Gowin_EMPU_M3 Hardware Design
Reference Manual

IPUG923-1.2E, 11/14/2022

Copyright © 2022 Guangdong Gowin Semiconductor Corporation. All Rights Reserved.

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

Revision History

Date	Version	Description
04/03/2020	1.0E	Initial version published.
07/16/2021	1.1E	● Known issues of read and write for SPI full-duplex fixed. ● The version of FPGA software updated.
11/14/2022	1.2E	SD-Card removed.

Contents

List of Figures

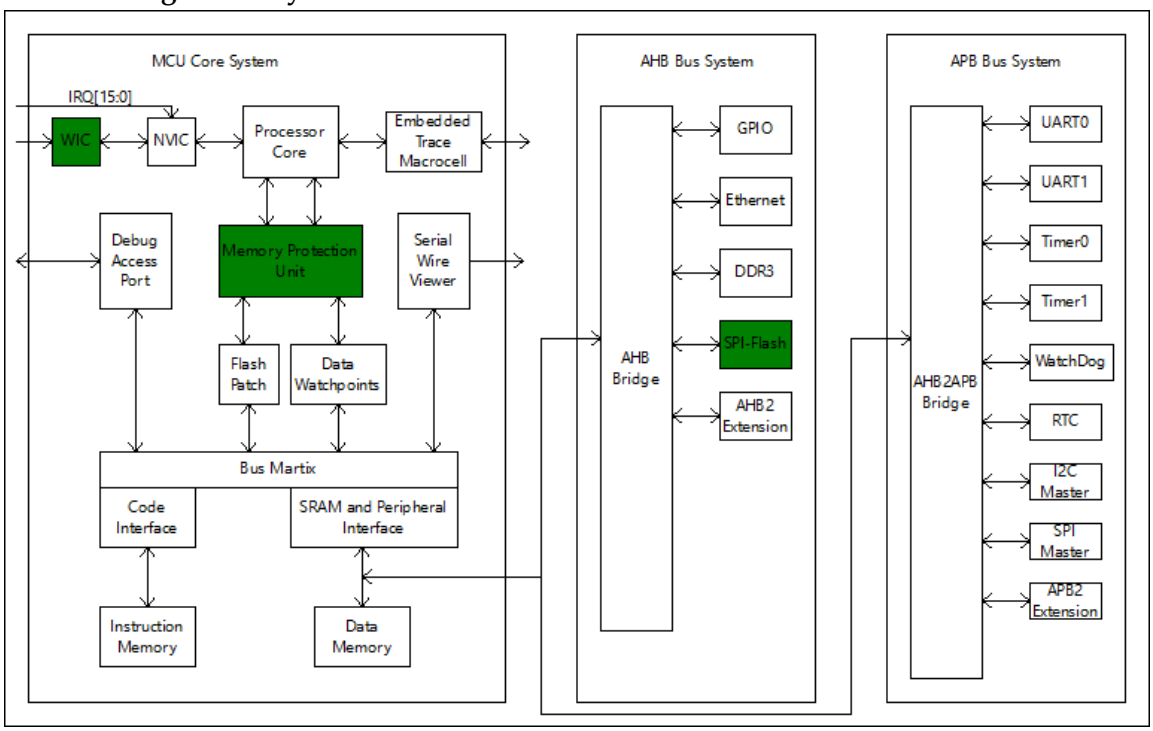
List of Tables

1 Hardware Architecture

1.1 System Architecture

Gowin_EMPU_M3 architecture consists of three levels, as shown in Figure 1-1.

Figure 1-1 System Architecture



The first level is MCU Core System, including ARM Cortex-M3 Processor Core, Instruction Memory, Data Memory and Peripheral Interface.

The second level is AHB Bus System, including GPIO, Ethernet, DDR3 Memory, SPI-Flash and AHB2 Extension.

The third level is APB Bus System, including UART0, UART1, Timer0, Timer1, WatchDog, RTC, I2C Master, SPI Master, and APB2 Extension.

1.2 System Feature

Gowin_EMPU_M3 consists of three sub-systems:

- Sub-system of MCU Core System
- Sub-system of AHB Bus System
- Sub-system of APB Bus System

1.2.1 Sub-system of MCU Core System

Processor Core

- ARM architecture v7-M Thumb2 supporting 16-bit and 32-bit instruction set
- Supports MPU (Memory Protection Unit)
- Supports WIC (Wakeup interrupt Controller)
- Supports Bit-banding
- Supports sleep mode to reduce system power consumption.
- Interrupt exception handling and normal thread model
- Support data and instruction Big/Little-endian format
- The system clock frequency is recommended below 40MHz.

NVIC（Nested Vector Interrupt Controller）

- Provides 16 user interrupt handling signals for users to extend interrupt handling function;
- Supports 256 interrupt priority levels;
- Saves processor status automatically during interrupts handling and recovers automatically at the end of interrupt handling.

Debugging System

Supports configurable debug levels:

- No debug
- Minimum debug
- Full debug minus DWT (Data Watchpoint and Trace)
- Full debug plus DWT

Supports configurable debug interface:

- Serial Wire
- JTAG and Serial Wire

Supports configurable Trace levels:

- No trace
- Standard trace. ITM (Instrumentation Trace Macrocell) and DWT (Data Watchpoint Trace), No ETM (Embedded Trace Macrocell)

- Full trace. ITM, DWT and ETM
- High trace. ITM, DWT, ETM and HTM（AHB Trace Macrocell）

Memory

- The Instruction Memory Size can be configured to 16KB, 32KB, 64KB or 128KB and bootload initial value file load is supportted;
- The Data Memory size can be configured to 16KB, 32KB, 64KB, or 128KB.

1.2.2 Sub-system of AHB Bus System

Peripherals of AHB bus interface:

- GPIO
- Ethernet
- DDR3 Memory
- SPI-Flash
- AHB2 Extension

1.2.3 Sub-system of APB Bus System

Peripherals of APB bus interface:

- UART0/1
- Timer0/1
- WatchDog
- RTC
- I2C Master
- SPI Master
- APB2 Extension

1.3 System Ports

The definition of Gowin_EMPU_M3 Ports is as shown in Table 1-1.

Table 1-1 Definition of System Ports

Name	I/O	Data Width	Description	Module
HCLK	in	1	System Clock	–
hwRstn	in	1	System Reset	–
LOCKUP	out	1	Core lockup state	–
HALTED	out	1	Core halt debug state	–
JTAG_3	inout	1	Debug Interface TRSTN	DAP
JTAG_4	inout	1	Debug Interface NC	
JTAG_5	inout	1	Debug Interface TDI	
JTAG_6	inout	1	Debug Interface NC	
JTAG_7	inout	1	Debug Interface TMS/SWDIO	

Name	I/O	Data Width	Description	Module
JTAG_8	inout	1	Debug Interface HOLDN	
JTAG_9	inout	1	Debug Interface TCK/SWCLK	
JTAG_10	inout	1	Debug Interface SIO1	
JTAG_11	inout	1	Debug Interface NC	
JTAG_12	inout	1	Debug Interface SCLK	
JTAG_13	inout	1	Debug Interface TDO/SWO	
JTAG_14	inout	1	Debug Interface SIO0	
JTAG_15	inout	1	Debug Interface SRSTN	
JTAG_16	inout	1	Debug Interface CS	
JTAG_17	inout	1	Debug Interface NC	
JTAG_18	inout	1	Debug Interface WPN	
IRQ	in	[15:0]	User Interrupt handling signal	NVIC
GPIO	inout	[15:0]	GPIO port	GPIO
UART0RXD	in	1	UART0 receive port	UART0
UART0TXD	out	1	UART0 transmit port	
UART1RXD	int	1	UART1 receive port	UART1
UART1TXD	out	1	UART1 transmit port	
TIMER0EXTIN	in	1	Timer0 external interrupt	Timer0
TIMER1EXTIN	in	1	Timer1 external interrupt	Timer1
RTCSRCCLK	in	1	RTC clock source: 32.768KHz	RTC
SCL	inout	1	Serial Clock	I2C Master
SDA	inout	1	Serial data	
MOSI	out	1	Master output/Slave input	SPI Master
MISO	in	1	Master input/Slave output	
SCLK	out	1	Clock signal	
NSS	out	1	Slave select signal	
RGMII_TXC	out	1	RGMII transmitting clock	Ethernet RGMII Interface
RGMII_TX_CTL	out	1	RGMII transmitting control	
RGMII_TXD	out	[3:0]	RGMII transmitting data	
RGMII_TXD	in	1	RGMII receiving clock	
RGMII_RX_CTL	in	10	RGMII receiving control	
RGMII_RXD	in	[3:0]	RGMII/MII receiving data	
GTX_CLK	in	1	RGMII 125MHz clock input	
GMII_RX_CLK	in	1	GMII receiving clock	Ethernet GMII Interface
GMII_RX_DV	in	1	GMII receiving enable	
GMII_RXD	in	[7:0]	GMII receiving data	
GMII_RX_ER	in	1	GMII receiving error	

