# Gowin Software Quick Start Guide

**SUG918-1.9E,2024-08-09**

## 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 |
|---|---|---|
| 05/07/2020 | 1.0E | Initial version published. |
| 09/07/2020 | 1.1E | ● RTL schematic added.<br>● File encryption added.<br>● Tcl command added. |
| 10/21/2020 | 1.1.1E | Use GowinSynthesis as an example to describe synthesis. |
| 06/10/2021 | 1.2E | ● Synplify Pro removed.<br>● MIPI IP in the design modified. |
| 11/02/2021 | 1.3E | Some descriptions updated. |
| 07/28/2022 | 1.4E | Modified the design to FIFO HS and updated the relevant descriptions. |
| 12/20/2022 | 1.5E | ● The function of viewing the schematic diagram of the netlist after synthesis added.<br>● Some figures updated. |
| 05/25/2023 | 1.5.1E | ● Figure 3-15 Synthesis Configuration and Figure 3-34 Jointly Debugging with GAO updated.<br>● The description of 3.10.1 Configuration updated. |
| 08/18/2023 | 1.6E | The descriptions of timing optimization removed. |
| 11/30/2023 | 1.7E | ● create_project, import_files, and run close commands added.<br>● Some screenshots in Chapter 3 Quick Start updated. |
| 12/29/2023 | 1.7.1E | ● The title of Figure 3-3 updated to FIFO HS Configuration.<br>● Minor description changes. |
| 02/02/2024 | 1.7.2E | Some screenshots in Chapter 3 Quick Start updated. |
| 06/28/2024 | 1.8E | ● Some screenshots in Chapter 3 Quick Start updated.<br>● Descriptions of Section 4.2 Tcl Quick Start updated. |
| 08/09/2024 | 1.9E | ● GVIO configuration added.<br>● The device information of FIFO_HS project updated.<br>● Some screenshots updated. |

# Contents

# List of Figures

# List of Tables

# 1 About This Guide

## 1.1 Purpose

This manual uses FIFO HS as an example to introduce Gowin Software and aims to help you get familiar with the usage and improve the design efficiency.

## 1.2 Related Documents

You can find the related documents at www.gowinsemi.com:

- SUG100, Gowin Software User Guide
- SUG935, Gowin Design Physical Constraints User Guide
- SUG940, Gowin Design Timing Constraints User Guide
- SUG114, Gowin Analyzer Oscilloscope User Guide
- SUG282, Gowin Power Analyzer User Guide
- SUG502, Gowin Programmer User Guide
- SUG550, GowinSynthesis User Guide
- SUG755, Gowin HDL Schematic Viewer User Guide
- SUG1018, Arora V Design Physical Constraints User Guide
- SUG1189, Gowin Virtual Input Output 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 |
|---|---|
| AO Core | Analysis Oscilloscope Core |
| BSRAM | Block Static Random Access Memory |
| DFF | D Flip-Flop |
| FloorPlanner | FloorPlanner |
| FPGA | Field Programmable Gate Array |

| Terminology and Abbreviations | Meaning |
|---|---|
| GAO | Gowin Analyzer Oscilloscope |
| GPA | Gowin Power Analyzer |
| GVIO | Gowin Virtual Input Output |
| I/O | Input/Output |
| IP Core | Intellectual Property Core |
| PnR | Place & Route |
| RTL | Register Transfer Level |

## 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 Introduction

## 2.1 Design Flow Introduction

Gowin Software is available in Windows and Linux. It supports GUI running mode and commands running mode. Take the GUI running mode in Windows and FIFO HS design as an instance to introduce quick start of Gowin Software.

The design uses FloorPlanner to add physical constraints, uses Timing Constraints Editor to add timing constraints, uses GAO to add GAO config file and to capture data, GPA to add GPA config file, and Programmer to download bitstream.

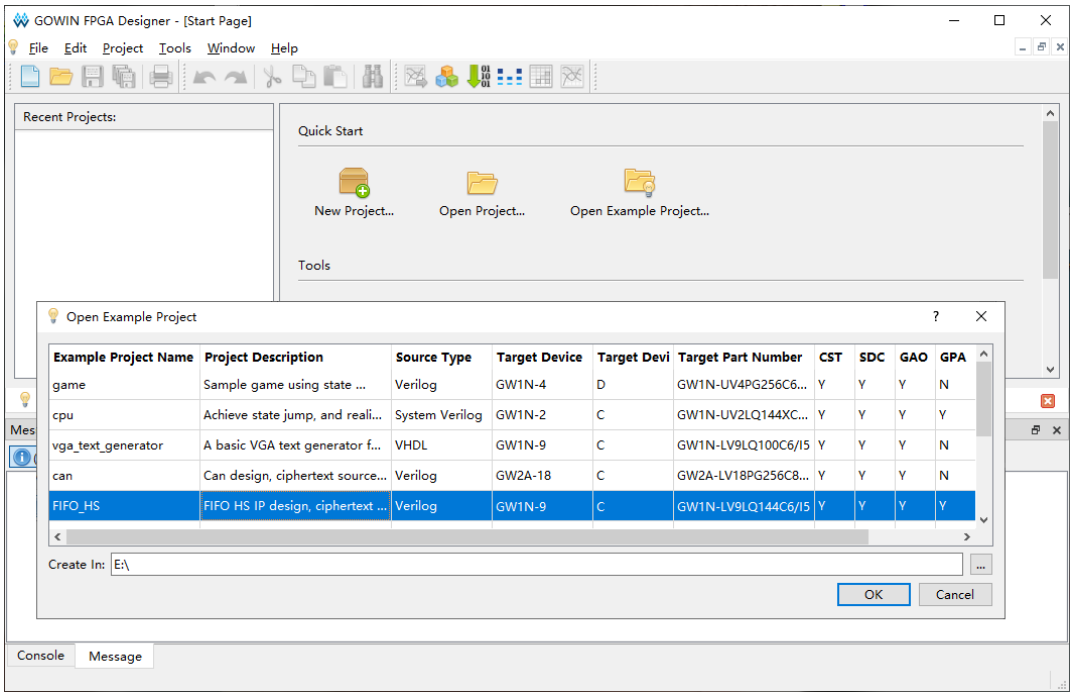
## 2.2 Quick Started Design Introduction

FIFO HS IP can complete the data transmission and buffering with different bit widths in the asynchronous clock domain, and configure different output control signals and data structures according to your requirements.

The whole design provides clock for FIFO HS through port, provides reset signal, enable signal and input data through logic, and finally uses GAO to collect data to verify the correctness of FIFO HS.

The design has been added to the sample project FIFO_HS, which can be quickly created by clicking "Start Page > Open Example Project…", as shown in Figure 2-1. Creating a project through the example will skip the previous steps and go directly to placement and routing, and the subsequent process. If you want to be familiar with the use of Gowin Software step by step, you can operate according to the guidelines in the document. The source files, constraint files and configuration files involved in the design are consistent with those in the sample project. You can save the files in the sample project for later use.

**Figure 2-1 Open Example Project**



# 3 Quick Start

## 3.1 Create a New Project

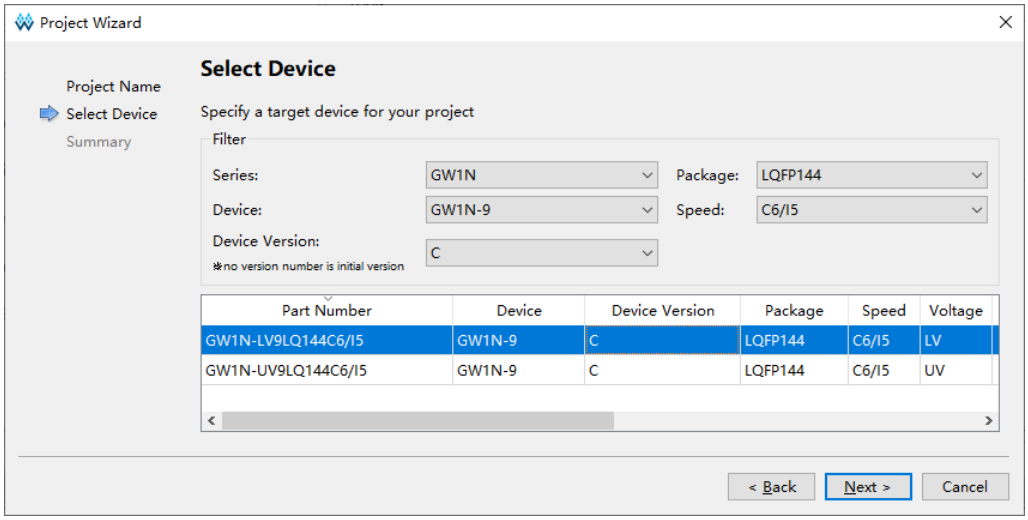
### 3.1.1 Create a New Project

Open Gowin Software and click "Start Page > Quick Start > New Project" to create a new project named as FIFO_HS. The device selected is as shown in Figure 3-1.

- Series: GW1N
- Device: GW1N-9
- Device Version: C
- Package: LQFP144
- Speed: C6/I5
- Part Number: GW1N-LV9LQ144C6/I5

Click "Next" until the project creation completed. For the details, please refer to *SUG100, Gowin Software User Guide*.

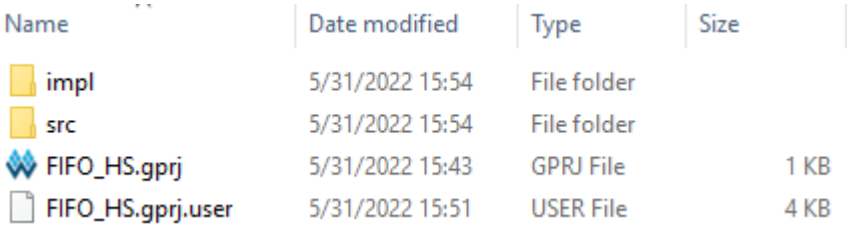
**Figure 3-1 Create a New Project**




After the project is created, the impl and src folders are generated

under the project creation path, as shown in Figure 3-2. impl contains synthesis and PnR files and src contains the source files.

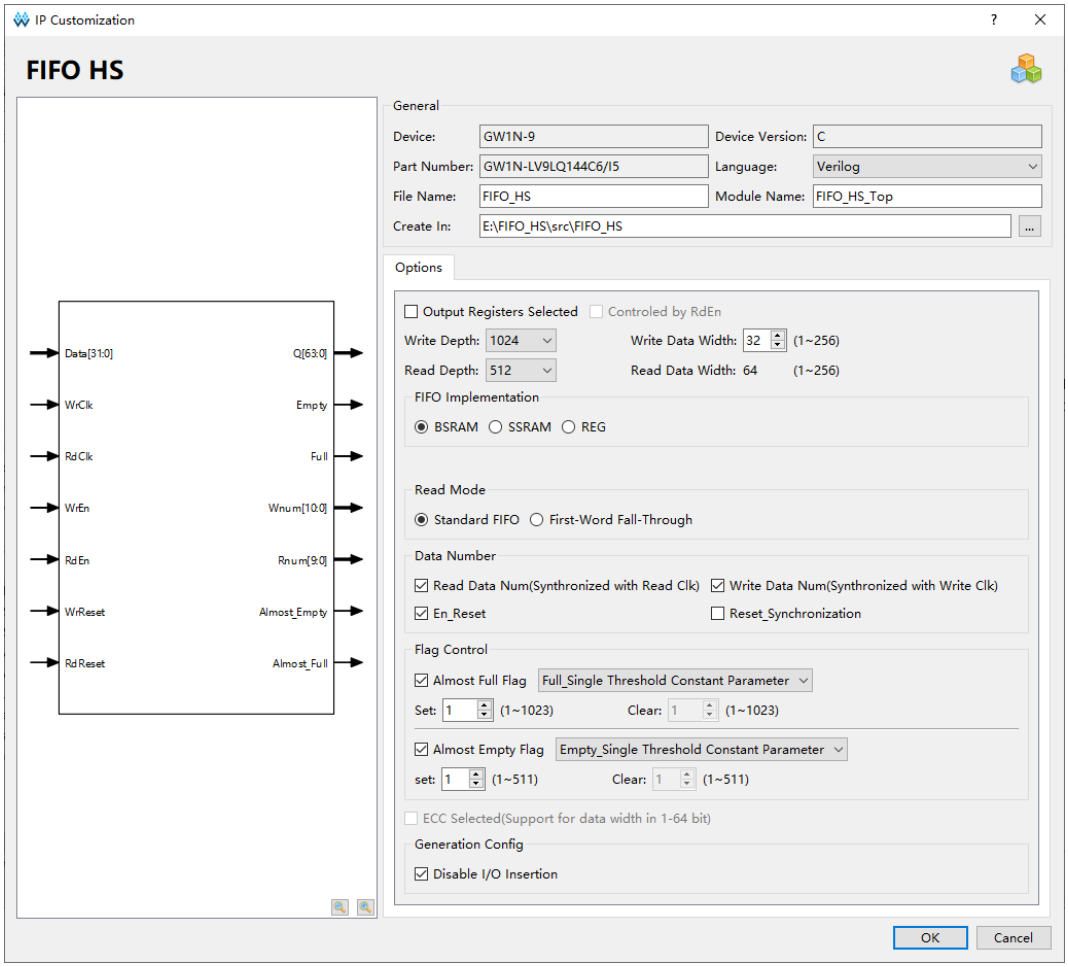
**Figure 3-2 Project Directory**




## 3.1.2 Generate FIFO HS IP

Click "Tools > IP Core Generator" to open the IP Core Generator window. Double-click "Memory Control > FIFO > FIFO HS" to open the IP Customization dialog box to configure as required. The FIFO HS configuration in this design is shown in Figure 3-3. Then click "OK" to generate FIFO HS IP.

