Gowin YunYuan Software

User Guide

SUG100-1.4E,11/15/2018

Copyright©2018 Guangdong Gowin Semiconductor Corporation. All Rights Reserved.

No part of this document may be reproduced or transmitted in any form or by any denotes, electronic, mechanical, photocopying, recording or otherwise, without the prior written consent of GOWINSEMI.

Disclaimer

GOWINSEMI®, LittleBee®, Arora™, and the GOWINSEMI logos are trademarks of GOWINSEMI and are registered in China, the U.S. Patent and Trademark Office and other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders, as described at www.gowinsemi.com. 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. All information in this document should be treated as preliminary. 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
09/06/2018	1.2E	Initial version published.
11/06/2018	1.3E	GW1NZ-1 and GW1NSR-2C supported.
11/15/2018	1.4E	● GW1N-6ES, GW1N-9ES and GW1NR-9ES devices removed; ● GW1N-1-MBGA160 and GW1N-1-PBGA204 packages removed; ● Supports GW1NSR-2; ● GW2AR-18-eLQFP144 package added.

Contents

List of Figures

List of Tables

1 About This Guide

1.1 Purpose

This manual predominantly documents Gowin software installation and operation and is designed to help users employ the software functionality to aid complex design. The software screenshots and the supported products listed in this manual are based on Windows 1.8.1Beta. 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 Supported Products

The information in this guide applies to the following products:

- GW1N series of FPGA products: GW1N-1, GW1N-2, GW1N-2B, GW1N-4, GW1N-4B, GW1N-6, GW1N-9;
- GW1NR series of FPGA products: GW1NR-4, GW1NR-4B, and GW1NR-9;
- GW1NS series of FPGA products: GW1NS-2, GW1NS-2C;
- GW2A series of FPGA products: GW2A-55 and GW2A-18;
- GW2AR series of FPGA products: GW2AR-18;
- GW1NZ series of FPGA products: GW1NZ-1;
- GW1NSR series of FPGA products: GW1NSR-2C and GW1NSR-2.

1.3 Related Documents

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

1. GW1N series of Products Data Sheet
2. GW1NR series of FPGA Products Data Sheet
3. GW1NS series of FPGA Products Data Sheet
4. GW2A series of FPGA Products Data Sheet
5. GW2AR series of FPGA Products Data Sheet

6. GW1NZ series of FPGA Products Data Sheet
7. GW1NSR series of FPGA Products Data Sheet
8. Gowin Design Constraints Guide
9. Gowin FPGA Design Guide
10. Gowin Analyzer Oscilloscope User Guide
11. Gowin Power Analyzer User Guide
12. Gowin IP Core Generator User Guide

1.4 Abbreviations and Terminology

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

Table 1-1 Abbreviations and Terminology

Abbreviations and Terminology	Name
CRC	Cyclic Redundancy Check
CS	Wafer Level Chip Scale Package
DLL	Delay-locked Loop
FPGA	Field Programmable Gate Array
FF	Flip-Flop
IDE	Integrated Development Environment
IP Core	Intellectual Property Core
LQ	Low-profile Quad Flat Package
MAC	Media Access Control
MG	Micro Ball Grid Array Package
PC	Personal Computer
PLL	Phase-locked Loop
PG	Plastic Ball Grid Array
QN	Quad Flat No-lead Package
RTL	Register Transfer Level
UG	Ultra Ball Grid Array Package

1.5 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 using the information provided below.

Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

+Tel: +86 755 8262 0391

2General Description

2.1 Introduction

Gowin Yunnan Software is the new hardware development design environment for Gowin products. It supports generic hardware description language, and helps users to quickly implement code synthesis, manage placement and routing, generate and download bitstreams, etc. The Gowin Yunyuan Software also incorporates the IP Core Generator, which is dedigned to help developers quickly implement complex designs and the online debug tool, Gowin Analysis Oscilloscope, which can help users to efficiently identify and assess signal design issues.

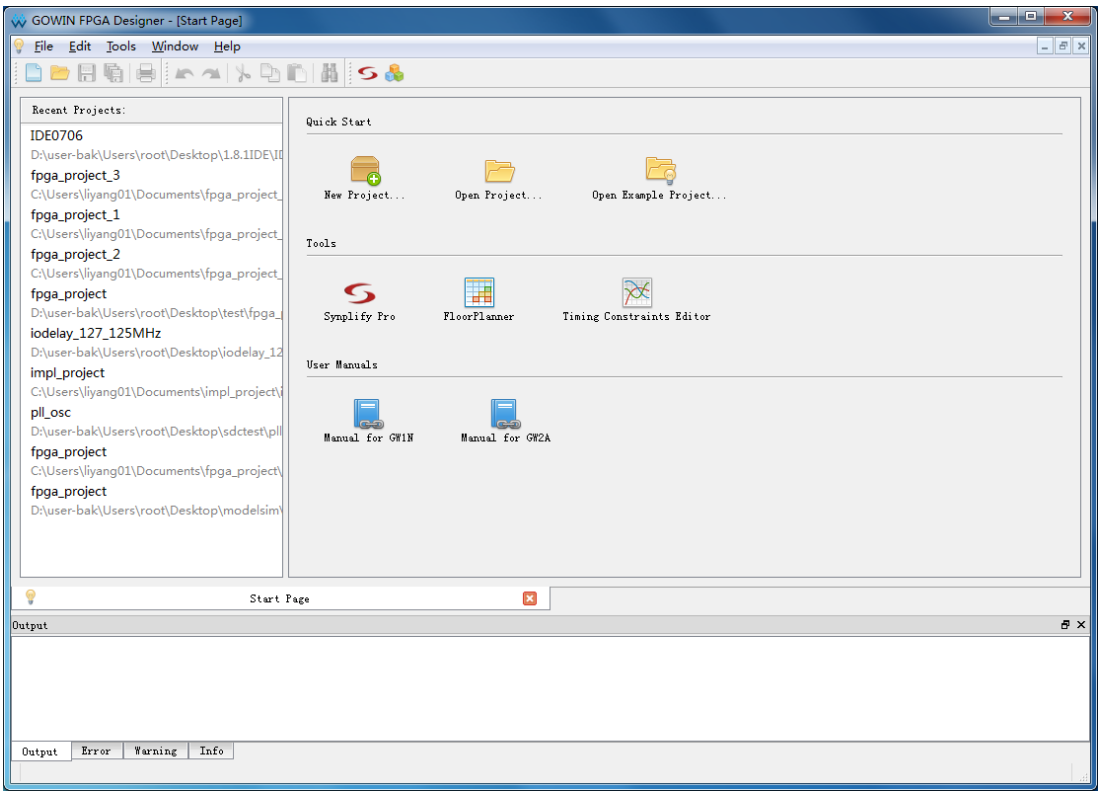
To meet different user demands, Gowin YunYuan Software supports the creation of RTL and Post-Synthesis projects.

The input file for RTL project is the RTL file which contains hardware description language.

The input file for post-synthesis project is the netlist file, which is generated by user RTL synthesis.

The Gowin YunYuan Software provides a GUI for both types of projects. Users can employ this software to quickly edit the constraints files, check the running results of SynplifyPro (RTL project only) and Place & Route, and start the GOWINSEMI FPGA download tool immediately to download the bitstream to the chip and implement the required functions. The interface of Gowin YunYuan Software is as shown in Figure 2-1.

Figure 2-1 GUI



2.2 Supported Devices

Table 2-1 lists the devices Gowin YunYuan software supports and the associated resources, packages, and speed.

Table 2-1 Devices Supported

Device	LUT4s	Flip-Flops	MULT18 x 18	PLL	Package	Speed
GW1N-1	1,152	864	0	1	WLCSP30	4/5/6
					QN32	
					QFN48	
					LQFN100	
					LQFN144	
					MBGA160	
					PBGA204	
GW1N-2	2,304	1,728	16	2	WLCSP72	4/5/6
					QFN32	
					QFN88	
					QFN48	
					LQFP100	
					LQFP144	
					MBGA160	
					PBGA256	
					PBGA256M	
GW1N-2B	2,304	1,728	16	2	WLCSP72	
					QFN32	
					QFN88	
					QFN48	4/5/6
					LQFP100	
					LQFP144	
					MBGA160	
					PBGA256	
					PBGA256M	
GW1N-4	4,608	3,456	16	2	WLCSP72	4/5/6
					QFN32	
					QFN88	
					LQFP100	
					LQFP144	
					MBGA160	
					PBGA256	
					QFN48	
					PBGA256M	
GW1N-4B	4,608	3,456	16	2	WLCSP72	4/5/6
					QFN32	
					QFN88	
					LQFP100	
					LQFP144	
					MBGA160	

Device	LUT4s	Flip-Flops	MULT18 x 18	PLL	Package	Speed
					PBGA256	
					QFN48	
					PBGA256M	
GW1NR-4	4608	3456	16	2	QFN88	4/5/6
					MBGA81	
GW1NR-4B	4608	3456	16	2	QFN88	4/5/6
					MBGA81	
GW1N-6	6912	5184	20	2	LQFP144	4/5/6
					UBGA332	
					PBGA256	
					LQFP100	
					QFN88	
					QFN48	
					LQFP176	
					MBGA160	
					WLCSP64	
GW1N-6ES	6912	5184	20	2	LQFP144	4/5/6
					UBGA332	
					PBGA256	
					LQFP100	
					QFN88	
					QFN48	
					LQFP176	
					MBGA160	
					WLCSP64	
GW1N-9	8640	6480	20	2	LQFP144	4/5/6
					UBGA332	
					PBGA256	
					LQFP100	
					QFN88	
					QFN48	
					LQFP176	
					MBGA160	
					WLCSP64	
GW1N-9ES	8640	6480	20	2	LQFP144	4/5/6
					UBGA332	
					PBGA256	
					LQFP100	
					QFN88	
					QFN48	
					LQFP176	
					MBGA160	
					WLCSP64	
GW1NR-9	8640	6480	20	2	LQFP144	4/5/6
					QFN88	
GW1NR-9ES	8640	6480	20	2	LQFP144	4/5/6

Device	LUT4s	Flip-Flops	MULT18 x 18	PLL	Package	Speed
					QFN88	
GW1NS-2	1728	1296	0	1	WLCSP36	4/5/6
					QFN32	
					QFN48	
					LQFP144	
					QFN32U	
GW1NS-2C	1728	1296	0	1	WLCSP36	4/5/6
					QFN32	
					QFN48	
					LQFP144	
					QFN32U	
GW1NZ-1	1152	864	0	1	WLCSP16	4/5/6
					QFN32	
GW1NSR-2	1728	1080	0	1	QFN48	4/5/6
GW1NSR-2C	1728	1080	0	1	QFN48	4/5/6
GW2A-18	20,736	15,552	48	4	PBGA484	6/7/8
					PBGA256	
					LQFP144	
GW2AR-18	20,736	15,552	48	4	QFN88	6/7/8
					LQFP144	
					LQFP176	
					eLQFP144	
GW2A-55	54,720	41,040	40	6	PBGA484	6/7/8
					PBGA1156	

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.

3 Installation

3.1 Environment Requirement

Windows: Win7/8/10(64bit), XP(32bit)

Linux: Centos6/7(64bit)、Red Hat 6/7(64bit)、SUSE 11/12(64bit)

Note!

YunYuan software does not currently support linux Centos5.

3.2 Software Download

The Gowin YunYuan software installation packages consist of Windows and Linux versions. They are available for download on the Gowin Website:

- Installation package for Windows: Gowin Yunyuan for win(Vx.x.xbeta).exe, with the download link: http://www.gowinsemi.com/support/download_eda/.
- Installation package for Linux is compressed with the name “Gowin Yunyuan for linux(Vx.x.xbeta).rar”. The download link is http://www.gowinsemi.com/support/download_eda/.

Note!

- Users need to register and log on to the Gowin website before downloading the installation package;
- The "x" in the installation package name “Vx.x.xbeta” refers to the software version.

3.3 Software Installation

Note!

- You must close anti-virus programs, such as 360 or Kingsoft AntiVirus, etc. before installing Gowin YunYuan software'
- The installation path should not contain any Chinese characters or spaces.
- Before installing any new versions of the Gowin YunYuan software, old versions should be installed.
- Table 3-1 shows the product options for the installation of Gowin YunYuan for Windows.

Table 3-1 Components to Install

Components	Description	Comments
Gowin YunYuan GUI	Gowin YunYuan GUI	Installation directory of the corresponding executable file: \x.x\IDE\bin\gowin_ide.exe
Pnr tool	Gowin place and routing tool, including software simulation library, IBS module, and software related documents	● Installation directory of the corresponding executable file for P&R software: \x.x\Pnr\bin\gowin.exe; ● Software simulation directory: \x.x\Pnr\lib; ● IBS module directory: \x.x\Pnr\ibs; ● Software related documents directory: \x.x\Pnr\doc.
Device programmer	Gowin devices download tool, including programmer user documents	● Installation directory of the corresponding executable file: \x.x\Programmer\bin\programmer.exe; ● The corresponding documents directory: \x.x\Programmer\doc.
SyplifyPro for Gowin	Gowin synthesis tool, including user documents	● Installation directory of the corresponding executable file: \x.x\SynplifyPro\bin\synplify_pro.exe; ● The corresponding documents directory: \x.x\SynplifyPro\doc.
Floorplanner	Gowin physical constraints editing tool	Installation directory of the corresponding executable file: \x.x\Floorplan\bin\floorplan.exe

Note!

"x.x" in the directory name in the above table refers to the software version.

Windows Installation

Refer to the steps below to install the Gowin Yunyuan software on Windows.

1. Double-click on the installer to launch the installation process. Follow the install wizard to complete the installation.

   Note!

   - In the process of installation, users can change the installation directory as required.
   - After installation, the software shortcut will be created on PC desktop by default "  ".

2. Double-click the shortcut “  ” to open Gowin YunYuan software.

Linux Installation

After decompressing the files, run the following command with root authority to open the YunYuan software: path/x.x.xBeta/IDE/bin/gw_ide.

Note!

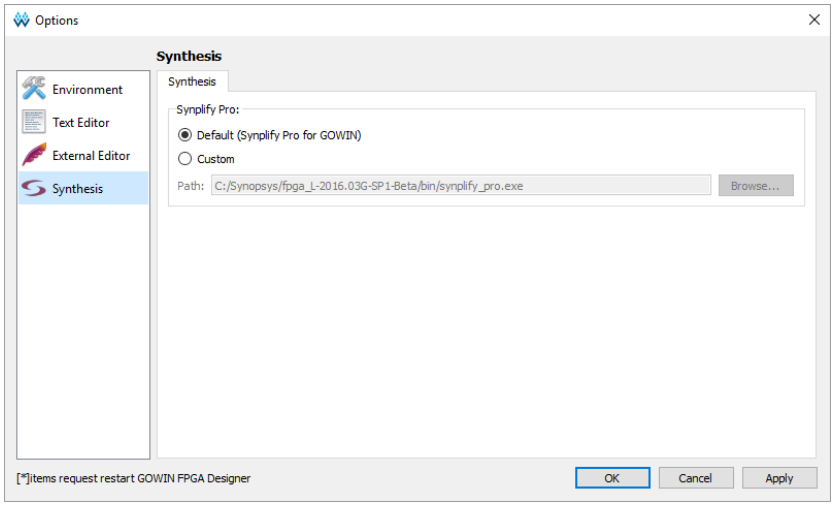
"x" in the installation package name “x.x.xbeta” refers to the software version.

Note!

If you have previously installed Gowin Yunyuan software, ensure the installation directory of the new Gowin Yunyuan software is the same as the call path of SynplifyPro.

1. Select “Tools > Options” from the menu bar;
2. Select the installation path for SynplifyPro, as shown in Figure3-1.

Figure3-1 SynplifyPro Call Path



3.4 Software License Configuration

After setting the "GOWIN_HOME" environment variable, Gowin Yunyuan software licenses are required. Gowin Yunyuan software licenses include:

- SynplifyPro License;
- Place and Routing License;

Gowin YunYuan software licenses support node-locked license and floating license.

Users can apply for the required licenses on the Gowin website http://www.gowinsemi.com/support/license/.

For license configuration, please refer to 3.5SynplifyPro License Configuration and 3.6Place and Layout License Configuration.

3.5 SynplifyPro License Configuration

3.5.1 Node-Locked License

The node-locked license is based on PC MAC and is only suitable for the users of this PC. Please refer to the following steps after acquiring the SynplifyPro Node-Locked License:

Note!

If the other partner's tool was installed in the "C:\Synopsys\" directory, and there is the license file in this directory, conflict issues will occur when Gowin Synplify is searching a license. Because Synplify will search the license in this directory first by default, no matter the environment variables are configured or not.

Taking win7, 64-bit operating system for instance, please refer to the following steps to configure the environment variables.

1. Save license file

   Save the license (licensefile.txt） to the target directory, for example, “E:\Synopsys\licensefile.txt”.

2. Configure the license.

   a). Right-click on the "Computer" link, select "properties", and click the "Advanced system settings" to open the "System Properties" window;

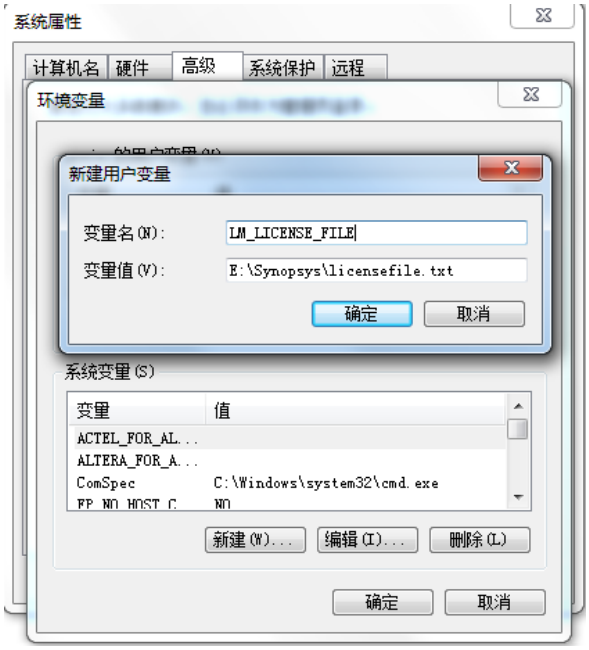
   b). In the "System Properties" View, click "Advanced > Environment Variables>New" to open the "New User Variable", as shown in Figure 3-2. Enter the variable name and the value in the "New User Variable" dialog box.

   Environment variable name:LM_LICENSE_FILE.

   Variable value: License file location, such as “E:\Synopsys\licensefile.txt ".

3. Click "OK ".

Figure 3-2 Environment Variables Setting



3.5.2 Floating License

When this configuration is employed, Gowin YunYuan is installed on your license server. Each PC in the LAN can use the floating license. Floating licenses have a user limit.

1. First, start floating license using Synopsys Common Licensing (SCL) in the Yunyuan software installation package file “GowinLicenseServerForWindows\SCL”.

   a). Click the “scl_v2018.06_windows.exe” SCL executable file to install the SCL software. Install it according to the wizard steps. Configure the "Site Information" as follows:

   - “Site ID”: The “siteid” value of the SynplifyPro License file (such as synp_license.lic), such as 29247;
   - “Site Administrator”: The administrator name of this PC;
   - “Contact Information”: Optional, the contact information of the administrator;

   b). Select the installation path according to the wizard steps, such as “C:\Synopsys\SCL”. Click "next" until the page of installation done appears. For example, the “Completing the SCL 2018.06 Setup Wizard” page appears. Click "Finish" to complete the SCL installation.

2. Go to the “2018.06\win32\bin” folder in the SCL installation path, “C:\Synopsys\SCL \2018.06\win32\bin”, for example. There are two ways to start teh floating license. Take Win7, 64 bits operating system for instance:

   - Commands Mode

   a) Copy the floating license file (such as synp_license.lic) to the local

SCL installation path;

b) Modify the PC name in the floating license file and check the MAC address;
for example:
for the "SERVER hostname1 F8BC12950972 27020" content in the license file:

- “hostname1” should be modified to the PC name;
- “F8BC12950972” is the PC MAC;
- “27020” is the port number, which should be modified per the PC installed SynplifyPro, such as “SERVER GaoYun-PC F8BC12950972 27020”.

c) Modify the snpslmd path in the floating license file.
On the basis of the "snpslmd.exe" path in SCL, modify this line “VENDOR snpslmd /path/to/snpslmd” in the license file to "VENDOR snpslmd C:\Synopsys\SCL\2018.06\win32\bin\snpslmd.exe", for example;

d) Open the cmd window and run the command to start the license service;

The command is “path\lmgrd.exe” –c “path\licensefile” –l “path\logfile” –z –s;

e) After running, check the log file in the "path\logfile" directory to confirm that if the license has been installed successfully.

- SCL Configuration Mode

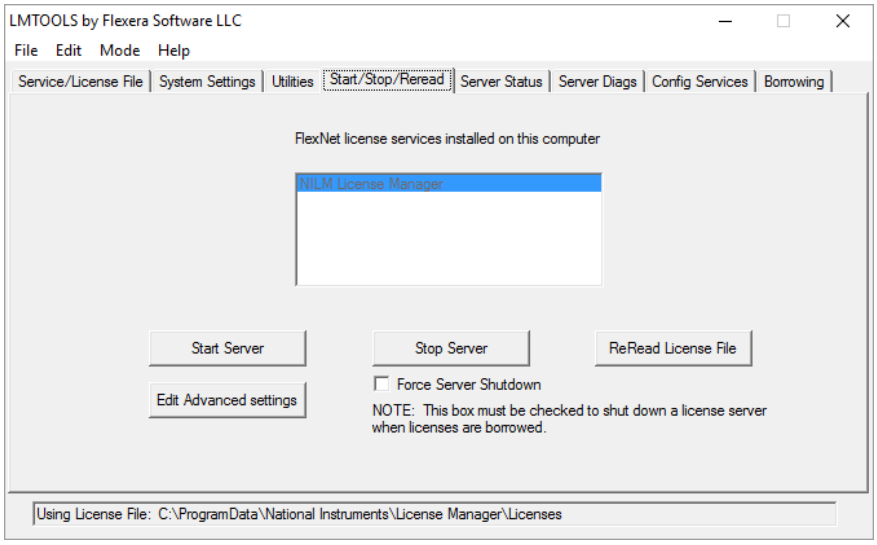
a) Copy the floating license (such as synp_license.lic) to the local SCL installation path;

b) Modify the PC name in the floating license file and check the MAC address (Refer to Step b of the command mode);

c) Modify the snpslmd path in the floating license file. Refer to Step c of the command mode;

d) Start "lmtools.exe" in snpslmd path; the GUI is as shown in Figure 3-3;

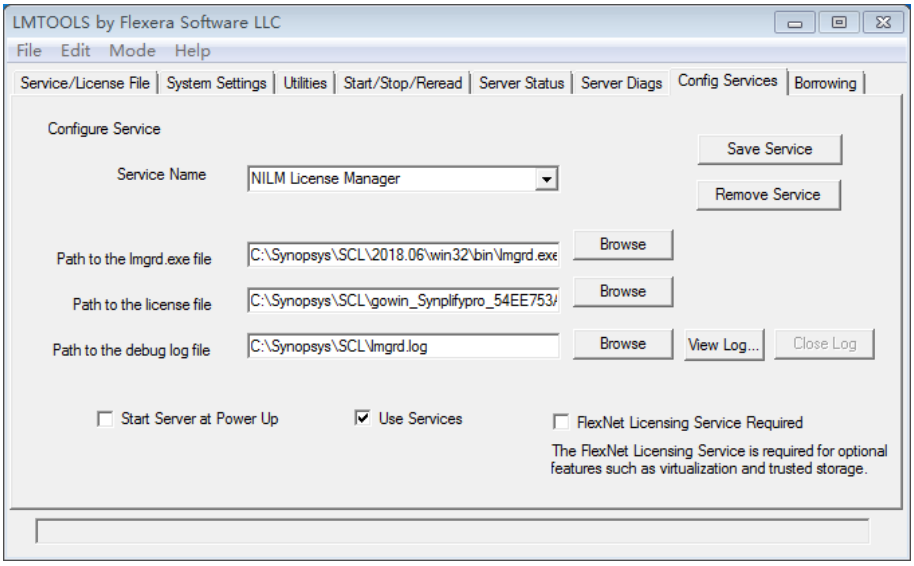
Figure 3-3 SCL Start GUI



e) See Figure 3-4, select "Config Services" to set the related files path;

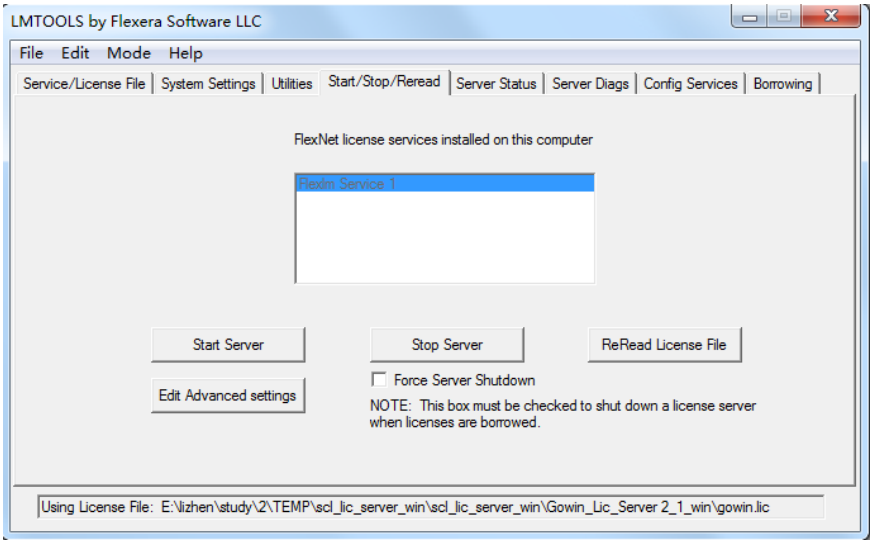
- “Path to the lmgrd.exe file”: lmgrd.exe file path;
- “Path to the license file”: license file path;
- “Path to the debug log file”: The path of the log file after license configuration. If there is no log file, you will need to create one before configuration.

