



Gowin RiscV_AE350_SOC Hardware Design Manual

MUG1031-1.3.1E, 01/23/2026

Copyright © 2026 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 means, 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
10/31/2023	1.0E	Initial version published.
01/26/2024	1.1E	● Port signals DDR3_CS_N, DDR3_RSTN, POR_RSTN and HW_RSTN added. ● DK-START-GW5AT138K V2.1 and Tang-MEGA-138K-Pro-Dock development board reference designs added. ● Extended_AHB2AHBBridge and Extended_Memory (BlockRAM) solutions updated; Extended_Memory (DDR3), Extended_AHB_CAN, Extended_AHB_Ethernet solutions added.
06/14/2024	1.2E	● Instruction memory supports multiple types of memory, including SPI Flash Memory, ITCM, and user-customized memory. ● Data memory supports multiple types of memory, including DDR3 Memory, DTCM, and user-customized memory. ● Supports shared access to DDR3 Memory by the MCU end and the FPGA end. ● DDR3 Memory supports user-customized configurations. ● Logic-extension SD peripheral supported.
03/21/2025	1.3	● The DDR3 Memory controller updated. ● The reference design development board type updated. ● The software version and some software screenshots updated.
01/23/2026	1.3.1	The description of debug of “Table 3-1 Core System Features” in “3.1 Core System” updated.

Contents

Contents	i
List of Figures	iii
List of Tables	v
1 About This Manual	1
1.1 Purpose	1
1.2 Terminology and Abbreviation	1
1.3 Support and Feedback	2
2 System Architecture	3
3 System Feature	5
3.1 Core System	5
3.2 Memory System	7
3.3 Bus Peripheral System	8
4 System Performance	11
4.1 System Clock	11
4.2 DDR3 Clock	11
4.3 Performance Indicators	12
5 System Ports	13
6 System Configuration	20
7 Design Flow	22
7.1 Hardware Target	22
7.2 Software Version	22
7.3 Design Flow	22
8 Detailed Design	24
8.1 Create a Project	24

8.2 IP Design	27
8.2.1 PLIC	29
8.2.2 Instruction Memory	29
8.2.3 Data Memory	32
8.2.4 I2C	35
8.2.5 PIT	36
8.2.6 SPI	37
8.2.7 UART1 and UART2	37
8.2.8 GPIO	38
8.2.9 WDT	39
8.2.10 RTC	39
8.2.11 SMU	40
8.2.12 DMAC	40
8.2.13 Extended APB Slave	41
8.2.14 Extended AHB Slave	41
8.2.15 Extended AHB Master	42
8.3 User Design	43
8.3.1 RiscV_AE350_SOC IP Design	43
8.3.2 PLL_ADV IP Design	43
8.3.3 User Design	45
8.4 Constraints	45
8.4.1 Physical Constraints	45
8.4.2 Timing Constraints	46
8.5 Configuration	47
8.5.1 Global Option	47
8.5.2 Synthesis Option	48
8.5.3 Place & Route Option	48
8.5.4 BitStream Option	50
8.6 Synthesis	51
8.7 Place & Route	51
8.8 Download	52
9 Reference Design	54

List of Figures

Figure 2-1 System Architecture.....	3
Figure 8-1 New Project	24
Figure 8-2 Project Name	25
Figure 8-3 Select Device.....	26
Figure 8-4 Summary	26
Figure 8-5 IP Core Generator	27
Figure 8-6 RiscV_AE350_SOC IP Core	28
Figure 8-7 PLIC Configuration	29
Figure 8-8 Instruction Memory Configuration	30
Figure 8-9 ITCM Configuration	31
Figure 8-10 Customized Configuration	31
Figure 8-11 DDR3 Configuration.....	33
Figure 8-12 DTCM Configuration.....	34
Figure 8-13 Customized Configuration	34
Figure 8-14 I2C Configuration.....	35
Figure 8-15 PIT Configuration.....	36
Figure 8-16 SPI Configuration	37
Figure 8-17 UART1 and UART2 Configuration.....	38
Figure 8-18 GPIO Configuration	38
Figure 8-19 WDT Configuration	39
Figure 8-20 RTC Configuration	39
Figure 8-21 SMU Configuration	40
Figure 8-22 DMAC Configuration.....	40
Figure 8-23 Extended APB Slave Configuration.....	41
Figure 8-24 Extended AHB Slave Configuration.....	42

Figure 8-25 Extended AHB Master Configuration.....	42
Figure 8-26 RiscV_AE350_SOC IP Design	43
Figure 8-27 PLL_ADV IP Design	44
Figure 8-28 Timing Analysis Report.....	47
Figure 8-29 General Configuration	47
Figure 8-30 Synthesis Configuration	48
Figure 8-31 Place Option	49
Figure 8-32 Route Option	49
Figure 8-33 Dual-Purpose Pin Option.....	50
Figure 8-34 BitStream Configuration	50
Figure 8-35 Synthesize	51
Figure 8-36 Place & Route.....	51
Figure 8-37 Download Configuration	52
Figure 8-38 Download Bitstream File.....	53

List of Tables

Table 1-1 Terminology and Abbreviations	1
Table 3-1 Core System Features	5
Table 3-2 Memory System Features	7
Table 3-3 Bus Peripheral System Features	8
Table 4-1 System Clock	11
Table 4-2 DDR3 Clock	11
Table 4-3 Performance Indicators	12
Table 5-1 System Port.....	13
Table 6-1 System Configuration.....	20
Table 8-1 Physical Constraints for JTAG in Reference Design (Andes AICE-MINI+ Debugger).....	45
Table 8-2 Physical Constraints for JTAG in Reference Design (Gowin FTDI USB Programmer)	45
Table 8-3 Physical Constraints for JTAG in Reference Design (Olimex Debugger).....	46

1 About This Manual

1.1 Purpose

This manual describes the system architecture, system features, system performance, system ports, system configuration, hardware design flow of Gowin RiscV_AE350_SOC.

1.2 Terminology and Abbreviation

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

Table 1-1 Terminology and Abbreviations

Terminology and Abbreviations	Meaning
AHB	Advanced High Performance Bus
APB	Advanced Peripheral Bus
BMC	Bus Matrix Controller
CAN	Controller Area Network
DDR	Double Data Rate Memory
DLM	Data Local Memory
DMAC	Direct Memory Access Controller
DSP	Digital Signal Processor
I2C	Inter-Integrated Circuit Bus
ILM	Instruction Local Memory
ISA	Instruction Set Architecture
LRU	Least Recently Used
MCU	Micro Controller Unit
PIT	Programmable Interval Timer

Terminology and Abbreviations	Meaning
PLIC	Platform Level Interrupt Controller
PLMT	Platform Level Machine Timer
PMP	Physical Memory Protection
PWM	Pulse Width Modulation
RISC-V	Reduced Instruction Set Computer V
RTC	Real Time Clock
SMU	System Manage Unit
SOC	System on Chip
SPI	Serial Peripheral Interface
UART	Universal Synchronous and Asynchronous Receiver/Transmitter
WDT	Watch Dog Timer
WFI	Wait for Interrupt

1.3 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.

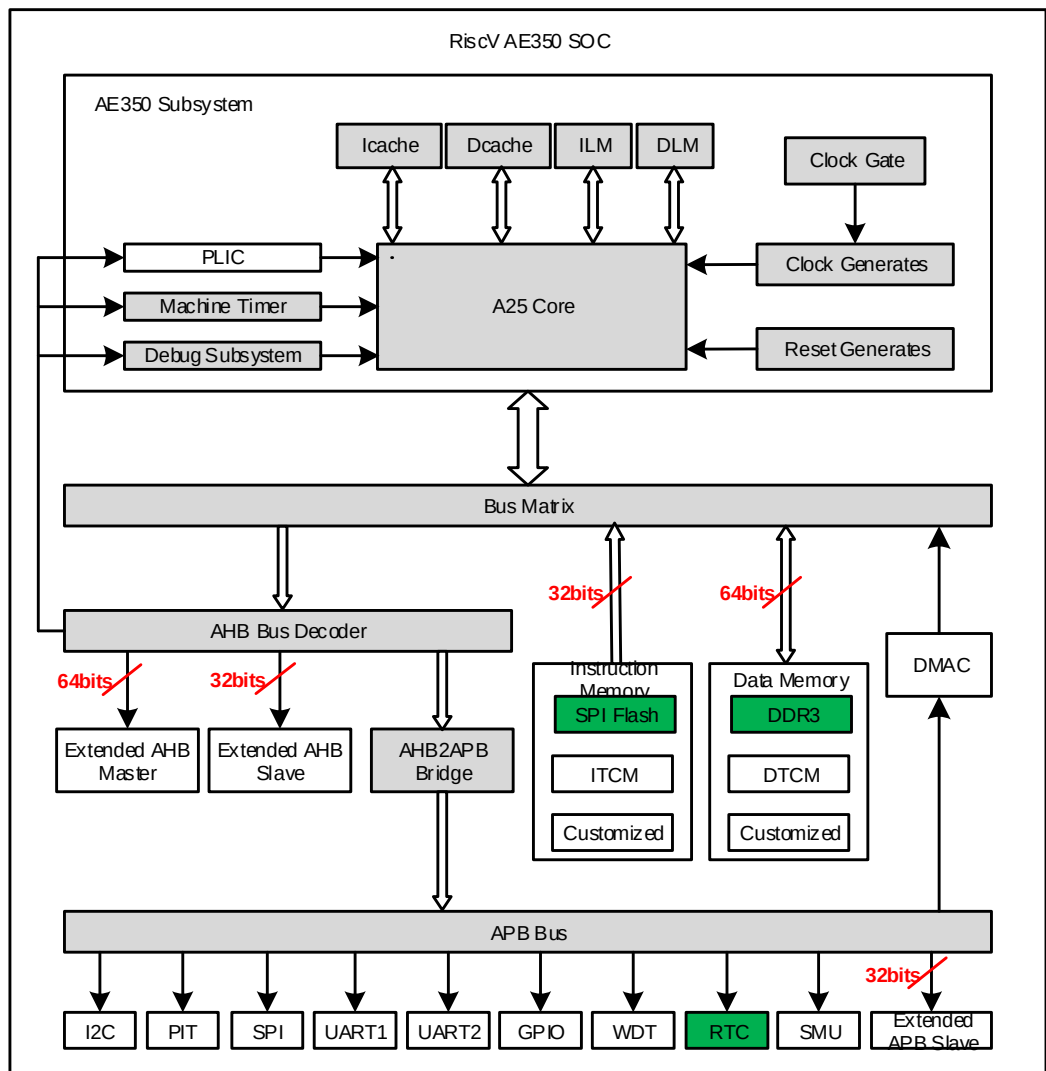
Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

2 System Architecture

Gowin RiscV_AE350_SOC system architecture is as shown in Figure 2-1.

Figure 2-1 System Architecture



Gowin RiscV_AE350_SOC includes the core system, memory system, and bus peripheral system.

- The core system includes RISC-V core, PLIC, PLMT, debug system, clock gating, clock system, reset system, I-Cache, D-Cache, ILM, and DLM.
- The memory system includes instruction memory and data memory. Instruction memory includes SPI Flash Memory, ITCM, or user-customized memory. Data memory includes DDR3 Memory, DTCM, or user-customized memory
- The bus peripheral system includes AHB bus peripherals and APB bus peripherals. The AHB bus peripherals include 64-bit Extended AHB Master, 32-bit Extended AHB Slave, and AHB2APB Bridge. The APB bus peripherals include I2C, PIT, SPI, UART1, UART2, GPIO, WDT, RTC, SMU, DMAC, 32-bit Extended APB Slave. The example also includes AHB/APB bus peripherals implemented by FPGA logic extension such as CAN, Ethernet, Flash, SD, etc.

3 System Feature

The following section describes the system features of Gowin RiscV_AE350_SOC in details.

3.1 Core System

The features of core system are shown in Table 3-1.

Table 3-1 Core System Features

System	Feature
MCU Core	5-stage in-order execution pipeline
	Frequency: Max. 800MHz
	Hardware multiplier radix-2/radix-4/radix-14/radix-256/fast
	Hardware divider
	Optional branch prediction
	4-entry return address stack (RAS) Static branch prediction Dynamic branch prediction 32/64/128/256-entry branch target buffer (BTB) 256-entry branch history 8-bit global branch history
	Machine mode, Supervisor mode and User mode
	Optional performance monitors
	Misaligned memory access
	RISC-V physical memory protection
ISA	RISC-V RV32I base integer instruction set
	RISC-V "C" standard extension for compressed instructions
	RISC-V "M" standard extension for integer multiplication and

System	Feature
	division
	RISC-V "A" standard extension for atomic instructions
	RISC-V "N" standard extension for user-level interrupt and exception handling
	RISC-V "F" and "D" standard extensions for single/double-precision floating-point
	DSP supported
Privilege	Machine, Supervisor and User privilege mode
	Page-based virtual memory, supervisor mode and SV32 virtual address translation scheme
	128 number of shared Translation Lookaside Buffer (TLB) entries
	8 number of instruction TLB entries
	8 number of data TLB entries
	16 number of support physical memory protection entries
	Hardware performance monitors
	Vectored PLIC extension
	Hardware stack protection extension
	Hardware performance throttling mechanism extension
Power Management	Wait-for-interrupt (WFI) mode
Debug	RISC-V extend debug support
	Number of breakpoints: 8
	External JTAG debug transport module
	JTAG: IEEE STD 1149.1 style 4-wire JTAG interface
PLIC	Configurable number of interrupts: 1-1023
	Number of interrupt priorities: 15
	Configurable number of targets: 1-16
	Vectored interrupt extension
	Support software interrupts
	Support 16 user interrupts
PLMT	The RISC-V architecture defines a machine timer that provides a real time counter and generates timer interrupts
Cache	I-Cache
	Size: 32KB
	Associativity: 4-way

System	Feature
	Replacement policy: pseudo-LRU Soft error protection: single/double error correction
	D-Cache Size: 32KB Associativity: 4-way Replacement policy: pseudo-LRU Soft error protection: single/double error detection
Local Memory	ILM Base: 0xA0000000 Size: 64KB Soft error protection: single error correction and double error detection
	DLM Base: 0xA0200000 Size: 64KB Soft error protection: single error correction and double error detection
	Slave port supports local memory as a slave, can be accessed by FPGA fabric as a master via the 64-bit Extended AHB Master bus

3.2 Memory System

The features of Memory system are shown in Table 3-2.

Table 3-2 Memory System Features

System	Feature
Instruction Memory	Address: 0x80000000~0x8FFFFFFF
	Size: Max. 256MB
	Bus: AHB bus interface
	SPI Flash Memory, ITCM or user-customized memory
	Frequency: less than 200MHz
Data Memory	Address: 0x00000000~0x7FFFFFFF
	Size: Max. 2GB
	Bus: AHB bus interface
	DDR3 Memory, DTCM or user-customized memory
	Frequency: less than 200MHz

3.3 Bus Peripheral System

The features of bus peripheral system are shown in Table 3-3.

Table 3-3 Bus Peripheral System Features

System	Feature
SMU	SMU controls the flow of power and clock changes
UART1	The UART controller handles communications to the UART interface. Compatible with the 16C550A register structure Support of the hardware flow control (CTS/RTS) Support of hardware handshaking to the DMA controller Option of by-8 or by-16 over-sampling frequency Support of 64-entry transmit/receive FIFO depth
UART2	The UART controller handles communications to the UART interface. Compatible with the 16C550A register structure Support of the hardware flow control (CTS/RTS) Support of hardware handshaking to the DMA controller Option of by-8 or by-16 over-sampling frequency Support of 64-entry transmit/receive FIFO depth
PIT	The PIT controller is a set of compact multi-function timers, which can be used as PWM or simple timers. Each multi-function timer provides the following 6 usage scenarios: One 32-bit timer Two 16-bit timers Four 8-bit timers One 16-bit PWM One 16-bit timer and one 8-bit PWM Two 8-bit timers and one 8-bit PWM
WDT	The WDT controller prevents the system from hanging if software is trapped in a deadlock condition. Internal/external clock source selection Separate timers for the watchdog interrupt and the system reset Eight choices of watchdog timer intervals Four choices of reset timer intervals Register write protection for watchdog timer control register and restart register Configurable magic number for register write protection

System	Feature
	Configurable magic number to restart the watchdog timer
RTC	RTC keeps track of current time relative to a base time. The time is stored in a RTC counter which records the amount of elapsed time since RTC is enabled. The frequency of clock source (before the clock divider) for the counter is 32.768KHz Separate second, minute, hour and day counters Periodic interrupts: half-second, second, minute, hour and day interrupts Programmable alarm interrupt with specified second, minute and hour values
GPIO	The GPIO controller supports up to 32 channels with independently programmable input/output control. Support of up to 32 GPIO channels Independent control of each channel Programmable I/O direction Optional pull-up/down control Optional support of interrupt trigger control Flexible combination of interrupt trigger modes: high/low level trigger and rising/falling/both edge trigger Optional de-bounce functionality for input channels
I2C	The I2C controller handles communications to I2C interface. Programmable to be either a master or a slave device Programmable clock/data timing Support of the I2C-bus Standard-mode (100 kb/s), Fast-mode (400 kb/s) and Fast-mode plus (1 Mb/s) Support of hardware handshaking to the DMA controller Support of the master-transmit, master-receive, slave-transmit and slave-receive modes Support of the multi-master mode Support of 7-bit and 10-bit addressing modes Support of general call addressing mode Support of auto clock stretch
SPI	The SPI controller handles communications to the SPI. The supported serial data formats range from 4 bits to 32 bits in length. Compliant with AMBA 2 AHB protocol specification Compliant with AMBA 3 APB protocol specification Support of MSB/LSB first transfer Support of DMA data transfer

System	Feature
	Support of programmable SPI SCLK Support of memory-mapped access (read-only) through AHB bus or EILM bus Support of SPI slave mode Quad I/O SPI interfaces TX/RX FIFO depth is 128 Programming port location on APB interfaces
DMAC	The DMAC enhances system performance by transferring large data blocks between devices in background to offload the processor. Compliant with AMBA AXI4 and APB4 Support of up to 8 DMA channels Support of up to 16 DMA request/acknowledge pairs for hardware handshake, 7 internal used, 8 for user Support of up to two AXI master ports for data transfers Support of up to two configurable DMA cores Support of an APB slave port for DMA register programming Support of 24–64 bits AXI address width Support of 32/64/128/256 bits AXI data width Support of narrow transfers on the AXI bus Support of group round-robin arbitration scheme with 2 priority levels Support of chain transfers
Extended APB Slave	For user to extend APB interface peripherals in FPGA fabric 32-bit data bit width Size: 64MB AE350_SOC is as a master
Extended AHB Slave	For user to extend AHB interface peripherals in FPGA fabric 32-bit data bit width Size: 128MB AE350_SOC is as a master
Extended AHB Master	For slave port supported, assesses local memory by a master in FPGA fabric 64-bit data bit width AE350_SOC is as a slave

4 System Performance

4.1 System Clock

The system clock of Gowin RiscV_AE350_SOC is shown in Table 4-1.