Name	I/O	Data Width	Description	Module
GTX_CLK	in	1	GMII 125MHz clock input	
GMII_GTX_CLK	out	1	GMII receiving clock	
GMII_TXD	out	[7:0]	GMII transmitting data	
GMII_TX_EN	out	1	GMII transmitting enable	
GMII_TX_ER	out	1	GMII transmitting error	
MII_RX_CLK	in	1	MII receiving clock	Ethernet MII Interface
MII_RXD	in	[3:0]	MII receiving data	
MII_RX_DV	in	1	MII receiving enable	
MII_RX_ER	in	1	MII receiving error	
MII_TX_CLK	in	1	MII transmitting clock	
MII_TXD	out	[3:0]	MII transmitting data	
MII_TX_EN	out	1	MII transmitting enable	
MII_TX_ER	out	1	MII transmitting error	
MII_COL	in	1	MII conflicting signal	
MII_CRS	in	1	MII carrier signal	
MDC	out	1	Manage channel clock	Ethernet
MDIO	inout	1	Manage channel data	
DDR_CLK_I	in	1	50MHz clock input	DDR3
DDR_INIT_COMPLETE_O	out	1	Initialization completed signal	
DDR_ADDR_O	out	[15:0]	Row address, Column address	
DDR_BA_O	out	[2:0]	Bank address	
DDR_CS_N_O	out	1	Chip select signal	
DDR_RAS_N_O	out	1	Row address strobe signal	
DDR_CAS_N_O	out	1	Column address strobe signal	
DDR_WE_N_O	out	1	Row write enable	
DDR_CLK_O	out	1	A clock signal provided to DDR3 SDRAM	
DDR_CLK_N_O	out	1	Compose a differential signal with DDR_CLK_O	
DDR_CKE_O	out	1	DDR3 SDRAM clock enable signal	
DDR_ODT_O	out	1	Terminating resistor control of memory signal	
DDR_RESET_N_O	out	1	DDR3 SDRAM reset signal	
DDR_DQM_O	out	[1:0]	DDR3 SDRAM data masking signal	
DDR_DQ_IO	inout	[15:0]	DDR3 SDRAM data	

Name	I/O	Data Width	Description	Module
DDR_DQS_IO	inout	[1:0]	DDR3 SDRAM data strobe signal	
DDR_DQS_N_IO	inout	[1:0]	Compose a differential signal with DDR_DQS_IO	
FLASH_SPI_HOLDN	inout	1	NC	SPI-Flash
FLASH_SPI_CSN	inout	1	Slave select signal	
FLASH_SPI_MISO	inout	1	Master input/Slave output	
FLASH_SPI_MOSI	inout	1	Master output/Slave input	
FLASH_SPI_WPN	inout	1	NC	
FLASH_SPI_CLK	inout	1	Clock signal	
APB2PSTRB	out	[3:0]	APB2 PSTRB	APB2 Extension
APB2PPROT	out	[2:0]	APB2 PPROT	
APB2PSEL	out	1	APB2 PSEL	
APB2PENABLE	out	1	APB2 PENABLE	
APB2PADDR	out	[31:0]	APB2 PADDR	
APB2PWRITE	out	1	APB2 PWRITE	
APB2PWDATA	out	[31:0]	APB2 PWDATA	
APB2PRDATA	in	[31:0]	APB2 PRDATA	
APB2PREADY	in	1	APB2 PREADY	
APB2PSLVERR	in	1	APB2 PSLVERR	
APB2PCLK	out	1	APB2 PCLK	
APB2PRESET	out	1	APB2 RESET	
AHB2HSEL	out	1	AHB2 HSEL	AHB2 Extension
AHB2HADDR	out	[31:0]	AHB2 HADDR	
AHB2HTRANS	out	[1:0]	AHB2 HTRANS	
AHB2HWRITE	out	1	AHB2 HWRITE	
AHB2HSIZE	out	[2:0]	AHB2 HSIZE	
AHB2HBURST	out	[2:0]	AHB2 HBURST	
AHB2HPROT	out	[3:0]	AHB2 HPROT	
AHB2HWDATA	out	[31:0]	AHB2 HWDATA	
AHB2HMASTLOCK	out	1	AHB2 HMASTLOCK	
AHB2HREADYMUX	out	1	AHB2 HREADYMUX	
AHB2HRDATA	in	[31:0]	AHB2 HRDATA	

Name	I/O	Data Width	Description	Module
AHB2HREADYOUT	in	1	AHB2 HREDAYOUT	
AHB2HRESP	in	[1:0]	AHB2 HRESP	
AHB2HMASTER	out	[3:0]	AHB2 MASTER	
AHB2HCLK	out	1	AHB2 HCLK	
AHB2HRESET	out	1	AHB2 RESET	

1.4 System Resource Statistics

The system resource statistics of Gowin_EMPU_M3 is as shown in Table 1-2.

Table 1-2 System Resource Statistics

Configuration \ Resources	LUTs	Registers	BSRAMs	DSP Macros
MCU Core System Minimum and No Peripherals	17138	2245	16	2
MCU Core System Default and No Peripherals	22732	4634	64	2
MCU Core System Maximum and No Peripherals	25290	5571	128	2
MCU Core System and All Peripherals Maximum	37345	12981	111	2

2 Hardware Design Flow

2.1 Hardware Environment

DK-START-GW2A55 V1.3: GW2A-LV55PG484C8/I7

2.2 Software Environment

Gowin_V1.9.8 Beta and above

2.3 IP Core Generator

Configure and generate Gowin_EMPU_M3 hardware design in IP Core Generator of Gowin software.

2.4 Programmer

Use Gowin Programmer to download hardware bitstream files and software programming BIN files in binary format.

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

2.5 Design Flow

Gowin_EMPU_M3 hardware design flow is as follows:

1. Configure MCU Core System, AHB Bus System and APB Bus System in IP Core Generator to generate Gowin_EMPU_M3 hardware design. Import to Gowin_EMPU_M3 project;
2. Instantiate Top Module, import user designs, and connect user designs with Top Module;
3. Add physical and timing Constraints;
4. Use GowinSynthesis to synthesis and generate the netlist file;
5. Use Place & Route to generate bitstream;
6. Use Programmer to download bitstream files to GW2A/-55/7.5 (GW2AN-55C).

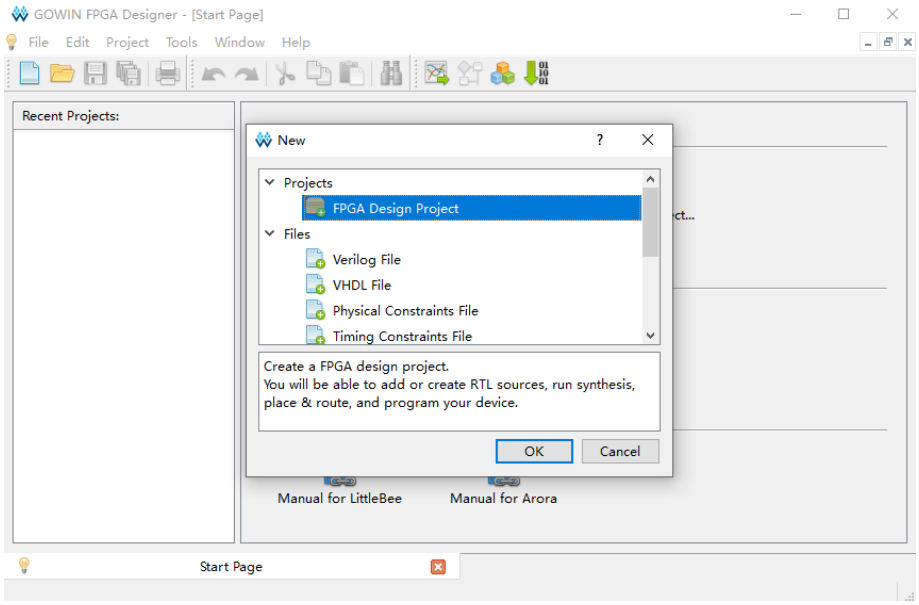
3 Project Template

3.1 Create a New Project

3.1.1 Create a New Project

Run Gowin software. Click "File > New… > FPGA Design Project" on the menu bar, as shown in Figure 3-1.

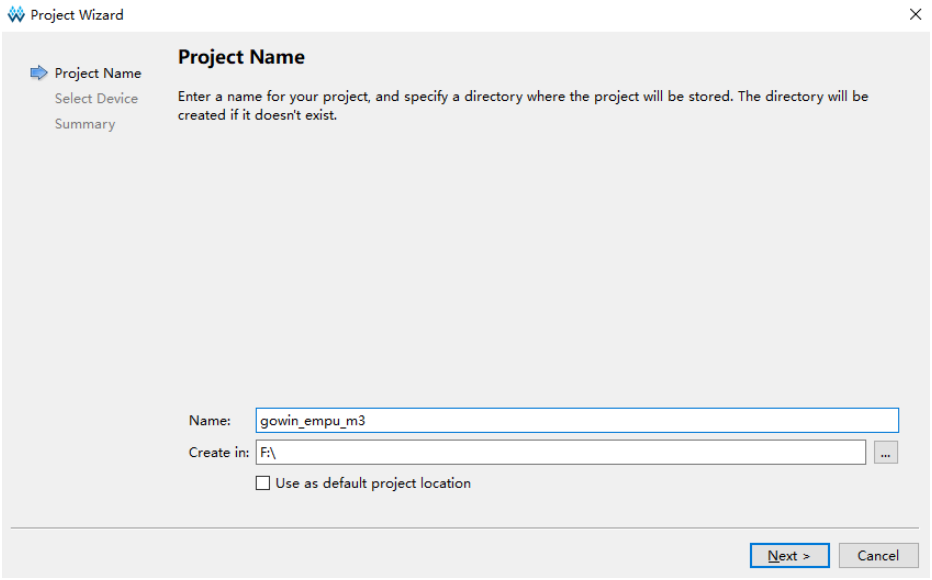
Figure 3-1 Create a FPGA Design Project



3.1.2 Set Project Name and Path

Enter the project name and select the project path, as shown in Figure 3-2.