Figure 3-4 SCL Configuration GUI



f) See Figure 3-5, start the license server after configuration. Select “Start/Stop/Reread”, and then click “Start Server” to view the log file and verify that the server has started successfully.

Figure 3-5 SCL License Start

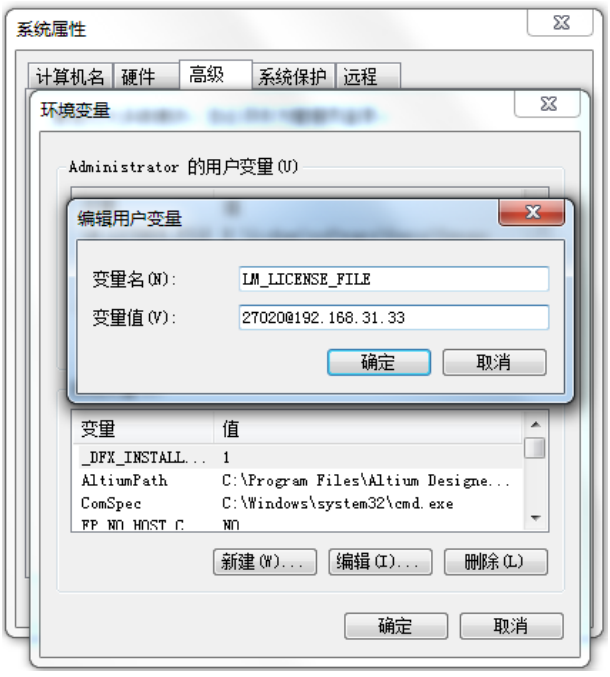


Note!

After starting server license, the system environment variable "LM_LICENSE_FILE" is required to be configured before the configuration of the SynplifyPro floating license. As shown in Figure 3-6:

- Environment Variable name: LM_LICENSE_FILE;
- Environment Variable value: License files location, such as: 27020@192.168.31.33 and “192.168.31.33” is this PC IP address.

Figure 3-6 Environment Variables Setting

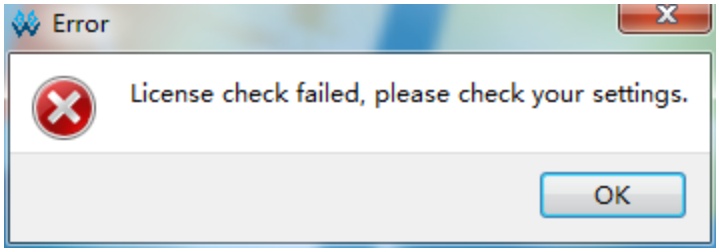


3.6 Place and Layout License Configuration

Gowin YunYuan software checks the license automatically when the user opens it. If the license is not correct, the software will not open, and an

error message will be displayed, as shown in Figure 3-7.

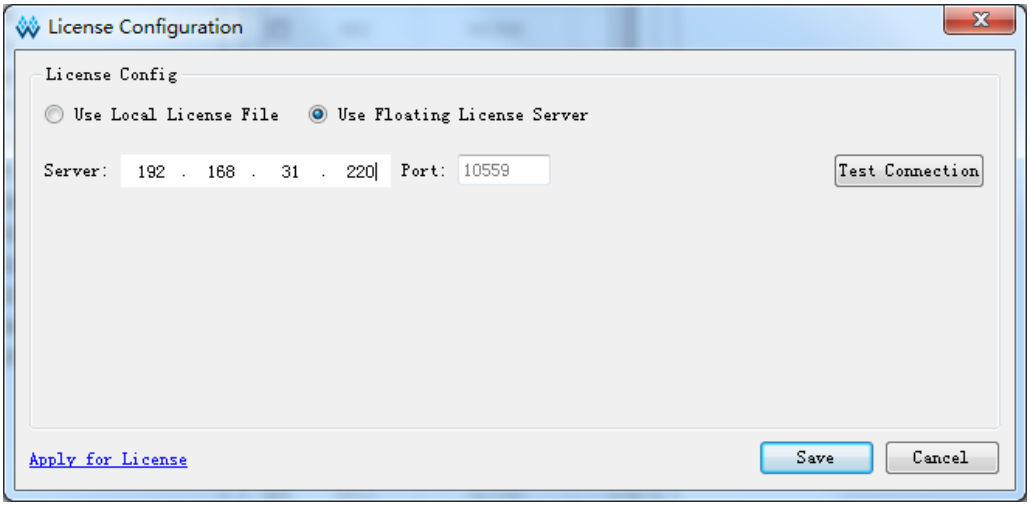
Figure 3-7 License Error



Users can configure the license after the YunYuan software has been installed. See the steps as below:

1. Open the Gowin YunYuan software;
2. In the menu bar, click “Help> Manage License…” to open the “License Configuration”, as shown in Figure 3-8.

Figure 3-8 Manage License options



3. User can choose local license file or floating license file.

   - Use Local License File
     Select “Browse…” to add the file path for node-locked License.

   Note!

   If there is no License file, apply the license by clicking “Apply for License” in the lower-left of the “License Configuration” View.

   - Use Floating License Server

   First, start floating license using PnrLicenseServer in Yunyuan software installation package file “GowinLicenseServerForWindows\PnrLicenseServer”. There are two ways to start PnrLicenseServer:

   a). Doubel click on “gwlic_server.exe”;

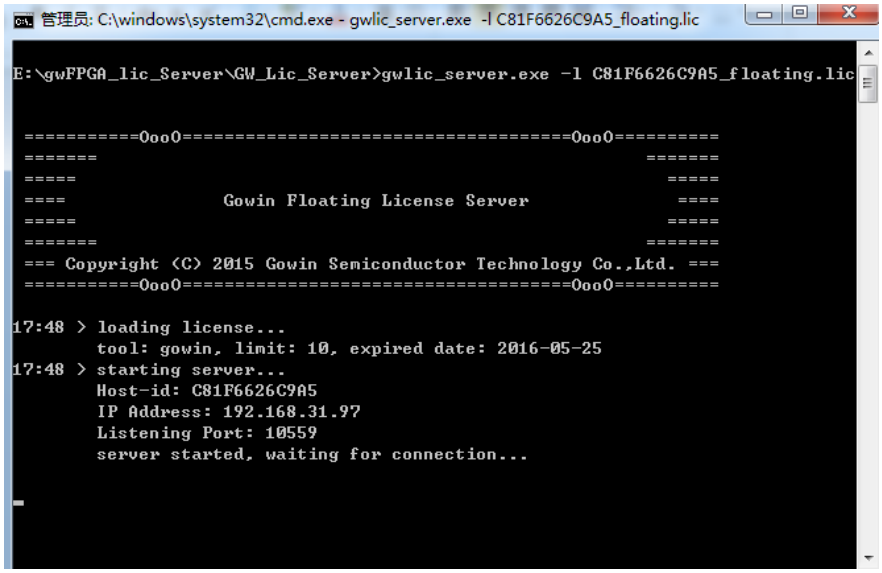
   Copy the floating license file (gowin_license.lic, for example) to

"GowinLicenseServerForWindows\PnrLicenseServer". Rename the floating license file as "gowin.lic". Double click "gwlic_server.exe".

b). Commands Mode

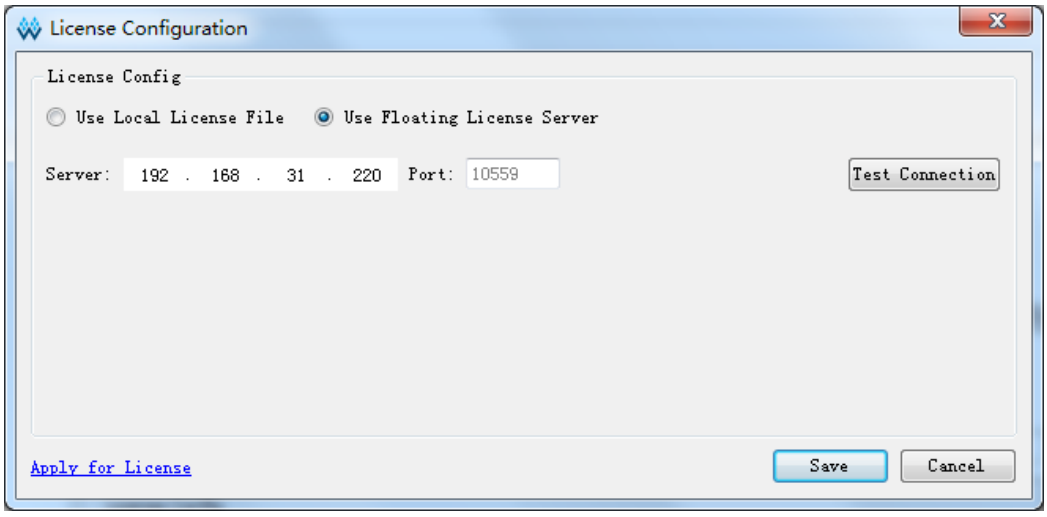
- Copy the floating license file (gowin_license.lic, for example) to "GowinLicenseServerForWindows\PnrLicenseServer". In the cmd window, execute the "gwlic_server.exe –l license.txt" command in the directory of gwlic_server.exe.
- Figure 3-9 shows the interface after the server has started.

Figure 3-9 Gowin Floating License Server Start Interface



- After successfully starting the server, input the server IP address in the "Licese Configuration" view, as shown in Figure 3-10.

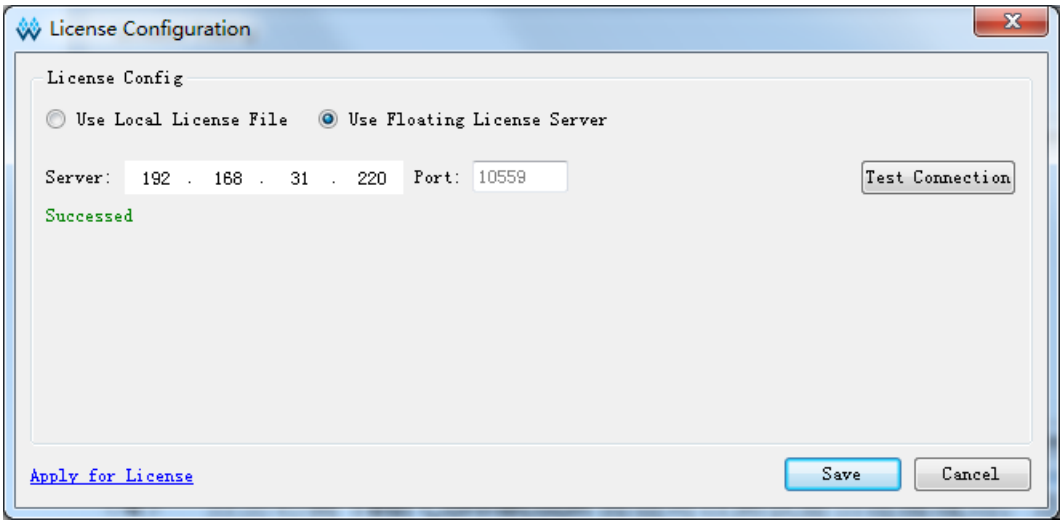
Figure 3-10 Floating License Configuration Interface



- Click the "Licese Configuration" button on the right to test whether the connection has been successful. If successful, "Successed" will appear in the "License Configuration" view,

as shown in Figure 3-11.

Figure 3-11 Test Connection



3.7 Software License Configuration (Linux)

3.7.1 SynplifyPro License Configuration

In the Linux version of the software, only floating licenses are available for SynplifyPro. Save the license.txt file to the lmgrd path. If it's saved to the other directory, absolute path needs to be used.

Take 64-bit Linux, Redhat 5, for instance, refer to the following steps to configure the license (only the floating license is supported):

1. In the “GowinLicenseServerForLinux\SCL\SynopsysInstaller” folder, install “SynopsysInstaller” according to the installation notes described in “installer_INSTALL_README.txt”:

   a). In the "SynopsysInstaller_v4.1.run" file path, modify the file permission, right-click to open the command line, and run

   “chmod 755 SynopsysInstaller_v4.1.run”.

   b). Switch to "root", execute the installation command in the "SynopsysInstaller_v4.1.run" path:

   “./SynopsysInstaller_v4.1.run -d /usr/synopsys/installer”.

   c). Exit "root". Make sure the installation path is available. Run the command as follows in the current path:

   “chmod -R 755 /usr/synopsys/installer”。

   d). Add Environment Variables:

   - If C shell is used, run the command as follows:“set path=(/usr/synopsys/installer $path)”
   - If Bourne or Bash shell is used, run the command as follows:“PATH=/usr/synopsys/installer:$PATH”“

export PATH”

e). Users can search the environment variables using the “export” command to confirm whether the path was added successfully.

2. Install the SCL according to the notes described in “scl_INSTALL_README.txt” in the “GowinLicenseServerForLinux\SCL\doc” path. The installation path is “path/ scl_lic_server_linux/v2018.06”, for example.

   a). If it is the suse system, delete the “scl_v2018.06_linux64.spf” file in the “GowinLicenseServerForLinux\SCL” path. If it is the other systems, such as CentOS, Redhat, UBUNTU, etc, delete the “scl_v2018.06_suse64.spf” file.

   b). After the installation of SynopsysInstaller is done, install the SCL software, run the command as follows:

   “installer -batch_installer -source <SOURCE> -target <DEST>”

   <SOURCE> is the spf file path. <DEST> is the SCL installation path. During installation, a window will appear to ask if the files are all installed correctly. Click "yes" to continue the installation.

3. Confirm and modify the license file:

   a). Modify PC name and confirm MAC address.
   In the floating license file, modify "hostname1" in “SERVER hostname1 000C293B1A2B 27020” to the PC name. "000C293B1A2B" is "HWaddr", and "27020" is the port number. Modify the license according to the PC on which SynplifyPro is installed, such as: “SERVER gaoyun 000C293B1A2B 27020”.

   b). Modify snpslmd path.
   On the basis of the "snpslmd.exe" path in SCL, modify this line “VENDOR snpslmd /path/to/snpslmd” in the license file to the local snpslmd path; for example: “VENDOR snpslmd path/scl_lic_server_linux/v2018.06/linux64/bin/snpslmd”.

4. Configure the License using SCL

   In the lmgrd path, execute the "./lmgrd -c license.txt -l lic.log" command to generate the log file "lic.log". View the lic.log to verify whether the configuration has been successful;

5. Configure the Environment Variable:

   a) Configure the "LM_LICENSE_FILE" environment variable. The value format should be "Port number @ Host name", for example: setenv LM_LICENSE_FILE 27020@gaoyun

   b) The default port number is 27020 during License installation. It can be modified by finding the character string "27020" in the license file and replacing this string with the designated port.

Note!

All the commands should be executed in the path of the license installation directory. Furhter license commands are as follows:

- Configure license——“./lmgrd –c licensefile –l lic.log”；

- Stop license——"./lmdown -c licensefile"；
- Check license status——"./lmstat -a -c port number@ host name".

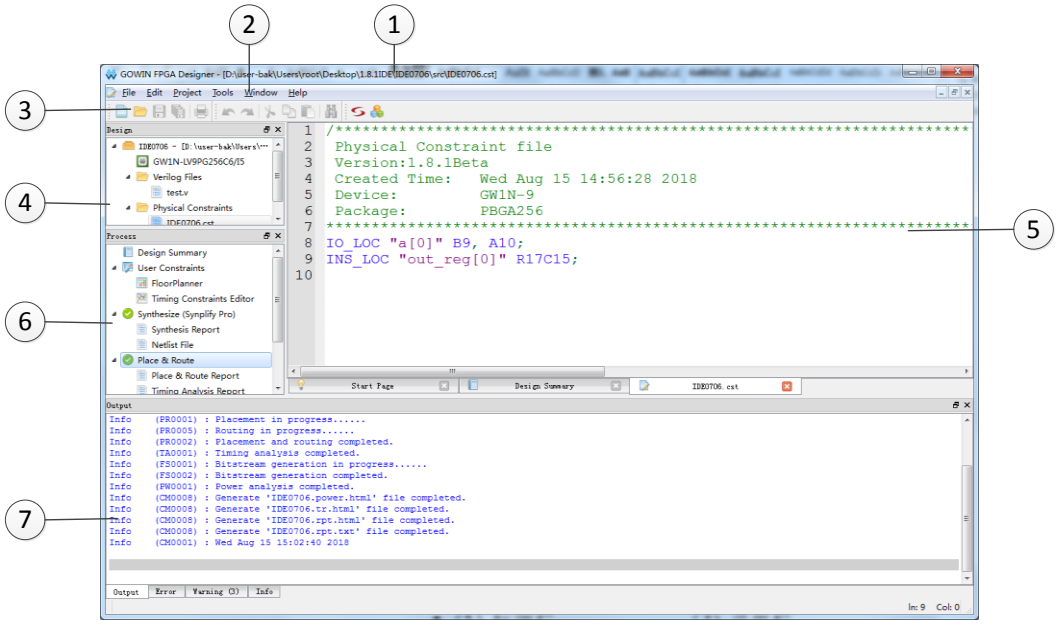
3.7.2 Place and Layout License Configuration

In Linux, the Gowin License configuration is the same as that of Windows, please refer to 3.6Place and Layout License Configuration.

4 Gowin Software GUI

Figure 4-1 shows the YunYuan software GUI. It consists of the title bar, menu bar, tool bar, project area (Design View), process area (Process View), source file editing area, and information output area (Output).

Figure 4-1 Gowin Software GUI



① Title Bar
② Menu
③ Tool Bar
④ Project Area
⑤ Source File Editing Area
⑥ Process Area
⑦ Information Output area

4.1 Title Bar

Title bar shows the current project path, name, and the name of the file that is currently open.

4.2 Menu

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:

1. File:
   - New: Used to create a new file and project;
   - Open: Used to open a new file and project;
   - Save: Used to "save" or "save as" the project files;
   - Close: Used to close project, project files, and page;
   - Print: Print;
   - Recent Files: Shows the files that were recently open. The user can click on the names of these files to reopen them;
   - Recent Projects: Shows the names of the projects that were recently open. The user can click on the names of these projects to reopen them;
   - Exit: Used to exit and close the Gowin YunYuan software.
2. Edit:
   - Undo: Used to undo your last operation;
   - Redo: Used to redo the last operation;
   - Cut: Used to cut the characters;
   - Copy: Used to copy the characters;
   - Paste: Used to paste the characters;
   - Select All
   - Find: Used to find or change key words;
   - Macros
3. Project
   - New File: Used to create a new file;
   - Add File: Used to add a file;
   - Clean: Used to clean all the generated files and file folders in the impl folder after running the project;
   - Set Device: Used to change the project device, package, and speed;
   - Configuration: Configuration for running the project;
4. Tools
   - Start Page
   - Synplify Pro: Front-end integration software;
   - Simulation

- IP Core Generator
- FloorPlanner: Physical constraints editor;
- Timing Constraints Editor;
- Gowin Analyzer Oscilloscope;
- Options: Used to open the required tools or set IDE parameters.

5. Window
   - Full Screen;
   - Tile;
   - Cascade;
   - Reset Layout: Restores initial settings;
   - Panels: Used to select whether to display GUI models or not;
   - Start Page;
   - Design Summary.
6. Help:
   - Contact Us: Click to contact us;
   - Manage License;
   - About: Shows software version and copyright information.

Note!

"Project" shows in the menu bar only after you create a new project.

4.3 Tool Bar

The Tool bar contains quick access buttons. These are as follows (from left to right): New File or Project, Open File or Project, Save, Save All, Print, Undo, Redo, Cut, Copy, Paste, and Find.

4.4 Project Area (Design)

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

4.5 Process Area (Process)

The process area provides FPGA design flow, including synthesis, place & route, and download bitstream files (program device). Users can also double click timing constraints editor and physical constraints editor to edit the constraints files.

4.6 Source File Editing Area

Users can view and edit source files in the source file editing area.

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

If the file currently displayed is modified, the term “File Changed” will appear in the file editing area, as shown in Figure 4-2. Select “Reload” to reload the file.

Figure 4-2 File Changed

To close the file currently displayed, click "File > Close", or click on the icon " " that appears in the upper-right of the file editing area.

To close all the files in the file editing area, click "File > Close All".

4.7 Information Output area

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

- Output: All information
- Error
- Warning
- Info: All other information

In the output tab, right-click and select "Clear" to clear all the information in the tab. Users can only clear the information that is displayed in the current tab if they right-click and select "Clear" in the Error tab, Warning tab, or Info tab.

If a Warning or an Error appears during running, the number of Warnings and Errors will be recorded and shown on each of the corresponding tabs.

5 Operation

Gowin YunYuan is available in Windows and Linux. It supports GUI running mode and commands running mode. Take the GUI running mode in Windows for instance. Please refer to the following for the operation details.

For the command lines used for the commands mode, please refer to *Appendix B Command Line Options*.