Table 4-1 System Clock

System Clock	Clock Frequency	Description
Core Clock	Up to 800MHz	RISC-V core clock
DDR clock	Less than 200MHz	Data memory clock
AHB clock	Less than 200MHz	AHB bus clock and instruction memory clock
APB clock	Less than 200MHz	APB bus clock
RTC clock	32.768KHz	RTC clock

Note!

- The core clock, DDR clock, and AHB/APB clock are asynchronous with each other.
- AHB clock and APB clock are synchronous with a 1:1 ratio.

4.2 DDR3 Clock

If DDR3 Memory is selected for Gowin RiscV_AE350_SOC Data Memory, the DDR3 Memory clocks are as shown in Table 4-2.

Table 4-2 DDR3 Clock

DDR3 Clock	Clock Frequency	Description
DDR3_MEMORY_CLK	Less than or equal to 400MHz	DDR3 chip operating clock input by users
DDR3_RW_CLK	Greater than or equal to one-quarter of DDR3_MEMORY_CLK	Reference input clock for FPGA fabric read/write DDR3
DDR3_CLK_IN	50MHz	DDR3 reference input clock

4.3 Performance Indicators

The performance indicators for Gowin RiscV_AE350_SOC are measured based on benchmark programs Dhrystone, CoreMark, and Whetstone, as shown in Table 4-3.

Table 4-3 Performance Indicators

Benchmark Program	Performance Indicator
Dhrystone	1.72 DMIPS/MHz
CoreMark	3.224845 CoreMark/MHz
Whetstone	0.783 MWIPS/MHz

5 System Ports

The system ports of Gowin RiscV_AE350_SOC are as shown in Table 5-1.

Table 5-1 System Port

Name	I/O	Width	Description	Module
POR_RSTN	input	1	Power on reset	-
HW_RSTN	input	1	Hardware reset	-
CORE_CLK	input	1	Core clock	-
DDR_CLK	input	1	Data memory clock	-
AHB_CLK	input	1	AHB or instruction memory clock	-
APB_CLK	input	1	APB clock	-
RTC_CLK	input	1	RTC clock	RTC
TCK_IN	input	1	JTAG clock	Debug Subsystem
TMS_IN	input	1	JTAG tms	
TRST_IN	input	1	JTAG trst	
TDI_IN	input	1	JTAG tdi	
TDO_OUT	output	1	JTAG tdo	
TDO_OE	output	1	JTAG tdo oe	
FLASH_SPI_CSN	inout	1	Flash chip select n	SPI Flash Instruction Memory
FLASH_SPI_MISO	inout	1	Flash master in and slave out	
FLASH_SPI_MOSI	inout	1	Flash master out and slave in	
FLASH_SPI_CLK	inout	1	Flash clock	
FLASH_SPI_HOLDN	inout	1	Flash hold n	
FLASH_SPI_WPN	inout	1	Flash write protection n	

Name	I/O	Width	Description	Module
FR_PCLK	input	1	Flash register APB clock	SPI Flash Instruction Memory and Flash Register R/W Mode
FR_PRESETN	input	1	Flash register APB reset n	
FR_PADDR	input	[31:0]	Flash register APB address	
FR_PENABLE	input	1	Flash register APB enable	
FR_PRDATA	output	[31:0]	Flash register APB read data	
FR_PREADY	output	1	Flash register APB ready out	
FR_PSEL	input	1	Flash register APB select	
FR_PWDATA	input	[31:0]	Flash register APB write data	
FR_PWRITE	input	1	Flash register APB write enable	
ROM_HADDR	output	[31:0]	ROM AHB address	Customized Instruction Memory
ROM_HRDATA	input	[31:0]	ROM AHB read data	
ROM_HREADY	input	1	ROM AHB ready in	
ROM_HRESP	input	1	ROM AHB response	
ROM_HTRANS	output	[1:0]	ROM AHB transmission type	
ROM_HWRITE	output	1	ROM AHB write enable	
ROM_HCLK	output	1	ROM AHB clock	
ROM_HRSTN	output	1	ROM AHB reset n	
DDR3_MEMORY_CLK	input	1	DDR3 memory clock	DDR3 Data Memory
DDR3_CLK_IN	input	1	DDR3 input clock	
DDR3_RSTN	input	1	DDR3 reset	
DDR3_LOCK	input	1	DDR3 lock	
DDR3_STOP	output	1	DDR3 stop	
DDR3_INIT	output	1	DDR3 initialized flag	
DDR3_BANK	output	[2:0]	DDR3 bank address	
DDR3_CS_N	output	1	DDR3 chip select	

Name	I/O	Width	Description	Module
DDR3_RAS_N	output	1	DDR3 row address select	
DDR3_CAS_N	output	1	DDR3 column address select	
DDR3_WE_N	output	1	DDR3 write enable	
DDR3_CK	output	1	DDR3 clock	
DDR3_CK_N	output	1	DDR3 clock differential	
DDR3_CKE	output	1	DDR3 clock enable	
DDR3_RESET_N	output	1	DDR3 reset	
DDR3_ODT	output	1	DDR3 on-die termination	
DDR3_ADDR	output	[13:0]	DDR3 address	
DDR3_DM	output	[1:0]	DDR3 data mask	
DDR3_DQ	inout	[15:0]	DDR3 data	
DDR3_DQS	inout	[1:0]	DDR3 data select	
DDR3_DQS_N	inout	[1:0]	DDR3 data select differential	
clk_lane4	input	1	Writing clock	DDR3 Data Memory and Shared DDR3: Logic R/W Mode
addr_lane4	input	[31:0]	Writing address	
wr_mask_lane4	input	[3:0]	Writing mask	
wr_data_lane4	input	[63:0] [31:0]	Writing data	
wr_en_lane4	input	1	Writing enable	
wr_go_lane4	input	1	Writing control	
burstcount_lane4	input	[7:0]	Writing burst amount	
wr_wait_lane4	output	1	Writing wait	
wr_done_lane4	output	1	Writing done	
clk_lane5	input	1	Reading clock	
addr_lane5	input	[31:0]	Reading address	
rd_en_lane5	input	1	Reading enable	
rd_go_lane5	input	1	Reading control	
burstcount_lane5	input	[7:0]	Reading burst amount	
rd_valid_lane5	output	1	Reading valid	
rd_data_lane5	output	[63:0]	Reading data	

Name	I/O	Width	Description	Module
		[31:0]		Customized Data Memory
rd_rdy_lane5	output	1	Reading ready	
DDR_HADDR	output	[31:0]	DDR AHB address	
DDR_HBURST	output	[2:0]	DDR AHB burst	
DDR_HPROT	output	[3:0]	DDR AHB protection	
DDR_HRDATA	input	[63:0]	DDR AHB read data	
DDR_HREADY	input	1	DDR AHB ready in	
DDR_HRESP	input	1	DDR AHB response	
DDR_HSIZE	output	[2:0]	DDR AHB size	
DDR_HTRANS	output	[1:0]	DDR AHB transmission type	
DDR_HWDATA	output	[63:0]	DDR AHB write data	
DDR_HWRITE	output	1	DDR AHB write enable	
DDR_HCLK	output	1	DDR AHB clock	
DDR_HRSTN	output	1	DDR AHB reset n	
EXT_INT	input	[15:0]	Extended interrupts	PLIC
APB_PADDR	output	[31:0]	APB slave address	Extended APB Slave
APB_PENABLE	output	1	APB slave enable	
APB_PRDATA	input	[31:0]	APB slave read data	
APB_PREADY	input	1	APB slave ready in	
APB_PSEL	output	1	APB slave select	
APB_PWDATA	output	[31:0]	APB slave write data	
APB_PWRITE	output	1	APB slave write enable	
APB_PSLVERR	input	1	APB slave error response	
APB_PPROT	output	[2:0]	APB slave protection	
APB_PSTRB	output	[3:0]	APB slave write strobe	
APB_PCLK	output	1	APB slave clock	
APB_PRSTN	output	1	APB slave reset n	
EXTS_HRDATA	input	[31:0]	AHB slave read data	Extended AHB Slave
EXTS_HREADYIN	input	1	AHB slave ready in	
EXTS_HRESP	input	1	AHB slave response	
EXTS_HADDR	output	[31:0]	AHB slave address	
EXTS_HBURST	output	[2:0]	AHB slave burst	

Name	I/O	Width	Description	Module
EXTS_HPROT	output	[3:0]	AHB slave protection	
EXTS_HSEL	output	1	AHB slave select	
EXTS_HSIZE	output	[2:0]	AHB slave size	
EXTS_HTRANS	output	[1:0]	AHB slave transmission type	
EXTS_HWDATA	output	[31:0]	AHB slave write data	
EXTS_HWRITE	output	1	AHB slave write enable	
EXTS_HCLK	output	1	AHB slave clock	
EXTS_HRSTN	output	1	AHB slave reset n	
EXTM_HADDR	input	[31:0]	AHB master address	Extended AHB Master
EXTM_HBURST	input	[2:0]	AHB master burst	
EXTM_HPROT	input	[3:0]	AHB master protection	
EXTM_HREADY	input	1	AHB master ready in	
EXTM_HSEL	input	1	AHB master select	
EXTM_HSIZE	input	[2:0]	AHB master size	
EXTM_HTRANS	input	[1:0]	AHB master transmission type	
EXTM_HWDATA	input	[63:0]	AHB master write data	
EXTM_HWRITE	input	1	AHB master write enable	
EXTM_HRDATA	output	[63:0]	AHB master read data	
EXTM_HREADYOUT	output	1	AHB master ready out	
EXTM_HRESP	output	1	AHB master response	
DMA_REQ	input	[7:0]	DMA requests	DMAC
DMA_ACK	output	[7:0]	DMA acknowledges	
SPI_HOLDN_IN	input	1	SPI hold n in	SPI and non-I/O ports
SPI_WPN_IN	input	1	SPI write protection n in	
SPI_CLK_IN	input	1	SPI clock in	
SPI_CSN_IN	input	1	SPI chip select n in	
SPI_MISO_IN	input	1	SPI miso in	
SPI_MOSI_IN	input	1	SPI mosi in	
SPI_HOLDN_OUT	output	1	SPI hold n out	
SPI_HOLDN_OE	output	1	SPI hold n oe	
SPI_WPN_OUT	output	1	SPI write protection n	

Name	I/O	Width	Description	Module
			out	
SPI_WPN_OE	output	1	SPI write protection n oe	
SPI_CLK_OUT	output	1	SPI clock out	
SPI_CLK_OE	output	1	SPI clock oe	
SPI_CSN_OUT	output	1	SPI chip select n out	
SPI_CSN_OE	output	1	SPI chip select n oe	
SPI_MISO_OUT	output	1	SPI miso out	
SPI_MISO_OE	output	1	SPI miso oe	
SPI_MOSI_OUT	output	1	SPI mosi out	
SPI_MOSI_OE	output	1	SPI mosi oe	
SPI_HOLDN	inout	1	SPI hold n	SPI and I/O ports
SPI_WPN	inout	1	SPI write protection n	
SPI_CLK	inout	1	SPI clock	
SPI_CSN	inout	1	SPI chip select n	
SPI_MISO	inout	1	SPI master in and slave out	
SPI_MOSI	inout	1	SPI master out and slave in	
I2C_SCL_IN	input	1	I2C clock in	I2C and non-I/O ports
I2C_SDA_IN	input	1	I2C data in	
I2C_SCL_OUT	output	1	I2C clock out	
I2C_SDA_OUT	output	1	I2C data out	
I2C_SCL	inout	1	I2C clock	I2C and I/O ports
I2C_SDA	inout	1	I2C data	
UART1_TXD	output	1	UART1 transmits data	UART1
UART1_RTSN	output	1	UART1 rts n	
UART1_RXD	input	1	UART1 receives data	
UART1_CTSN	input	1	UART1 cts n	
UART1_DSRN	input	1	UART1 dsr n	
UART1_DCDN	input	1	UART1 dcd n	
UART1_RIN	input	1	UART1 rin	
UART1_DTRN	output	1	UART1 dtr n	
UART1_OUT1N	output	1	UART1 out1 n	

Name	I/O	Width	Description	Module
UART1_OUT2N	output	1	UART1 out2 n	UART2
UART2_TXD	output	1	UART2 transmits data	
UART2_RTSN	output	1	UART2 rts n	
UART2_RXD	input	1	UART2 receives data	
UART2_CTSN	input	1	UART2 cts n	
UART2_DSRN	input	1	UART2 dsr n	
UART2_DCDN	input	1	UART2 dcd n	
UART2_RIN	input	1	UART2 rin	
UART2_DTRN	output	1	UART2 dtr n	
UART2_OUT1N	output	1	UART2 out1 n	
UART2_OUT2N	output	1	UART2 out2 n	
CH0_PWM	output	1	PWM channel 0 output	PWM
CH0_PWMOE	output	1	PWM channel 0 output enable	
CH1_PWM	output	1	PWM channel 1 output	
CH1_PWMOE	output	1	PWM channel 1 output enable	
CH2_PWM	output	1	PWM channel 2 output	
CH2_PWMOE	output	1	PWM channel 2 output enable	
CH3_PWM	output	1	PWM channel 3 output	
CH3_PWMOE	output	1	PWM channel 3 output enable	
GPIO_IN	input	[31:0]	GPIO in	GPIO and non-I/O ports
GPIO_OE	output	[31:0]	GPIO output enable	
GPIO_OUT	output	[31:0]	GPIO output	
GPIO	inout	[31:0]	GPIO	GPIO and I/O ports
CORE0_WFI_MODE	output	1	Core WFI mode status	SMU
RTC_WAKEUP	output	1	RTC wakeup status	

6 System Configuration

The system configuration of Gowin RiscV_AE350_SOC is as shown in Table 6-1.