Figure 3-2 Set Project Name and Path



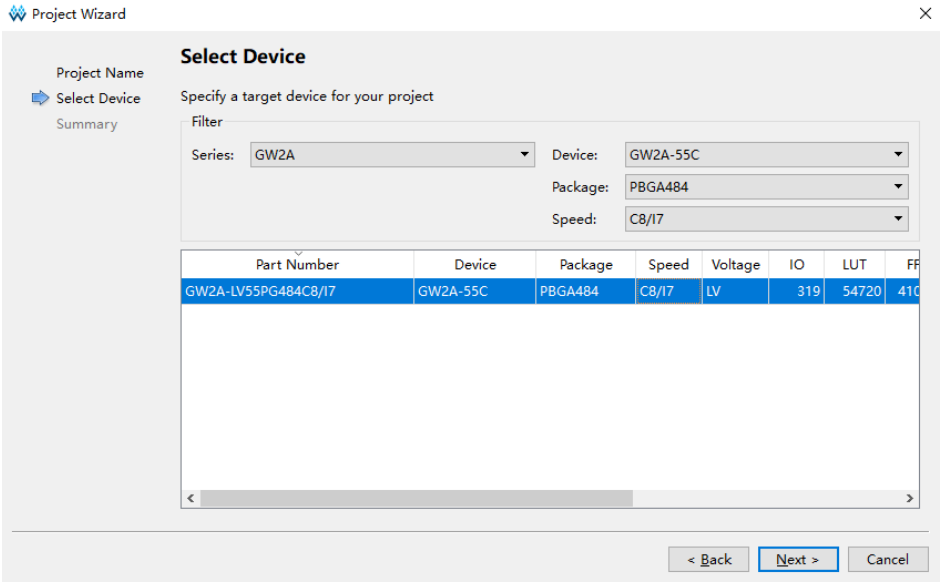
3.1.3 Select Device

Select "Series", "Device", "Package", "Speed" and "Part Number", as shown in Figure 3-3.

Take reference design in SDK for an instance:

- Series: GW2A
- Device: GW2A-55C
- Package: PBGA484
- Speed: C8/I7
- Part Number: GW2A-LV55PG484C8/I7

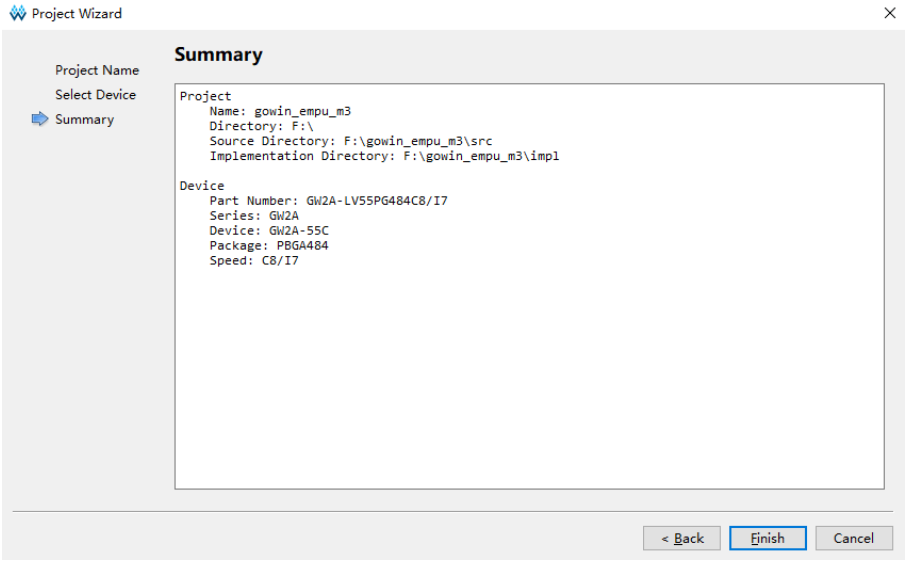
Figure 3-3 Select Device



3.1.4 Project Creation Completed

The project creation is completed as shown in Figure 3-4.

Figure 3-4 Complete Project Creating



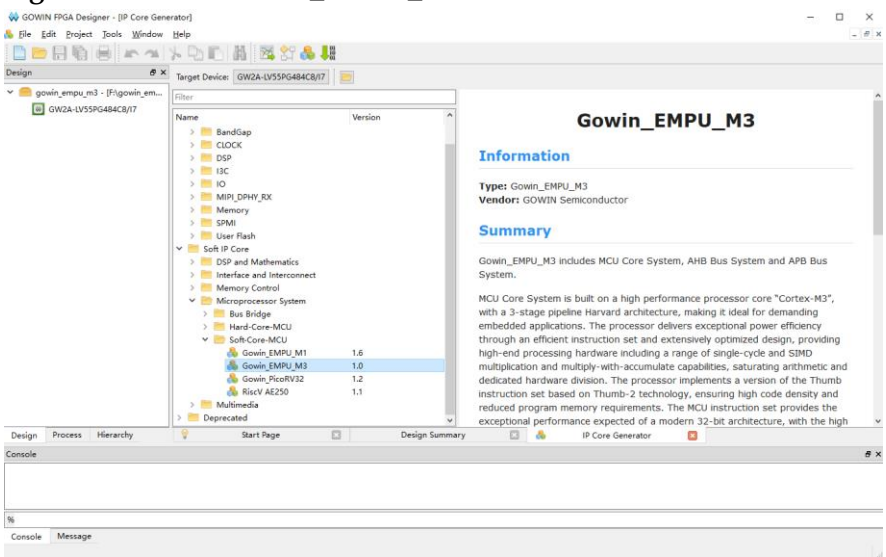
3.2 Hardware Design

Use IP Core Generator to generate Gowin_EMPU_M3 hardware designs.

Select "Tools > IP Core Generator" in the menu bar or "  " to open the IP Core Generator.

Select "Soft IP Core > Microprocessor System > Soft-Core-MCU > Gowin_EMPU_M3 1.0", as shown in Figure 3-5.

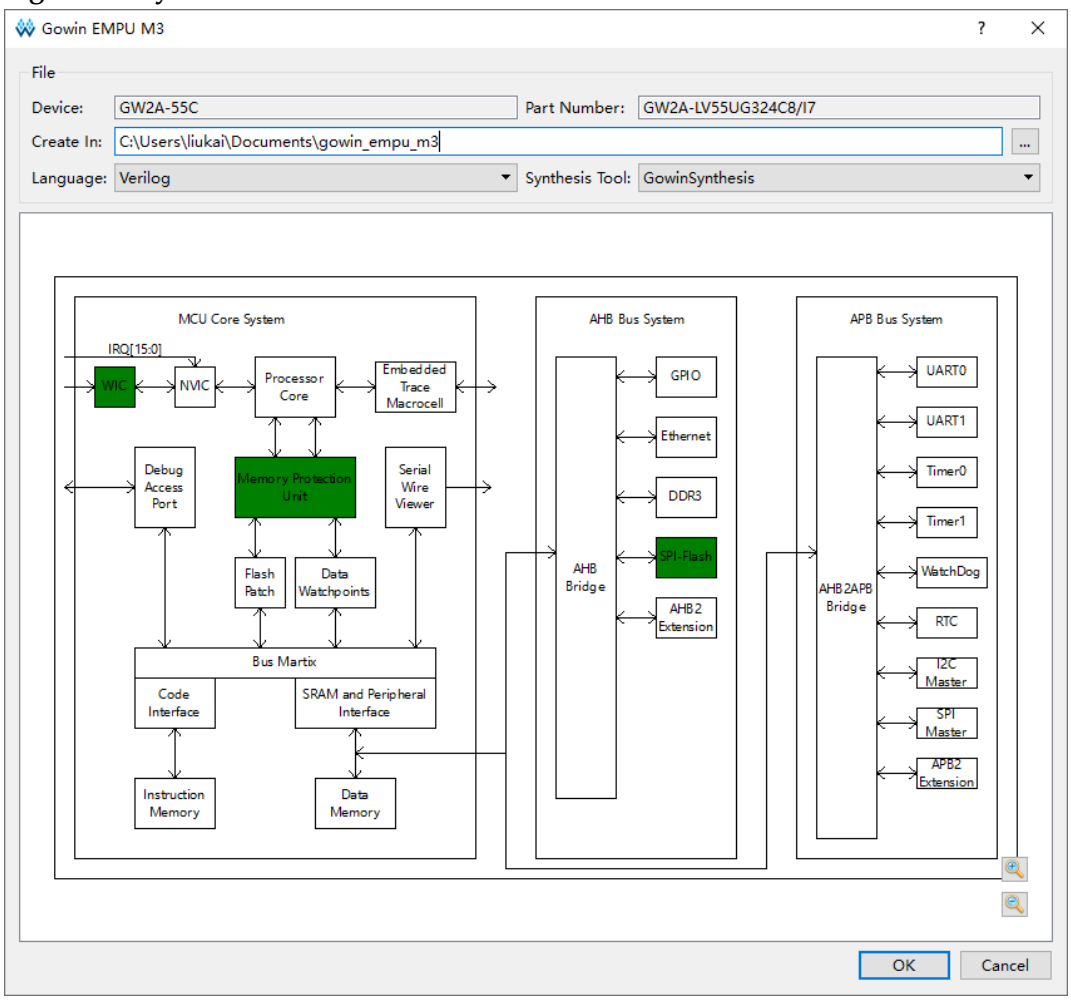
Figure 3-5 Select Gowin_EMPU_M3



The system architecture of Gowin_EMPU_M3 is as shown in Figure 3-6.

It consists of three sub-systems: MCU Core System, AHB Bus System and APB Bus System.

Figure 3-6 System Architecture



3.2.1 MCU Core System Configuration

MCU Core System configuration options are as shown in Table 3-1.

