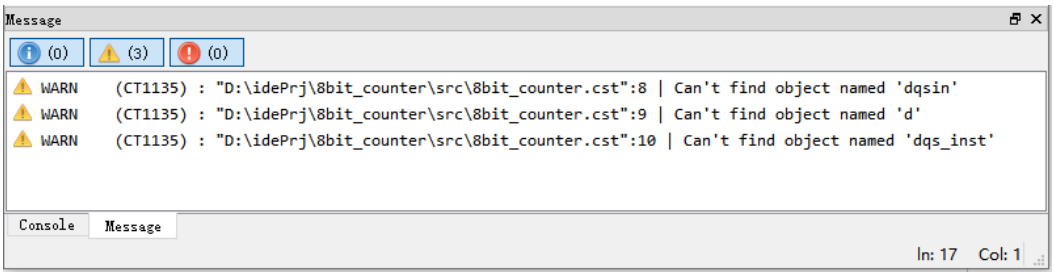
3.8 Information Output Area

The information output area displays the processing information when the software is running. You can view different outputs by manually switching between the tabs:

- Console page includes Tcl commands, warnings, errors, etc.
- Message page includes note, warning, error.

In Console page, right-click and select "Clear" to clear all the information in the tab. You can configure the message page to display note only, warning only or error only. The number of notes, warnings and errors will be recorded and shown on each of the corresponding tabs, as shown in Figure 3-10. Right-click and select "Clear" on "Message" page to clear the page information.

Figure 3-10 Information Output Area



After selecting an Error or Warning message reported by PnR, right-click and select "Help" or press the shortcut key "F1", the "GOWIN Help" of this error or warning will pop up, and the help information of this error or message will be described in details in the document. Some common warnings or errors are shown in Table 3-1, and help documents can be viewed by clicking "Help > View Help", which supports Chinese and English versions.

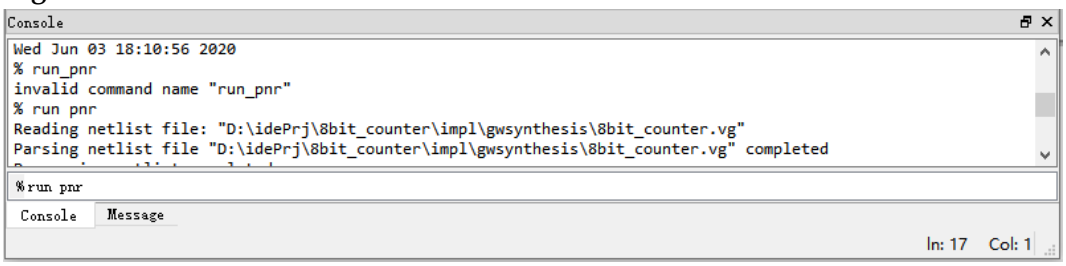
Table 3-1 Common Warnings and Errors

Name	Code	Description
Warning	WARN (PA1002): <file>:<line> | Invalid parameterized value <value>(<parameter>) specified for instance <instanceName>	The specified instance sets invalid parameters.
	WARN (PA1008): <file>:<line> | Object <name> is already defined	The net or port has already been defined.
	WARN (PA1001) : Dangling net <netName>(source:<instanceName>) in module <moduleName> has no destination	The net in the specified module has no destination.
	WARN (CT1098) : <file>:<line> | Group name <name> is already defined	The constraints group name has alredy been defined.
	WARN (CT1101) : <file>:<line> | Location column <number> is out of chip range(<maxColumn>)	The location column is out of chip range.
Error	ERROR (PA2000): <file>:<line> | Syntax error near token <name>	There is an error near token.
	ERROR (PA2001): <file>:<line> | Module <moduleName> is already defined	The module name has already been defined.
	ERROR (PA2017): The number(<value>) of <instType> in the design exceeds the resource limit(<maxValue>) of current device	The number of the devices in the design file exceeds the resource limit.
	ERROR (PA2025): No <instType> resource in current device	There are resources that are not supported by the chip in the design.
	ERROR (PA2054): <file>:<line> | <name> is already declared	The name has already been declared.

The Tcl editing window locates at the bottom of the Console page. You can enter Tcl commands in the window and press the Enter key to execute, as shown in Figure 3-11. For the details of Tcl, please refer to 8 Tcl

Commands.

Figure 3-11 Tcl Commit Window



4 Gowin Software Usage

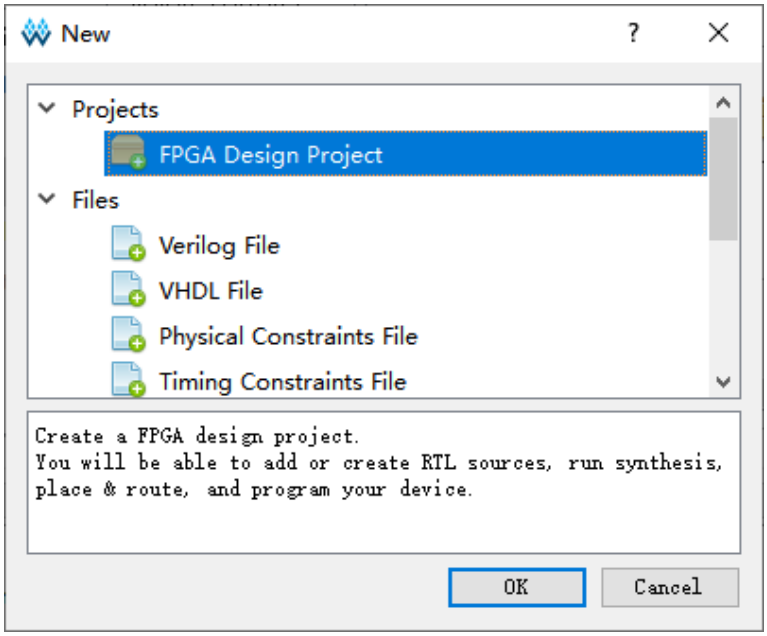
Gowin Software supports interface mode and command line mode. For command line mode, please refer to 8 Tcl Commands.

Take Gowin Software in Windows10 as an example to introduce how to use the software.

4.1 Create a New Project

1. From the File menu, choose "File> New…" to open the "New" dialog, as shown in Figure 4-1.

Figure 4-1 Create a New Project



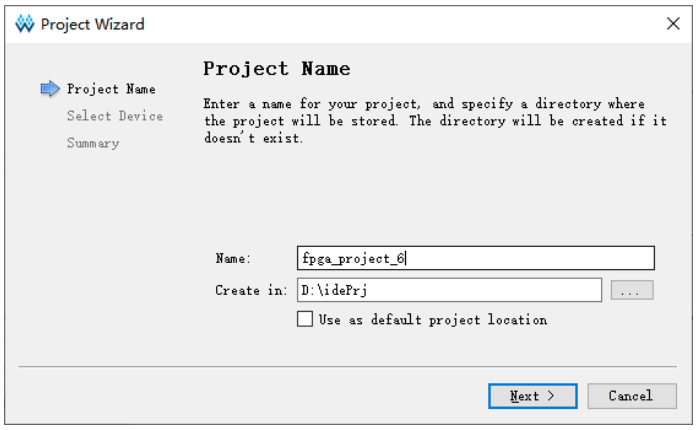
Note!

There are three ways to open a "New" dialog:

- Use the "Ctrl+N" shortcut.
- Click the "New File or Project" icon in the toolbar.
- Click "Quick Start > New Project" on the Start Page.

2. Select "FPGA Design Project", and then click "OK" to open "Project Wizard", as shown in Figure 4-2.

Figure 4-2 Project Wizard



3. Create the project "Name" and "Create in", as shown in Figure 4-2.

   a) Type the project name in the "Name" text box.

   b) Click the "..." icon to choose the project path.

   If you select "Use as default project location", the project location will be set as the default, and all later projects you create will be saved to this location.

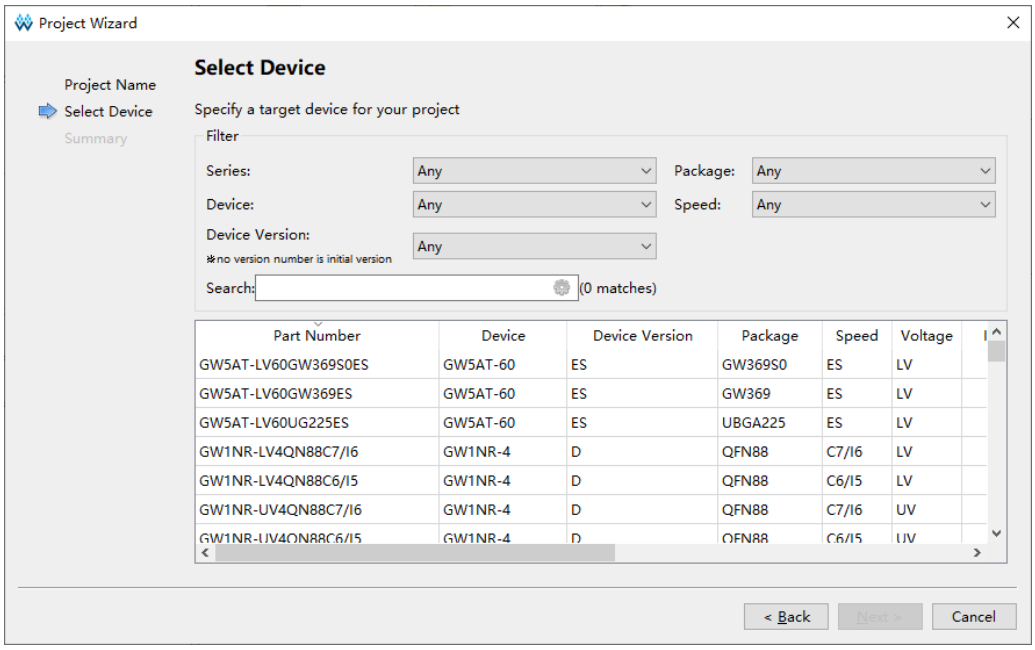
Note!

- The file path length is limited in Windows and Linux, with a limit of 260 characters in Windows and 4096 characters in Linux. If beyond the limit, you cannot delete or copy the file path.
- The path separator is "\" in Windows; for example, E:\Gowin\ide.

4. Click "Next" to set the device, including Series, Device, Package, Speed, and Device Version. You can also search for devices using the "Search" button.

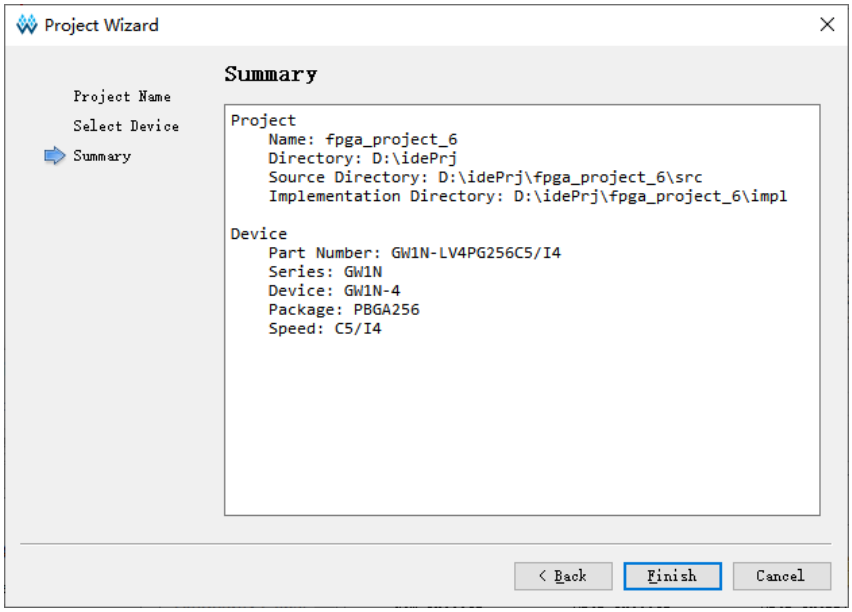
   - You can select series in Series.
   - You can select device in Device.
   - You can select package in Package.
   - You can select speed in Speed.
   - You can select device version in Device Version.
   - You can select part number in Part Number, and it displays the detailed information; for devices without version, the version information column is blank, and the version of the same device is displayed in reverse order, as shown in Figure 4-3.

Figure 4-3 Select Device



5. Click "Next" to open the project information Summary window, as shown in Figure 4-4 .

Figure 4-4 Project Information Summary



6. Click "Finish", and the project now is created.

4.2 Open an Existing Project

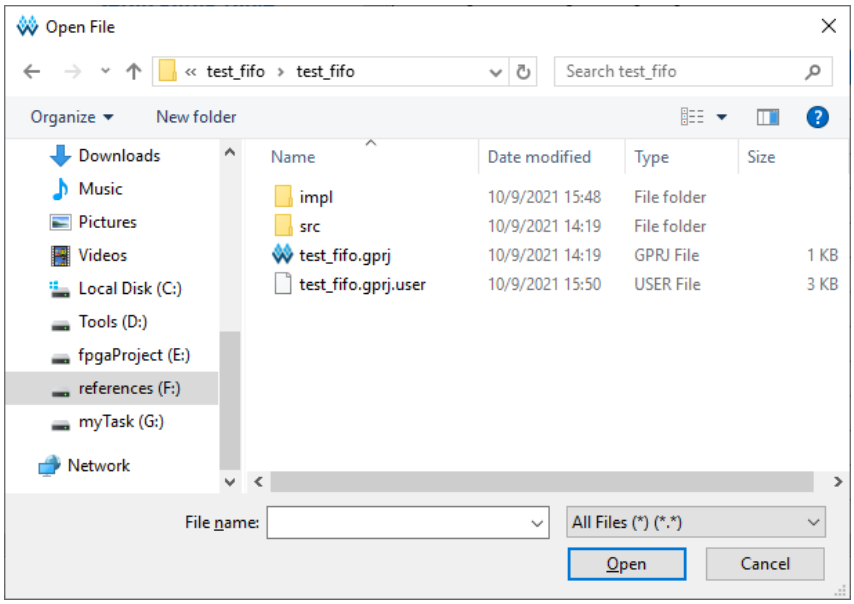
Use one of the following five methods to open an existing project.

Open from Tool Bar

1. You can click the "" icon in the tool bar to open the "Open File" dialog box, as shown in Figure 4-5.

2. Choose the project file (*.gprj) and click "Open" to open it.

Figure 4-5 Open an Existing Project



Open from Menu Bar

1. In the menu bar, select "File > Open …" to open the "Open File" dialog box, as shown in Figure 4-5.
2. Choose the project file (*.gprj) and click "Open" to open it.

Open from Start Page

1. On the start page, click "  " to open "Open Project" dialog box.
2. Choose the project file (*.gprj) and click "Open" to open it.

Open from Recent Projects

In the menu bar, click "File > Recent Projects" to open your required project.

Note!

- You can also open the project in the "Start Page > Recent Projects" list.
- The "Recent Projects" list shows the recently opened projects.
- If the project has been deleted, the "Open Project" dialog box will pop up.

Open from Project File

Find the *.gprj file, and double-click on *.gprj file to open the project automatically.

4.3 Edit a Project

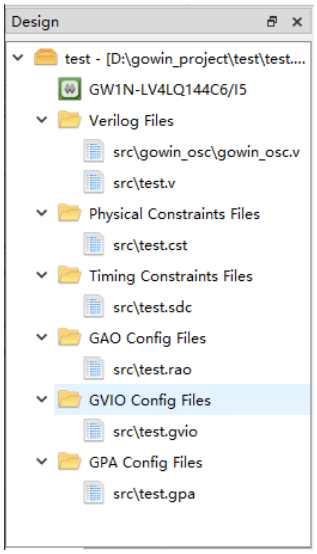
After creating or opening a project, you can edit the device information and the related files in the project design area, as shown in Figure 4-6.

The Project Design Area contains the following:

- The project path

- Part number
- The current project files, including user design files (Source Files), physical constraints files (.cst), timing constraints files (.sdc), GAO config files (.gao, .rao), GPA config files (.gpa), and virtual input and output configuration files (.gvio), etc.

Figure 4-6 Project Design Area



4.3.1 Edit a Project Device

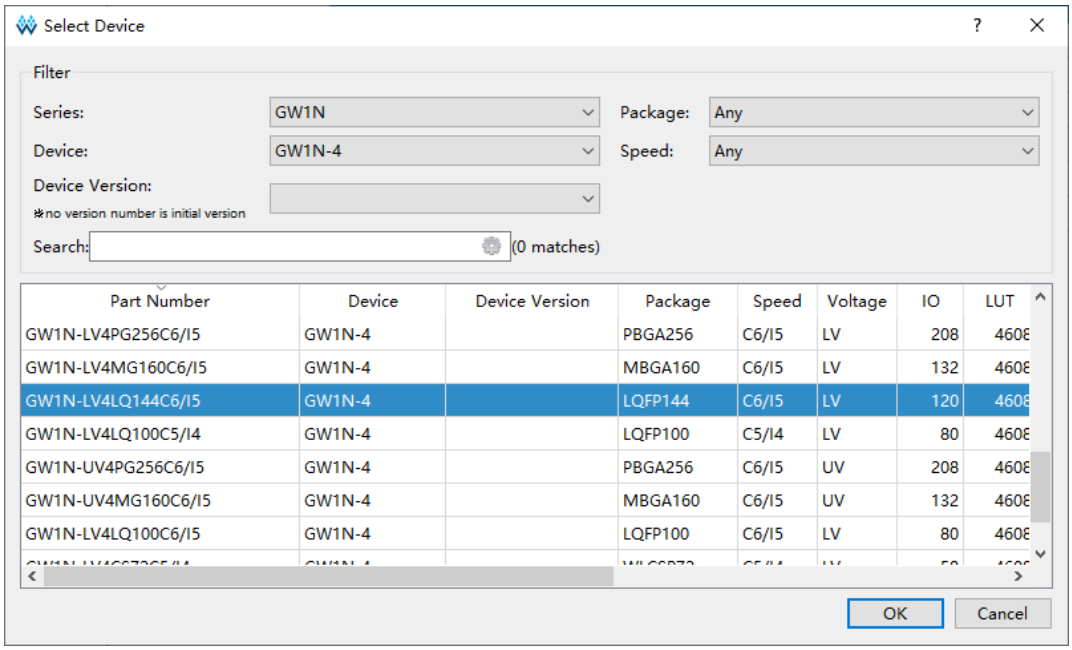
The chip information used in the current FPGA project can be edited in the project design view.

1. As shown in Figure 4-6, double-click "GW1N-LV4LQ144C6/I5" to open the "Select Device" dialog box, or choose "Set Device" from Project pull-down list, as shown in Figure 4-7.
2. In the "Select Device" dialog box, select part number from the "Part Number", then you can edit the device. The "Part Number" column displays detailed information about the selected chip, including device, device version, package, speed, voltage, and the resources such as IO/LUT/FF/SSRAM/BSRAM/User Flash/DSP/PLL in the chip.

Note!

If the Device Version is empty, it means it is the initial version; the device version information will be marked after the date code on the chip, and the device selected in the "Part Number" column needs to be consistent with the device version used.

Figure 4-7 Project Device Info.



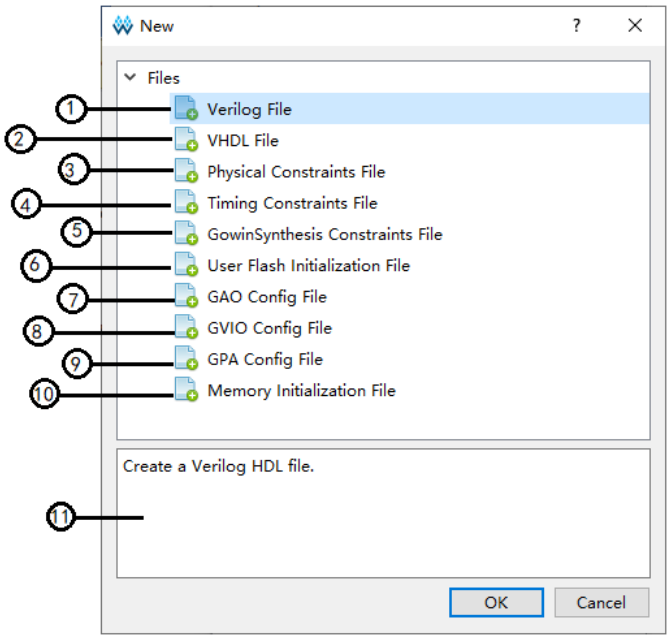
4.3.2 Edit a Project File

Files that need to be added in projects include RTL design files (Source Files), constraints files, and configuration files. Refer to the following descriptions to edit the project files.

Create Design and Constraints Files

1. Click "  " in the tool bar, or select "File > New File…" in the menu bar to open the "New..." dialog box.
2. As shown in Figure 4-8, select a file.

Figure 4-8 Create a New File Dialog Box

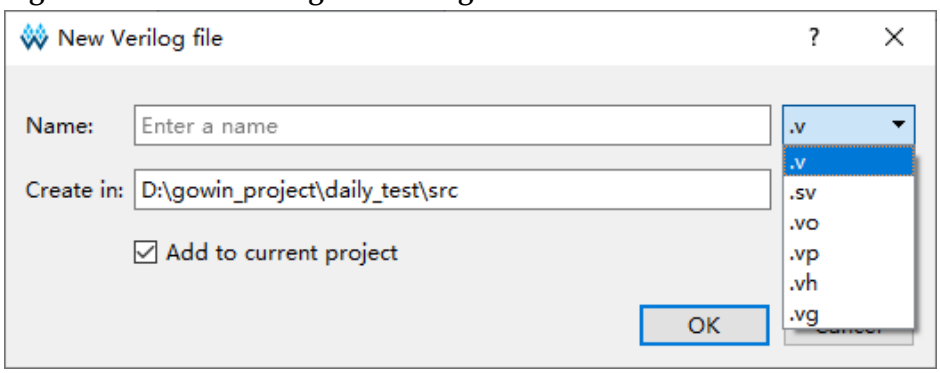


① User Verilog File ② User VHDL File

③ Physical Constraints File
④ Timing Constraints File
⑤ GowinSynthesis Constraints File
⑥ User Flash Initialization file
⑦ GAO Config File
⑧ Virtual Input/Output Config File
⑨ Power Analysis Config File
⑩ Memory Initialization File
⑪ File Description

3. Take creating a Verilog File for an instance. Select "Verilog File" to open the Verilog File dialog box, as shown in Figure 4-9. Check "Add to current project" by default, i.e. the new design file will be added to the current project by default.

Figure 4-9 Create Verilog File Dialog Box



4. Type the file name and click "OK".

Create a Configuration File

1. Click "  " in the tool bar, or select "File > New File…" in the menu bar to open the "New..." dialog box, as shown in Figure 4-8.
2. As shown in Figure 4-8, select a file. Take creating a GPA Config File for an instance. Select "GPA Config File" and to open New GPA Config File dialog box; type the file name, click "OK", and the new GPA profile will be automatically added to the project design area, as shown in Figure 4-10.
3. Double click on this configuration file in the project design area to open a window for editing, as shown in Figure 4-11.

Figure 4-10 New GPA Config File Dialog Box

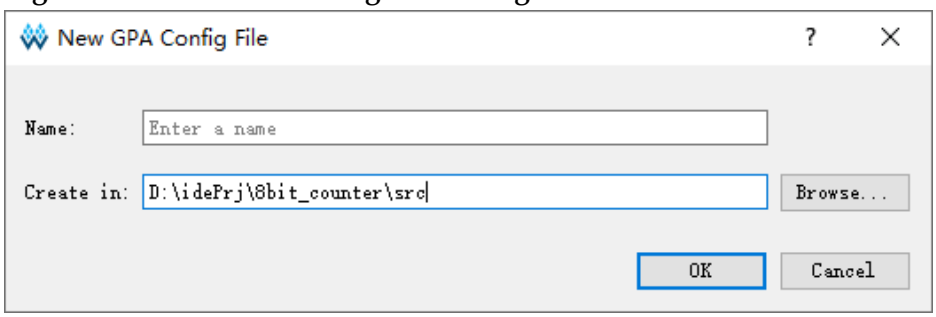
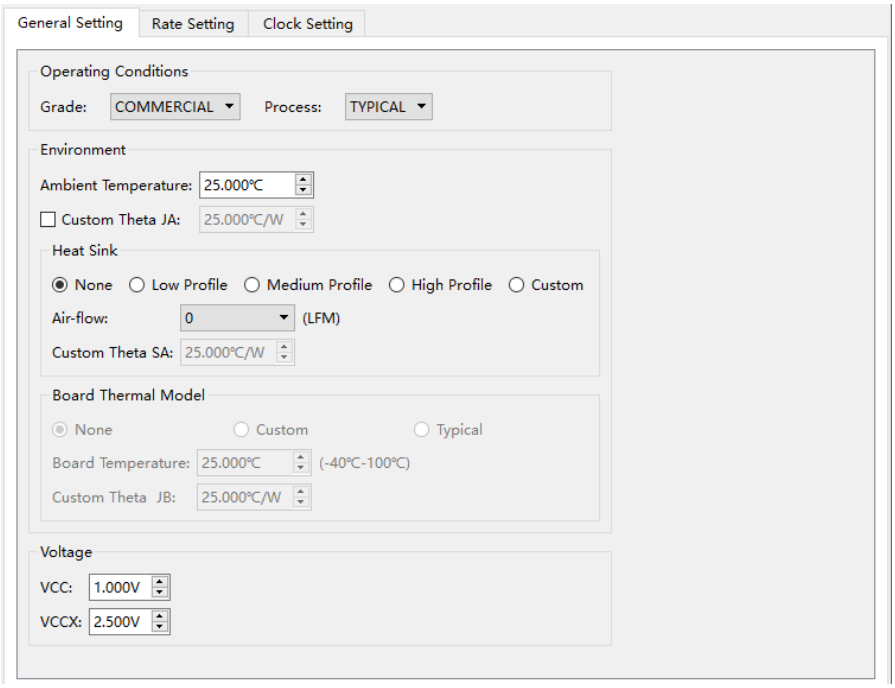


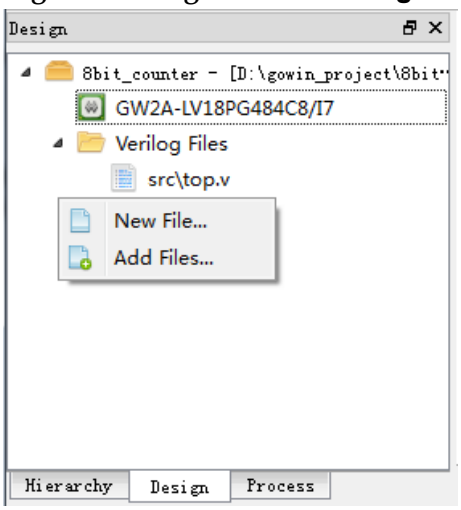
Figure 4-11 GPA Config File Window



Add Project Files

1. As shown in Figure 4-12, right-click in the blank of the project design area, select "Add Files…" to open the "Select Files" dialog box.
2. Select single or multiple project files to add. Gowin Software will automatically classify the files in the project design area. If the added files are not RTL design files, netlist files, constraints files, GPA configuration files, or GAO configuration files, "Other Files" will be added in the project design area.

