



Gowin SPI Flash Interface (With External Flash) IP

User Guide

IPUG1015-1.1E, 06/08/2023

Copyright © 2023 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
02/22/2023	1.0E	Initial version published.
06/08/2023	1.1E	● IP name updated from SPI Flash Interface to SPI Flash Interface (With External Flash). ● Interface screenshots updated.

Contents

Contents	i
List of Figures	iii
List of Tables	v
1 About This Guide	1
1.1 Purpose	1
1.2 Related Documents	1
1.3 Terminology and Abbreviations	1
1.4 Support and Feedback	2
2 Overview	3
2.1 Overview	3
2.2 Features	3
2.3 Resource Utilization	4
3 Functional Description	5
3.1 System Block Diagram	5
3.2 Port List	7
3.3 Parameter Configuration	9
3.4 Register Description	9
3.4.1 SPI Transfer Format Register (0x10)	10
3.4.2 SPI Transfer Control Register (0x20)	11
3.4.3 SPI Command Register (0x24)	12
3.4.4 SPI Address Register (0x28)	13
3.4.5 SPI Data Register (0x2C)	13
3.4.6 SPI Control Register (0x30)	13
3.4.7 SPI Status Register (0x34)	14
3.4.8 SPI Interrupt Enable Register(0x38)	14
3.4.9 SPI Interrupt Status Register (0x3C)	14
3.4.10 SPI Interface Timing Register(0x40)	15
3.4.11 Configuration Register (0x7C)	15
3.5 Hardware Configuration Options	15
3.5.1 TX FIFO depth	15

3.5.2 RX FIFO depth.....	16
3.5.3 Memory-Mapped Read.....	16
3.5.4 SPI Interface Clock Parameters	16
3.6 Read and Write Operation Flow	16
3.6.1 Write Enable (06H) Operating Process	17
3.6.2 Write Disable (04H) Operating Process	17
3.6.3 Read Status Register (05H/35H) Operating Process.....	18
3.6.4 Write Status Register (01H) Operating Process.....	18
3.6.5 Read Data (03H) Operating Process.....	19
3.6.6 Page Program (02H) Operating Process	20
3.6.7 Sector Erase (20H) Operating Process.....	21
3.6.8 32 KB Block Erase (52H) Operating Process.....	22
3.6.9 64 KB Block Erase (D8H) Operating Process	23
3.6.10 60 Erase (60/ C7H) Operating Process.....	24
3.6.11 Read Manufacturer/Device ID (90H) Operating Process	24
3.6.12 Read JEDEC ID (9FH) Operating Process	25
3.6.13 Read Unique ID (4BH) Operating Process.....	26
3.6.14 Fast Read Dual Output (3BH) Operating Process	27
3.6.15 Fast Read Dual I/O (BBH) Operating Process	28
3.6.16 Read Manufacturer/Device ID Dual I/O (92H) Operating Process	29
3.6.17 Quad Page Program (32H) Operating Process.....	30
3.6.18 Fast Read Quad Output (6BH) Operating Process	32
3.6.19 Fast Read Quad I/O (EBH) Operating Process.....	33
3.6.20 Read Manufacturer/Device ID Quad I/O (94H) Operating Process	33
3.7 Timing Description	34
3.7.1 SPI Interface Timing	34
3.7.2 AHB Interface Timing	35
3.7.3 APB Interface Timing	36
4 Interface Configuration	38
5 Reference Design	41
6 File Delivery	42
6.1 Documentation.....	42
6.2 Design Source Code (Encryption)	42
6.3 Reference Design	42

List of Figures

Figure 3-1 System Block Diagram	6
Figure 3-2 Gowin SPI Flash Interface (With External Flash) IP I/O Ports Diagram.....	7
Figure 3-3 Write Enable Timing Diagram.....	17
Figure 3-4 Write Disable Timing Diagram	18
Figure 3-5 Read Status Register Timing Diagram	18
Figure 3-6 Write Status RegisterTiming Diagram	19
Figure 3-7 Read Data Timing Diagram	19
Figure 3-8 Page Program Timing Diagram	21
Figure 3-9 Sector Erase Timing Diagram	22
Figure 3-10 32 KB Block Erase Timing Diagram	23
Figure 3-11 64KB Block Erase Timing Diagram	23
Figure 3-12 Chip Erase Timing Diagram	24
Figure 3-13 Read Manufacture ID/Device ID Timing Diagram	25
Figure 3-14 Read JEDEC ID Timing Diagram	26
Figure 3-15 Read Unique ID Timing Diagram.....	27
Figure 3-16 Fast Read Dual Output Timing Diagram	28
Figure 3-17 Fast Read Dual Timing Diagram	29
Figure 3-18 Read Manufacturer/Device ID Dual I/O Timing Diagram	30
Figure 3-19 Quad Page Program Timing Diagram	31
Figure 3-20 Fast Read Quad Output Timing Diagram	32
Figure 3-21 Fast Read Quad I/O Timing Diagram	33
Figure 3-22 Read Manufacturer/Device ID Quad I/O Timing Diagram	34
Figure 3-23 Timing Diagram of SPI Flash Read Interface	35
Figure 3-24 Timing Diagram of SPI Flash Write Interface	35
Figure 3-25 AHB Bus Transfer Interface Timing without Waiting State	35
Figure 3-26 AHB Bus Transfer Interface Timing with Waiting State	36
Figure 3-27 APB Bus Write Interface Timing	36
Figure 3-28 APB Bus Read Interface Timing	37
Figure 4-1 Open IP Core Generator	38
Figure 4-2 Open SPI Flash Interface (With External Flash) IP core	39
Figure 4-3 SPI Flash Interface (With External Flash) IP Core Port Diagram.....	39
Figure 4-4 Basic Information Configuration Interface	40

Figure 4-5 Options	40
Figure 5-1 Reference Design Block Diagram	41

List of Tables

Table 1-1 Terminology and Abbreviations	1
Table 2-1 Gowin SPI Flash Interface (With External Flash) IP	3
Table 2-2 Resource Utilization	4
Table 3-1 Gowin SPI Flash Interface (With External Flash) IP IO Port List	7
Table 3-2 Gowin SPI Flash Interface (With External Flash) IP Parameters	9
Table 3-3 Register List	9
Table 3-4 SPI Transfer Format Register	10
Table 3-5 SPI Transfer Control Register	11
Table 3-6 Dummy Cycle Settings in Common Transfer Formats	12
Table 3-7 SPI Command Register	12
Table 3-8 SPI Address Register	13
Table 3-9 SPI Data Register	13
Table 3-10 SPI Control Register	13
Table 3-11 SPI Status Register	14
Table 3-12 SPI Interrupt Enable Register	14
Table 3-13 SPI Interrupt Status Register	14
Table 3-14 SPI Interface Timing Register	15
Table 3-15 Configuration Register	15
Table 3-16 List of SPI Flash Some Commands	16
Table 6-1 Documents List	42
Table 6-2 SPI Flash Interface Design Source Code List	42
Table 6-3 RefDesign Folder Contents List	42

1 About This Guide

1.1 Purpose

The purpose of Gowin SPI Flash Interface (With External Flash) IP User Guide is to help you to quickly learn the features and usage of Gowin SPI Flash Interface (With External Flash) IP by providing the descriptions of functions, ports, timing, configuration and call, reference design, etc.

1.2 Related Documents

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

- [DS100, GW1N series of FPGA Products Data Sheet](#)
- [DS117, GW1NR series of FPGA Products Data Sheet](#)
- [DS891, GW1NRF series of FPGA Products Data Sheet](#)
- [DS821, GW1NS series of FPGA Products Data Sheet](#)
- [DS861, GW1NSR series of FPGA Products Data Sheet](#)
- [DS881, GW1NSER series FPGA Products Data Sheet](#)
- [DS102, GW2A series of FPGA Products Data Sheet](#)
- [DS226, GW2AR series of FPGA Products Data Sheet](#)
- [DS961, GW2ANR series of FPGA Products Data Sheet](#)
- [DS976, GW2AN-55 Data Sheet](#)
- [DS971, GW2AN-18X& 9X Data Sheet](#)
- [SUG100, Gowin Software User Guide](#)

1.3 Terminology and Abbreviations

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

Table 1-1 Terminology and Abbreviations

Terminology and Abbreviations	Meaning
AHB	Advanced High performance Bus
APB	Advanced Peripheral Bus
FIFO	First Input First Output
FPGA	Field Programmable Gate Array

Terminology and Abbreviations	Meaning
SPI	Serial Peripheral Interface

1.4 Support and Feedback

Gowin Semiconductor provides customers with comprehensive technical support. If you have any questions, comments, or suggestions, please feel free to contact us directly using the information provided below.