Table 6-1 System Configuration

Option	Default Value	Description
Enable Extended Interrupts	Disable	Enables extended 16 external interrupt signals
SPI Flash Instruction Memory	Enable	Instruction memory: SPI Flash Memory
Flash Register R/W Mode	Disable	Enables APB bus interface in Flash register mode
ITCM Instruction Memory	Disable	Instruction memory: ITCM
ITCM Size	Disable	ITCM size
ITCM Initialization Path	Disable	ITCM initialization path
Customized Instruction Memory	Disable	Instruction memory: User-customized Memory
DDR3 Data Memory	Enable	Data memory: DDR3 Memory
Default DDR3 Settings	Enable	Default DDR3 Memory Interface IP parameter configuration
Customized DDR3 Settings	Disable	User-customized DDR3 Memory Interface IP parameter configuration
Shared DDR3: Logic R/W Mode	Disable	DDR3 Memory shared by MCU-side and FPGA-side
Addr Width	Disable	The address bus width for accessing DDR3 Memory at the FPGA side.
Data Width	Disable	The data bus width for accessing

Option	Default Value	Description
		DDR3 Memory at the FPGA side.
DTCM Data Memory	Disable	Data memory: DTCM
DTCM Size	Disable	DTCM size
Customized Data Memory	Disable	Data memory: User-customized memory
Enable Extended AHB Slave	Disable	Enables extended 32-bit AHB Slave interface with AE350 as Master
Enable Extended AHB Master	Disable	Enables extended 64-bit AHB Slave interface with AE350 as Slave
Enable Extended DMAC	Disable	Enables extended 8 groups of DMA request/acknowledge signals
Enable I2C	Disable	Enables I2C
Enable I2C I/O Ports	Enable	Enables I2C "INOUT" ports
Enable PIT	Disable	Enables PIT
PWM Channel 0	Disable	Enables PWM channel 0
PWM Channel 1	Disable	Enables PWM channel 1
PWM Channel 2	Disable	Enables PWM channel 2
PWM Channel 3	Disable	Enables PWM channel 3
Enable SPI	Disable	Enables SPI
Enable SPI I/O Ports	Enable	Enables SPI "INOUT" ports
Enable UART1	Disable	Enables UART1
Enable UART2	Disable	Enables UART2
Enable GPIO	Disable	Enables GPIO
Enable GPIO I/O Ports	Enable	Enables GPIO "INOUT" ports
Enable WDT	Disable	Enables WDT
Enable RTC	Enable	Enables RTC
Enable SMU	Disable	Enables SMU
Enable Extended APB Slave	Disable	Enables extended 32-bit APB Slave interface with AE350 as Master

7 Design Flow

7.1 Hardware Target

- DK_START_GW5AST-LV138FPG676A V1.0
 - GW5AST-LV138FPG676AC1/I/O
 - GW5AST-138B
- TANG_MEGA-138K_Pro_Dock
 - GW5AST-LV138FPG676AES
 - GW5AST-138B

7.2 Software Version

Tested software version: Gowin_V1.9.11.01 (64-bit).

7.3 Design Flow

Gowin RiscV_AE350_SOC IP design flow is shown below:

1. Use IP Core Generator tool in Gowin Software to configure and generate RiscV_AE350_SOC IP design.
2. Configure the PLL_ADV IP option to generate a PLL_ADV IP design to provide clocking resources for the RiscV_AE350_SOC IP using IP Core Generator.
3. In the hardware design, instantiate the RiscV_AE350_SOC IP, instantiate the PLL_ADV IP, add other user logic designs, and connect the modules to form a complete top-level design.
4. Use the physical constraints tool FloorPlanner in Gowin Software to add physical constraints with the reference to the development board.
5. Use the timing constraints tool Timing Constraints Editor in Gowin Software to add timing constraints with the reference to the software

timing analysis report.

6. Configure synthesis, Place & Route, and bitstream options.
7. Run the synthesis tool GowinSynthesis to synthesize RiscV_AE350_SOC hardware design and generate the netlist file.
8. Run the Place & Route tool to generate the bitstream files.
9. Run Programmer to download the bitstream file.

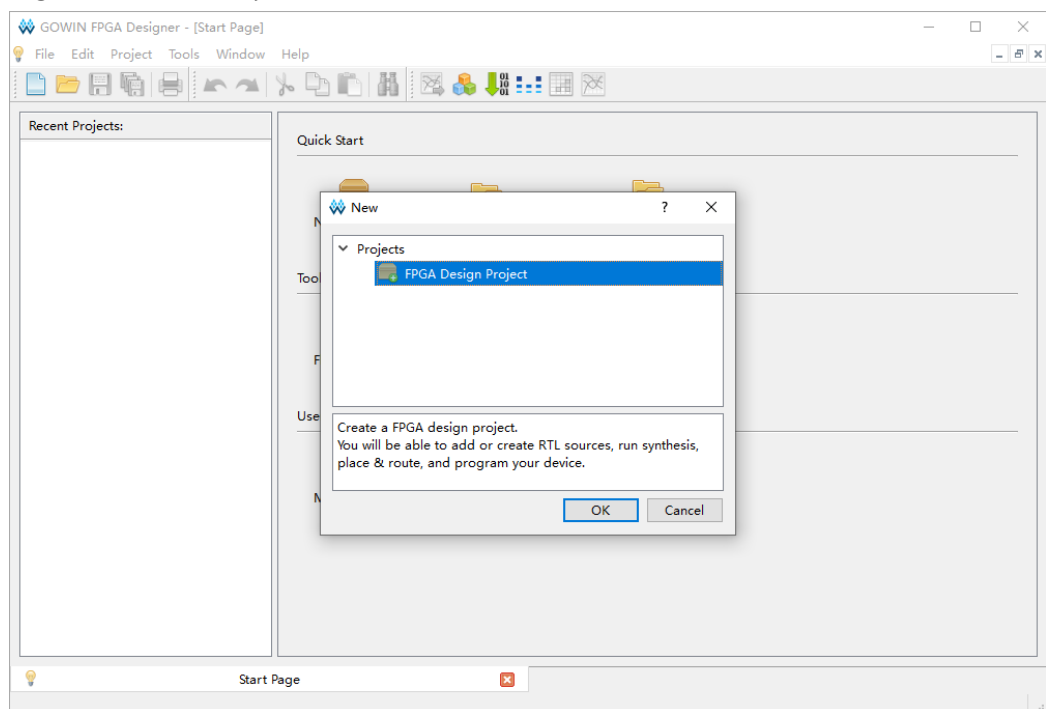
8 Detailed Design

8.1 Create a Project

Step 1

Double click to open Gowin Software. Click "File > New... > FPGA Design Project" on the menu bar, or click "", or click "New Project..." under "Quick Start" tab, then click "OK", as shown in Figure 8-1.

Figure 8-1 New Project



Step 2

Create the project name and path, as shown in Figure 8-2.

Figure 8-2 Project Name

Project Wizard

Project Name

Enter a name for your project, and specify a directory where the project will be stored. The directory will be created if it doesn't exist.

Name:

Create in: ...

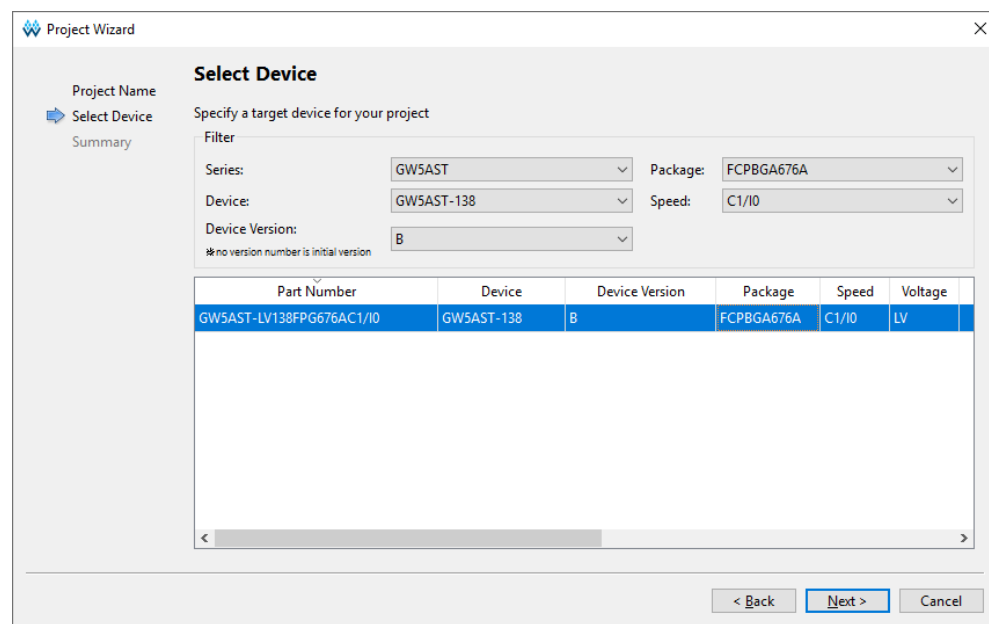
☐ Use as default project location

Next > **Cancel**

Step 3

- Set the device information, including Series, Device, Device Version, Package, Speed and Part Number, and click Next, as shown in Speed: C1/I0
 - Part Number: GW5AST-LV138FPG676AC1/I0
- Figure 8-3.
- For example:
- Series: GW5AST
 - Device: GW5AST-138
 - Device Version: B
 - Package: FCPBGA676A
 - Speed: C1/I0
 - Part Number: GW5AST-LV138FPG676AC1/I0

Figure 8-3 Select Device



The "Project Wizard" dialog box is shown with the "Select Device" step selected in the left sidebar. The main area is titled "Select Device" and contains the instruction "Specify a target device for your project". Below this is a "Filter" section with four dropdown menus: "Series" (set to GW5AST), "Device" (set to GW5AST-138), "Device Version" (set to B), "Package" (set to FCPBGA676A), and "Speed" (set to C1/I0). A note below the dropdowns states: "*no version number is initial version". Below the filter section is a table with the following data:

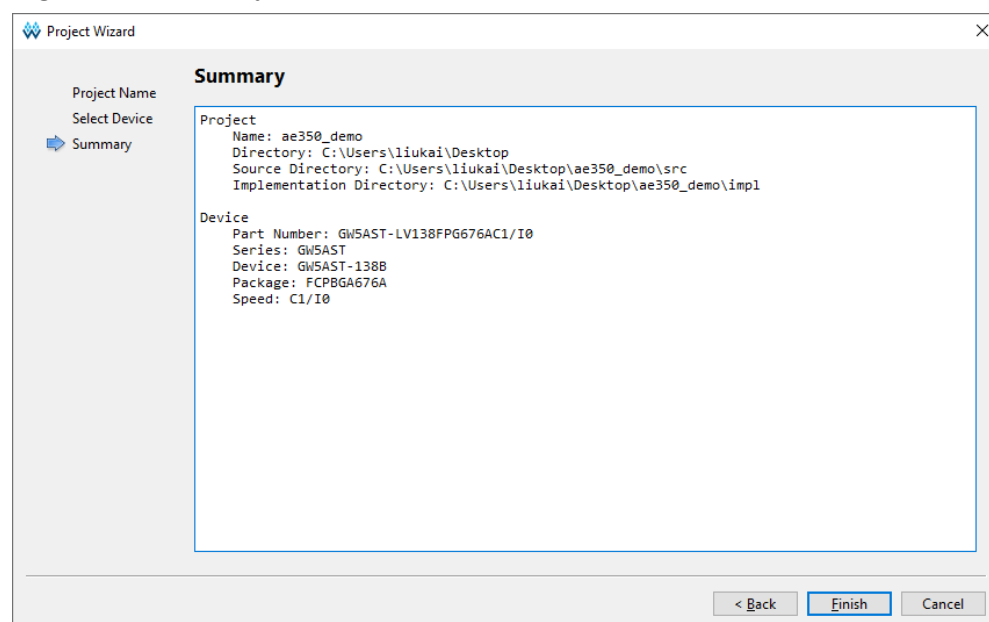
Part Number	Device	Device Version	Package	Speed	Voltage
GW5AST-LV138FPG676AC1/I0	GW5AST-138	B	FCPBGA676A	C1/I0	LV

At the bottom of the dialog are three buttons: "< Back", "Next >" (highlighted), and "Cancel".

Step 4

Click "Finish" to complete the project creation, as shown in Figure 8-4.

Figure 8-4 Summary



The "Project Wizard" dialog box is shown with the "Summary" step selected in the left sidebar. The main area is titled "Summary" and contains a text box with the following information:

```
Project
Name: ae350_demo
Directory: C:\Users\liukai\Desktop
Source Directory: C:\Users\liukai\Desktop\ae350_demo\src
Implementation Directory: C:\Users\liukai\Desktop\ae350_demo\impl

Device
Part Number: GW5AST-LV138FPG676AC1/I0
Series: GW5AST
Device: GW5AST-138B
Package: FCPBGA676A
Speed: C1/I0
```

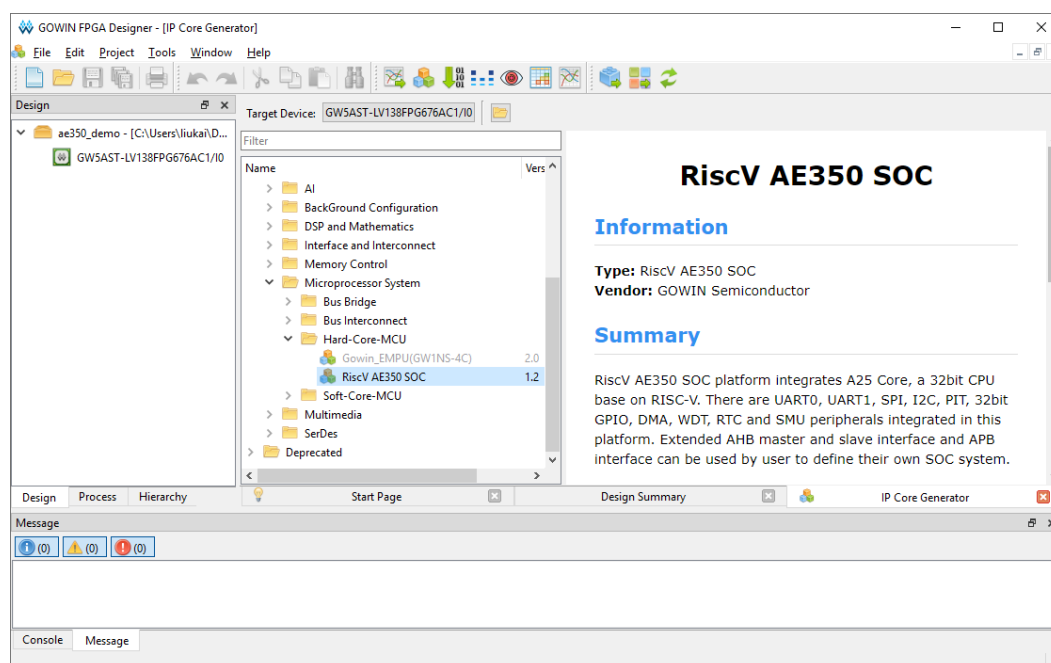
At the bottom of the dialog are three buttons: "< Back", "Finish" (highlighted), and "Cancel".

8.2 IP Design

Use IP Core Generator tool in Gowin Software to configure and generate RiscV_AE350_SOC IP design.

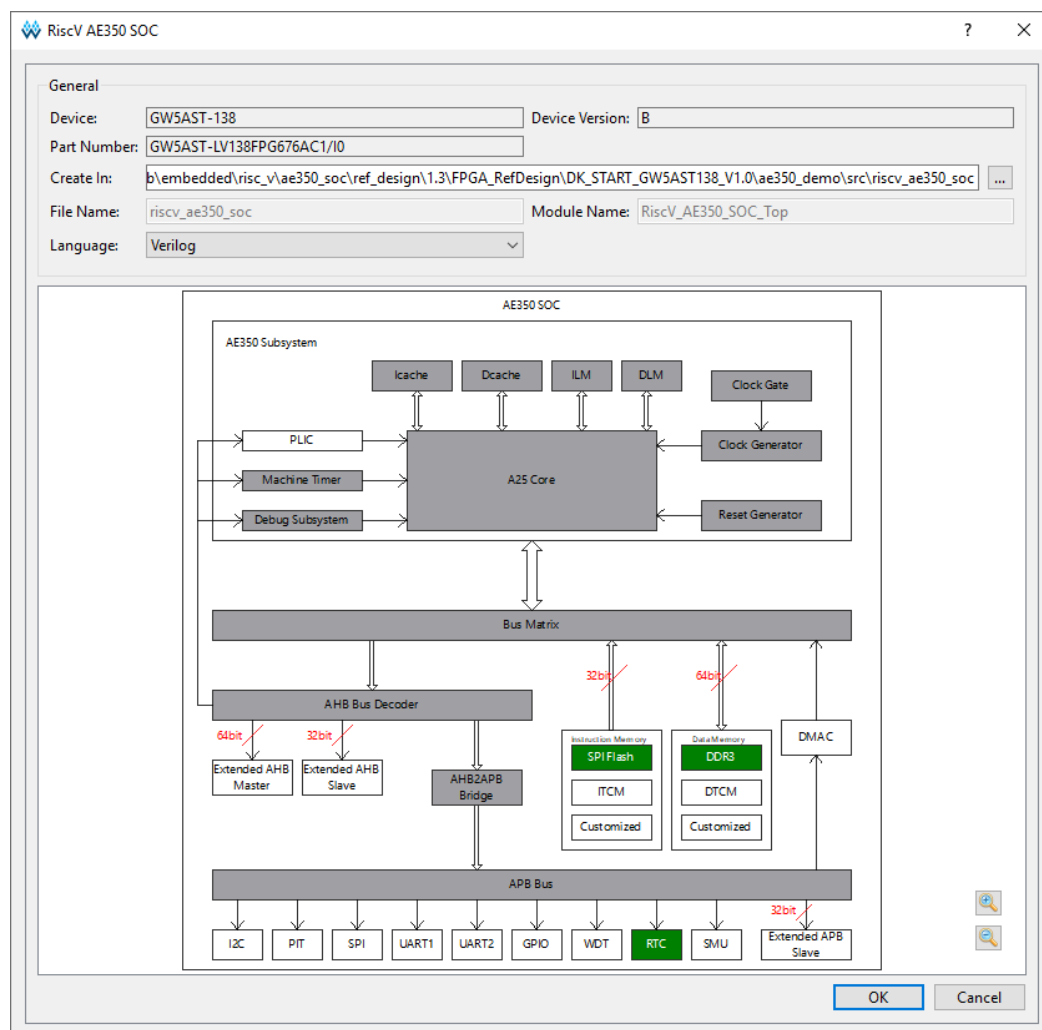
Select "Tools > IP Core Generator" on the menu bar or click the "🧩" on the tool bar to open IP Core Generator. Select "Soft IP Core > Microprocessor System > Hard-Core-MCU > RiscV AE350 SOC 1.2", as shown in Figure 8-5.