Figure 4-12 Right-click in Design View

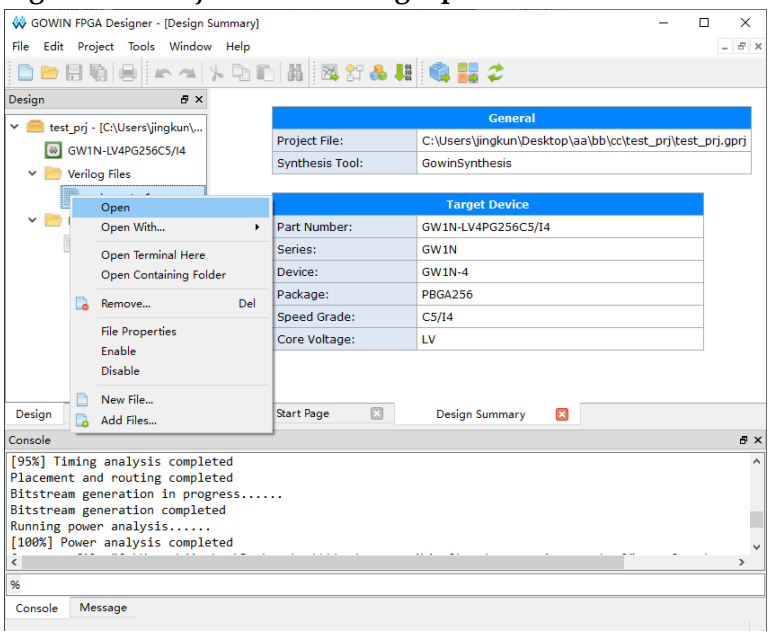


Modify Project Files

Use the following two methods to open the project files, as shown in Figure 4-13.

1. Double-click any file in the project design area; the file will open in the source file editing area.
2. Right-click on the file that is to be modified and click "Open".

Figure 4-13 Project Files Editing Options

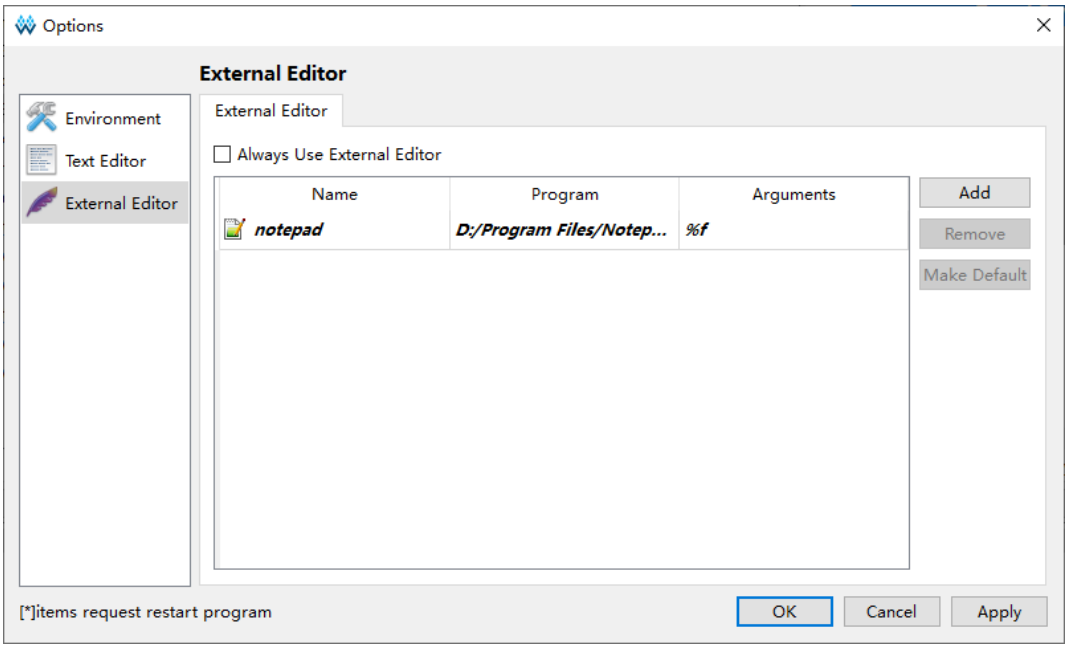


If you have configured a third-party text editor by clicking "Tools > Options", select "Open With…" to open the design file with the third-party text editor. If you select "Add External Editor", you can add other external editor, as shown in Figure 4-14. If you check "Always Use External Editor", the external editor will always be used to open files. If you select "Open Containing Folder", you can open the folder. If you select "Open Terminal Here", you can use command line mode.

If you modify and save a file by an external editor, Gowin Software will generate a reload prompt.

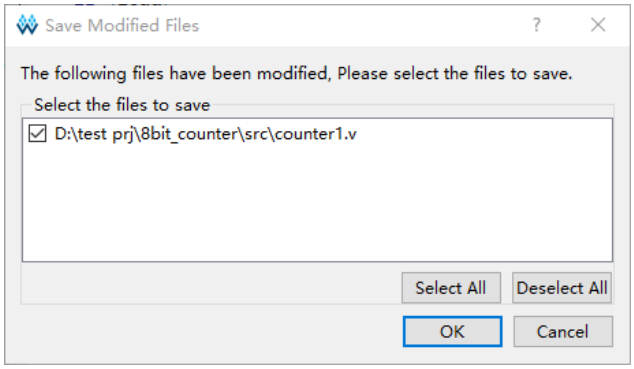
If you close the unsaved file after editing, Gowin Software will pop up a warning.

Figure 4-14 External Editor



After project files are modified, if you run Synthesize or Place & Route before saving these files, the "Save Modified Files" dialog box will pop up, as shown in Figure 4-15.

Figure 4-15 Save Modified Files Dialog Box



Click "OK"; the files will be saved and then perform previous operations automatically. Click "Cancel"; the files will not be saved and Synthesize or Place & Route will not be performed.

Delete Project Files

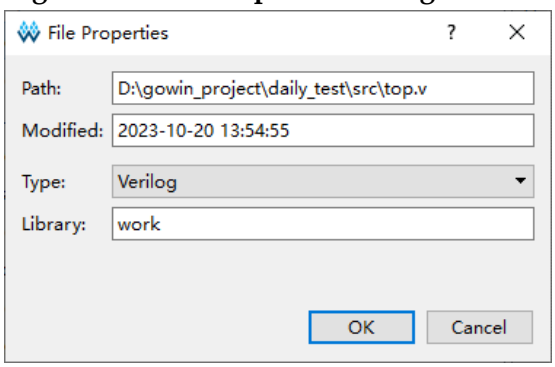
1. Select the file in the project design area.
2. Right-click and select "Remove", or directly press the Delete key to open the "Remove Files" dialog box. If you select the "Remove Permanently on Disk", the file is deleted from the current project and on disk. Otherwise, the file is deleted only from the current project.

Edit Project File Attributes

Right-click any file in the project design area, select File Properties from the right-click list, and "File Properties" dialog box pops up, as shown

in Figure 4-16. The Path, Modified time, Type, and Library information are displayed in the dialog box. The file type can be modified in the Type drop-down list. After clicking "OK", the file will automatically move to the selected type in the Design view. Library is used to specify the library used by VHDL files synthesis, and the default is work. If there are multiple libraries, you need to separate them with semicolon.

Figure 4-16 File Properties Dialog Box



If more than one Verilog file or VHDL file is selected, the "File Properties" dialog box does not display the path and modified time. If both Verilog and VHDL files are selected, the "File Properties" dialog box does not display the path, modified time, or Type.

The use of Library is as follows.

- If the top-level (or upper-level) entity in the design has the component of bottom entity, it does not need to care which library the bottom entity belongs to and it can use the default value work.
- If the top-level (or upper-level) entity in the design use "uut1:entity library name. bottom name", such as uut1:entity mb.sub1 to call the bottom entity, then the attribute library of the vhdl file where the bottom entity is located should be the library name (e.g. mb).
- If the package has a component of the bottom entity, the top-level (or upper-level) entity does not need to care which library the bottom entity belongs to when the bottom entity is called through the package, and it can use the default value work.
- If the package has a component of the bottom entity, the top-level (or upper-level) entity uses "uut: package library name. package name. bottom entity name, such as uut1:work.pack.sub1 to call bottom entity, it does not need to care which library the bottom entity belongs to, and it can use the default value work.

Enable Project Files

You can see the "Enable" and "Disable" options by righting-click on any files in the project design area, as shown in Figure 4-13. The files are in the project compilation process when it is enabled, and out of project compilation process when it is disabled.

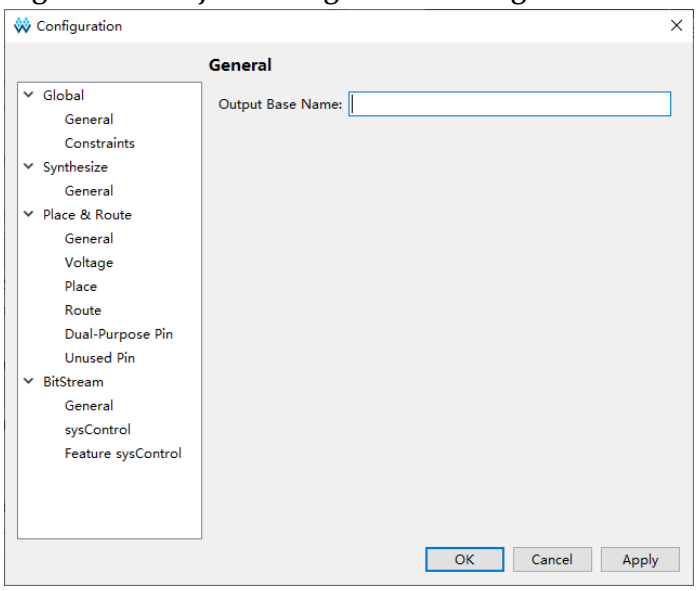
1. Set Enable/Disable by right-clicking, including single/batch files setting.

2. For constraints files or configuration files of the same type, only one file can be enabled; when you create or add a new file of the same type, the previous one will be disabled.

4.3.3 Edit Project Configuration

Right click "Synthesize" or "Place & Route" in the Project Design area to open the project configuration dialog box, as shown in Figure 4-17.

Figure 4-17 Project Configuration Dialog Box



As shown in Figure 4-17, the configurable project options include "Global", "Synthesize", "Place & Route", and "BitStream". The details of the options are as follows.

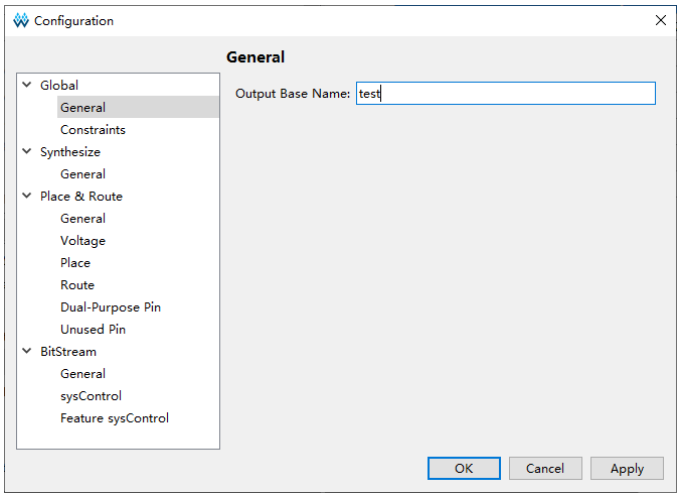
Global

Global configuration option includes General and Constraints.

General

The General configuration is as shown in Figure 4-18; it used to specify the base name of the output file, which defaults to the name of the current project.

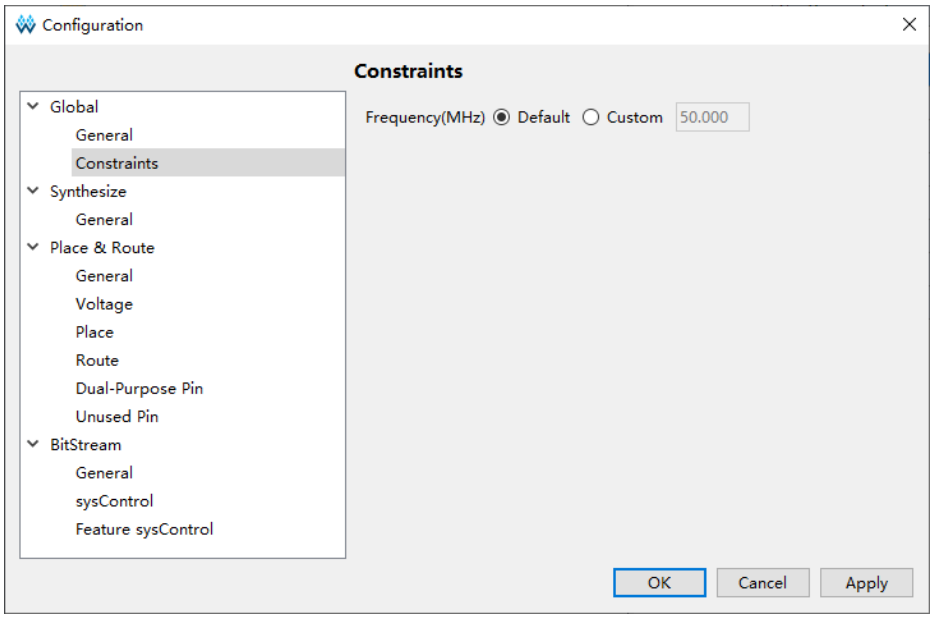
Figure 4-18 General Configuration Option



Constraints

The "Constraints" configuration is shown in Figure 4-19. This option allows users to set global frequency; if the frequency is specified in the timing constraint file, you can override the global frequency with timing constraint. The default is set to "Default" (For LittleBee family, the default is 50MHz, and for Arora family, the default is 100MHz).

Figure 4-19 Frequency(MHz) Configuration Option



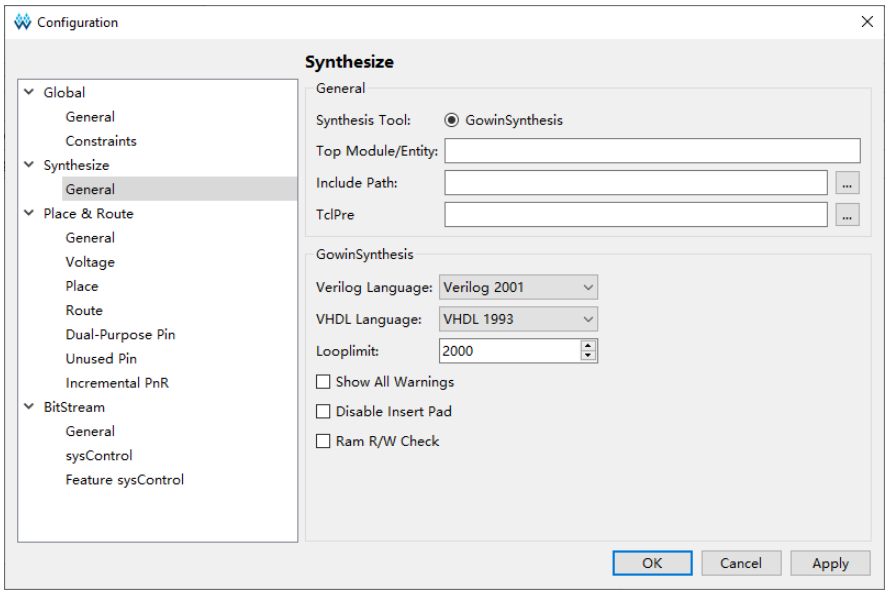
Synthesize

General

General configuration option is as shown in Figure 4-20.

The synthesis tool is GowinSynthesis; hovering the mouse over some options will display their explanations.

Figure 4-20 GowinSynthesis Configuration Options



The detailed descriptions are shown below.

- Top Module/Entity: Specify top module/entity.
- Include Path: Specify include path.
- TclPre: Management file that specifies the software version; when synthesizing, it automatically changes the version number and date, so the running design version can be easily found.
- Verilog Language: System Verilog 2017, Verilog 2001 and Verilog 95, and the default is Verilog 2001. This option is displayed on the interface only when a design file in VHDL format is detected in the current project.
- VHDL Language: VHDL 1993, VHDL 2008, and VHDL 2019; and the default is VHDL1993. The option is displayed in the interface only when VHDL format design files are detected in the current project.
- Looplimit: Set the loop limit value of the default editor in RTL, and the default Value is 2000.
- Show All Warnings: All warnings will be printed during synthesis if this option is checked, which is not checked by default.
- Disable Insert Pad: Whether to insert I/O buffer to the post-synthesis netlist, not checked by default.
- Ram R/W Check: If there is a read or write conflict in RAM, check this option and bypass logic will be inserted around RAM to prevent simulation mismatches. If this option is disabled, bypass logic will not be generated, unchecked by default.

Note!

For the details, see *SUG550, GowinSynthesis User Guide*.

Place & Route

Place & Route includes General, Voltage, Place, Route, Unused Pin and Dual-purpose Pin, and the detailed descriptions are shown in Table 4-1.

Table 4-1 PnR Configuration Options

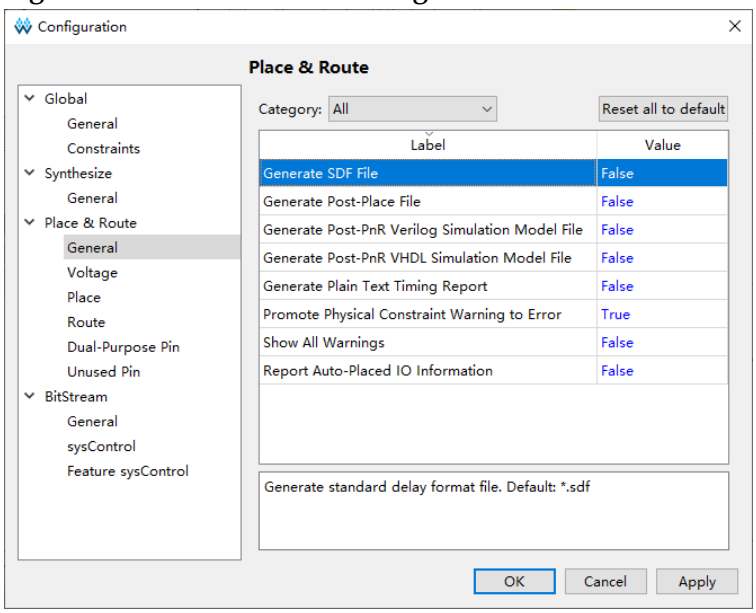
Option	Description
General	Set PnR parameters
Voltage	Set the voltage VCCX
Place	Set placement parameters
Route	Set route parameters
Unused Pin	Used to set different IO atrributes for unused GPIOs
Dual-Purpose Pin	Used to configure the I/O corresponding to the package in the selected device, mainly for configuring dual-purpose pins.

Reset all to default: Reset all configurations on the page to the default values.

General

The General configuration is as shown in Figure 4-21.

Figure 4-21 Place & Route Configuration



The descriptions of options in Figure 4-21 are as follows.

- Generate SDF File: Generate a standard delay format file with the extension .sdf for netlist timing simulation after PnR, and the default value is False. For the usage, see chapter 7 Simulation Files.
- Generate IBIS File: Generate the file specified by the input/output buffer information, with the extension .ibs, and the default is False.

- Generate Post-Place File: Generate a file containing only BSRAM placement information, with the extension .posp, and the default value is False.
- Generate Post-PNR Verilog Simulation Model File: Generate a timing simulation model file in Verilog for timing simulation with .vo extension, and the default value is False.
- Generate Post-PNR VHDL Simulation Model File: Generate a timing simulation model file in VHDL language with .vho extension, and the default value is False.
- Generate Plain Text Timing Report: Generate a timing report in text format, with the extension .tr, and the default value is False.
- Promote Physical Constraint Warning to Error: Promote the physical constraint warning to an error, and the default value is True.
- Show All Warnings: Output all the Warning information when PNR is running, and the default value is False.
- Report Auto-Placed IO Information: Report the location information of auto-placed IO, and the default value is False.

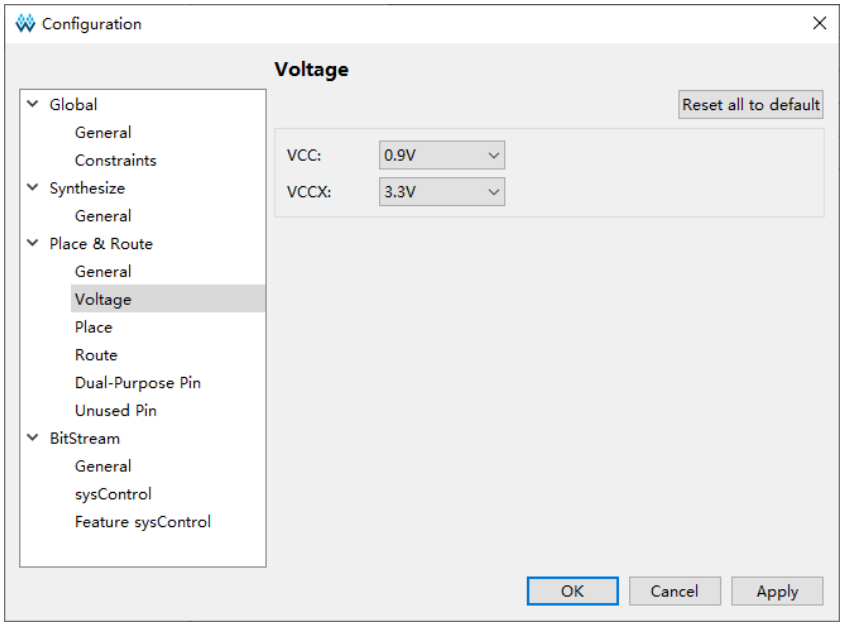
Voltage

The Voltage configuration is shown in Figure 4-23. Through this option, you can set the VCC and VCCX. Different devices may have different VCC and VCCX settings. Click "Reset all to default", and the configured VCC and VCCX will be restored to the default value.

Note!

- The VCC configuration affects the delay data of GW1NZ-1/GW1NZ-2 devices.
- The VCCX configuration affects the results of power consumption calculations.

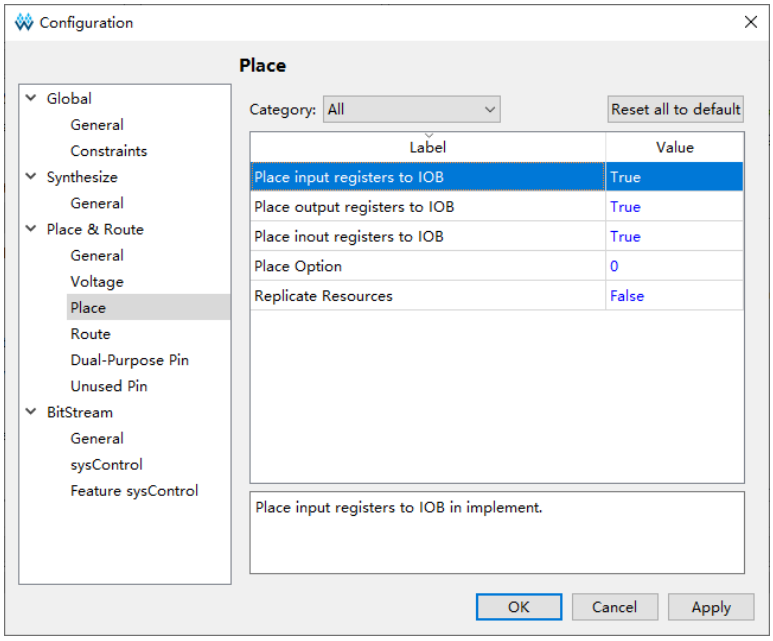
Figure 4-22 Voltage Configuration



Place

The Place configuration is as shown in Figure 4-23.

Figure 4-23 Place Configuration



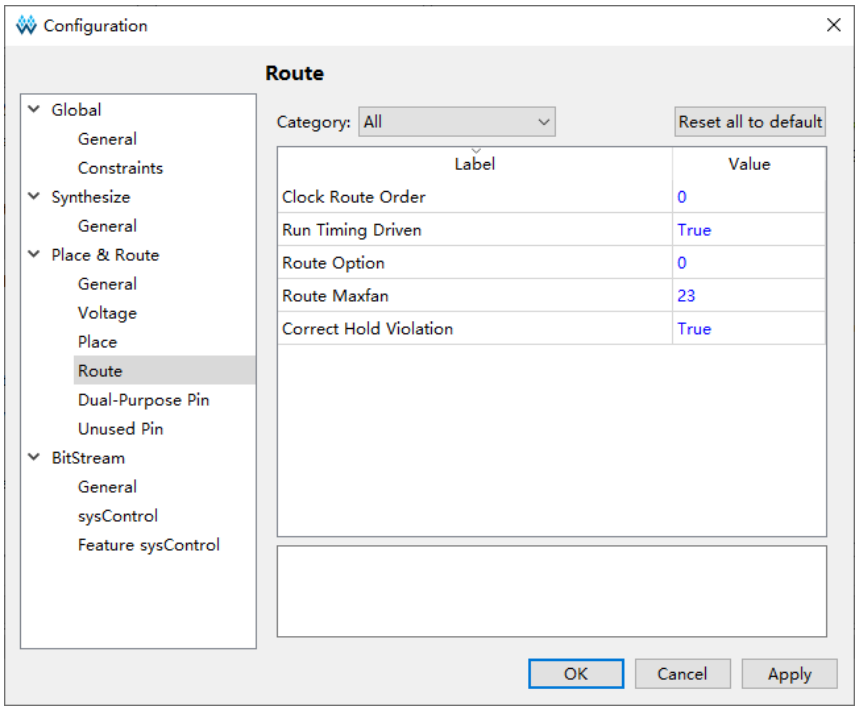
The descriptions of options in Figure 4-23 are as follows.

- Place input register to IOB: Place registers driven by input Buffer to IOB; for GW5A(S)(T)-138/GW5AT-75 devices, the default is False; for the other devices, the default value is True.
- Place output register to IOB: Place registers driven by output/tristate Buffer to IOB; for GW5A(S)(T)-138 devices, the default is False; for the other devices, the default value is True.
- Place inout register to IOB: Place registers driven by in/out Buffer to IOB; for GW5A(S)(T)-138 devices, the default is False; for the other devices, the default value is True.
- Place Option: Placement algorithm option. For GW5A(N)(S)(R)(T) series of devices, the options are 0, 1, 2, 3, and 4. For other series of devices, the options are 0, 1, and 2, and the default value is 0.
- Replicate Resources: Replicate resources with high fanout to reduce fanout and get better timing results, and the default value is False. Only GW5A(N)(S)(R)(T) series support this option, and it will not be displayed on the configuration interface of other devices.

Route

The Route configuration is as shown in Figure 4-24.

Figure 4-24 Route Configuration



The descriptions of options in Figure 4-24 are as follows.

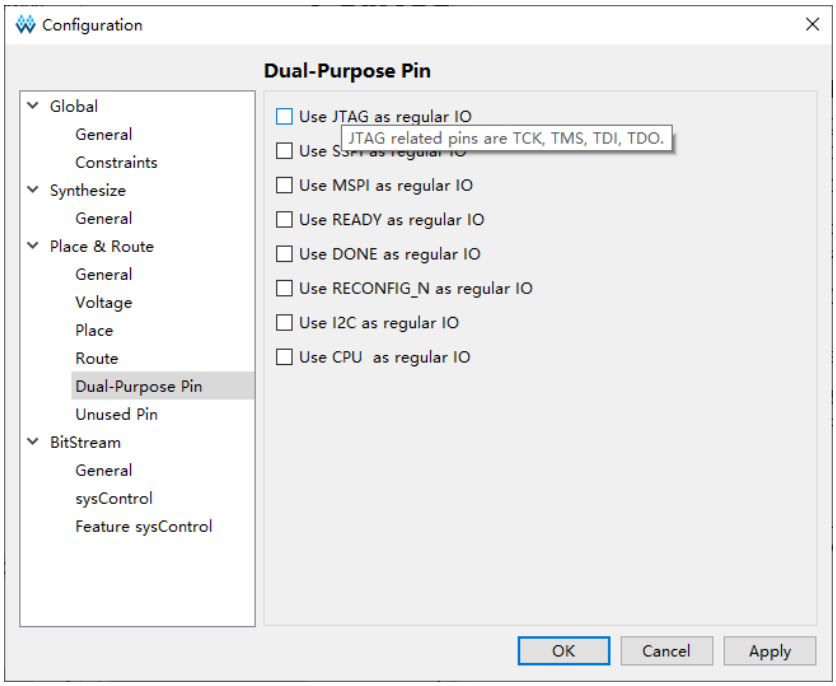
- Clock Route Order: Specify the route order for clock lines other than those generated by the clock primitive, and the values include 0 and 1, and the default is 0.
  - when it is 0, the order is based on the number of fanouts of net from highest to lowest.
  - when it is 1, the order is based on the frequency from highest to lowest.
- Run Timing Driven: Optimize route by Timing Driven, and the default value is True.
- Route Option: Route algorithm with the value of 0, 1 and 2, and the default value is 0.
  - When it is 0, the default route algorithm is used.
  - When it is 1, the compilation speed is sacrificed to try to find a better route.
  - When it is 2, the route speed will be improved.
- Route Maxfan: Based on the route optimization, set the maximum fanout of route. The value should be an integer greater than 0 and less than or equal to 100, and a smaller value may cause route failure. This option does not control long wire and clk route. For GW1NZ-1/GW1N-2/GW1NR-2/GW1N-1P5, the default value of Route Maxfan is 10, and 23 for other devices.
- Correct Hold Violation: Automatic repair of timing Hold problems via

routing, and the default is True.