5.1 Create a New Project

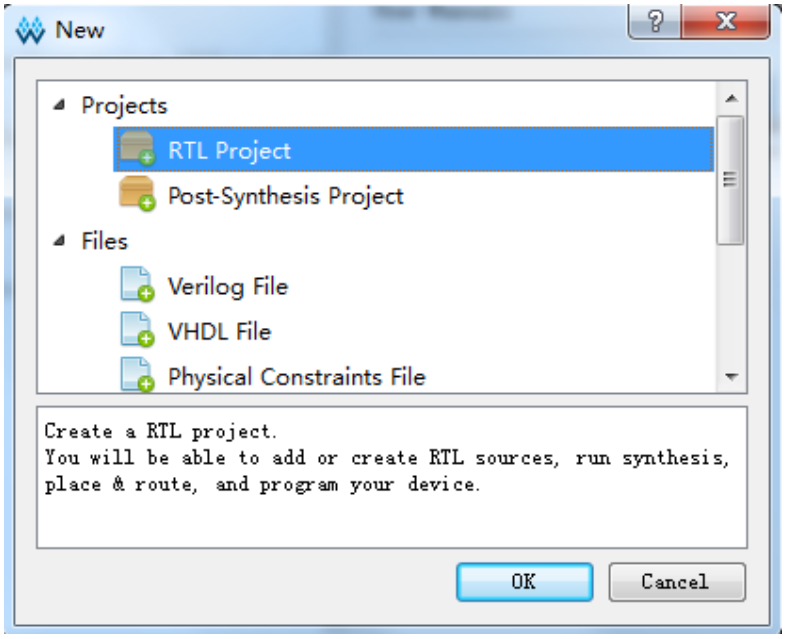
YunYuan software supports two kinds of projects: RTL and Post-Synthesis projects.

1. RTL projects employ user RTL design as input, and the entire flow includes synthesis, place&route, and downloading bitstream files.
2. Post-Synthesis projects use the synthesized logic netlist as input, so synthesis can be skipped, and place&route and download bitstream files can be implemented directly.

5.1.1 Create a RTL Project

1. From the File menu, choose “File> New…” to open the “New” dialog, as shown in Figure 5-1.

Figure 5-1 Create a New Project



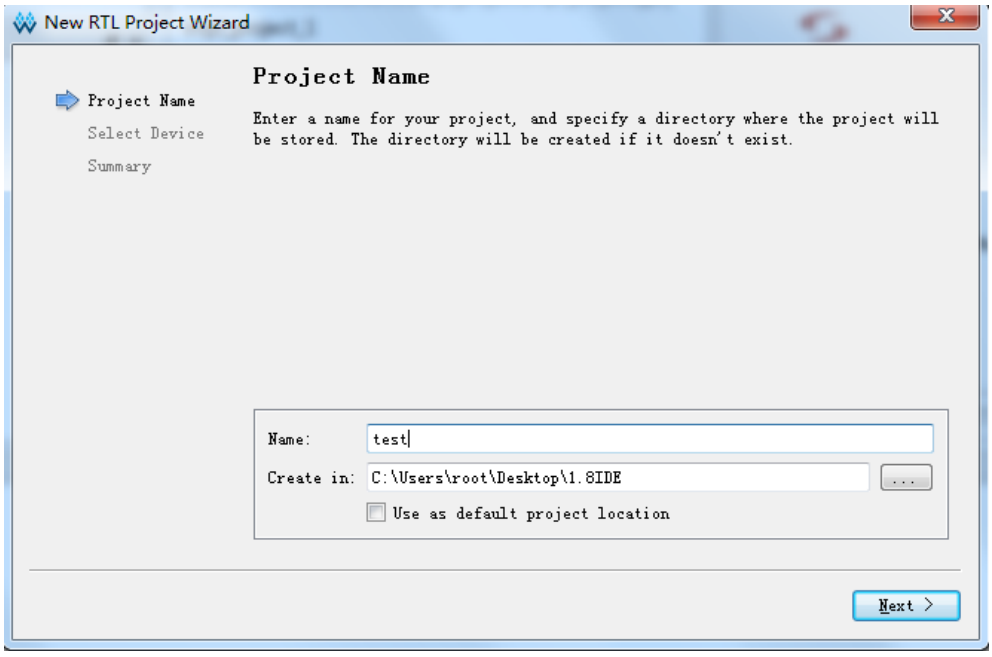
Note!

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

- Using the "Ctrl+N" short cut;
- By clicking on the “New File or Project” icon that appears on the tool-bar;
- By selecting “Quick Start>New Project” on the Start Page.

2. Select “RTL Project”, and then click “OK” to open “New RTL Project Wizard”, as shown in Figure 5-2.

Figure 5-2 Create RTL Project Wizard



3. Create the project "Name" and "Create in", as shown in Figure 5-2;

a). Enter the project name in the "Name" field;

b). Click on the “...” icon to choose the project path.

Note!

- No Chinese characters or spaces can be included in the project path;
- The file path length is limited in both Windows and Linux. You cannot delete or copy the files with the length going over the limits；
- The path separator is “\” in Windows; for example, E:\Gowin\ide;
- If users select “Use as default project location”, the project location will be set as the default location, and all later projects you create will be saved to this location.

4. Click "Next" to select the device, as shown in Figure 5-3:

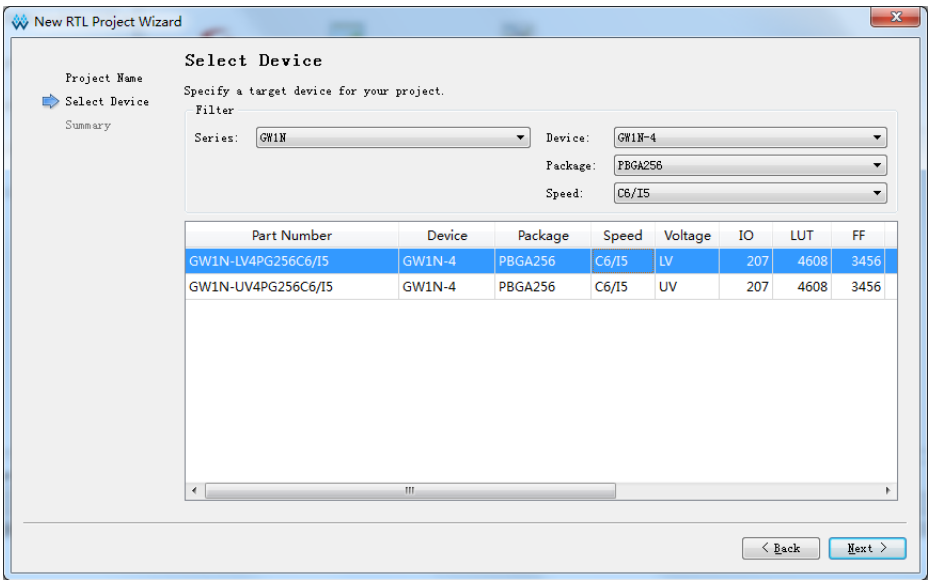
   a). In this step, users can select the target device series, package, and speed.

   b). Choose package type from the “Package” drop-down list;

   c). Choose speed grade from the “Speed” drop-down list;

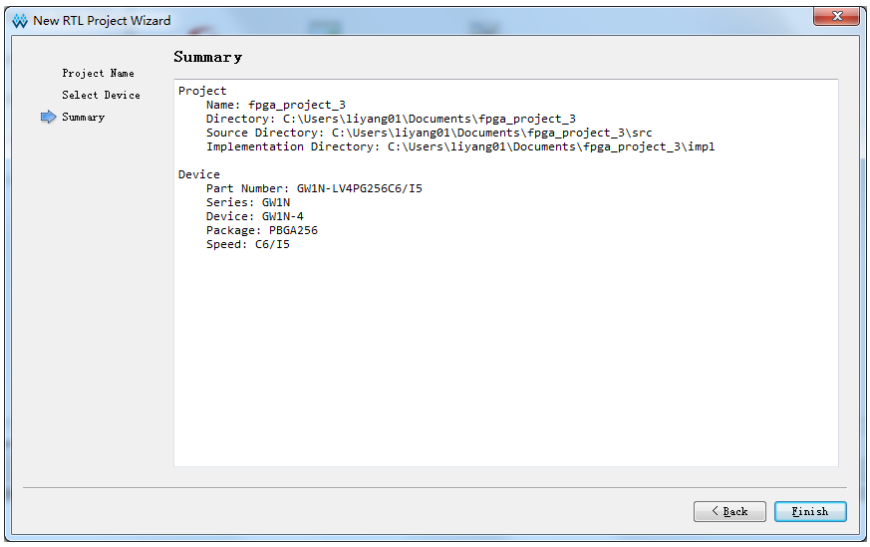
   d). Choose the detailed part number from the "Part Number" sub-window. It displays the detailed resource information related to the selected device.

Figure 5-3 Select Device



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

Figure 5-4 Project Information Summary



6. Click "Finish". The project now is created.

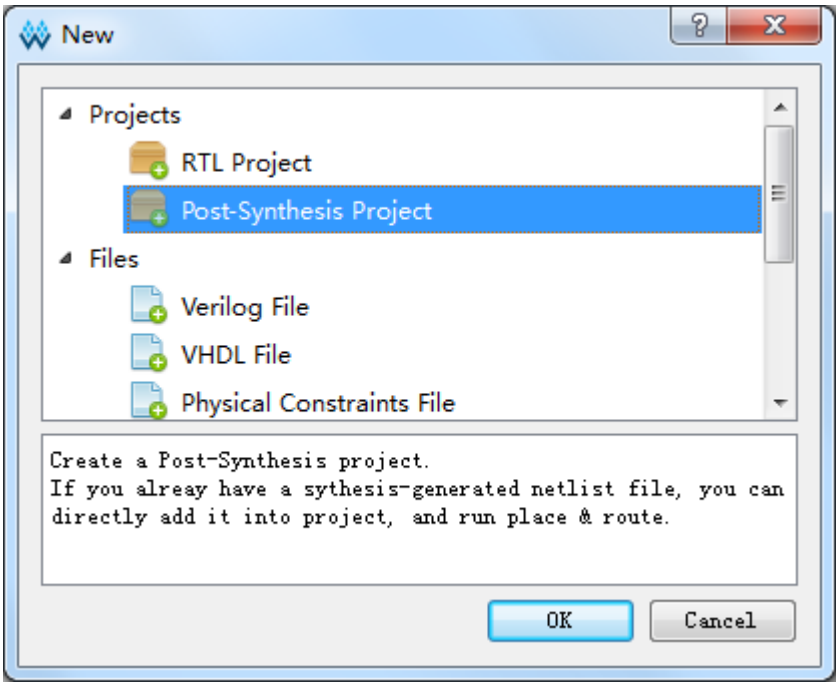
Note!

- The device can be changed after the project has been created. Please refer to 5.3Edit a Project>5.3.1Modify Project Device for further details.
- Users can add source files and constraints files after the project has been created. Please refer to 5.3Edit a Project>5.3.2Edit a Project File for the details.

5.1.2 Create a Post-Synthesis Project

1. From the File menu, choose “File> New…” to open the “New”dialog, as shown in Figure 5-5.

Figure 5-5 Create a New Project



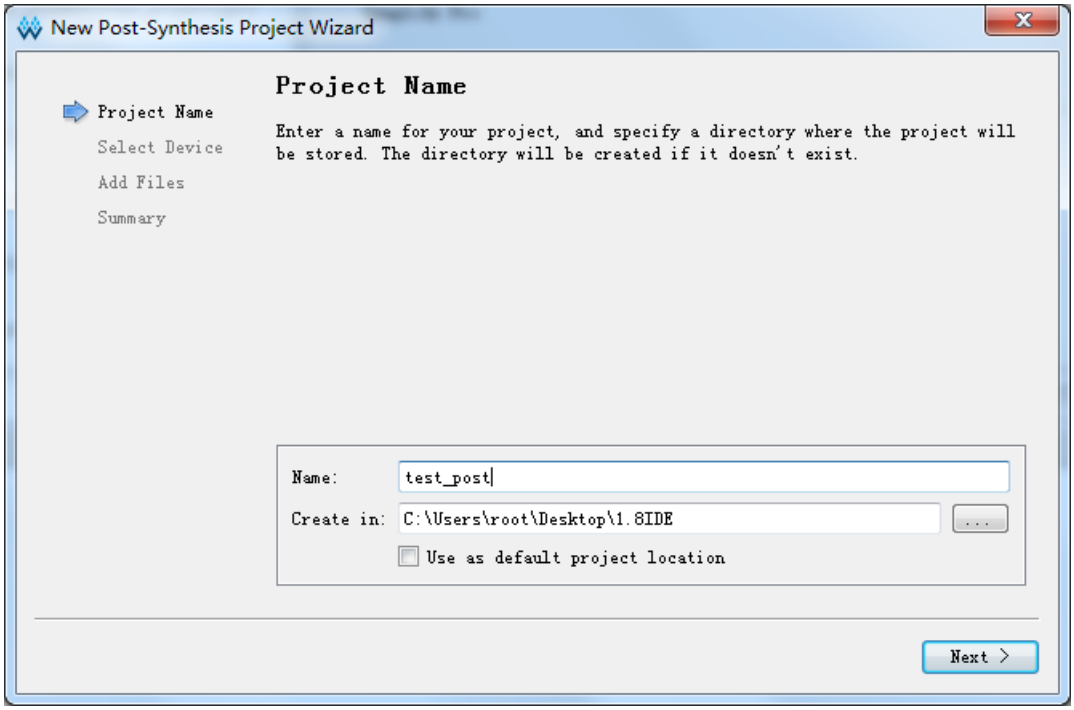
Note!

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

- Using the "Ctrl+N" short cut;
- Clicking on the “New File or Project” icon in the toolbar;
- By selecting “Quick Start>New Project” on the Start Page.

2. Select “Projects>Post-Synthesis Project”, and then click “OK” to open “New Post-Synthesis Project Wizard”, as shown in Figure 5-6.

Figure 5-6 Create a Post-Synthesis Project



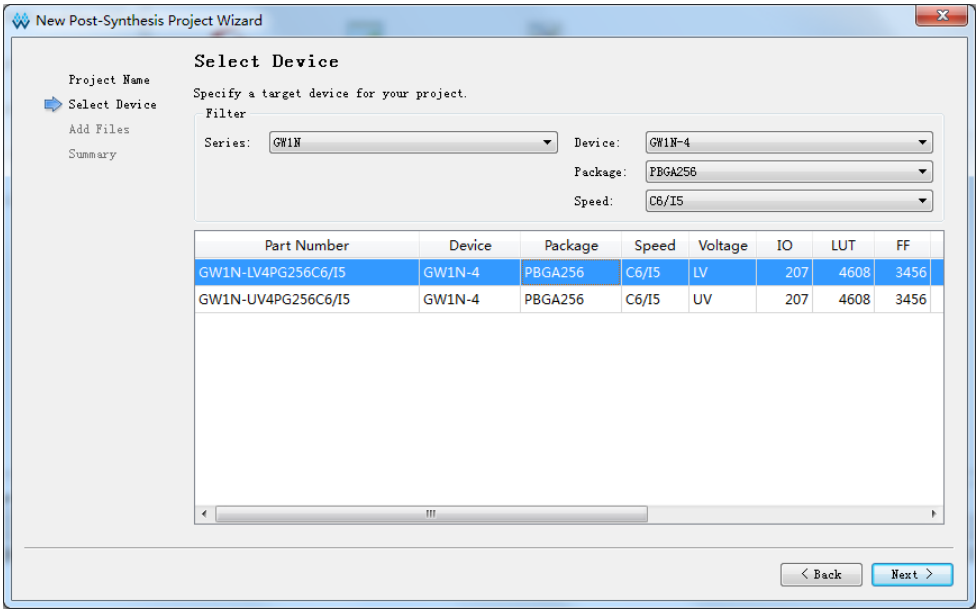
3. Create the project "Name" and "Create in", as shown in Figure 5-6;

   a). Enter the project name in the "Name" field;

   b). Click on the “...” icon to choose the project path.

4. Click "Next" to select the device, as shown in Figure 5-7: In this step, users can select the target device series, package, and speed. Choose the detailed part number from the "Part Number" sub-window. It displays the detailed resource information related to the selected device.

Figure 5-7 Select Device

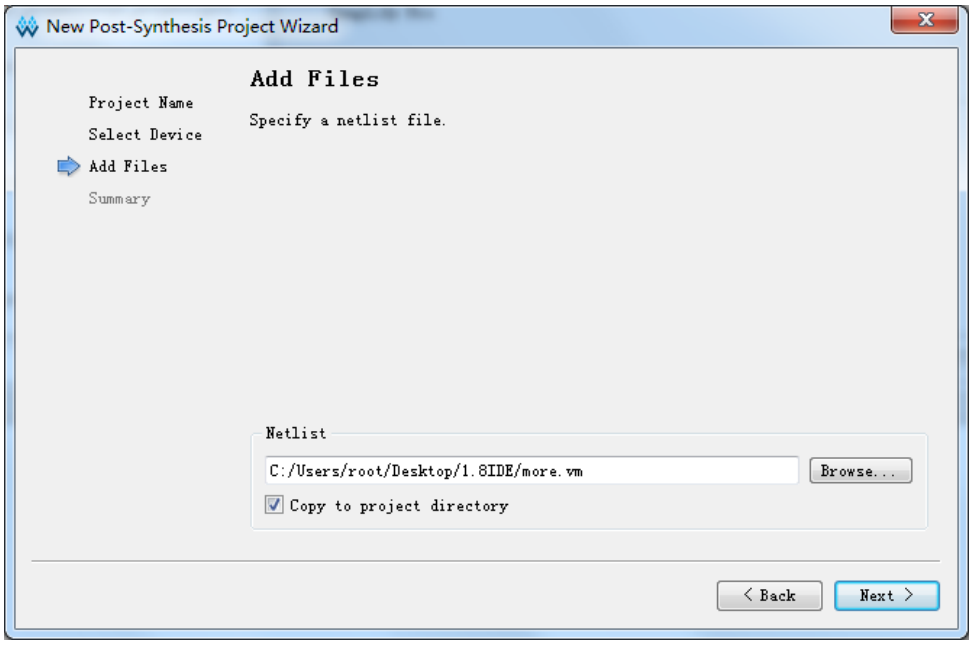


5. Click "Next" to add the Netlist files with the name suffix .v, .vm, .vo, .sv, .vma, and .vp, as shown in Figure 5-8.

   Note!

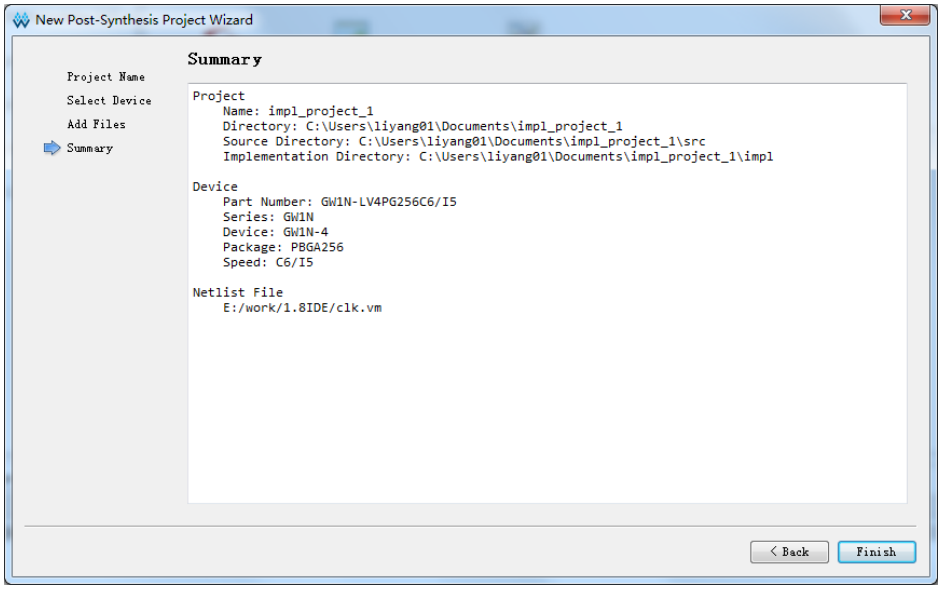
   If users select “Copy to project directory”, the netlist files will be copied to the project location.

Figure 5-8 Add Files



6. Click "Next" to open the Project Information Summary window, as shown in Figure 5-9. This includes the new post-synthesis project information, device information, and source files information.

Figure 5-9 Project Information Summary



7. Click "Finish". The project now is created.

Note!

- The device can be changed after the project has been created. Please refer to 5.3Edit a Project>5.3.1Modify Project Device for further details.
- Users can add source files and constraints files after the project has been created. Please refer to 5.3Edit a Project>5.3.2Edit a Project File for the details.

5.2 Open an Existing Project

Use one of the following methods to open an existing project:

Method 1

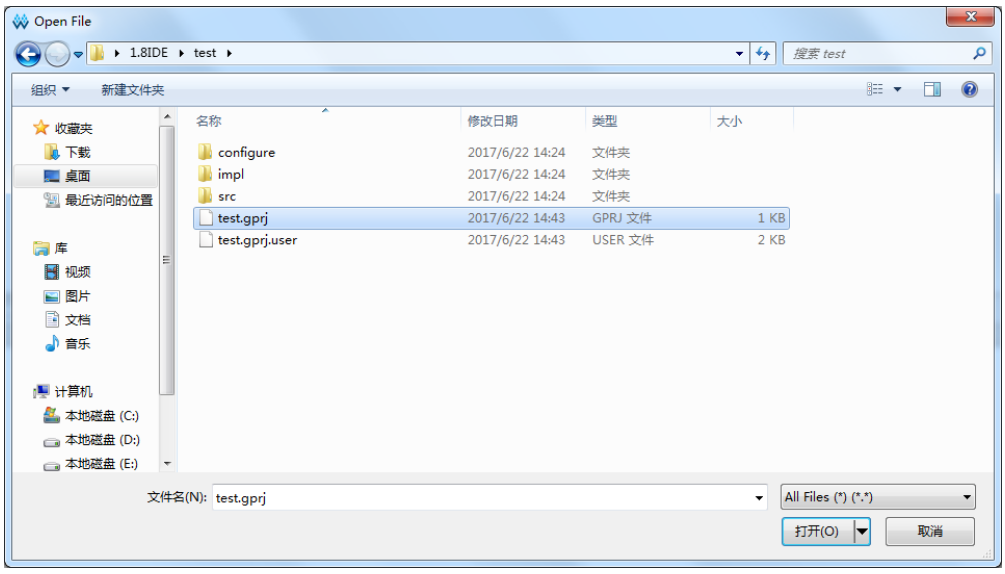
1. From the File menu, select “File> Open …” to open the “Open File” dialog box, as shown in Figure 5-10.

Note!

Users can also click on the “ ” icon in the tool bar to open the “Open File” dialog box.

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

Figure 5-10Open an Existing Project



Method 2

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

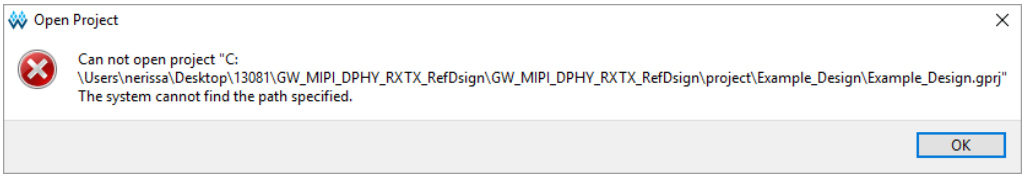
Method 3

From the File menu, click “File> Recent Projects”, to open your required project.

Note!