**Figure 3-3 FIFO HS Configuration**




After generation, IP design files and simulation files are generated under the IP creation path, as shown in Figure 3-4.

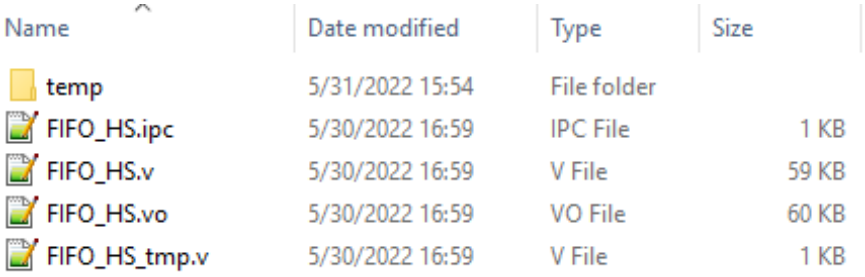
- .v file is an IP design file, encrypted.

- _tmp.v is an IP design template file.
- .vo file is an IP simulation model file, unencrypted.
- .ipc file is an IP configuration file. The user can load the file to modify the configuration.
- temp contains the files required to generate the IP.
- The doc, model, sim, and tb contain the simulation files: readme text, simulation model, simulation script, and testbench.

**Note!**

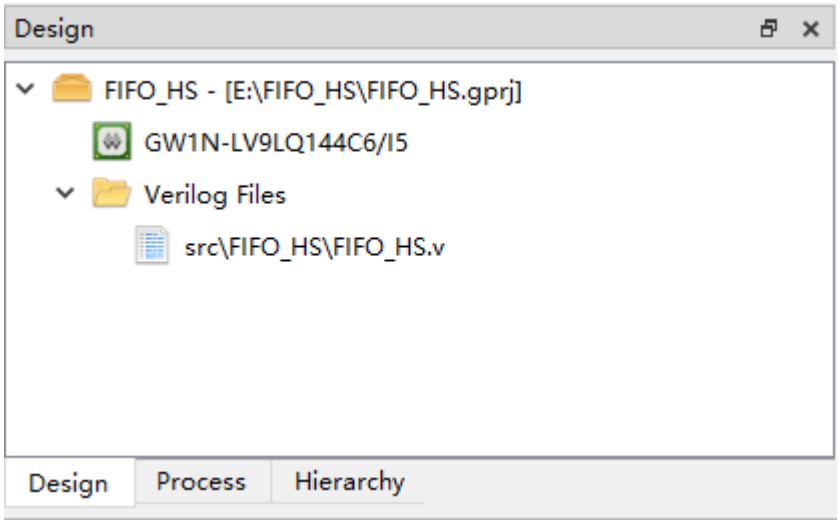
- For Gowin Software 1.9.8.06 and later versions, if VHDL is selected as the Language during IP generation, .vho file will be generated under the IP creation path, which is an IP simulation model file in plaintext.
- At present, for some IPs, the created path still generates doc, model, sim, and tb folders, indicating readme text, simulation model, simulation script, and testbench simulation file. The IP directory is subject to IP Core Generator in use.

**Figure 3-4 FIFO HS IP Directory**




After FIFO HS IP is generated, the Design window is as shown in Figure 3-5.

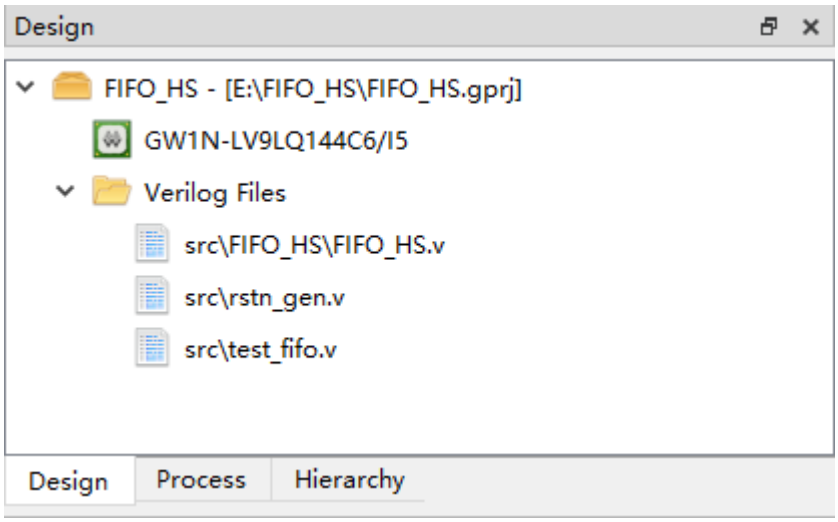
**Figure 3-5 Design Window**




## 3.1.3 Load File

In order to test FIFO HS, some design files need to be loaded or created, as shown in Figure 3-6. For the steps to load files, you can see Section 2.2 Quick Started Design Introduction.

**Figure 3-6 Load Files**




# 3.2 RTL Schematic

After the source file is loaded, you can view the RTL design schematic by clicking "Tools > Schematic Viewer > RTL Design Viewer" to help you better understand the RTL logic. For details, see *SUG755, Gowin HDL Schematic Viewer User Guide*.

# 3.3 GAO Configuration

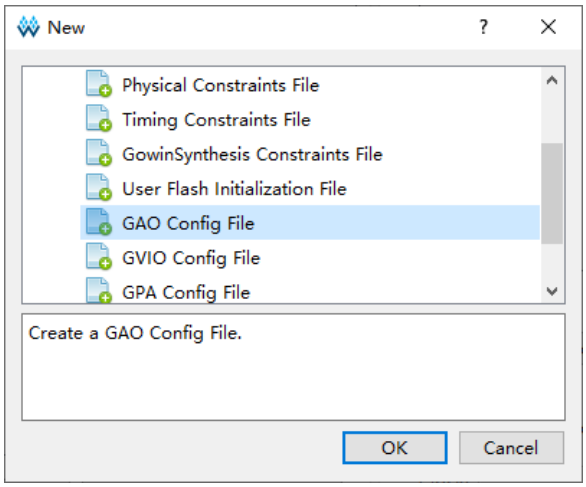
Gowin Software supports two signal capture sources: RTL-level signal capture and post-synthesis netlist-level signal capture; GAO config. file can be created after the source files are created or loaded at the RTL level, and GAO config. file can be created after the synthesis is completed at the post-synthesis netlist level. GAO config. file can be used to capture data and verify the the design. In addition, Gowin Software provides Standard Mode GAO and Lite Mode GAO. For the usage, see *SUG114, Gowin Analyzer Oscilloscope User Guide*.

This design uses RTL-level signal capture and Standard Mode GAO as an instance.

## 3.3.1 Create Standard Mode GAO Config File

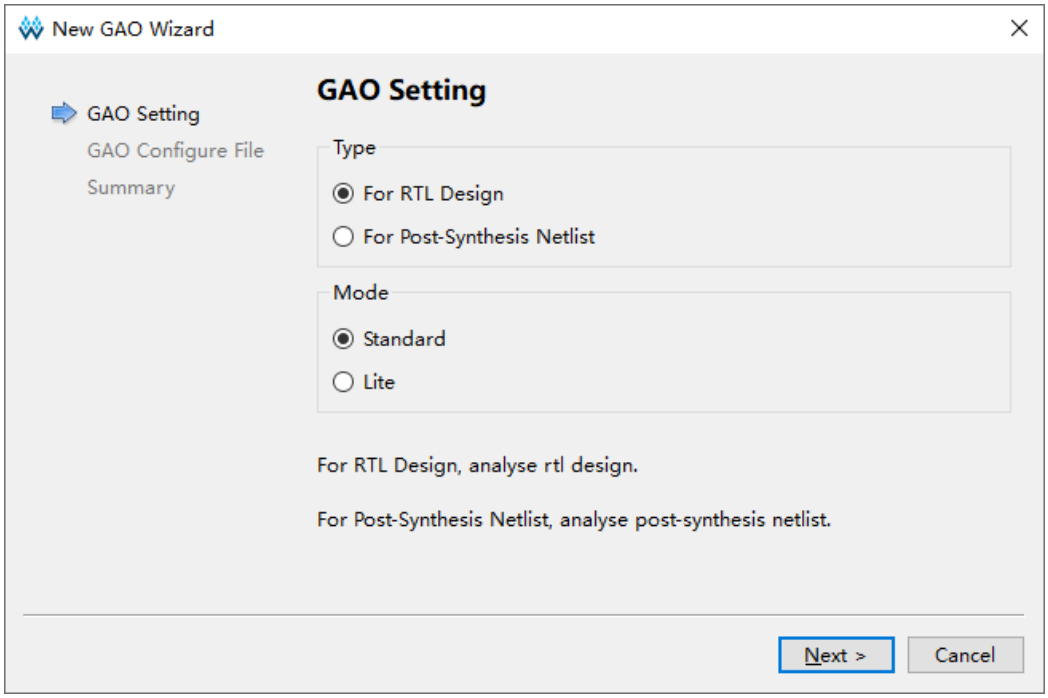
Select "Design > New File…" to open "New" dialog box, and select "GAO Config File" in "New", as shown in Figure 3-7. Click "OK".

**Figure 3-7 Create GAO Config File**




Select "For RTL Design" in Type, and "Standard" in Mode, as shown in Figure 3-8. Click "Next". The file name is FIFO HS. Then click "Next" until finished.

**Figure 3-8 GAO Setting**




## 3.3.2 Configure Standard Mode GAO

After file created, you can configure the number of AO cores, trigger options and capture options. The trigger options include match unit, trigger port, match type and expressions; The capture options include sample clock, capture, capture utilization and capture signals. In this design the number of AO cores is 1 and the trigger options and capture options configuration are shown in Figure 3-9 and Figure 3-10.

**Figure 3-9 Trigger Options Configuration**

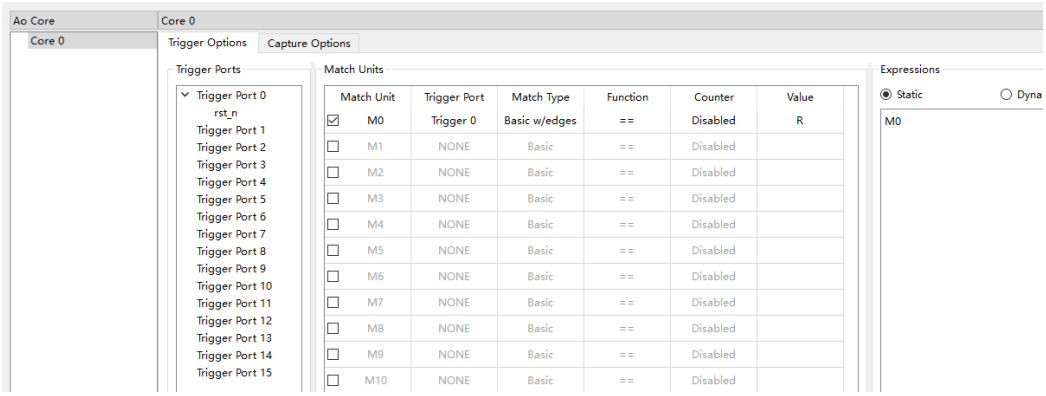
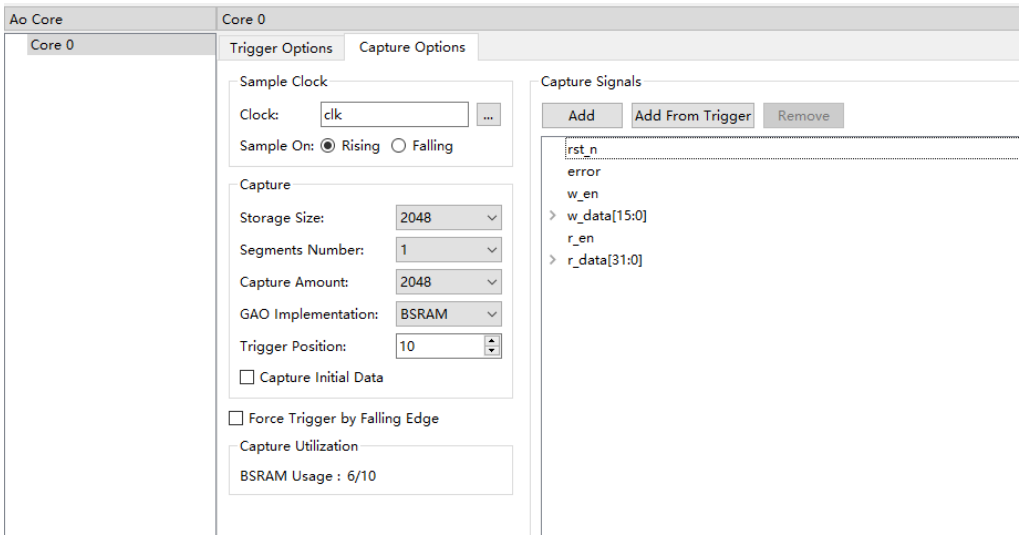
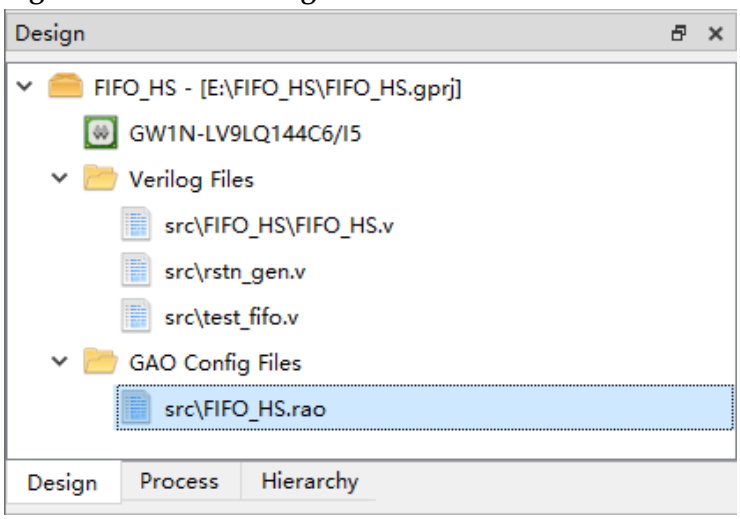