Dual-Purpose Pin

The Dual-purpose Pin is a configuration that conforms to Gowin device customization, and hovering the mouse over the option will display its explanation. The configuration is as shown in Figure 4-25.

Figure 4-25 Dual-Purpose Pin Configuration



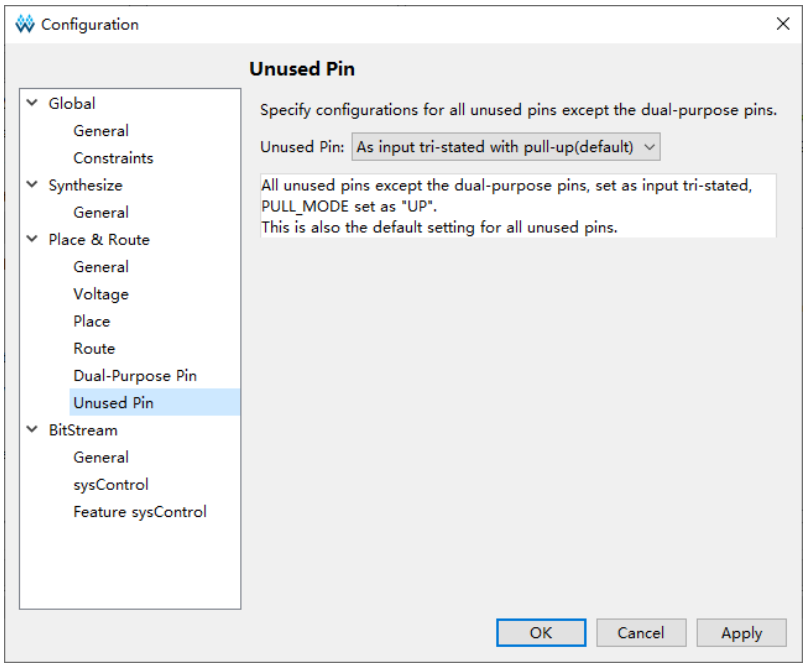
The dual-purpose pins are described as follows.

- Use JTAG as regular IO: Use relevant pins of JTAG as regular IO pins.
- Use SSPI as regular IO: Use relevant pins of SSPI as regular IO pins. For GW5A-25 MBGA121N, this option is checked by default and cannot be changed.
- Use MSPI as regular IO: Use relevant pins of MSPI as regular IO pins.
- Use READY as regular IO: Use the READY pin as regular IO pin.
- Use DONE as regular IO: Use the DONE pin as regular IO pin.
- Use RECONFIG_N as regular IO: Use the RECONFIG_N pin as regular IO pin.
- Use I2C as regular IO: Use relevant pins of I2C as regular IO.
- Use CPU as regular IO: Only GW5A(N)(S)(R)(T) series support this option. Use relevant pins of CPU as regular IO.

Unused Pin

The Unused Pin option allows users to set different IO attributes for unused GPIOs. There are two options: "As input tri-stated with pull-up (default)" and "As open drain driving ground", as shown in Figure 4-26.

Figure 4-26 Unused Pin Configuration

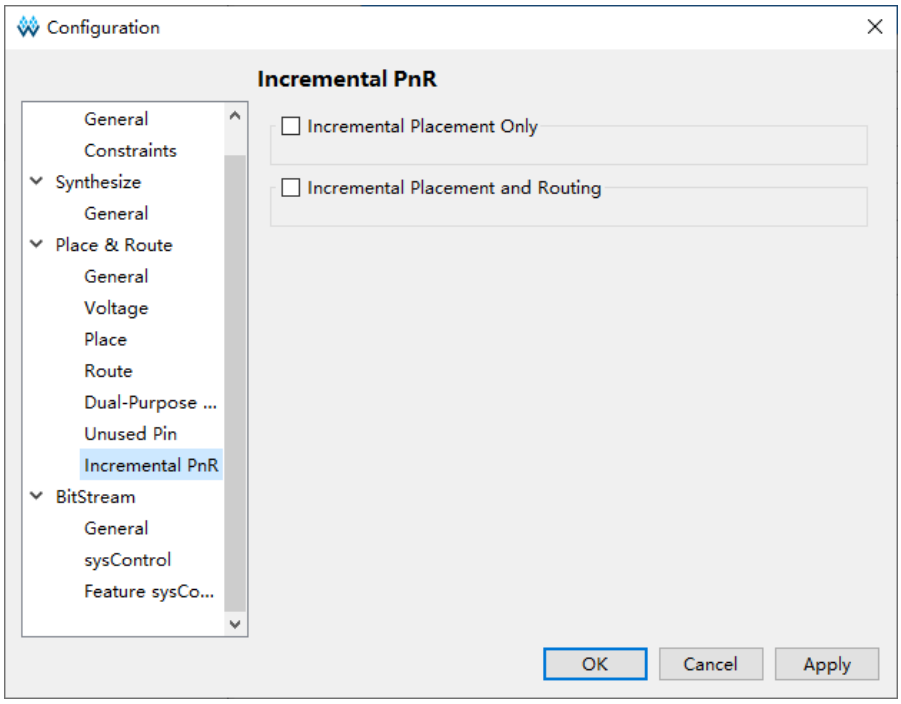


- As input tri-stated with pull-up (default): Default option; all unused GPIOs will be configured as input tri-stated with weak pull-up.
- As open drain driving ground: All unused GPIOs are configured to output with OPEN_DRAIN ON.

Incremental PnR

The Incremental PnR can be used to enable incremental compilation, which can reuse the placement or routing results from the previous running, thereby reducing the time spent on re-PnR, and improving overall efficiency. The option configuration is as shown in Figure 4-27.

Figure 4-27 Incremental PnR Configuration



The descriptions for the options in Figure 4-27 are as follows:

Incremental Placement Only: Only increments placement information. By default, this option is unchecked. When it is checked, the following sub-options are displayed:

- Auto: Automatically select the existing incremental placement file (*.p) in the project directory when redoing the placement. If no *.p file exists in the project directory, a prompt will pop up.
- Specify the previous placement file: Manually specify the incremental placement file (*.p).

Incremental Placement and Routing: Increments placement and routing information. By default, this option is unchecked. When it is checked, the following sub-options are displayed, and the "Incremental Placement Only" option becomes unavailable.

- Auto: Automatically select the existing incremental placement and routing file (*.pr) in the project directory when redoing placement and routing. If no *.pr file exists in the project directory, a prompt will pop up.
- Specify the previous placement and routing file: Manually specify the incremental placement and routing file (*.pr).

BitStream

You can configure the bitstream file format and frequency, etc. via BitStream. Hovering the mouse over the option will display its explanation. BitStream option includes General, sysControl, and Feature sysControl, and the descriptions are as shown Table 4-2.

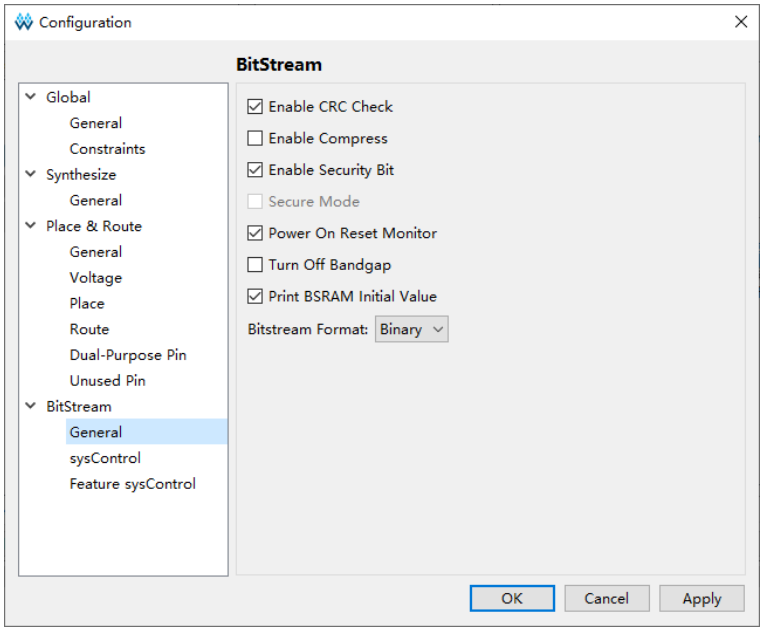
Table 4-2 BitStream Configuration Options

Option	Description
General	Set BitStream parameters
sysControl	Set BitStream system control parameters
Featrue sysControl	Set BitStream functional system control parameters

General

The General configuration is as shown in Figure 4-28.

Figure 4-28 Bitstream General Configuration



The description for each parameter in Figure 4-28 is as follows.

- Enable CRC Check: Enable CRC Check, checked by default.
- Enable Compress: Enable bitstream file compress, not checked by default.
- Enable Encryption: Encrypt the bitstream file, and only Arora Family devices support this option; for other devices, this option is not displayed on the interface, not checked by default.
- Key (Hex): Key (Hex) can be configured only when "Enable Encryption" is checked, and users can customize the secret key; only Arora Family devices support this option; for other devices, this option is not displayed on the interface. The key is 0 by default. When you check this option and run Place &Route, a key file with the extension .ekey will be generated.
- Enable Security Bit: Enable security bit, and add security bit to the bitstream file; after adding, the bitstream cannot be read back again, checked by default.
- Secure Mode: Enable secure mode. Use JTAG pin as GPIO, and

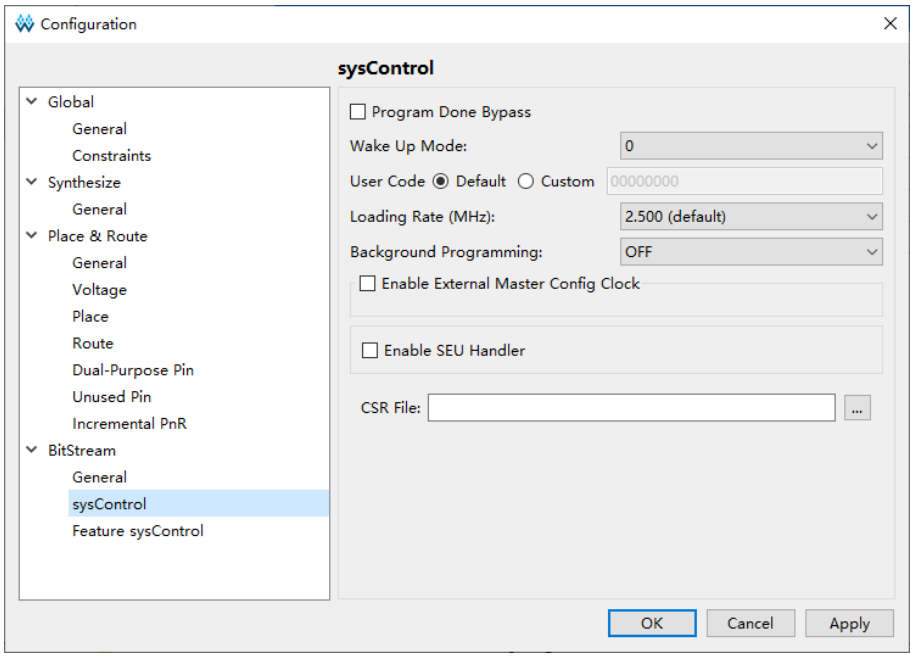
device can be programmed only once. This function is only supported by GW1NSER-4C, unchecked by default.

- Power On Reset Monitor: Power on reset monitoring, checked by default. When this option is checked, it will continuously monitor any possible voltage drop in the power rails. If the power rail voltage falls below the POR threshold, all RAM bits will be cleared and the used I/Os will be set to tri-state through internal weak pull-up resistors, and then the configuration and initialization will be completed in turn.
- Turn Off Bandgap: Turn off Bandgap, unchecked by default. Bandgap is used to provide constant voltage and current for some modules in the chip; turning off Bandgap can reduce device power. Only GW1N-1 supports this option; for the other devices, this option will not be displayed on the configuration interface.
- Print BSRAM Initial Value: Print BSRAM initial value to the bitstream file, checked by default. For GW1N and GW2A series of devices, after checking this option, it prints the initial values of all BSRAM locations to the bitstream file, and the initial values of BSRAM locations that are not occupied are printed as 0. For GW5A(N)(S)(R)(T) devices, after checking this option, it prints the initial values of all BSRAMs within the columns where utilized BSRAMs are located to the bitstream file, and the initial values of the BSRAM locations within that column that are not utilized will be printed as 0.
- Bitstream Format: Used to specify the bitstream file format, Text and Binary, and Binary by default. When the Text is selected, the *.fs file in plain text format is generated; when the Binary is selected, a bitstream file in *.fs, *.bin, and *.binx formats is generated. *.bin and *.binx are bitstream files in binary format, *.binx file contains header annotation information, and *.bin does not have header annotation information.

sysControl

The sysControl configuration is as shown in Figure 4-29.

Figure 4-29 sysControl Configuration



The description for each parameter in Figure 4-29 is as follows.

- Program Done Bypass: When the Done Final signal is active, the external Done signal keeps low, so that the new bitstream can be forwarded after the bitstream is loaded.
- Wake Up Mode: Enable wake up mode, with the value of 0 and 1, and the default is 0.
  - When wake up mode is 0, DONE pin can be pulled up or down.
  - When wake up is 1,
    a) If DONE pin is pulled up, download and run normally.
    b) If DONE pin is pulled down, download normally; and DONE pin needs to be pulled up and keep TCK connected to the pulse signal to wake up chip.
- User Code: Users can customize User Code, and the defined value will be reflected in the generated bitstream file, and the User Code will be checked when the bitstream file is downloaded through the Programmer. The default value is Default (00000000).
- Loading Rate: In AutoBoot mode and MSPI mode, the rate of loading bitstream from Flash to SRAM. For GW1N-4/GW1NRF-4B/GW1NR-4, the default is 2.100MHz; for GW1NS-4/GW1NSR-4/GW1NSER-4C with speed grade C7/I6, the default is 2.6MHz; for GW5A(S)(T)-138/GW5A(R)-25, the default is 35.000MHz; for the other devices, the default is 2.500MHz. For the details, see *UG290, Gowin FPGA Products Programming and Configuration Guide*; *UG714, Arora V 25K FPGA Product Programming and Configuration Guide*; *UG704, Arora V 138K FPGA Product Programming and Configuration Guide*. The loading rate and calculation method of different devices are

different.

- The loading rate of following packages only supports 2.500MHz.
  a) GW1N-2: LQFP100X/LQFP144X/MBGA132X/WLCSP42H/MBGA49
  b) GW1N-2 Version B: LQFP100X/LQFP144X/MBGA132X/MBGA121X
  c) GW1N-2 Version C: LQFP100X/LQFP144X/MBGA132X/MBGA121X /MBGA49/QFN32X
  d) GW1NR-2: MBGA49P/MBGA49PG/MBGA49G
  e) GW1NR-2 Version B: MBGA49P/MBGA49PG/MBGA49G
  f) GW1NR-2 Version C: MBGA49P/MBGA49PG/MBGA49G
  g) GW1N-1P5: LQFP100X
  h) GW1N-1P5 Version B: LQFP100X/QFN48X
  i) GW1N-1P5 Version C: LQFP100X/QFN48X
- The loading rate and calculation of following devices is as shown in Table 4-3.
  a) GW1NZ-1
  b) GW1N-2/GW1N-1P5/GW1NR-2 except the above packages supporting 2.500MHz
  c) GW1NSER-4C/GW1NS-4/GW1NSR-4/GW1NS-4C/ GW1NSR-4C except the part number with speed grade C7/I6
  d) GW1N-9/GW1NR-9
  e) GW2A-18/GW2AR-18/GW2ANR-18(Version C)
  f) GW2A-55/GW2AN-55(Version C)

Table 4-3 Loading Rate and Formula (1)

Loading Rate (MHz)	Formula
2.500 (default)	250 / 100
5.435	250 / 46
5.682	250 / 44
5.952	250 / 42
6.250	250 / 40
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

Loading Rate (MHz)	Formula
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

- The loading rate and calculation of following devices is as shown in Table 4-4.

  GW1N-1/GW1N-1S/GW1NR-1

Table 4-4 Loading Rate Value and Formula (2)

Loading Rate (MHz)	Formula
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
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

Loading Rate (MHz)	Formula
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

- The loading rate and calculation of following devices is as shown in Table 4-5.
  a) GW2AN-9X
  b) GW2AN-18X

Table 4-5 Loading Rate and Formula (3)

Loading Rate (MHz)	Formula
2.500 (default)	200 / 80
1.562	200 / 128
1.587	200 / 126
1.613	200 / 124
1.639	200 / 122

Loading Rate (MHz)	Formula
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
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

Loading Rate (MHz)	Formula
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

- The loading rate and calculation of following devices is as shown in Table 4-6.

  GW1N-4/GW1NRF-4B/GW1NR-4

Table 4-6 Loading Rate and Formula (4)

Loading Rate (MHz)	Formula
2.100 (default)	210 / 100
4.565	210 / 46
4.773	210 / 44
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

Loading Rate (MHz)	Formula
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

- The loading rate and calculation of following devices is as shown in Table 4-7.

  GW1NSER-4C/GW1NS-4/GW1NSR-4/GW1NS-4C/ GW1NSR-4C with speed grade C7/I6

Table 4-7 Loading Rate and Formula (5)

Loading Rate (MHz)	Formula
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
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

Loading Rate (MHz)	Formula
26.000	260 / 10
32.500	260 / 8
43.333	260 / 6
65.000	260 / 4

- The loading rate and calculation of following devices is as shown in Table 4-8.

  GW5A(S)(T)-138/GW5A(R)-25/GW5AT-75

Table 4-8 Loading Rate and Formula (6)

Loading Rate (MHz)	Formula
35.000(default)	210 / 6
52.500	210 / 4
70.000	210 / 3
105.000	210 / 2

- The loading rate and calculation of following devices is as shown in Table 4-9.

  GW5AT-60/GW5A(N)(R)T-15

Table 4-9 Loading Rate and Formula (7)

Loading Rate (MHz)	Formula
2.500 (default)	210 / 84
1.667	210 / 126
1.694	210 / 124
1.721	210 / 122
1.750	210 / 120
1.780	210 / 118
1.810	210 / 116
1.842	210 / 114
1.875	210 / 112
1.909	210 / 110
1.944	210 / 108
1.981	210 / 106
2.019	210 / 104
2.059	210 / 102
2.100	210 / 100
2.143	210 / 98
2.188	210 / 96

Loading Rate (MHz)	Formula
2.234	210 / 94
2.283	210 / 92
2.333	210 / 90
2.386	210 / 88
2.442	210 / 86
2.561	210 / 82
2.625	210 / 80
2.692	210 / 78
2.763	210 / 76
2.838	210 / 74
2.917	210 / 72
3.000	210 / 70
3.088	210 / 68
3.182	210 / 66
3.281	210 / 64
3.387	210 / 62
3.500	210 / 60
3.621	210 / 58
3.750	210 / 56
3.889	210 / 54
4.038	210 / 52
4.200	210 / 50
4.375	210 / 48
4.565	210 / 46
4.773	210 / 44
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

Loading Rate (MHz)	Formula
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
70.000	210 / 3
105.000	210 / 2

- Background Programming: You can re-program Flash without interrupting the current FPGA running. If background programming value of the device is only OFF, the configuration option will not be displayed on the configuration interface.

Figure 4-30 Select I2C

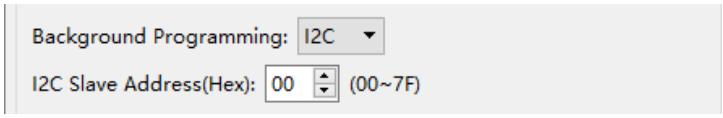
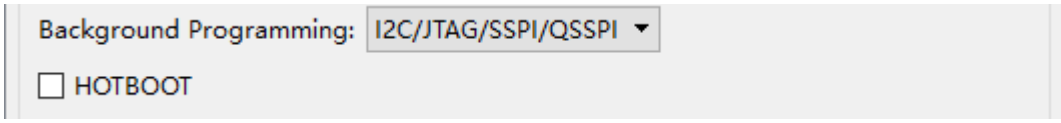


Figure 4-31 Select I2C/JTAG/SSPI/QSSPI



The devices that support background programming and their values are shown in Table 4-10.

Table 4-10 Background Programming Value

Device	Background Programming Value
● GW1N-1P5/GW1N-2/GW1NR-2 ● Version B: GW1N-4/GW1NR-4, GW1NRF-4 ● Version D: GW1NR-4 ● GW1NS-4/GW1NSR-4 ● GW1N-9/GW1NR-9 ● GW1NZ-1	OFF, JTAG; OFF by default
Version B: GW1N-1P5/GW1N-2/GW1NR-2	OFF, JTAG, I2C; OFF by default
Version C: GW1N-2/GW1NR-2/GW1N-1P5	OFF, GoConfig, GoConfig Mode1, JTAG, I2C; OFF by default
GW2AN-18X/GW2AN-9X	OFF, GoConfig, UserLogic, I2C/JTAG/SSPI/QSSPI; OFF by default
GW5A(N)(S)(R)(T)	OFF, UserLogic, JTAG/SSPI/QSSPI,

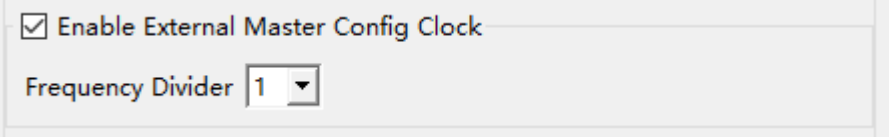
Device	Background Programming Value
	OFF by default

Background Programming values and using considerations are described as follows.

- OFF: Off Background Programming, if the device is GW2AN-18X or GW2AN-9X, "Use MSPI as regular" in the "Dual-Purpose Pin" dialog box is unchecked and non-configurable.
- JTAG: Use JTAG mode for background programming
- I2C: Use I2C mode for background programming: For GW1N-1P5/GW1N-2/GW1NR-2 devices in version B, "I2C Slave Address (Hex)" is used to set the address of the I2C device, with the value range of 00~7F, as shown in Figure 4-30. After selecting I2C, "Use JTAG as regular IO" in the "Dual-Purpose Pin" dialog box is unchecked and not configurable. For GW1N-2/GW1N-1P5/GW1NR-2 devices in version C, when using I2C mode for background programming, there is no option "I2C Slave Address (Hex)" in the configuration dialog box; and "Use RECONFIG as regular IO" is unchecked and non-configurable in the "Dual-Purpose Pin" dialog box.
- GoConfig: Use goConfig IP for background programming.
- UserLogic: Use internal logic for background programming.
- I2C/JTAG/SSPI/QSSPI: Use I2C/JTAG/SSPI/QSSPI mode for background programming.
- JTAG/SSPI/QSSPI: Use JTAG/SSPI/QSSPI mode for background programming.
- For GW2AN-18X/GW2AN-9X devices, when GoConfig, UserLogic, or I2C/JTAG/SSPI/QSSPI is selected, the option "HOTBOOT" will be displayed in the dialog box, optional and unchecked by default, as shown Figure 4-31.
- For GW5A(S)(T)-138/GW5A(R)-25 devices, when UserLogic or JTAG/SSPI/QSSPI is selected, the option "HOTBOOT" will be displayed in the dialog box, optional and unchecked by default, as shown Figure 4-31.
- For GW1N-1P5/GW1N-2/GW1NR-2 in version B, if the I2C is included in the two switched configuration items, Synthesize and Place & Route will be changed to expired; for GW1N-2/GW1NR-2/GW1N-1P5 in version C, if the I2C is included in the two switched configuration items, only Place & Route will be changed to expired.
- For GW2AN-18X/GW2AN-9X, if the configuration items GoConfig and UserLogic are switched with I2C/JTAG/SSPI/QSSPI and OFF, Synthesize and Place & Route will be changed to expired, otherwise only Place & Route will be changed to expired.

- Enable External Master Config Clock: Enable the external master config clock, only GW5A(N)(S)(R)(T) series support this option; for other devices, this option is not displayed on the configuration interface, unchecked by default. When this option is checked, the configuration option "Frequency Divider" will appear in the dialog box, as shown in Figure 4-32. For GW5A(S) (R)-25/GW5AT-60/GW5A(N)(R)T-15 devices, the values are 1, 2, 4, 8, and the default is 1. For GW5A(S) (R)-25/GW5AT-60 devices, the values are 1 and even numbers from 2 to 1022; and there are totally 512 values, and the default is 1.

Figure 4-32 Frequency Divider Option



- Enable SEU Handler: Enable Single-Event Upsets Handler (SEU Handler). Only GW5A(N)(S)(R)(T) series support this function, unchecked by default. When this option is checked, the configuration sub-options "Enable SEU Handler CheckSum", "Enable Error Detection only", "Enable Error Detection and Correction", "Stop SEU Handler when detected uncorrectable ECC error or CRC checksum mismatch error", "Mode", and "Enable Error Injection". Among them, "Enable Error Detection only" and "Enable Error Detection and Correction" cannot be checked at the same time, as shown in Figure 4-33.

Figure 4-33 Enable SEU Handler Option

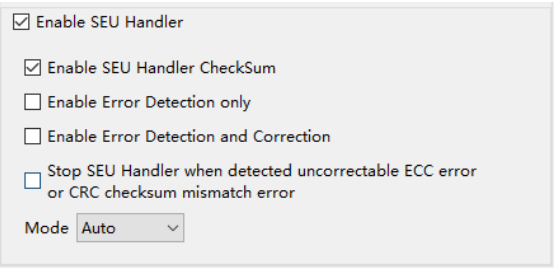
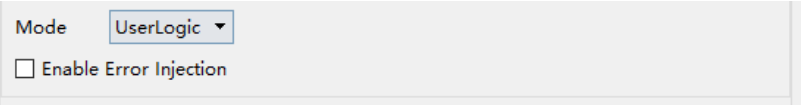


Figure 4-34 Enable Error Injection Option



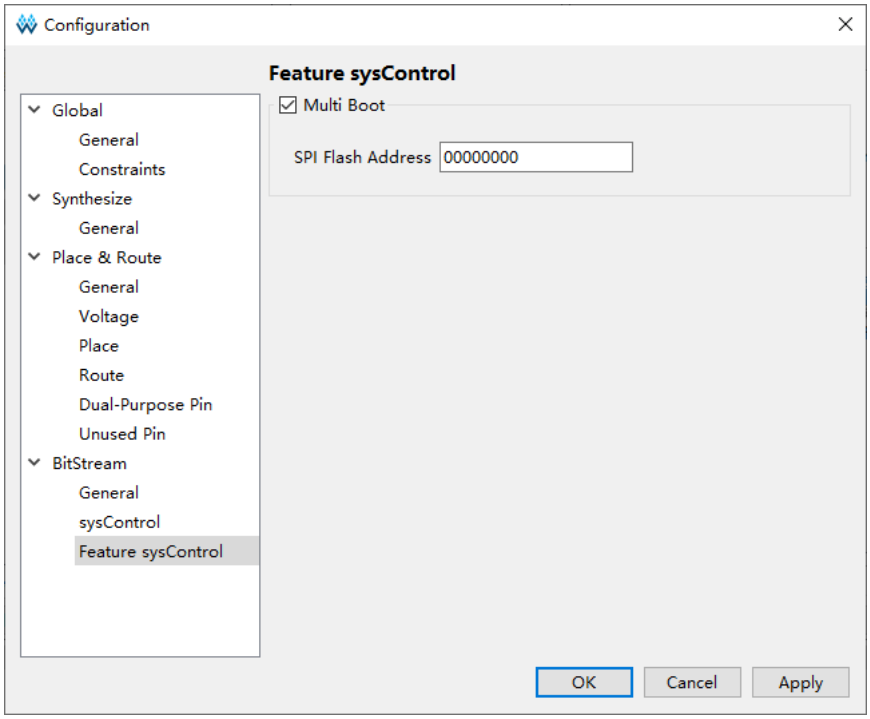
  - Enable SEU Handler CheckSum: Enable Single-Event Upsets Handler, detection, calculation and comparison, unchecked by default.
  - Enable Error Detection only: Enable error detection only, unchecked by default.
  - Enable Error Detection and Correction: Enable error detection and correction, unchecked by default.