- Users can also open recent projects from the projects list that is displayed on the left side of the start page.
- Recent Projects shows the recently opened projects. Users can click on the names of the files to re-open them;
- If the project was deleted, the “Open Project” dialog box will appear, as shown in Figure 5-11.

Figure 5-11 Open Project



Method 4

Find the project you established and find the *.gprj file. Doule-click on the *.gprj file to open the project with Gowin YunYuan software automatically.

5.3 Edit a Project

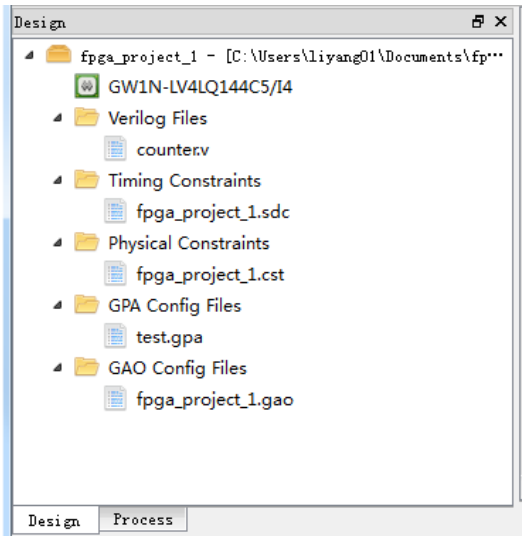
After creating or opening a project, users can edit the device information and the related files in the project design area, as shown in

Figure 5-12.

The Project Design Area contains the following:

1. The project path;
2. Chip info: Chip type, package type, and speed;
3. The current project files, including user design files, physical constraints files (.cst.), timing constraints files (.sdc), GAO config files (.gao), and GPA config files (.gpa), etc.

Figure 5-12 Project Design Area



5.3.1 Modify Project Device

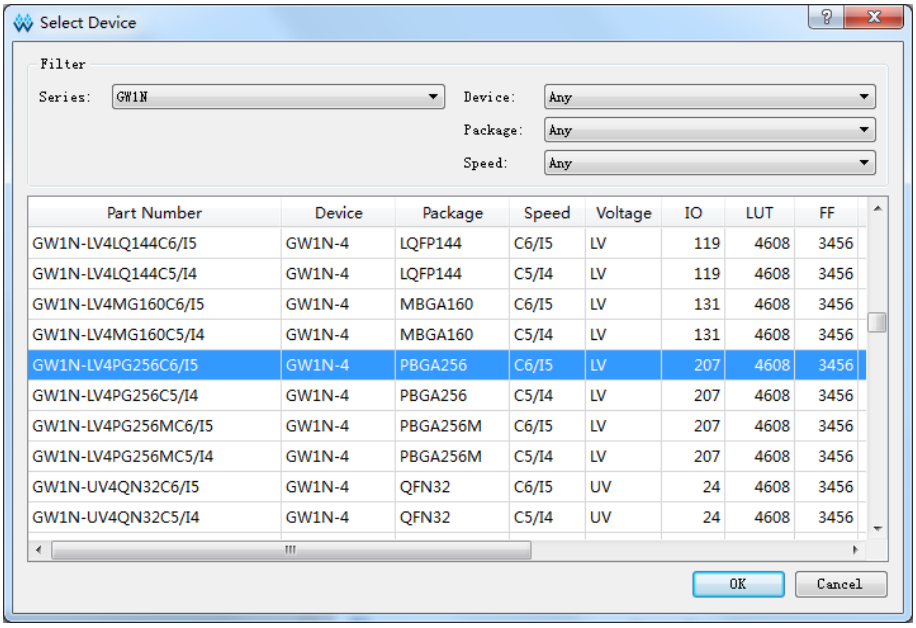
The second line in the project design area shows the device info. for the current project. Refer to the steps as below to modify the current project device.

1. As shown in Figure 5-12, double-click “GW1N-LV4LQ144C5” to open the “Select Device” view, as shown in Figure 5-13;
2. In the “Select Device”view, select “Series” and “Device”, select “Package” from the package drop-down list and select “Speed” from the speed drop-down list, and then select the detailed part number from the “Port Number” sub-window. Click "OK".

Note!

The “Port Number” sub-window displays the detailed resource information related to the selected device.

Figure 5-13 Project Device Information



5.3.2 Edit a Project File

RTL projects need RTL design files (Source Files), constraints files, and configuration files. Constraints files contain the Physical Constraints File and Timing Constraints File; configuration files contain the GAO Config File and GPA Config File.

An RTL project can have multiple design files; however, only one physical constraints file and one timing constraints file can be active at a time.

The user netlist file is added to the Post-Synthesis project when it is created; as such, users only need to add the corresponding constraints files and configuration files after the project has been created.

Take an RTL project, for instance. Refer to the following sections to edit the project files.

1. Create a New Project File

   a). As shown in Figure 5-14, right-click on a blank area of the project design area, select “New Files…” to open "New" dialog box, as shown in Figure 5-15;

   b). As shown in Figure 5-15, select the file type and click "OK" to create a new file.

Figure 5-14 Right-click Menu

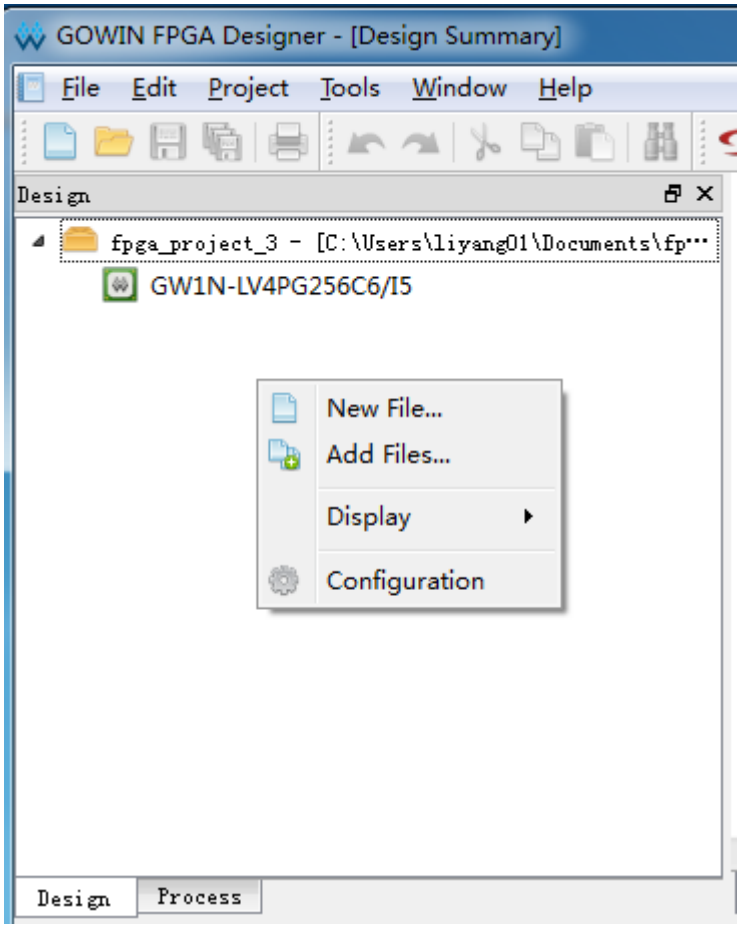
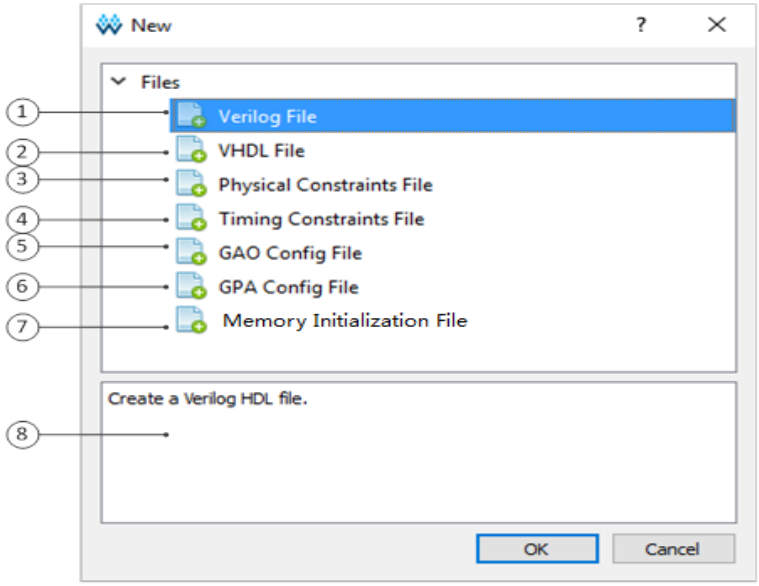


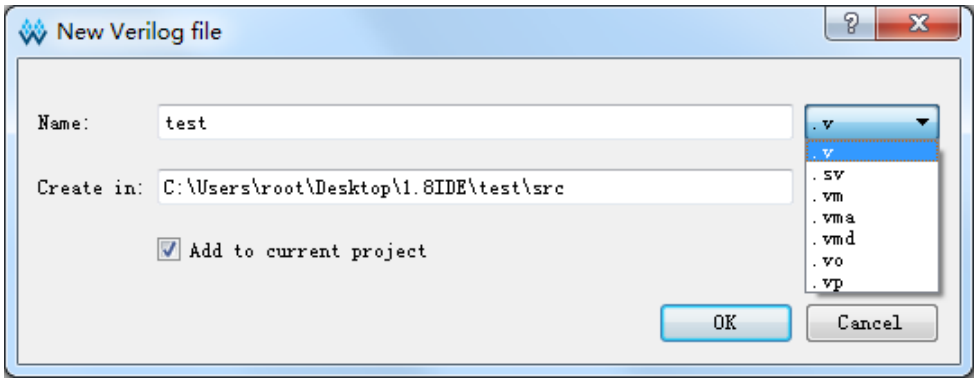
Figure 5-15 Create a New File



① Verilog File
② VHDL File
③ Physical Constraints File
④ Timing Constraints File
⑤ GAO Config File
⑥ GPA Config File
⑦ Memory Initialization File
⑧ File Description

c). Take creating a Verilog File, for instance. Select “Verilog File” and click "OK" to open the Verilog File view, as shown in Figure 5-16.

Figure 5-16 Create a Verilog File



d). Enter the file name and click "OK".

Note!

- Users can select file extensions from the drop-down list. “Add to current project” is selected by default.
- User can open and edit the newly created blank file in the source file editing area.

2. Create a Configuration File

a) As shown in Figure 5-14, right-click on a blank area of the project design area, select “New Files…” to open the "New" dialog box, as shown in Figure 5-15;

b) As shown in Figure 5-15, select the file type and click "OK" to create a new file. Take creating a GPA Config File, for instance. Select “GPA Config File” and click "OK" to open the GPA Config File view, as shown in Figure 5-17.

c) The newly created Config File will not be directly opened in the source file editing area directly. Users need to double-click the Config File in the project Design area to open and edit the blank Config File, as shown in Figure 5-18.

Figure 5--17 Create Config File

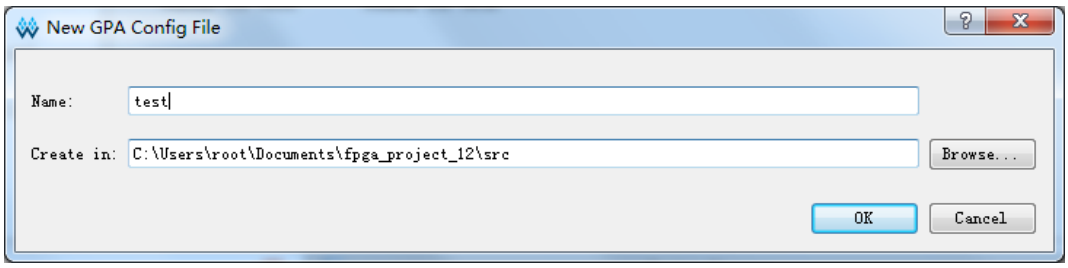
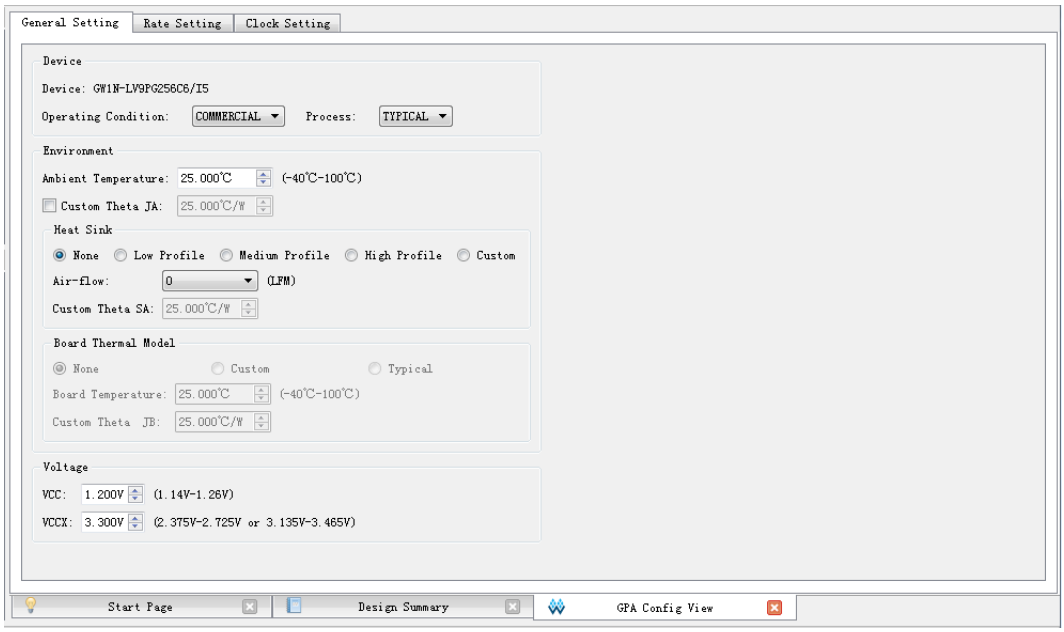


Figure 5-18 Config Files Setting



Note!

- If the newly added Verilog File has the same name as an existing file, the same name prompt will appear, as shown in Figure 5-19;
- If the newly added constraints files already exist, Figure 5-20 will pop up when the user selects “Add to current project”. Files cannot be added; however, new blank files will still be opened in the source files editing area; If “Add to current project” is not selected, the prompt will not pop up;
- If the newly added Config files already exist, a warning message will be displayed, as per Figure 5-21.

Figure 5-19 Same Name Prompt

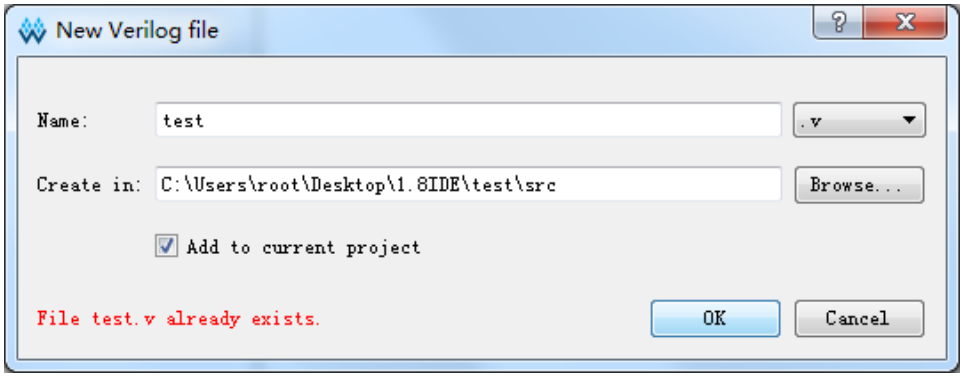


Figure 5-20 Constraints Files Existence

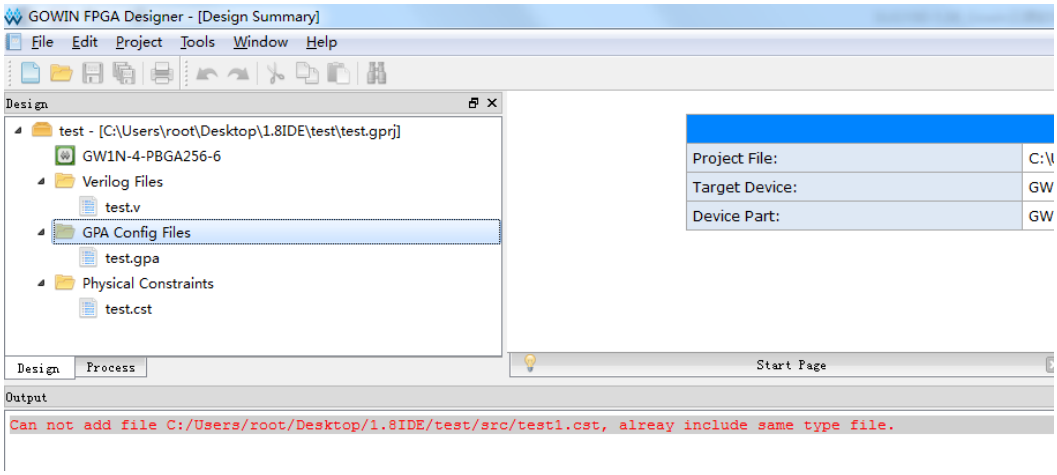
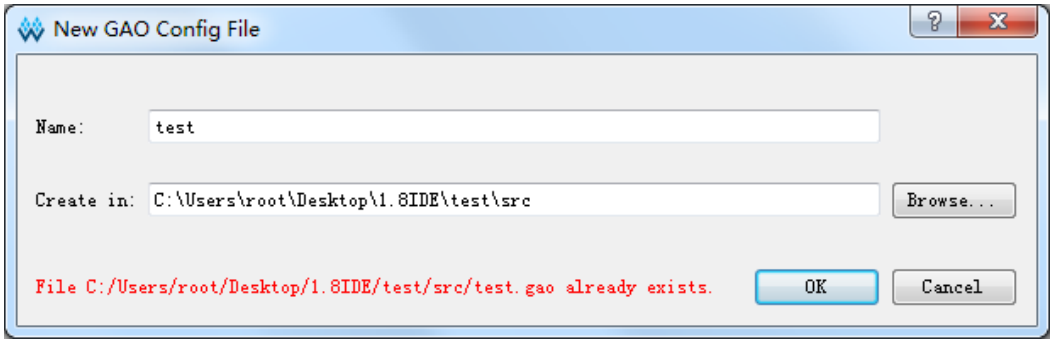


Figure 5-21 Config Files Existence



3. Add Project Files

   a) As shown in Figure 5-22, right-click in the blank of the project design area, select “Add Files…” to open the "Select Files" dialog box;

   b). Select single or multiple project files to add.

Note!

- If the added files are not the project files, Figure 5-23 will pop up to confirm whether the user needs to copy them into the project source directory.
- If the users add RTL files and constraints files at the same time, YunYuan will automatically classify the files in the project design area;
- From the menu bar, click "Project> Add Files…" to add project files.

Figure 5-22 Right-click Actions in Design View

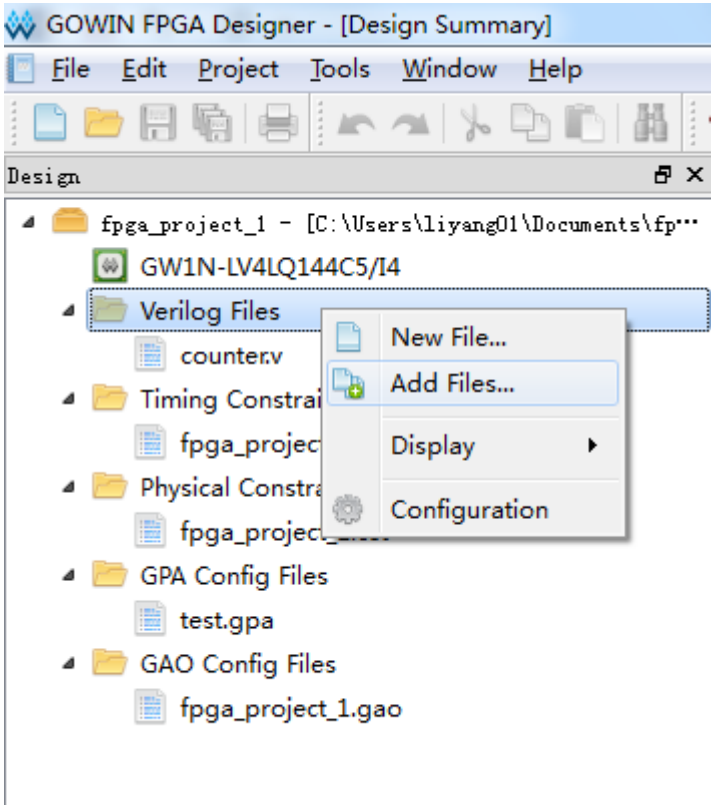
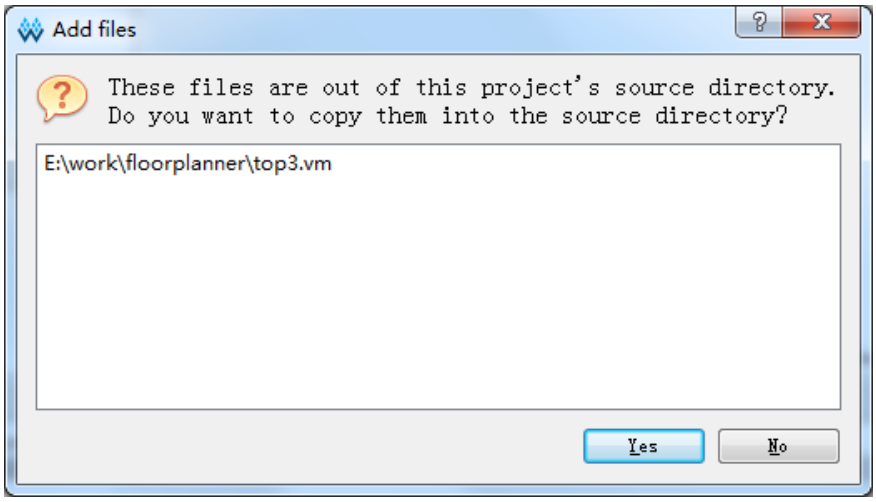


Figure 5-23 Copy File



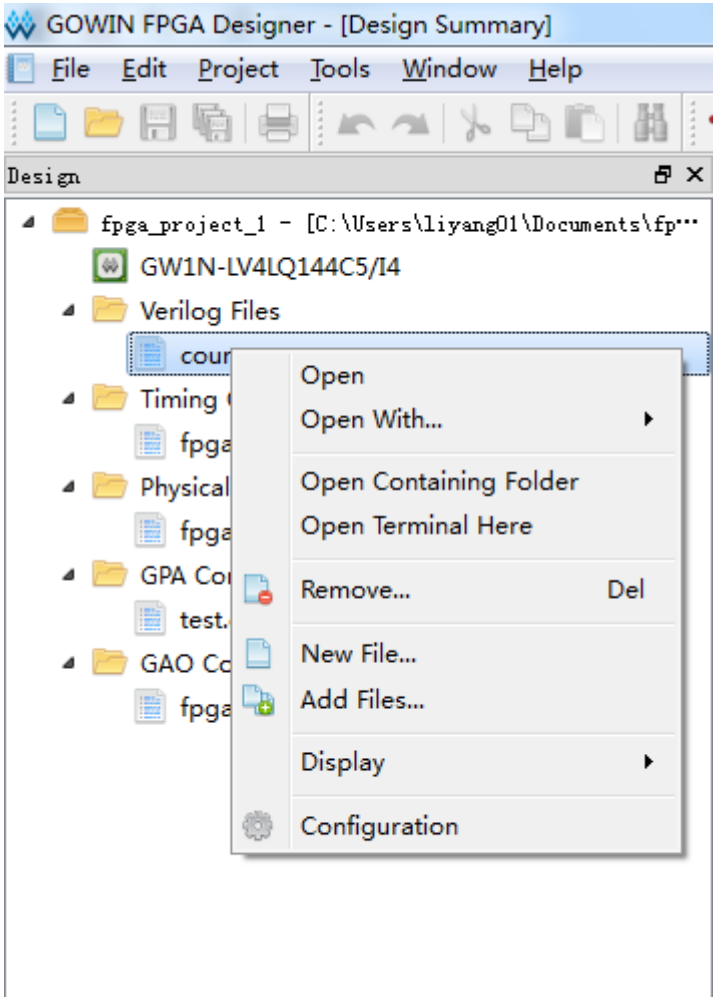
4. Modify Project Files

Use the following two methods to open and modify the project files, as shown in Figure 5-24:

a) Double-click any file in the project design area; the file will open in the source file editing area;

b) Right-click on the name of the file that is to be modified and select "Open".