Website: www.gowinsemi.com

E-mail: support@gowinsemi.com

2 Overview

2.1 Overview

Flash has a non-volatile feature and SPI means serial peripheral interface. SPI Flash is a Flash memory device that is operated through a serial interface. Due to its simple interface and small package, SPI Flash is widely used in audio and video equipments, mobile phones, mobile peripherals, office equipments and industrial control equipments.

Gowin has designed a generic SPI Flash Interface (With External Flash) IP that provides a common command interface for users to interconnect with the SPI Flash chip to fulfill access requirements.

Table 2-1 Gowin SPI Flash Interface (With External Flash) IP

Gowin SPI Flash Interface (With External Flash) IP	
Logic Resource	Please refer to Table 2-2.
Delivered Doc.	
Design Files	Verilog (encrypted)
Reference Design	Verilog
TestBench	Verilog
Test and Design Flow	
Synthesis Software	GowinSynthesis
Application Software	Gowin Software (V 1.9.9 Beta-1 and above)

Note!

For the devices supported, you can click [here](#) to get the information.

2.2 Features

- Supports AHB and APB bus configuration interface
- Supports Normal, Dual, and Quad SPI interfaces
- Supports memory-mapped access (read-only)
- Adopts configurable SPI SCLK
- Supports configurable TX/RX FIFO depth

2.3 Resource Utilization

Gowin SPI Flash Interface (With External Flash) IP can be implemented by Verilog language. Its performance and resource utilization may vary when the design is employed in different devices, or at different densities, speeds, or grades. Using Gowin GW1NSR-4C series of FPGA as an instance, the resource utilization of Gowin SPI Flash Interface (With External Flash) IP is as shown in Table 2-2.

Table 2-2 Resource Utilization

Device	Speed Grade	Name	Resource Utilization	Note
GW1NSR-4C	-6	LUT	814	—
		REG	361	
		BSRAM	2	

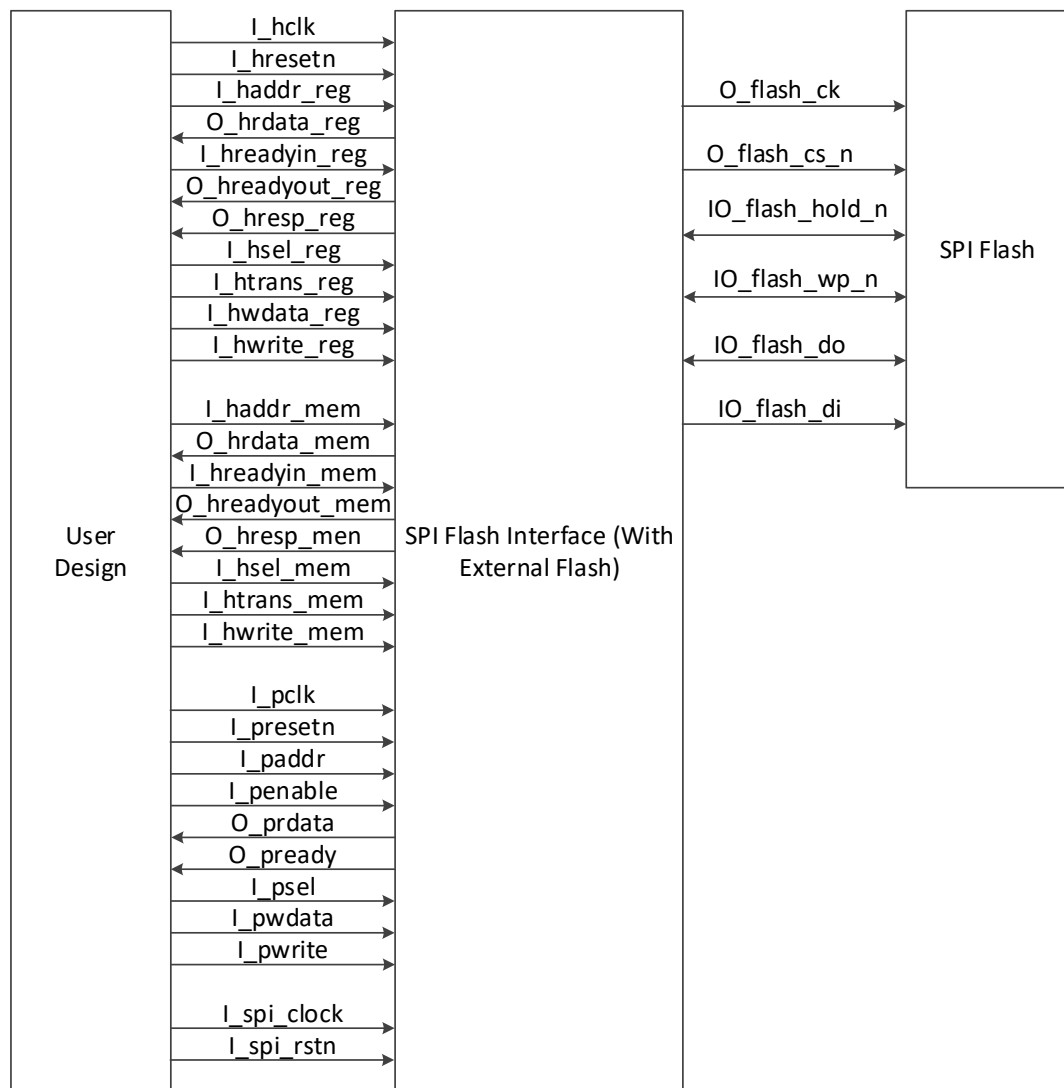
3 Functional Description

3.1 System Block Diagram

Gowin SPI Flash Interface (With External Flash) IP is a serial peripheral device interface controller IP that acts as a master device and connects to SPI Flash devices. SPI transfer format and timing can be configured through the AHB or APB bus interface. The data transferred by SPI can be operated in memory-mapped read-only mode or register read-write mode through the AHB bus interface, or in register read-write mode through the APB bus interface.

The format of the SPI transfer in register read/write mode includes command, address and data segments. A list of corresponding registers is provided in the IP; commands, addresses and data can be transferred through the AHB or APB bus.

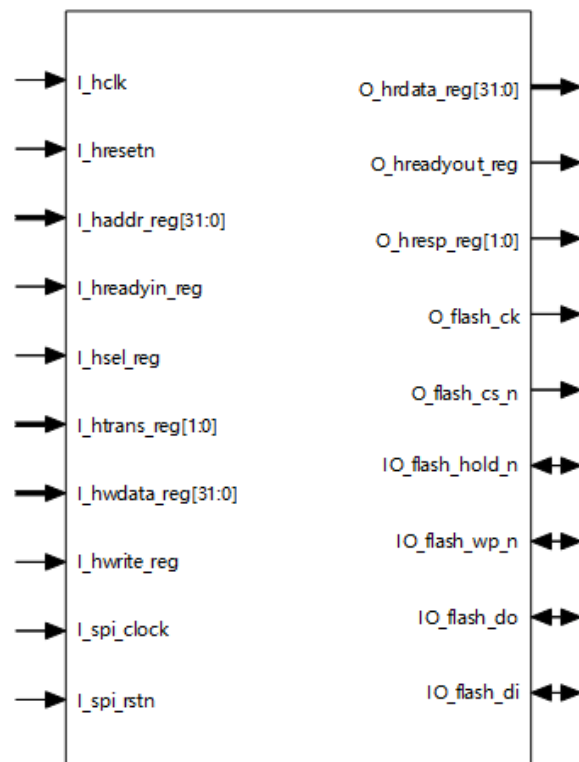
For the SPI transfer in memory-mapped read-only mode, data in the corresponding address space can be obtained simply by transferring the address through the AHB bus, and only the AHB bus is supported.

Figure 3-1 System Block Diagram

3.2 Port List

The I/O ports of Gowin SPI Flash Interface (With External Flash) IP are shown in Figure 3-2.

Figure 3-2 Gowin SPI Flash Interface (With External Flash) IP I/O Ports Diagram



Ports vary slightly depending on the parameters.

The details of I/O ports of Gowin SPI Interface IP are shown in Table 3-1.