- Stop SEU Handler when detected uncorrectable ECC error or CRC checksum mismatch error: Stop SEU Handler when uncorrectable ECC error or CRC checksum mismatch error is detected, unchecked by default.
- Mode: Select SEU Handler start or stop mode; the values are Auto and UserLogic, and the default is Auto.
- Enable Error Injection: The option Enable Error Injection appears when UserLogic mode is selected, unchecked by default, as shown in Figure 4-34.
- CSR File: Used to specify the CSR file.

Feature sysControl

For Gowin devices apart from GW5A(N)(S)(R)(T) series, the Feature sysControl configuration is as shown in Figure 4-35.

Figure 4-35 Feature sysControl Configuration for Gowin Devices apart from GW5A(N)(S)(R)(T)

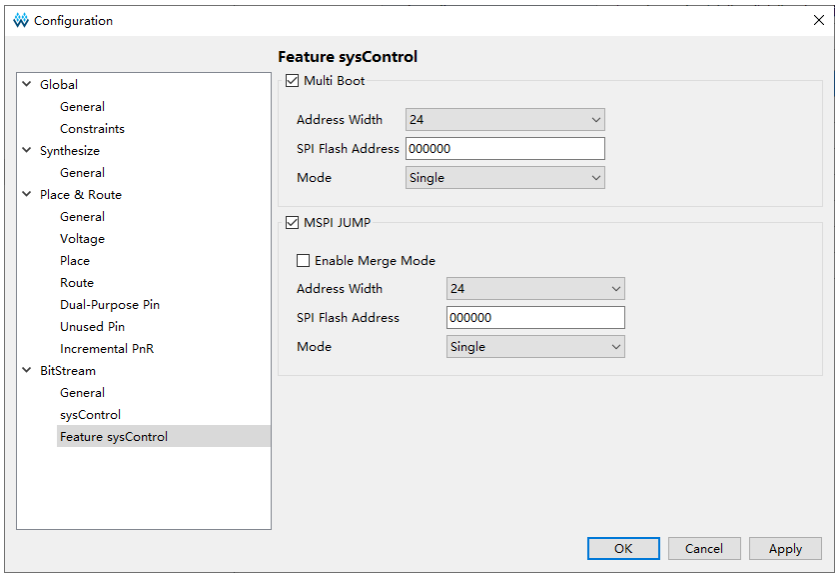


Multi Boot is checked by default, and the sub-configuration SPI Flash Address will appear.

- SPI Flash Address: Specify SPI Flash address. The SPI Flash address refers to the initial address to which the bitstream will be loaded for the next multiboot. For GW2AN-18X and GW2AN-9X, the default is 000000; for the devices other than GW5A(N)(S)(R)(T) devices, the default value is 00000000. For further details, see *SUG502, Gowin Programmer User Guide*.

For GW5A(N)(S)(R)(T) series devices, the Feature sysControl configuration is as shown in Figure 4-36.

Figure 4-36 Feature sysControl Configuration for GW5A(N)(S)(R)(T)



Multi Boot is unchecked by default, and the following sub-configurations will be displayed when it is checked.

Table 4-11 Sub-configurations

Name	Description
Address Width	Configure SPI Flash address width; the values are 24 and 32, and the default is 24.
SPI Flash Address	Specify SPI Flash address. The SPI Flash address refers to the initial address to which the bitstream will be loaded for the next multiboot. The default is 000000; for further details, see *SUG502, Gowin Programmer User Guide*.
Mode	Configure SPI Flash address access mode; the values are Single, Fast, Dual and Quad, and the default is Single.

MSPI JUMP is unchecked by default, and the following sub-configurations will be displayed when it is checked.

Table 4-12 Sub-configurations

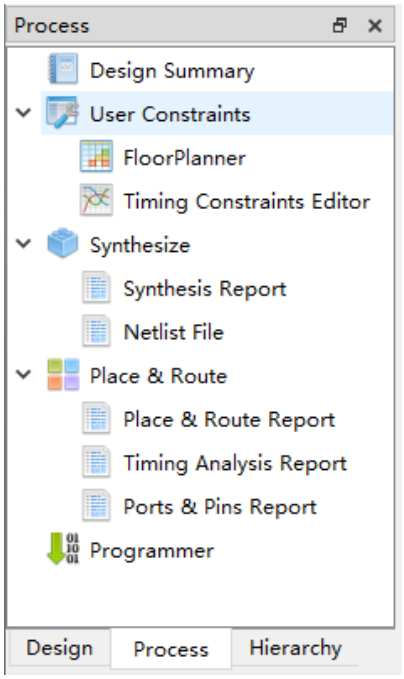
Name	Description
Enable Merge Mode	Using this option will merge the MSPI JUMP bitstream file into a general bitstream file, unchecked by default.
Address Width	Configure SPI Flash address width; the values are 24 and 32, and the default is 24.
SPI Flash Address	Specify SPI Flash address, and the default is 000000.
Mode	Configure SPI Flash address access mode; the values are Single, Fast, Dual and Quad, and the default is Single.

4.4 Manage a Project

The Process view provides the process of FGPA design flow, as shown in Figure 4-37. The Process View includes following operations.

- View design summary
- Start physical constraints editor
- Start timing constraints editor
- Run Synthesis
- View Synthesis report
- Run Place & Route
- View reports generated after Place & Route
- Start Programmer

Figure 4-37 Project Process View



4.4.1 Design Summary

A new project created, the software will analyze the project and provide a report of Design Summary, which will include the project file path, synthesis information and device information, as shown in Figure 4-38. There are three following ways to open the Design Summary:

- From the menu bar, select "Window > Design Summary".
- In the Process View, double-click "Design Summary".
- In the Process View, right-click "Design Summary", and select "Open".

Figure 4-38 Project Summary

General	
Project File:	D:\gowin_project\daily_test\daily_test.gprj
Synthesis Tool:	GowinSynthesis

Target Device	
Part Number:	GW1N-UV4PG256C6/I5
Series:	GW1N
Device:	GW1N-4
Device Version:	B
Package:	PBGA256
Speed Grade:	C6/I5
Core Voltage:	UV

Note!

For devices without version, the Device Version will not be displayed in the table.

4.4.2 User Constraints

User constraints provide quick access to open and create constraints files. User constraints include physical and timing constraints.

For the details, please refer to *SUG940, Gowin Design Timing Constraint User Guide*, and *SUG935, Gowin Design Physical Constraints User Guide*; *SUG1018, Arora V Design Physical Constraints User Guide*.

4.4.3 Synthesize

GowinSynthesis is the synthesis tool developed by Gowin. It supports GOWINSEMI library files and their implementations. It supports System Verilog 2017, Verilog 2001, Verilog 95, VHDL 1993 and VHDL 2008.

Right-click "Synthesize" and select "Configuration" in the Project Design area to open the configuration dialog box, as shown in Figure 4-20.

Synthesize provides functions of running synthesis, setting synthesis parameters, and managing Netlist File and Synthesis Report. For the Synthesis Report, see 6.1 Synthesis Report.

Refer to the following steps to run Synthesize.

1. Configure synthesis, and you can refer to 4.3.3 Edit Project Configuration.
2. Run Synthesize
3. In the Process pane, double click "Synthesize" or right-click "Synthesize" and select "Run" to start synthesis of source files. If the synthesis is successful, the icon " " appears before Synthesize; if not, the icon " " appears.
4. After synthesis completed successfully, double click "Netlist Report", "Synthesis Report" or right-click and select "Open" to view the Netlist Report and synthesis report. The post-synthesis generated netlist file is *.vg, and synthesis report file is *_syn.rpt.html.

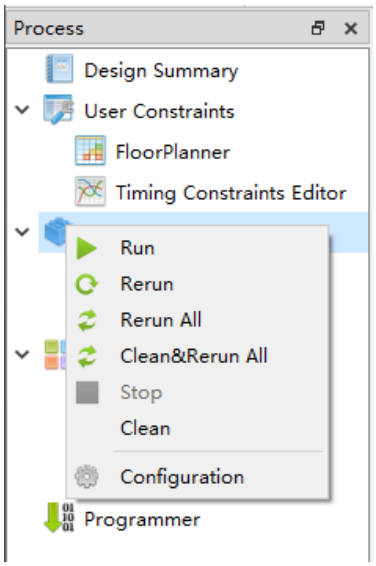
If the Synthesize icon is " " before synthesis, double click "Netlist File", "Synthesis Report" or right click to select "Open" to synthesize first, and the netlist file or synthesis report can be opened after successful

synthesis.

The right-click operations of Synthesize is as shown in Figure 4-39.

- Run: Only when the icon before Synthesize is "  " (Initial Status), "  " (Failure Status), or "  " (Expired Status), you can select Run to start synthesis of source files.
- Rerun: No matter what Synthesize status is, you can select Rerun to restart synthesis of source files.
- Rerun All: If this option selected, regardless of the status of "Place & Route" and "Synthesize", the source file will be synthesized and placed & routed again
- Clean & Rerun All: Clean the gwsynthesis and pnr folders under the project folder impl, and synthesize and place & route the source files again.
- Stop: Stop Synthesize
- Clean: Clean the generated folder after synthesis (a folder generated by GowinSynthesis is gwsynthesis). Click it and a prompt will pop up.
- Configuration: Configure Synthesis parameters

Figure 4-39 Right-click Synthesize



4.4.4 Place & Route

Place and route provides the functions of running PnR, setting parameters, and managing the generated files.

Note!

Place & Route will be implemented after running Synthesize.

Refer to the following steps to run Place & Route.

1. Configure Place & Route, please refer to 4.3.3 Edit Project Configuration.

2. Run Place & Route, double-click "Place & Route", or right-click and select "Place & Route > Run" to generate bitstream files and related reports. If running successfully, the "✔" icon will appear before Place & Route. Otherwise, the "❗" will appear;
3. After Place & Route has been run successfully, double-click on "View Post PnR Report" or right-click and select "Open" to view the report.
4. You can view four kinds of reports: Place & Route Report, Timing Analysis Report, Ports & Pins Report, and Power Analysis Report. These reports can not be edited. See 6.2 Place & Route Report , 6.3 Ports and Pins Report, 6.4 Timing Report, 6.5 Power Analysis Report for the details.

Note!

- If the report is already opened and it is regenerated by running Place & Route again, a update prompt will pop up.
- Before running Place & Route, if the status icon before Place & Route is " ", double click the report or right-click and select "Open" to run Place & Route first. The report will be opened after Place & Route runs successfully.

Right-click operations of Place & Route are as follows.

- Run: You can select "Run" to start Place & Route only when the icon before Place & Route is " ", " ", or " ".
- Rerun: Regardless of the Place & Route status, you can select Rerun to rerun Place & Route.
- Rerun All: If this option selected, regardless of the status of "Place & Route" and "Synthesize", the source file will be Synthesize and placed & routed again.
- Clean & Rerun All: Clean the gwsynthesis and pnr folders under the project folder impl, and synthesize and place & route the source files again.
- Stop: Stop Place & Route
- Clean: Clean all the generated files after PnR; a prompt will pop up when you click this option. A Warning message will be reported if the deletion of a folder fails.
- Configuration: Configure Place & Route parameters.

4.4.5 Programmer

Bitstream files will be generated after Gowin Software has run placement and routing. Start Gowin FPGA programmer to download the bitstream files to the chip to realize user-required functions.

Note!

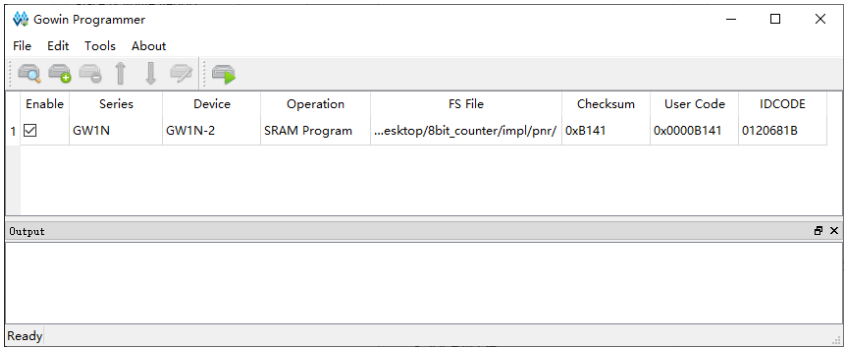
Programmer will be implemented after running Synthesize and Place & Route. If you do not run synthesize and Place & Route first, warnings will appear.

Double-click "Programmer" or right click and select "Run" to open Gowin FPGA Programmer, as shown in Figure 4-40.

Note!

The Programmer in the Linux installation package does not work with Linux Red Hat 5.10, only works with Red Hat 6 and above, and the Linux core version needs to be 2.18 and above.

Figure 4-40 Gowin Programmer



For the usage, please refer to *SUG502, Gowin Programmer User Guide*.

4.5 Archive and Restore a Project

Gowin Software can archive project and restore archived project. Use "Archive Project" and "Restore Archived Project" under "Project" in the menu bar to archive or restore archived project.

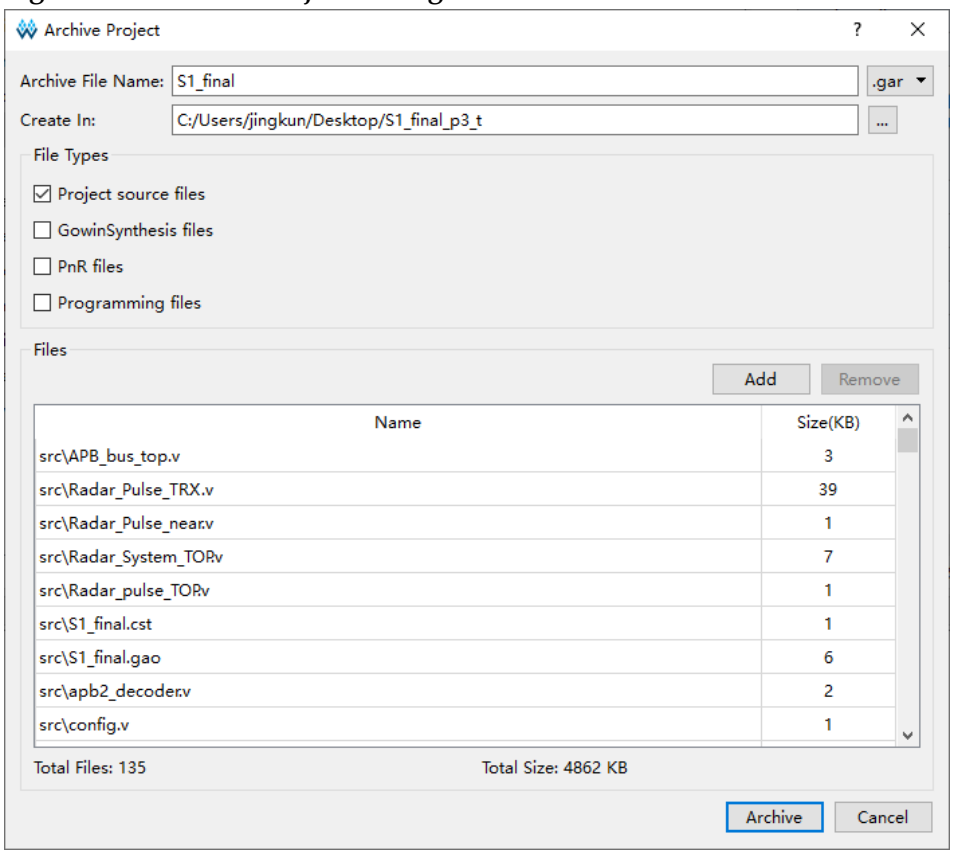
4.5.1 Archive a Project

A dialog box will pop up when you click "Project > Archive Project", as shown in Figure 4-41; an explanation is displayed when your mouse hover over an option.

- Archive File Name is the archived file name. The default name is the same as the current archived project name with extension .gar.
- Create In is the path for the archived file, and the default is the current project path.
- The File Types include Project source files (checked by default), GowinSynthesis files, PnR files and Programming files.
  - Project source files: Includes all the files under the path /src where the project is located.
  - GowinSynthesis files: Includes project files (*.prj), netlist files (*.vg), synthesis reports (*_syn.rpt.html), resource statistics files (*_syn_rsc.xml) generated synthesis under the path /impl/gwsynthesis where the project is located.
  - PnR files: Includes the files generated by PnR under the path /impl/pnr where the project is located.
  - Programming files: Includes the bitstream file *.fs, *.bin and *.binx generated by PnR under the path /impl/pnr where the project is located.

- When a file type is checked, the source file, path and size of the current project are displayed below.
- Add and Remove can be used to add and remove archived files.
- After clicking Archive, a prompt box will pop up if the files in the project are not saved.
- After archiving, a prompt will pop up, indicating the success or failure of the archiving.
- When the archiving is completed, two files will be generated under the "Create In" path: the archived project *.gar and the archived file *.garlog. The file with .gar suffix compresses and stores all the archived files. The log file with *.garlog suffix is used for checking which files are archived and whether the archiving is successful.

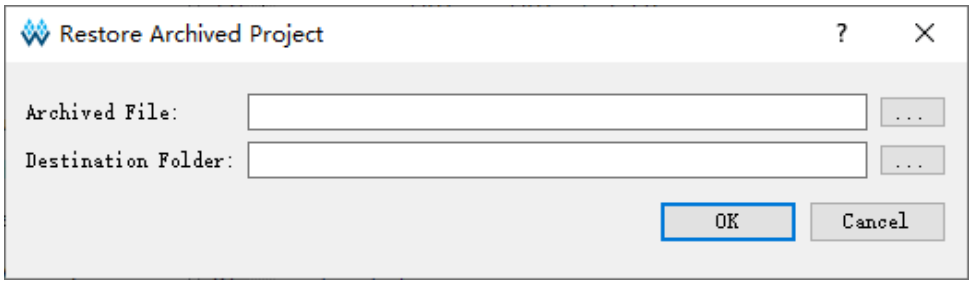
Figure 4-41 Archive Project Dialog Box



4.5.2 Restore Archived Project

Restore Archived Project dialog will pop up when you clicks Project in the menu bar, as shown in Figure 4-42 .

Figure 4-42 Restore Archived Project Dialog Box



Click the button on the right side of Archived File to select the archived file to restore. After selecting, "Destination Folder" is automatically updated to the path where the archive file is. Click "OK" and a dialog box will pop up.

4.6 Exit Software

There are two ways to exit Gowin Software.

1. Select "File > Exit" from the File menu.
2. Click the "" icon on the upper right of the IDE.

Note!

- If files are not saved, IDE will prompt you to save the files first.
- Save, Save All, and Save As…are only available for text editing.
- Project configuration modification and project files addition and deletion will not be saved to project configuration files in time; they will be saved automatically when you close the software.
- If the software is running, you cannot exit software by clicking.

5 Tools Integrated in Gowin Software

5.1 Physical Constraints Editor

Gowin FloorPlanner is designed in-house by Gowin, and it supports reading and editing the attributes and locations of I/O, Primitive, Block (BSRAM and DSP), and Group, etc. It also supports the generation of new placement and constraints files according to your configuration. These files define the I/O attributes, primitives and locations, etc. Gowin FloorPlanner, supporting Gowin all FPGA products, provides an editing way to improve design efficiency, and it also supports timing optimization.

FloorPlanner can be started using two methods.

1. If no FPGA project is created, you can select "Tools > FloorPlanner" directly from the menu bar. You will need to add netlist files and devices information by selecting "File > New".
2. If an FPGA project is already created, run "Synthesize", and then double-click "FloorPlanner" directly in the Process View. The Floorplanner will then load the project files directly. The FloorPlanner includes Summary, Netlist, Chip Array, Package View, as shown in Figure 5-1 and Figure 5-2.

Note!

- For more details, see *SUG935, Gowin Design Physical Constraints Guide*; *SUG1018, Arora V Design Physical Constraints User Guide*.
- The FloorPlanner also supports timing optimization.

Figure 5-1 Chip Array View

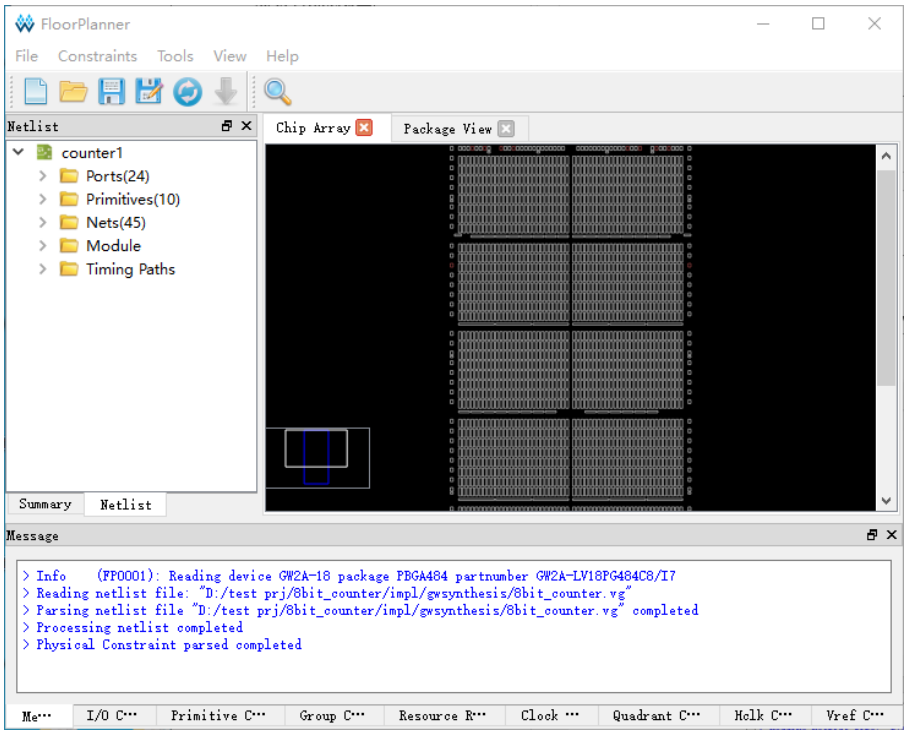
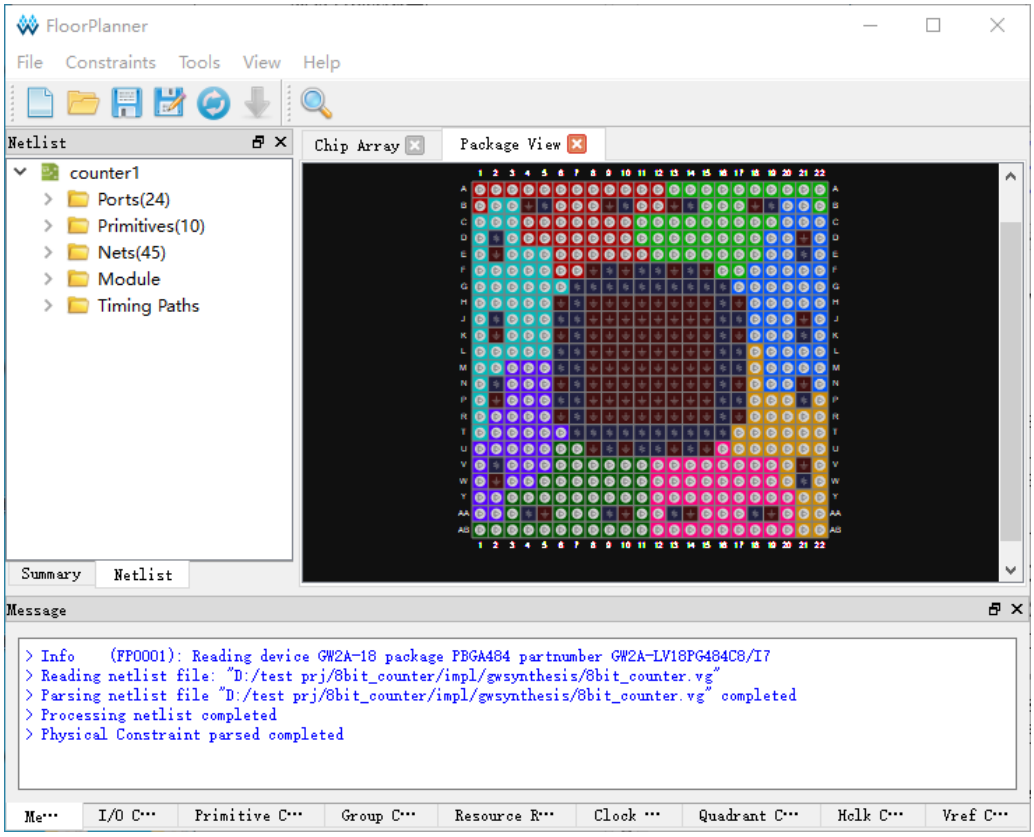


Figure 5-2 Package View



5.2 Timing Constraint Editor

The Gowin Timing Constraints Editor is designed in-house by Gowin, and supports multiple timing constraints commands, including clock constraints, I/O constraints, path constraints, and clock report editing. The Timing Constraints Editor allows an easy and quick timing constraint editing for Gowin all FPGA products.

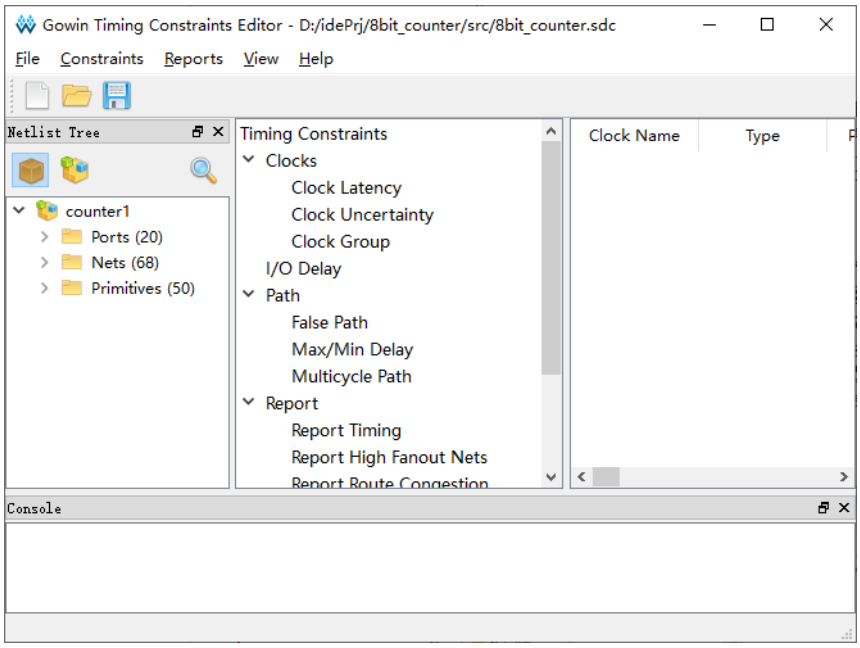
Timing Constraints Editor can be started using two methods:

1. If no FPGA project is created, you can select "Tools > Timing Constraints Editor" from the menu bar. Add netlist files by selecting "File > New".
2. If an FPGA project is already created, Run "Synthesize", double-click "Timing Constraints Editor", and the Timing Constraints Editor will load project files directly, as shown in Figure 5-3.