Figure 5-24 Project Files Editing Actions



Note!

- Users can also select “Open With>Add External Editor” to add an external editor, as shown in Figure 5-25. Users can add external editors according to their needs.
- Users can select“Open Containing Folder”to open the file folder.
- Users can select“Open Terminal Here”to open the command line window. Users can select to run in command line mode.
- If users modify and save a file that has been opened in YunYuan software, the Gowin YunYuan software wiill generate a change notice, as shown in Figure 5-26.
- If users modify and close unsaved files, they will be prompted to save the change as shown in Figure 5-27.

Figure 5-25 External Editor

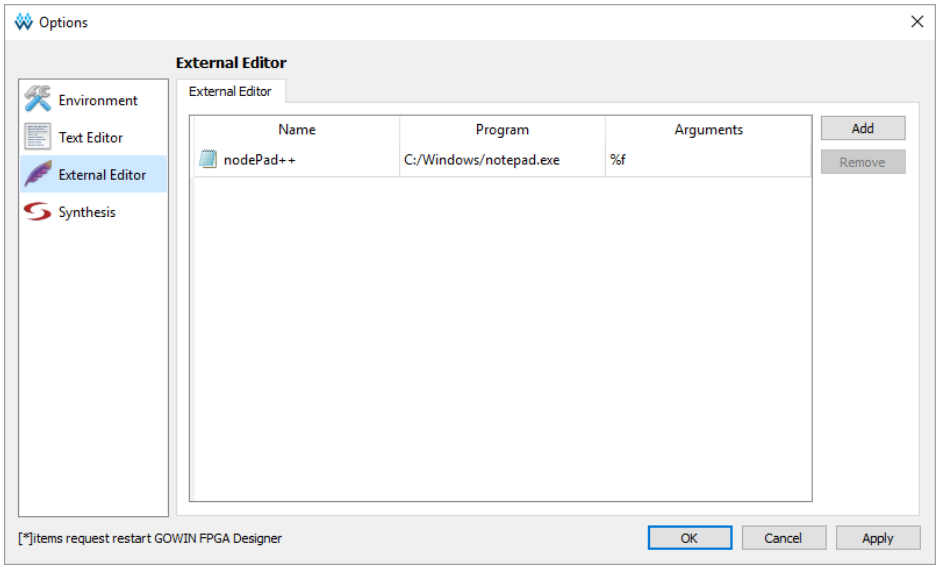
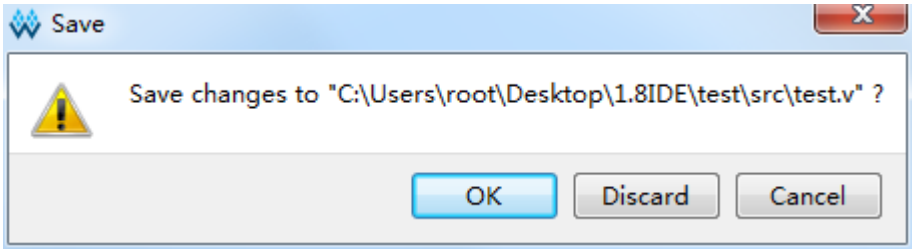


Figure 5-26 Change Project File

Figure 5-27 Save Project File

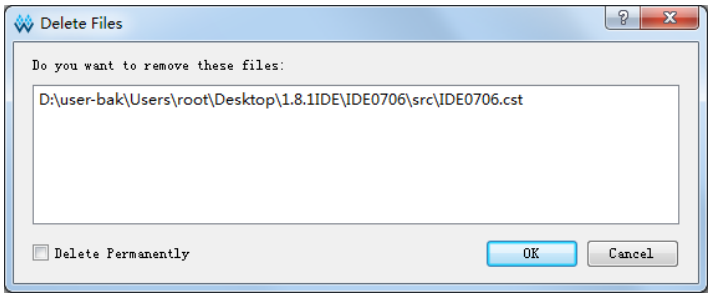


5. Delete Project Files

   a) Select the file in the project design area;

   b) Right-click and select “Remove” or press “Delete” on the keyboard. “Delete Files” will be displayed, as shown in Figure 5-28. If the user selects “Delete Permanently”, the file will be deleted from the current project and the disk. If “Delete Permanently” is not selected, the file will only be deleted from the current project.

Figure 5-28 Confirm File Deletion



Note!

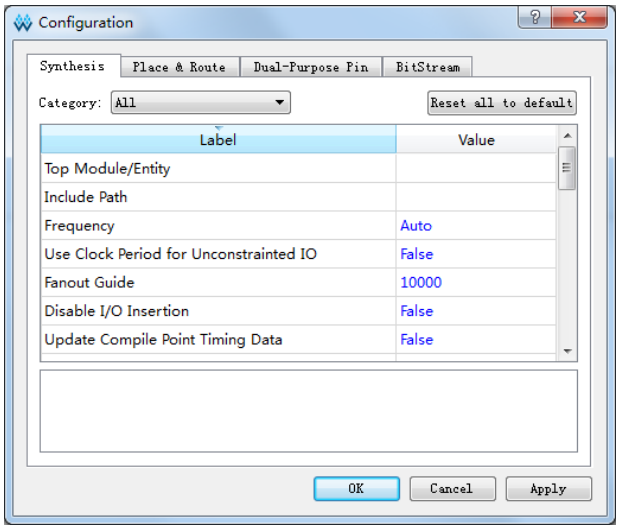
If the file that is open in the editing area is deleted, the delete file notice will pop up, as shown in Figure 5-29.

Figure 5-29Delete File Notice

5.3.3 Modify Project Configuration

1. Right-click on the Project Design area;
2. Select “Configuration” or select "Project->Configuration...” from the menu bar to open the project configuration view, as shown in Figure 5-30.

Figure 5-30Project Configuration View



As shown in Figure 5-30, the project configuration contains the Synthesis Configuration, Dual-Purpose Pin Configuration, Place&Route Configuration, Compile Configuration, and Bitstream Configuration tabs:

- Synthesis: Used to configure the parameters for optimizing user design with Synplify;

- Place&Route: Used to configure parameters for placing and routing;
- Dual-Purpose Pin: Used to configure duplicated pins;
- Bitstream: Used to configure download speed and enable CRC check, compress, etc.

See the information presented below for further details.

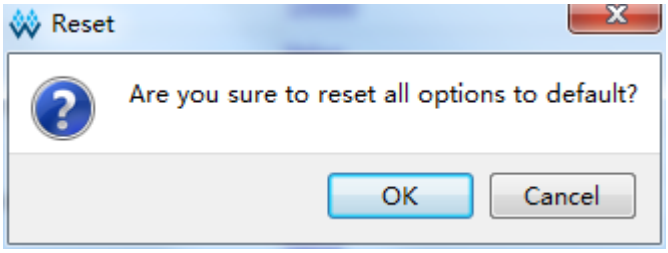
1. Synthesis

- In the configuration view, select Synthesis, and then select the parameters options from the "Category" drop-down list. The default value is "All";
- Select the parameter that needs to be configured in "Lable". The corresponding description shows at the bottom of "Configuration" view;
- Double-click the corresponding value and configure based on your requirements. Click "Apply" to put the current configuration into effect. Click "OK" to complete all the configuration.

Note!

- For further details about how to configure common parameters, please refer to Appendix A SynplifyPro Attributes and Directives;
- For further details about the configuration methods, please refer to the Synplifypro manuals in the YunYuan installation directory: installPatj\SynplifyPro\doc;
- For the Post-synthesis, there are no Synthesis options because the design file has been synthesized;
- Reset all to default: Reset all configuration on this page to default values; "Reset" will pop up when you click it, as shown in Figure 5-31.

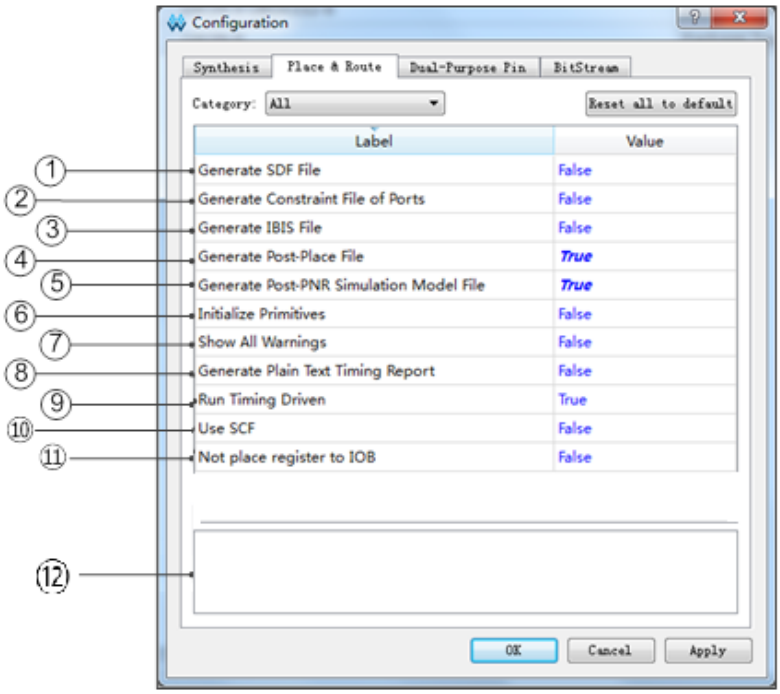
Figure 5-31 Reset



2. Place & Route

Place & Route configuration, which was independently designed by Gowin, is compatible with Gowin YunYuan software. Users can modify the value configuration, as shown in Figure 5-32.

Figure 5-32 Place&Route Configuration



① Generate SDF file
② Generate Constraints Files of Ports
③ Generate IBIS Files
④ Generate Post-Place file
⑤ Generate Post-PNR Simulation Model file
⑥ Initialize Primitives
⑦ Show all Warnings
⑧ Generate Plain Text Timing Report
⑨ Run Timing Driven
⑩ Use SCF
⑪ Not Place Register to IOB
⑫ Attributes Description

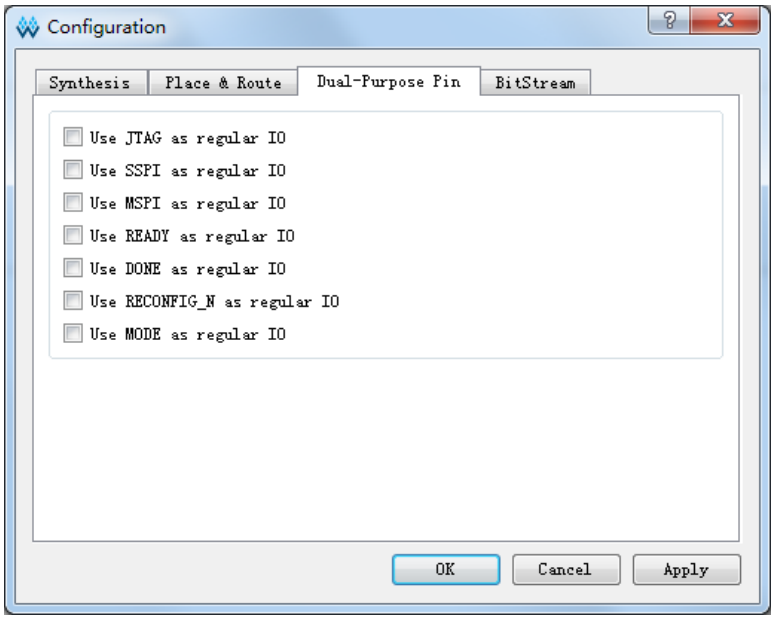
Note!

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

3. Dual-Purpose Pin

In the Dual-Purpose Pin tab, users can configure the multiplexing pins in different modes for the selected device; see Figure 5-33 for the detailed configuration options.

Figure 5-33 Configure Multiplexing Pins



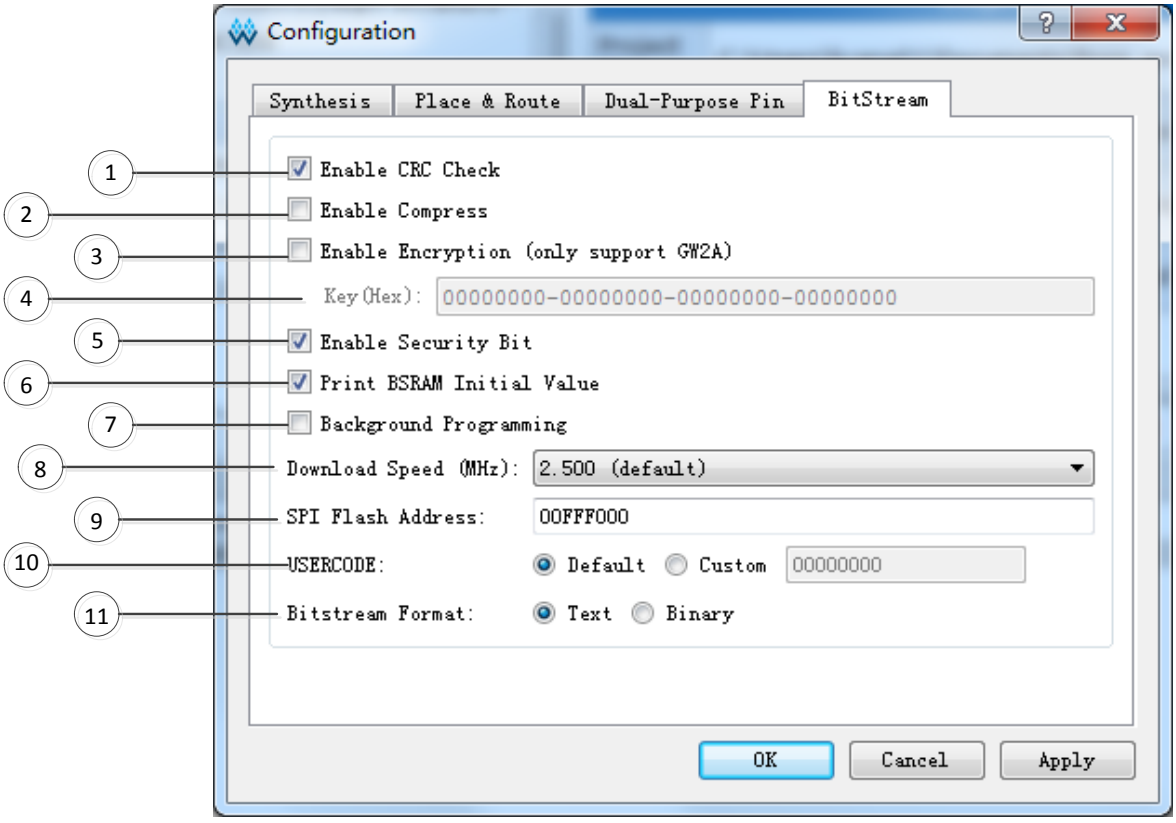
Note!

The JTAGSEL_N and JTAG pins are exclusive. When the JTAG pin is not checked, the JTAGSEL_N pin will be checked by default. When the JTAG pin is checked, the JTAGSEL_N pin will not be checked by default.

4. Bitstream

Users can configure the bitstream files format or frequency, etc. See Figure 5-34 for the detailed options.

Figure 5-34 Configure Bitstream File



① Enable a CRC Check
② Enable a Compress
③ Enable Encryption (only support GW2A)
④ User Defined Key (Hex)
⑤ Enable Security Bit
⑥ Print BSRAM Initial Value
⑦ Remote Upgrade
⑧ Download Speed (MHz)
⑨ SPI Flash Address
⑩ User Code
⑪ Bitstream Format

Note!

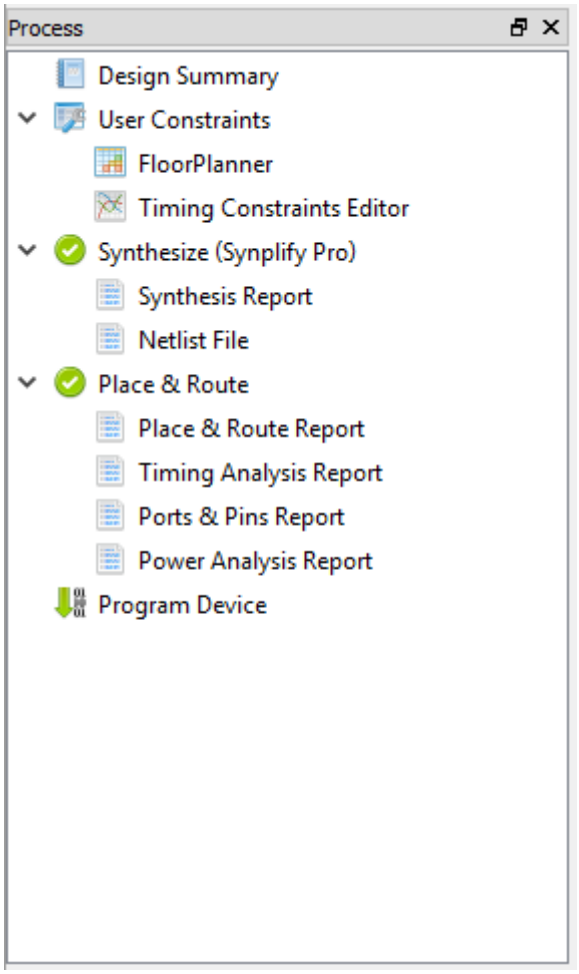
The SPI Flash address refers to the initial address to which the bitstream will be loaded for the next multiboot. For further details, please refer to *Gowin Programmer User Guide*.

5.4 Manage a Project

The Process view provides a system-level overview of the FGPA design flow, as shown in Figure 5-35. The Process View incorporates the following actions:

- View design summary;
- Start FloorPlanner;
- Start timing constraints editor;
- Implement Synthesis;
- View post Place & Route report;
- Implement Place & Route;
- Check the report generated after Place & Route;
- Start GOWINSEMI FPGA programmer, etc.

Figure 5-35 RTL Project Process Flow



Note!

There are no Synthesis (Synplify Pro) options for the post-synthesis because the design file has been synthesized.

5.4.1 Design Summary

When you create an RTL project, YunYuan software will provide a project summary, as shown in Figure 5-36. Use one of the following three methods 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 then select "Run".

Figure 5-36 Project Summary

Project Summary			
Project File:	C:\Users\liyang01\Documents\fpga_project_1\fpga_project_1.gprj		
Target Device:	GW1N-LV4LQ144C5/I4	Device Series:	GW1N
Device Part:	GW1N-4	Package Type:	LQFP144
Core Voltage:	LV	Speed Grade:	C5/I4

5.4.2 User Constraints

User constraints provide quick access to and creation of constraints files. For the detailed operation, please refer to the *Gowin Design and Constraint User Guide*.

User constraint contains the FloorPlanner and Timing Constraints Editor.

Please follow the steps outlined below to use the FloorPlanner:

1. Double-click “FloorPlanner” or select “Run” by right-clicking menu. YunYuan software will start the Synthesis and Compile Netlist first, and then open the Physical Constraints Editor.
2. If the physical constraints file already exists in the project, the editor will read it directly when the editor is opened.
3. If the existing physical constraints file (.cst) is modified and saved, click on the  icon in the Physical Constraints Editor to reload the modified constraints file;
4. If the project does not include the physical constraints file, and there is no constraints file in the source file location with the same name, YunYuan will prompt the user to create a constraints file, as shown in Figure 5-37;
5. If the project does not include a physical constraints file, but a constraints file with the same name exists in the source file directory, a warning dialoge box will appear and ask whether you want to override it or not, as shown in Figure 5-39.

Note!

Users can open the Floor Planner and Timing Constraints Editor directly instead of implementing design files synthesis by selecting "Tools > Floor Planner/Timing Constraints Editor".

Figure 5-37 Constraint File Prompt

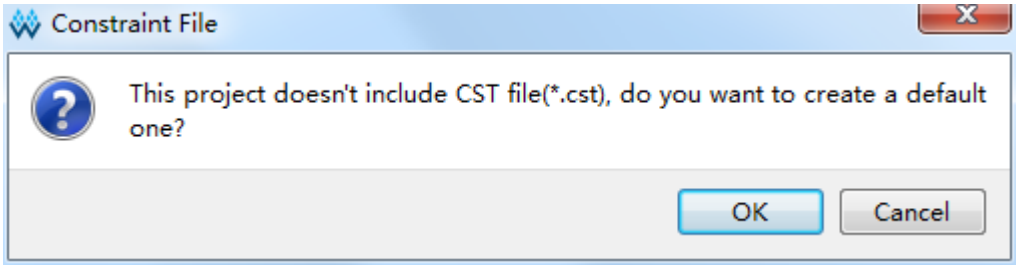
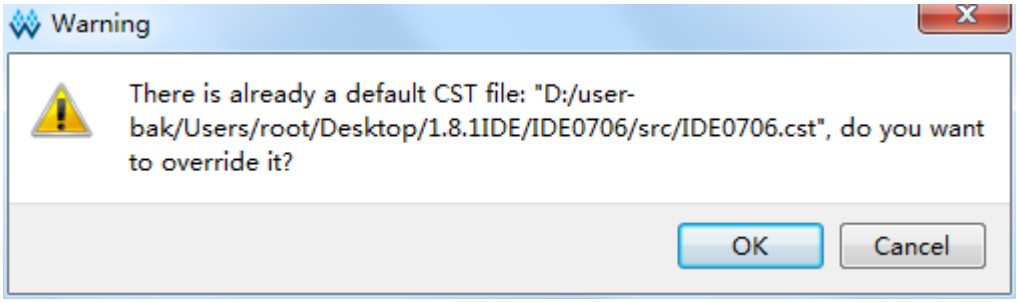


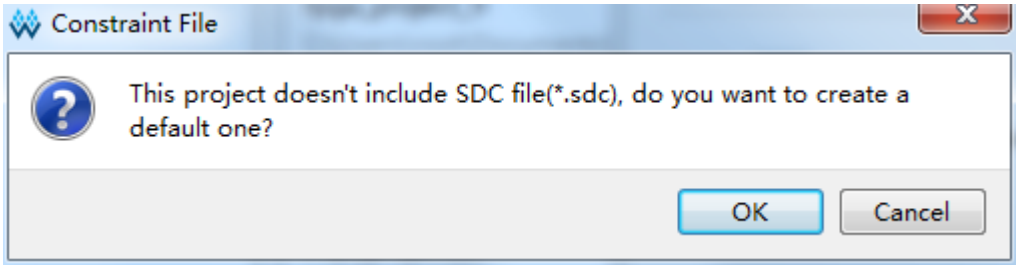
Figure 5-38 Warning-Creating a Constraints File



Refer to the following steps to use the Timing Constraints Editor:

1. Double-click on the "Timing Constraints Editor" link or select "Run" from the right-clicking menu. YunYuan software will start Synthesis first, and then open the Timing Constraints Editor.
2. If the timing constraints file (.sdc) exists in the project, the editor will read it directly when the editor is opened.
3. If the project does not include a timing constraints file, and there is no constraints file with the same name in the source file locaiton, YunYuan will prompt the user to create a constraint file, as shown in Figure 5-39;
4. If the project does not include a timing constraint file, but a constraint file exists with the same name exists in the source file directory, after systhesis, a warning dialoge box will appear and ask whether you want to override it or not, as shown in Figure 5-39.