Figure 8-5 IP Core Generator



Double click "RiscV AE350 SOC 1.2" to open the RiscV_AE350_SOC IP Core as shown in Figure 8-6.

Figure 8-6 RiscV_AE350_SOC IP Core



The configurable system functions of RiscV_AE350_SOC include PLIC, Instruction Memory (including SPI Flash, ITCM, and Customized), Data Memory (including DDR3, DTCM, and Customized), I2C, PIT, SPI, UART1, UART2, GPIO, WDT, RTC, SMU, DMAC, Extended APB Slave, Extended AHB Slave, and Extended AHB Master, as described in Table 6-1.

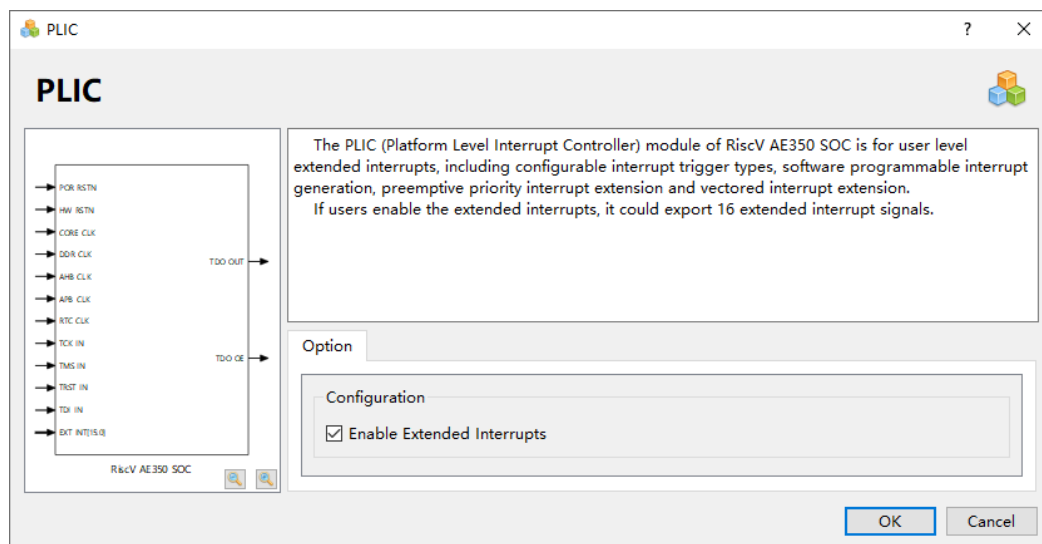
The following sections describe the configuration in details.

8.2.1 PLIC

Double click to open "PLIC", as shown in Figure 8-7.

If "Enable Extended Interrupts" is selected, the extended 16 external interrupt signals are enabled, disabled by default.

Figure 8-7 PLIC Configuration



8.2.2 Instruction Memory

SPI Flash

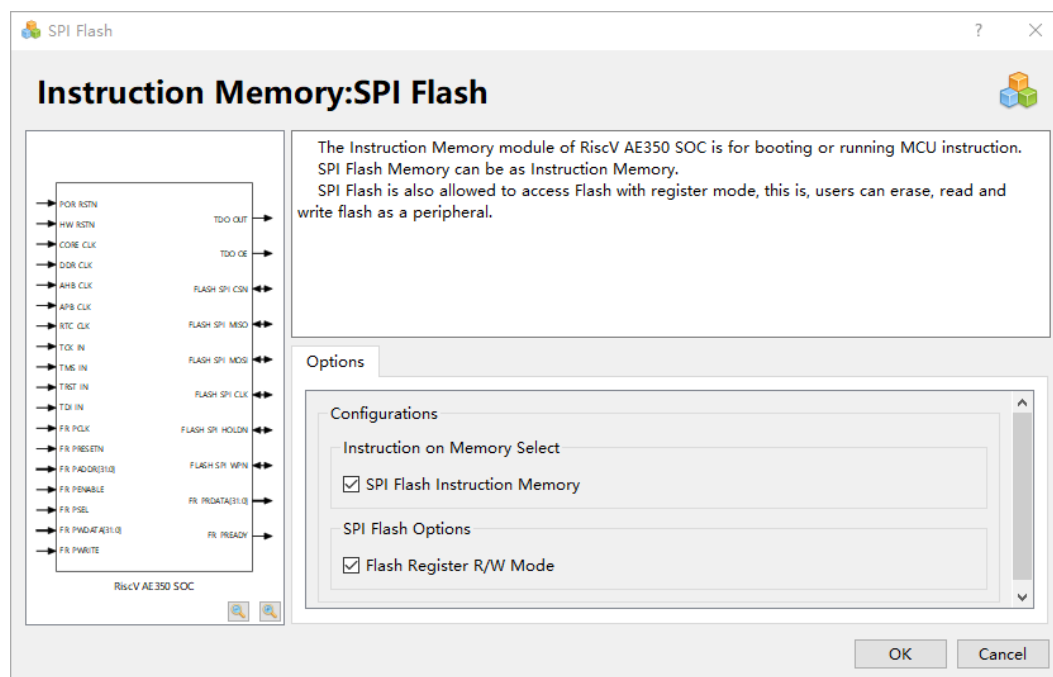
Double click to open "SPI Flash" to configure SPI Flash instruction memory, as shown in Figure 8-8.

If you select "SPI Flash Instruction Memory", the SPI Flash Memory is enabled as instruction memory, enabled by default.

If "SPI Flash Instruction Memory" is enabled, Flash Register R/W Mode can be configured. If "Flash Register R/W Mode" is selected, the APB bus interface in Flash Register Mode is enabled, and Flash can be used as a peripheral device to read, write and erase data through the register, disabled by default.

Regarding the application example of Flash Register R/W Mode, refer to "...\\example\\Flash".

Figure 8-8 Instruction Memory Configuration



ITCM

Double-click to open "ITCM", configure the options for ITCM instruction memory, as shown in Figure 8-9.

If "ITCM Instruction Memory" is selected, ITCM is enabled as instruction memory, disabled by default.

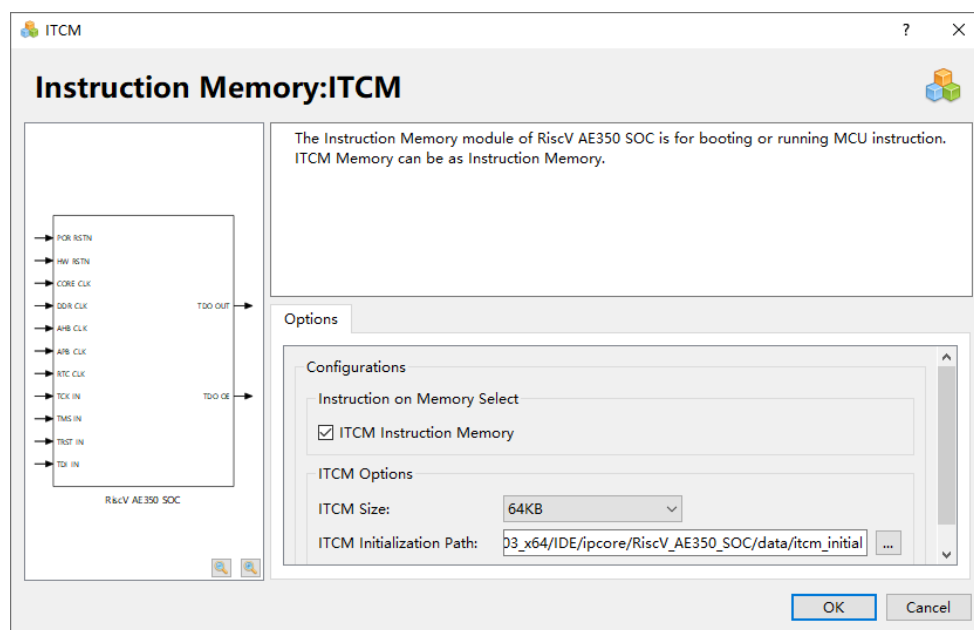
If "ITCM Instruction Memory" is enabled, you can configure the ITCM Size and ITCM Initialization Path.

ITCM Size can be configured as 32KB, 64KB, 128KB, 256KB, and 512KB, with a default of 64KB.

ITCM Initialization Path allows you to configure the initialization path of ITCM, i.e., the executable program for RiscV_AE350_SOC software programming, which is by default the preconfigured LED program under the installation path of Gowin Software. The initialization file of ITCM is in the form of hexadecimal files including itcm0, itcm1, itcm2, and itcm3, which can be generated by using the tool "...\\example\\Emb_TCM\\tool\\make_hex".

Regarding the application example of ITCM, refer to "...\\example\\Emb_TCM".

Figure 8-9 ITCM Configuration



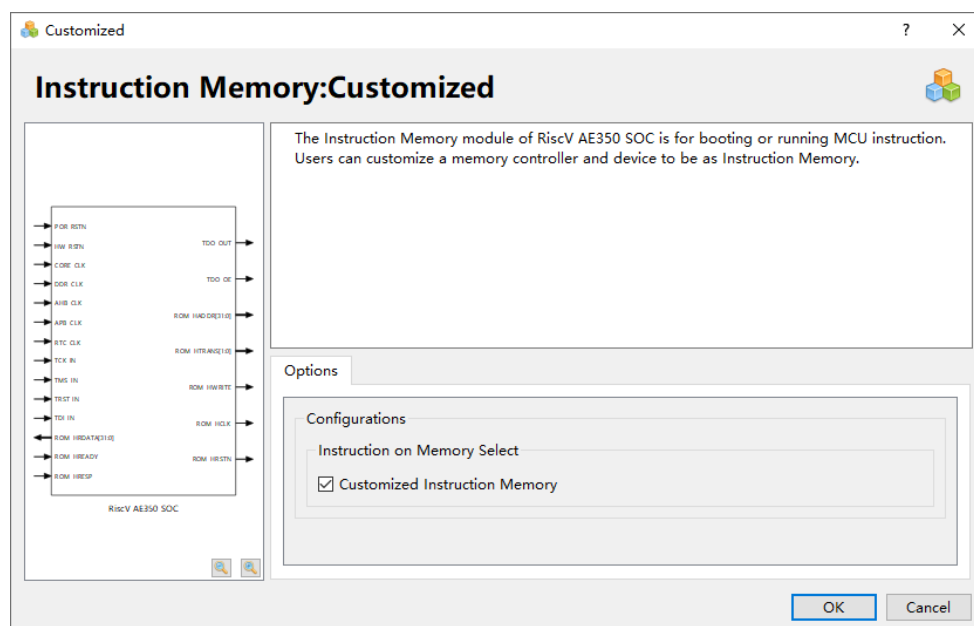
Customized

Double-click to open "Customized", configure the options for Customized instruction memory as shown in Figure 8-10.

If "Customized Instruction Memory" is selected, the ROM AHB bus interface is enabled, allowing users to customize memory as instruction memory, disabled by default.

Regarding the application example of Customized, refer to "...\\example\\Ext_Customized".

Figure 8-10 Customized Configuration



To summarize the applications of Instruction Memory, it can be divided into the following situations:

If you use the on-board off-chip SPI Flash Memory, choose "SPI Flash Instruction Memory".

If you use the FPGA logic's BlockRAM, choose "ITCM Instruction Memory".

If you want to use a different type of memory, choose "Customized Instruction Memory".

8.2.3 Data Memory

DDR3

Double-click to open "DDR3" and configure the options for DDR3 data memory, as shown in Figure 8-11.

If "DDR3 Data Memory" is selected, DDR3 Memory is enabled as the data memory, enabled by default.

If "DDR3 Data Memory" is enabled, you can configure DDR3 Settings and DDR3 Options. DDR3 Settings include "Default DDR3 Settings" and "Customized DDR3 Settings".

Default DDR3 Settings represent the default parameter configuration of the DDR3 Memory Interface IP pre-installed under the installation path of Gowin Software.

If users want to choose different parameter configurations for the DDR3 Memory Interface, a user interface for this purpose is provided. Users can also configure various parameter options in the "DDR3 Memory Interface" IP. Then, place the generated DDR3 Memory Interface IP parameter configuration file "gwmc_param.v" under the installation path "...\\IDE\\ipcore\\RiscV_AE350_SOC\\data\\ddr3_custom_settings" of Gowin Software. Finally, select "Customized DDR3 Settings" to generate the RiscV_AE350_SOC IP, which is the user-customized DDR3 Memory Interface.

DDR3 Options include "Shared DDR3: Logic R/W Mode", "Addr Width", and "Data Width".

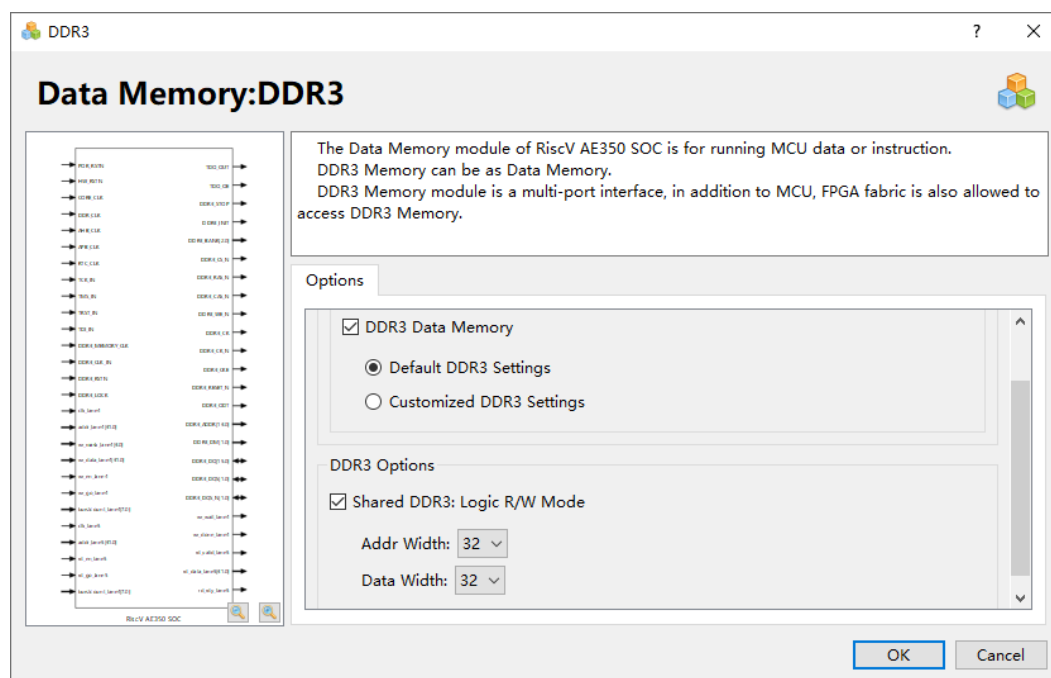
Shared DDR3: Logic R/W Mode indicates FPGA fabric-side read/write DDR3 Memory, allowing both MCU side and FPGA fabric side to access DDR3 Memory simultaneously, disabled by default.

Addr Width indicates the width of the address bus, which can be configured as 32.

Data Width indicates the width of the data bus, which can be configured as 32 or 64, with a default of 32.

Regarding the application example of Shared DDR3: Logic R/W Mode, refer to "... \example\DDR3_Shared".

Figure 8-11 DDR3 Configuration



DTCM

Double-click to open "DTCM" and configure the options for DTCM data memory, as shown in Figure 8-12.

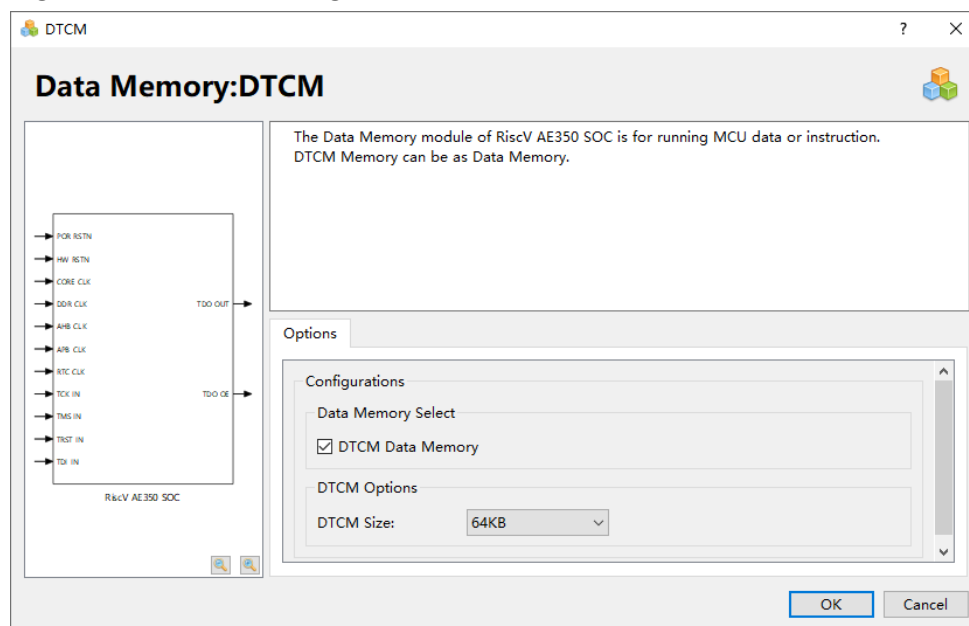
If "DTCM Data Memory" is selected, DTCM is enabled as the data memory, disabled by default.