**Figure 3-10 Capture Options Configuration**




After configuration, click "Save" to finish and the design window is as shown in Figure 3-11.

**Figure 3-11 GAO Config Files**



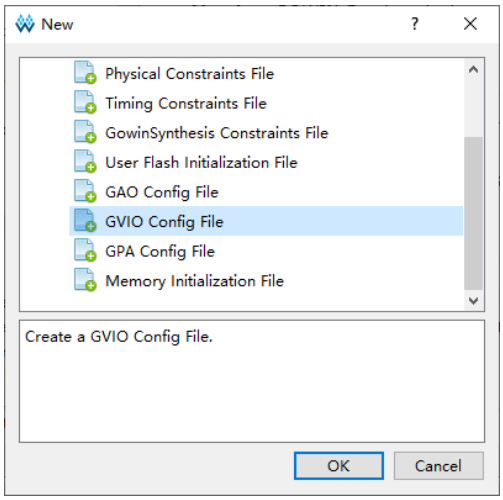
# 3.4 GVIO Configuration

Gowin Virtual Input Output (GVIO) can monitor and drive internal FPGA signals in real-time. When jointly debugging with the online logic analyzer Gowin Analyzer Oscilloscope (GAO), GVIO provides a more powerful debugging environment. This debugging environment can generate internal signal stimuli and obtain logic responses through the GAO tool, aiming to help users quickly perform system analysis and fault localization, thereby improving design efficiency. For detailed usage of GVIO, *SUG1189, Gowin Virtual Input Output User Guide*.

## 3.4.1 Create GVIO Configuration File

Select "Design > New File..." in Gowin Softare. In the "New" dialog box that appears, choose to create a new "GVIO Config File," as shown in Figure 3-12. Click "OK," define the file name as FIFO_HS, and the file path defaults to the src folder under the project. Click "OK" again to complete the creation of the GVIO configuration file.

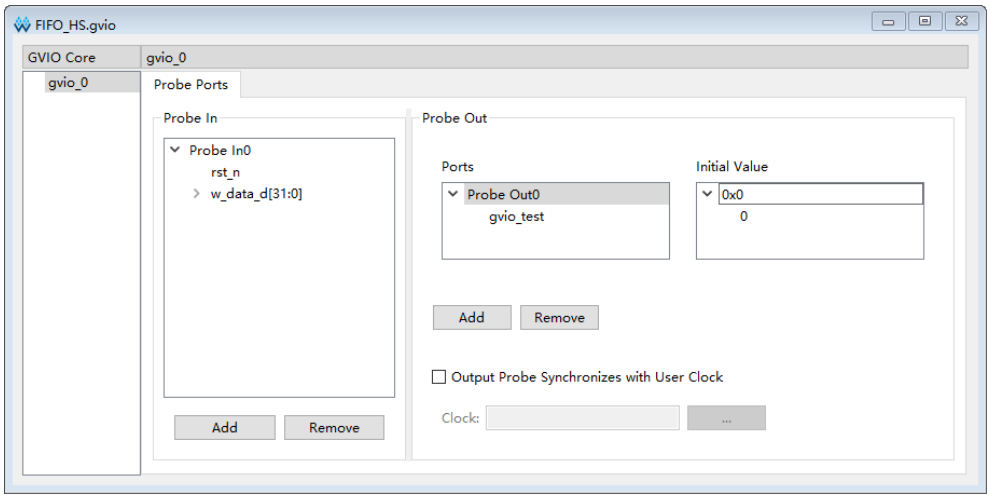
**Figure 3-12 Create GVIO Configuration File**




## 3.4.2 Configuration Options

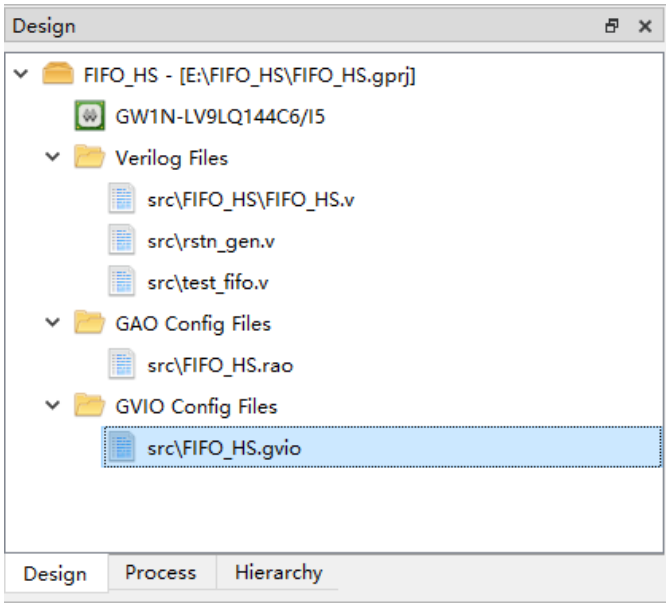
Double-click the configuration file (.gvio) in the "Design" view. The GVIO Config window will pop up. The GVIO configuration window includes the GVIO Core view for configuring the number of AO cores and their corresponding signal configuration view. The core signal configuration view consists of "Probe In" view for configuring sampling signals and "Probe Out" view for configuring stimulus signals. In this design, the number of AO core is set to 1. The configuration of the sampling and stimulus signals is shown in Figure 3-13.

**Figure 3-13 GVIO Configuration Window**




After the configuration is completed, click "Save" on the toolbar to complete the GVIO configuration file. The GVIO configuration file will be displayed in the "Design" window, as shown in Figure 3-14.

**Figure 3-14 GVIO Configuration File**



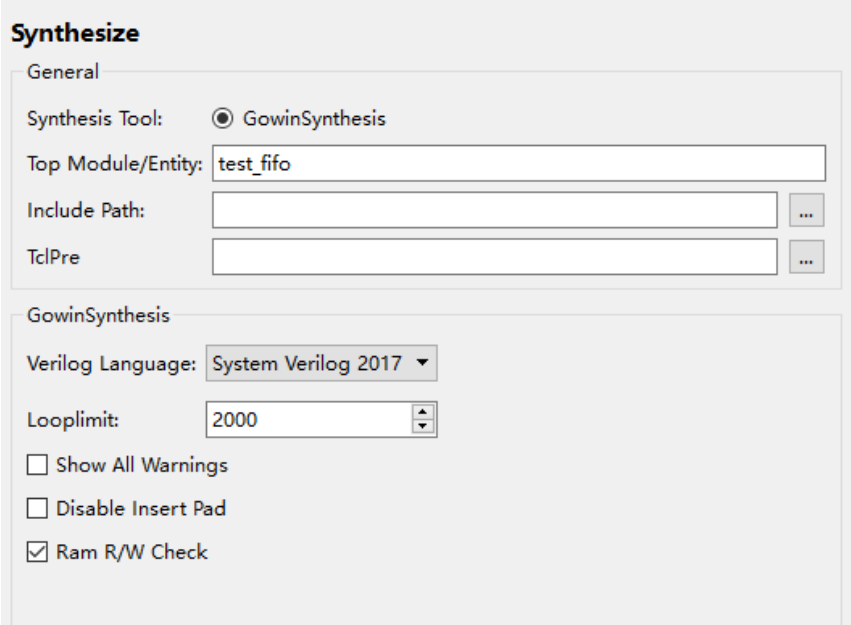

# 3.5 Use GowinSynthesis to Synthesize

## 3.5.1 Configuration

Select "Process > Synthesize (right-click) > Configuration" to open "Configuration" dialog box. For details, see *SUG550, GowinSynthesis User Guide*.

The top module/entity is test_fifo, as shown in Figure 3-15.

**Figure 3-15 Synthesis Configuration**




In addition, you can add some attributes and instructions to the source file to control synthesis. For the details, see *SUG550, GowinSynthesis User Guide*. As shown in Figure 3-16, in this design, a specific net is retained without optimization during the synthesis by using the/* synthesis syn_keep=1 */ attribute.

**Figure 3-16 Attributes and Instructions of GowinSynthesis**

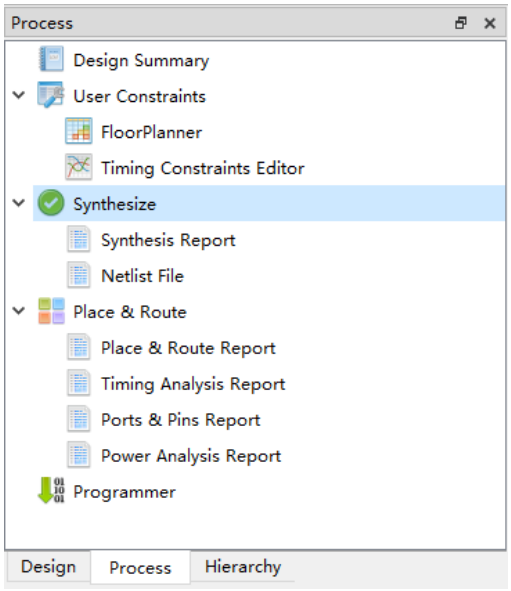
```
67  reg     [1:0]          ALT_CNT_d;
68  reg     [7:0]          rand_num;
69  reg     [9:0]          rand_cnt;
70  reg     [11:0]         start_rdmck;
71  reg                    fifo_empty_d;
72  wire    [WRSIZE-1:0] w_data_d/* synthesis syn_keep=1 */;
73  wire                   load;
74  wire    [RDSIZE-1:0] r_data;
75  wire    [WNSIZE:0]   w_num;
76  wire    [RNSIZE:0]   r_num;
77  wire                   fifo_full;
78  wire                   fifo_empty;
79  wire                   fifo_alempty;
80  //test state machine
```

## 3.5.2 Synthesize

After synthesis configuration, you can start to synthesize.

Double-click "Synthesize" in Process window to synthesize, as shown in Figure 3-17. When the icon changes to " ", you can double-click Synthesis Report to view the report and double-click Netlist File to view the netlist file.

**Figure 3-17 Synthesis Completed**




After synthesis, the gwsynthesis folder is generated under the \impl path. The folder contains all the files and folders generated in synthesis, as shown in Figure 3-18.

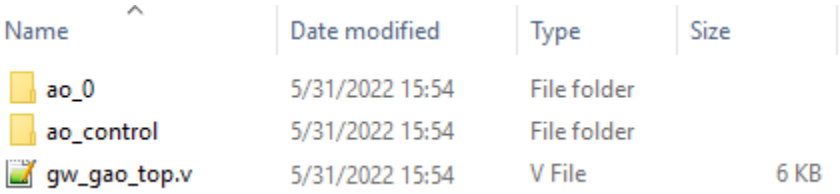
**Figure 3-18 gwsynthesis Directory**

| Name | Date modified | Type | Size |
|---|---|---|---|
| RTL_GAO | 5/31/2022 15:54 | File folder | |
| FIFO_HS.log | 5/31/2022 15:51 | LOG File | 6 KB |
| FIFO_HS.prj | 5/31/2022 15:50 | PRJ File | 2 KB |
| FIFO_HS.vg | 5/31/2022 15:51 | VG File | 454 KB |
| FIFO_HS_syn.rpt.html | 5/31/2022 15:51 | 360 se HTML Doc... | 29 KB |
| FIFO_HS_syn_resource.html | 5/31/2022 15:51 | 360 se HTML Doc... | 3 KB |
| FIFO_HS_syn_rsc.xml | 5/31/2022 15:51 | XML Document | 1 KB |

If the project contains the GAO config file, after PnR, RTL_GAO folder is generated under the project creation path \impl\gwsynthesis, as shown in Figure 3-18, and this folder contains all the files generated by the RTL GAO synthesis as shown in Figure 3-19.