Figure 5-39 Timing Constraint File



5.4.3 Synthesize (Synplify Pro)

Synthesize (Synplify Pro) is the Pro FPGA synthesis software customized by Synopsys for GOWINSEMI. It supports GOWINSEMI library files implementation, VHDL, and Verilog, etc.

Synthesize (Synplify Pro) provides functions of running Synplify Pro, setting Synplify Pro parameters and managing Netlist File and Synthesis Report.

Refer to the following steps to run Synthesize (Synplify Pro):

1. Configure synthesis attributes:

   To configure the Synthesis attributes configuration, please refer to 5.3.3Modify Project Configuration;

2. Run Synthesize (Synplify Pro)

   In the Process View, double-click “Synthesis” or right-click “Synthesize (Synplify Pro) > Run” to start the synthesis of the source files.
   If the synthesis is successful, the ✔ status icon will appear before Synthesize (Synplify Pro); if not, the ❗ status icon will appear.

3. After synthesis has been completed successfully, double click “Netlist Report” or right-click and select "Run" to view the Netlist Report. The netlist file will be generated with the project name.

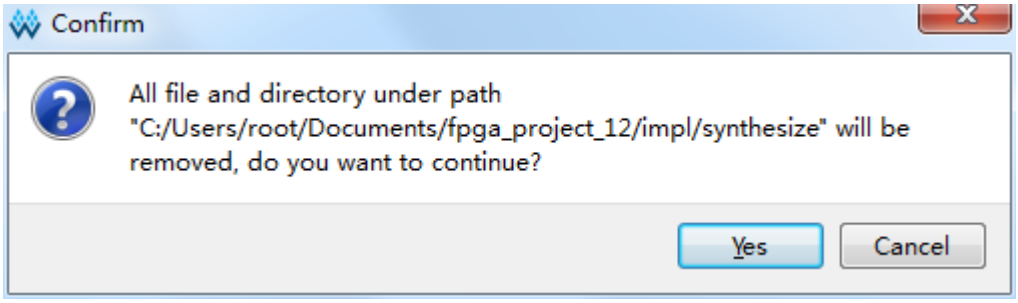
   Note!

- If the Synthesize (Synplify Pro) icon is before synthesis, double click “Netlist File” or right-click to select "Run" to synthesize first. The netlist file will open after successful synthesis;
- If the synthesize (Synplify Pro) icon is before synthesis, double-click “Synthesis Report” or right-click and select "Run" to synthesize first. The Synthesis Report will open after successful synthesis.

Right-clicking actions of Synthesize (Synplify Pro):

- Run: Users can select Run to start synthesis of source files only when the icon before Synthesis is , , or ;
- Rerun: Regardless of the Synthesize (Synplify Pro) status is, users can select Rerun to restart synthesis of the source files;
- Rerun All: Regardless of the Synthesize (Synplify Pro) status is, users can select Rerun to restart synthesis of the source files;
- Stop: Right-click and select Stop during synthesis; click "OK" in the prompted "Confirm" to stop synthesis;
- Clean: Select to clean all the generated files and file folders in the synthesize folder. When users select this option, "Confirm" dialog box will be displayed, as shown in Figure 5-40. Click "Yes" to confirm clean.
- Configuration: Used to configure Synthesis parameters.

Figure 5-40 "Clean" Confirm



5.4.4 Place & Route

Place and route includes the functionality required to run Place & Route, set the place and route parameters, and manage the post-P&R report.

Note!

Place & Route will be implemented after Synthesize (Synplify Pro) is run.

Refer to the following steps to run Place & Route:

1. Configure Place & Route attributes:

   To configure the Place & Route attributes, please refer to “5.3.3Modify Project Configuration”;

2. Run Place & Route:

   Double-click “Place&Route”, 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 "Run" to view the report.

   Users can view four kinds of files: Place & Route Report, Timing Analysis Report, Ports & Pins Report,and Power Analysis Report. These reports can not be edited.

   Note!

   - If the report is already opened and is regenerated by implementing Place & Route, YunYuan will ask the users whether they would like the report to be updated.;
   - Before implementing Place & Route, if the status icon before Place & Route is , double click the report or right-click and select "Run" to run Place & Route first. The report will open after Place & Route has been successfully.

Right-clicking actions of Place&Route:

- Run: Users can only select “Run” to start Place & Route only when the icon before Place&Route is or ;
- Rerun: Regardless of the Place&Route status, users can select Rerun to rerun Place&Route;
- Rerun All: Regardless of the Place&Route status, users can select Rerun all to rerun Place&Route;
- Stop: Right-click and select Stop during placement and routing. Click "OK" in the prompted "Confirm" window to stop Place&Route;
- Clean: Select to clean all the generated files and file folders after running Place & Route. When this option is selected, the "Confirm" dialog box will be displayed. Click "Yes" to confirm clean.
- Configuration: Used to configure Place & Route parameters.

5.4.5 Program Device

Bitstream files will be generated (.fs file) after Gowin YunYuan software has run placement and routing. Start Gowin FPGA programmer to download the bitstream files to the chip to realize user-required functions.

Note!

Bitstream generation will be implemented after running Synthesis, Compiling Netlist, and implementing Place & Route.

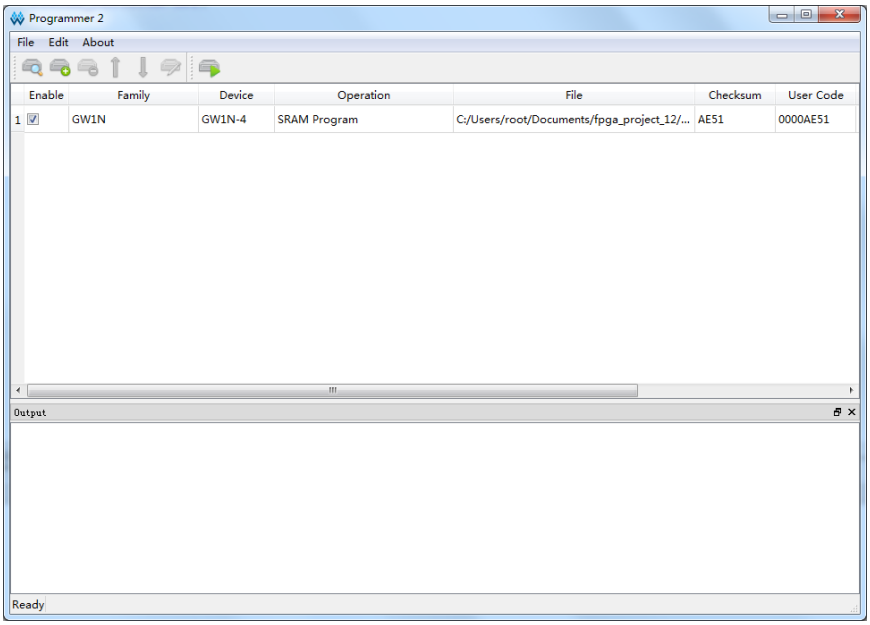
Double-click “Program Device” or right click and select “Run” to open the Gowin FPGA Programmer, as shown in Figure 5-41.

Right-click “Program Device”, select “Rerun All” to rerun all steps, including Synthesize, Place & Route, and Program Device.

Note!

The programmer in the Linux installation package applies to Linux Red Hat 5.10. If you need the Red Hat 6/7 programmer, please download it from the Gowin website, rename it as "Programmer", and replace the programmer in the Gowin YunYuan software installation package.

Figure 5-41 Gowin Programmer



For detailed instructions on how to use the Gowin programmer, please refer to *Gowin YunYuan Programmer User Guude*.

5.5 Exit IDE

Use the following two methods to exit IDE:

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 actions;
- 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 the user exits the IDE.

6 Tools in YunYuan Software

6.1 Synplify Pro

Synthesis is the Pro FPGA synthesis software customized by Synopsys for GOWINSEMI. It supports GOWINSEMI library files implementation, VHDL, and Verilog, etc.

For more detailed information about the operation of Synplify Pro, please refer to the manuals in the *"Help" drop-down list that appears on the Synplify Pro menu bar.*

6.2 FloorPlanner

The Gowin Floor Planner is Gowin self-developed tool used for Palce & Route and physical constraints editing. It supports attributes and location reading and modification of I/O, Primitive, block, Net, and Group, etc. It also supports the generation of new layout and constraints files per user configuration. The files define I/O attributes, primitives, module location, etc. Gowin Floor Planner supports all GOWINSEMI FPGA devices.

FloorPlanner can be started using two methods:

1. If no FPGA project is created, users can select "Tools > FloorPlanner" directly from the menu bar. The user will need to add netlist files, constraints files, and devices information by selecting "File > New";
2. If an FPGA project is already created, double-click “FloorPlanner” directly in the Process View. The Floorplanner will then load the project files directly before displaying them.

The FloorPlanner contains Chip Array and Package View. See

Figure 6-1 and Figure 6-2 for examples. For more detailed operation of FloorPlanner, please refer to the *Gowin Design Constraints Guide*. The FloorPlanner also supports timing optimization.

Figure 6-1 Chip Array View

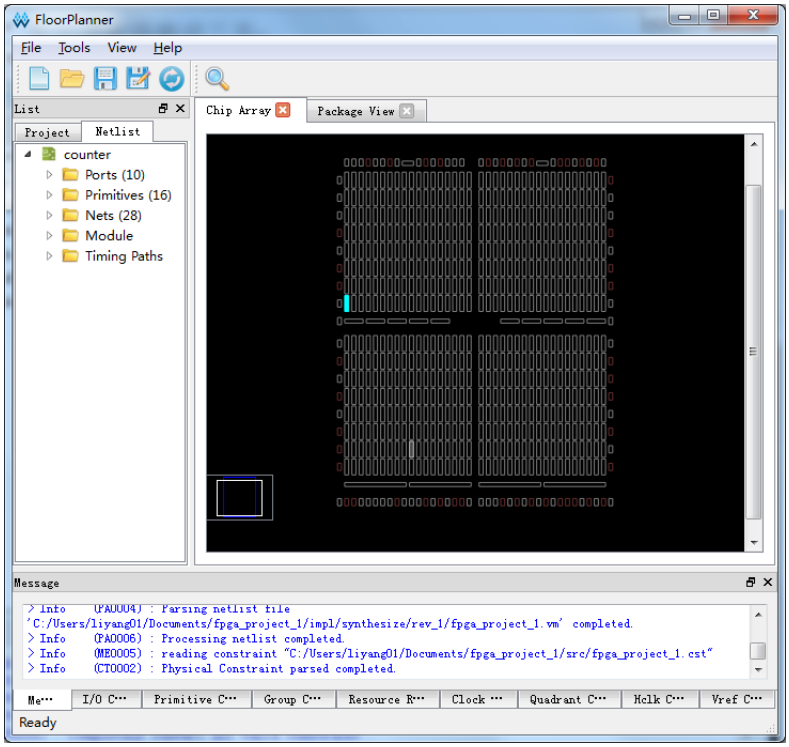
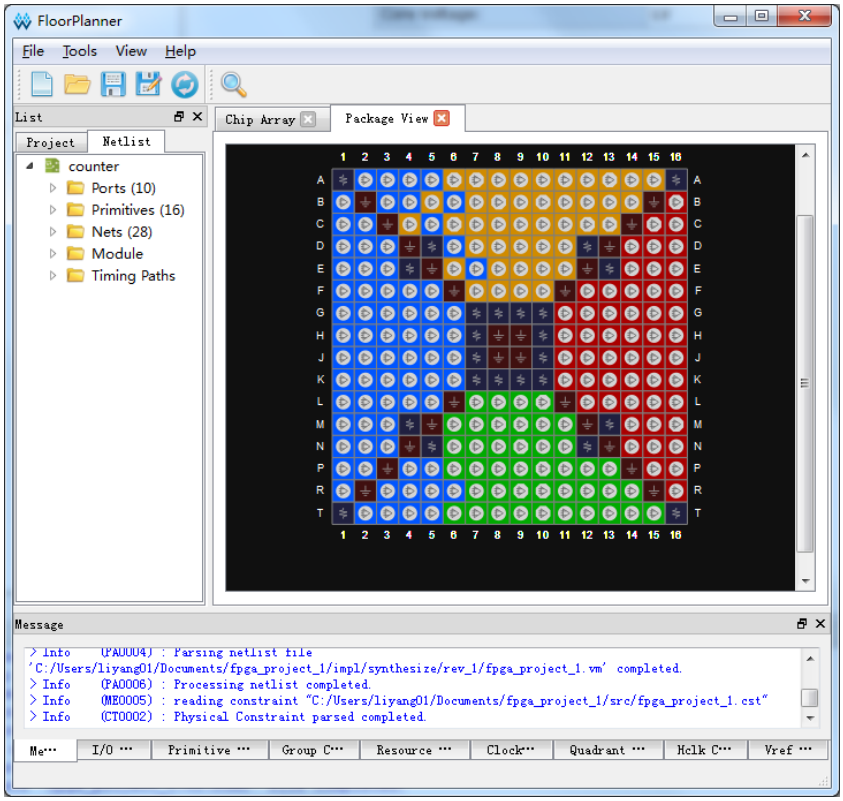


Figure 6-2 Package View



6.3 Timing Constraints Editor

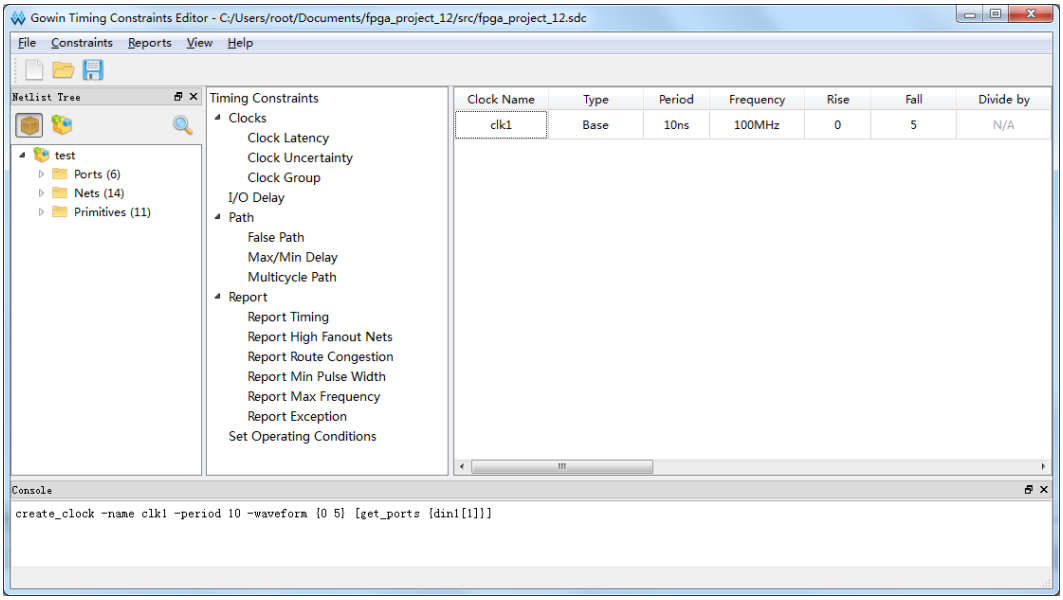
The Gowin Timing Constraints Editor supports multiple timing constraints commands editing, including clock constraints, I/O constraints, path constraints, and clock report constraints. The Timing Constraints Editor allows an easy and quick timing constraints editing. It supports all GOWINSEMI FPGA devices.

Timing Constraints Editor can be started using two methods:

1. If no FPGA project is created, users 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, double-click “Timing Constraints Editor”, and the Timing Constraints Editor will load project files directly and display them, as shown in Figure 6-3.

For the detailed operation, please refer to *Gowin Design and Constraint User Guide*.

Figure 6-3 Clock Creation Interface



6.4 Simulation

Simulation is essential during the complete FPGA design flow. Function simulation (Pre-simulation) is required before and after synthesis to verify the suitability of the RTL design. Timing simulation (Post-simulation) is required after placement and routing to evaluate the devices time delay.

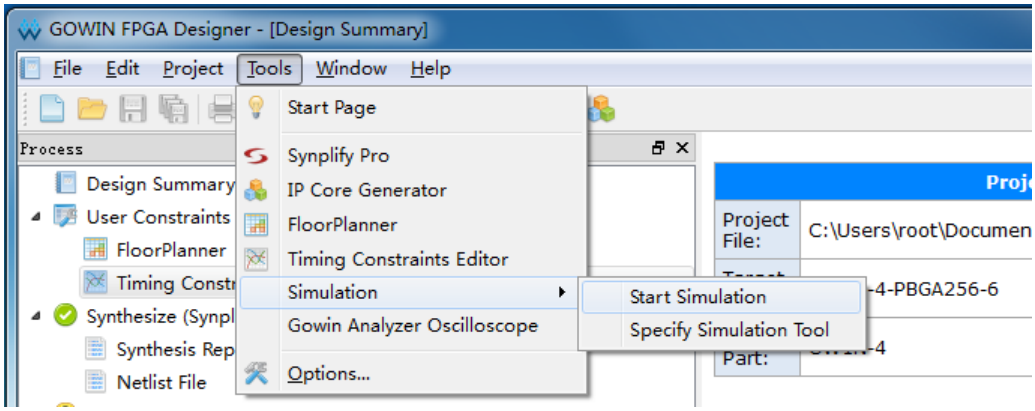
Users can select to use the third-party simulation tools for function simulation and timing simulation.

Select "Tools > Simulation" in the menu bar to start the simulation, as shown in Figure 6-4.

Note!

Apply to the third-party for the simulation tool license.

Figure 6-4 Start Simulation



The library files for the simulation are available at Gowin\1.8\Pnr\lib\, of which prim_sim.v is for function simulation, and prim_tsim.v is for the timing simulation. For more detailed information of the third-party simulation tool, please refer to *the third-party user guides*.

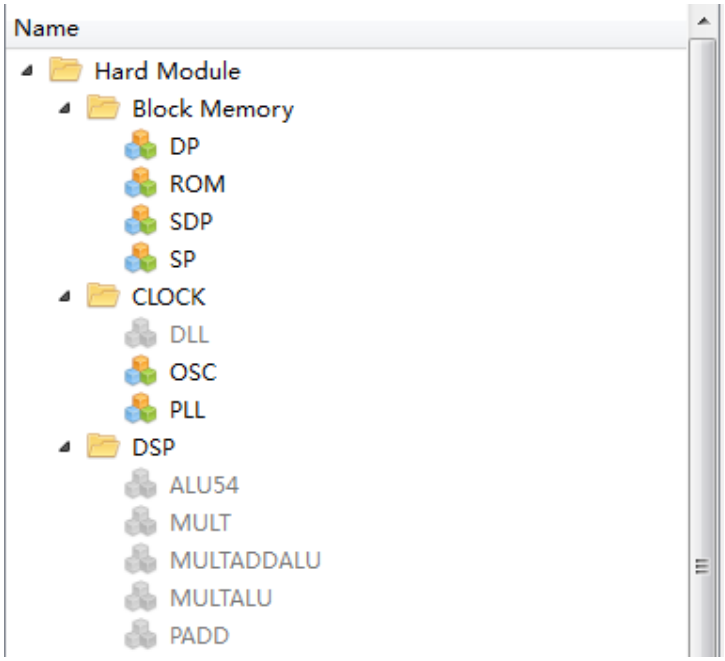
6.5 IP Core Generator

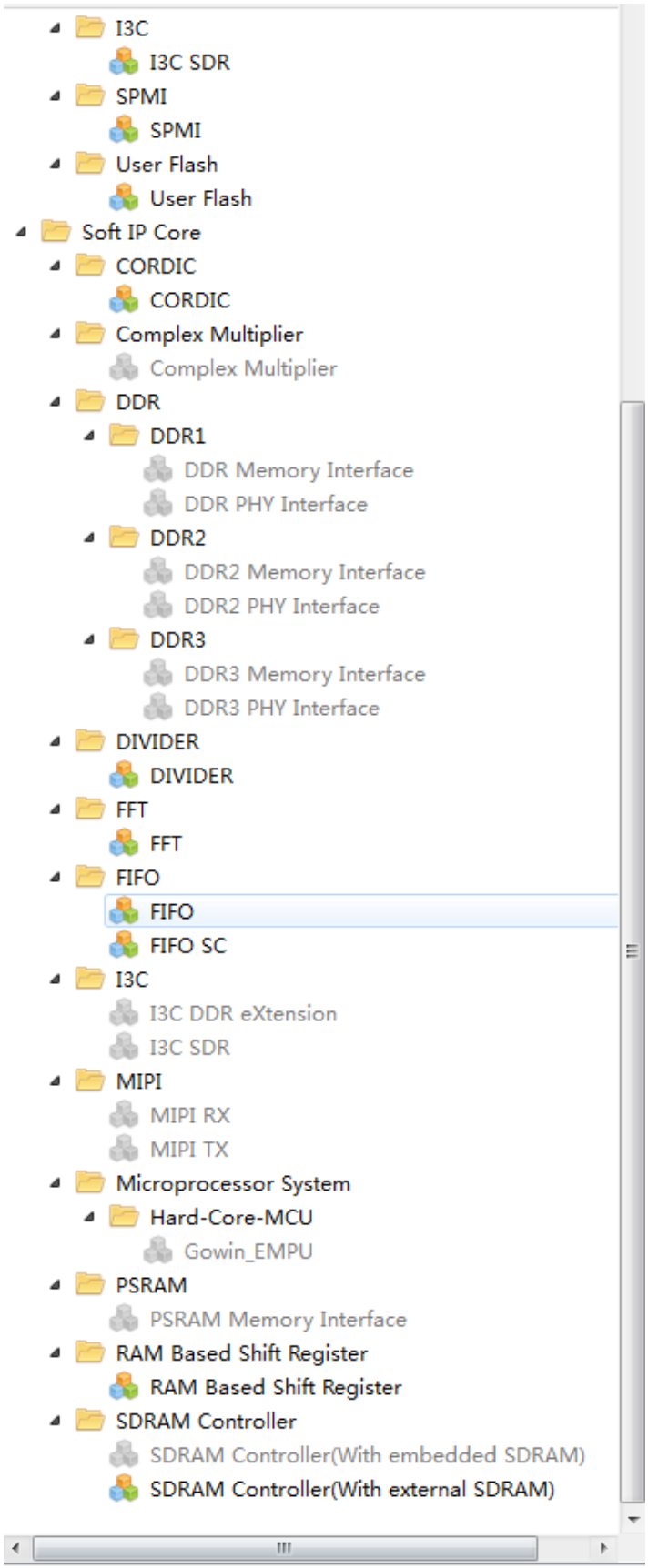
The IP Core Generator in Gowin software is mainly used to generate instantiation components and IPs, which the users can call to implement the required functions. As such, they provide users with a convenient method of creating complex designs. The IP Core Generator includes the modules associated with primitives and the IP Cores associated with reference designs, as shown in Figure 6-5.