If "DTCM Data Memory" is enabled, you can configure the DTCM Size.

DTCM Size can be configured as 32KB, 64KB, 128KB, 256KB, and 512KB, with a default of 64KB.

Regarding the application example of DTCM, see "... \example\Emb_TCM".

Figure 8-12 DTCM Configuration



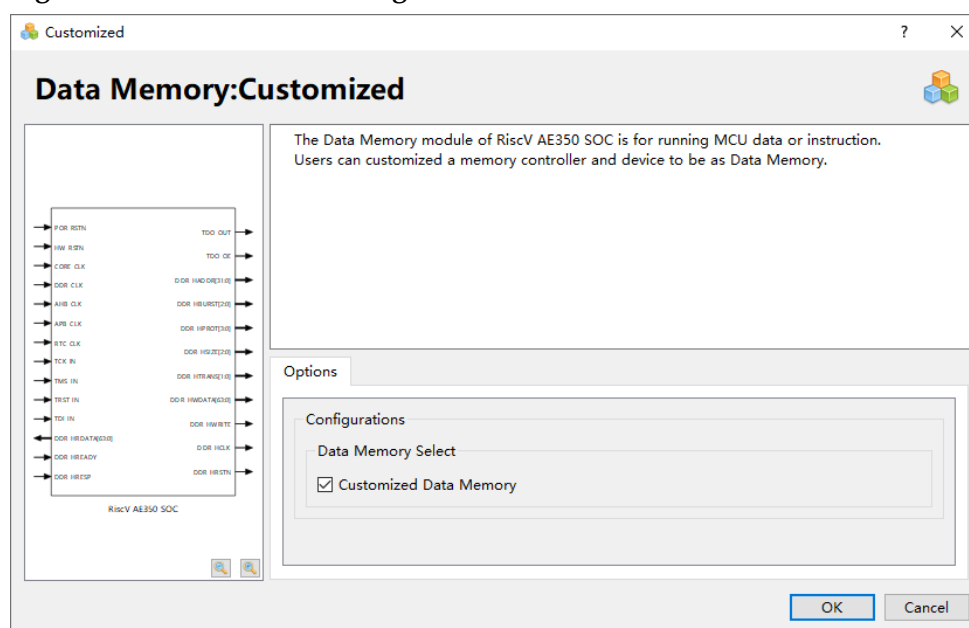
Customized

Double-click to open "Customized" and configure the options for Customized data memory, as shown in Figure 8-13.

If "Customized Data Memory" is selected, the DDR AHB bus interface is enabled, allowing users to customize memory as data memory, disabled by default.

Regarding the application example of Customized, see "...\\example\\Ext_Customized".

Figure 8-13 Customized Configuration



To summarize the applications of Data Memory, it can be divided into the following situations:

If you use on-board off-chip DDR3 Memory, select "DDR3 Data Memory" and then configure Default DDR3 Settings or Customized DDR3 Settings depending on the DDR3 Memory used.

If you want to use FPGA logic's BlockRAM, select "DTCM Data Memory".

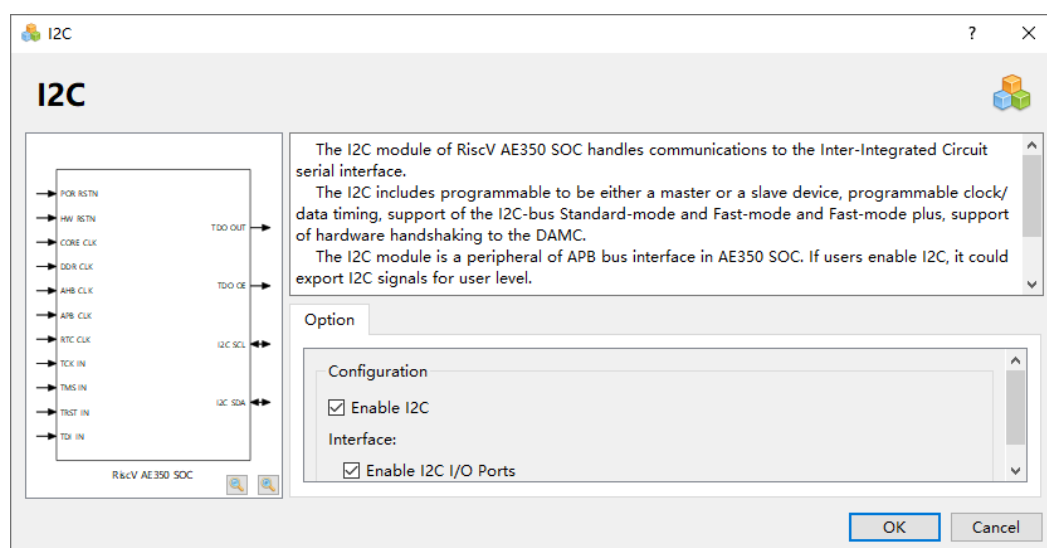
If you want to use a different type of memory, select "Customized Data Memory".

8.2.4 I2C

Double click to open "I2C", as shown in Figure 8-14.

- If "Enable I2C" is selected, I2C is enabled, disabled by default.
- If I2C is enabled, I2C port can be configured.
 - If "Enable I2C I/O Ports" is selected, I2C "INOUT" will be enabled, otherwise "IN" and "OUT" will be enabled, and the default is "INOUT".

Figure 8-14 I2C Configuration

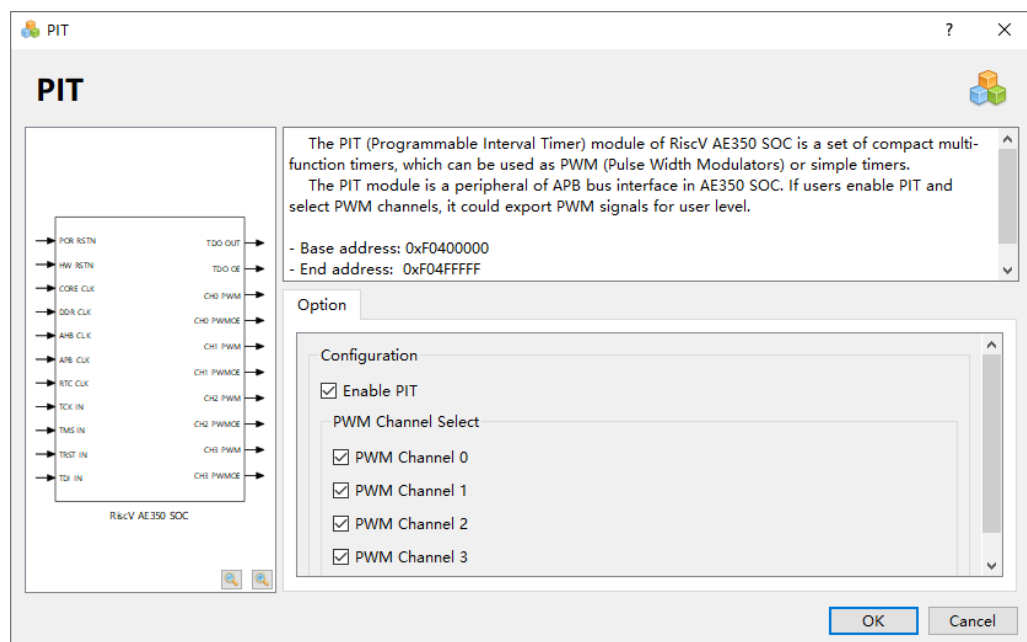


8.2.5 PIT

Double click to open "PIT", as shown in Figure 8-15.

- If "Enable PIT" is selected, PIT is enabled, disabled by default.
- If PIT is enabled, PIT channel can be configured.
 - If "PWM Channel 0" is selected, PWM Channel 0 is enabled, disabled by default.
 - If "PWM Channel 1" is selected, PWM Channel 1 is enabled, disabled by default.
 - If "PWM Channel 2" is selected, PWM Channel 2 is enabled, disabled by default.
 - If "PWM Channel 3" is selected, PWM Channel 3 is enabled, disabled by default.

Figure 8-15 PIT Configuration

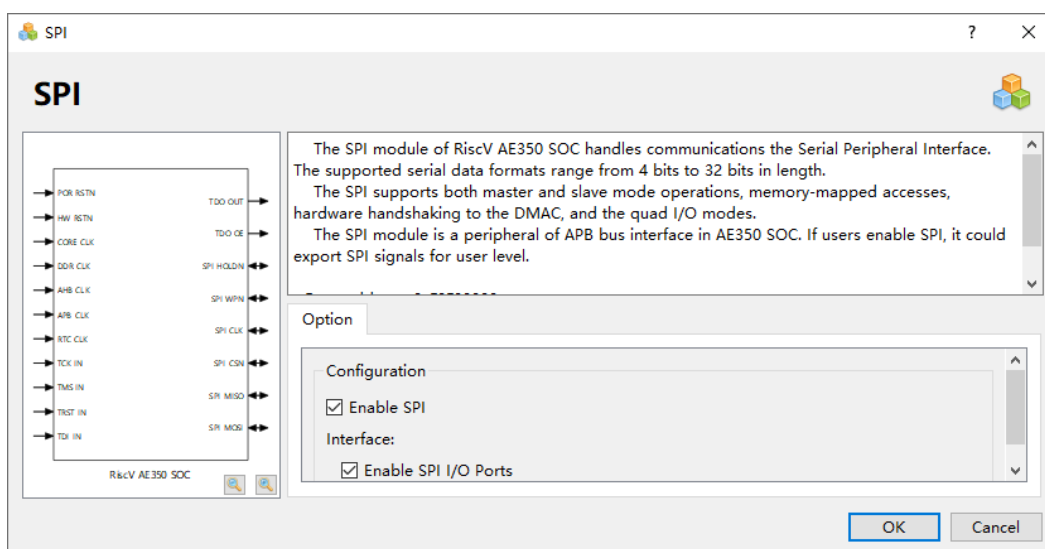


8.2.6 SPI

Double click to open SPI, as shown in Figure 8-16.

- If "Enable SPI" is selected, SPI is enabled, disabled by default.
- If SPI is enabled, SPI port can be configured.
 - If "Enable SPI I/O Ports" is selected, SPI "INOUT" will be enabled, otherwise "IN", "OUT", and "OE" will be enabled, and the default is "INOUT".

Figure 8-16 SPI Configuration

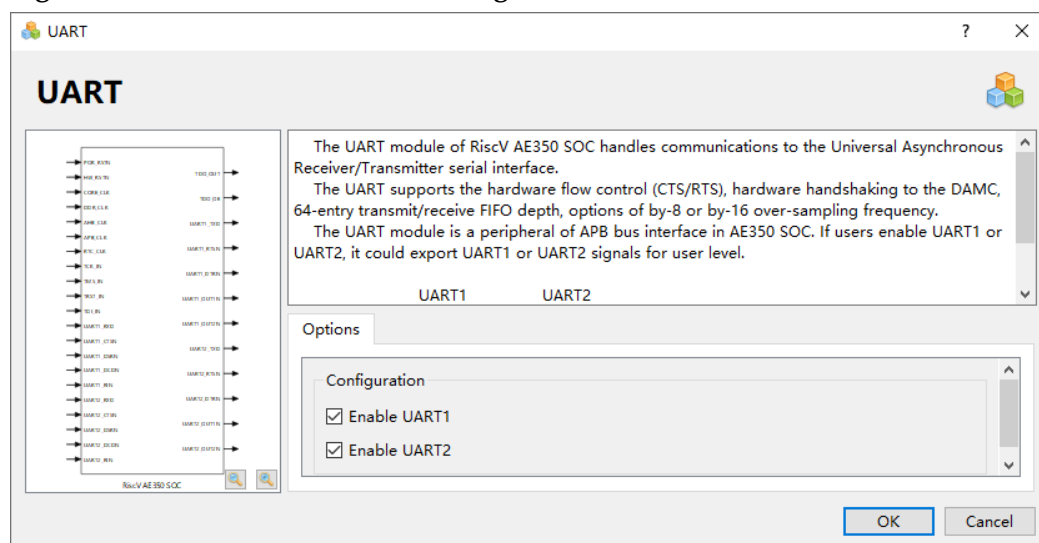


8.2.7 UART1 and UART2

Double click "UART1" or "UART2" to open UART1 or UART2, as shown in Figure 8-17.

- If "Enable UART1" is selected, UART1 is enabled, disabled by default.
- If "Enable UART2" is selected, UART2 is enabled, disabled by default.

Figure 8-17 UART1 and UART2 Configuration

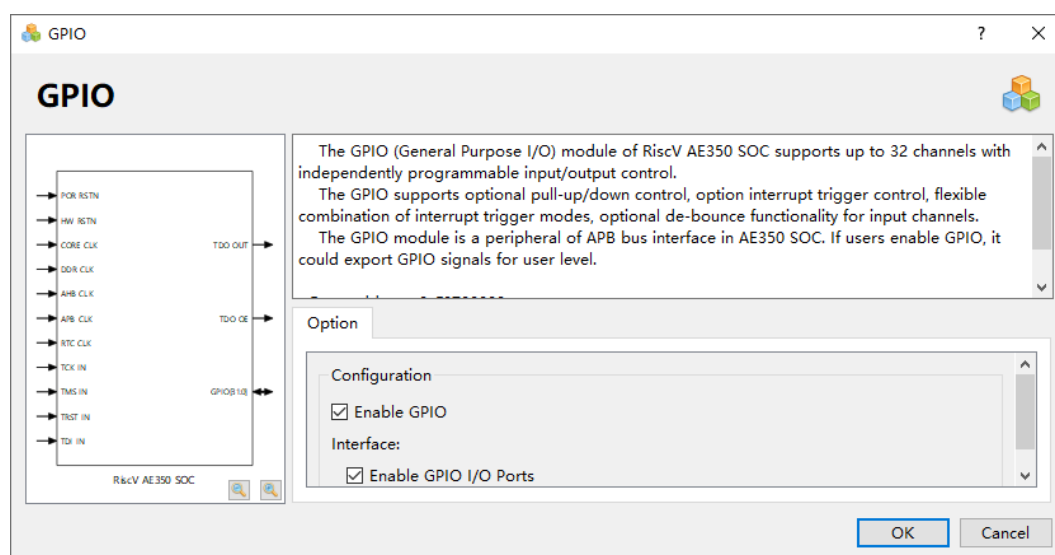


8.2.8 GPIO

Double click to open "GPIO", as shown in Figure 8-18.

- If "Enable GPIO" is selected, GPIO is enabled, disabled by default.
- If GPIO is enabled, GPIO port can be configured.
 - If "Enable GPIO I/O Ports" is selected, GPIO "INOUT" will be enabled, otherwise "IN", "OUT", and "OE" will be enabled, and the default is "INOUT".

Figure 8-18 GPIO Configuration

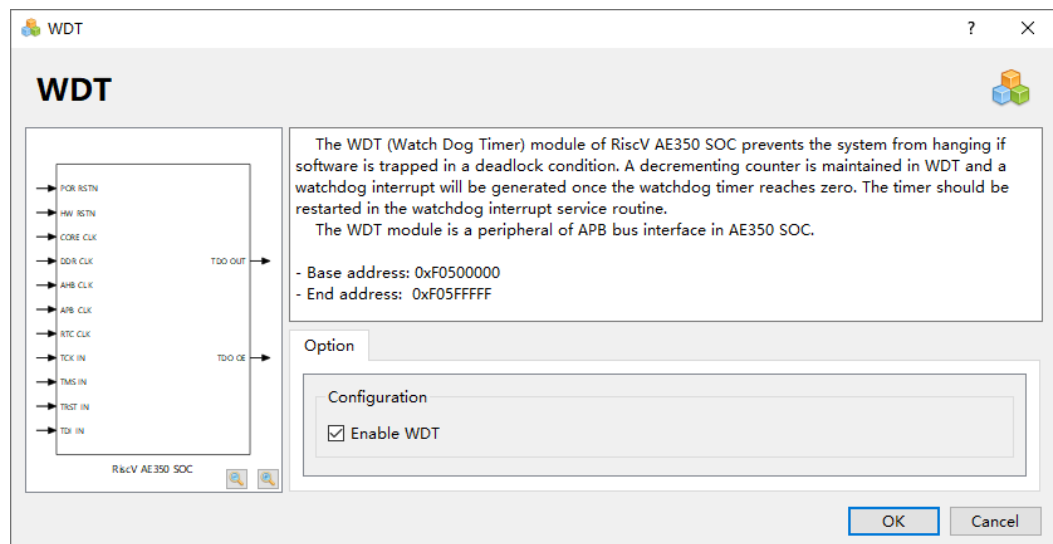


8.2.9 WDT

Double click to open "WDT", as shown in Figure 8-19.