- ao_0 contains the parameter files of the AO core.
- ao_control contains the parameter files of the control AO core.
- gao.v is the netlist file GAO post-synthesis, encrypted.
- gw_gao_top.v is the top file of GAO, connecting ao, ao_control and jtag modules.
- The other files are generated during GAO synthesis.

**Figure 3-19 GAO Directory**




# 3.6 View Schematic Diagram of the Netlist after Synthesis

After completing the synthesis, you can view the schematic diagram of the entire design through the menu bar "Tools > Schematic Viewer > Post-Synthesis Netlist Viewer" to help you better understand the logic of the design after synthesis. For more details, see *SUG755-1.2.1E_Gowin HDL Schematic Viewer User Guide*.

# 3.7 Physical Constraints

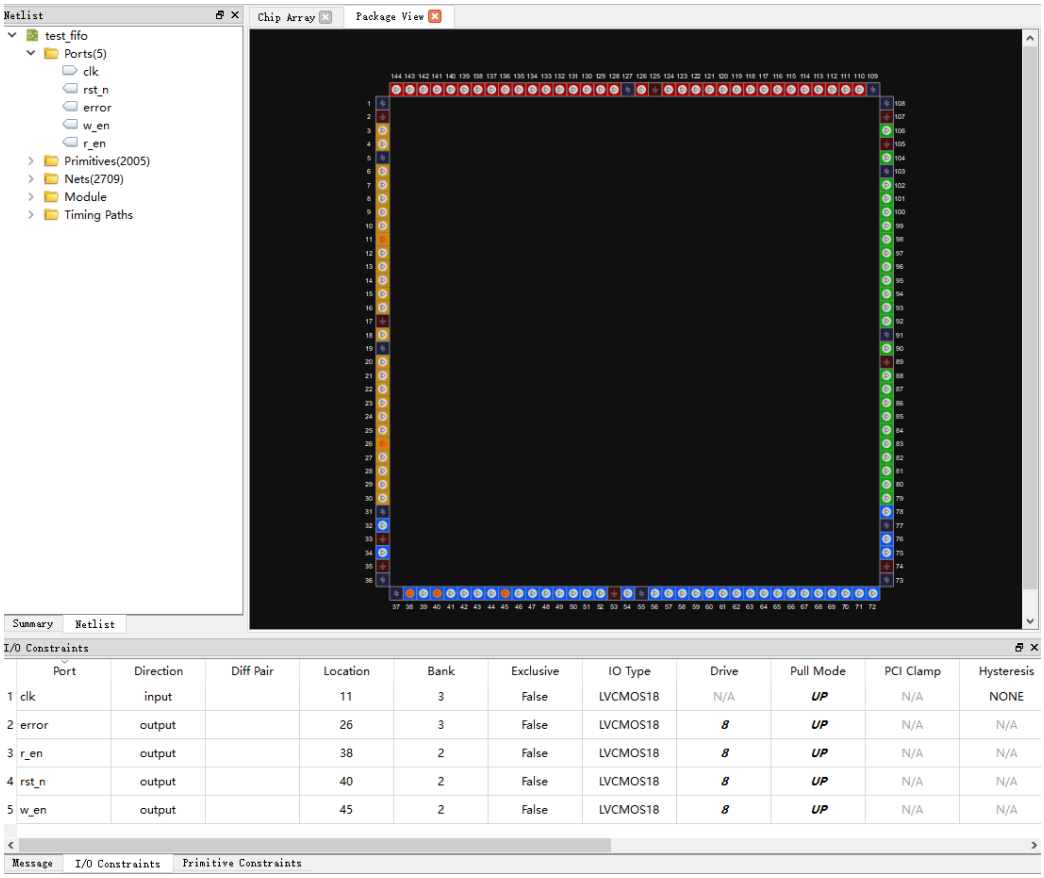
After synthesis, you can use FloorPlanner or write manually to add physical constraints. In this design, FloorPlanner is selected. For more details, please refer to the *SUG935, Gowin Design Physical Constraints User Guide* and *SUG1018, Arora V Design Physical Constraints User Guide*.

## 3.7.1 Create New Physical Constraints

Click "Process > User Constraints > FloorPlanner" to open FloorPlanner, which supports I/O, Primitive, and Group physical constraints. This design only adds I/O constraints and uses it as an instance.

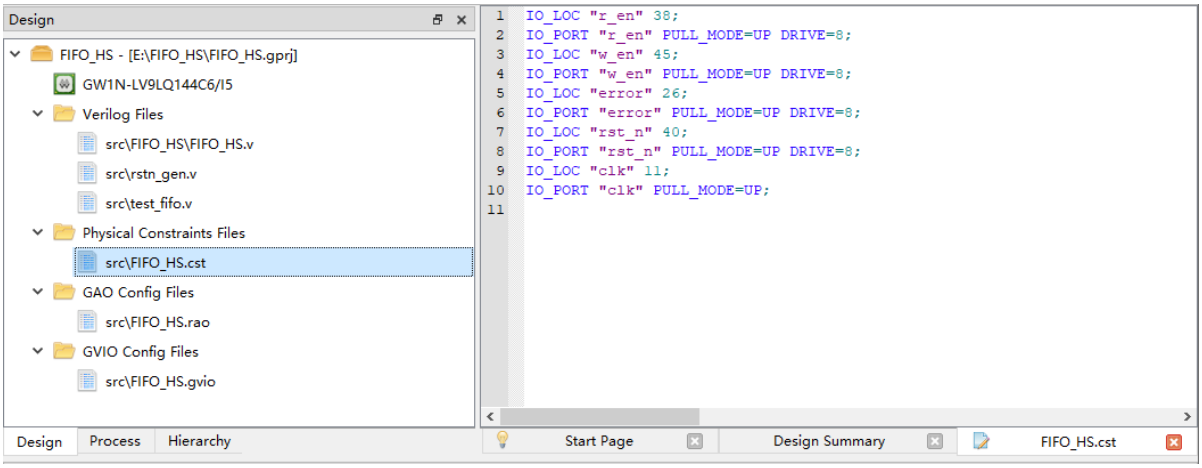
You can create I/O constraints in I/O Constraints window. Drag the port row to be constrained in the Netlist or I/O Constraints window to a specific location in the Package View or Chip Array view. After finished, the port location displays in the IOB, as shown in Figure 3-20.

**Figure 3-20 I/O Constraints**




After constraints finished, click "Save" to generate physical constraints files as shown in Figure 3-21.

**Figure 3-21 Physical Constraints Display**




In PnR, if there is no physical constraints file, the PnR will be automatically performed. If there is a physical constraint file, the PnR will be performed according to the physical constraints file.

### 3.7.2 Modify Physical Constraints

After physical constraints files generated, you can modify the constraints by FloorPlanner. Click "Save" to finish.

## 3.8 Timing Constraint

After synthesis, you can use Timing Constraints Editor or write manually to add timing constraints. In this design, Timing Constraints Editor is selected. For more details, please refer to *SUG940, Gowin Design Timing Constraints User Guide*.

### 3.8.1 Create New Timing Constraints

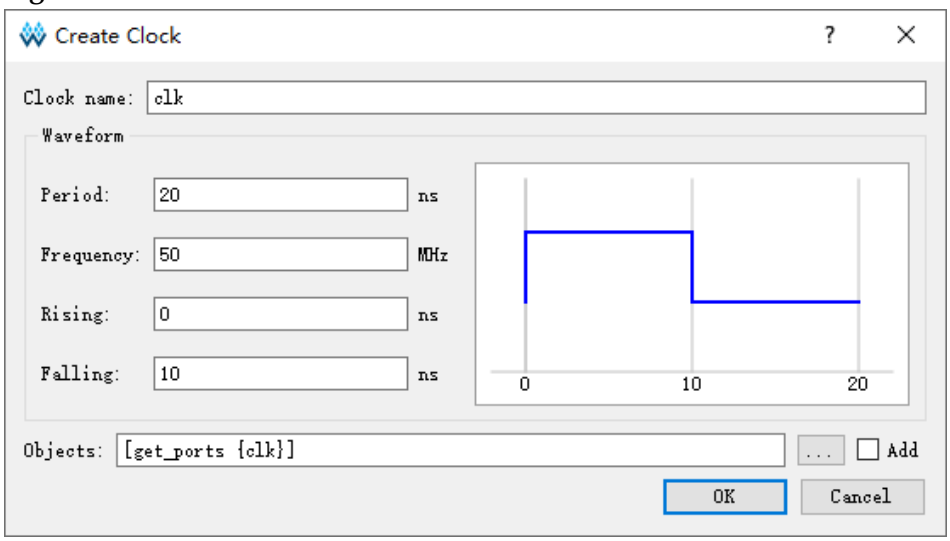
Click "Process > User Constraints > Timing Constrains Editor" to open Timing Constrains Editor, which supports clock, I/O and timing report constraints. This design adds clock and timing report constraints and uses them as instances.

**Clock Constraints**

Select "Clocks" under "Timing Constraints", right-click in the blank space on the right and select "Create Clock". This will open the "Create Clock" dialog, as shown in Figure 3-22. And create the following constraints:

- Clock name: clk
- Period: 20
- Frequency: 50
- Rising: 0
- Falling: 10
- Source Object: get_ports {clk}

**Figure 3-22 Clock Constraints**



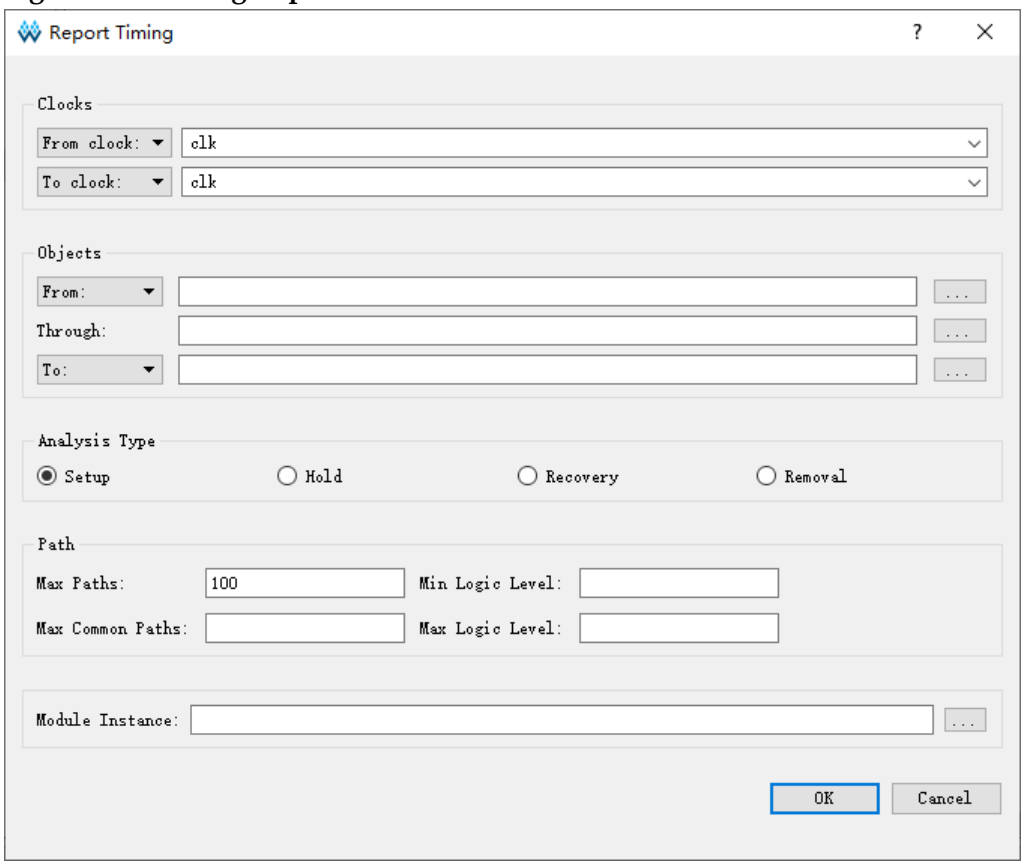

The design uses GAO, so the clock tck_pad_i is created in the same way as clk. The relationship between clk and tck_pad_i is an asynchronous

clock. If you do not want to use Gowin Software to analyze this relationship, you can create a clock group constraint through the timing constraint editor.