Table 3-1 MCU Core System Configuration Options

Options	Description
Enable User Interrupts	Enable16 user interrupt handling signals; Disable by default
Enable MPU	Enable Memory Protection Unit; Enable by default
Enable WIC	Enable Wakeup Interrupt Controller; Enable by default
Enable Bit-banding	Enable Bit-banding; Enable by default
IRQ Priority Level Width	Configure interrupt priority bit width with value range from 3 to 8; The default value is 3.
WIC Lines	Configure Wakeup Interrupt Controller Lines with value range from 3 to 243;

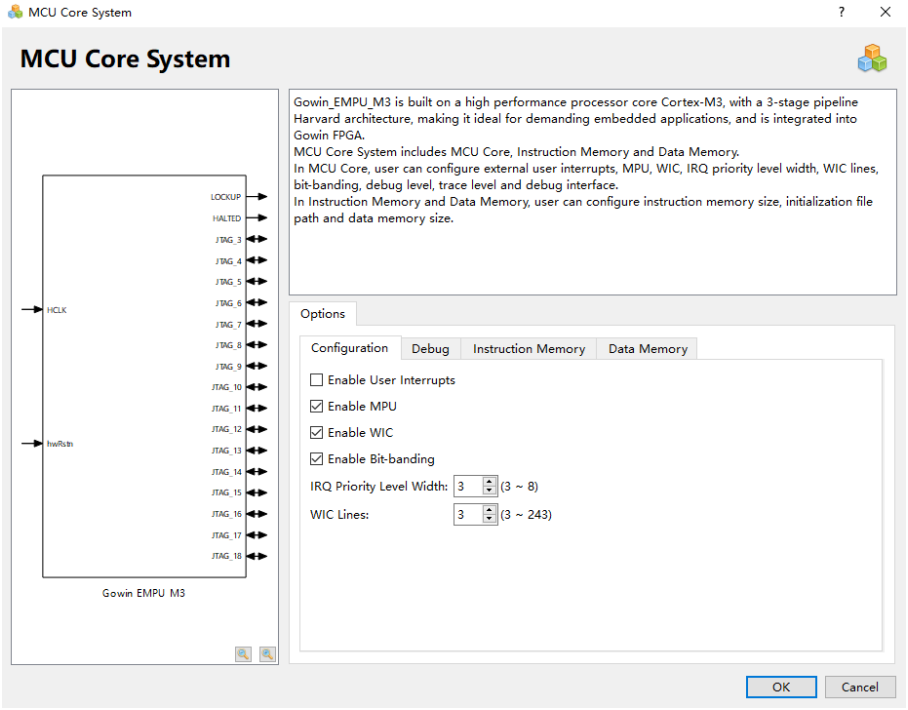
Options	Description
	The default value is 3.
Debug Level	Configure debug system level No debug Minimum debug Full debug minus DWT Full debug plus DWT Full debug plus DWT by default
Trace Level	Configure trace system level No trace Standard trace. ITM and DWT, No ETM Full trace. ITM, DWT and ETM High trace. ITM, DWT, ETM and HTM Standard trace. ITM and DWT, No ETM by default
Debug Interface	Configure Debug Interface Serial Wire JTAG and serial wire JTAG and serial wire by default
Instruction Memory Size	16KB, 32KB, 64KB or 128KB; 64KB by default
Initialization File Path	Initialization file path of bootload for off-chip SPI-Flash download and boot
Data Memory Size	16KB, 32KB, 64KB or 128KB; 64KB by default

The configuration options of MCU Core System include Configuration, Debug, Instruction Memory and Data Memory.

Configuration

The Configuration options include User Interrupts, MPU, WIC, Bit-banding, IRQ Priority Level Width and WIC Lines, as shown in Figure 3-7.

Figure 3-7 Configuration



- User Interrupts Configuration
  - Provides 16 user interrupt handling signals for users to extend interrupt handling function;
  - If you select Enable User Interrupts, Gowin_EMPU_M3 supports user interrupt handling, otherwise it does not.
  - Disable by default
- MPU Configuration
  - If Enable MPU is selected, Gowin_EMPU_M3 supports MPU, otherwise it does not.
  - Disable by default
- WIC Configuration
  - If Enable WIC is selected, Gowin_EMPU_M3 supports WIC, otherwise it does not.
  - Disable by default
- Bit-banding Configuration
  - If Enable Bit-banding is selected, Gowin_EMPU_M3 supports Bit-banding, otherwise it does not.
  - Disable by default
- IRQ Priority Level Width Configuration
  - Configure interrupt priority bit width with value range from 3 to 8. Interrupt priority levels from 8 to 256 can be supported.

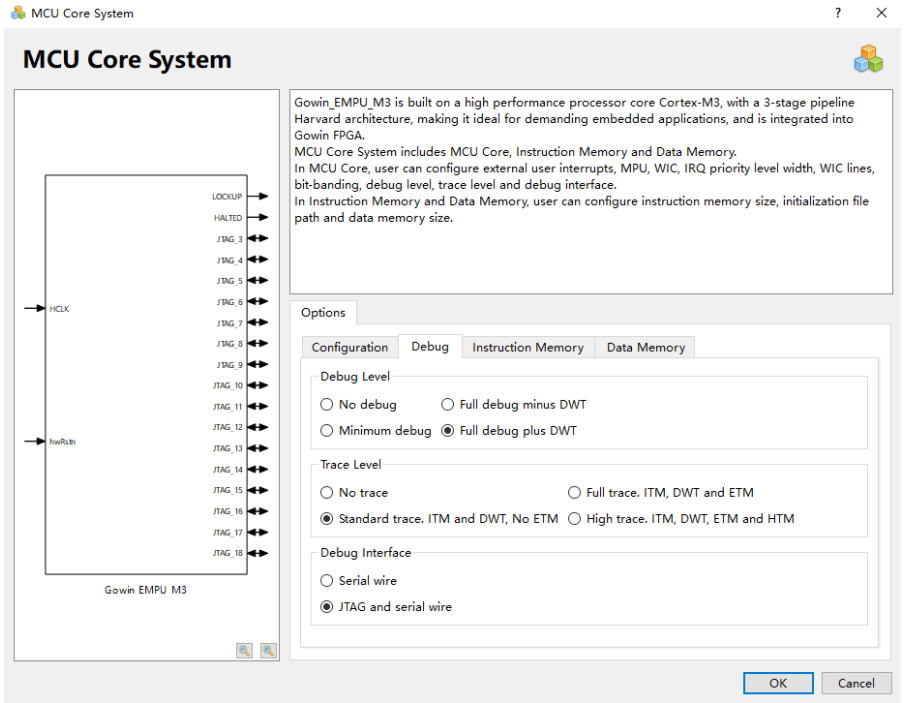
- The default value is 3.

- WIC Lines Configuration
  - Configure WIC Lines with value range from 3 to 243
  - The default value is 3.

Debug

Select "Debug", as shown in Figure 3-8, you can configure Debug Level, TraceLevel and Debug Interface.

Figure 3-8 Debug



- Debug Level Configuration

  Configure debug system level:
  - No debug
  - Minumum debug
  - Full debug minus DWT
  - Full debug plus DWT

  Note!

  Full debug plus DWT by default

- Trace Level Configuration

  Configure trace system level:
  - No trace
  - Standard trace. ITM and DWT, No ETM
  - Full trace. ITM, DWT and ETM

- High trace. ITM, DWT ETM and HTM

Note!

Standard trace. ITM and DWT, No ETM by default

- Debug Interface Configuration

  Configure Debug Interface

  - Serial Wire
  - JTAG and serial wire

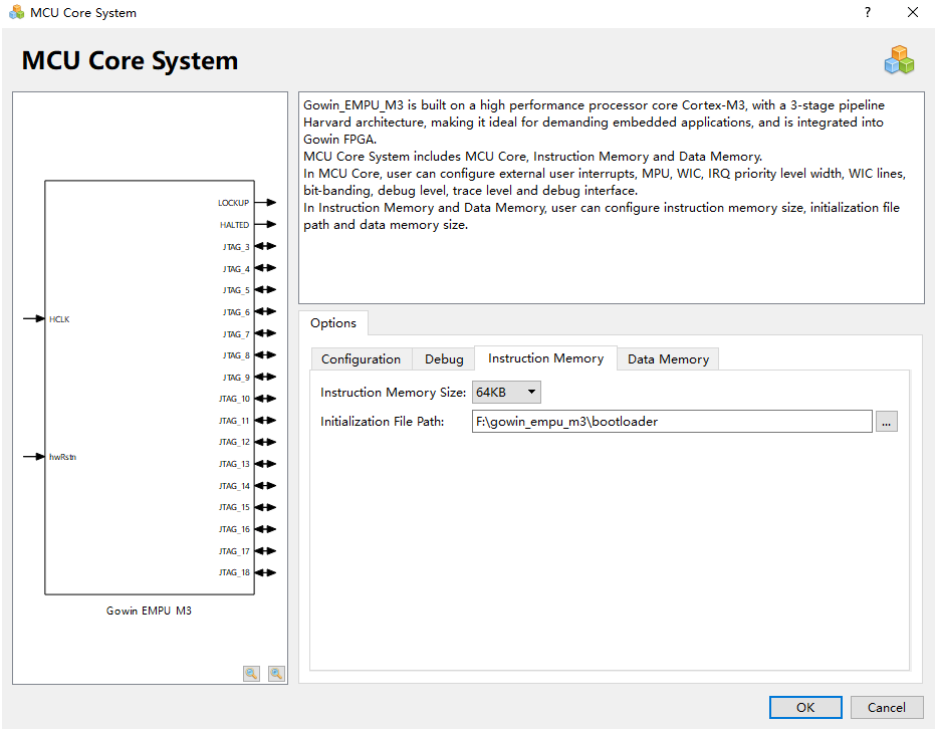
Note!

JTAG and serial wire by default

Instruction Memory

Select "Instruction Memory", as shown in Figure 3-9, you can configure Instruction Memory Size and Initialization File Path.