Table 3-1 Gowin SPI Flash Interface (With External Flash) IP IO Port List

No.	Signal Name	I/O	Description	Note
1	I_hclk	I	AHB bus working clock	The I/O of all the signals take SPI Flash Interface (With External Flash) IP as a reference.
2	I_hresetn	I	AHB bus reset, active-low.	
3	I_haddr_reg	I	The address of AHB reg mode	
4	O_hrdata_reg	O	Read data in AHB reg mode	
5	I_hreadyin_reg	I	Master device ready signal in AHB reg mode	
6	O_hreadyout_reg	O	Slave device ready signal in AHB reg mode	
7	O_hresp_reg	O	Transfer response in AHB reg mode	
8	I_hsel_reg	I	Slave device selection signal in AHB reg mode	
9	I_htrans_reg	I	Transfer type in AHB reg mode	
10	I_hwdata_reg	I	Write data in AHB reg mode	

No.	Signal Name	I/O	Description	Note
11	I_hwrite_reg	I	Read/write signal in AHB reg mode. 1 for write; 0 for read.	
12	I_haddr_mem	I	The address of AHB mem mode Only valid in memory-mapped mode	
13	O_hrdata_mem	O	Read data in AHB mem mode Only valid in memory-mapped mode	
14	I_hreadyin_mem	I	Master device ready signal in AHB mem mode Only valid in memory-mapped mode	
15	O_hreadyout_mem	O	Slave device ready signal in AHB mem mode Only valid in memory-mapped mode	
16	O_hresp_mem	O	Transfer response in AHB mem mode Only valid in memory-mapped mode.	
17	I_hsel_mem	I	Slave device selection signal in AHB mem mode Only valid in memory-mapped mode	
18	I_htrans_mem	I	Transfer type in AHB mem mode Only valid in memory-mapped mode	
19	I_hwrite_mem	I	Read/write signal in AHB mem mode. 0 for Read and no write operation. Only valid in memory-mapped mode.	
20	I_pclk	I	APB bus working clock	
21	I_presetn	I	APB bus reset, active-low.	
22	I_paddr	I	The address of APB reg mode	
23	I_penable	I	Slave device enable signal in APB reg mode	
24	O_prdata	O	Read data in APB reg mode	
25	O_pready	O	Slave device ready signal in APB reg mode	
26	I_psel	I	Slave device selection signal in APB reg mode	
27	I_pwdata	I	Write data in APB reg mode	
28	I_pwrite	I	Read/write signal in APB reg mode. 1 for write; 0 for read.	
29	I_spi_clock	I	SPI Flash reference clock	

No.	Signal Name	I/O	Description	Note
30	I_spi_rstn	I	SPI Flash reset, active-low.	
31	O_flash_ck	O	SPI Flash clock signal	
32	O_flash_cs_n	O	SPI Flash chip selected signal	
33	IO_flash_hold_n	IO	SPI Flash hold signal	
34	IO_flash_wp_n	IO	SPI Flash write protect signal	
35	IO_flash_do	I/O	SPI Flash data signal MISO	
36	IO_flash_di	O	SPI Flash data signal MOSI	

3.3 Parameter Configuration

Table 3-2 Gowin SPI Flash Interface (With External Flash) IP Parameters

No.	Name	Range	Default	Description
1	Bus Type	AHB/APB	AHB	AHB means AHB bus is supported. APB means APB bus is supported.
2	Memory-mapped Read Support	Yes/No	No	Indicates whether memory-mapped read mode is supported, and data is read-only.
3	SPI Mode	Normal/Dual/Quad	Normal	● Normal: Normal SPI supported ● Dual: Dual SPI supported ● Dual: Quad SPI supported
4	SPI Clock Divider	0~128	0	Indicates the ratio of SCLK frequency to source clock frequency. When the SPI Clock Divider value is N with different values, the corresponding relationship is as follows. ● 0: $F_{sclk} = F_{source_clk}$ ● 1~128: $F_{sclk} = F_{source_clk} / (N * 2)$
5	TX FIFO Depth	2/4/8/16/32/64/128	4	RX FIFO depth
6	RX FIFO Depth	2/4/8/16/32/64/128	4	TX FIFO depth

3.4 Register Description

Table 3-3 Register List

Address Offset	Register	Description
0x00~0x0C	-	Reserved
0x10	TransFmt	SPI transfer format register

Address Offset	Register	Description
0x20	TransCtrl	SPI transfer control register
0x24	Cmd	SPI command register
0x28	Addr	SPI address register
0x2C	Data	SPI data register
0x30	Ctrl	SPI control register
0x34	Status	SPI status register
0x38	IntrEn	SPI interrupt enable register
0x3C	IntrSt	SPI interrupt status register
0x40	Timing	SPI interface timing register
0x44~0x78	-	Reserved
0x7C	Config	Configuration register

Note!

Register read/write definition:

- RO: Read-only
- RW: Read/Write
- W1C: Read, write 1 to clear.

3.4.1 SPI Transfer Format Register (0x10)

Table 3-4 SPI Transfer Format Register

Name	Bit(s)	Type	Description	Default
Reserved	31:18	-	-	0x0
AddrLen	5:16 PM	RW	Address length (Byte) 0X0: 1 byte 0X1: 2 bytes 0X2: 3 bytes 0X3: 4 bytes	0x2
Reserved	15:13	-	-	0x0
DataLen	12:8	RW	Data length (Bit) The actual data length is (DataLen+1) bits.	0x07
DataMerge	7	RW	Data merge mode enable, valid only DataLen = 0x07. • 0x1: 4 bytes transfer • 0x0: Only the lowest byte transfer	0x1
Reserved	6:4	-	-	0x0
LSB	3	RW	Data transfer order • 0X0: Most significant bit priority • 0X1: Least significant bit priority	0x0
Reserved	2:0	-	-	0x0

Note!

Before the register operation starts, you need to set the value of the SPI transfer format register, e.g., for a 3-byte address with a data length of 8, set the SPI transfer format register value to 0x00020780.

3.4.2 SPI Transfer Control Register (0x20)

Table 3-5 SPI Transfer Control Register

Name	Bit(s)	Type	Description	Default
Reserved	31	-	-	0x0
CmdEn	30	RW	SPI command segment enable 0x0: Disable command segment 0x1: Enable command segment	0x0
AddrEn	29	RW	SPI address segment enable 0x0: Disable address segment 0x1: Enable address segment	0x0
AddrFmt	28	RW	SPI address segment format 0x0: Single mode 0x1: The format is the same as that of data segment	0x0
TransMode	27:24	RW	Transfer Mode 0x0: Write and read simultaneously 0x1: Write-only 0x2: Read-only 0x3: Write/Read 0x4: Read/Write 0x5: Write/ Dummy/Read 0x6: Read/ Dummy/Write 0x7: No data (CmdEn or AddrEn must be enabled) 0x8: Dummy/Write 0x9: Dummy/Read 0xA~0xf: Reserved	0x0
DualQuad	23:22	RW	SPI data format 0x0: Single mode 0x1: Dual I/O mode 0x2: Quad I/O mode 0x3: Reserved	0x0
TokenEn	21	RW	Token byte transfer enable The address segment is appended with 1 byte during SPI read operation, and the value is selected from TokenValue. 0x0: Disable 0x1: Enable	0x0
WrTranCnt	20:12	RW	Number of write data units - WrTranCnt indicates the number of data units transferred from the data register to the SPI bus. The actual quantity is WrTranCnt+1. - WrTranCnt only works when the TransMode is 0,1,3,4,5,6 or 8. - For TransMode 0, WrTranCnt must be equal to RdTranCnt.	0x0

Name	Bit(s)	Type	Description	Default
TokenValue	11	RW	Token byte value 0x0: The value is 0x00. 0x1: The value is 0x69.	0x0
DummyCnt	10:9	RW	Number of Dummy data - The actual quantity is DummyCnt+1. - The number of Dummy cycles at SPI interface is (DummyCnt+1)*((DataLen+1)/SPI IO width). - The data pin is high resistance during Dummy. - DummyCnt is valid only when TransMode is 5,6,8,9. - Dummy cycle settings in common transfer formats are shown in Table 3-6.	0x0
RdTranCnt	8:0	RW	Read the number of data units - RdTranCnt indicates the number of data units received from the SPI bus and stored in the data register. The actual quantity is RdTranCnt+1. - RdTranCnt only works when the TransMode is 0,2,3,4,5,6 or 9. - For TransMode 0, WrTranCnt must be equal to RdTranCnt.	0x0

Table 3-6 Dummy Cycle Settings in Common Transfer Formats

Transfer Mode	DummyCnt+1	DataLen+1	DualQuad	#Dummy Cycles on the SPI Interface
Dummy Cycle	1	8	Single	8
	1	8	Dual	4
	1	8	Quad	2
	2	8	Quad	4
	3	8	Quad	6
	1	32	Quad	8

3.4.3 SPI Command Register (0x24)

Table 3-7 SPI Command Register

Name	Bit(s)	Type	Description	Default
Reserved	31:8	-	-	0x0
CMD	7:0	RW	SPI Command	0x0