Start the IP Core Generator by selecting “IP Core Generator” from the Tools menu.

For further details about the each IP call method, please refer to *Gowin IP Core Generator User Guide.*

Figure 6-5 IP Core Generator Page





Note!

The current device does not support the Hard Modules and the Soft IP Cores which are displayed as grey.

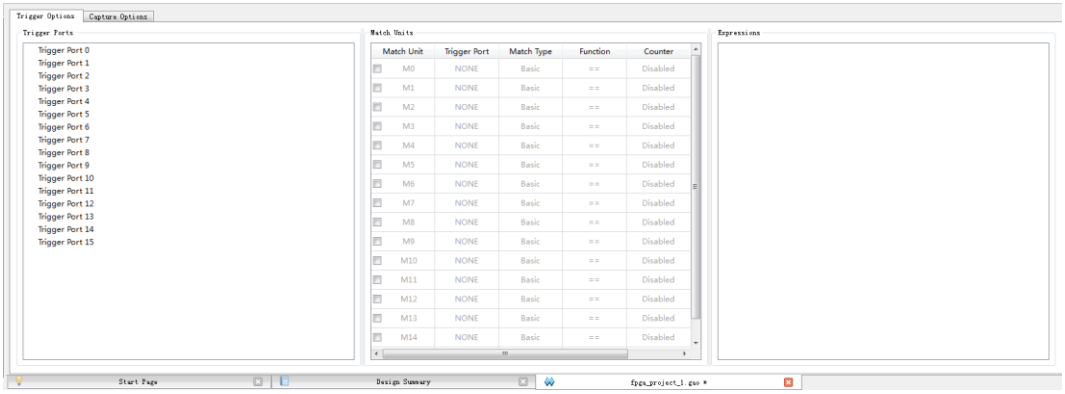
6.6 Gowin Analyzer Oscilloscope

The Gowin Analyzer Oscilloscope (GAO) is a digital signal analyzer that was independently designed by Gowin. It helps users to analyze signal timing in design more easily, and quickly conduct system analysis and fault location, thereby improving design efficiency.

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

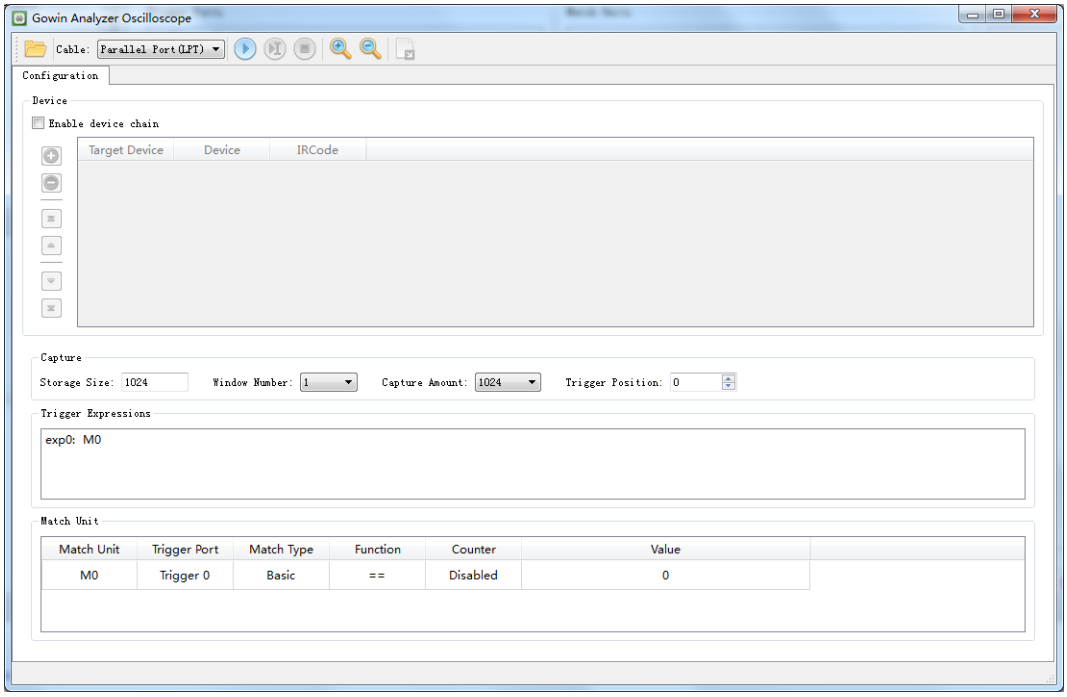
Before starting the GAO, create the GAO configuration file (.gao) in the Project View to open the GAO configuration view, as shown in Figure 6-6.

Figure 6-6 GAO Configuration View



After the configuration file has been created, select “Tools > Gowin Analyzer Oscilloscope" from the menu bar to open the Gowin Analyzer Oscilloscope, as shown in Figure 6-7.

Figure 6-7 GAO



For further details about how to use the Gowin Analyzer Oscilloscope, please refer to the *Gowin Analyzer Oscilloscope User Guide*.

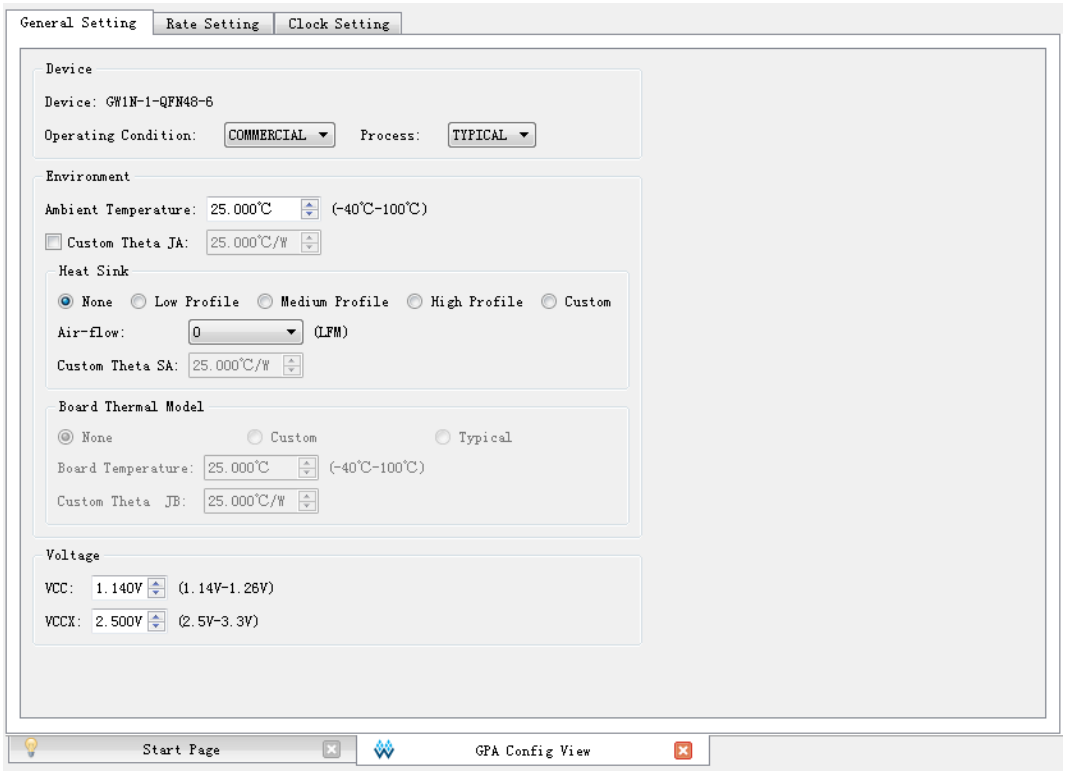
6.7 Gowin Power Analyzer

The Gowin Power Analyzer (GPA) estimates the power dissipation for your design and provides rich user configuration options. It should be configured per the actual design. The closer the configuration is to the actual design, the more accurate the power dissipation analysis will be.

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

1. In the File menu, select “File> New…” to open a “New” dialog box;
2. Select “GPA Config File” and enter “Name” in the pop-up "New GPA Config File" ;
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 6-8.

Figure 6-8 GPA Config View



For further details about how to use the GPA, please refer to the *Gowin Power Analysis User Guide*.

6.8 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.

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.

The Gowin Memory Initialization File supports the binary, hexadecimal, and address-hex formats. The following are examples of the .mi file format.

1. Bin File

Bin file is a text file that consists of the 0 and 1 binary numbers. The line represents address depth, and the column represents data width.

```
#File_format=Bin
#Address_depth=16
#Data_width=32
00001100000100000000100100010000
10000000010010000100000001000000
01000000100000001000000010000000
00100000100001001100000011000000
00000100000001000000010000000100
01000010001010100000101000001010
00100010010001000000010001000110
01000101000001110000010101000001
01100100001001001000010000100100
01001001010010010010100101001001
01100101001001010000010100100101
11000001101001010000010110100101
01001000100010000010110000001100
10000101001011010100110100101101
01101100001100110000011100011001
00001001001010010100000110010000
```

2. Hex File

The Hex file is similar to the Bin file. It consists of hexadecimal numbers 0~F. The line represents the address depth, and the binary bits in each line represent data width.

```
#File_format=Hex
```

```
#Address_depth=8
#Data_width=16
3A40
A28E
0B52
1C49
D602
0801
03E6
4C18
```

3. Address-Hex File

Address-Hex file is the file that includes both the data and the address with data record. The address and the data is composed of the hexadecimal number of 0~F. In each line, the address is located before the colon, and the data is located after the colon. The address with no data record is 0 by default.

```
#File_format=AddrHex
#Address_depth=256
#Data_width=16
9:FFFF
23:00E0
2a:001F
30:1E00
```

Based on the new configuration file (.mi), refer to the following steps to use the initialization file editor:

1. In the File menu, select “File> New…” to open the “New” dialog box;
2. Select “Memory Initialization File”, as shown in Figure 6-9. Click “OK” and enter the initialization file name in the pop-up "New File" window, and then click “OK”, as shown in Figure 6-10. The Initialization File Configuration View is as shown in Figure 6-11. Enter the initial value on the left side and configure the initialization file format and depth/width and view on the right side;
3. 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 BSRAM 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 display the error message. If the depth and width values are

less than the BSRAM 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 address and numbers on the left can be configured as binary, hexadecimal, address-hex, etc.

4. Enter the initial value on the left table, and set the view layout.
   - Right-click on the table header to configure the column number which can be displayed as 1, 8, or16, as shown in Figure 6-12.
   - Double-click and enter 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 write according to your needs or batch setting, as shown in Figure 6-13.
5. Save the file.

Figure 6-9 New Memory Initialization File

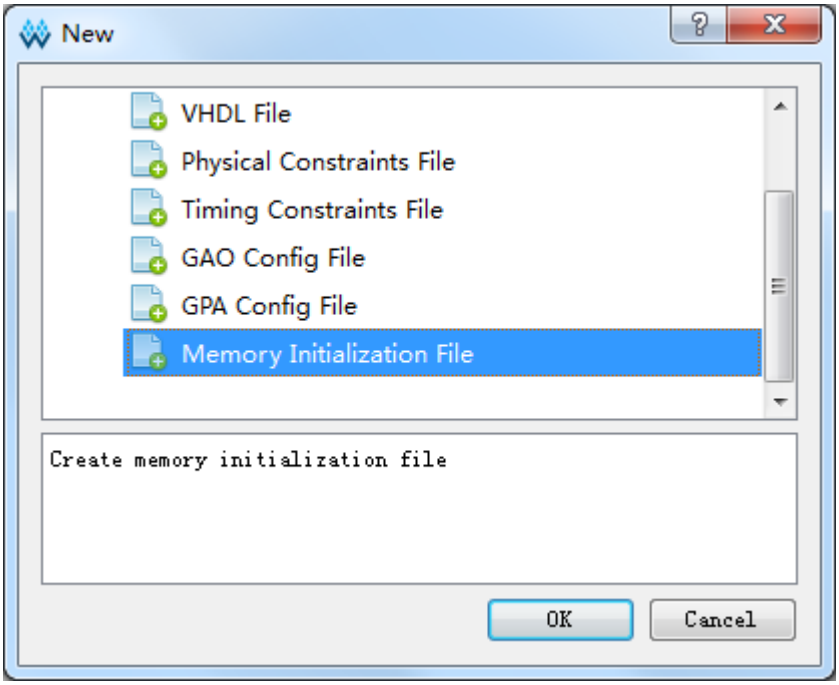


Figure 6-10 New File

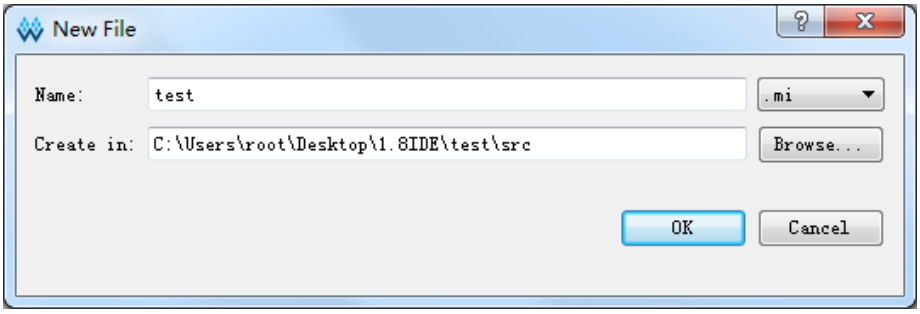


Figure 6-11 Initialization File Configuration

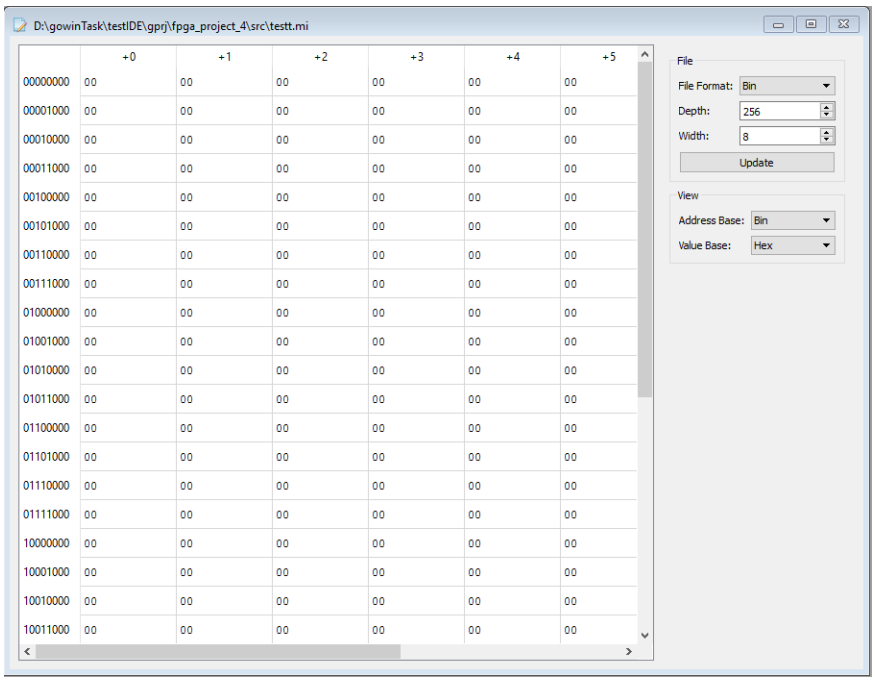


Figure 6-12 Column Setting

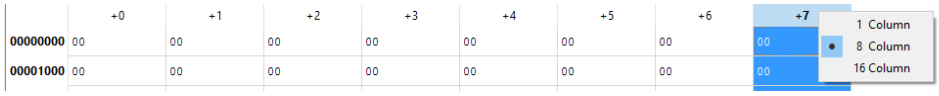
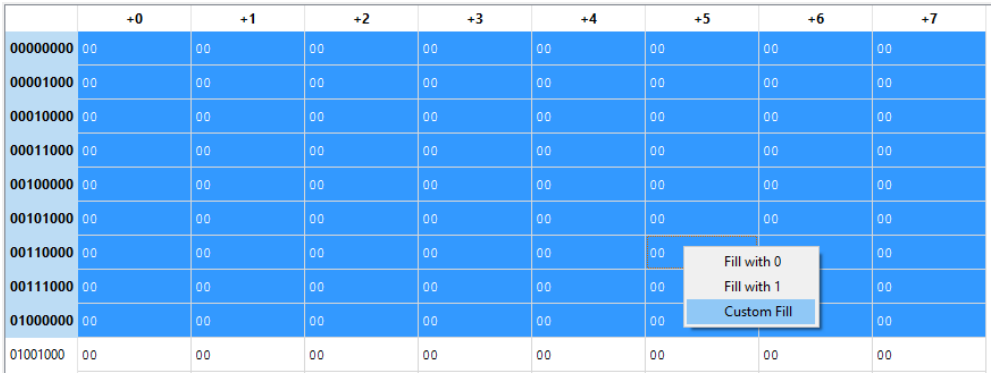


Figure 6-13 Batch Setting



7 Description of Gowin YunYuan Output Files

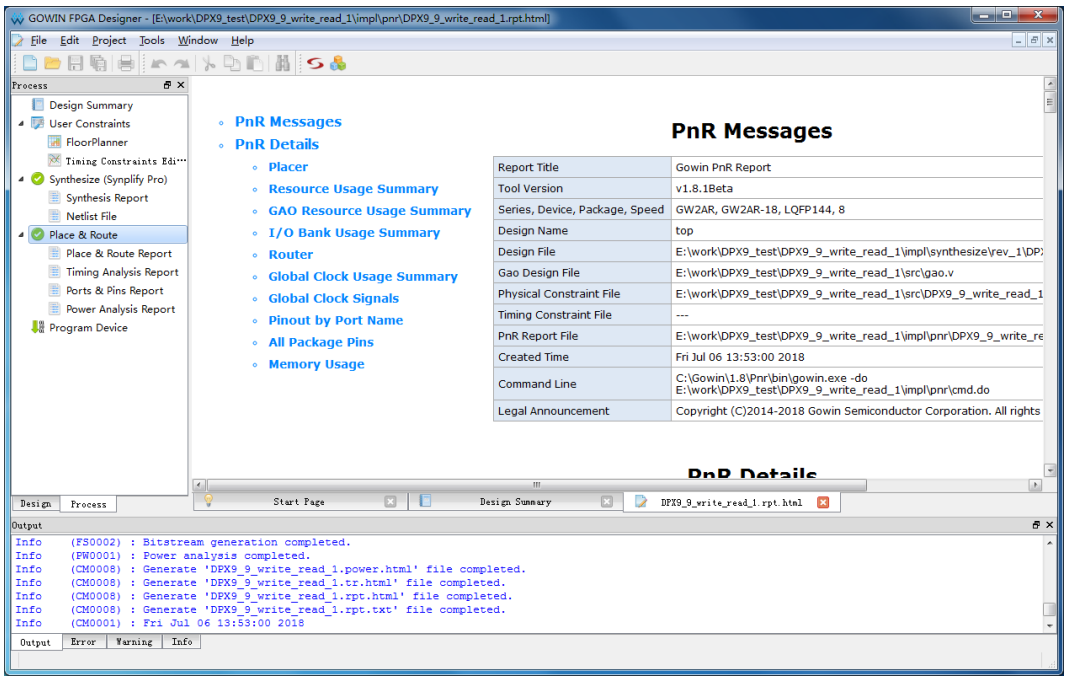
In the process of FPGA design, Gowin YunYuan software generates bitstream files, and pin constraints location files (io.cst), and also generates multiple reports depending on different parameters. The reports include the place&route report, timing report, and power analysis report. In addition, you can right-click on Place & Route to modify the configuration and attributes and generate pins constraints, physical netlist files, etc.

7.1 Place&Route Report

The Place&Route Report describes the resource, memory consumption, time consumption, etc. occupied by the user design, with the file suffix as .rpt.

Double-click “Place & Route Report” in the Process View to open the project that corresponds to the Place&Route report, as shown in Figure 7-1.

Figure 7-1 Place&Route Report

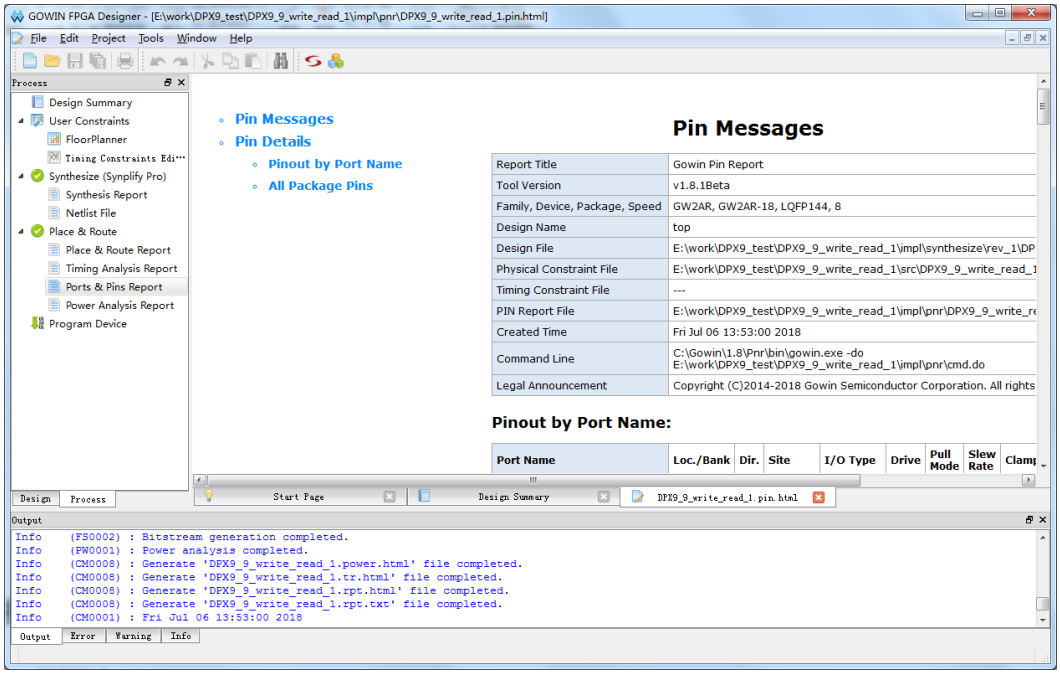


7.2 Ports and Pins Report

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

Double-click “Ports & Pins Report” in the Process View to open the report corresponding to the project, as shown in Figure 7-2.