### Timing Report Constraint

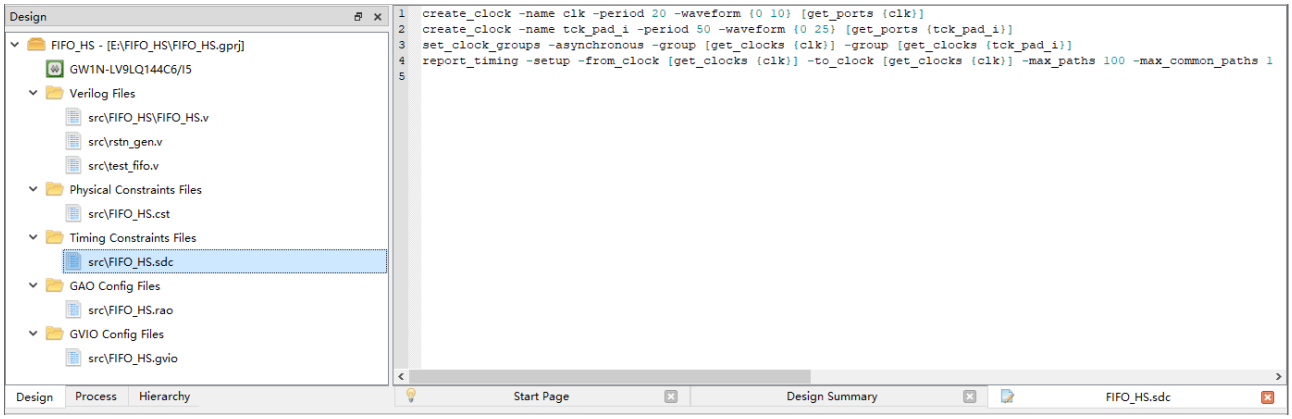
Select "Timing Constraints > Report > Report Timing", right-click in the blank space on the right and select "Create Report". In the popped-up "Report Timing" dialog, configure the parameters; the setup paths for clk to clk are reported, limiting the number of paths to 100, as shown in Figure 3-23.

**Figure 3-23 Timing Report Constraint**




After constraints is finished, click "Save" to generate timing constraints, as shown in Figure 3-24.

**Figure 3-24 Timing Constraints Display**



In PnR, if there is no timing constraints file, the PnR will be automatically performed. If there is a timing constraint file, the PnR will be performed according to the timing constraints file.

### 3.8.2 Modify Timing Constraints

After timing constraints files are generated, you can modify the constraints by Timing Constrains Editor. Click "Save" to finish.

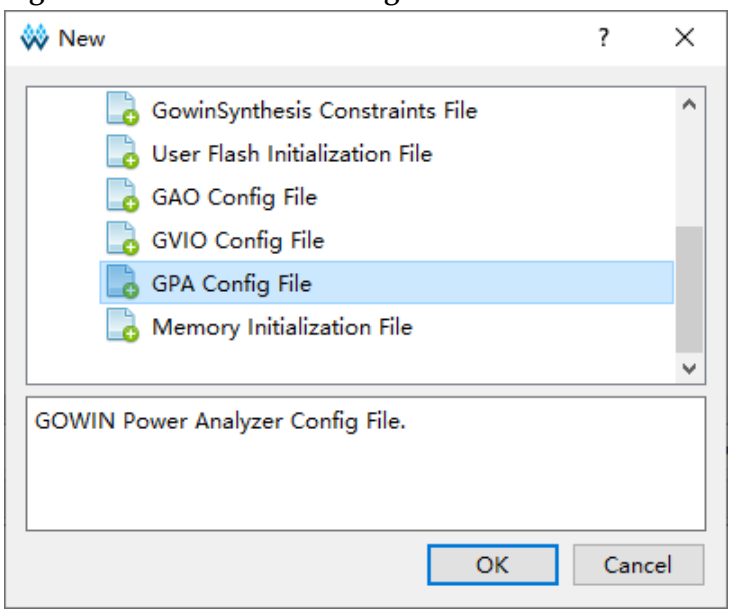
## 3.9 GPA Configuration

After synthesis, you can create a GPA config file to analyze power. For the usage, please refer to *SUG282, Gowin Power Analyzer User Guide*.

### 3.9.1 Create GPA Config File

Select "Design > New File…" to open "New" dialog box, and select "GPA Config File" in "New", as shown in Figure 3-25. Click "OK". The file name is FIFO_HS and the file is under src by default. Then click "OK" to finish.

**Figure 3-25 Create GPA Config File**




### 3.9.2 Configure GPA

After GPA config file is created, configure General Setting, Rate Setting and Clock Setting.

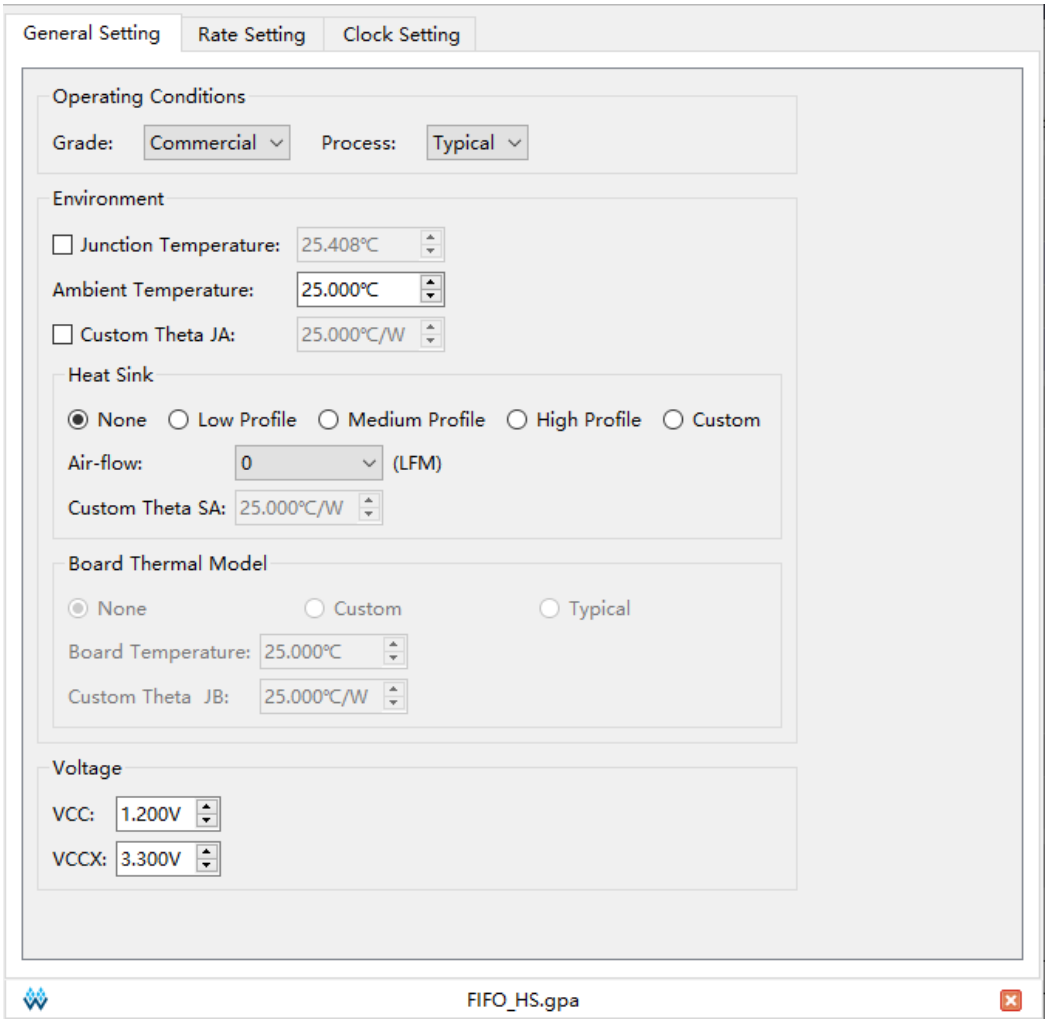
- General Setting includes the parameters of device, package, speed grade, temperature grade, thermal impedance, and voltage.
- Rate Setting is used to configure signal transition rate. You can set transition rate of IO or Net, or use the default value.
- Clock Setting is used to configure clock and enable features of BSRAM, I/O and DFF.

#### General Setting

In this design, the general setting is configured as follows: commercial

temperature, 25℃ ambient temperature, no heat sink, VCCX 3.3V and VCC 1.2V, as shown in Figure 3-26.

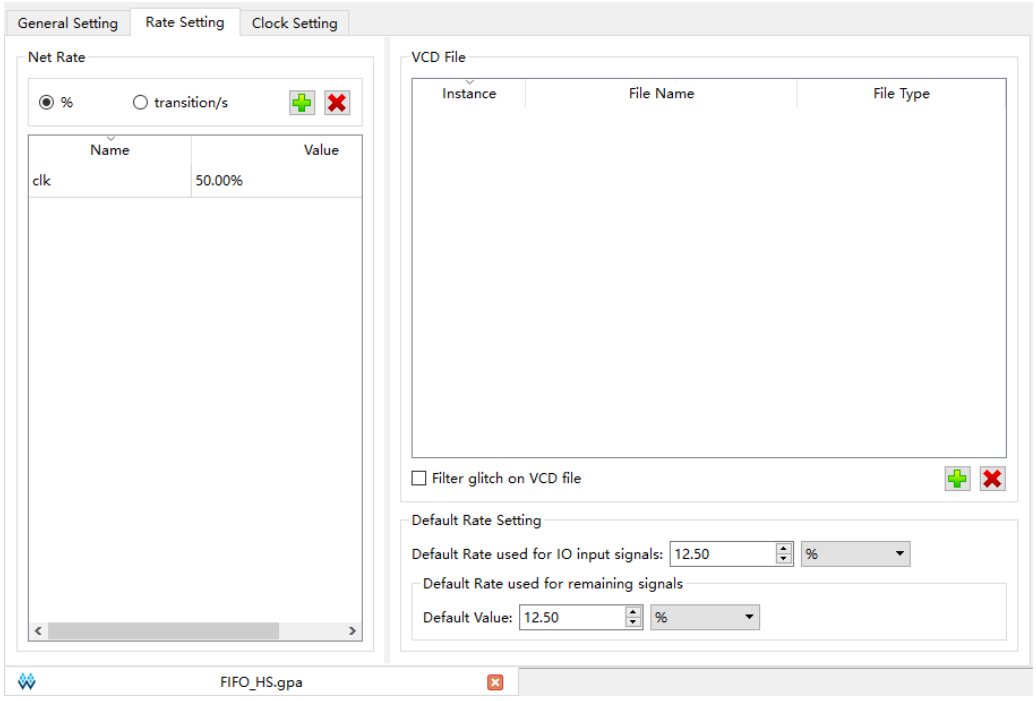
**Figure 3-26 General Setting Configuration**




### Rate Setting

In this design, the transition rate of clk is 50% and the remaining signals use the default value 12.5%, as shown in Figure 3-27.

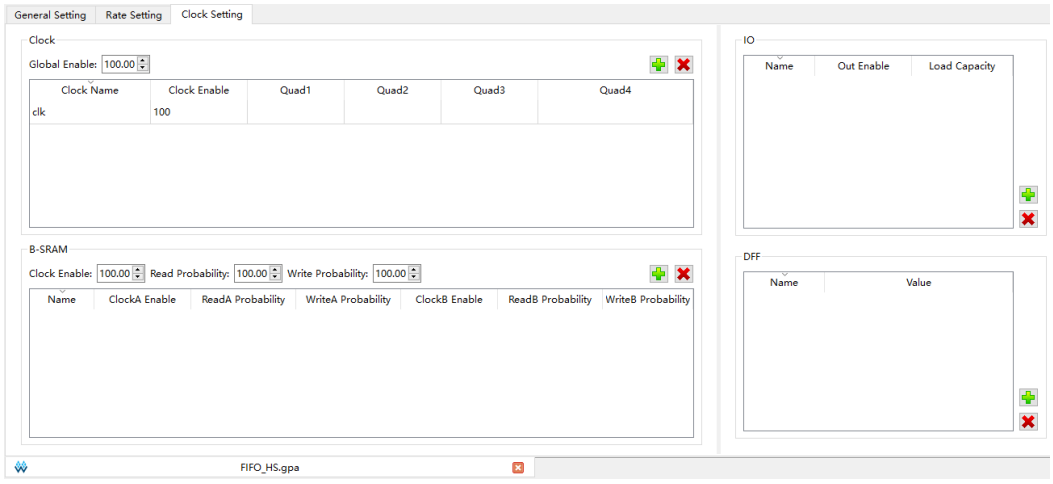
**Figure 3-27 Rate Setting Configuration**




## Clock Setting

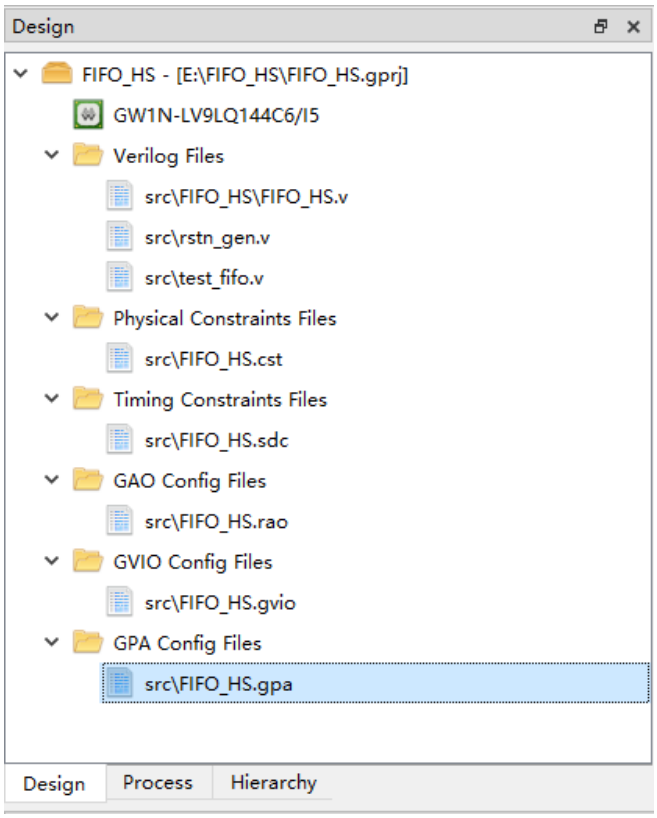
In this design, the clock is created in the timing analysis, and the rest are not set, as shown in Figure 3-28.