Figure 3-9 Instruction Memory



- Instruction Memory size can be configured to 16KB, 32KB, 64KB or 128KB. 64KB by default
- Configure initialization file path of bootload for off-chip SPI-Flash download and boot Different bootload file paths are loaded according to different Instruction Memory sizes, as shown below.
  - If the Instruction Memory Size is configured to 16KB, the Initialization File Path loads the INSMEM_Size_16KB bootload Initialization File of the software development kit;
  - If the Instruction Memory Size is configured to 32KB, the Initialization File Path loads the INSMEM_Size_32KB bootload

Initialization File of the software development kit;

- If the Instruction Memory Size is configured to 64KB, the Initialization File Path loads the INSMEM_Size_64KB bootload Initialization File of the software development kit;
- If the Instruction Memory Size is configured to 128KB, the Initialization File Path loads the INSMEM_Size_128KB bootload Initialization File of the software development kit;

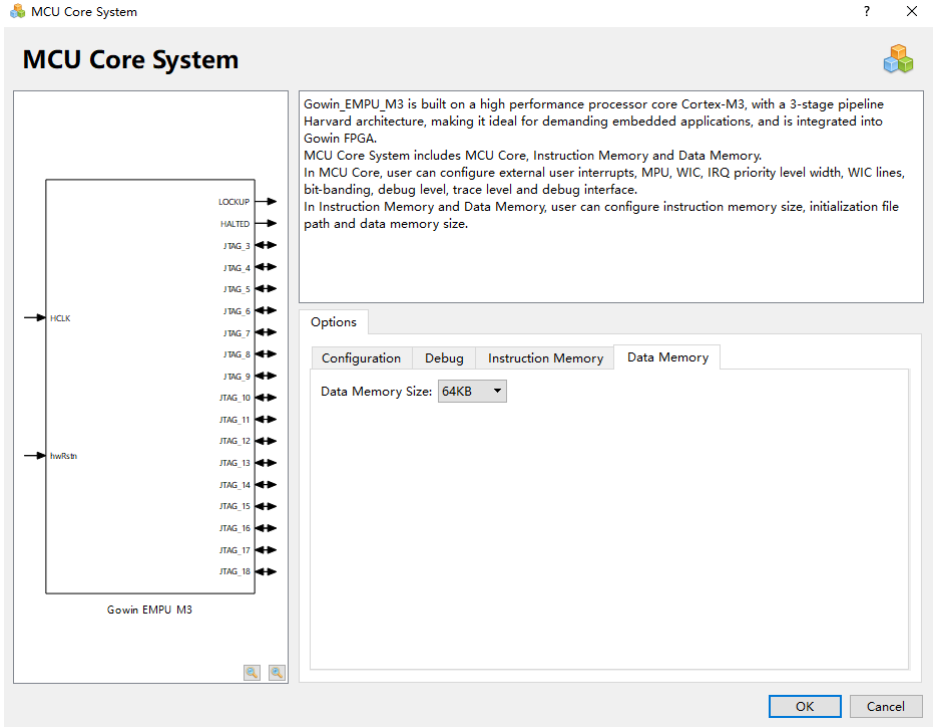
Note!

The file name of Instruction Memory Initialization path cannot contain numbers.

Data Memory

Select "Data Memory" to configure Data Memory Size, as shown in Figure 3-10.

Figure 3-10 Data Memory



Data Memory size can be configured to 16KB, 32KB, 64KB or 128KB, 64KB by default.

3.2.2 AHB Bus System Configuration

AHB Bus System configuration options are as shown in Table 3-2.

Table 3-2 AHB Bus System Configuration Options

Options	Description
Enable GPIO	Enable GPIO Disable by default
Enable Ethernet	Enable Ethernet Disable by default

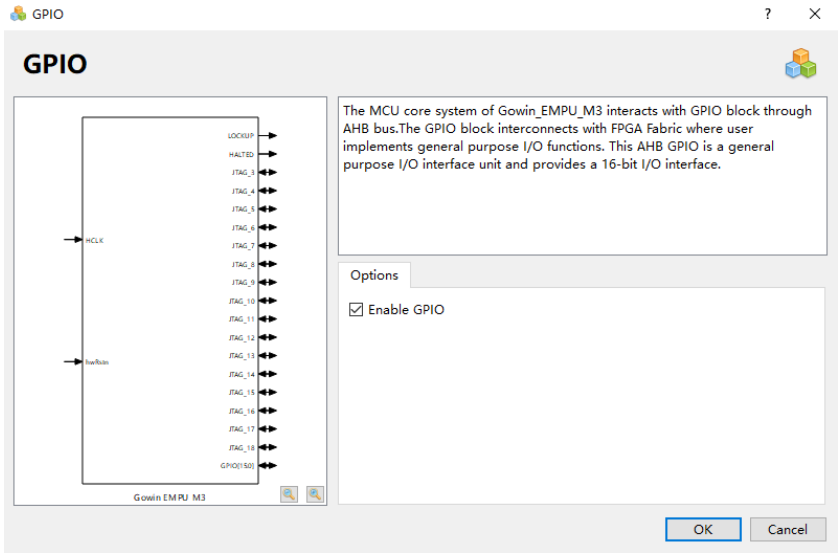
Options	Description
Interface	Ethernet selects Interface (RGMII/GMII/MII) RGMII by default
RGMII Input Delay	RGMII input delay The default Value is 100
MIIM Clock Divider	MIIM clock divider The default Value is 20
Enable DDR3	Enable DDR3 Memory Disable by default
Enable SPI-Flash	Enable SPI-Flash download function and read, write and erasure of Memory; Enable by default
Enable AHB2 Extension	Enable AHB2 Extension interface Disable by default

GPIO

Double click to configure GPIO, as shown in Figure 3-11.

If Enable GPIO is selected, Gowin_EMPU_M3 supports GPIO, disable by default.

Figure 3-11 GPIO Configuration



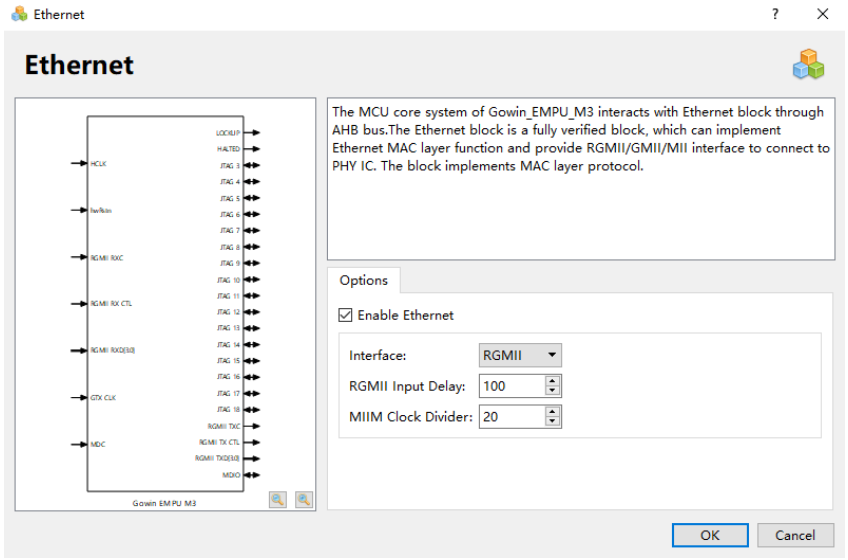
Ethernet

Double click to open Ethernet, as shown in Figure 3-12.

- If Enable Ethernet is selected, Gowin_EMPU_M3 supports Ethernet, disable by default.
- If Enable Ethernet has been selected, you can configure Interface, RGMII Input Delay, and MIIM Clock Divider.
  - Select Interface. Options are RGMII, GMII, MII, or GMII/MII, RGMII by default.

- If you select RGMII, you can configure RGMII Input Delay and the value is 100.
- If MIIM Clock Divider is selected, you can configure MIIM Clock Divider. The default value is 20.
- If RGMII or GMII is selected, 125MHz clock input must be provided to GTX_CLK.

Figure 3-12 Ethernet Configuration



PDM2PCM

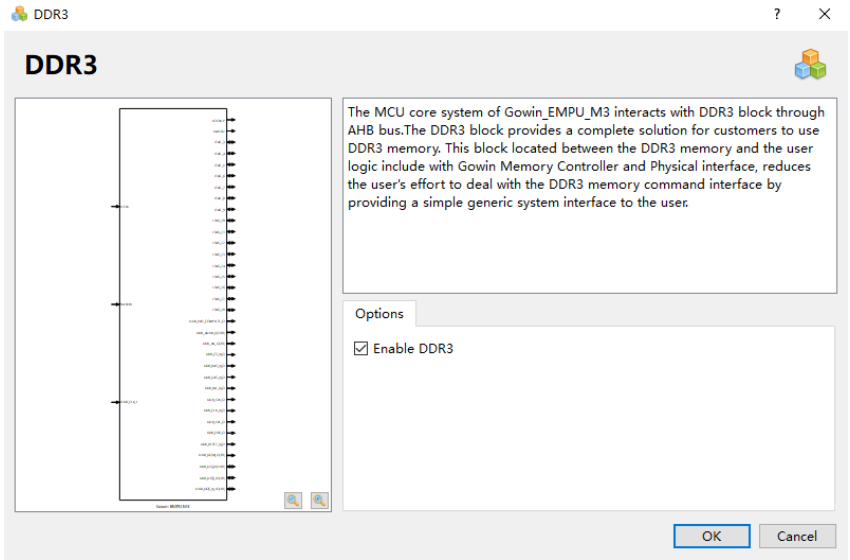
Double click to configure DDR3, as shown in Figure 3-13.

If Enable DDR3 is selected, Gowin_EMPU_M3 supports DDR3 Memory, disable by default.

The internal clock frequency of DDR3 is 150MHz.

50MHz clock input must be provided to DDR_CLK_I.

Figure 3-13 DDR3 Configuration



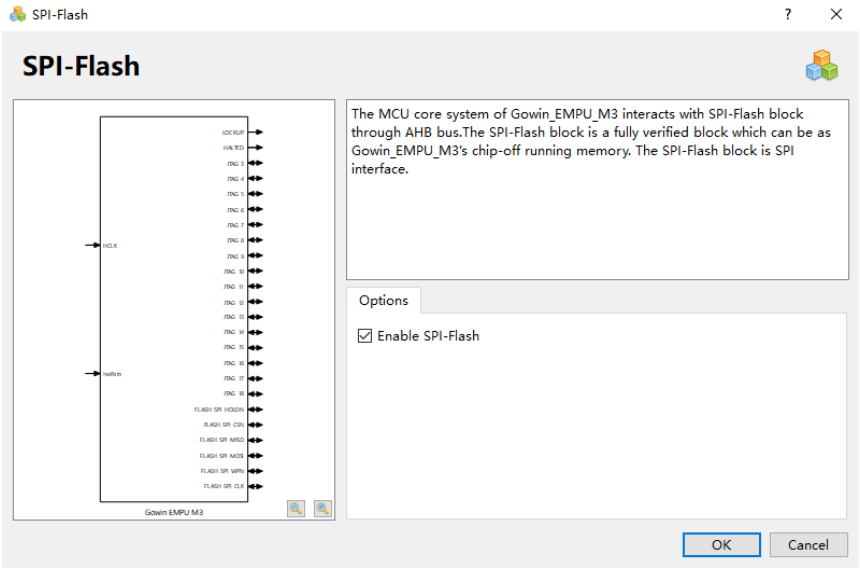
SPI-Flash

SPI-Flash supports download function and the read, write and erasure of Memory.