Note!

For the details, see *SUG940, Gowin Design Timing Constraint User Guide*.

Figure 5-3 Create Timing Constraints



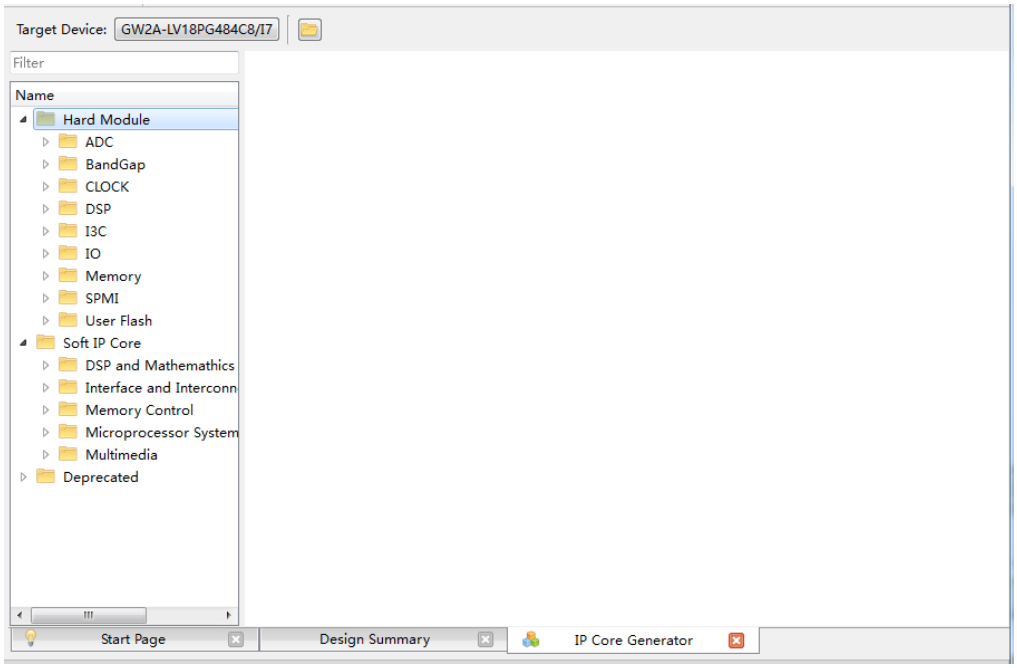
5.3 IP Core Generator

The IP Core Generator in Gowin Software is mainly used to generate hard and soft IPs, which you can call to implement the required functions. As shown in Figure 5-4, its main functions are as follows.

- Supports Soft IP core, Hard module information preview
- Supports customized Soft IP core and Hard module.
- Supports Hard module instance generation.
- Can save configuration automatically.
- Supports IP generation code language selection.

- Some Soft IP can generate incentive file automatically.
- Can display available IPs by selecting device.

Figure 5-4 IP Core Generator



Start the IP Core Generator by selecting "IP Core Generator" from the Tools menu. For the details, you can see following manuals.

- For the details of ADC, BANDGAP, I3C and SPMI, you can see *SUG283, Gowin Primitives User Guide*; *UG299, Arora V Series Analog to Digital Converter (ADC) User Guide*.
- For the details of CLOCK, you can see *UG286, Gowin Clock User Guide*; *UG306, Arora V Clock User Guide*.
- For the details of DSP, you can see *UG287, Gowin Digital Signal Processing User Guide*; UG305, Arora V Digital Signal Processing (DSP) User Guide.
- For the details of IO Logic, you can see *UG289, Gowin Programmable IO (GPIO) User Guide;* UG300, Arora V BSRAM & SSRAM User Guide.
- For the details of BSRAM & SSRAM, you can see *UG285, Gowin BSRAM & SSRAM User Guide*; UG300, Arora V BSRAM & SSRAM User Guide.
- For the details of User Flash, you can see *UG295, Gowin User Flash User Guide*.
- For the soft IPs reference design, you can click this link.

Note!

The grayed out Hard Module or Soft IP Core is not supported by the current device.

5.4 Gowin Analyzer Oscilloscope

The Gowin Analyzer Oscilloscope (GAO) is a digital signal analyzer that is designed in-house by Gowin. It helps you to analyze signal timing in design more easily, and quickly perform system analysis and fault locating, thereby improving design efficiency.

GAO supports RTL-level signal capture and netlist-level signal capture after synthesis, and provides standard version (Standard) and simplified version (Lite). Standard GAO can support up to 16 AOs, each of which can be configured with one or more trigger ports, supporting multi-level static or dynamic trigger expressions. Lite GAO is easy to configure, and you do not need to set trigger conditions, and it also can capture the initial value of the signal, which is convenient for you to analyze the state of power-on. After signal capture, the waveform can be exported, supporting *.csv, *.vcd and *.prn file formats. *.csv and *.prn files can be directly used in matlab and other third-party simulation tools, and *.vcd file can be directly used in ModelSim.

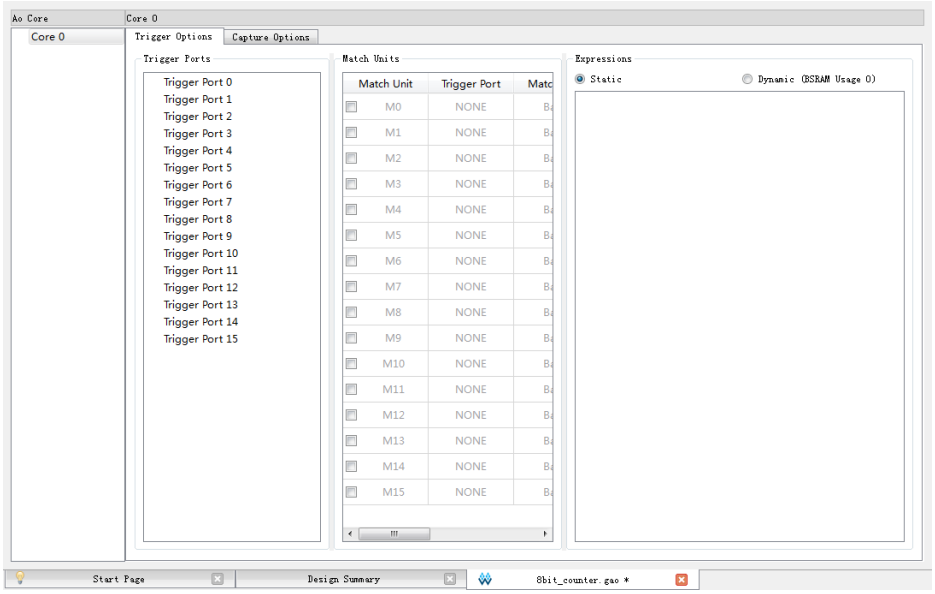
Note!

Matlab and ModelSim tools require a third-party license.

The GAO includes Gowin GAO configuration and the Gowin Analyzer Oscilloscope. Gowin GAO configuration is mainly used to insert position information into the design, which is predominantly based on the sampling clock, trigger unit, and trigger expression; Gowin Analyzer Oscilloscope connects software and target hardware through the JTAG interface, and visually displays the data of the sampled signal set by GAO Config File with the waveform.

Before starting GAO configuration, create the GAO config file in the Project Area to open the GAO configuration view, taking standard version as an example, as shown in Figure 5-5.

Figure 5-5 GAO Config File View

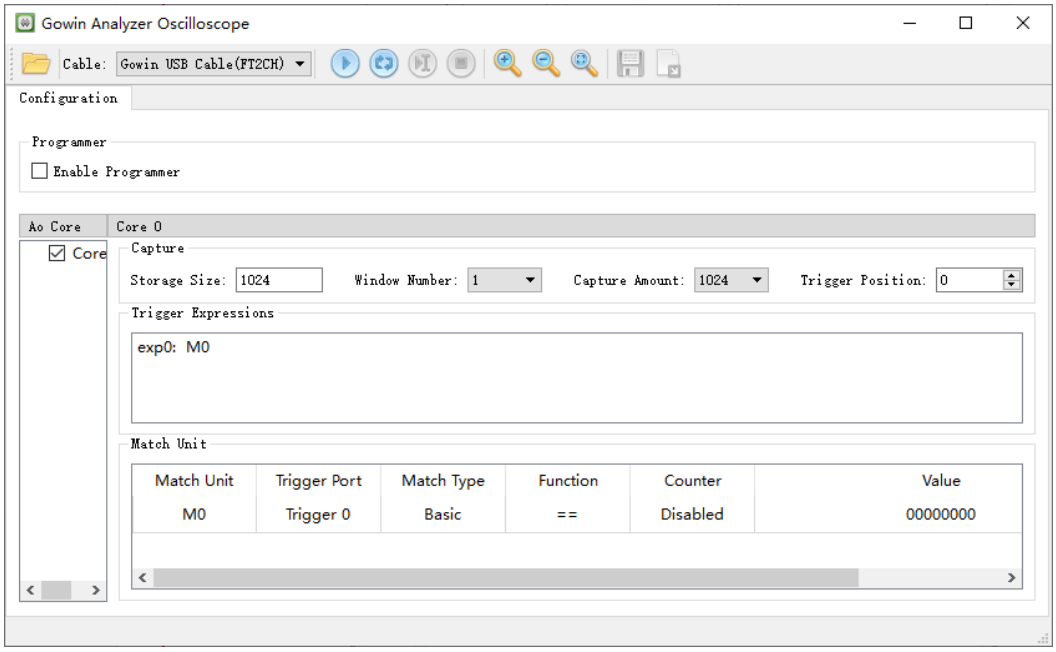


After the configuration file is created, select "Tools > Gowin Analyzer Oscilloscope" from the menu bar to open the Gowin Analyzer Oscilloscope, as shown in Figure 5-6.

Note!

For the details, see *SUG114, Gowin Analyzer Oscilloscope User Guide*.

Figure 5-6 GAO View



5.5 Gowin Power Analyzer

Gowin Power Analyzer (GPA) helps to analyze the power consumption of your design and provides many configuration options. You can configure the parameters according to the actual design. GPA automatically estimates the power consumption and generates a power consumption analysis report according to the parameters.

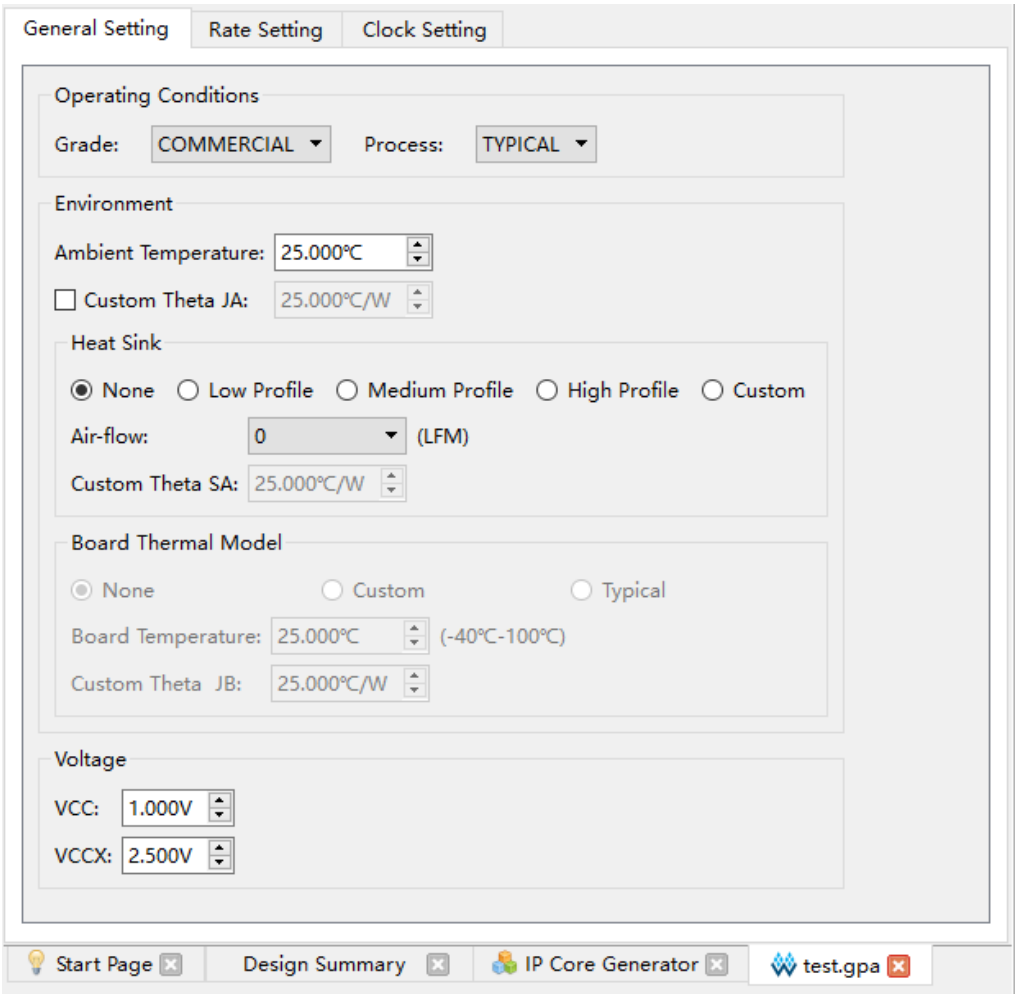
Based on the new configuration file (.gpa), follow the below steps to start the GPA.

1. In the Design area, select "File> New…" to open a "New" dialog box.
2. Select "GPA Config File" and type a Name in the pop-up dialog box.
3. Click "OK", and the new GPA config file will be displayed in the Project Design View.
4. Double click on the file name to open the GPA Config view, as shown in Figure 5-7.

Note!

For the details, see *SUG282, Gowin Power Analysis User Guide*.

Figure 5-7 GPA Config File View



5.6 Memory Initialization File Editor

The Memory Initialization File is an ASCII file with an .mi suffix. You can generate the corresponding Initialization File according to your design to specify the initial value for the memory of each address. The Memory Initialization File editor can be used to open and edit the existed .mi file.

Each line in the Memory Initialization File correlates with one memory. The number of lines is the number of memories and also the memory address depth. The number of columns represents each memory bit; that is, the memory data width. The address decreases from top to bottom with the most significant bit first for each line.

Gowin memory initialization file is based on .mi file. For the details, see *UG285, Gowin BSRAM & SSRAM User Guide*. The steps are as follows:

1. In the Design area, select "File> New…" to open a "New" dialog box.
2. Select "Memory Initialization File", as shown in Figure 5-8. Click "OK" and type the initialization file name in the pop-up "New File" dialog box, and then click "OK". The Initialization File Configuration View is as shown in Figure 5-9.

3. Start the file initialization view as shown in Figure 5-10. Type the initial value on the left side and configure the initialization file format and depth/width and view on the right side.
4. On the right side, configure the depth and width for the initialization file and the format for the address and initial values in the left table.
   - The depth and width values should be same as the Block Memory or Shadow Memory address depth and data width set in the IP Core Generator. If the address depth and data width in the initialization file are greater than the values set in the IP Core Generator, the IP Core Generator will prompt an error message. If the depth and width values are less than the address depth and data width set in the IP Core Generator, the value of the unassigned address will be initialized to 0 by default. Click "Update" after configuration.
   - The display format of the addresses and values on the left in the table can be configured as binary, hexadecimal, and address-hexadecimal, etc.
5. Type the initial value and set the view in the left table.
   - Right-click the table header to configure the number of columns, which can be configured as 1, 8, or 16, as shown in Figure 5-11.
   - Double-click and type the initial value, or right-click to set the value. "Fill with 0" means the initial value is 0, "Fill with 1" means each bit of the initial value is 1, and "Custom Fill" means you can type the value according to your needs; you can also set initial values in batches, as shown in Figure 5-12.
6. Save the file.

Figure 5-8 New Dialog Box

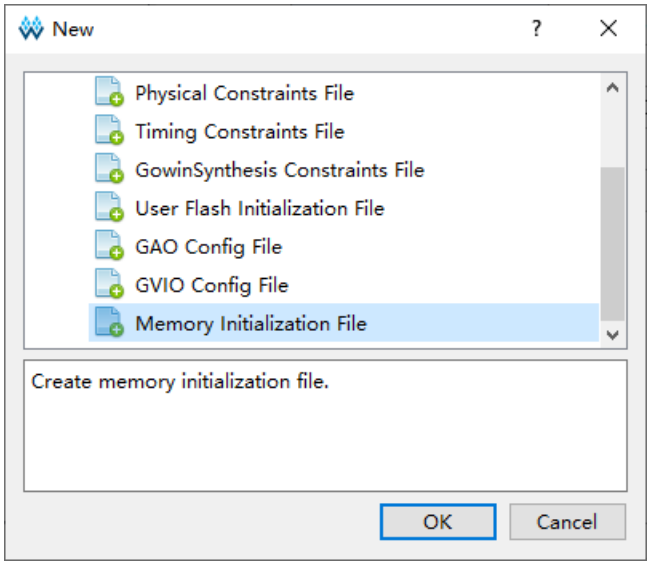


Figure 5-9 New File Dialog Box

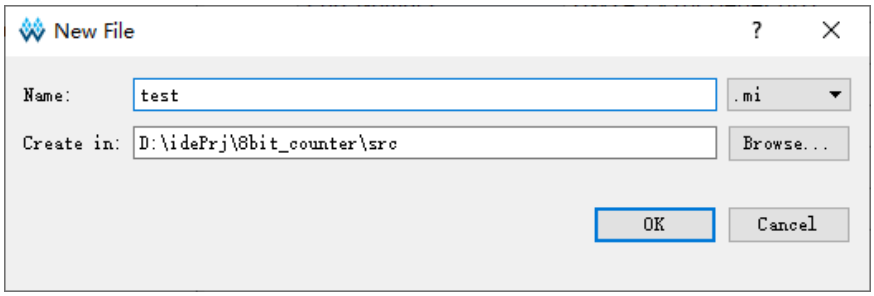


Figure 5-10 Initialization File Configuration View

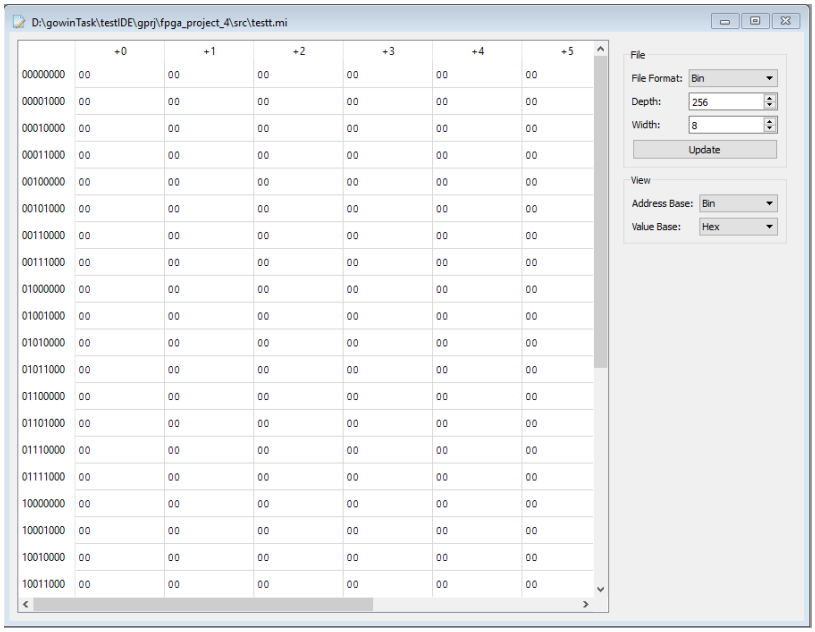


Figure 5-11 Column Setting

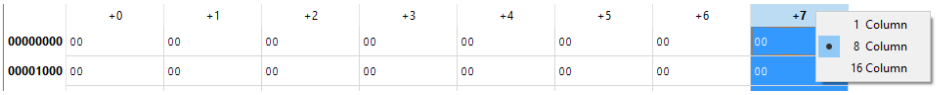
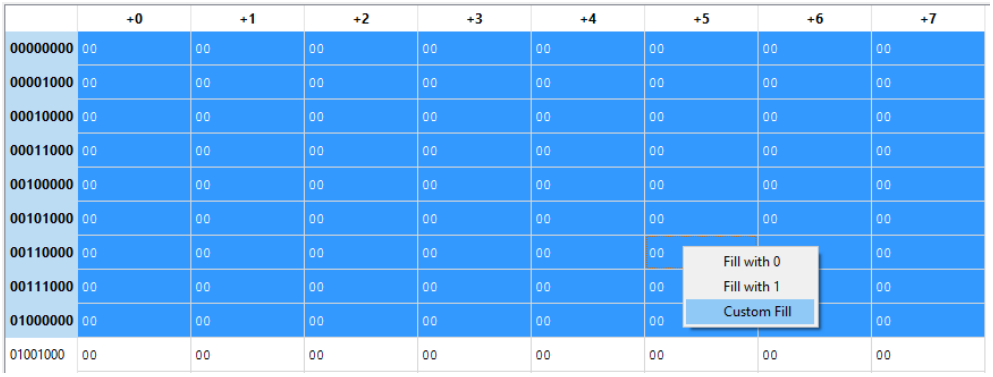


Figure 5-12 Batch Setting



5.7 User Flash Initialization File Editor

User Flash Initialization file is an ASCII file with an .fi suffix. You can generate the corresponding Initialization File according to your design to specify the initial value for the User Flash of each address. The User Flash Initialization File editor can be used to open and edit the existed. fi file.

The name of User Flash initialization file is *.fi(file_name.fi). Each line in the Memory Initialization File correlates with one memory. The number of lines is the number of memories that needs to be initialized. The contents in the header bracket represent the ordinate address and the abscissa address respectively, separated by a semicolon. The contents after the brackets in each line represent the data initialized by the memory, and the data supports binary and hexadecimal, MSB first. The following are examples of the .fi file format.

5.7.1 Bin File

Bin file is a text file that consists of the 0 and 1.

```
//Copyright (C)2014-2024 Gowin Semiconductor Corporation.
//All rights reserved.
//File Title: User Flash Initialization File
//Tool Version: V1.9.10(64-bit)
//Part Number: GW1N-LV4PG256C6/I5
//Device-package: GW1N-4-PBGA256
//Device Version: D
//Flash Type: FLASH256K
//File Format: Bin
//Created Time: 2024-06-28 14:31:12
[0:0] 00000000000100000001000100010000
[1:1] 00000000000100010001000000000001
```

5.7.2 Hex File

The Hex file is similar to the Bin file. It consists of hexadecimal numbers 0~F.

```
//Copyright (C)2014-2024 Gowin Semiconductor Corporation.
//All rights reserved.
//File Title: User Flash Initialization File
//Tool Version: V1.9.10(64-bit)
//Part Number: GW1N-LV4PG256C6/I5
//Device-package: GW1N-4-PBGA256
//Device Version: D
```

//Flash Type: FLASH256K

//File Format: Hex

//Created Time: 2024-06-28 14:41:24

[0:0] 00101110

[1:1] 00111001

Based on the new configuration file (. fi), and you can follow the below steps to use the initialization file editor.

1. In the Design area, select "File> New…" to open a "New" dialog box.
2. Select "User Flash Initialization File", as shown in Figure 5-13. Click "OK" and type the initialization file name in the pop-up "New File" dialog box, and then click "OK", as shown in Figure 5-14. The devices supported by User Flash Initialization File Editor are the same as the devices supported by User Flash Primitives. If the device you selected does not support User Flash, "Current device do not support flash" will pop up at the bottom of "New File" Dialog box after you click "OK".

Figure 5-13 New Dialog Box

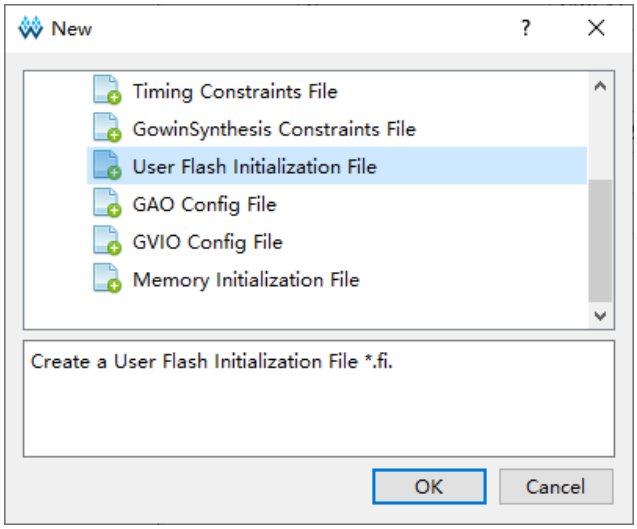
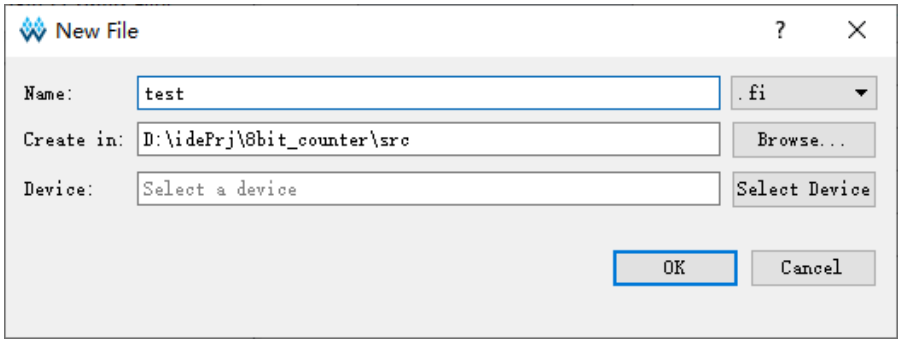
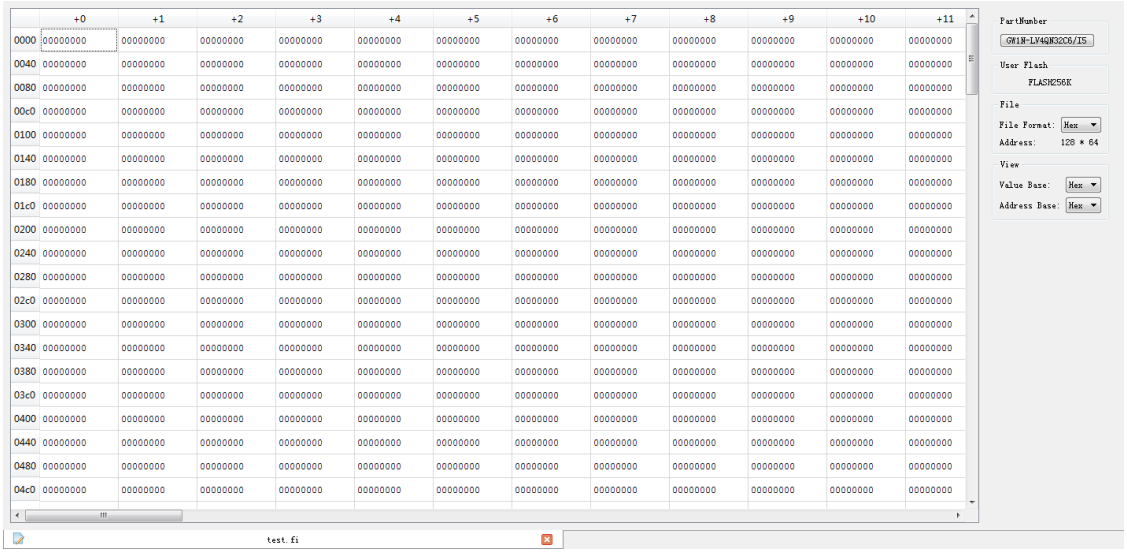


Figure 5-14 New File Dialog Box



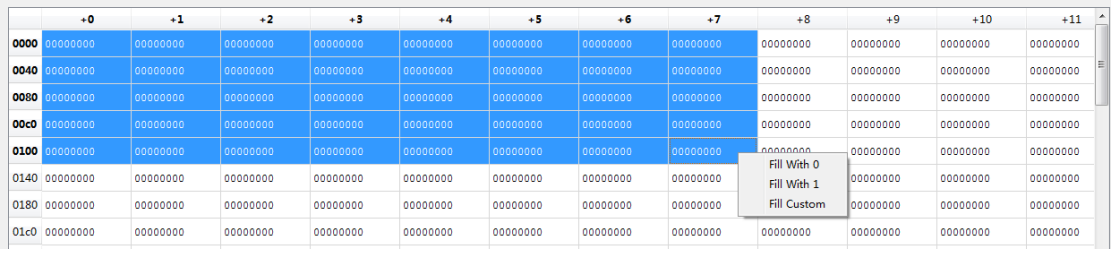
3. Start the file initialization view as shown in Figure 5-15. Enter the initial value on the left side and configure the initialization file format and view on the right side; Part Number and User Flash are also displayed on the right.