**Figure 3-28 Clock Setting Configuration**




After configuration, click "Save" to finish and the design window is as shown in Figure 3-29.

**Figure 3-29 GPA Config Files**




In PnR, if there is no GPA config file, the PnR will be automatically performed. If there is a GPA config file, the PnR will be performed according to the GPA config file.

# 3.10 Place & Route

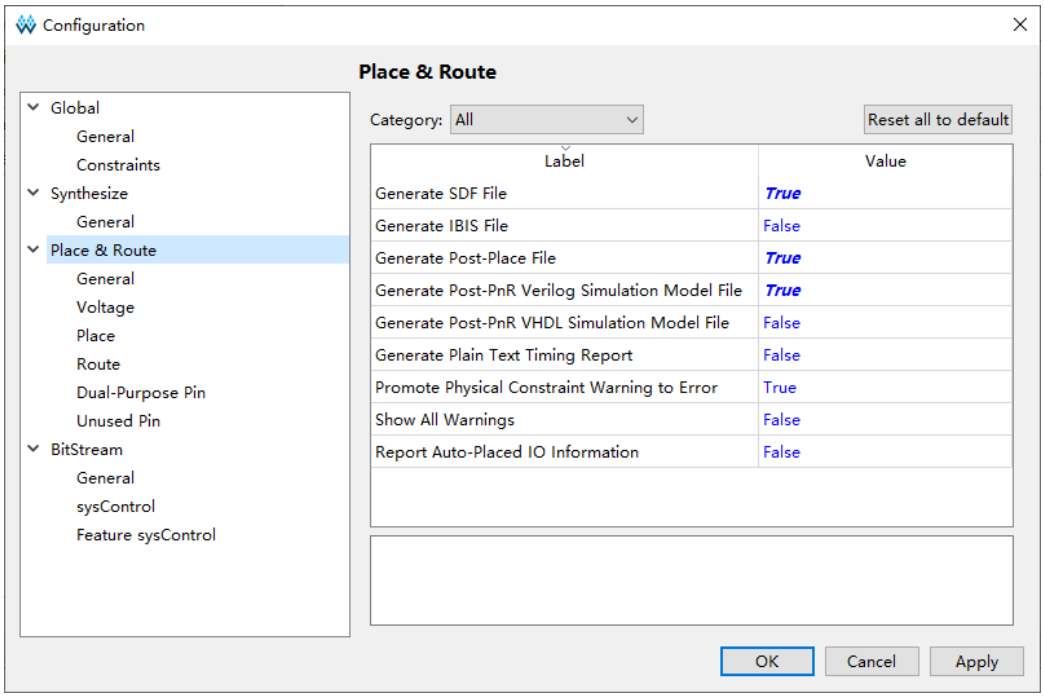
After synthesis and the creation of physical constraints files, timing constraints file, GPA config file as required, you can start PnR.

## 3.10.1 Configuration

Select "Process > Place & Route (right-click) > Configuration" to open "Configuration" dialog box to configure Place & Route and Bitstream. For the details, see *SUG100, Gowin Software User Guide*.

In this design, "Generate SDF File", "Generate Post-Place File" and "Generate Post-PNR Verilog Simulation Model File" in "General" option are configured to True. "Place output register to IOB" in "Plcae" option is configured to False, and the rest options use default values, as shown in Figure 3-30.

**Figure 3-30 Place & Route Configuration**

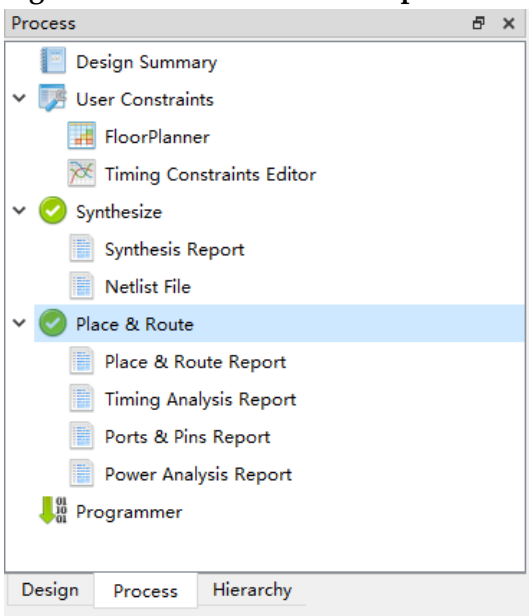



## 3.10.2 Run PnR

After configuration, you can run PnR.

Double-click Place & Route in Process window to start PnR based on physical constraints and GAO configuration, start timing analysis based on timing constraints, and start power analysis based on power analysis configuration. After PnR, the icon before the Place & Route changes to " " , as shown in Figure 3-31.

**Figure 3-31 Place & Route Completed**




After finishing PnR, the pnr folder is generated under the project creation path \impl, as shown in Figure 3-32. The folder contains all the

files generated in PnR, including the bitstream file, the netlist file after PnR, and the output reports. For the details, refer to 3.13 Output Files.

**Figure 3-32 PnR Directory**

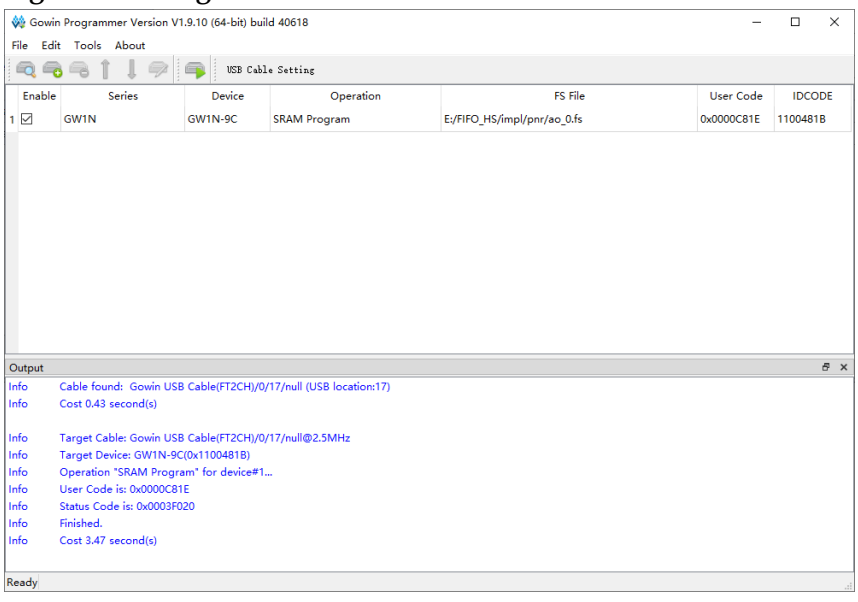
| Name | Date modified | Type | Size |
|---|---|---|---|
| ao_0.fs | 5/31/2022 15:51 | FS File | 1,732 KB |
| cmd.do | 5/31/2022 15:51 | DO File | 1 KB |
| device.cfg | 5/31/2022 15:51 | CFG File | 1 KB |
| FIFO_HS.db | 5/31/2022 15:51 | Data Base File | 43 KB |
| FIFO_HS.log | 5/31/2022 15:51 | LOG File | 2 KB |
| FIFO_HS.pin.html | 5/31/2022 15:51 | 360 se HTML Doc... | 35 KB |
| FIFO_HS.posp | 5/31/2022 15:51 | POSP File | 1 KB |
| FIFO_HS.power.html | 5/31/2022 15:51 | 360 se HTML Doc... | 8 KB |
| FIFO_HS.rpt.html | 5/31/2022 15:51 | 360 se HTML Doc... | 40 KB |
| FIFO_HS.rpt.txt | 5/31/2022 15:51 | TXT File | 29 KB |
| FIFO_HS.sdf | 5/31/2022 15:51 | SDF File | 2,321 KB |
| FIFO_HS.timing_paths | 5/31/2022 15:51 | TIMING_PATHS File | 32 KB |
| FIFO_HS.tr.html | 5/31/2022 15:51 | 360 se HTML Doc... | 1 KB |
| FIFO_HS.vo | 5/31/2022 15:51 | VO File | 561 KB |
| FIFO_HS_tr_cata.html | 5/31/2022 15:51 | 360 se HTML Doc... | 8 KB |
| FIFO_HS_tr_content.html | 5/31/2022 15:51 | 360 se HTML Doc... | 844 KB |

# 3.11 Download Bitstream

Run Place & route to generate the bitstream file and download it with Programmer to verify the design. For the usage, please see *SUG502, Gowin Programmer User Guide*.

Select "Process > Program Device (double-click)" to open Programmer, and the programmer automatically identifies the bitstream file. After the development board is ready, click "Program/Configure" to download the bitstream to the development board. Figure 3-33 shows the completion of the bitstream download.

**Figure 3-33 Programmer**

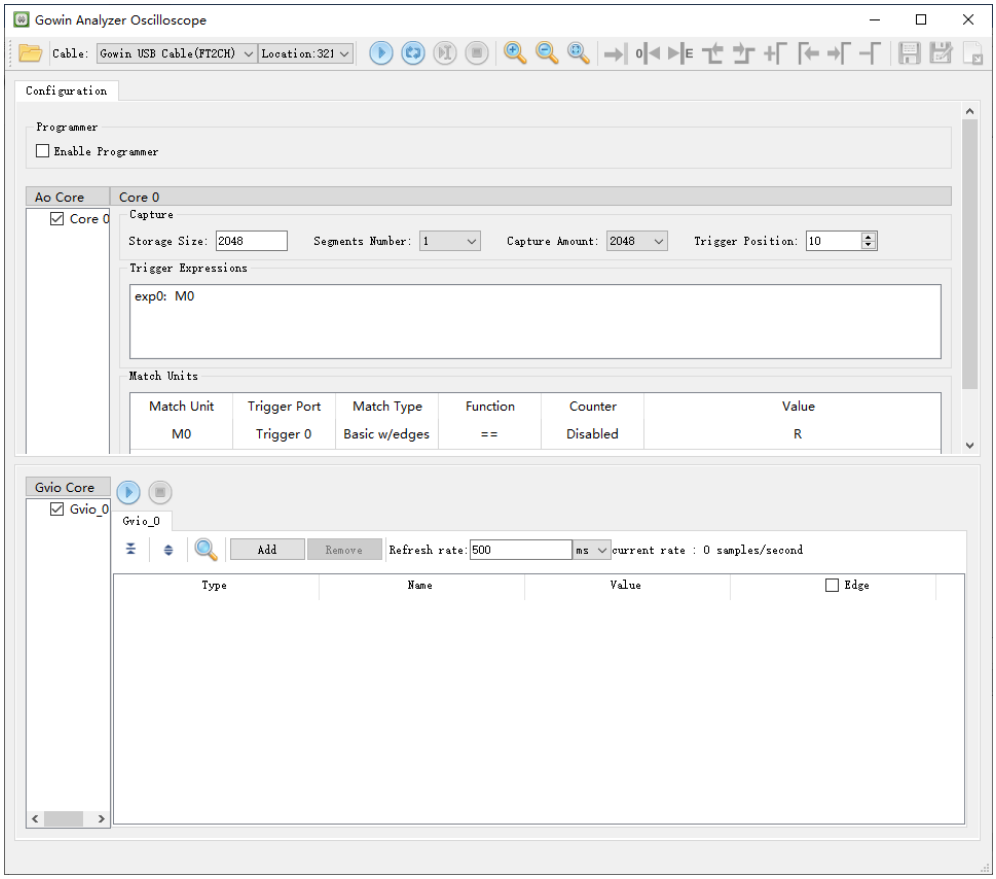


# 3.12 Debugging with GVIO and Data Acquisition with GAO

After the bitstream is downloaded, you can use GAO to verify the design or use GVIO to debug. For the GAO usage, see *SUG114, Gowin Analyzer Oscilloscope User Guide*; for the GVIO usage, see SUG1189, Gowin Virtual Input Output User Guide.