If "Enable WDT" is selected, WDT is enabled, disabled by default.

Figure 8-19 WDT Configuration

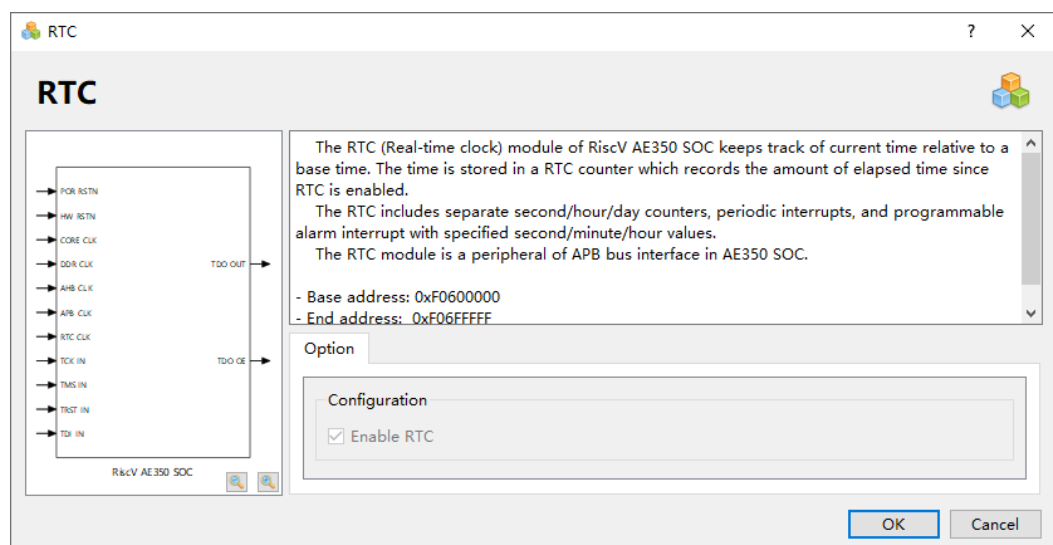


8.2.10 RTC

Double click to open RTC, as shown in Figure 8-20.

RTC is enabled by default and cannot be disabled.

Figure 8-20 RTC Configuration

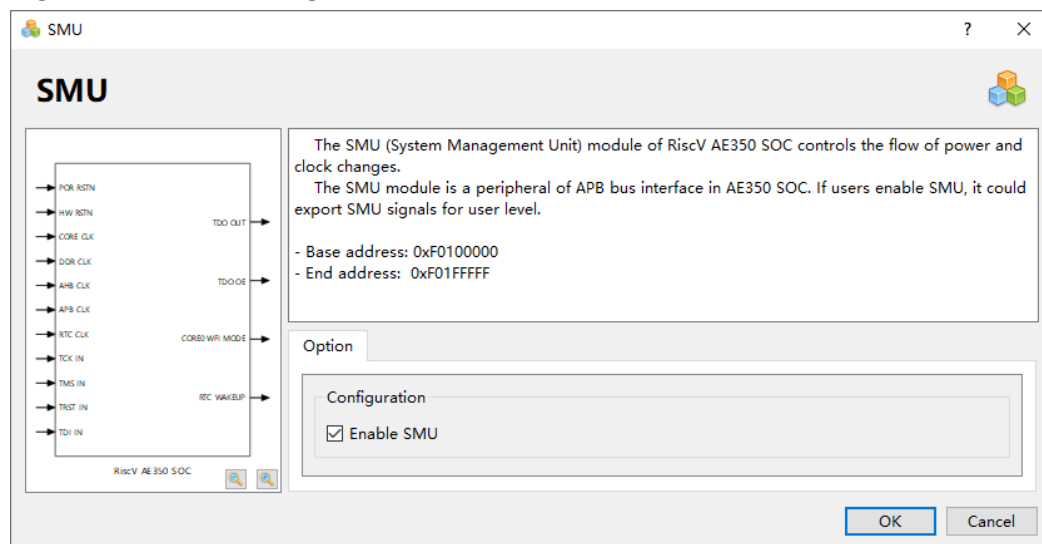


8.2.11 SMU

Double click to open "SMU", as shown in Figure 8-21.

If "Enable SMU" is selected, SMU is enabled, disabled by default.

Figure 8-21 SMU Configuration

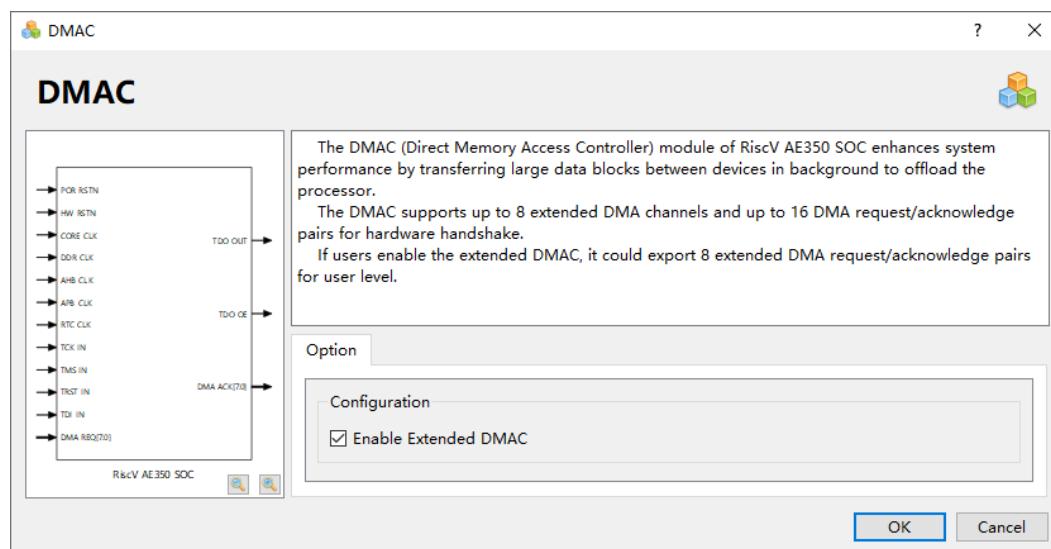


8.2.12 DMAC

Double click to open "DMAC", as shown in Figure 8-22.

If "Enable Extended DMAC" is selected, the extended 8 groups of DMA request/acknowledge signals are enabled, disabled by default.

Figure 8-22 DMAC Configuration



8.2.13 Extended APB Slave

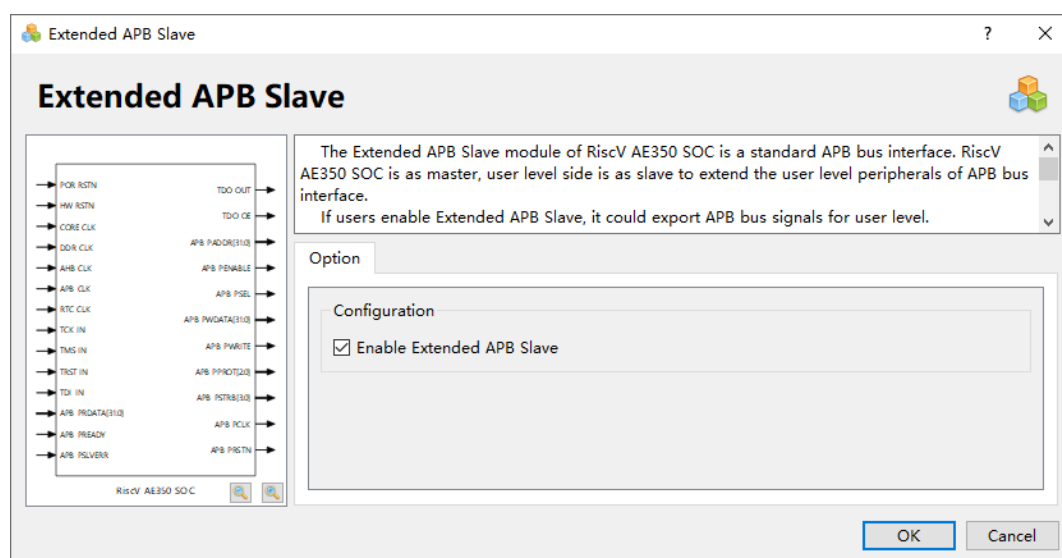
Double click to open "Extended APB Slave", as shown in Figure 8-23.

If "Enable Extended APB" is selected, the extended 32-bit APB Interface is enabled, disabled by default.

Through Extended APB Slave, APB interface peripherals implemented by FPGA logic extension can be connected.

Regarding the application example of Extended APB Slave, see "...\\example\\SD_Host", "...\\example\\APB_Slave_Bridge", "...\\example\\Flash".

Figure 8-23 Extended APB Slave Configuration



8.2.14 Extended AHB Slave

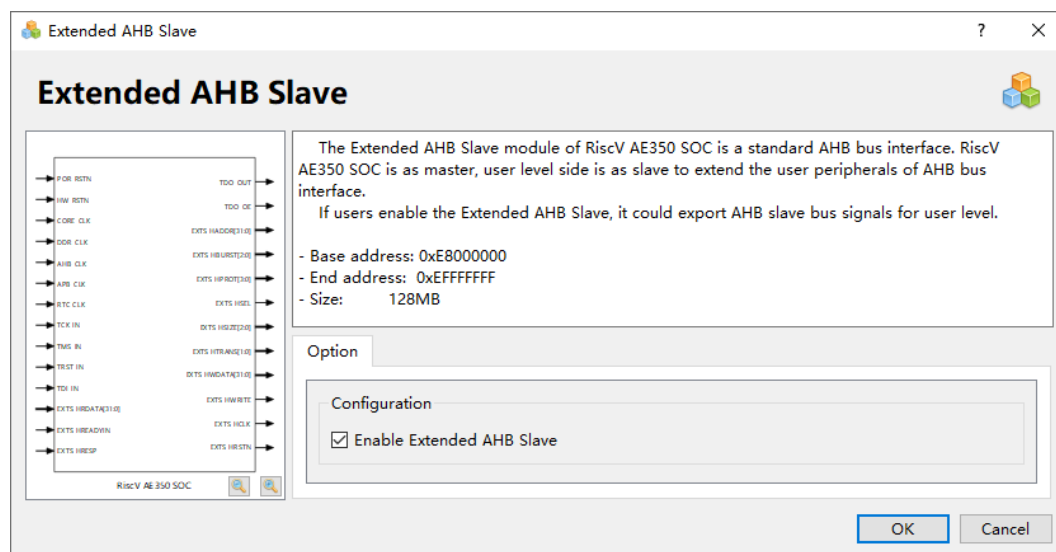
Double click to open "Extended AHB Slave", as shown in Figure 8-24.

If "Enable Extended AHB" is selected, the extended 32-bit AHB Interface is enabled, disabled by default.

Through Extended AHB Slave, AHB interface peripherals implemented by FPGA logic extension can be connected.

Regarding the application example of Extended AHB Slave, see "...\\example\\Ethernet", "...\\example\\AHB_Slave_Bridge", "...\\example\\CAN", "example\\AXI4_Lite".

Figure 8-24 Extended AHB Slave Configuration

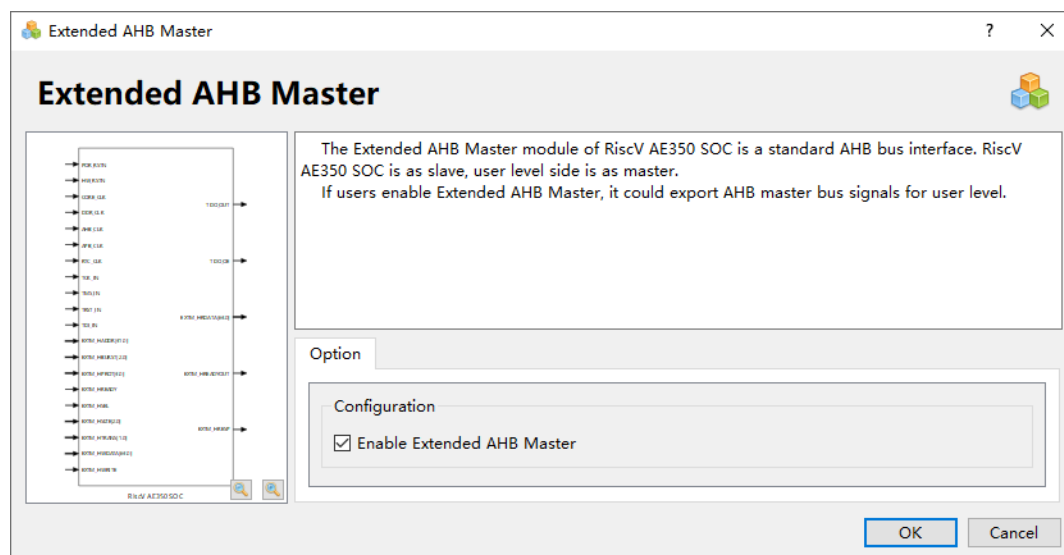


8.2.15 Extended AHB Master

Double click to open "Extended AHB Master", as shown in Figure 8-25.

If "Enable Extended AHB Master" is selected, 64-bit AHB Master interface is enabled, disabled by default.

Figure 8-25 Extended AHB Master Configuration



With this AHB Master interface, the AE350_SOC system can act as a Slave and the FPGA fabric system can act as a Master, and the FPGA fabric system accesses the ILMs and DLMs within the AE350_SOC system, thus expanding the storage space of the FPGA fabric system.

Regarding the application example of Extended AHB Master, see

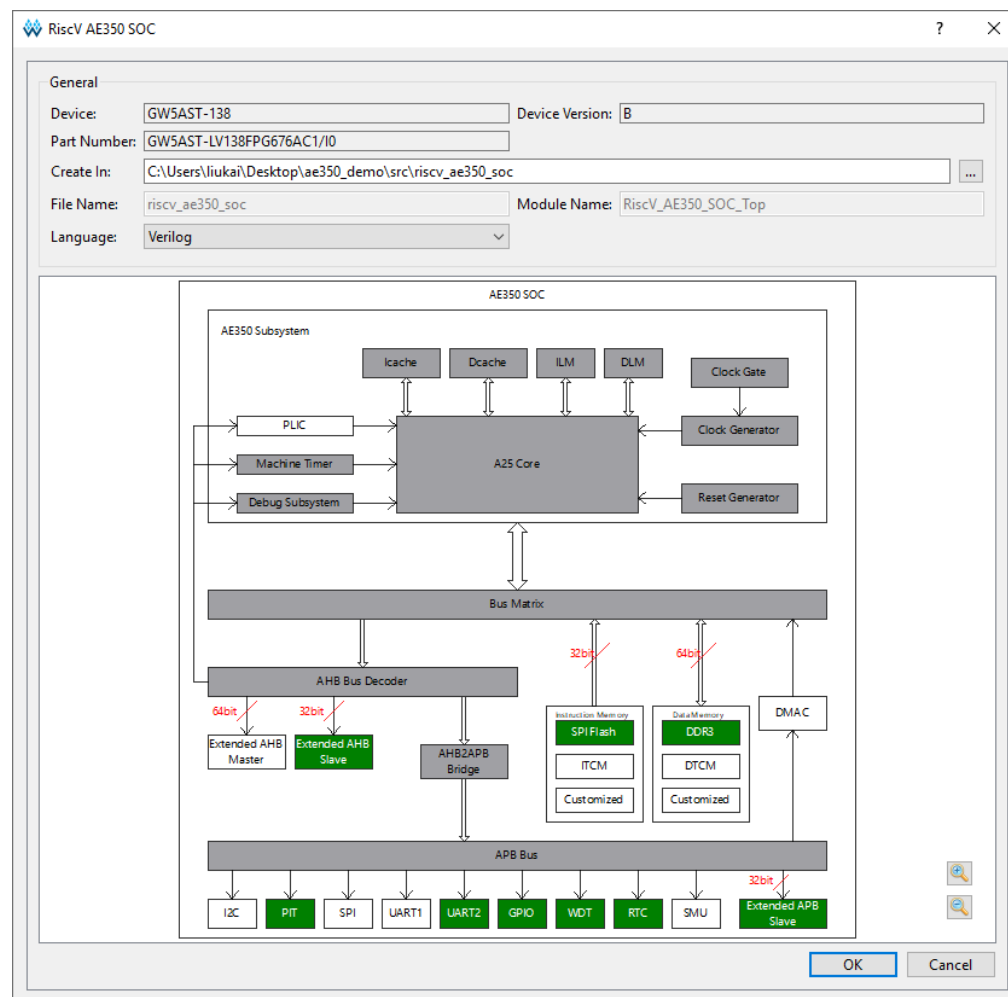
"...\example\AHB_Master".