Figure 5-15 Initialization File Configuration View



4. On the right side, configure Part Number, file format, address and value.
   - Click "Part Number", then "Select Device" dialog box will pop up; you can select the other part number.
   - The format of address and value can be configured as binary, octal, decimal, hexadecimal, etc.
5. Type the initial value and set the view. Double-click and type the initial value, or right-click to set the value. "Fill with 0" means the initial value is 0, "Fill with 1" means each bit of the initial value is 1, and "Custom" means you can type the value according to your needs; you can also set initial values in batches, as shown in Figure 5-16.

Figure 5-16 Batch Setting



6. Save the file.

5.8 Schematic Viewer

You can understand the design logic by Schematic Viewer, which is helpful to the later modification. Schematic Viewer includes RTL Design Viewer, Post-Synthesis Netlist Viewer. Schematic Viewer uses common component symbols to build circuits, including adders, multipliers, registers, and gates, non-gates, and inverters, etc.

You can click "Tools > Schematic Viewer > RTL Design Viewer/ Post-Synthesis Netlist Viewer to open GUI, as shown in Figure 5-17 and Figure 5-18.

Figure 5-17 RTL Design Viewer

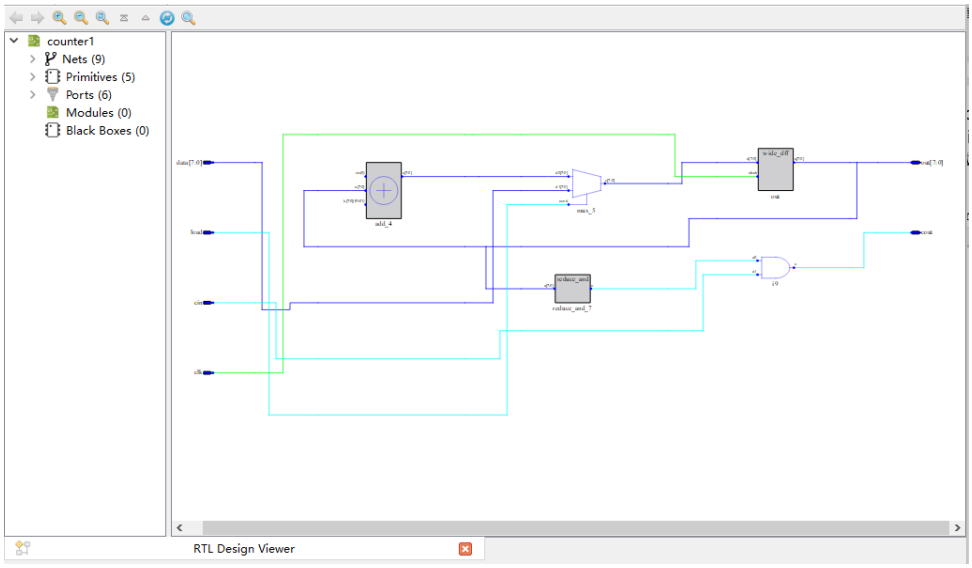
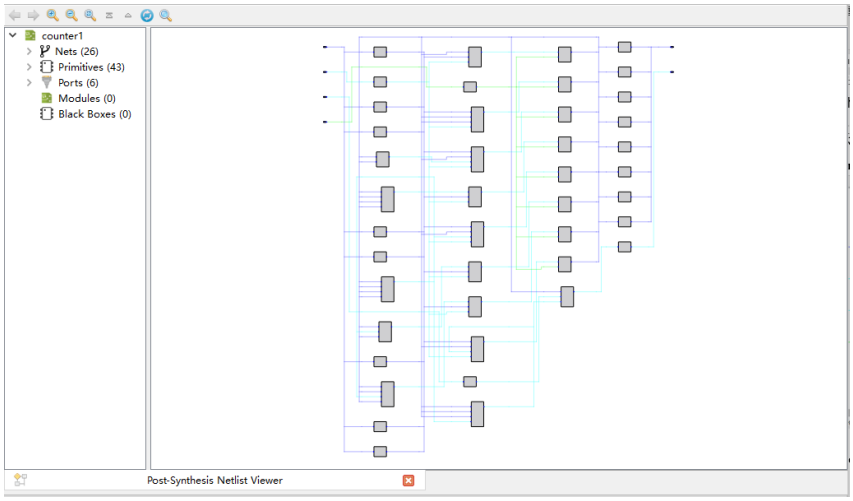


Figure 5-18 Post-Synthesis Netlist Viewer



Schematic Viewer displays backward "", forward "", zoom in "", zoom out "", zoom "", top view "", upper level view "", reload "", and search "". The design hierarchy is displayed on the left side, including Modules, Ports, Nets, Primitives, and Black Boxes.

Note!

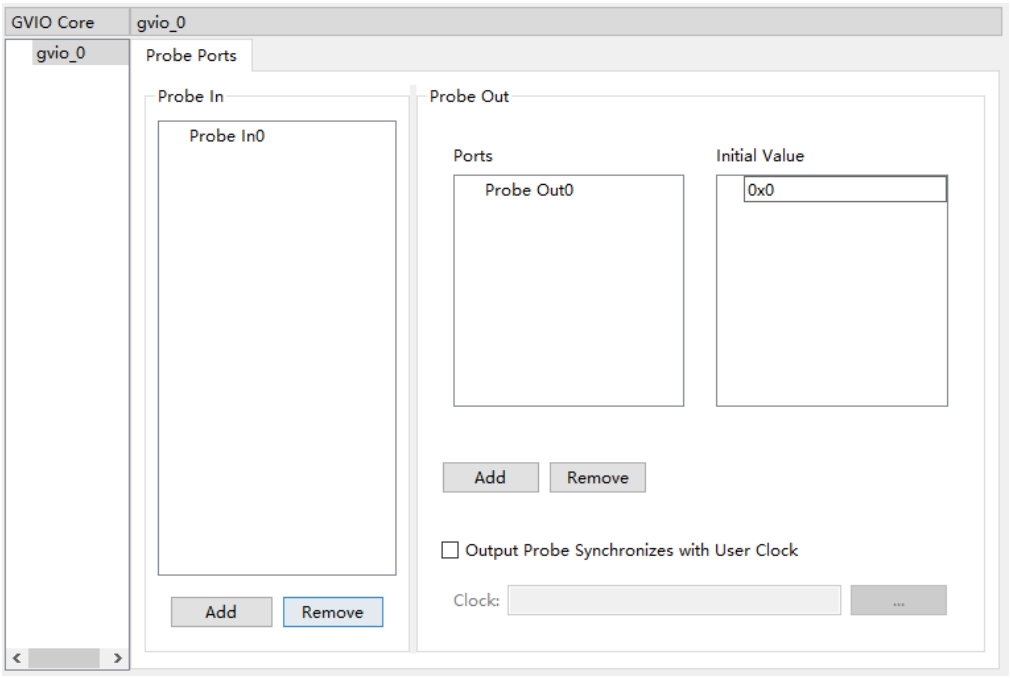
For the details, you can see *SUG755, Gowin HDL Schematic Viewer User Guide*.

5.9 Virtual Input/Output Debugging Tool

Gowin Virtual Input/Output Core (GVIO) is a customizable core that can monitor and drive internal FPGA signals in real time. Its input ports are used to monitor FPGA signals, functioning like virtual LEDs, while its output ports are used to drive FPGA signals, functioning like virtual switches.

Before launching the GVIO configuration file window, you need to create a new GVIO configuration file in the project area and then open the configuration file window, as shown in Figure 5-19.

Figure 5-19 GVIO Configuration File Window



Note!

For configuration and usage instructions of the Virtual Input/Output Core, see *SUG1189, Gowin Virtual Input Output User Guide*.

5.10 Eye Diagram Analysis Tool GoBert

GoBert is a tool designed in-house by Gowin Semiconductor for analyzing the eye diagrams of SerDes RX signals. It helps users evaluate SerDes reception quality, thereby enhancing the performance and reliability of their designs. When using GoBert to test signal reception quality, users need to load their functional application onto the development board. Once the functional application is operating normally, the eye diagram testing can be initiated.

Click the "Tools" option on the menu bar or the "  " button on the toolbar of Gowin Software to launch the GoBert window, as shown in Figure 5-20.

Figure 5-20 GoBert Window

Note!

For the configuration and usage of the eye diagram analysis tool GoBert, you can see *SUG1198, Gowin GoBert User Guide*.

6 Output Files

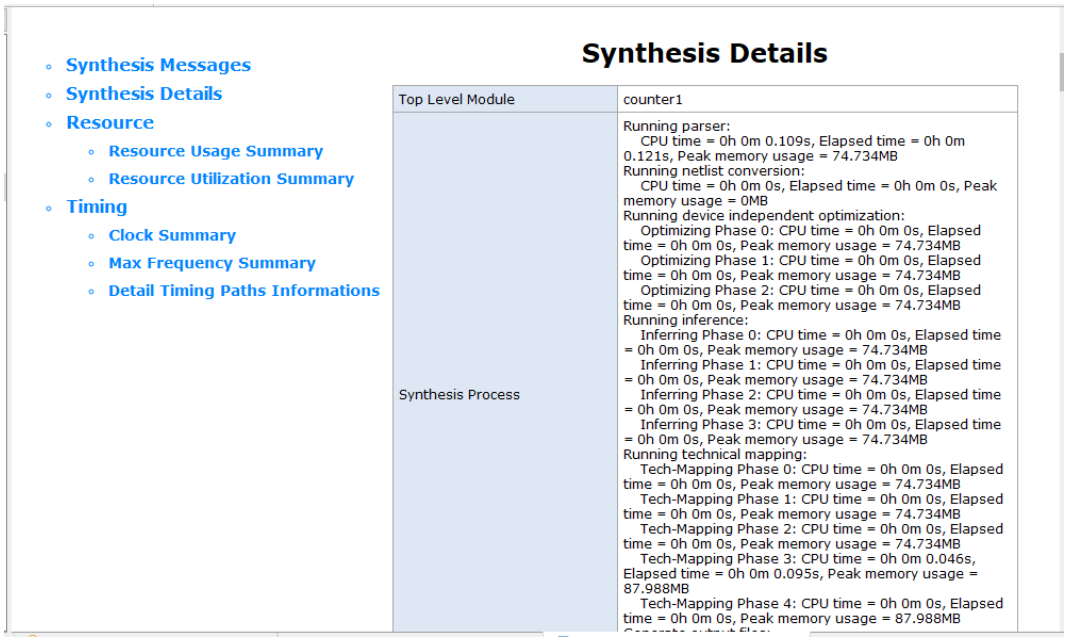
In the process of FPGA design, in addition to bitstream file, Gowin Software can also generate multiple reports for reference by different operating parameters. They are synthesis, place & route, ports, timing and power reports. In addition, you can right-click Place & Route to modify the configuration to generate pins constraints, timing simulation files, etc.

6.1 Synthesis Report

GowinSynthesis will generate synthesis reports and netlist files after synthesis.

The report named *_syn.rpt.html is generated, including Synthesis Message, Synthesis Details, Resource, and Timing, as shown in Figure 6-1.

Figure 6-1 GowinSynthesis Report



The details are as follows.

- Synthesis Message: Includes design file, constraint file, software

version, device, report creation time and legal statement, etc.

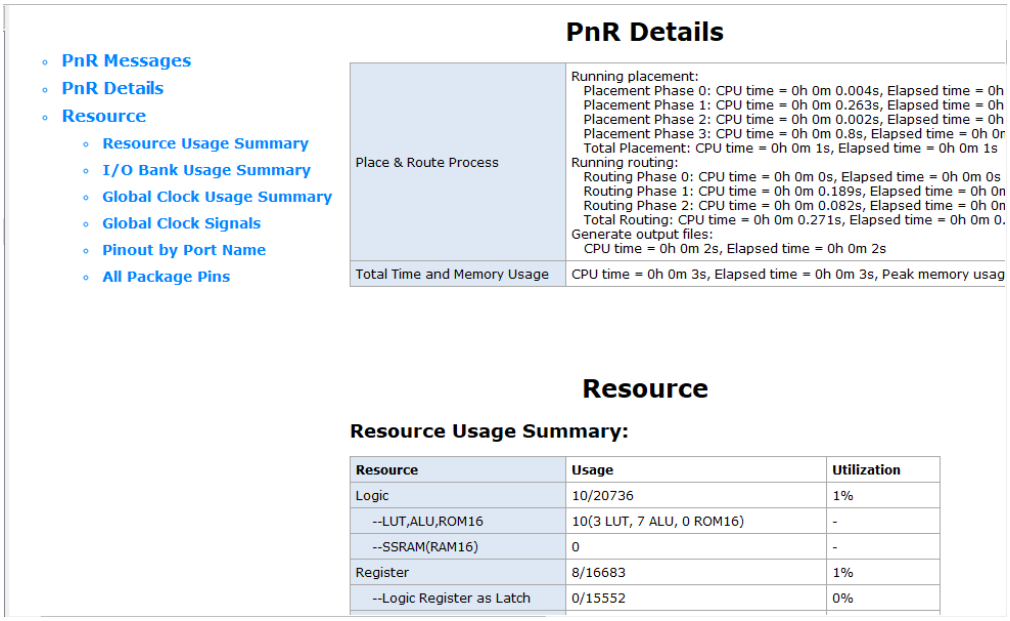
- Synthesis Details: Includes top module of the design file, the synthesis running time and CPU running time, as well as the memory peak at each stage, and total CPU running time as well as memory peak.
- Resource: Includes resource statistics and device utilization statistics.
- Timing: Includes Clock Summary, Max Frequency Summary, Detail Timing Paths Informations.

6.2 Place & Route Report

The Place & Route Report describes the resource, memory consumption, time consumption, etc. occupied by the user design, with the suffix .rpt.html, and you can check *.rpt.html file for further details.

Double-click "Place & Route Report" in the Process View to open the Place & Route report, as shown in Figure 6-2.

Figure 6-2 Place & Route Report



The details are as follows.

- PnR Messages: Includes report name, path and name of design, physical constraints file, timing constraints file, software version, device information, report creation time and declaration.
- PnR Details:
  - The time used in each stage of place and the total time of place, including the time of GAO place if there is GAO in the project.
  - The time used in each stage of route and the total time of route, including the time of GAO route if there is GAO in the project.
  - The time used to generate the output file.
- Resource:

- Resource Usage Summary: Device resources utilization in user design.
- I/O BANK0 Usage Summary: I/O BANK0 in user design
- Global Clock Usage Summary: Global clock used
- Global Clock Signals: Clock signals used in the user design
- Pinout by Port Name: Pinout in the user design
- All Package Pins: Details of all the pins in the device package

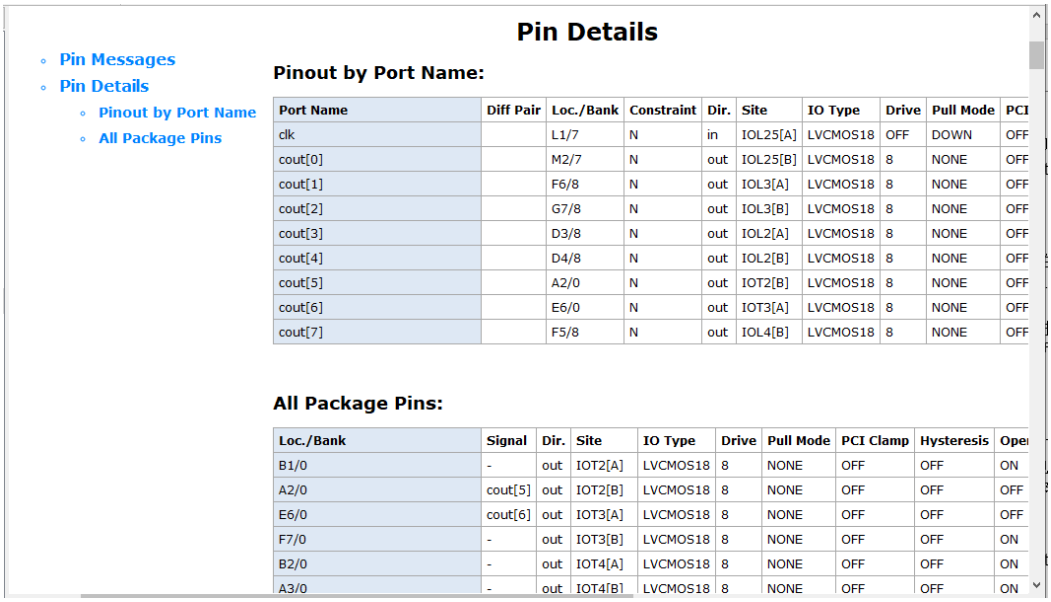
If the project has a GAO, it also includes the GAO Resource Usage Summary.

6.3 Ports and Pins Report

The Ports and Pins Report is the ports and pins files after placement. It includes ports type, attributes, and locations, etc. The generated file is saved with .pin.html suffix, and you can view .html file for details.

Double-click "Ports & Pins Report" in the Process View to open the report, as shown in Figure 6-3.

Figure 6-3 Ports & Pins Report



The details are as follows:

- Pin Messages: Includes report name, path and name of design, physical constraints file, timing constraints file, software version, device information, report creation time and declaration.
- Pin Details:
  - Pinout by Port Name: Pinout in the user design
  - All Package Pins: Details of all the pins in the device package

Note!

For devices other than GW1N-1P5/GW1N-2/GW1NR-2 and GW2AN-18X/GW2AN-9X, if no Bank V_{CCIO} constraint is added, the voltage values corresponding to some single-ended input port IO Type may not match the value of Bank V_{CCIO} in the ports report, which is normal. For example, the IO Type in the report is LVCMOS18, which corresponds to a voltage value of 1.8, but the Bank V_{CCIO} is 1.2

6.4 Timing Report

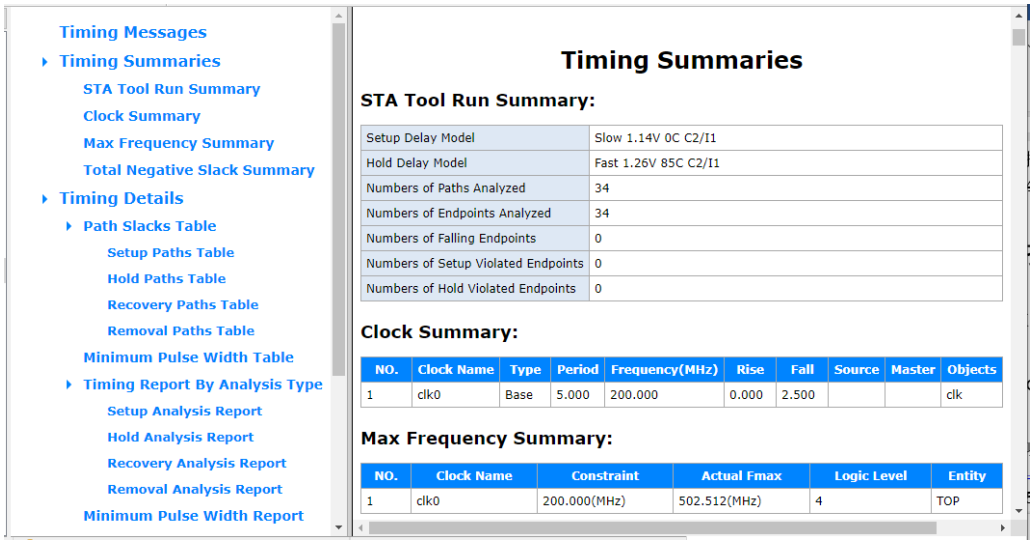
The timing report performs a thorough analysis of the timing model in the circuit netlist, calculates the timing path delays in the circuit, and determines whether they are met the requirements. The Timing report includes setup check, holdup check, restoring and removal time check, Min. clock pulse check, max. fanout path, Place & Route congestion report, etc. by default, and provides the Max. frequency report.

Double-click "Timing Analysis Report" in the Process View to open the timing analysis report for the project, as shown in Figure 6-4.

Note!

For the details, see *SUG940, Gowin Design Timing Constraints User Guide*.

Figure 6-4 Timing Report



6.5 Power Analysis Report

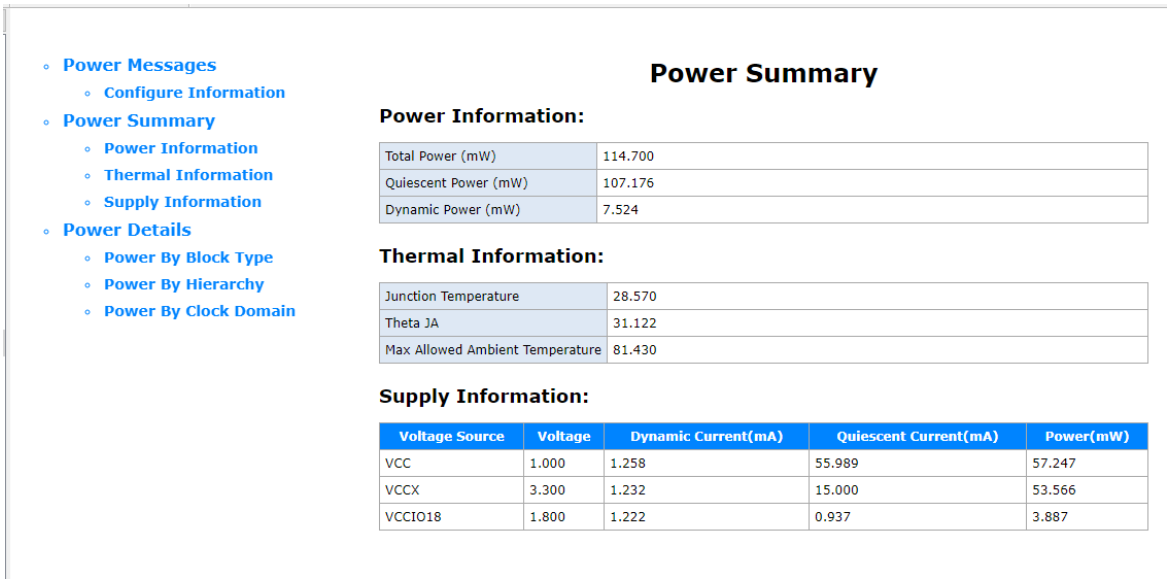
The Power Analysis Report mainly includes the power consumption estimation for your design. It is designed to help you evaluate the basic power consumption of your design.

Double-click "Power Analysis Report" in the Process View to open the analysis report, as shown in Figure 6-5.

Note!

For the details, see *SUG282, Gowin Power Analyzer User Guide*

Figure 6-5 Power Analysis Report



- Power Messages
  - Configure Information
- Power Summary
  - Power Information
  - Thermal Information
  - Supply Information
- Power Details
  - Power By Block Type
  - Power By Hierarchy
  - Power By Clock Domain

Power Summary

Power Information:

Total Power (mW)	114.700
Quiescent Power (mW)	107.176
Dynamic Power (mW)	7.524

Thermal Information:

Junction Temperature	28.570
Theta JA	31.122
Max Allowed Ambient Temperature	81.430

Supply Information:

Voltage Source	Voltage	Dynamic Current(mA)	Quiescent Current(mA)	Power(mW)
VCC	1.000	1.258	55.989	57.247
VCCX	3.300	1.232	15.000	53.566
VCCIO18	1.800	1.222	0.937	3.887

7 Simulation Files

Gowin Software provides input files for simulation. Simulation includes function simulation and timing simulation. Function simulation, also known as pre simulation, is to verify whether the circuit meets the design requirements, and it is characterized by not considering the circuit gate delay and net delay.

Timing simulation, also known as post simulation or post PnR simulation, is the process of verifying whether a circuit can meet the design under certain timing conditions and whether there are timing violations, taking into account the effects of the circuit path delay and gate delay after the circuit has been mapped to a specific process environment.

7.1 Function Simulation Files

Function simulation includes user RTL design simulation and post-synthesis netlist simulation. Take the design described in Verilog language as an example, the files required: user design RTL file before synthesis, netlist file after synthesis (*.vg), stimulus file (testbench) *tb.v and functional simulation library file prim_sim.v.

Note!

- The directory where the simulation library files are located: installPath\IDE\simlib.
- The name of the simulation library file in Vhdl format is prim_sim.vhd.
- Since the generated soft IP is encrypted, when there is an IP in the design, you need to use the .vo/.vho file after soft IP generation as the function simulation file. These .vo/.vho files are located in the src\ipName under the current project directory src.

7.2 Timing Simulation Files

Taking the timing simulation described in Verilog as an example, the required files include logical netlist file *.vo, delay file *.sdf, stimulus file *tb.v, and the timing simulation library prim_tsim.v.

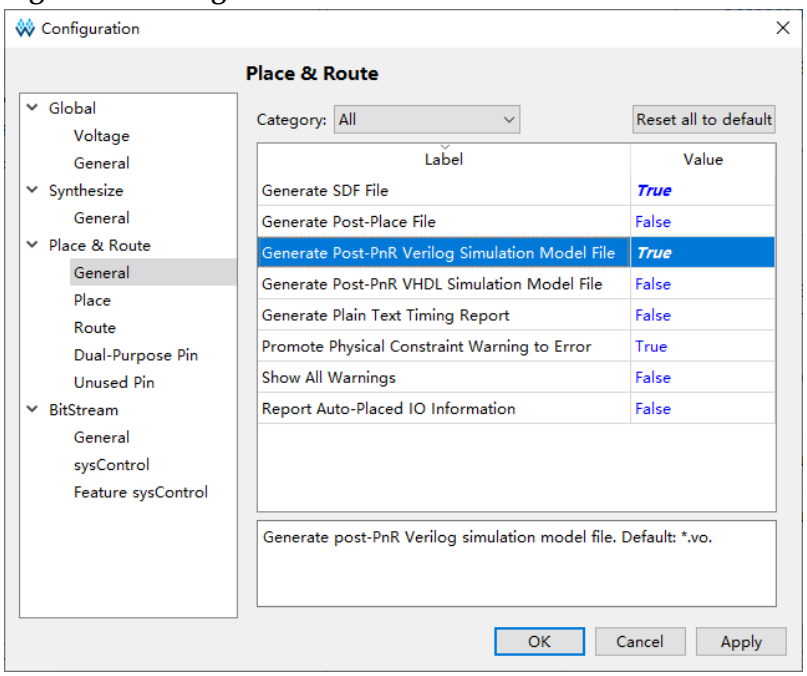
Note!

The timing precision of delay data in the SDF file is 1 ps.

The *.vo and *.sdf can be generated in the running project. The specific steps are as follows.