Click the GAO icon in the Gowin Software toolbar to open the GAO interface, which automatically identifies the .gao and .gvio config files, as shown in Figure 3-34.

**Figure 3-34 Jointly Debugging with GAO and GVIO**




There are two "Start" buttons in the interface. The upper "Start" button controls the operation of GAO, and the lower "Start" button controls the operation of GVIO. GAO and GVIO can run simultaneously or independently. Take the simultaneous operation of GAO and GVIO as an example in the followings.

The gvio_test signal acts on the rst_n signal in the design, which is active high and is stimulated through GVIO. When gvio_test is low, it does not affect the FIFO HS design, as shown in Figure 3-35. When gvio_test is high, the design remains in a reset state. Click on "GAO > Configuration" in the Gowin Analyzer Oscilloscope interface, double-click "Match Units", modify the "Value" to X, and the captured waveform is as shown in Figure 3-36.

Click the "Start" icon in the GAO interface to capture data. After finishing capturing data, a window is generated to display the waveform. The window supports cursor, zoom-out and so on so as to facilitate you to analyze the data.

**Figure 3-35 Jointly Debugging and Sampling with GAO and GVIO when gvio_test=0**

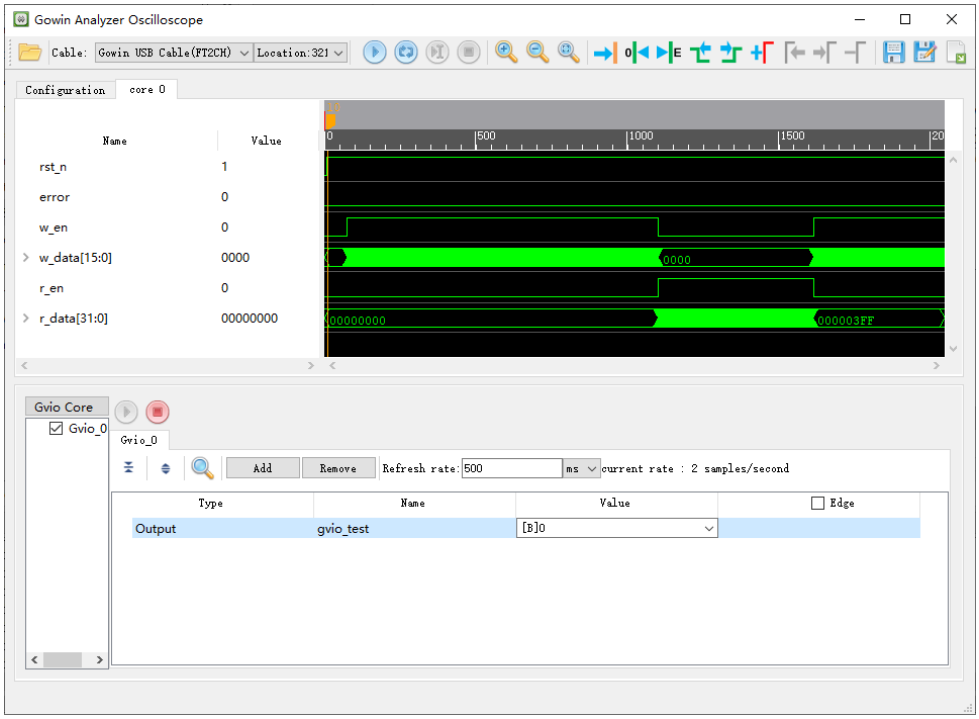
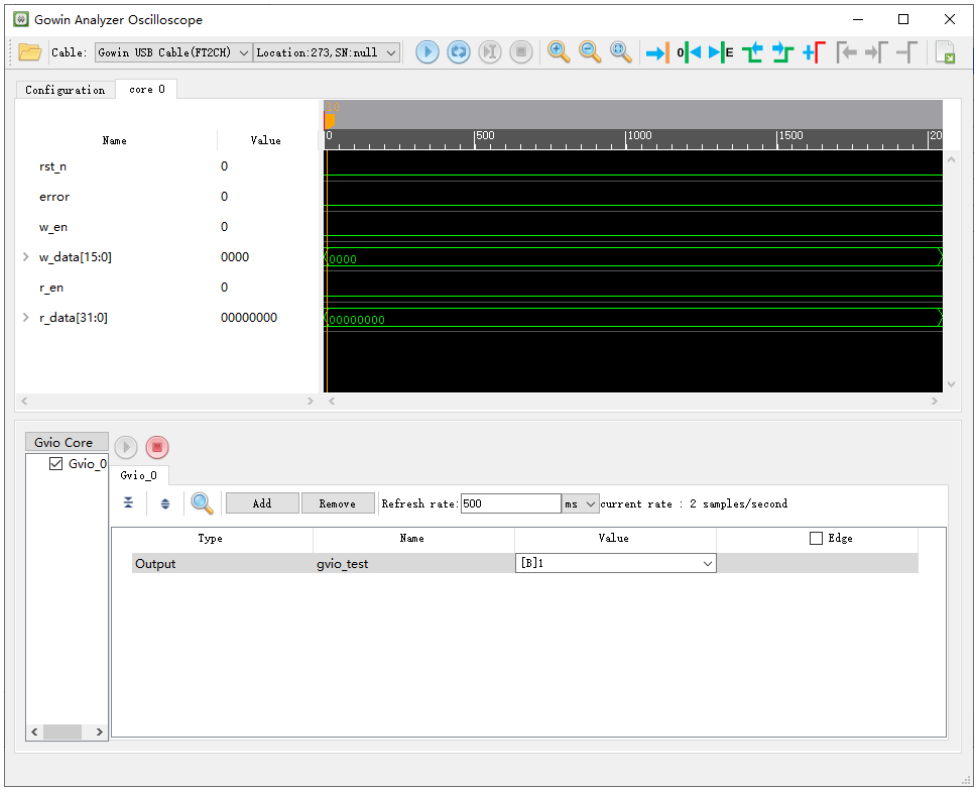



**Figure 3-36 Jointly Debugging and Sampling with GAO and GVIO when gvio_test=1**



# 3.13 Output Files

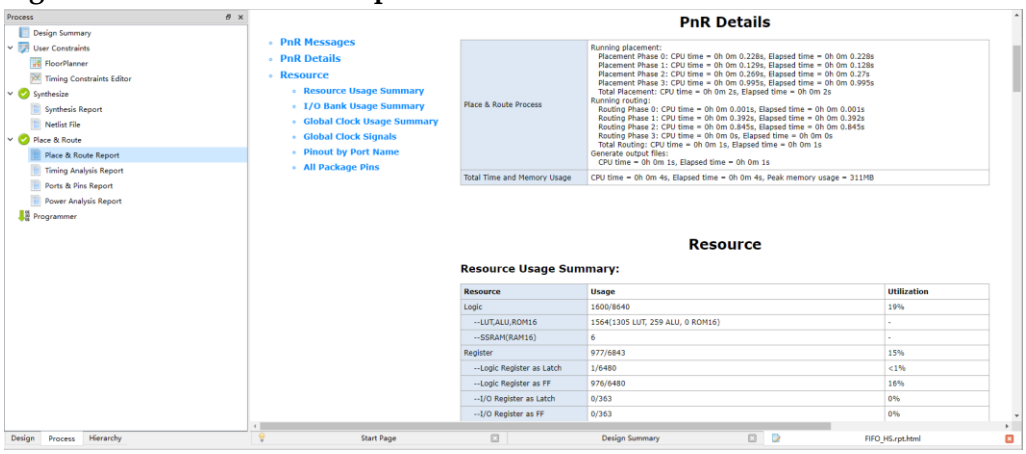
## 3.13.1 Place & Route Report

The Place & Route Report describes the resource, memory consumption, time consumption, etc. occupied by the user design, with the file extension name .rpt.html. Check the *.rpt.html file for further details.

Double-click "Place & Route Report" in the Process window to open Place & Route report, as shown in Figure 3-37.

For the details, refer to 6.2 Place & Route Report of *SUG100 Gowin Software User Guide*.

**Figure 3-37 Place & Route Report**



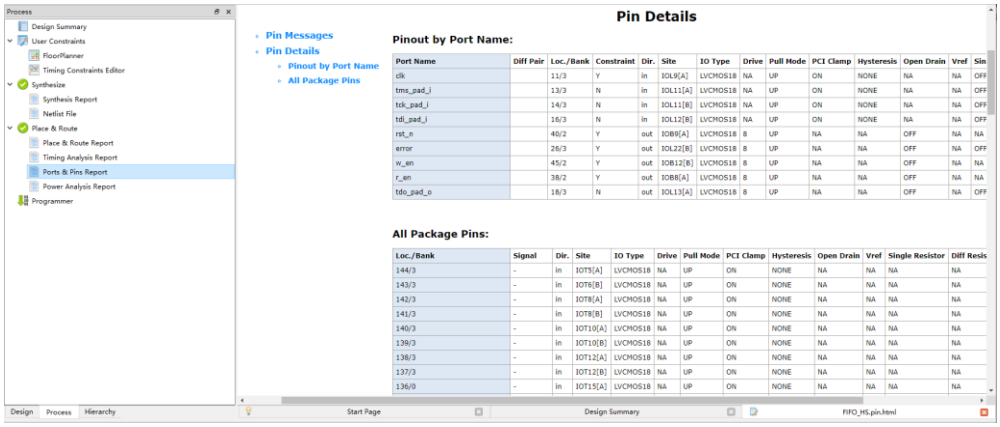

## 3.13.2 Ports and Pins Report

The Ports and Pins Report is the ports and pins files after placement. It includes port types, attributes, and locations, etc. The generated file is saved with extension name .pin.html. Check the .pin.html file for further details.

Double-click Ports & Pins Report in the Process window to open Ports & Pins Report, as shown in Figure 3-38.

For the details, refer to 6.3 Ports & Pins Report of *SUG100，Gowin Software User Guide*.

**Figure 3-38 Ports & Pins Report**



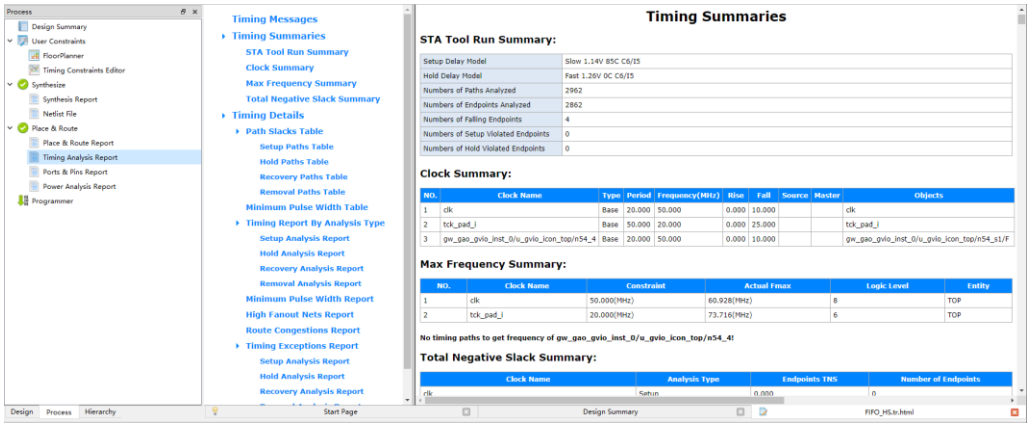
## 3.13.3 Timing Report

The timing report includes setup check, hold check, recovery time check, removal time check, min. clock pulse check, max. fan out path, Place & Route congestion report, etc. by default. The timing report also includes the max. frequency report.

Double-click Timing Analysis Report in the Process window to open the timing analysis report for the project, as shown in Figure 3-39.

For the details, please refer to *SUG940, Gowin Design Timing Constraints User Guide*.