8.3 User Design

8.3.1 RiscV_AE350_SOC IP Design

Use IP Core Generator tool in Gowin Software to configure RiscV_AE350_SOC IP, then click "OK", as shown in Figure 8-26.

Figure 8-26 RiscV_AE350_SOC IP Design



8.3.2 PLL_ADV IP Design

Configure the PLL_ADV IP option to generate a PLL_ADV IP design to provide clocking resources for the RiscV_AE350_SOC IP using IP Core Generator.

- Select "Tools > IP Core Generator" on the main menu or click "🔧" on the toolbar to open the IP Core Generator; select "Hard Module > CLOCK > PLL_ADV 1.0", and refer to Table 4-1 and Table 4-2 to configure and generate the PLL_ADV IP, as shown in Figure 8-27.

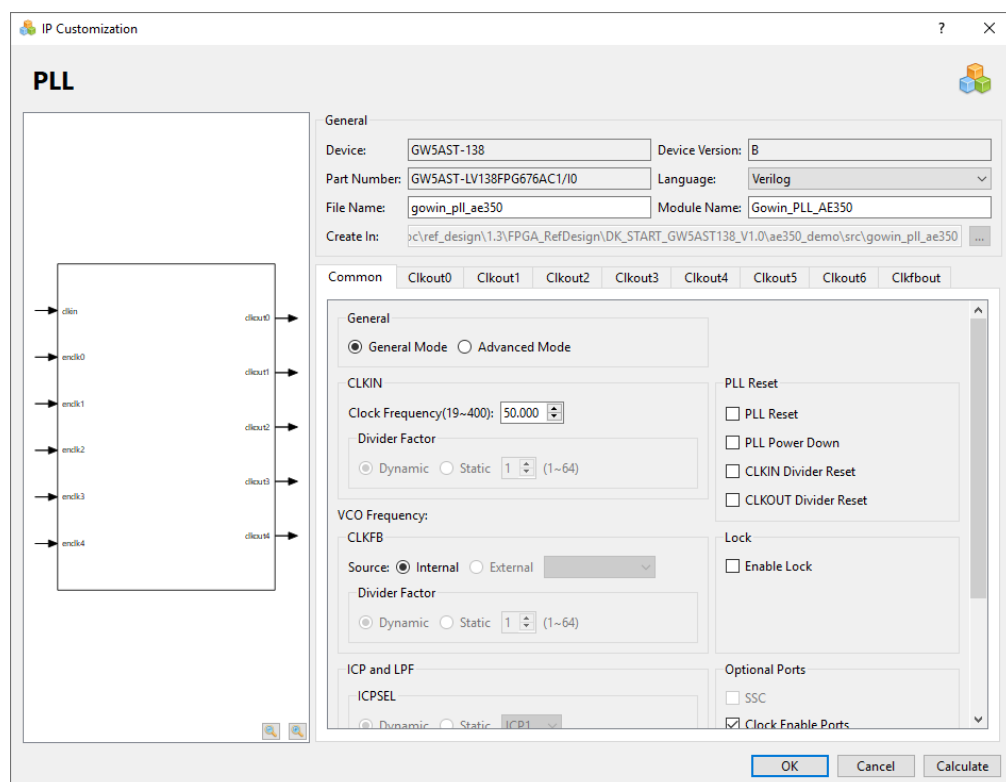
For example: PLL_ADV IP for the RiscV_AE350_SOC core:

- Clkout0: DDR clock
- Clkout1: CORE clock
- Clkout2: AHB clock
- Clkout3: APB clock
- Clkout4: RTC clock

If DDR3 Memory and Shared DDR3: Logic R/W Mode are selected for RiscV_AE350_SOC Data Memory, PLL_ADV IP for RiscV_AE350_SOC DDR3 Memory are as follows:

- Clkout0: DDR3 input clock
- Clkout1: DDR3 R/W clock
- Clkout2: DDR3 memory clock

Figure 8-27 PLL_ADV IP Design



Note!

- RiscV_AE350_SOC's core clock is provided by "PLL_R[0] > clkout1" directly; it is necessary to use "**clkout1**" of PLL_ADV IP to generate the core clock.
- The DDR3 Memory clock of RiscV_AE350_SOC is provided by "PLL_L[0] > clkout2", it is recommended to use "**clkout2**" of PLL_ADV IP to generate the DDR3 Memory clock.
- DDR3 R/W clock $\geq$ (DDR3 memory clock/4).

8.3.3 User Design

In the top module of hardware design, instantiate the RiscV_AE350_SOC IP, instantiate the PLL_ADV IP, add other user logic designs, and connect the modules to form a complete hardware design.

8.4 Constraints

8.4.1 Physical Constraints

Use the physical constraints tool FloorPlanner in Gowin Software to add physical constraints with the reference to the development board.

For example, the JTAG interface connection for the DK_START_GW5AST-LV138FPG676A V1.0 development Board reference design and Andes AICE-MINI+ debugger is as shown in Table 8-1.

Table 8-1 Physical Constraints for JTAG in Reference Design (Andes AICE-MINI+ Debugger)

JTAG Interface	DVK Board	Andes AICE-MINI+ Debugger
GND	J3-8	P3
TMS	J3-3	P4
TCK	J3-4	P6
VREF (3.3V)	J3-7	P7
TRST	J3-5	P10
TDO	J3-6	P11
TDI	J60-3	P12

For example, the JTAG interface connection for the DK_START_GW5AST-LV138FPG676A V1.0 development Board reference design and Gowin FTDI USB programmer is as shown in Table 8-2.

Table 8-2 Physical Constraints for JTAG in Reference Design (Gowin FTDI USB Programmer)

JTAG Interface	DVK Board	Gowin FTDI USB Programmer
GND	J3-8	P2
TMS	J3-3	P9
TCK	J3-4	P1
VREF (3.3V)	J3-7	P6

JTAG Interface	DVK Board	Gowin FTDI USB Programmer
TDO	J3-6	P5
TDI	J60-3	P3

For example, the JTAG interface connection for the DK_START_GW5AST-LV138FPG676A V1.0 development Board reference design and Olimex debugger is as shown in Table 8-3.

Table 8-3 Physical Constraints for JTAG in Reference Design (Olimex Debugger)

JTAG Interface	DVK Board	Olimex Debugger
GND	J3-8	P4
TMS	J3-3	P7
TCK	J3-4	P9
VREF (3.3V)	J3-7	P1
TRST	J3-5	P3
TDO	J3-6	P13
TDI	J60-3	P5

Note!

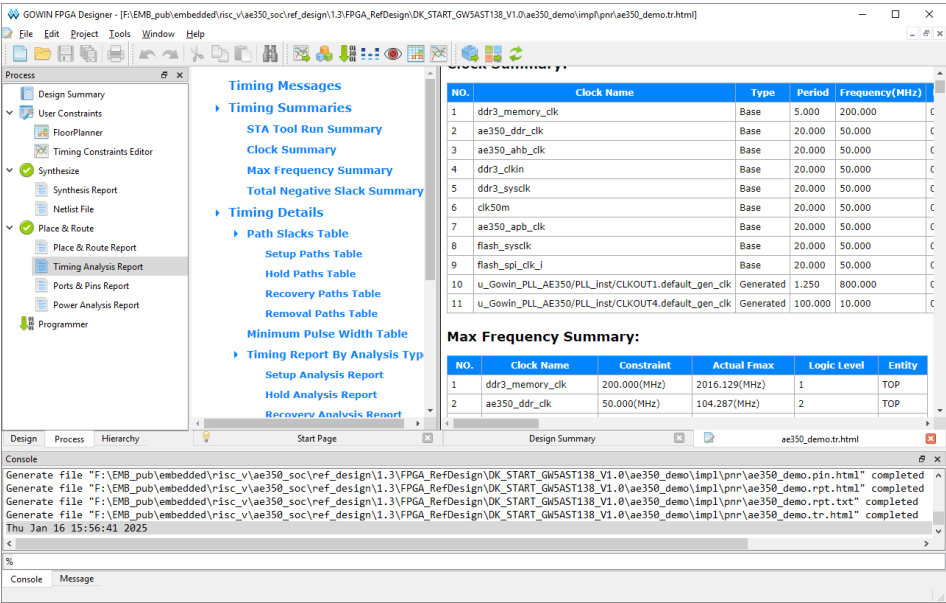
- RiscV_AE350_SOC's core clock is provided by "PLL_R[0] > clkout1" directly; it must constrain the location of this PLL_ADV IP to "**PLL_R[0]**". For example, **INS_LOC "u_Gowin_PLL_AE350/PLL_inst" PLL_R[0]**.
- The DDR3 Memory clock of RiscV_AE350_SOC is provided by "PLL_L[0] > clkout2", it is recommended to constrain the location of this PLL_ADV IP to "**PLL_L[0]**". For example, **INS_LOC "u_Gowin_PLL_DDR3/PLL_inst" PLL_L[0]**.

8.4.2 Timing Constraints

Use the timing constraints tool Timing Constraints Editor in Gowin Software to add timing constraints with the reference to the software timing analysis report.

Click "Place & Route > Timing Analysis Report" to check the timing analysis report, as shown in Figure 8-28.

Figure 8-28 Timing Analysis Report



8.5 Configuration

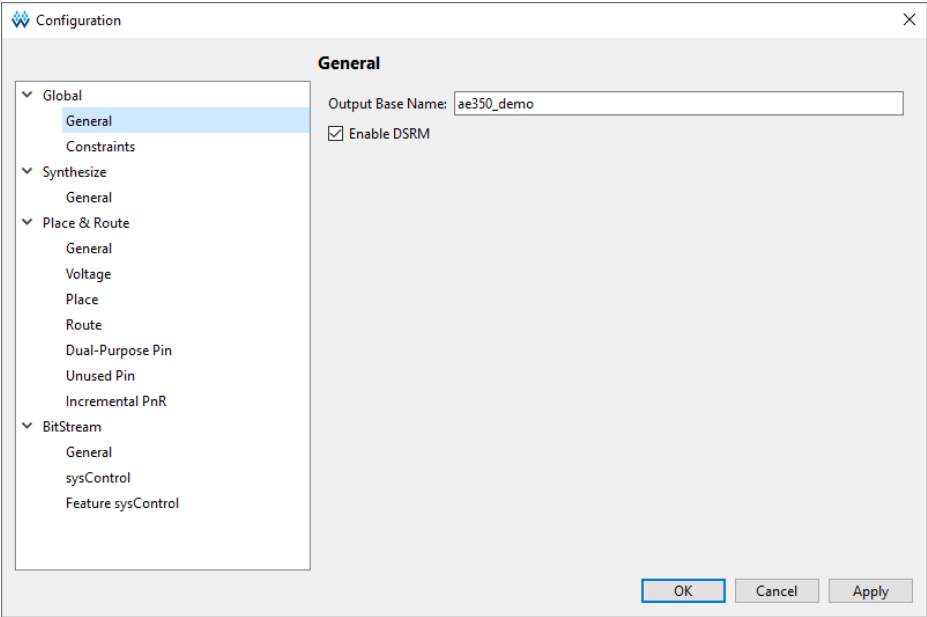
8.5.1 Global Option

Global option includes "General" and "Constraints".

"General" option is as shown in Figure 8-29. For example:

- Enable DSRAM: If enabled, DSRAM is synthesized.
- Output Base Name: Typically, the name of the hardware design project.

Figure 8-29 General Configuration

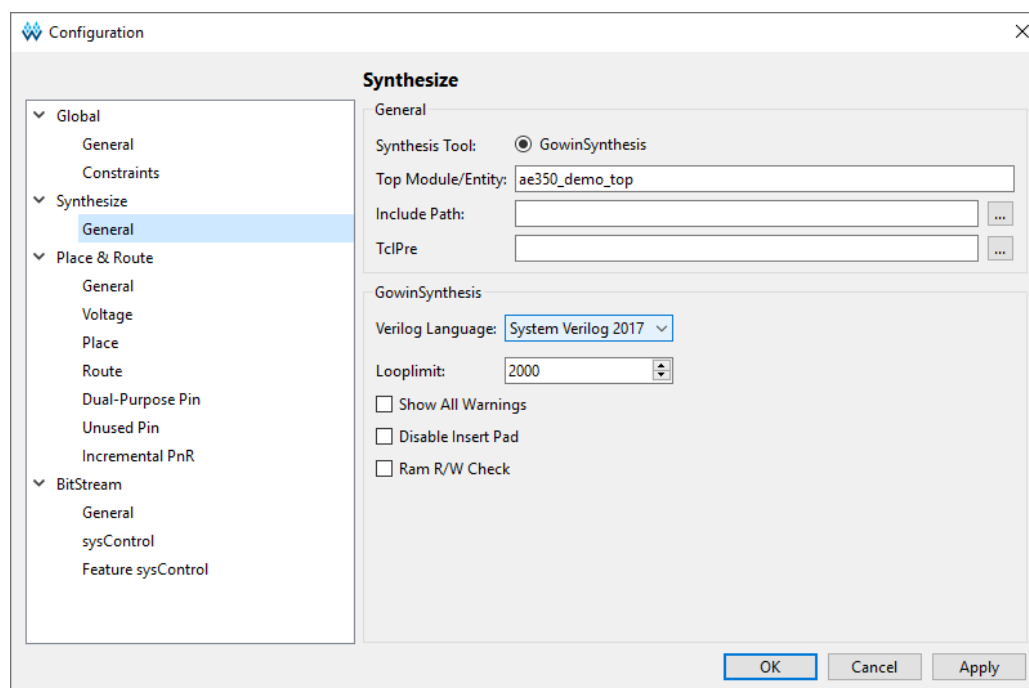


8.5.2 Synthesis Option

The synthesis configuration is as shown in Figure 8-30. For example:

- Configure "Top Module/Entity" according to the top module name in the hardware design.
- Configure "Include Path" according to the actual header file path in the hardware design.
- Configure Verilog Language, such as System Verilog 2017.

Figure 8-30 Synthesis Configuration



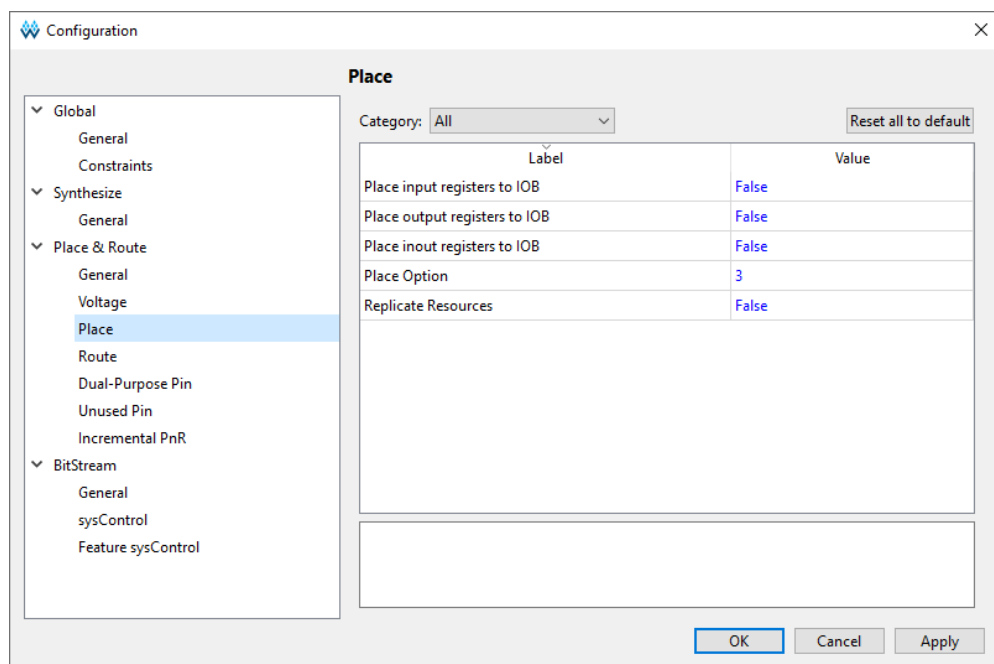
8.5.3 Place & Route Option

Place & Route option includes "General", "Voltage", "Place", "Route", "Dual-Purpose Pin" and "Unused Pin".

You can configure these parameters according to the actual requirements of the hardware design, for example:

- The value in Place option is 3, as shown in Figure 8-31.
- The value in Route option is 2, as shown in Figure 8-32.
- Use "SSPI", "MSPI" and "CPU" as regular IOs in Dual-Purpose Pin option, as shown in Figure 8-33.