Double click to configure SPI-Flash, as shown in Figure 3-14.

If Enable SPI-Flash is selected, Gowin_EMPU_M3 supports SPI-Flash, disable by default.

Figure 3-14 SPI-Flash Configuration

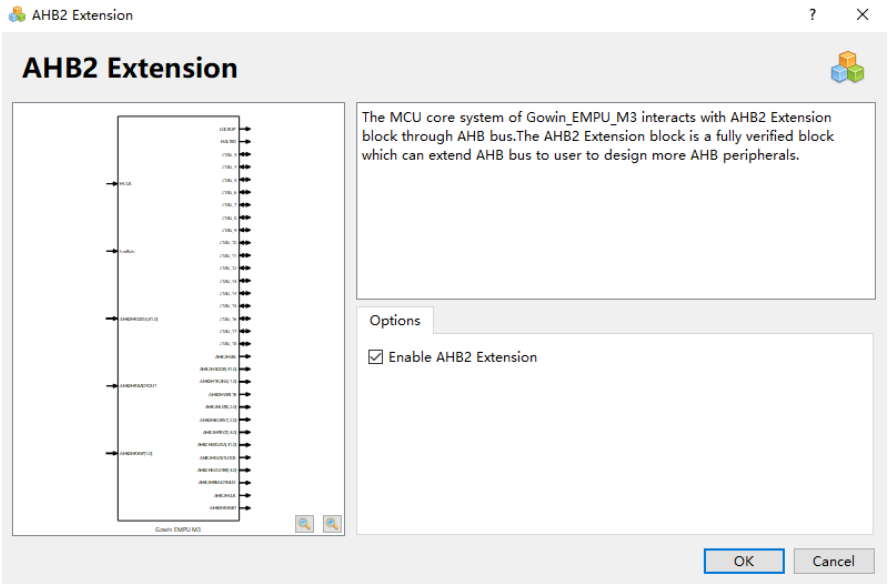


AHB2 Extension Configuration

Double click to open AHB2 Extension, as shown in Figure 3-15.

If Enable AHB2 Extension is selected, then Gowin_EMPU_M3 supports AHB2 Extension, and you can connect to the extended AHB peripherals, disable by default.

Figure 3-15 AHB2 Extension Configuration



3.2.3 APB Bus System Configuration

APB Bus System configuration options are as shown in Table 3-3.

Table 3-3 AHB Bus System Configuration Options

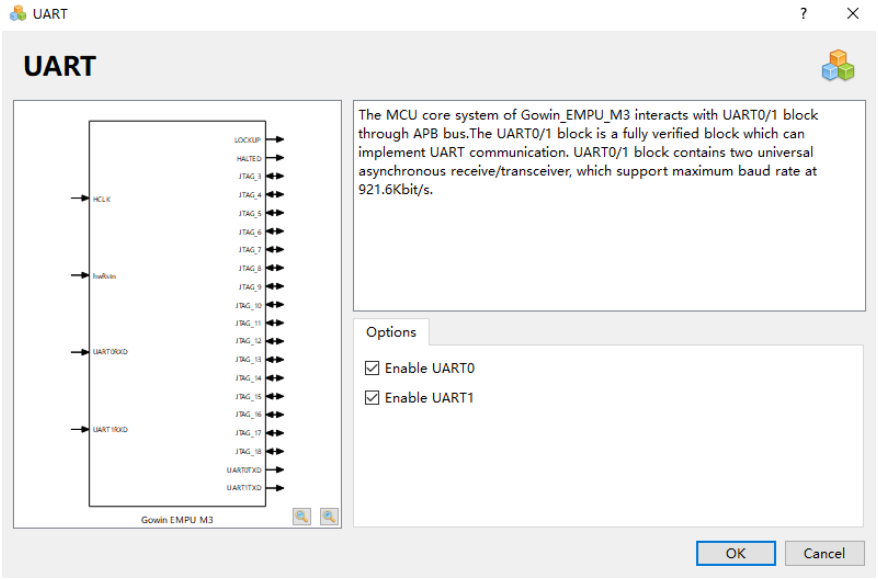
Options	Description
Enable UART0	Enable serial port0 Disable by default
Enable UART1	Enable serial port1 Disable by default
Enable Timer0	Enable Timer0 Disable by default
Enable Timer1	Enable Timer1 Disable by default
Enable WatchDog	Enable Watchdog Disable by default
Enable RTC	Enable RTC Disable by default
Enable I2C Master	Enable I2C Master Disable by default
Enable SPI Master	Enable SPI Master Disable by default
Enable APB2 Extension	Enable APB2 Extension interface Disable by default

UART

Double click UART0 or UART1, as shown in Figure 3-16.

- If Enable UART0 is selected, Gowin_EMPU_M3 supports UART0, disable by default.
- If Enable UART1 is selected, Gowin_EMPU_M3 supports UART1, disable by default.

Figure 3-16 UART Configuration

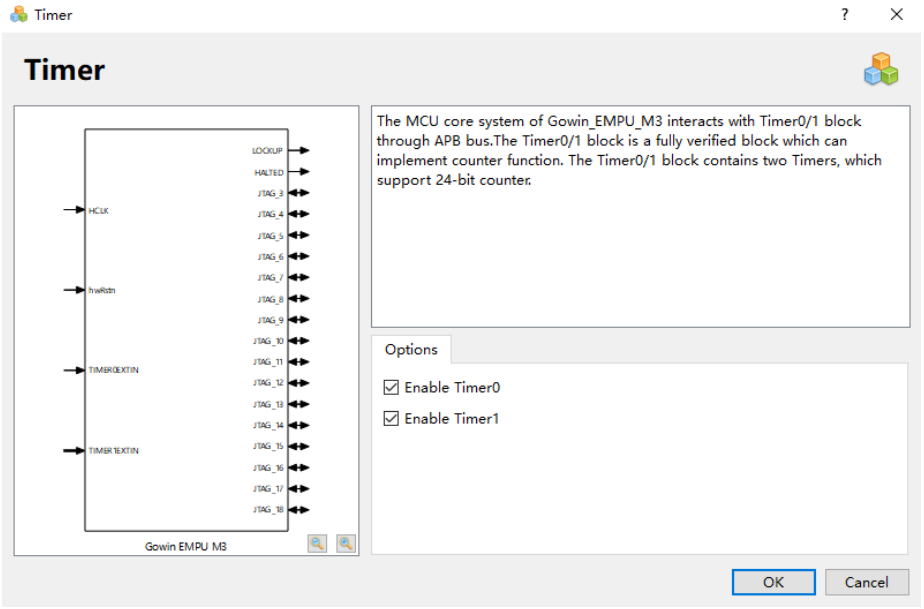


Timer

Double click to configure Timer0 or Timer1, as shown in Figure 3-17.

- If Enable Timer0 is selected, Gowin_EMPU_M3 supports Timer0, disable by default.
- If Enable Timer1 is selected, Gowin_EMPU_M3 supports Timer1, disable by default.

Figure 3-17 Timer Configuration

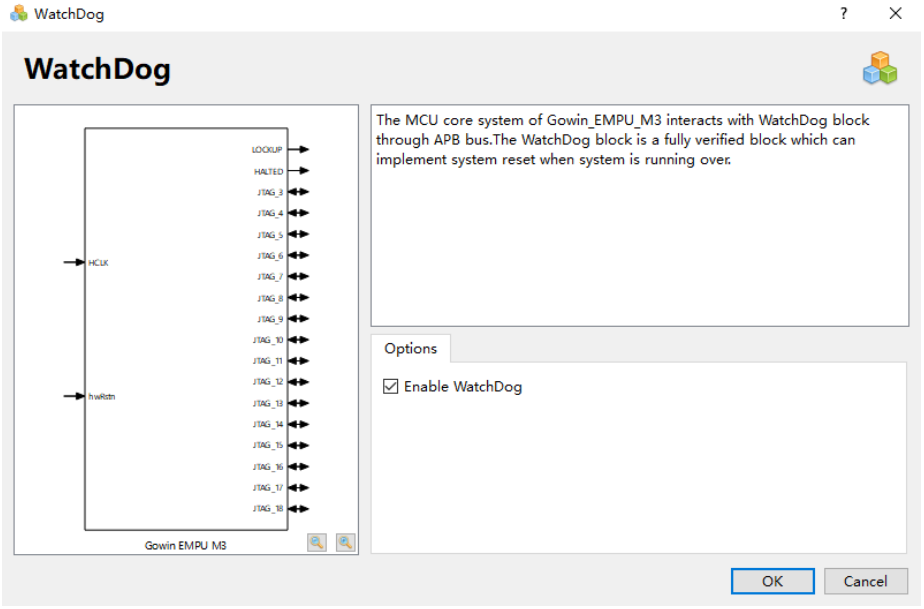


WatchDog

Double click to configure WatchDog, as shown in Figure 3-18.

If Enable WatchDog is selected, Gowin_EMPU_M3 supports WatchDog, disable by default.