3.4.4 SPI Address Register (0x28)

Table 3-8 SPI Address Register

Name	Bit(s)	Type	Description	Default
ADDR	31:0	RW	SPI address	0x0

3.4.5 SPI Data Register (0x2C)

Table 3-9 SPI Data Register

Name	Bit(s)	Type	Description	Default
DATA	31:0	RW	SPI transmits or receives data. For write operations, data is written to the TX FIFO. Low bytes first transfer. If the TX FIFO is full and the SPIActive bit of the status register is 1, the hready signal pulls down to insert the wait state. For read operations, read data from RX FIFO. Low bytes first transfer. If the RX FIFO is empty and the SPIActive bit of the status register is 1, the hready signal pulls down to insert the wait state. When the TX FIFO is empty, the SPI transfer will hold until more data is written to the TX FIFO; when the RX FIFO is full, the SPI transfer will hold until the RX FIFO has space to receive data. During write operations, if more data than WrTranCnt is written to the TX FIFO, the excess data goes to the next transfer or is cleared by resetting the TX FIFO.	0x0

3.4.6 SPI Control Register (0x30)

Table 3-10 SPI Control Register

Name	Bit(s)	Type	Description	Default
Reserved	31:3	-	-	0x0
TXFIFORST	2	RW	TX FIFO reset Write 1 to reset; clear automatically after the reset operation is completed.	0x0
RXFIFORST	1	RW	RX FIFO reset Write 1 to reset; clear automatically after the reset operation is completed.	0x0
SPIRST	0	RW	SPI reset Write 1 to reset; clear automatically after the reset operation is completed.	0x0

3.4.7 SPI Status Register (0x34)

Table 3-11 SPI Status Register

Name	Bit(s)	Type	Description	Default
Reserved	31:30	-	-	0x0
TXNUM[7:6]	29:28	RO	Number of valid data in the TX FIFO [7:6]	0x0
Reserved	27:26	-	-	0x0
RXNUM[7:6]	25:24	RO	Number of valid data in the RX FIFO [7:6]	0x0
TXFULL	23	RO	The flag of TX FIFO full	0x0
TXEMPTY	22	RO	The flag of TX FIFO empty	0x1
TXNUM[5:0]	21:16	RO	Number of valid data in the TX FIFO [5:0]	0x0
RXFULL	15	RO	The flag of RX FIFO full	0x0
RXEMPTY	14	RO	The flag of RX FIFO empty	0x1
RXNUM[5:0]	13:8	RO	Number of valid data in the RX FIFO [5:0]	0x0
Reserved	7:1	-	-	0x0
SPIActive	0	RO	The flag of SPI register active The SPIActive changes to 1 when the SPI command register is written, and to 0 when the transfer is completed.	0x0

3.4.8 SPI Interrupt Enable Register(0x38)

Table 3-12 SPI Interrupt Enable Register

Name	Bit(s)	Type	Description	Default
Reserved	31:5	-	-	0x0
EndIntEn	4	RW	SPI transfer end interrupt enable Controls whether an interrupt is triggered when SPI transfer completes.	0x0
Reserved	3:0	-	-	0x0

3.4.9 SPI Interrupt Status Register (0x3C)

Table 3-13 SPI Interrupt Status Register

Name	Bit(s)	Type	Description	Default
Reserved	31:5	-	-	0x0
EndInt	4	W1C	SPI transfer end interrupt This bit is set to 1 when SPI transfer end interrupt is triggered.	0x0
Reserved	3:0	-	-	0x0

3.4.10 SPI Interface Timing Register(0x40)

Table 3-14 SPI Interface Timing Register

Name	Bit(s)	Type	Description	Default
Reserved	31:12	-	-	0x0
Reserved	11:8	-	-	0x2
SCLK_DIV	7:0	RW	SPI interface clock to source clock frequency ratio depends on configuration. - When the SCLK_DIV value is in the range 0~127, SCLK frequency = ((SCLK_DIV+1) x2) * source clock frequency). - When the SCLK_DIV value is 0xff, the SCLK frequency is equal to the source clock spi_clock frequency.	0xff

3.4.11 Configuration Register (0x7C)

Table 3-15 Configuration Register

Name	Bit(s)	Type	Description	Default
Reserved	31:8	-	-	0x0
TxFIFOSize	7:4	RO	TX FIFO depth depends on configuration. 0x0: 2 words 0x1: 4 words 0x2: 8 words 0x3: 16 words 0x4: 32 words 0x5: 64 words 0x6: 128 words	0x1
RxFIFOSize	3:0	RO	RX FIFO depth depends on configuration. 0x0: 2 words 0x1: 4 words 0x2: 8 words 0x3: 16 words 0x4: 32 words 0x5: 64 words 0x6: 128 words	0x1

3.5 Hardware Configuration Options

3.5.1 TX FIFO depth

The depth of TX FIFO can be determined by the IP core generator, corresponding to TxFIFOSize in the configuration register (0x7C). The depth can be set to 2, 4, 8, 16, 32, 64, and 128. The default depth value is 4.

3.5.2 RX FIFO depth

The depth of RX FIFO can be determined by the IP core generator, corresponding to RXFIFOSize in the configuration register (0x7C). The depth can be set to 2, 4, 8, 16, 32, 64, and 128. The default depth value is 4.

3.5.3 Memory-Mapped Read

The IP provides an interface for memory-mapped read. The IP core generator is used to determine if this interface is supported.

3.5.4 SPI Interface Clock Parameters

The IP core generator is used to determine the SPI interface clock frequency, corresponding to SCLK_DIV in the SPI interface timing register (0x40). For example:

When the SPI Clock Divider is 0, the SCLK_DIV value is 255.

When the SPI Clock Divider is 1~128, the SCLK_DIV value is 0~127.

3.6 Read and Write Operation Flow

In this manual, it uses SPI Flash W25Q64DW as an example to illustrate the data read and write operation steps of SPI Flash Interface (With External Flash) IP, and the list of SPI Flash some commands is as follows.

Table 3-16 List of SPI Flash Some Commands

Command	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Normal SPI						
Write Enable	06H	–	–	–	–	–
Write Disable	04H	–	–	–	–	–
Read Status Register-1	05H	(S7-S0)	–	–	–	–
Read Status Register-2	35H	(S15-S8)	–	–	–	–
Write Status Register	01H	(S7-S0)	(S15-S8)	–	–	–
Page Program	02H	A23-A16	A15-A8	A7-A0	D7-D0	D7-D0
Sector Erase(4KB)	20H	A23-A16	A15-A8	A7-A0	–	–
Block Erase(32KB)	52H	A23-A16	A15-A8	A7-A0	–	–
Block Erase(64KB)	D8H	A23-A16	A15-A8	A7-A0	–	–
Chip Erase	C7/60H	–	–	–	–	–
Read Data	03H	A23-A16	A15-A8	A7-A0	(D7-D0)	–
Read Manufacturer/ Device ID	90H	dummy	dummy	00H	(MF7-MF0)	(ID7-ID0)
Read JEDEC ID	9FH	(MF7-MF0)	(ID15-ID8)	(ID7-ID0)	–	–
Read Unique ID	4BH	dummy	dummy	dummy	dummy	(UID63-UID0)
Dual SPI						
Fast Read Dual Output	3BH	A23-A16	A15-A8	A7-A0	dummy	(D7-D0,..

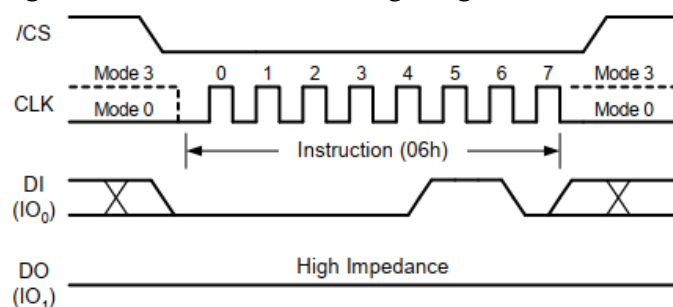
Command	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
						.)
Fast Read Dual I/O	BBH	A23-A8	A7-A0,M7-M0	(D7-D0,...)	–	–
Read Manufacturer/Device ID by Dual I/O	92H	A23-A8	A7-A0,M7-M0	(MF7-MF0,ID7-ID0)	–	–
Quad SPI						
Quad Page Program	32H	A23-A16	A15-A8	A7-A0	D7-D0	D7-D0
Fast Read Quad Output	6BH	A23-A16	A15-A8	A7-A0	dummy	(D7-D0,...)
Fast Read Quad I/O	EBH	A23-A0,M7-M0	(xxxx,D7-D0)	(D7-D0,...)	–	–
Read Manufacturer/Device ID by Quad I/O	94H	A23-A0,M7-M0	xxxx,(MF7-MF0,ID7-ID0)	(MF7-MF0,ID7-ID0)	–	–

3.6.1 Write Enable (06H) Operating Process

The process is as follows:

- Set SPI transfer control register (0x20) to 0x47000000
 - CmdEn = 1
 - TransMode = 0x7 (no data)
- Set SPI command register (0x24) to 0x00000006
CMD = 0x06