1. After creating/opening the project, the Value of "Generate SDF File" and "Generate Post-PNR Verilog Simulation Model File" is set to True in the Place & Route option, then click "OK", as shown in Figure 7-1.

Figure 7-1 Configuration



2. Run Place & Route, and you can find the vo and sdf files under the project path of impl/pnr/.

8 Tcl Commands

Gowin Software supports command line mode. In the following description, the content in <> must be specified, and the content in [] is optional. When creating file names, special characters such as /, *, - and space are not supported. Additionally, when specifying an IP module_name, names starting with a digit are prohibited.

8.1 Start Command Line Mode

8.1.1 gw_sh.exe

Syntax

Command: Take Windows system as an example, start \x.x\IDE\bin\gw_sh.exe under the installation directory.

Parameter:

[script file]

Parameter is null: Enter the command line console mode directly.

script file: Execute a specified script file, optional.

Application Examples:

Start Command Line mode

gw_sh.exe

#Execute script file; for the examples, you can see the section Tcl Quick Start in *SUG918, Gowin Software Quick Start Guide*.

gw_sh.exe script_file

8.2 Command

8.2.1 Command Type

IPFlow

Project

8.2.2 Command List

IPFlow

create_ipc
generate_target
get_ips
list_property
read_ipc
report_property
set_property
source
write_ip_tcl

Project

add_file
create_project
import_files
open_project
rm_file
run
run close
saveto
set_device
set_file_enable
set_file_prop
set_option
source

8.3 Command Description

8.3.1 add_file

Adds design files to the project.

Syntax

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

Usage

Name	Description
[-type]	Specifies the type of design file to add
[-disable]	Disables the file
[-h/--help]	Displays help information
<file>	File(s) to add

Type

Project

Description

When adding design files, two types of file path separators / or \\ are supported on Windows and Linux systems. Both relative and absolute paths are supported. In Gowin Software, the relative path is relative to the current project directory, and in command line mode, it is relative to the path from which gw_sh is launched.

Parameter

- <file>: The design file(s) to be added. Multiple files can be specified, separated by spaces.
- [-type <type>]: The add_file command will automatically determine the file type based on the file extension. Alternatively, this option can be used to explicitly specify the file type. Supported file types include verilog, vhdl, sv, vg, cst, sdc, gao, gpa, gsc, etc.
- [-disable]: Disables the added files. The disabled files are only added to the file list and are excluded from implementation processes. Related command: set_file_enable.
- [-h/--help]: Displays help information.

Example

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

8.3.2 create_ipc

Creates IPC file with default configuration.

Syntax

create_ipc -name <ipName> -module_name <moduleName> [-language <arg>] [-file_name <fileName>] [-dir <path>] [-force]

Usage

Name	Description
-name	IP name
-module_name	The module name of the IP to be created
[-language]	Language for the IP file, template file, and simulation file
[-file_name]	IP file name
[-dir]	The path to generate the IP
[-force]	Overwrites existing files

Type

IPFlow

Description

This command is used to create an IPC file for the IP core in the IP Core Generator.

Parameter

- -name <ipName>: Specifies the name of the IP, and the name can be obtained from the IP Core Generator.
- -module_name <moduleName>: Specifies the module name of the IP to be created.
- [-language <arg>]: Specifies the language of the generated IP files, template files, and simulation files (Verilog/VHDL). If not specified, the default is Verilog.
- [-file_name <fileName>]: Specifies the name of the generated IP file. If not specified, it will be the same as the name specified by the module_name.
- [-dir <path>]: Specifies the directory path to generate the IP. If not specified, it will be generated into the current project's src folder.
- [-force]: Overwrite existing files.

Example

In the following example, the IP core specified by -name is created in the current project's src folder, and the module name, language, and file name is specified.

create_ip -name fifo -module_name FIFO_Top -language Verilog -file_name fifo

See also

generate_target

8.3.3 create_project

Creates a project.

Syntax

create_project [-name <prjName>] [-dir <path>] [-pn <pnName>] [-device_version <arg>] [-force] [-h/--help]

Usage

Name	Description
[-name]	The name of the project to be created
[-dir]	The path to generate the project
[-pn]	The part number for the project to be created
[-device_version]	The device version for the project to be created
[-force]	Overwrites existing files
[-h/--help]	Displays help information

Type

Project

Description

Creates a project. For the file path format, see 8.3.1 add_file.

Parameter

- [-name <prjName>]: Specifies the name of the project to be created.
- [-dir <path>]: Specifies the path where the project will be created. If the specified path does not exist, a new path will be created. If this option is not specified, the default will be the current working path or the path where the TCL script is located.
- [-pn <pnName>]: Specifies the part number for the project to be created.
- [-device_version <arg>]: Specifies the device version for the part number. For devices with only an initial version, the value of the device version will be NA.
- [-force]: Overwrites existing projects.
- [-h/--help]: Displays help information.

Example

create_project -name prj0 -dir D:/tclprj -pn GW1N-UV4LQ144C6/15 -device_version B

create_project -name prjlest -pn GW1N-UV4LQ144C6/15 -

device_version B -force

8.3.4 generate_target

Generates target files for specified objects.

Syntax

generate_target <objects> [-force]

Usage

Name	Description
<objects>	Specifies the object to generate the target file
[-force]	Overwrite existing files

Type

IPFlow

Description

This command is used to generate target files for the specified IP objects (get_ips) and adds the IP core design files to the current project.

Parameter

- <objects>: Specifies one or more IP objects to generate design files. An object is specified using [get_ips module_name]. Multiple objects are specified using [get_ips module_name0 module_name1 …].
- [-force]: Overwrites existing files.

Example

In the following example, design files and template files for the specified IP are generated, and load them into the current project.

generate_target [get_ips FIFO_Top]

See also

- generate_target
- create_ipc
- read_ipc

8.3.5 get_ips

Specifies IP objects.

Syntax

get_ips <module_name>

Usage

Name	Description
<module_name>	Specifies the module name of the IP

Type

IPFlow

Description

Specifies the IP objects in the current project.

Parameter

<module_name>: Specifies the module name of the IP object, which can be one or multiple.

Example

In the following example, a single IP object in the current project is specified.

get_ips FIFO_Top

In the following example, multiple IP objects in the current project are specified.

get_ips FIFO_Top FIFO_Top_1

See also

- generate_target
- list_property
- report_property
- set_property

8.3.6 import_files

Copies files or directories to the current project.

Syntax

import_files [-file <file>] [-dir <path>] [-fileList <fileList>] [-force] [-h/--help]

Usage

Name	Description
[-file]	Specifies the file(s) to copy
[-dir]	Specifies the path to copy from
[-fileList]	Specifies a file list.
[-force]	Overwrites existing files with the same name
[-h/--help]	Displays help information

Type

Project

Description

Copy files or directories to the current project path /src. The values for -file, -dir, and -fileList can be absolute or relative paths. If a tcl script is executed via gwsh.exe, the relative path is based on the tcl script's path. If the import_files command is executed via the console window in Gowin Software, the relative path is based on the current working directory (pwd). If no options are specified after import_files, it defaults to copying all files

specified by the add_file command to the project path /src.

Parameter

- [-file <file>]: Adds one or more files to the project path /src.
- [-dir <path>]: Adds all files and subfolders in the path to the project path /src.
- [-fileList <fileList>]: Specifies a file list where each line represents a file to be added to the project. Examples are as follows:
  - D:/test1.v
  - D:/test2.v
  - This option allows adding all the files specified in each line of the file to the project path /src.
- [-force]: Overwrites files with the same name in the project path /src.
- [-h/--help]: Displays help information.

Example

import_files -file D/test1 .v -force

import_files -file D:/test1 .v D:/test2.v -force

import_files -dir D:/sourceFile

import_files -fileList log, and the file contents are as follows:

D:/Test1.v

D:/Test2.v

8.3.7 list_property

Lists the properties of an object.

Syntax

list_property <object>

Usage

Name	Description
<object>	The object that queries properties.

Type

IPFlow

Description

Obtains a list of all options for the specified IP object.

Parameter

<object>: Specifies the IP object that queries option. Use [get_ips module_name] to specify.

Example

In the following example, all properties of the specified IP object are

listed.

list_property [get_ips FIFO_Top]

See also

- report_property
- set_property

8.3.8 open_project

Opens a project.

Syntax

open_project <file>] [-pn] [-device_version] [-h/--help]

Usage

Name	Description
<file>	Specifies the project file
[-pn]	Specifies part number
[-device_version]	Specifies device version
[-h/--help]	Displays help information

Type

Project

Description

This command is used to open a project, and allows specifying a new part number for the opened project. For the file path format, see 8.3.1 add_file.

- <file>: Specifies the name of the project file to be opened.
- [-pn]: Specifies the part number for the project.
- [-device_version]: Specifies the device version for the project.
- [-h/--help]: Displays help information.

Example

open_project D:\test.gprj

8.3.9 read_ipc

Reads ipc files.

Syntax

read_ipc <file>

Usage

Name	Description
<file>	IPC files

Type

IPFlow

Description

Read the specified IPC file and configure options through the IPC file. For the file path format, see 8.3.1 add_file.

Parameter

<file>: Specifies the name of the IPC file to be opened.

Example

In the following example, an IPC file is read from the specified path.

read_ipc D:/gowin_project/src/fifo/fifio.ipc

See also

generate_target

8.3.10 report_property

Reports properties of an object.

Syntax

report_property <object>

Usage

Name	Description
<object>	The object that queries properties.

Type

IPFlow

Description

This command is used to obtain the option names, types, and values for the specified IP object.

Parameter

<object>: Specifies the IP object that queries option. Use [get_ips module_name] to specify.

Example

In the following example, all the properties of a specified IP object are listed.

report_property [get_ips FIFO_Top]

See also

- set_property
- list_property

8.3.11 rm_file

Removes design files.

Syntax

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

Usage

Name	Description
<-files>	Design files to be removed
[-h/--help]	Displays help information

Type

Project

Description

Removes design files; and for the file path format, see 8.3.1 add_file.

Parameter

- <-files> Specifies the design files to be removed. Multiple files can be specified, separated by spaces.
- [-h/--help]: Displays help information.

Example

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

8.3.12 run

Runs flows.

Syntax

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

Usage

Name	Description
<syn/pnr/all>	Specifies the name of the flow to be run
[-h/--help]	Displays help information

Type

Project

Description

Runs a specified flow or all flows.

Parameter

- <syn/pnr/all>: Specifies the name of the flow to be run. The available

flow names are "syn" (synthesis), "pnr" (place and route), or you can specify all to run all flows.

- [-h/--help]: Displays help information

Example

run pnr

run all

8.3.13 run close

Closes a project.

Syntax

run close

Type

Project

Description

Closes the current project.

Example

run close

8.3.14 saveto

Saves project data to a tcl script.

Syntax

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

Usage

Name	Description
[-all_options]	Saves all option information
[-h/--help]	Displays help information
<file>	The file to be saved

Type

Project

Description

Saves the current project design data to a tcl script. For the file path format, see 8.3.1 add_file.

Parameter

- [-all_options]: By default, the saveto command only saves the modified options, i.e., those different from the default values. By using -all_options, all option information will be saved.
- [-h/--help]: Displays help information.
- <file>: The file to be saved.

Example

saveto project.tcl

saveto -all_options project.tcl

saveto -all_options D:/gowin_project/project.tcl

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

8.3.15 set_device

Sets device part number.

Syntax

set_device [-device_version <value>] [-h/--help] <part number>

Usage

Name	Description
[-device_version <value>]	Sets device version
[-h/--help]	Displays help information
<part number>	Sets device part number

Type

Project

Description

Sets device part number.

Parameter

- <part number>: Specifies the part number of the target device, such as GW1N-UV4LQ144C6/I5.
- [-device_version <value>]: Specifies the device version. Supported values are NA|B|C|D.
- [-h/--help]: Displays help information.

Example

set_device GW1N-LV1CS30C6/I5

set_device –device_version C GW1N-UV4LQ144C6/I5

8.3.16 set_file_enable

Sets file enable properties.

Syntax

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

Usage

Name	Description
<file>	Specifies the design file to be set
<true|false>	Whether the file can be used or not

Name	Description
[-h/--help]	Displays help information

Type

Project

Description

Sets whether the file can be used. For the file path format, see 8.3.1 add_file.

Parameter

- <file>: Specifies the file to be set.
- <true|false>: true means the file can be used, and false means it cannot.
- [-h/--help]: Displays help information.

Example

set_file_enable top.v false

set_file_enable D:/gowin_project/top.v

set_file_enable D:\\gowin_project\\top.v

8.3.17 set_file_prop

Sets file properties.

Syntax

set_file_prop <file> [-lib <name>] [-h/--help]

Usage

Name	Description
<file>	Specifies the design files to be set
[-lib <name>]	Sets the library name for the file
[-h/--help]	Displays help information

Type

Project

Description

Sets properties for the specified files. For the file path format, see 8.3.1 add_file.

Parameter

- <file>: Specifies the file(s) to be set. Multiple files can be listed, separated by spaces.
- [-lib <name>]: Sets the library name for the file. This option is only valid for VHDL files.
- [-h/--help]: Displays help information.

Example

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

8.3.18 set_csr

Specifies the CSR file.

Syntax

set_csr [-h/--help] <file>

Usage

Name	Description
<file>	The CSR file to be specified.
[-h/--help]	Displays help information

Type

Project

Description

Specifies the CSR file. For the file path format, see 8.3.1 add_file.

Parameter

- <files>: Specifies the CSR file.
- [-h/--help]: Displays help information.

Example

set_csr a.csr

set_csr D:/gowin_project/a.csr

set_csr D:\\gowin_project\\a.csr

8.3.19 set_option

Sets project-related property configurations and flow options.

Syntax

set_option [options] [-h/--help]

Usage

Name	Description
[options]	Specifies configurations or flow options
[-h/--help]	Displays help information

Type

Project

Description

Sets project-related property configurations and flow options.

Parameter

- [options]: Specifies configurations or flow options.
- [-h/--help]: Displays help information.

Global Property Configuration

-output_base_name

Specify the output file name.

Syntax

-output_base_name <name>

Usage

Name	Description
<name>	Specifies the output file name

Type

Project

Description

Specifies the name of the output file. This option is only used to set the base name, and different flows will append appropriate extensions based on the output file type. For example, if -output_base_name abc is set, the netlist file generated by Gowin Synthesis will be named abc.vg.

Parameter

<name>: Specifies the output file name.

Example

set_option -output_base_name abc

-global_freq

Specify frequency value

Syntax

-global_freq <default|value>

Usage

Name	Description
<default|value>	Specifies frequency value

Type

Project

Description

Specifies the frequency value. The default setting is: 50 MHz for the LittleBee family, 100 MHz for Arora family.

Parameter

<default|value>frequency value

Example

set_option -global_freq 80

Synthesis Property Configuration

-synthesis_tool

Specifies Synthesis tool.

Syntax

-synthesis_tool <tool>

Usage

Name	Description
<tool>	Specifies Synthesis tool

Type

Project

Description

Specifies Synthesis tool GowinSynthesis.

Parameter

<tool>: Specifies Synthesis tool GowinSynthesis.

Example

set_option -synthesis_tool GowinSynthesis

-top_module

Specifies Top Module/Entity.

Syntax

-top_module <name>

Usage

Name	Description
<name>	Specifies top module

Type

Project

Description

Specifies top module.

Parameter

<name>: Specifies top module.

Example

set_option -top_module test

-include_path

Specifies Include Path.

Syntax

-include_path <path or path list>

Usage

Name	Description
<path or path list>	Specifies Include Path

Type

Project

Description

Specifies the include path. When specifying multiple include paths, separate them with semicolons (;) and enclose all paths within curly braces {}. For example: -include_path {/path1;/path2;/path3}. Both relative and absolute paths are supported. A relative path is interpreted relative to the program's current working directory.

Parameter

<path or path list>: Specifies Include Path.

Example

set_option -include_path D:/project

-verilog_std

Specifies Verilog language standard.

Syntax

-verilog_std<v1995|v2001|sysv2017>

Usage

Name	Description
<v1995|v2001|sysv2017>	Specifies Verilog language standard

Type

Project

Description

Specifies Verilog language standard: Verilog 95/Verilog 2001/System Verilog 2017, and the default is Verilog 2001.

Parameter

<v1995|v2001|sysv2017>: Specifies Verilog language standard.

Example

set_option -verilog_std v1995

-vhdl_std

Specifies VHDL language standard.

Syntax

-vhdl_std <vhd1993|vhd2008|vhd2019>

Usage

Name	Description
< vhd1993|vhd2008|vhd2019>	Specifies VHDL language standard

Type

Project

Description

Specifies VHDL language standard: VHDL 1993/VHDL 2008/VHDL 2019, and the default is VHDL1993.

Parameter

<vhd1993|vhd2008|vhd2019>: Specifies VHDL language standard.

Example

set_option -vhdl_std vhd2008

-print_all_synthesis_warning <0|1>

Specifies whether to print all synthesis warning messages. The default is 0.

Syntax

-print_all_synthesis_warning <0|1>

Usage

Name	Description
<0|1>	0: Do not print all warning messages. 1: Print all warning messages.

Type

Project

Description

Specifies whether to print all synthesis warning messages. The default is 0.

Parameter

<0|1>: Specifies whether to print all synthesis warning messages.

Example

set_option -print_all_synthesis_warning 1

-disable_io_insertion

Enables or disables I/O insertion.

Syntax

-disable_io_insertion <0|1>

Usage

Name	Description
<0|1>	0: Enables I/O insertion 1: Disables I/O insertion

Type

Project

Description

Enables or disables I/O insertion. The default is 0.

Parameter

<0|1>: Enables or disables I/O insertion.

Example

set_option -disable_io_insertion 1

-looplimit <value>

The loop limit value of default compiler in RTL.

Syntax

-looplimit <value>

Usage

Name	Description
<value>	Looplimit value

Type

Project

Description

The loop limit value of default compiler in RTL, with the default value being 2000.

Parameter

<value>: The loop limit value of default compiler in RTL.

Example

set_option -looplimit 1000

-maxfan <value>

Set fanout value.

Syntax

-maxfan <value>

Usage

Name	Description
<value>	Maxfan value

Type

Project

Description

Set the fanout value for input ports, nets, or register output ports, with the default value being 10000.

Parameter

<value>: The fanout value for input ports, nets, or register output ports.

Example

set_option -maxfan 5000

-rw_check_on_ram

Insert bypass logic around RAM.

Syntax

-rw_check_on_ram <0|1>

Usage

Name	Description
<0|1>	0: Disable 1: Enable

Type

Project

Description

If there are read or write conflicts in RAM, enabling this option will insert bypass logic around the RAM to prevent simulation mismatches, with the default being 0.

Parameter

<0|1>: Enables or disables the insertion of bypass logic around RAM.

Example

set_option -rw_check_on_ram 1

Place & Route Property Configuration

-vccx

Specifies vccx value.

Syntax

-vccx <value>

Usage

Name	Description
<value>	Specifies vccx value

Type

Project

Description

Specifies vccx value.

Parameter

<value>: Specifies vccx value.

Example

set_option -vccx 3.3

-vcc

Syntax

-vcc <value>

Usage

Name	Description
<value>	Specifies vcc value

Type

Project

Description

Specifies vccx value.

Parameter

<value>: Specifies vccx value.

Example

set_option -vcc 3.3

-gen_sdf

Whether to generate the SDF file.

Syntax

-gen_sdf <0|1>

Usage

Name	Description
<0|1>	0: Do not generate the SDF file. 1: Generates the SDF file.

Type

Project

Description

Specifies whether the Place & Route generates an SDF, and the default is 0.

Parameter

<0|1>: Enable control for generating the SDF file.

Example

set_option -gen_sdf 1

-gen_io_cst

Whether to generate the physical constraint file for the port, named *.io.cst.

Syntax

-gen_io_cst <0|1>

Usage

Name	Description
<0|1>	0: Does not generate the *.io.cst file. 1: Generates the *.io.cst file.

Type

Project

Description

Specifies whether the Place & Route generates the physical constraint file for the port, named *.io.cst. The default is 0.

Parameter

<0|1>: Generates physical constraints file enable control for port.

Example

set_option -gen_io_cst 1

-gen_ibis

Specifies whether to generate the input/output buffer information file named *.ibs.

Syntax

-gen_ibis <0|1>

Usage

Name	Description
<0|1>	0: Do not generate *.ibs file. 1: Generates *.ibs file.

Type

Project

Description

Specifies whether Place & Route generates the input/output buffer information file named *.ibs.

Parameter

<0|1>: Generates the input/output buffer information file enable control.

Example

set_option -gen_ibis 1

-gen_posp

Whether to generate a device placement file.

Syntax

-gen_posp <0|1>

Usage

Name	Description
<0|1>	0: Do not generate *.posp file. 1: Generates *.posp file.

Type

Project

Description

Specifies whether Place & Route generates a device placement file named *.posp, which contains only BSRAM placement information. The default value is 0.

Parameter

<0|1>: Generates device placement file enable control.

Example

set_option -gen_posp 1

-gen_text_timing_rpt

Generate a timing report in text format.

Syntax

-gen_text_timing_rpt <0|1>

Usage

Name	Description
<0|1>	0: Do not generate *.tr file. 1: Generates *.tr file.

Type

Project

Description

Specifies whether Place & Route generates a timing report in text format with the filename *.tr. The default value is 0.

Parameter

<0|1>: Generates a text-format timing report enable control.

Example

set_option -gen_text_timing_rpt 1

-gen_verilog_sim_netlist

Specifies whether to generate a Verilog timing simulation model file.

Syntax

-gen_verilog_sim_netlist <0|1>

Usage

Name	Description
<0|1>	0: Do not generate *.vo file. 1: Generates *.vo file.

Type

Project

Description

Specifies whether Place & Route generates a Verilog timing simulation model file named *.vo, and the default is 0.

Parameter

<0|1>: Generates enable control for the Verilog timing simulation model file.

Example

set_option -gen_verilog_sim_netlist 1

-gen_vhdl_sim_netlist

Generates VHDL timing simulation model file or not.

Syntax

-gen_vhdl_sim_netlist <0|1>

Usage

Name	Description
<0|1>	0: Do not generate *.vho file. 1: Generates *.vho file.

Type

Project

Description

Specifies whether Place & Route generates a VHDL timing simulation model file named *.vho, and the default is 0.

Parameter

<0|1>: Generates enable control for the VHDL timing simulation model file.

Example

set_option -gen_vhdl_sim_netlist 1

-show_init_in_vo

Adds default initial values to the instances in the timing simulation model file.

Syntax

-show_init_in_vo <0|1>

Usage

Name	Description
<0|1>	0: Do not add default initial values to the instances in the timing simulation model file. 1: Adds default initial values to the instances in the timing simulation model file.

Type

Project

Description

Adds default initial values to the instances in the generated Place & Route timing simulation model file, and the default is 0.

Parameter

<0|1>: Enable control for adding default initial values to the instances in the generated Place & Route timing simulation model file.

Example

set_option -show_init_in_vo 1

-show_all_warn

Whether to show all warnings.

Syntax

-show_all_warn<0|1>

Usage

Name	Description
<0|1>	0: Do not output all warning messages during Placement & Routing. 1: Outputs all warning messages during Placment & Routing.

Type

Project

Description

Outputs all warning messages when running Place & Route, and the default is 0.

Parameter

<0|1>: Enable control for outputting all warning messages when running Place & Route.

Example

set_option -show_all_warn 1

-timing_driven

Whether optimizes timing driven for Place & Route.

Syntax

-timing_driven <0|1>

Usage

Name	Description
<0|1>	0: Disables Place & Route timing driven. 1: Enables Place & Route timing driven.

Type

Project

Description

Timing-driven optimization is performed when running Place & Route. The default value is 1.

Parameter

<0|1>: Enable control for timing-driven optimization when running Place & Route.

Example

set_option -timing_driven 1

-cst_warn_to_error

Elevates physical constraint warnings to errors.

Syntax

-cst_warn_to_error <0|1>

Usage

Name	Description
<0|1>	0: Physical constraint warnings will not be elevated to errors. 1: Physical constraint warnings will be elevated to errors.

Type

Project

Description

Elevates physical constraint warnings to errors when running Place & Route. The default value is 1.

Parameter

<0|1>: Enable control for elevating physical constraint warnings to errors when running Place & Route.

Example

set_option -cst_warn_to_error 1

-rpt_auto_place_io_info

Reports auto-placed IO location information.

Syntax

-rpt_auto_place_io_info <0|1>

Usage

Name	Description
<0|1>	0: Auto-placed IO location information will not be reported. 1: Auto-placed IO location information will be reported.

Type

Project

Description

Reports auto-placed IO location information when running Place & Route. The default value is 0.

Parameter

<0|1>: Enable control for reporting auto-plzaced IO location information.

Example

set_option -cst_warn_to_error 1

-place_option

Placement algorithm option.

Syntax

-place_option <0|1|2|3|4>

Usage

Name	Description
<0|1|2|3|4>	0: Uses the default placement algorithm. 1: Uses placement algorithm 1. 2: Uses placement algorithm 2. 3: Uses placement algorithm 3. 4: Uses placement algorithm 4.

Type

Project

Description

Placement algorithm option. The default value is 0.

Parameter

<0|1|2|3|4>: Placement algorithm option。

Example

set_option -place_option 1

-route_option

Routing algorithm option.

Syntax

-route_option <0|1|2>

Usage

Name	Description
<0|1|2>	0: Uses the default routing algorithm. 1: Uses the default routing algorithm 1. 2: Uses the default routing algorithm 2.

Type

Project

Description

Routing algorithm option. The default value is 0.

Parameter

<0|1|2>: Routing algorithm option.

Example

set_option -route_option 1

-ireg_in_iob

Place registers connected to input buffers to IOB.

Syntax

-ireg_in_iob <0|1>

Usage

Name	Description
<0|1>	0: Do not place registers connected to input buffers to IOB. 1: Place registers connected to input buffers to IOB.

Type

Project

Description

Enabling this option will cause Place & Route to place registers connected to input buffers to IOB. The default value is 1.

Parameter

<0|1>: Enable control for placing registers connected to input buffers to IOB.

Example

set_option -ireg_in_iob 1

-oreg_in_iob

Place registers connected to output/tristate buffers to IOB.

Syntax

-oreg_in_iob <0|1>

Usage

Name	Description
<0|1>	0: Do not place registers connected to output/tristate buffers to IOB. 1: Place registers connected to output/tristate buffers to IOB.

Type

Project

Description

Enabling this option will cause Place & Route to place registers connected to output/tristate buffers to IOB. The default value is 1.

Parameter

<0|1>: Enable control for place registers connected to output/tristate buffers to IOB.

Example

set_option -oreg_in_iob 1

-ioreg_in_iob

Place registers connected to bidirectional buffers to IOB.

Syntax

-ioreg_in_iob <0|1>

Usage

Name	Description
<0|1>	0: Do not place registers connected to bidirectional buffers to IOB. 1: Place registers connected to bidirectional buffers to IOB.

Type

Project

Description

Enabling this option will cause Place & Route to place registers connected to bidirectional buffers in IOB. The default value is 1.

Parameter

<0|1>: Enable control for placing registers connected to bidirectional buffers to IOB.

Example

set_option -ioreg_in_iob 1

-replicate_resources

Replicate resources for high fanout to reduce fanout and improve timing.

Syntax

-replicate_resources <0|1>

Usage

Name	Description
<0|1>	0: Do not replicate resources for high fanout to reduce fanout. 1: Replicate resources for high fanout to reduce fanout.

Type

Project

Description

Enabling this option will cause Place & Route to replicate resources for high fanout to reduce fanout and improve timing results. The default value is 0.

Parameter

<0|1>: Enable control for replicate resources for high fanout to reduce fanout.

Example

set_option -replicate_resources 1

-clock_route_order

Specifies the routing order for clock wires other than those generated by Clock primitives. The optional values are 0 and 1, with the default value being 0.