**Figure 3-39 Timing Report**



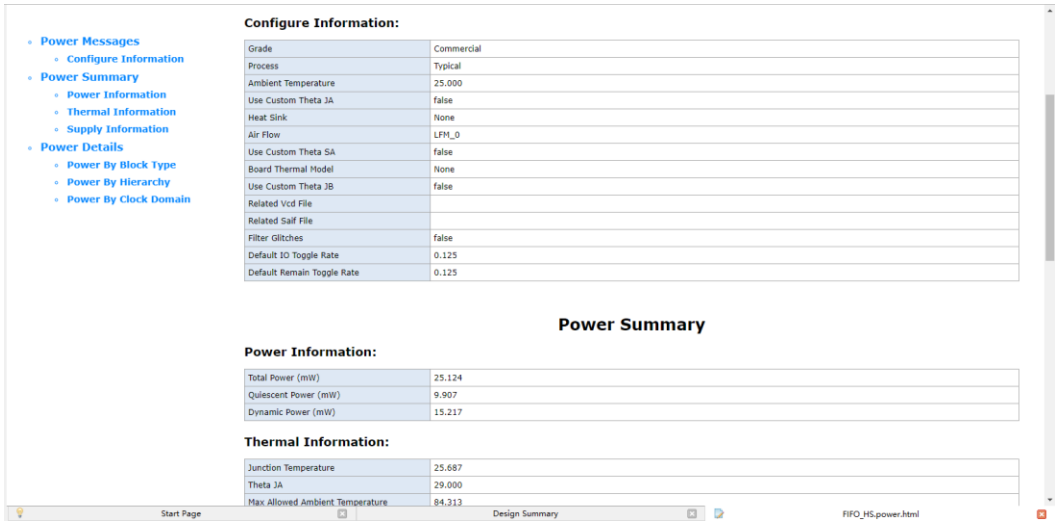

## 3.13.4 Power Analysis Report

The Power Analysis Report helps you evaluate the basic power consumption of your design.

Double-click Power Analysis Report in the Process window to open the power analysis report as shown in Figure 3-40.

For the details, please refer to chapter 4 Power Analysis Report of *SUG282, Gowin Power Analysis User Guide*.

**Figure 3-40 Power Analysis Report**

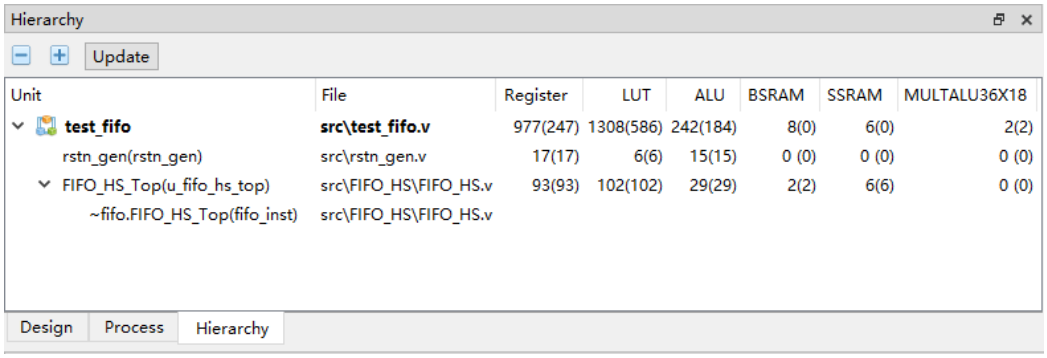


# 3.14 File Encryption

## 3.14.1 Source File Encryption

When you need to encrypt and protect source files, you can encrypt the selected module and its sub modules in Hierarchy window, as shown in Figure 3-41. For details, see *SUG100, Gowin Software User Guide*.

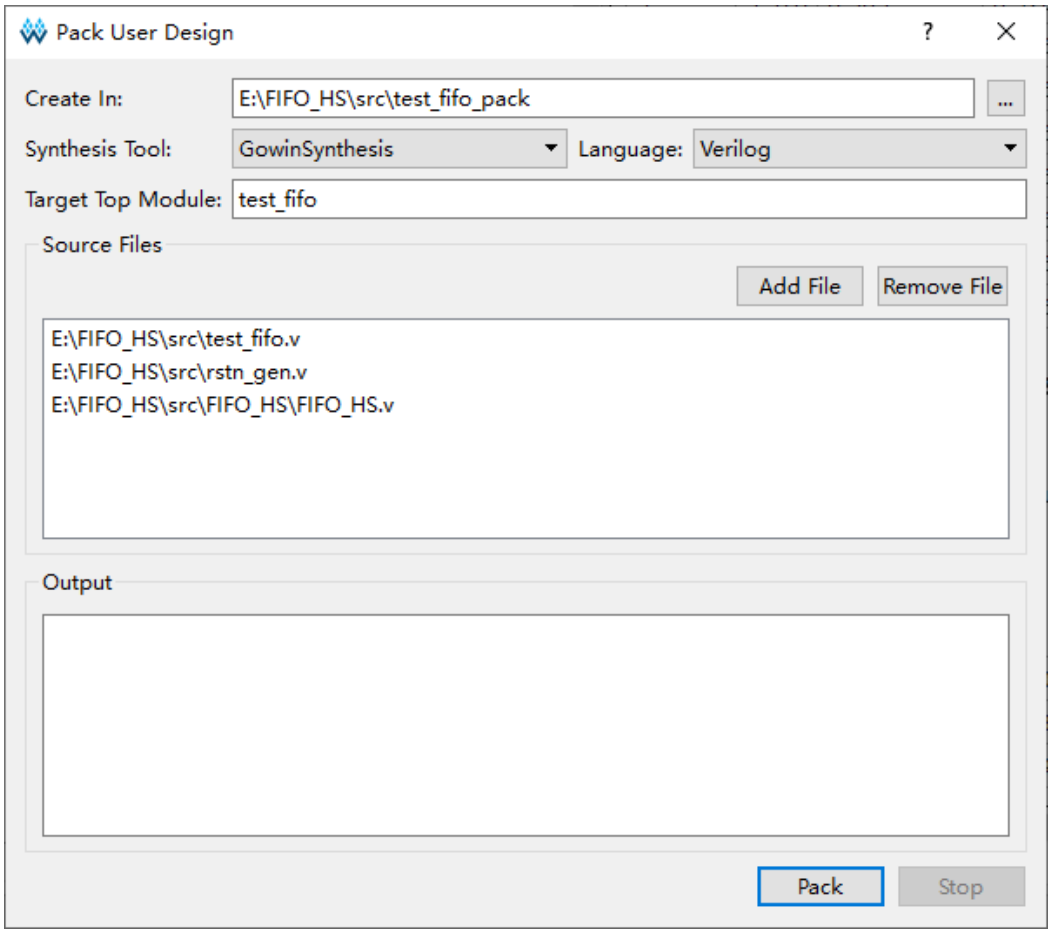
**Figure 3-41 Hierarchy Window**




Take module module test_fifo as an example to introduce the file encryption.

You can right-click test_fifo in the Hierarchy window and select "Pack User Design" in the right-click list to open the dialog box, as shown in Figure 3-42.

**Figure 3-42 Pack User Design Dialog Box**




Select test_fifo as the top module. Click "Pack" to start encryption. The relevant information will be printed in the Output window.

After the encryption, two files are generated under the destination path (E:\FIFO_HS\src\test_fifo_pack): test_fifo_gowin.vp and test_fifo_sim.v.

- test_fifo_gowin.vp: Encrypted files that can be used by others.
- test_fifo_sim.v: Flattened synthesized plaintext netlist file that can be used for simulation.

## 3.14.2 Simulation File Encryption

The simulation file provided by Gowin is plaintext. In order to protect the simulation file, it can be encrypted by using a third-party simulation software, such as Modelsim and VCS, and the license of the tool needs to be obtained. Here it uses test_fifo_sim.v as an example to introduce the encryption.

**Encryption by Modelsim**

When using Modelsim, the steps to encrypt the simulation file are as follows:

1. Add macro `protect and `endprotect before and after the encrypted in the simulation file test_fifo_sim.v.

2. Run command: vlog +protect test_fifo_sim.v.
3. After running the command, test_fifo_sim.vp is generated in the work library, which is the encrypted file of test_fifo_sim.v that can be used for Modelsim simulation.

**Encryption by VCS**

When using VCS, the steps to encrypt the simulation file are as follows:

1. Add macro `protect128 and `endprotect128 before and after the encrypted in the simulation file test_fifo_sim.v.
2. Run command: vcs +v2k -protect128 test_fifo_sim.v.
3. After running the command, test_fifo_sim.vp is generated under the current path, which is the encrypted file of test_fifo_sim.v that can be used for VCS simulation.

# 4 Tcl

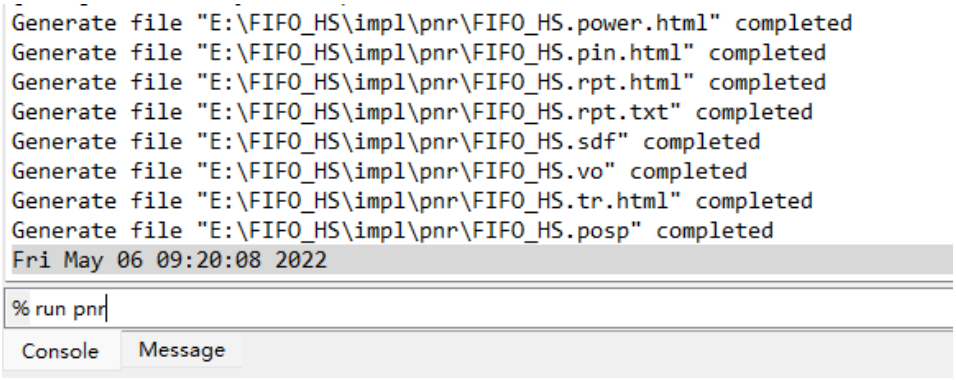
The previous chapters introduce the way to implement the entire design process by using GUI. Gowin Software also provides tcl commands for some settings. Take FIFO HS design in Windows as an example to introduce the usage of tcl commands. For the details, see Chapter 8 Tcl Commands of *SUG100, Gowin Software User Guide*.

## 4.1 Tcl Execution

### 4.1.1 Execution Using Tcl Editing Window

At the bottom of the Console page is the tcl editing window, where you can enter the tcl commands and press Enter to run, as shown in Figure 4-1.

**Figure 4-1 Tcl Editing Window**

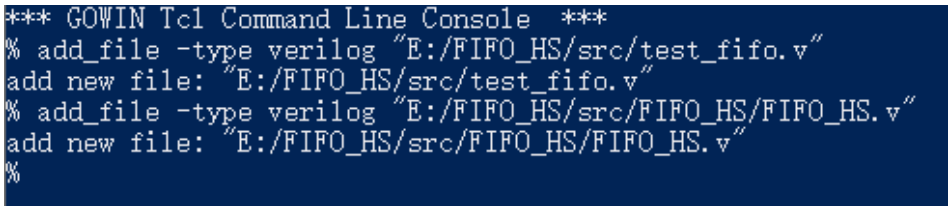



### 4.1.2 Execution Using Tcl Command Line

Start command: \x.x\IDE\bin\gw_sh.exe [script file] under the installation directory

The First Way: enter gw_sh.exe to start. This mode executes in the same way as the Tcl editing window, executing tcl commands one by one, as shown in Figure 4-2.

**Figure 4-2 Tcl Command Line Example**




The Second Way: use gw_sh.exe [script file] to execute the script file, shown in Figure 4-3. Tcl script file can contain all the supported tcl commands, such as, device, design file, option, and run information. Tcl script file can be generated by handwriting or saveto command, but saveto command The tcl script file can be generated by hand or by saveto command, but the saveto command does not include the run command when generating the tcl script, so you can add the run command if needed. For tcl script details, see 4.2 Tcl Quick Start.

**Figure 4-3 Tcl Script File Example**


# 4.2 Tcl Quick Start

The use of Tcl commands in command line mode is the same as that in Tcl command editing window. Taking gw_sh.exe [script file] as an example, we will introduce the usage of Tcl commands. Using this Tcl script, a new project, FIFO_HS_tcl, is created based on the existing project FIFO_HS; then load the design file from FIFO_HS into the new project FIFO_HS_tcl, configure the project, and then run the flow. The content of the Tcl script is described as follows:

```
#create project
create_project -name FIFO_HS_tcl -dir E:/tcl -pn GW1N-LV9LQ144C6/I5 -device_version C
#import design file to FIFO_HS_tcl/src
import_files -file "E:/FIFO_HS/src/FIFO_HS/FIFO_HS.v"
import_files -file "E:/FIFO_HS/src/rstn_gen.v"
import_files -file "E:/FIFO_HS/src/test_fifo.v"
import_files -file "E:/FIFO_HS/src/FIFO_HS.cst"
import_files -file "E:/FIFO_HS/src/FIFO_HS.sdc"
import_files -file "E:/FIFO_HS/src/FIFO_HS.rao"
import_files -file "E:/FIFO_HS/src/FIFO_HS.gvio"
import_files -file "E:/FIFO_HS/src/FIFO_HS.gpa"
##########Global Configuration##########
```

```
#set output base name
set_option -output_base_name FIFO_HS
#set global frequency
set_option -global_freq 50.000
##########Synthesis Configuration########
#set synthesis tool
set_option -synthesis_tool gowinsynthesis
#set top module
set_option -top_module test_fifo
#set verilog language
set_option -verilog_std sysv2017
#set ram r/w check
set_option -rw_check_on_ram 1
##########Place & Route Configuration#########
#set generate sdf file
set_option -gen_sdf 1
#set generate post-place file
set_option -gen_posp 1
#set generate post-pnr verilog simulation model file
set_option -gen_verilog_sim_netlist 1
#set place output registers to IOB
set_option -oreg_in_iob 0
############################################
#set run process
run all
```


GOWIN
PROGRAMMING FOR THE FUTURE