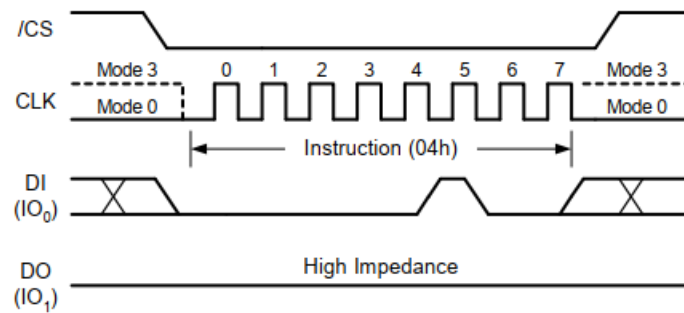
Figure 3-3 Write Enable Timing Diagram



3.6.2 Write Disable (04H) Operating Process

The process is as follows:

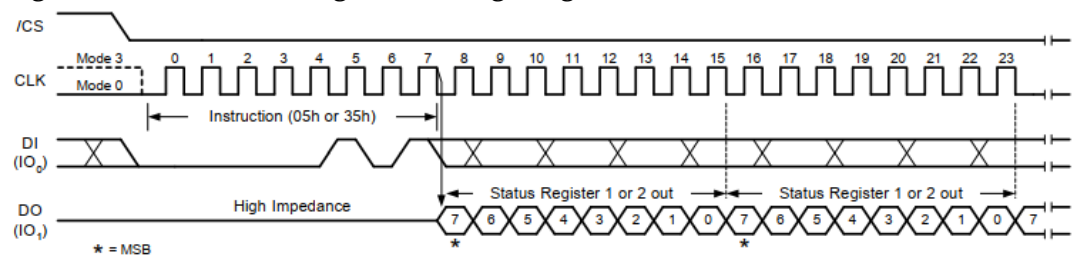
- Set SPI transfer control register (0x20) to 0x47000000
 - CmdEn = 1
 - TransMode = 0x7 (no data)
- Set SPI command register (0x24) to 0x00000004
CMD = 0x04

Figure 3-4 Write Disable Timing Diagram

3.6.3 Read Status Register (05H/35H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x42000000
 - a) $CmdEn = 1$
 - b) $TransMode = 0x2$ (Read-only)
2. Set SPI command register (0x24) to 0x00000005
 $CMD = 0x05$
3. Read SPI data register (0x2C)

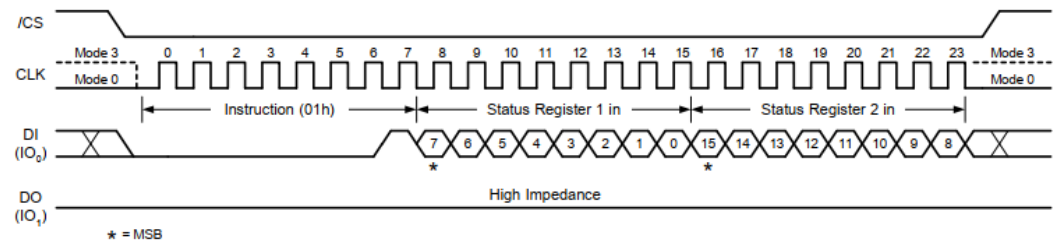
Figure 3-5 Read Status Register Timing Diagram

3.6.4 Write Status Register (01H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x41000000
 - a) $CmdEn = 1$
 - b) $TransMode = 0x1$ (Write-only)
2. Set SPI data register (0x2C) to 0x00000200
 $DATA = 0x00000200$ (Close Protect Bit, QE bit holds 1)
3. Set SPI command register (0x24) to 0x00000001
 $CMD = 0x01$

Figure 3-6 Write Status Register Timing Diagram



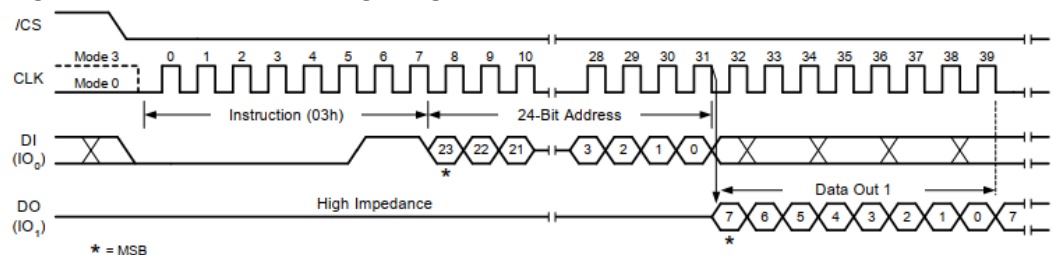
3.6.5 Read Data (03H) Operating Process

For example, the operating process of read 16 bytes in succession is as follows:

1. Set SPI transfer control register (0x20) to 0x6200000f.
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x2 (read-only)
 - d) RdTranCnt = 15 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x00000003
CMD = 0x03
5. Read SPI data register (0x2C)

Read up to 4 bytes at a time; more than 4 bytes, read this register in multiple times.

Figure 3-7 Read Data Timing Diagram



3.6.6 Page Program (02H) Operating Process

For example, the operating process of write 16 bytes in succession is as follows; there are two steps, the first is to erase data and the second is to write data.

1. Erase data

- a) Set SPI transfer control register (0x20) to 0x47000000
 - i. CmdEn = 1
 - ii. TransMode = 0x7 (no data)
- b) Set SPI command register (0x24) to 0x00000006
CMD = 0x06
- c) Set SPI transfer control register (0x20) to 0x67000000
 - i. CmdEn = 1
 - ii. AddrEn = 1
 - iii. TransMode = 0x7 (no data)
- d) Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
- e) Set SPI command register (0x24) to 0x00000020
CMD = 0x20 (sector erase)

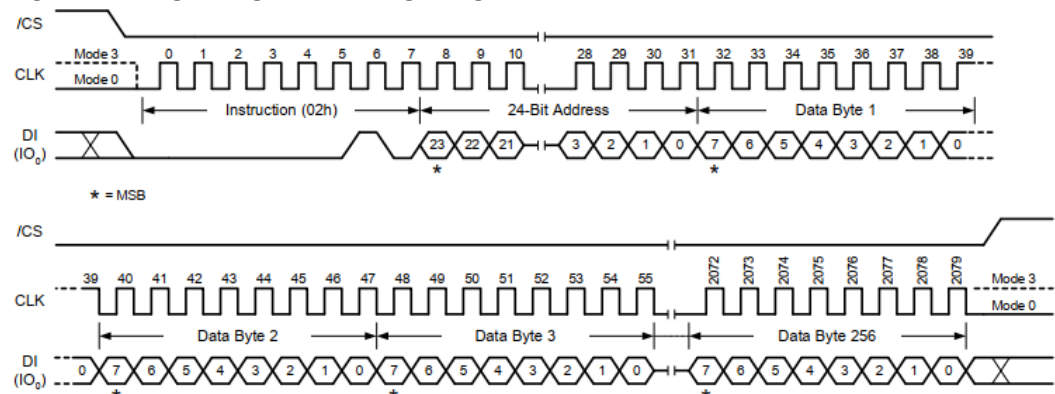
2. Write data

- a) Set SPI transfer control register (0x20) to 0x47000000
 - i. CmdEn = 1
 - ii. TransMode = 0x7 (no data)
- b) Set SPI command register (0x24) to 0x00000006
CMD = 0x06
- c) Set SPI transfer control register (0x20) to 0x6100f000
 - i. CmdEn = 1
 - ii. AddrEn = 1
 - iii. TransMode = 0x1 (write-only)
 - iv. WrTranCnt = 15 (Total number of transfer - 1)
- d) Set SPI control register (0x30) to 0x00000004
TXFIFORST = 1
- e) Set SPI interrupt enable register (0x38) to 0x00000010
EndIntEn = 1
- f) Set SPI data register (0x2C) to 0x33221100
DATA = 0x33221100 (1~4 bytes)

- g) Set SPI data register (0x2C) to 0x77665544
DATA = 0x77665544 (5~8 bytes)
- h) Set SPI data register (0x2C) to 0x33221100
DATA = 0xbbaa9988 (9~12 bytes)
- i) Set SPI data register (0x2C) to 0x77665544
DATA = 0xffeeddcc (13~16 bytes)
- j) Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
- k) Set SPI command register (0x24) to 0x00000002
CMD = 0x02
- l) Read SPI interrupt status register (0x3C)
If EndInt is 1, the transfer ends.
- m) Set SPI interrupt status register (0x3C) to 0x00000010
EndInt= 1 (Write 1 to clear)

Note!