Figure 3-18 WatchDog Configuration



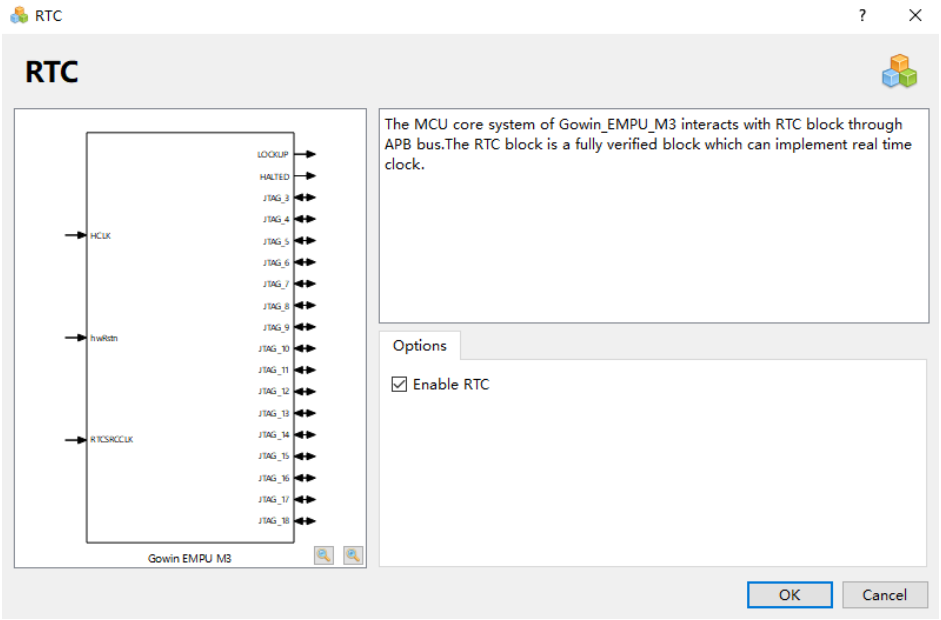
RTC

Double click to configure RTC, as shown in Figure 3-19.

If Enable RTC is selected, Gowin_EMPU_M3 supports RTC, disable by default.

3.072MHz clock input must be provided to RTCSRCCL. The division in RTC is 1Hz.

Figure 3-19 RTC Configuration



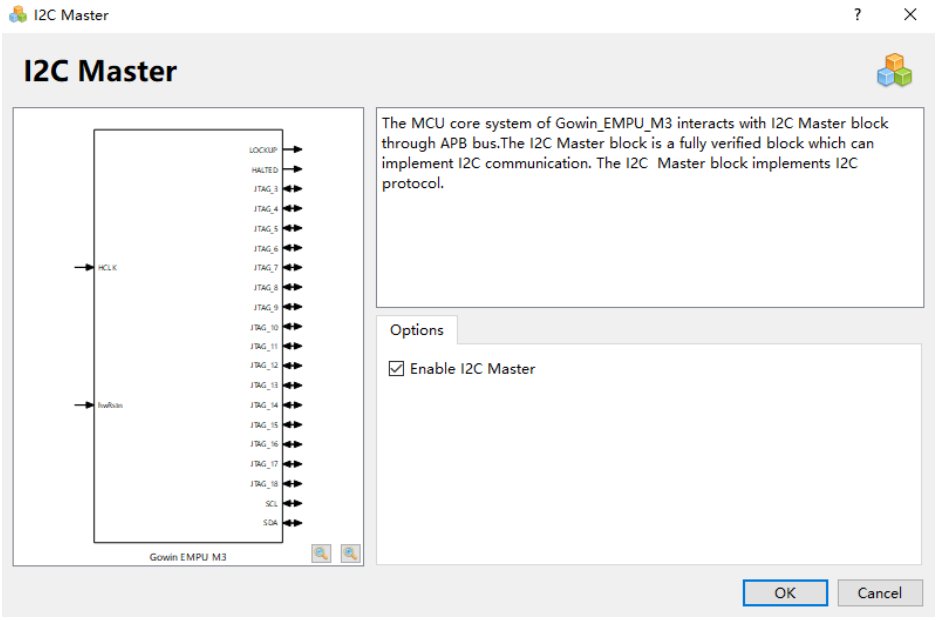
I2C Master Configuration

Double click to configure I2C Master, as shown in Figure 3-20.

If Enable I2C Master is selected, Gowin_EMPU_M3 supports I2C Master,

disable by default.

Figure 3-20 I2C Master Configuration

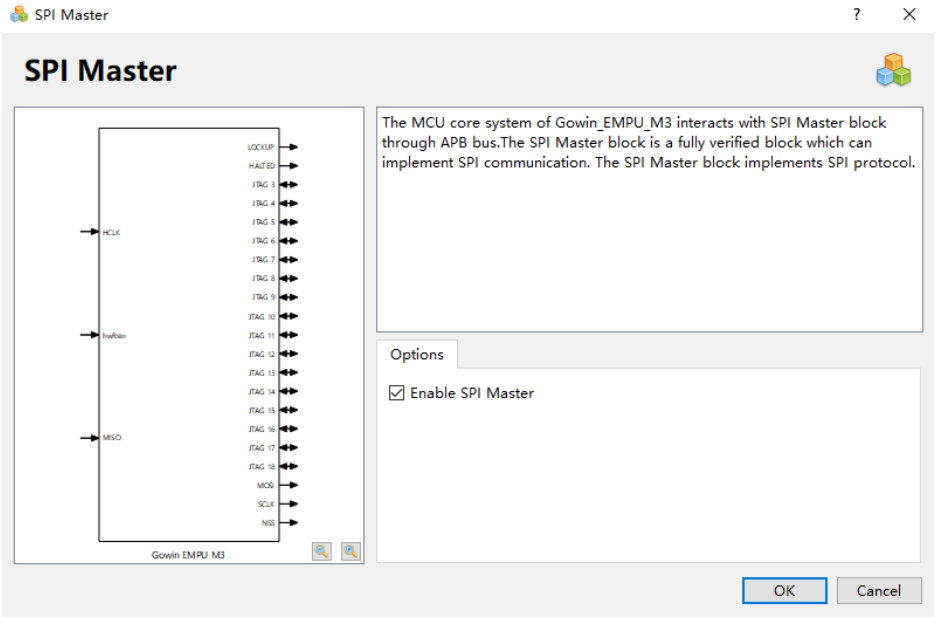


SPI Master Configuration

Double click to configure SPI Master, as shown in Figure 3-21.

If Enable SPI Master is selected, Gowin_EMPU_M3 supports SPI Master, disable by default.

Figure 3-21 SPI Master Configuration



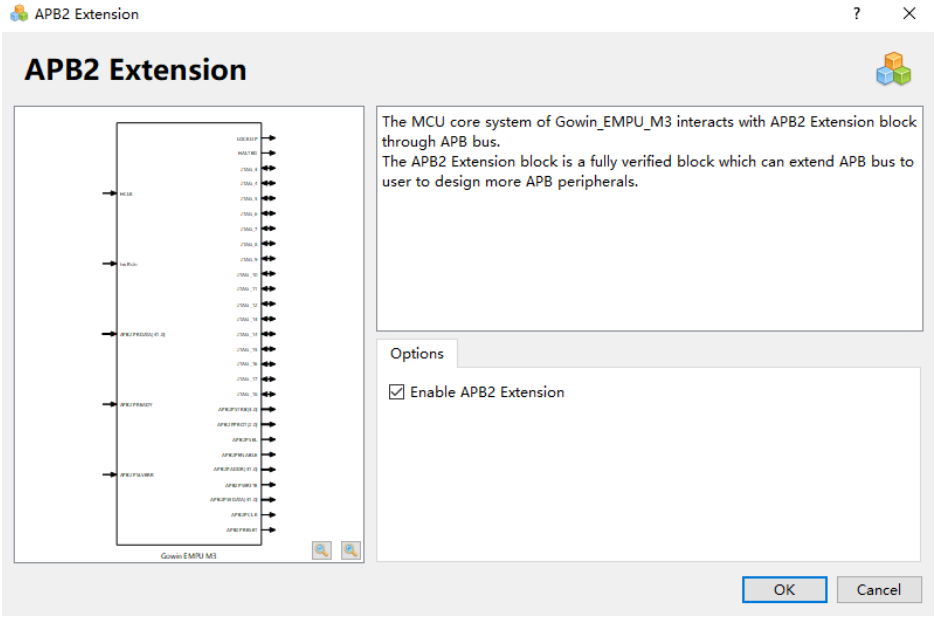
APB2 Extension Configuration

Double click to open APB2 Extension, as shown in Figure 3-22.

If Enable APB2 Extension is selected, then Gowin_EMPU_M3 supports

APB2 Extension, and you can connect to the extended APB peripherals, disable by default.

Figure 3-22 APB2 Extension Configuration



3.3 User Design

- After the configuration of Gowin_EMPU_M3, you can generate Gowin_EMPU_M3 hardware design.
- Instantiate Gowin_EMPU_M3 Top Module.
- Import user designs and connect it with Gowin_EMPU_M3 to form a complete RTL design

3.4 Constraint

After RTL design is completed, physical constraints can be generated according to the used development board and the IO.

Timing constraints file can be generated according to timing requirements.

For how to generate physical constraints, please refer to *SUG101, Gowin Design Constraints Guide*.

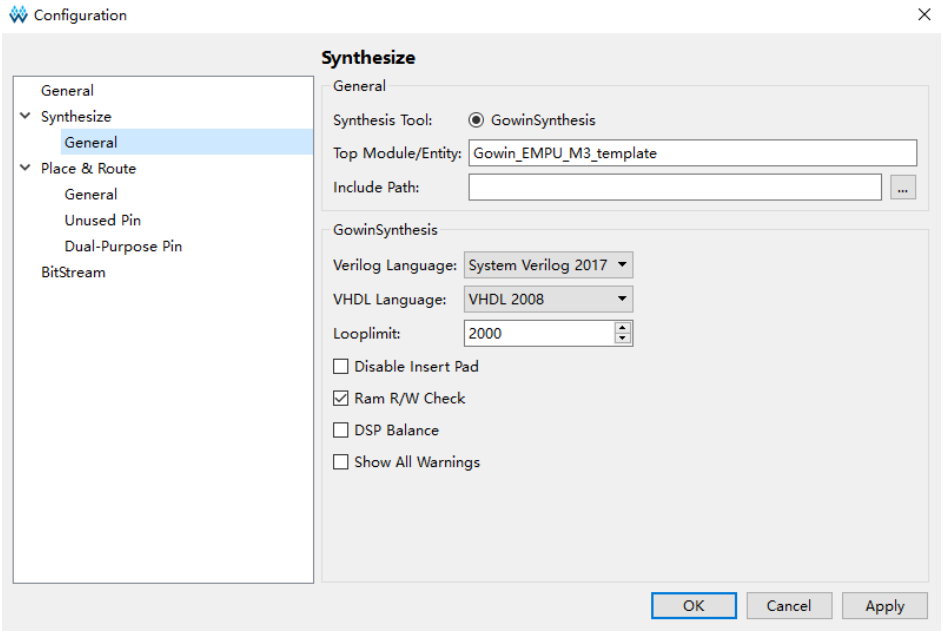
3.5 Configuration

3.5.1 Synthesis Configuration

Synthesis configuration is as shown in Figure 3-23.