Syntax

-clock_route_order <0|1>

Usage

Name	Description
<0|1>	0: Route clock wires in the order of net fanout, from highest to lowest. 1: Route clock wires in the order of frequency, from highest to lowest.

Type

Project

Description

Specifies the routing order for clock wires other than those generated by clock primitives.

Parameter

<0|1>: Specifies the routing order for clock wires other than those generated by clock primitives.

Example

set_option -clock_route_order 1

-route_maxfan

Sets maximum fanout for routing.

Syntax

-route_maxfan <value>

Usage

Name	Description
< value>	Sets maximum fanout for routing.

Type

Project

Description

Based on routing optimization, this option is used to set the maximum fanout for routing. The value should be an integer greater than 0 and less than or equal to 100. For GW1NZ-1/GW1N-2/GW1NR-2/GW1N-1P5 devices, the default value of this option is 10, while for other devices, the default value is 23.

Parameter

<value>: Sets maximum fanout for routing.

Example

set_option -route_maxfan 60

-correct_hold_violation

Automatic hold violation fixing for routing.

Syntax

-correct_hold_violation <0|1>

Usage

Name	Description
<0|1>	0: Do not automatically fix hold violations during routing. 1: Automatically fix hold violations during routing.

Type

Project

Description

Enabling this option allows automatically fixing hold timing violations during routing. The default value is 1.

Parameter

<0|1>: Enable control for automatically fixing hold timing violations during routing

Example

set_option -correct_hold_violation 1

-inc_place <0|auto|file>

Incremental placement.

Syntax

-inc_place <0|auto|file>

Usage

Name	Description
<0|auto|file >	0: Disables incremental placement. auto: Automatic incremental placement. file: Specifies a *.p file for incremental placement.

Type

Project

Description

Enabling this option allows incremental placement. The default value is 0.

Parameter

<0|auto|file >: Enable control for incremental placement.

Example

set_option -inc_place auto

- -inc_pnr <0|auto|file>

Incremental place & route.

Syntax

-inc_pnr <0|auto|file>

Usage

Name	Description
<0|auto|file >	0: Disables incremental place & route. auto: Automatic incremental place & route. file: Specifies a *.p file for incremental place & route.

Type

Project

Description

Enabling this option allows incremental place & route. The default value is 0.

Parameter

<0|auto|file >: Enable control for incremental place & route.

Example

set_option -inc_pnr auto

Note!

For detailed usage of Place & Route-related running, see Place & Route in Section 4.3.3 of this document.

Pin Multiplexing Property Configuration

-use_jtag_as_gpio

Multiplexes JTAG-related pins as general IO pins.

Syntax

-use_jtag_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as JTAG-dedicated pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes JTAG-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing JTAG-related pins as general IO pins.

Example

set_option -use_jtag_as_gpio 1

-use_sspi_as_gpio

Multiplexes SSPI-related pins as general IO pins.

Syntax

-use_sspi_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as SSPI-dedicated pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes SSPI-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing SSPI-related pins as general IO pins.

Example

set_option -use_sspi _as_gpio 1

-use_mspi_as_gpio

Multiplexes MSPI-related pins as general IO pins.

Syntax

-use_mspi_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as MSPI-dedicated pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes MSPI-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing MSPI-related pins as general IO pins.

Example

set_option -use_mspi _as_gpio 1

-use_ready_as_gpio

Multiplexes READY-related pins as general IO pins.

Syntax

-use_ready_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as READY-dedicated pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes READY-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing READY-related pins as general IO pins.

Example

set_option -use_ready_as_gpio 1

-use_done_as_gpio

Multiplexes DONE-related pins as general IO pins.

Syntax

-use_done_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as DONE-dedicated pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes DONE-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing DONE-related pins as general IO pins.

Example

set_option -use_ done_as_gpio 1

-use_reconfign_as_gpio

Multiplexes RECONFIG_N-related pins as general IO pins.

Syntax

-use_ reconfign_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as RECONFIG_N-related pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes RECONFIG_N-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing RECONFIG_N-related pins as general IO pins.

Example

set_option -use_reconfign_as_gpio 1

-use_i2c_as_gpio

Multiplexes I2C-related pins as general IO pins.

Syntax

-use_i2c_as_gpio <0|1>

Usage

Name	Description
<0|1>	0: Uses as I2C-related pins. 1: Multiplexes as general IO pins.

Type

Project

Description

Multiplexes I2C-related pins as general IO pins, and the default is 0.

Parameter

<0|1>: Enable control for multiplexing I2C-related pins as general IO pins.

Example

set_option -use_i2c_as_gpio 1

BitStream Property Configuration

-bit_format

Specifies the format of the generated bitstream file content.

Syntax

-bit_format <txt|bin>

Usage

Name	Description
<txt|bin>	The format of the bitstream file content.

Type

Project

Description

Used to specify the format of the generated bitstream file content.

Parameter

<txt|bin>: the format of the generated bitstream file content.

Example

set_option -bit_format txt

-bit_crc_check

Cyclic redundancy check.

Syntax

-bit_crc_check <0|1>

Usage

Name	Description
<0|1>	0: Disables cyclic redundancy check. 1: Enables cyclic redundancy check.

Type

Project

Description

Enables CRC for the bitstream file. The default is 1.

Parameter

<0|1>: Enables CRC for the bitstream file.

Example

set_option -bit_crc_check 1

-bit_compress

Compresses the bitstream file.

Syntax

-bit_compress <0|1>

Usage

Name	Description
<0|1>	0: Do not compress the bitstream file. 1: Compresses the bitstream file.

Type

Project

Description

Compresses the generated bitstream file. The default value is 1.

Parameter

<0|1>: Compresses the bitstream file.

Example

set_option -bit_compress 1

-bit_encrypt

Encrypts the bitstream file.

Syntax

-bit_encrypt <0|1>

Usage

Name	Description
<0|1>	0: Do not encrypt the bitstream file. 1: Encrypts the bitstream file.

Type

Project

Description

Encrypts the bitstream file. This feature is only supported for Arora family devices. The default value is 0.

Parameter

<0|1>: Encrypts the bitstream file.

Example

set_option -bit_encrypt 1

-bit_encrypt_key

Defines the encrypted key.

Syntax

-bit_encrypt_key <key>

Usage

Name	Description
<key>	The encrypted key.

Type

Project

Description

This option works in conjunction with the -bit_encrypt option, allowing users to define a custom encrypted key. The default value is all 0.

Parameter

<key>: The encrypted key.

Example

set_option -bit_encrypt_key 00000000000000000000000000001101

-bit_security

Security bit enable control.

Syntax

-bit_security <0|1>

Usage

Name	Description
<0|1>	0: Disables security bit. 1: Enables security bit.

Type

Project

Description

Security bit enable control, and the default is 1.

Parameter

<0|1>: Security bit enable control.

Example

set_option -bit_security 1

-bit_incl_bsram_init

Prints initial values of BSRAM to the Bitstream file.

Syntax

-bit_incl_bsram_init <0|1>

Usage

Name	Description
<0|1>	0: Do not print initial values of BSRAM to the Bitstream file； 1: Prints initial values of BSRAM to the Bitstream file.

Type

Project

Description

This option is used to control whether the initial values of BSRAM are printed into the bitstream file. The default value is 1. For GW1N and GW2A devices, when this option is set to 1, all positions of the BSRAM will have their initial values printed in the bitstream file, and the initial values of unoccupied BSRAM positions are printed as 0. For GW5A(N)(S)(R)(T) devices, when this option is set to 1, all initial values of BSRAM in the same column as the occupied BSRAM will be printed to the bitstream file, and the initial values of unoccupied BSRAM positions in that column are printed as 0.

Parameter

<0|1>: Enable control for printing initial values of BSRAM to the Bitstream file.

Example

set_option -bit_incl_bsram_init 1

-bg_programming

Background programming function.

Syntax

-bg_programming <off | jtag | i2c | goconfig | userlogic | i2c_jtag_sspi_qsspi | jtag_sspi_qsspi>

Usage

Name	Description
<off|jtag|i2c|goconfig|userlogic| i2c_jtag_sspi_qsspi|jtag_sspi_qsspi>	Off: Do not use background programming function.

Name	Description
	jtag: Background programming via JTAG mode. i2c: Background programming via I2C mode. goconfig: Background programming using goConfig IP. userlogic: Background programming using FPGA internal logic. i2c_jtag_sspi_qsspi: Background programming using I2C/JTAG/SSPI/QSSPI mode. jtag_sspi_qsspi: Background programming using JTAG/SSPI/QSSPI mode.

Type

Project

Description

It allows programming of Flash without disrupting the FPGA chip's current operations. The default setting is off.

Parameter

< off | jtag | i2c | goconfig | userlogic | i2c_jtag_sspi_qsspi | jtag_sspi_qsspi>: Background programming method.

Example

set_option -bg_programming userlogic

-hotboot

Hotboot mode.

Syntax

-hotboot <0|1>

Usage

Name	Description
<0|1>	0: Disables hotboot mode. 1: Enables hotboot mode.

Type

Project

Description

Hotboot mode enable control. The default is 0.

Parameter

<0|1>: Hotboot mode enable control.

Example

set_option -hotboot 1

-i2c_slave_addr

Sets I2C device address.

Syntax

-i2c_slave_addr <value>

Usage

Name	Description
<value>	Sets I2C device address.

Type

Project

Description

Sets the I2C device address, and the configurable range is from 00 to 7F. The default is 00.

Parameter

<value>: I2C device address.

Example

set_option -i2c_slave_addr 2F

-secure_mode

Enables secure mode.

Syntax

-secure_mode <0|1>

Usage

Name	Description
<0|1>	0: Disables secure mode. 1: Enables secure mode.

Type

Project

Description

Enables secure mode, and at this time the JTAG pins are used as GPIOs; the bitstream file can only be programmed into the device once. The default is 0.

Parameter

<0|1>: Enables control for secure mode.

Example

set_option -secure_mode 1

-loading_rate

The loading speed of bitstream data from Flash to SRAM in AutoBoot and MSPI configuration modes.

Syntax

-loading_rate <value>

Usage

Name	Description
<value>	The loading speed of bitstream data from Flash to SRAM in AutoBoot and MSPI configuration modes.

Type

Project

Description

The loading speed of bitstream data from Flash to SRAM in AutoBoot and MSPI configuration modes, and the default value is 2.500MHz.

Parameter

<value>: The loading speed of bitstream data from Flash to SRAM.

Example

set_option -loading_rate 21.000MHz

-seu_handler

Enable Single Event Upset (SEU) handler.

Syntax

-seu_handler <0|1>

Usage

Name	Description
<0|1>	0: Disables SEU handler. 1: Enables SEU handler.

Type

Project

Description

Enables SEU handler, and the default is 0.

Parameter

<0|1>: Enable control for SEU handler.

Example

set_option -seu_handler 1

-seu_handler_mode

Selects the mode for booting or stopping the SEU Handler.

Syntax

-seu_handler_mode <auto|userlogic>

Usage

Name	Description
<0|1>	auto: Automatically enables the SEU handler after the chip wakes up. userlogic: Uses logic to enable or stop the SEU handler.

Type

Project

Description

Selects the mode for booting or stopping the SEU Handler, and the default is auto.

Parameter

< auto|userlogic>: The mode for booting or stopping the SEU Handler.

Example

set_option -seu_handler_mode userlogic

-seu_handler_checksum

Enables SEU processing, checking, calculation, and comparison.

Syntax

-seu_handler_checksum <0|1>

Usage

Name	Description
<0|1>	0: Disables SEU processing, checking, calculation, and comparison. 1: Enables SEU processing, checking, calculation, and comparison.

Type

Project

Description

Enables SEU processing, checking, calculation, and comparison, and the default is 0.

Parameter

<0|1>: Enables control for SEU processing, checking, calculation, and comparison.

Example

set_option -seu_handler_checksum 1

-error_detection

Enables error detection only.

Syntax

-error_detection <0|1>

Usage

Name	Description
<0|1>	0: Disables error detection. 1: Enables error detection only.

Type

Project

Description

Enables error detection only, and the default is 0.

Parameter

<0|1>: Enables control for error detection only.

Example

set_option -error_detection 1

-error_detection_correction

Enables error detection and correction.

Syntax

-error_detection_correction <0|1>

Usage

Name	Description
<0|1>	0: Disables error detection and correction. 1: Enables error detection and correction.

Type

Project

Description

Enables error detection and correction, and the default is 0.

Parameter

<0|1>: Enables control for error detection and correction.

Example

set_option -error_detection_correction 1

-stop_seu_handler

Stops SEU Handler.

Syntax

-stop_seu_handler <0|1>

Usage

Name	Description
<0|1>	0: Do not stop the SEU Handler when an uncorrectable ECC error or CRC checksum mismatch is detected. 1: Stops the SEU Handler when an uncorrectable ECC error or CRC checksum mismatch is detected.

Type

Project

Description

Stops the SEU Handler when an uncorrectable ECC error or CRC checksum mismatch is detected. The default is 0.

Parameter

<0|1>: Enable control for stopping SEU Handler.

Example

set_option -stop_seu_handler 1

-osc_div

Sets the frequency division ratio for the extended control register.

Syntax

-osc_div <4|8|16|32>

Usage

Name	Description
<4|8|16|32>	4: Sets the frequency division ratio of the extended control register to 4. 8: Sets the frequency division ratio of the extended control register to 8. 16: Sets the frequency division ratio of the extended control register to 16. 32: Sets the frequency division ratio of the extended control register to 32.

Type

Project

Description

Sets the frequency division ratio for the extended control register, and the default is 8.

Parameter

<4|8|16|32>: The frequency division ratio for the extended control register.

Example

set_option -osc_div 8

-error_injection

Enables error injection.

Syntax

-error_injection <0|1>

Usage

Name	Description
<0|1>	0: Disables error injection. 1: Enables error injection.

Type

Project

Description

Enables error injection, and the default is 0

Parameter

<0|1>: Enables control for error injection.

Example

set_option -error_injection 1

-ext_cclk

Enables external primary configuration clock.

Syntax

-ext_cclk <0|1>

Usage

Name	Description
<0|1>	0: Disables external primary configuration clock. 1: Enables external primary configuration clock.

Type

Project

Description

Enables external primary configuration clock. The default is 0.

Parameter

<0|1>: Enable control for external primary configuration clock.

Example

set_option -ext_cclk 1

-ext_cclk_div

Sets the frequency divider parameter.

Syntax

-ext_cclk_div <value>

Usage

Name	Description
<value>	Frequency divider Parameter

Type

Project

Description

Sets the frequency divider parameter.

Parameter

<value>: Sets the frequency divider parameter.

Example

set_option -ext_cclk_div 4

-multi_boot

Multi Boot enable control.

Syntax

-multi_boot <0|1>

Usage

Name	Description
<0|1>	0: Disables Multi Boot. 1: Enables Multi Boot.

Type

Project

Description

Multi Boot enable control, and the default is 0.

Parameter

<0|1>: Multi Boot enable control.

Example

set_option -multi_boot 1

-multiboot_address_width

Configures SPI Flash address width.

Syntax

-multiboot_address_width<24|32>

Usage

Name	Description
<24|32>	24: Sets the SPI Flash address width to 24. 32: Sets the SPI Flash address width to 32.

Type

Project

Description

Configures SPI Flash address width, and the default is 24.

Parameter

<24|32>: SPI Flash address width.

Example

set_option -multiboot_address_width 32

-multiboot_spi_flash_address

Specifies SPI Flash address.

Syntax

-multiboot_spi_flash_address <value>

Usage

Name	Description
<value>	SPI Flash address.

Type

Project

Description

Specifies the SPI Flash address. The SPI Flash address refers to the start address of the bitstream file to be loaded during the next multiboot, with the default value of 000000.

Parameter

< value>: SPI Flash address.

Example

set_option -multiboot_spi_flash_address 000110

-multiboot_mode

Configures SPI Flash address access mode.

Syntax

-multiboot_mode <single | fast | dual | quad>

Usage

Name	Description
< single|fast|dual|quad >	single: Uses single mode fast: Uses fast mode dual: Uses gedual mode quad: Uses gequad mode

Type

Project

Description

Configures SPI Flash address access mode, and the default is single.

Parameter

<single | fast | dual | quad>: SPI Flash address access mode.

Example

set_option -multiboot_mode single

-mspi_jump

MSPI JUMP enable control.

Syntax

-mspi_jump<0|1>

Usage

Name	Description
<0|1>	0: Disables MSPI JUMP. 1: Enables MSPI JUMP.

Type

Project

Description

MSPI JUMP enable control, and the default is 0.

Parameter

<0|1>: MSPI JUMP enable control.

Example

set_option -mspi_jump 1

-merge_jumpbit

Merges the MSPI JUMP bitstream file into the general bitstream file.

Syntax

-merge_jumpbit <0|1>

Usage

Name	Description
<0|1>	0: Do not merge the bitstream file. 1: Merges the bitstream file.

Type

Project

Description

Merges the MSPI JUMP bitstream file into the general bitstream file，and the default is 0.

Parameter

<0|1>: Enables control for merging the MSPI JUMP bitstream file into the general bitstream file.

Example

set_option -merge_jumpbit 1

-mspijump_address_width

Configures SPI Flash address width.

Syntax

-mspijump_address_width <24|32>

Usage

Name	Description
<24|32>	24: Sets the SPI Flash address width to 24. 32: Sets the SPI Flash address width to 32.

Type

Project

Description

Configures SPI Flash address width, and the default is 24.

Parameter

<24|32>: SPI Flash address width.

Example

set_option - mspijump_address_width 32

-mspijump_spi_flash_address

Specifies SPI Flash address.

Syntax

-mspijump_spi_flash_address <value>

Usage

Name	Description
<value>	SPI Flash address

Type

Project

Description

Specifies SPI Flash address, and the default is 000000.

Parameter

<value>: SPI Flash address

Example

set_option - mspijump_spi_flash_address 000110

-mspijump_mode<single | fast | dual | quad>

Configures SPI Flash address access mode.

Syntax

-mspijump_mode <single | fast | dual | quad>

Usage

Name	Description
< single|fast|dual|quad >	single: Usagesingle mode fast: Usagefast mode dual: Usagedual mode quad: Usagequad mode

Type

Project

Description

Configures SPI Flash address access mode, and the default is single.

Parameter

< single | fast | dual | quad >: SPI Flash address access mode.

Example

set_option -mspijump_mode single

-program_done_bypass

Forward new bitstream.

Syntax

-program_done_bypass <0|1>

Usage

Name	Description
<0|1>	0: Disables this function. 1: Enables this function.

Type

Project

Description

After configuring this option, when the internal signal of Done Final is activated, the external Done Pin will remain low, allowing the new bitstream to be forwarded after the current bitstream is loaded. The default value is 0.

Parameter

<0|1>: Enable control for forwarding new bitstream.

Example

set_option -program_done_bypass 1

-power_on_reset_monitor <0|1>

Power on reset.

Syntax

-power_on_reset_monitor <0|1>

Usage

Name	Description
<0|1>	0: Disables power on reset. 1: Enables power on reset.

Type

Project

Description

Enable control for power on reset. The default is 0.

Parameter

<0|1>: Enable control for power on reset.

Example

set_option -power_on_reset_monitor 1

-turn_off_bg

Bandgap function.

Syntax

-turn_off_bg <0|1>

Usage

Name	Description
<0|1>	0: Enables Bandgap. 1: Disables Bandgap.

Type

Project

Description

Enable control for Bandgap. The default is 0.

Parameter

<0|1>: Enable control for Bandgap.

Example

set_option -turn_off_bg 1

-wakeup_mode

Wake Up Mode enable control.

Syntax

-wakeup_mode <0|1>

Usage

Name	Description
<0|1>	0: Disables Wake Up Mode. 1: Enables Wake Up Mode.

Type

Project

Description

Enable control for Wake Up Mode. The default is 0.

Parameter

<0|1>: Enable control for Wake Up Mode.

Example

set_option -wakeup_mode 1

-user_code

Custom User Code.

Syntax

-user_code <default|value>

Usage

Name	Description
<default|value>	Custom User Cod value.

Type

Project

Description

Users can customize the User Code. The default value is default (00000000).

Parameter

<default|value>: Custom User Code.

Example

set_option -user_code 00000010

Note!

For a detailed explanation of the BitStream-related options, see BitStream in Section 4.3.3 of this document.

Unused Pin Property Configuration

-unused_pin

Sets different IO properties for unused GPIOs.

Syntax

-unused_pin <default|open_drain>

Usage

Name	Description
<default|open_drain>	Default: All unused GPIO pins will be configured as input tri-state with weak pull-up. open_drain：open_drain: All unused GPIO pins will be configured as output with setting OPEN DRAIN as ON.

Type

Project

Description

Sets different IO properties for unused GPIOs.

Parameter

<default|open_drain>: Sets different IO properties for unused GPIOs.

Example

set_option -unused_pin open_drain

Note!

For a detailed explanation of Unused Pin-related options, see Unused Pin in Section 4.3.3 of this document.

8.3.20 set_property

Sets the properties of an object.

Syntax

set_property [-dict <args>] <name> <value> <objects>

Usage

Name	Description
[-dict]	A property list of (name/value) pairs to be set.
<name>	The name of the property to be set. Not applicable when using -dict.
<value>	The value of the property to be set. Not applicable when using -dict.
<objects>	The object of the property to be set.

Type

IPFlow

Description

Configure options and their corresponding values for specified IP object.

Parameter

- [-dict]: Specifies a dictionary containing multiple pairs of options and their corresponding values. Each pair is defined as (<name> <value>), and multiple pairs are separated by spaces. The dictionary should be enclosed in curly braces {}.
- <name>: Specifies the name of the option to be configured. The description is in the form of CONFIG.property, and the property refers to the option name.
- <value>: Specifies the value for the corresponding option. The value needs to be valid according to the property type. If the option value is a string, it should follow the original form.

- <objects>: Specifies one or more IP objects for which the option is to be configured. A single object is specified using [get_ips module_name]. To specify multiple objects, use [get_ips module_name0 module_name1 …].

Example

The -dict option is used to specify multiple properties in the current design at once:

set_property -dict {CONFIG.Data_Width 16 CONFIG.Write_Depth 1024 CONFIG.Read_Depth 1024} [get_ips FIFO_Top]

Use name, value, objects to specify a single property in the current design:

set_property CONFIG.Data_Width {16} [get_ips FIFO_Top]

This example shows how to set a property value that contains a hyphen ("-") or spaces:

set_property {CONFIG.Almost_Full_Type} {Full-Single Threshold Constant Parameter} [get_ips FIFO_Top]

Note!

In some cases, the option value may contain special characters, such as a hyphen ("-") or spaces, which could cause the value to be incorrectly parsed. In such cases, it is necessary to enclose the option value in curly braces {}.

See also

- list_property
- report_property

8.3.21 source

In tcl command editor window of Gowin Software, or after launching command-line mode, this command can be used to execute a tcl script. For file path formats, see 8.3.1 add_file.

Syntax

source <file>

Usage

Name	Description
<file>	The tcl script to be executed.

Type

IPFlow, Project

Description

Sets different IO properties for unused GPIOs.

Parameter

<file>: The tcl script to be executed.

Example

source project.tcl

source D:/gowin_project/project.tcl

source D:\\gowin_project\\project.tcl

8.3.22 write_ip_tcl

Exports a tcl script that can regenerate the given IP.

Syntax

write_ip_tcl [-ip_name <newModuleName>] [-multiple_files] [-force] [<tcl_filename>] <objects>

Usage

Name	Description
[-ip_name]	Sets the IP module name
[-multiple_files]	Creates a separate .tcl file for each IP object
[-force]	Overwrites existing files
[<tcl_filename>]	Exported tcl file
<objects>	IP objects that export tcl files

Type

IPFlow

Description

This command is used to export tcl script files for specified IP objects, enabling regeneration of the IPs via tcl scripts. For file path formats, see 8.3.1 add_file.

Parameter

- [<tcl_filename>]: Name of the generated tcl script file. If not specified, the current project name is used.
- <objects>: Specifies one or more IP objects to generate the tcl script. A single object is specified using [get_ips module_name], and multiple objects are specified using [get_ips module_name0 module_name1 …].
- [-ip_name <newModuleName>]: Renames the IP module name in the generated tcl script. Only one object can be specified.
- [-multiple_files]: Generates tcl script files for all specified IPs, with file names based on the respective module_name. This option is mutually exclusive with -ip_name and <tcl_filename>.
- [-force] Overwrites existing tcl files with the same name.

Example

In this example, a tcl file is created, and this file is used to specify the IP object FIFO_Top, but when using the source command, it is created with the new name FIFO_Top_new:

write_ip_tcl -ip_name FIFO_Top_new [get_ips FIFO_Top]

In this example, a separate tcl file is generated for each specified IP module in the project.

write_ip_tcl -multiple_files [get_ips FIFO_Top FIFO_Top_1]

In this example, multiple specified IP modules from the project are written into a single Tcl file:

write_ip_tcl [get_ips FIFO_Top FIFO_Top_1] my_fifo.tcl

9 Appendix

9.1 File Description

Gowin Software supports adding physical constraints, timing constraints, and other files during the project design, and a variety of execution files are generated in the overall design; this section will introduce these files supported by Gowin Software as shown below.

Table 9-1 Source Files

File Type	Definition	Function
.gsc	Synthesis constraints file	Constraint files for the GowinSynthesis
.ipc	IP configuration file	IP Core Generator can load an .ipc file to recreate the IP after modifying the configuration
.cst	Physical constraints file	Used to add physical constraints to the design
.sdc	Timing constraints file	Used to add timing constraints to the design
.fi	User Flash initialization file	Initialization assignment to User Flash; you can select to load it when downloading the bitstream through the Programmer.
.rao	RTL-level GAO configuration file	Capture the RTL signal before synthesis
.gao	Post-synthesis GAO configuration file	Capture the Netlist signal after synthesis
.gvio	Virtual Input/Output Configuration File	Used to real-time monitor and drive internal FPGA signals
.gpa	Power analysis configuration file	Used to analyze the power analysis of the design
.mi	Memory initialization file	Initialization assignment of memory; you can use this initialization file when generating the memory through IP Core Generator.
.v	Verilog source file	Verilog description file containing circuit structure and function
.sv	System Verilog source file	System Verilog description file containing circuit structure and function
.vhd	VHDL source file	VHDL description file containing circuit structure and function

Table 9-2 Execution Files

File Type	Definition	Function
.vg	Post-synthesis netlist file	The post-synthesis netlist file using GowinSynthesis
_syn.rpt.html	Synthesis report file	Used to view information such as resource utilization and timing analysis after synthesis
.fs	Bitstream file	Used by Gowin Programmer to download
.bin	Bitstream files in bin format	Used by Gowin Programmer to download
.ekey	key file	Used to decrypt the encrypted BitStream files during download using Gowin Programmer
.vo	Timing simulation model file in verilog post PnR	Used for Verilog model file with the flatten structure for timing simulation
.vho	Timing simulation model file in vhdl post PnR	Used for VHDL model file with the flatten structure for timing simulation
.sdf	Standard delay format file	Used for netlist timing simulation after PnR
.ibs	Input/output buffer information specified files	-
.tr	Timing report in text format	-
.rpt.txt	PnR report in text format	-
.rpt.html	PnR report in html format	-
.tr.html	Timing analysis report in html format	-
.pin.html	Port attributes report in html format	-
.power.html	Power analysis report in html format	-
.p	Incremental placement file	Used for incremental placement
.pr	Incremental placement and routing file	Used for incremental placement and routing

9.2 File and Folder Naming Rules

Gowin folders and files naming rules: The names cannot contain ? " / \ < > * | : characters; folder names can not contain spaces, file names can contain spaces but spaces cannot appear at the beginning and end of the name.

For the file path filling in the dialog box of each component of Gowin Software, it will be judged according to the above rules, and a pop-up window will be displayed if it does not comply with the rules.

9.3 Security Declaration

During installation and use, Gowin Software does not collect any information from users or access network data ports in the background; all data and information are kept locally, and no automatic updates are made to the software.


GOWIN
PROGRAMMING FOR THE FUTURE