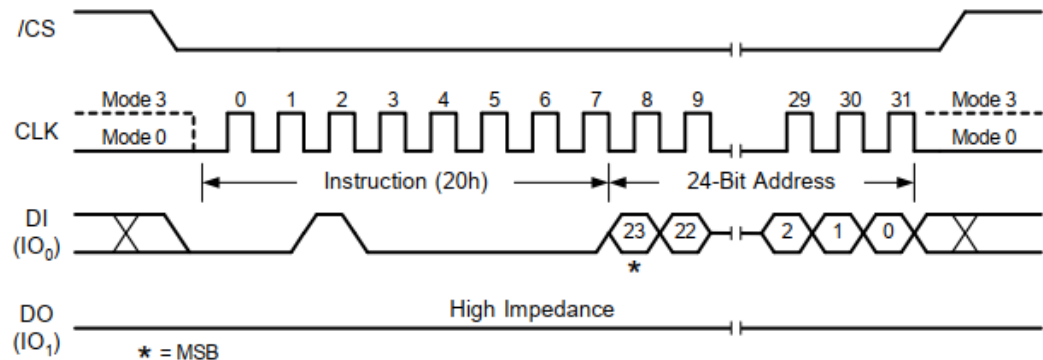
- The erase command is to set the corresponding area data of Flash to 1. There are 4 erase commands including sector erase, 32KB block erase, 64KB block erase, and chip erase. If there are other valid data in the erased area, the user needs to read the other data in the erased area before write, and then write back in time.
- The amount of write data cannot exceed the TX FIFO depth.

Figure 3-8 Page Program Timing Diagram**3.6.7 Sector Erase (20H) Operating Process**

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x47000000
 - a) CmdEn = 1
 - b) TransMode = 0x7 (no data)
2. Set SPI command register (0x24) to 0x00000006
CMD = 0x06
3. Set SPI transfer control register (0x20) to 0x67000000
 - a) CmdEn = 1

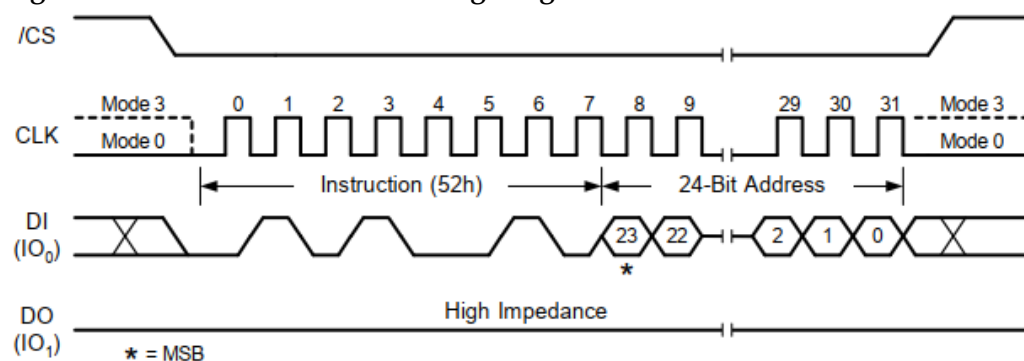
- b) AddrEn = 1
- c) TransMode = 0x7 (no data)
- 4. Set SPI address register (0x28) to 0x00000000.
ADDR = 0x00000000
- 5. Set SPI command register (0x24) to 0x00000020
CMD = 0x20 (sector erase)

Figure 3-9 Sector Erase Timing Diagram

3.6.8 32 KB Block Erase (52H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x47000000
 - a) CmdEn = 1
 - b) TransMode = 0x7 (no data)
2. Set SPI command register (0x24) to 0x00000006
CMD = 0x06
3. Set SPI transfer control register (0x20) to 0x67000000
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x7 (no data)
4. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
5. Set SPI command register (0x24) to 0x00000052
CMD = 0x52 (32 KB block erase)

Figure 3-10 32 KB Block Erase Timing Diagram

3.6.9 64 KB Block Erase (D8H) Operating Process

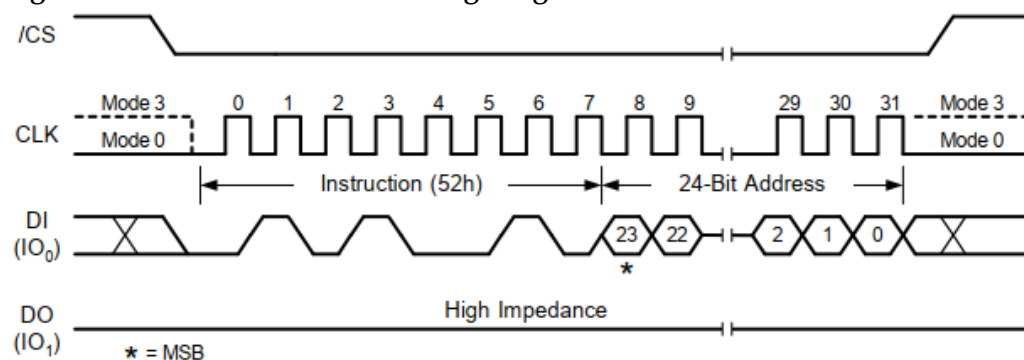
The process is as follows:

1. Set SPI transfer control register (0x20) to 0x47000000
 - a) $\text{CmdEn} = 1$
 - b) $\text{TransMode} = 0x7$ (no data)
2. Set SPI command register (0x24) to 0x00000006

$\text{CMD} = 0x06$
3. Set SPI transfer control register (0x20) to 0x67000000
 - a) $\text{CmdEn} = 1$
 - b) $\text{AddrEn} = 1$
 - c) $\text{TransMode} = 0x7$ (no data)
4. Set SPI address register (0x28) to 0x00000000

$\text{ADDR} = 0x00000000$
5. Set SPI command register (0x24) to 0x000000D8

$\text{CMD} = 0xD8$ (64KB block erase)

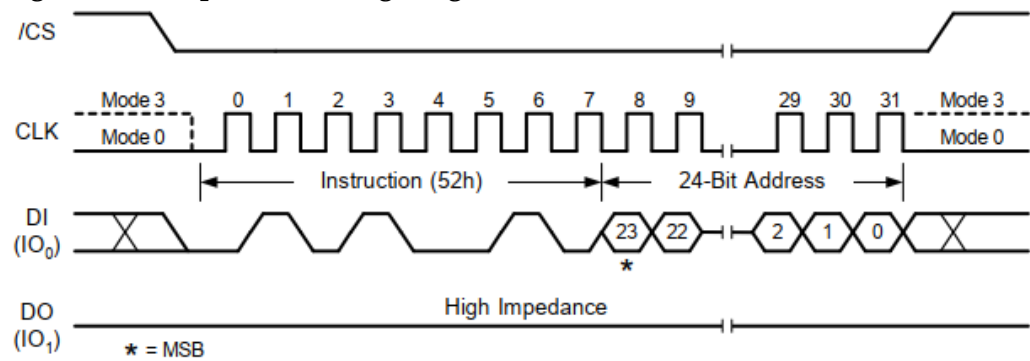
Figure 3-11 64KB Block Erase Timing Diagram

3.6.10 60 Erase (60/ C7H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x47000000
 - a) CmdEn = 1
 - b) TransMode = 0x7 (no data)
2. Set SPI command register (0x24) to 0x00000006
CMD = 0x06
3. Set SPI transfer control register (0x20) to 0x47000000
 - a) CmdEn = 1
 - b) TransMode = 0x7 (no data)
4. Set SPI command register (0x24) to 0x00000060
CMD = 0x60 (chip erase)

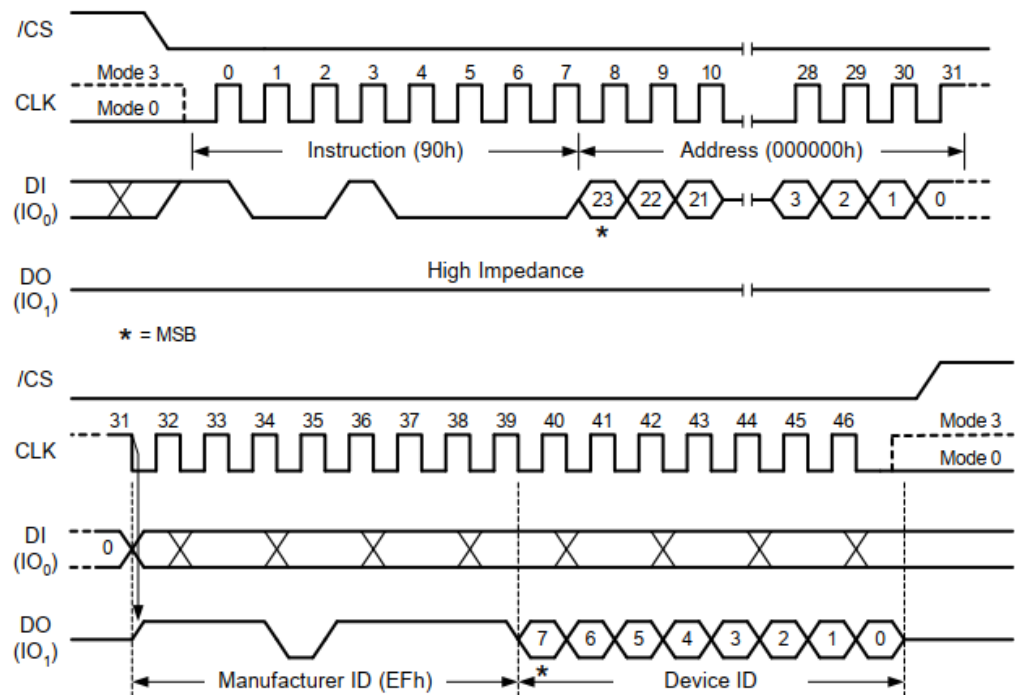
Figure 3-12 Chip Erase Timing Diagram



3.6.11 Read Manufacturer/Device ID (90H) Operating Process

The process is as follows:

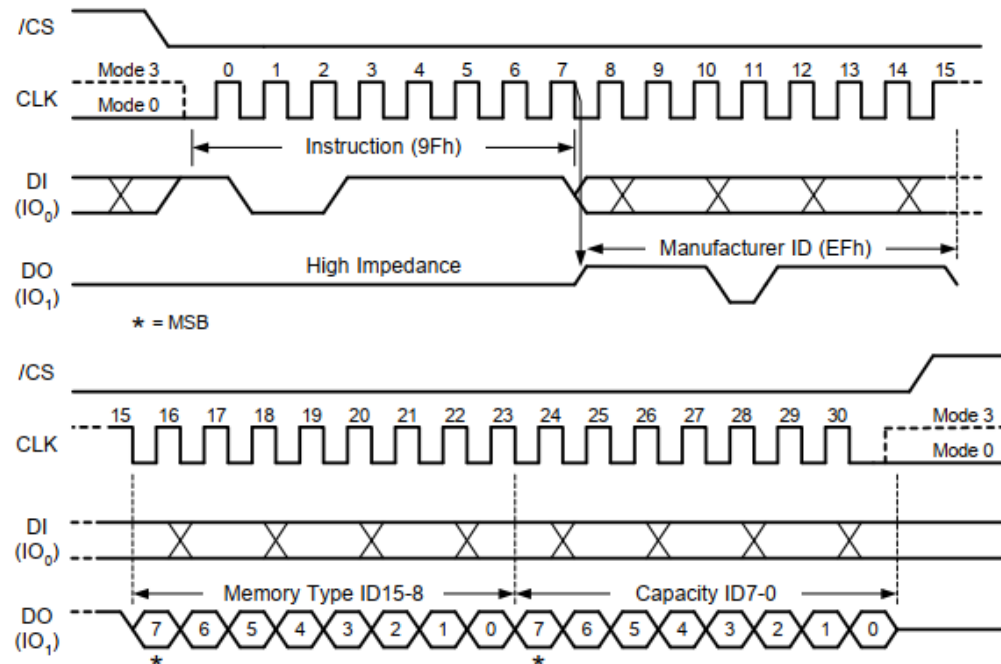
1. Set SPI transfer control register (0x20) to 0x62000001
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x2 (read-only)
 - d) RdTranCnt = 1 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x00000090
CMD = 0x90
5. Read SPI data register (0x2C)

Figure 3-13 Read Manufacture ID/Device ID Timing Diagram

3.6.12 Read JEDEC ID (9FH) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x42000002
 - a) CmdEn = 1
 - b) TransMode = 0x2 (Read-only)
 - c) RdTranCnt = 2 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI command register (0x24) to 0x0000009F
CMD = 0x9F
4. Read SPI data register (0x2C)

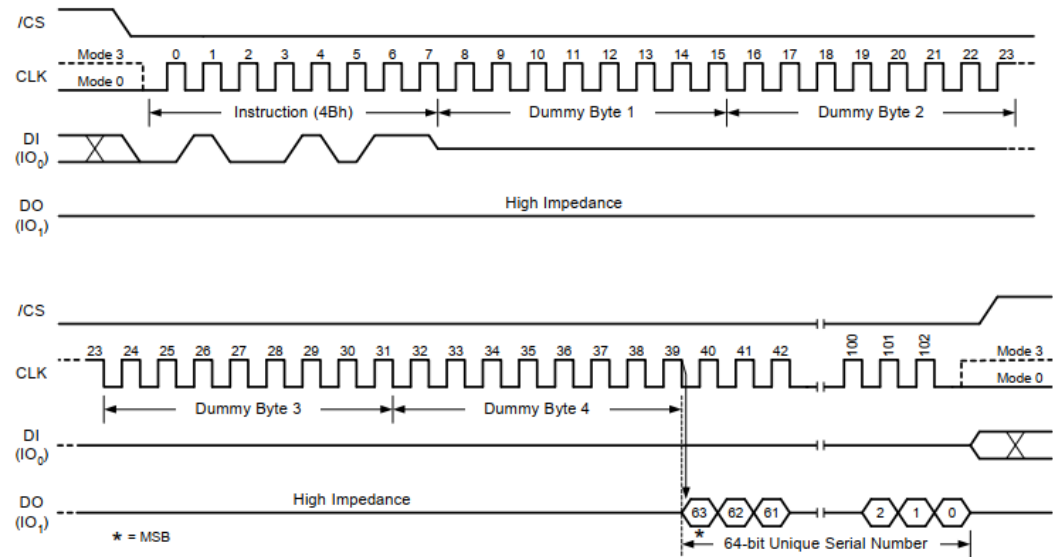
Figure 3-14 Read JEDEC ID Timing Diagram

3.6.13 Read Unique ID (4BH) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x69000007
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x9 (Dummy, read)
 - d) RdTranCnt = 7 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x0000004B
CMD = 0x4B
5. Read SPI data register (0x2C)

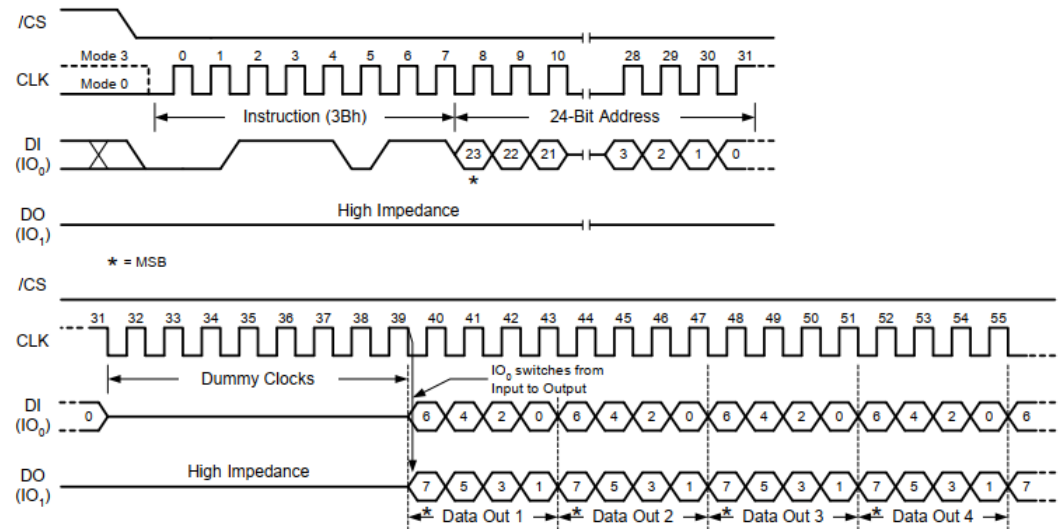
Since Unique ID is 64 bits, the SPI data register needs to be read twice.