Figure 8-31 Place Option



Configuration

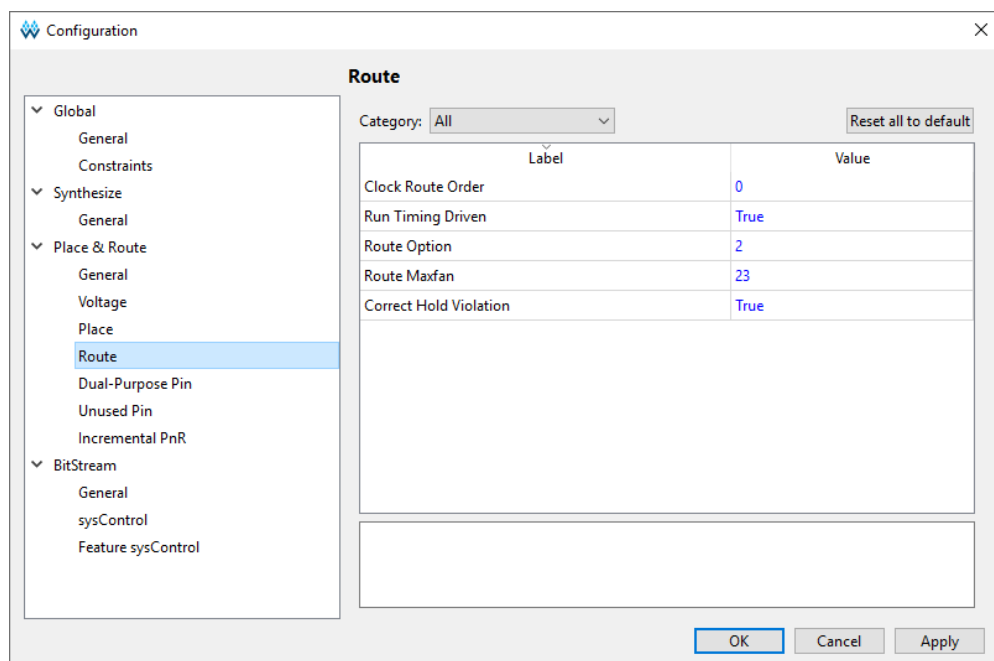
Place

Category: All Reset all to default

Label	Value
Place input registers to IOB	False
Place output registers to IOB	False
Place inout registers to IOB	False
Place Option	3
Replicate Resources	False

OK Cancel Apply

Figure 8-32 Route Option



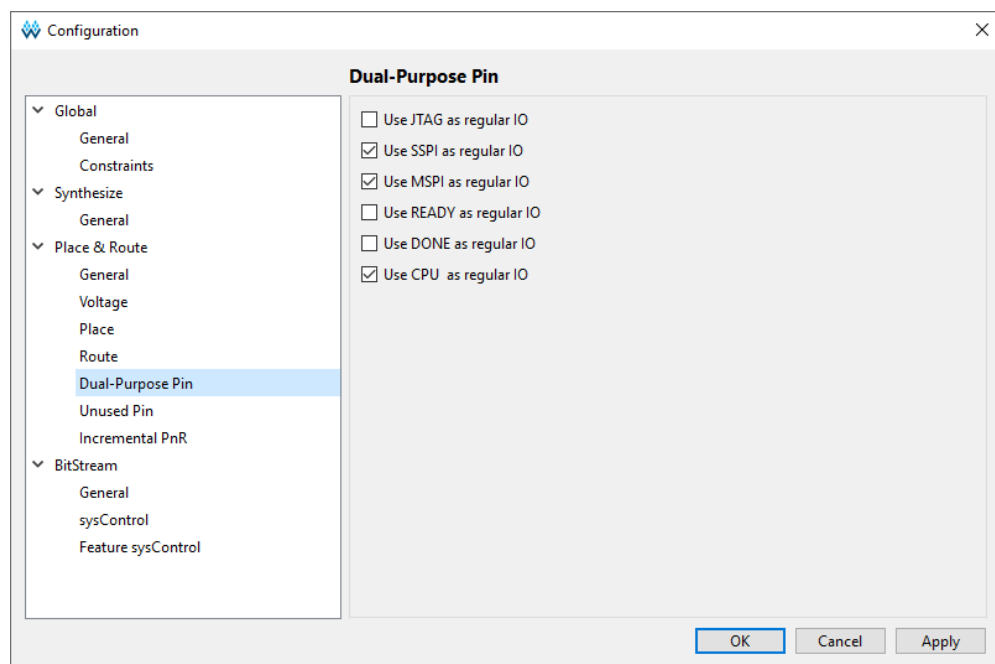
Configuration

Route

Category: All Reset all to default

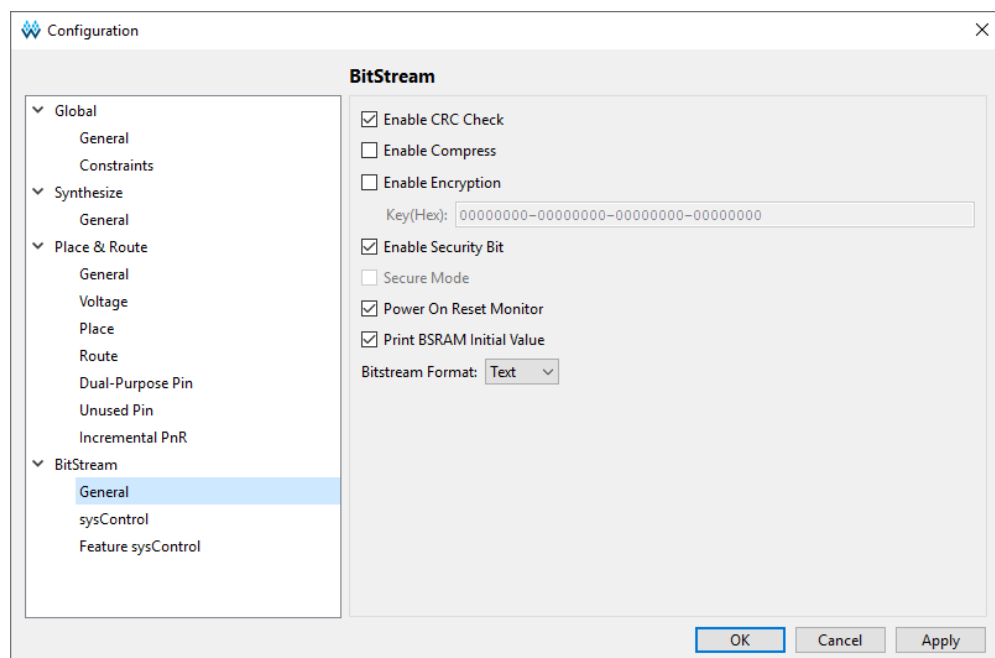
Label	Value
Clock Route Order	0
Run Timing Driven	True
Route Option	2
Route Maxfan	23
Correct Hold Violation	True

OK Cancel Apply

Figure 8-33 Dual-Purpose Pin Option

8.5.4 BitStream Option

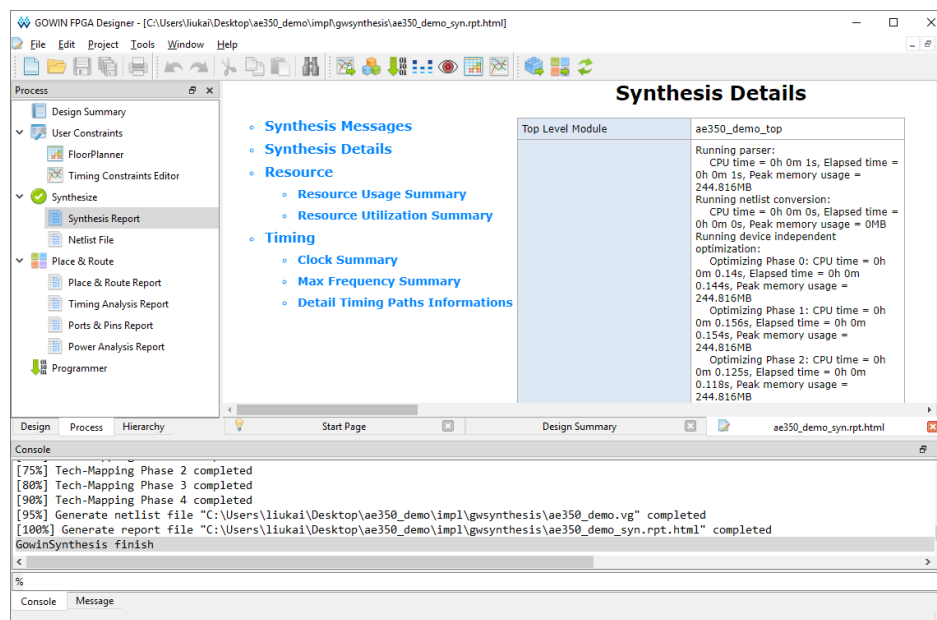
The BitStream option includes General, sysControl, and Feature sysControl, as shown in Figure 8-34.

Figure 8-34 BitStream Configuration

8.6 Synthesis

Run the synthesis tool GowinSynthesis to synthesize RiscV_AE350_SOC hardware design and generate the netlist file, as shown in Figure 8-35.

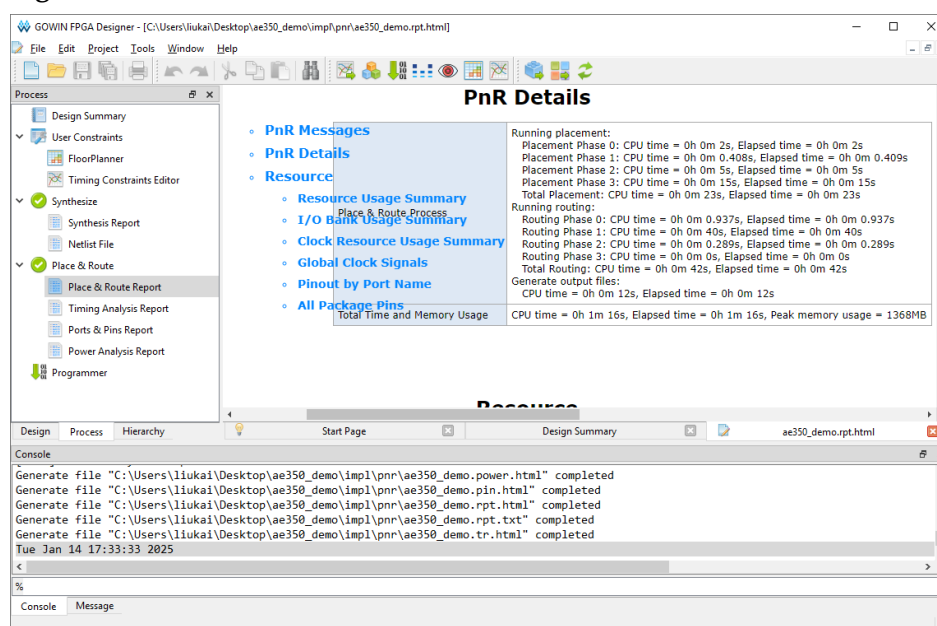
Figure 8-35 Synthesize



8.7 Place & Route

Run the Place & Route tool in Gowin software and generate the bitstream files, as shown in Figure 8-36.

Figure 8-36 Place & Route



8.8 Download

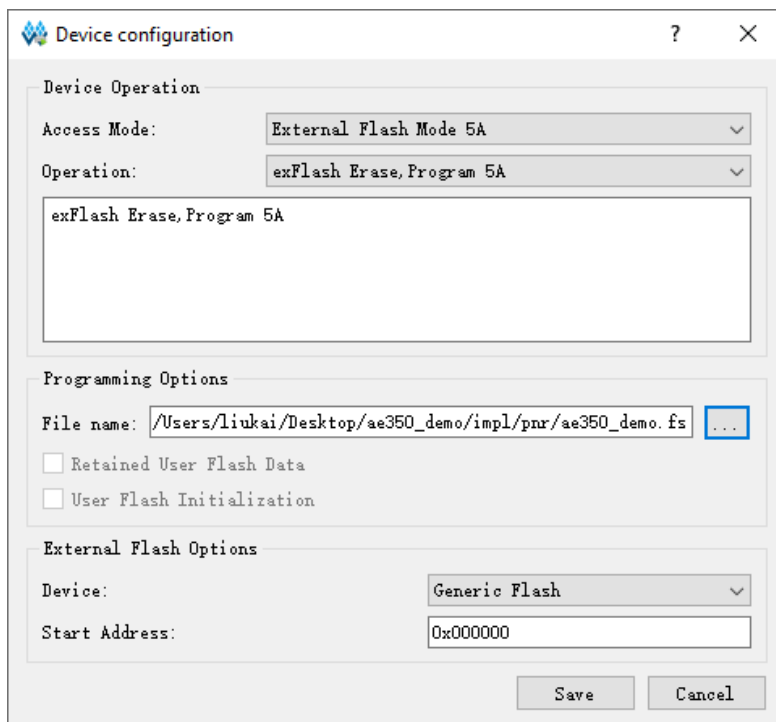
Run Programmer to download the bitstream file.

Click "Process > Program Device" in Gowin Software, or click "Tools > Programmer" on main menu, or click "🔧" to open Programmer.

Select "Edit > Configure Device" on Programmer menu bar or "Configure Device" (🔧) on the tool bar to open the "Device configuration", as shown in Figure 8-37 Download Configuration.

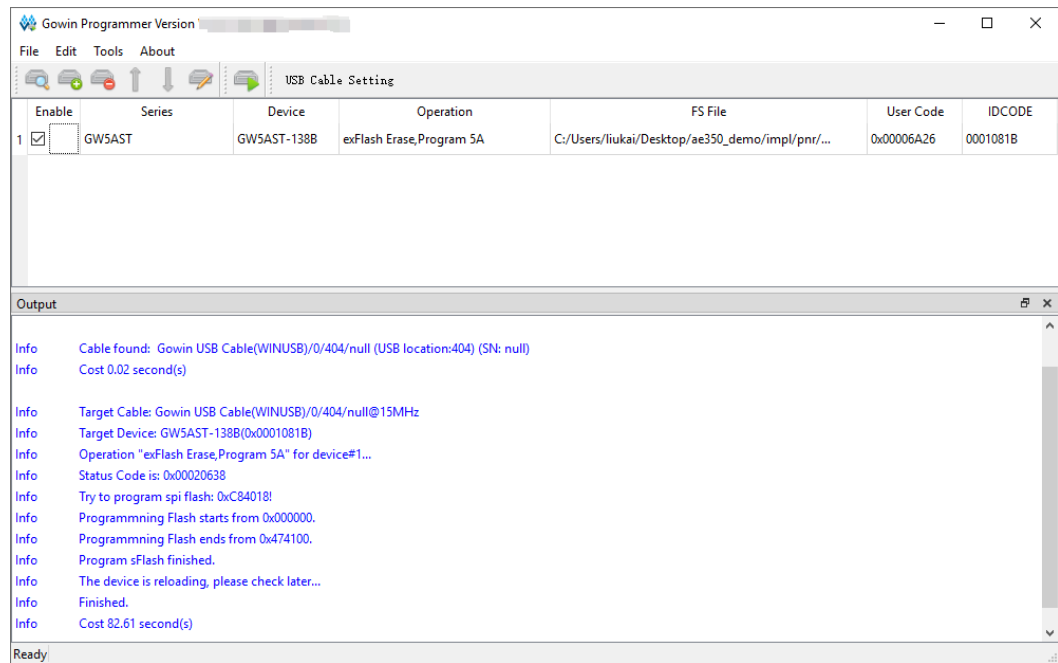
- Select "External Flash Mode 5A" in "Access Mode" drop-down list.
- Select "exFlash Erase, Program 5A" or "exFlash Erase, Program, Verify 5A" in "Operation" drop-down list.
- Import bitstream files using "Programming Options > File name" option.
- Select "Generic Flash" in "External Flash Options > Device" option.
- Configure the start address as "0x000000" in "External Flash Options > Start Address" option.
- Click "Save" to complete the configuration.

Figure 8-37 Download Configuration



After completing the download configuration, click " on the toolbar to download the bitstream file, as shown in Figure 8-38.

Figure 8-38 Download Bitstream File



9 Reference Design

Gowin RiscV_AE350_SOC provides reference design for hardware design:

- ...\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_demo
- ...\\ref_design\\FPGA_RefDesign\\Tang_MEGA_138K_Pro_Dock\\ae350_demo
- ...\\example\\Ethernet\\ref_design\\FPGA_RefDesign\\Tang_MEGA_138K_Pro_Dock\\ae350_ext_ethernet
- ...\\example\\CAN\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_ext_can
- ...\\example\\AHB_Slave_Bridge\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ahb_to_ahb_16_bridge
- ...\\example\\Flash\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_ext_flash
- ...\\example\\SD_Host\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_ext_sd
- ...\\example\\APB_Slave_Bridge\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\apb_to_apb_16_bridge
- ...\\example\\AHB_Master\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_ahb_master
- ...\\example\\Ext_Customized\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_customized
- ...\\example\\DDR3_shared\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_shared_ddr3
- ...\\example\\Emb_TCM\\ref_design\\FPGA_RefDesign\\DK_START_GW5AST138_V1.0\\ae350_emb_tcm
- ...\\example\\AXI4_Lite\\ref_design\\FPGA_RefDesign\\DK_START_GW5

AST138_V1.0\ae350_axi4_ram

- ...\\example\\AXI4_Lite\\ref_design\\FPGA_RefDesign\\DK_START_GW5
AST138_V1.0\ae350_axi4_reg
- ...\\example\\Realtime_Processing\\ref_design\\FPGA_RefDesign\\DK_S
TART_GW5AST138_V1.0\ae350_rpu