- Configure "Top Module/Entity" according to the top module name in the design;
- Configure "Include Path" according to the file path in the design;
- Configure "Verilog Language" according to System Verilog 2017.

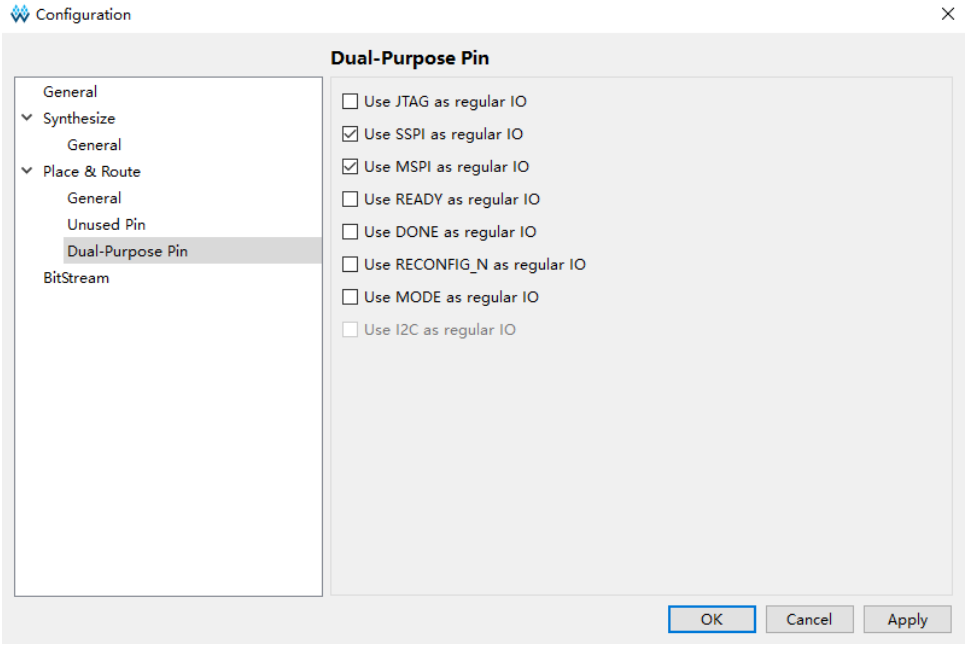
Figure 3-23 Top Module Configuration



3.5.2 Dual-Purpose Pin Configuration

If Gowin_EMPU_M3 uses off-chip SPI-Flash downloading and boot, use MSPI as regular IO, as shown in Figure 3-24.

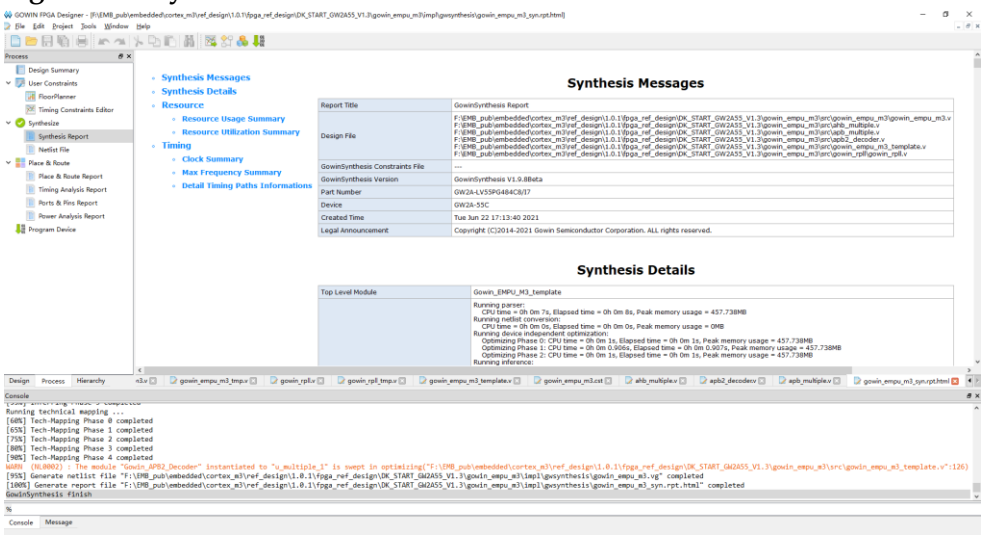
Figure 3-24 Dual-Purpose Pin Configuration



3.6 Synthesize

Run GowinSynthesis, the synthesis tool of Gowin Software, to complete the synthesis of RTL design and generate netlist files, as shown in Figure 3-25.

Figure 3-25 Synthesize

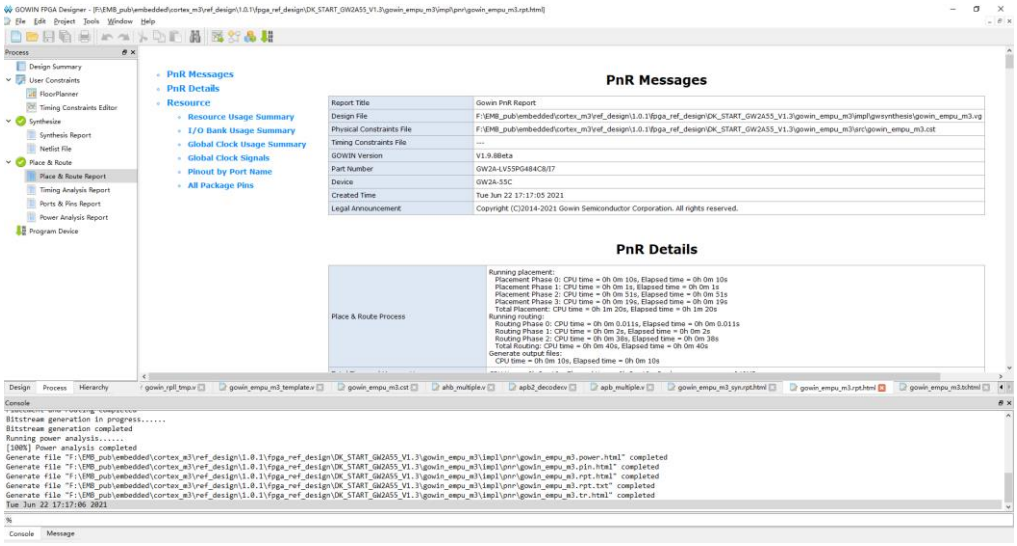


For the use of tool, please refer to *SUG100, Gowin Software User Guide*.

3.7 Place & Route

Run the Place & Route tool in Gowin software and generate the bitstream files, as shown in Figure 3-26.

Figure 3-26 Place & Route



For the use of tool, please refer to *SUG100, Gowin Software User Guide*.

3.8 Download

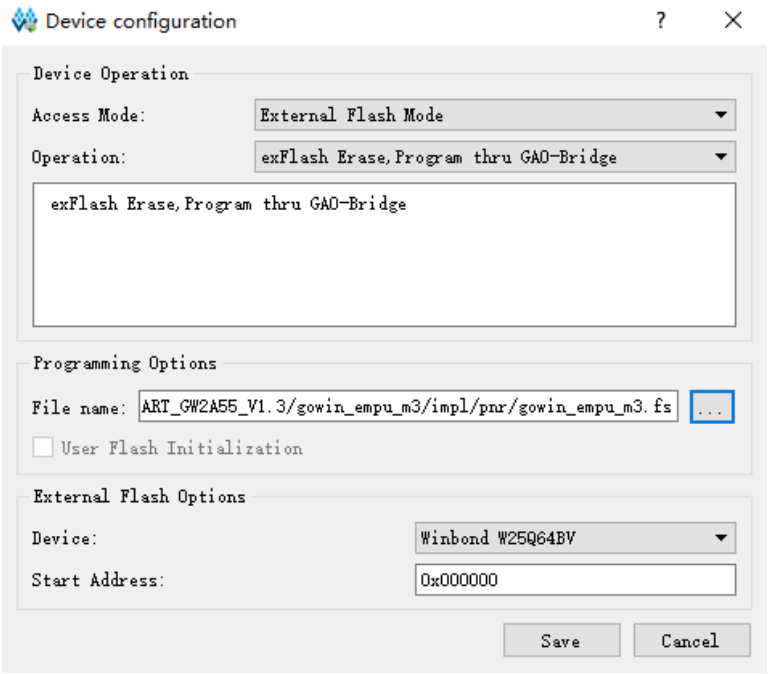
Run Programmer to download the bitstream file.

Click "Edit > Configure Device" in the menu bar or "Configure Device " ( ) in the tool bar to open the "Device configuration".

- Select "External Flash Mode" in Access Mode drop-down list;

- Select "exFlash Erase, Program thru GAO-Bridge" or "exFlash Erase, Program, Verify thru GAO-Bridge" in Operation drop-down list;
- Import the required bitstream file in "Programming Options > File name" option.
- Select based on the on-board Flash in "External Flash Options > Device" (such as Winbond W25Q64BV);
- Configure the start address as "0x000000" in "External Flash Options > Start Address";
- Click "Save" as shown in Figure 3-27.

Figure 3-27 Download



After device configuration, click Program/Configure " " in the Programmer toolbar to complete bit stream files downloading.

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

4 Reference Design

Gowin_EMPU_M1 offers hardware reference design:
Gowin_EMPU_M3\ref_design\FPGA_RefDesign


GOWIN
PROGRAMMING FOR THE FUTURE