Figure 7-2 Ports & Pins Report



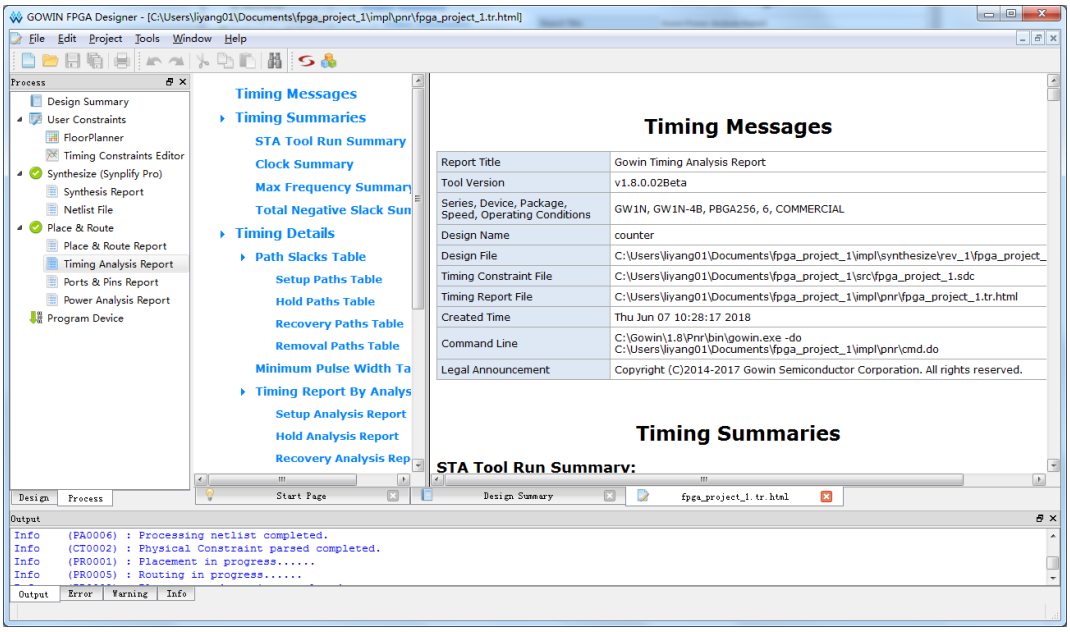
7.3 Timing Report

The timing report is available in web format and text format. The default is web format.

The Timing report includes set-up time check, hold-time check, restoring 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 View to open the timing analysis report for the project, as shown in Figure 7-3.

Figure 7-3 Timing Report



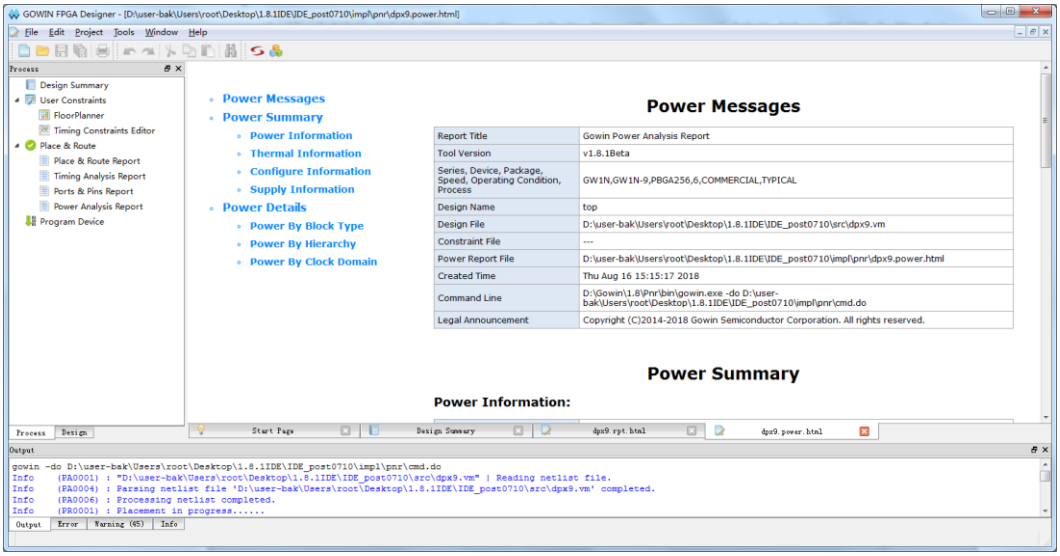
7.4 Power Analysis Report

The Power Analysis Report mainly includes the power dissipation 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 that corresponds to the project, as shown in Figure 7-4.

To adjust the factors that influence power dissipation, please refer to *Gowin Power Analysis User Guide*.

Figure 7-4 Power Analysis Report



Appendix A SynplifyPro Attributes and Directives

A.1Attributes and Directives

full_case

For Verilog designs only. Indicates that all possible values have been given, and that no additional hardware is needed to preserve the signal values.

Syntax

object /* synthesis full_case */ ;

Description

"object" can be a case, casex, or casez statement.

parallel_case

For Verilog designs only. Forces a parallel-multiplexed structure rather than a priority-encoded structure.

Syntax

object /* synthesis full_case */ ;

Description

"object" can be a case, casex, or casez statement.

translate_on/translate_off

Synthesizes designs originally written for use with other synthesis tools without needing to modify the source code. Allows you to synthesize designs that were originally written for use with other synthesis tools without needing to modify the source code. All source code that is between these two directives is ignored during synthesis.

Syntax

/* synthesis translate_off */

code

/* synthesis translate_on */

syn_encoding

Overrides the default FSM Compiler encoding for a state machine and applies the specified encoding.

Syntax

object /* synthesis syn_encoding = "value" */ ;

Description

- object defines the state register;
- value includes:

  default: The default is that the tool automatically picks an encoding style that results in the best performance.

  onehot: Only two bits of the state register change (one goes to 0, one goes to 1) and only one of the state registers is hot (driven by 1) at a time;

  gray: Adopts gray encoding style;

  sequential: Adopts natural encoding style;

  safe: This implements the state machine in the default encoding and adds reset logic to force the state machine to a known state if it reaches an invalid state.

  original: This respects the encoding you set, but the software still does state machine and reachability analysis.

syn_keep

Preserves the specified net and keeps it intact during optimization and synthesis.

Syntax

object /* synthesis syn_keep = 0|1 */ ;

Description

object is the declared object of wire or reg.

syn_hier

Controls the amount of hierarchical transformation across boundaries on module or component instances during optimization.

Syntax

In an FDC file:

define_attribute {object } syn_hier { value }

define_global_attribute syn_hier {flatten}

In a Verilog file:

object /* synthesis syn_hier = " value " */ ;

In a VHDL file:

attribute syn_hier of object : architecture is " value " ;

Description

syn_hier value includes:

- soft（default value）: Synplify decides the optimum boundary, affecting the specified design cell only;
- firm: Preserve the design cell interface, and the cell is allowed to across boundary, affecting the specified design cell only;
- hard: Preserve the design cell interface strictly, affecting the specified design cell only;
- fixed: Preserve the design cell interface, fix all the cross boundaries during optimization, and preserve the ports interface;
- remove: Remove the hierarchy of property-declaration, but not affecting the lower hierarchy cells;
- macro: Preserve the design cell interface and content strictly;
- flatten: Use flatten combined with other values. 如 flatten,soft 等于 flatten；flatten,firm 平面化低于属性所在层次的所有层次，保留设计单元的界面保留对跨越边界的 cell 的优化；flatten,remove 平面化包括属性所在层次和低于该层次的所有层次，在低层声明的该属性取代在高层声明的这一属性。

loop_limit

Specifies a loop iteration limit for the loop in a Verilog design when the loop index is variable, not constant.

Syntax

beginning_of_loop_statement /* synthesis loop_limit integer */;

syn_looplimit

Specifies a loop iteration limit for a for loop in a VHDL design when the loop index is a variable, not a constant.

Syntax

attribute syn_looplimit : integer;

attribute syn_looplimit of labelName : label is value;

syn_noprune

Prevents optimizations for instances and black-box modules (including technology-specific primitives) with unused output ports. By default, the synthesis tool removes any module that does not drive logic as part of the synthesis optimization process.

Syntax

In an FDC file:

define_attribute {module|instance} syn_noprune {0|1}

In a Verilog file:

object /* synthesis syn_noprune = 1 */ ;

In a VHDL file:

attribute syn_noprune : boolean

Description

The object can be module, declaration, or instances.

syn_preserve

Prevents sequential optimizations such as constant propagation, inverter push-through, and FSM extraction.

Syntax

In a Verilog file:

object /* synthesis syn_preserve = 0 |1 */

In a VHDL file:

attribute syn_preserve of object : objectType is true | false;

Description

object can be register signal or Module.

A.2 Mapping Attributes and Directives

syn_allow_retiming

Determines if registers can be moved across combinational logic to improve performance.

Syntax

In an FDC file:

define_attribute {register} syn_allow_retiming {1|0}

define_global_attribute syn_allow_retiming {1|0}

In a Verilog file:

object /* synthesis syn_allow_retiming = 0 | 1 */;

In a VHDL file:

attribute syn_allow_retiming of object : objectType is true | false;

syn_ramstyle

Specifies the implementation for an inferred RAM. Can be mapped to registers signal driven by RAM or RAM instances.

Syntax

In an FDC file:

define_attribute { signalname [bitRange]} -syn_ramstyle value

define_global_attribute syn_ramstyle value

In a Verilog file:

object /* synthesis syn_ramstyle =value */

In a VHDL file:

attribute syn_ramstyle of object : objectType is value;

syn_romstyle

You can infer ROM architectures using a case statement in your code. For the synthesis tool to implement a ROM, at least half of the available addresses in the case statement must be assigned a value.

Syntax

In an FDC file:

define_attribute { romPrimitive } syn_romstyle {logic | block_ram | lpm_rom | MLAB | distributed | select_rom }

In a Verilog file:

object /* synthesis syn_romstyle = "logic | block_ram | lpm_rom | MLAB | distributed | select_rom" */ ;

In a VHDL file:

attribute syn_romstyle of object : objectType is "logic | block_ram | lpm_rom | MLAB | distributed | select_rom";

syn_replicate

Controls replication of registers during optimization.

Syntax

In an FDC file:

define_global_attribute syn_replicate {0 | 1};

In a Verilog file:

object /* synthesis syn_replicate = 1 | 0 */;

In a VHDL file:

attribute syn_replicate : boolean;

attribute syn_replicate of object : signal is true|false;

syn_direct_enable

Controls the assignment of a clock enable net to the dedicated enable pin of a storage element (flip-flop).

Syntax

In an FDC file:

efine_attribute {object} syn_direct_enable {1};

In a Verilog file:

object /* synthesis syn_direct_enable = 1 */;

In a VHDL file:

attribute syn_direct_enable of object : objectType is true;

syn_direct_reset

Controls the assignment of a net to the dedicated reset pin of a synchronous storage element (flip-flop).

Syntax

In an FDC file:

define_attribute syn_direct_reset {0|1};

In a Verilog file:

object /* synthesis syn_direct_reset = 0|1*/;

In a VHDL file:

attribute syn_direct_reset : boolean;

attribute syn_direct_reset of Object : signal is true|false;

syn_direct_set

Controls the assignment of a net to the dedicated set pin of a synchronous storage element (flip-flop).

Syntax

In an FDC file:

define_attribute syn_direct_set {0|1};

In a Verilog file:

object /* synthesis syn_direct_set = 0|1*/;

In a VHDL file:

attribute syn_direct_set : boolean;

attribute syn_direct_set of Object : signal is true|false;

syn_black_box

Specifies that a module or component is a black box for synthesis. A black box module has only its interface defined for synthesis, its contents are not accessible and cannot be optimized during synthesis. A module can be a black box whether or not it is empty.

Syntax

In an CDC file:

define_attribute { object }{syn_black_box}{1}

In a Verilog file:

object /* synthesis syn_black_box */ ;

In a VHDL file:

attribute syn_black_box of object : objectType is true;

syn_maxfan

Overrides the default (global) fanout guide for an individual input port, net, or register output.

Syntax

In an FDC file:

define_attribute {object} syn_maxfan {integer}

In a Verilog file:

object /* synthesis syn_maxfan = "value" */;

In a VHDL file:

attribute syn_maxfan of object : objectType is "value";

syn_pipeline

Specifies that registers that are outputs of multipliers can be moved to

improve timing.

Syntax

In an FDC file:

define_attribute { register } syn_pipeline {0|1}

In a Verilog file:

object /* synthesis syn_pipeline = {1|0} */ ;

In VHDL file:

attribute syn_pipeline of object : objectType is {true|false};

syn_probe

Inserts probe points for testing and debugging the internal signals of a design.

Syntax

In an FDC file:

define_attribute {n:netName} syn_probe{probePortname|1|0}

In a Verilog file:

object /* synthesis syn_probe = "string" | 1 | 0 */;

In a VHDL file:

attribute syn_probe of object : signal is "string" | 1 | 0;

syn_useenables

Controls the use of clock-enable registers within a design.

Syntax

In an FDC file:

define attribute {register|signal} syn_useenables {0|1}

In a Verilog file:

object /* synthesis syn_useenables = "0|1" */;

In a VHDL file:

attribute syn_useenables of object : objectType is "true|false";

syn_netlist_hierarchy

Determines if the generated netlist is to be hierarchical or flat.

Syntax

In an FDC file:

define_global_attribute syn_netlist_hierarchy {0 | 1}

In a Verilog file:

object /* synthesis syn_netlist_hierarchy = 0 | 1 */;

In a VHDL file:

attribute syn_netlist_hierarchy of object : objectType is true | false;

Appendix B Command Line Options

-d | -design <file>

Required, specifies the logic netlist file for input after synthesis.

-p device-package-speed

- Optional, specifies the device info, including device, package, and speed.
- The default device: *GW1N-4*，Package: *PBGA256*，Speed: *5*.
- Example: gowin –d test.v –p gw2a-55-PBGA1156-6

  //Device:*GW2A-55*; Package: *PBGA1156*; Speed: *6*.

  //Abbreviated as: gowin –d test.v –p 55-1156-6

-c | -cst <file>

Optional. Specifies the Physical Constraints File .cst.

-sdc <file>

Optional. Specifies the Timing Constraints File .sdc.

-cfg <file>

- Optional. Specifies the configuration file, used for multiplexing pins and bitstream files configuration;
- If the file name is not specified, YunYuan software will directly read “device.cfg” in the current;
- If this file does not exist, or this option is not used, YunYuan software will read the default configuration file. The default configuration file is directly located in the software installation directly, such as /x.x/Pnr/cfg.

-o | -out <fileName>

- Optional. Used to output the physical netlist for the simulation;
- If the filename is not specified, the filename for physical netlist defaults to the input netlist file name + “.vo”.

-top_module

Optional. Specify the top module for input netlist files.

-bit <fileName>

- Optional, Used to generate bitstream files;
- If the filename is not specified, the filename for bit files defaults to the input netlist file name + ".fs".

-ibs <fileName>

- Optional. Used to generate IBIS files;
- If filename is not specified, the filename for IBIS files defaults to the input netlist file name + ".ibs".

-gao

Optional. Specifies the GAO flow file.

-gpa <fileName>

- Optional. Generates power analysis report in html format by default;
- If the filename is not specified, the filename for the power analysis report defaults to the input netlist file name + "_power.html".

-t | -tr <fileName>

Optional. Generates the timing analysis report in html format by default;

-tt <fileName>

Optional. Generates the timing analysis report in text format by default;

-s | -sdf <fileName>

- Optional. Generates the standard delay file used for timing simulation;
- If the filename is not specified, the filename for the delay file defaults to the input netlist file name + ".sdf".

-ocst | -oc <fileName>

- Optional. Generates the port location constraints file;
- If the filename is not specified, the filename for location constraints file defaults to: the input netlist file name + ".csto".

-timing

Optional. Used to run timing driven.

-posp <fileName>

- Optional. Generates the instance placement file after P&R, used for reading floorplan;
- If the filename is not specified, the filename for the instance placement file defaults to the input netlist file name + ".posp".

-prep <*.posp>

Optional. Sepcifies the posp file.

-init_all

Optional. Outputs all initial values of the Instance to the generated physical netlist.

-warning_all

Optional. Displays all warning information. If this parameter is not used, only 10 warnings will be displayed.

-do <cmdFile>

- Specifies the file used to write software running commands.
- design.do file reference style:

```
-d design.vm
-p GW2A-18-PBGA484-6
-cfg device.cfg
-bit
-tr
```

-h

Optional. Output Gowin help info. with the parameters -h | --h | -H | --H | -help |--help.

-v |-V

Optional. Displays Gowin version Info.

Appendix C Design Instance

Take the user design demo.v, for instance. Refer to the following use of Gowin YunYuan Software:

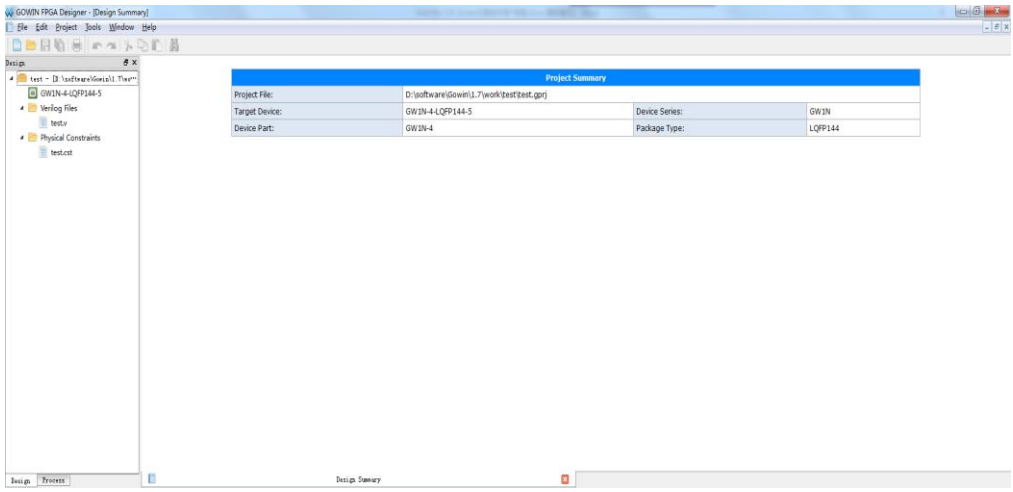
```
module demo (clk_50M, rst_n,led);
input clk_50M;
input   rst_n;
output[3:0] led;
reg[3:0]      led;
reg[24:0]   cnt;
reg           clk_led;

always@(posedge clk_50M or negedge rst_n) begin
    if (!rst_n) begin
            clk_led <= 1'b0;
            cnt <= 25'd0;
    end
    else begin
        if (cnt == 25'd2499_9999) begin
                clk_led <= ~clk_led;
                cnt <= 25'd0;
        end
        else begin
            cnt <= cnt + 25'd1;
        end
    end
```

```
end
always@(posedge clk_led or negedge rst_n) begin
    if(!rst_n) begin
            led <= 4'h1;
    end
    else begin
            led <= {led[2:0],led[3]};
    end
end
endmodule
```

C.1 Create a Project

Refer to 5.1Create a New Project and 5.2Open an Existing Project. Open YunYuan software, create RTL project and add demo.v to the project, and configure design parameters, as shown in Figure C-1.

Figure C-1 Gowin YunYuan IDE

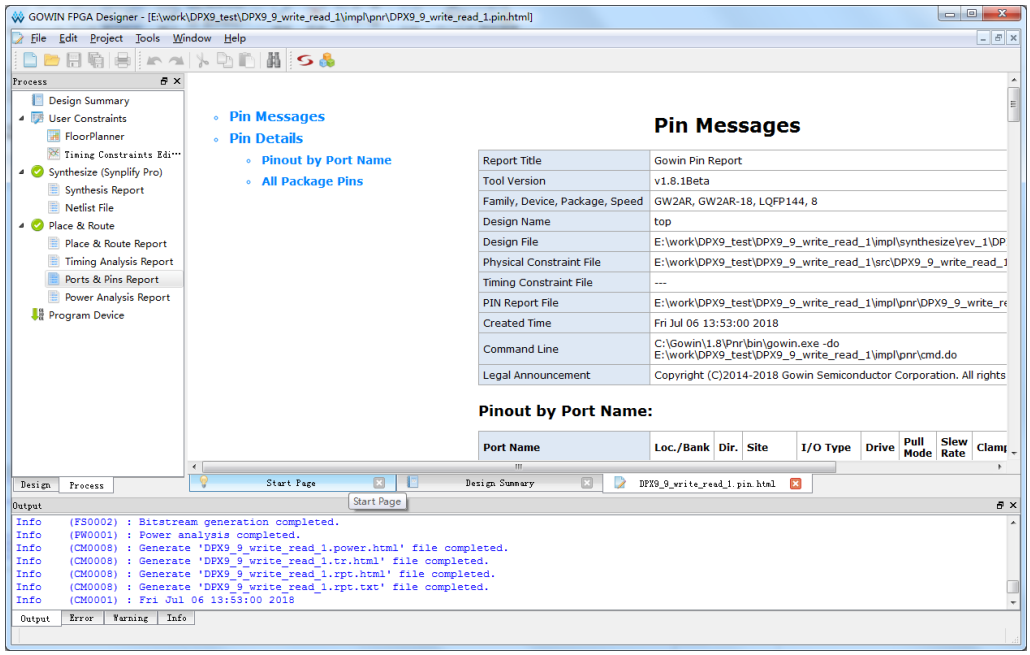


C.2 Implement a Project

See the instructions presented below to implement a project:

1. In Process View, double click “Synthesize (Synplify Pro)” or right-click and select "Run" to synthesize your design;
2. After synthesis, double click “Place & Route” or right-click and select "Run" for logic netlist placement and routing and bitstream file generation, as shown in Figure C-2.

Figure C-2 Implement a Project



C.3 Download Bitstream

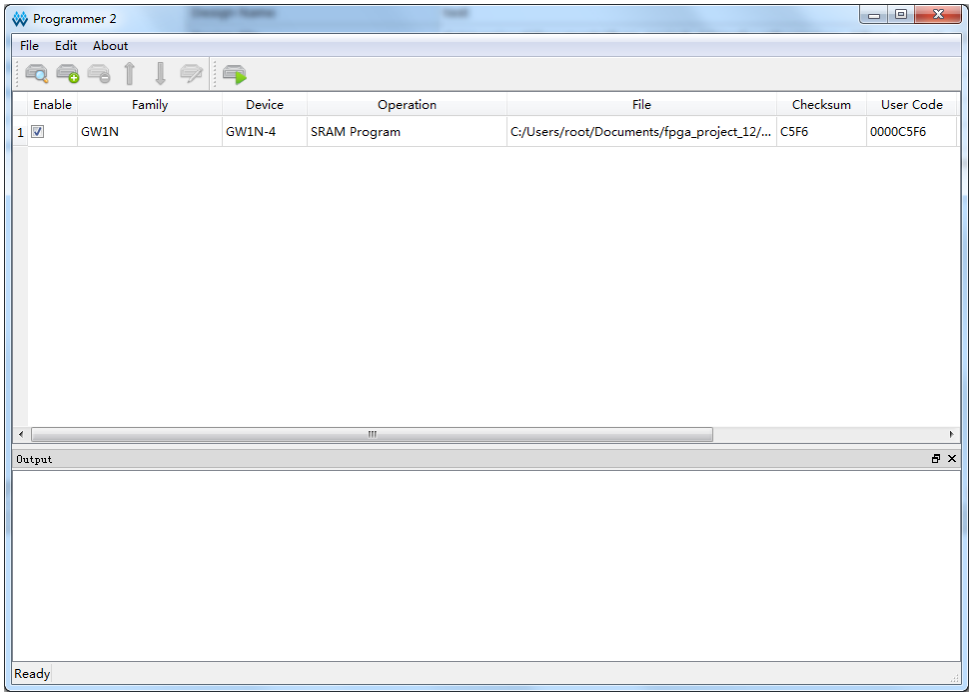
Refer to the following steps to download bitstream:

6. After running placement and routing, double click “Program Device” or right-click "run" in the “Process” view to open the Gowin Programmer, as shown in Figure C-3. The Gowin Programmer will automatically load bitstream files to the downloader.
7. Click “  ” to download bitstream.

Note!

- You can also use the two methods described below to download bitstream files:
- Use short cut F5.

Figure C-3 Programmer




GOWIN
PROGRAMMING FOR FUTURE