Figure 3-15 Read Unique ID Timing Diagram

3.6.14 Fast Read Dual Output (3BH) Operating Process

For example, the operating process of read 4 bytes in succession is as follows:

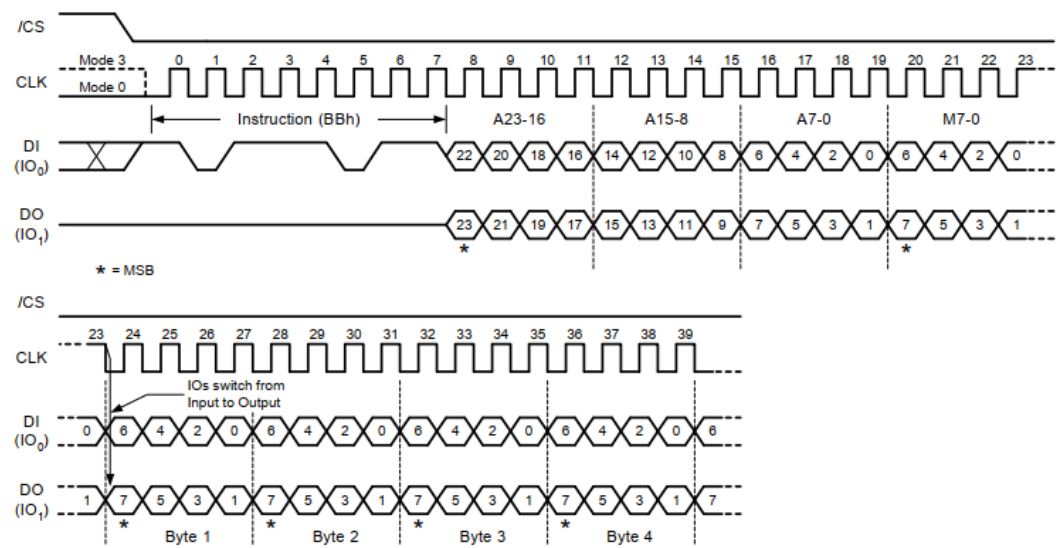
1. Set SPI transfer control register (0x20) to 0x69400003
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x9 (Dummy, read)
 - d) DualQuad = 0x1 (Dual I/O mode)
 - e) RdTranCnt = 3 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x0000003B
CMD = 0x3B
5. Read SPI data register (0x2C)

Figure 3-16 Fast Read Dual Output Timing Diagram

3.6.15 Fast Read Dual I/O (BBH) Operating Process

For example, the operating process of read 4 bytes in succession is as follows:

1. Set SPI transfer control register (0x20) to 0x79400003
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) AddrFmt = 1
 - d) TransMode = 0x9 (Dummy, read)
 - e) DualQuad = 0x1 (Dual I/O mode)
 - f) RdTranCnt = 3 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x000000BB
CMD = 0xBB
5. Read SPI data register (0x2C)

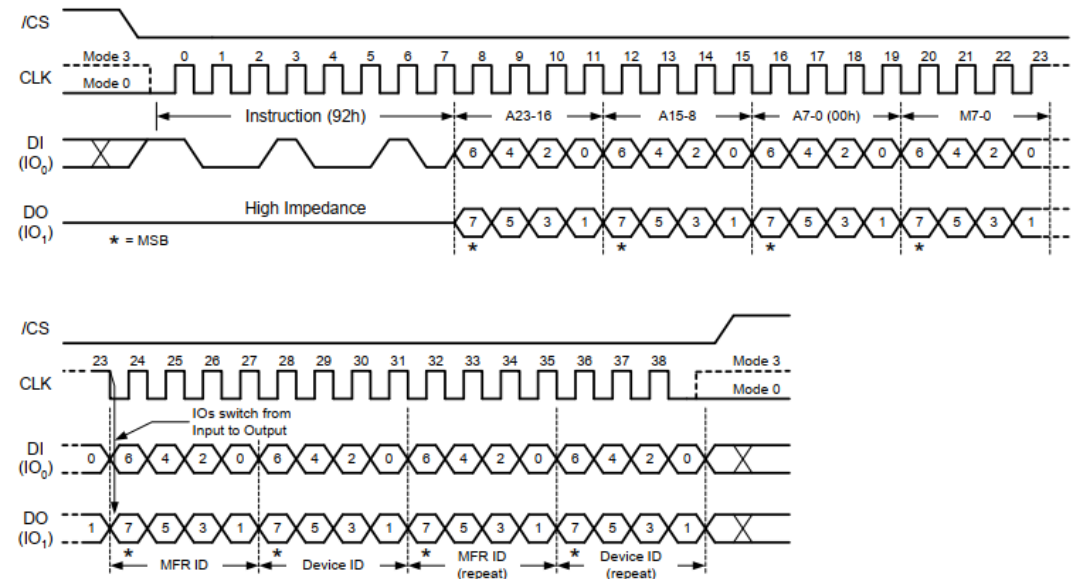
Figure 3-17 Fast Read Dual Timing Diagram

3.6.16 Read Manufacturer/Device ID Dual I/O (92H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x72600001
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) AddrFmt = 1
 - d) TransMode = 0x2 (read-only)
 - e) DualQuad = 0x1 (Dual I/O mode)
 - f) TokenEn = 1 (Token byte enable)
 - g) RdTranCnt = 1 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x00000092
CMD = 0x92
5. Read SPI data register (0x2C)

Figure 3-18 Read Manufacturer/Device ID Dual I/O Timing Diagram



3.6.17 Quad Page Program (32H) Operating Process

For example, the operating process of write 4 bytes in succession is as follows; there are two steps, the first is to erase data and the second is to write data.

1. Erase data

- a) Set SPI transfer control register (0x20) to 0x47000000
 - i. CmdEn = 1
 - ii. TransMode = 0x7 (no data)
- b) Set SPI command register (0x24) to 0x00000006
CMD = 0x06
- c) Set SPI transfer control register (0x20) to 0x67000000
 - i. CmdEn = 1
 - ii. AddrEn = 1
 - iii. TransMode = 0x7 (no data)
- d) Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
- e) Set SPI command register (0x24) to 0x00000020
CMD = 0x20 (sector erase)

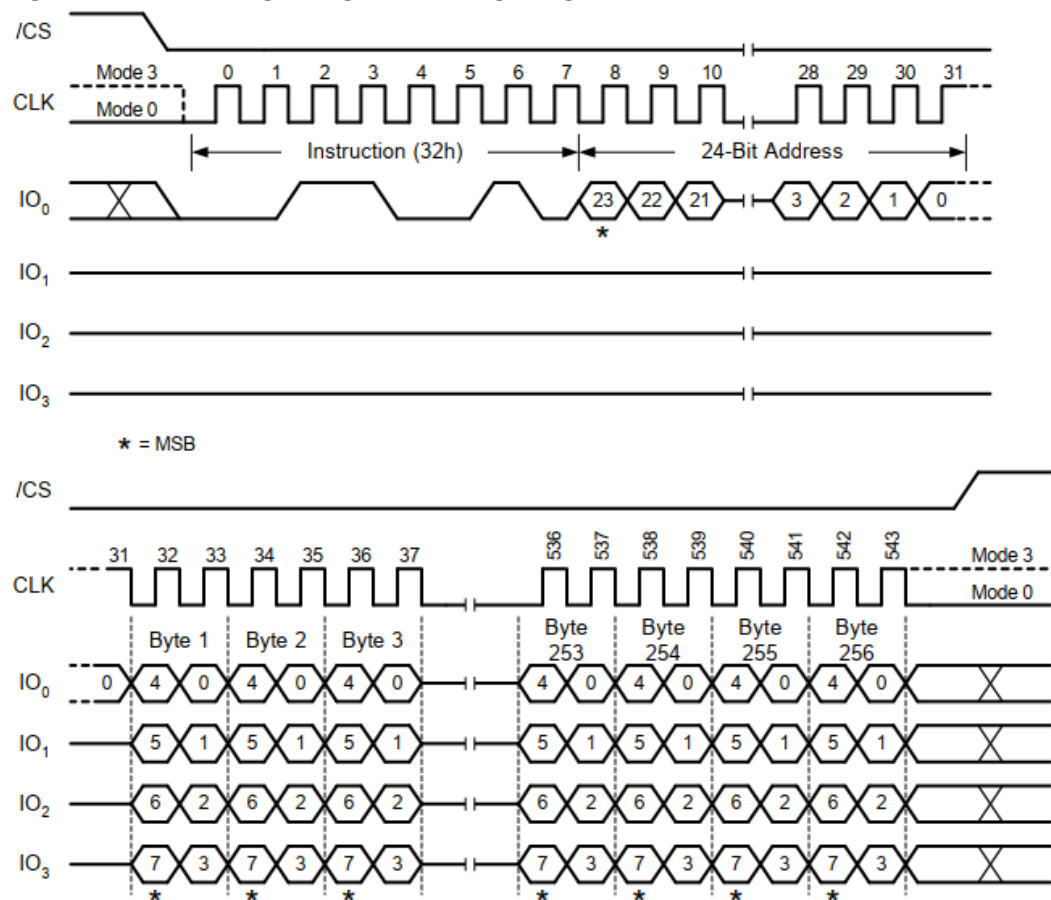
2. Write data

- a) Set SPI transfer control register (0x20) to 0x47000000
 - i. CmdEn = 1
 - ii. TransMode = 0x7 (no data)
- b) Set SPI command register (0x24) to 0x00000006
CMD = 0x06

- c) Set SPI transfer control register (0x20) to 0x61803000
 - i. CmdEn = 1
 - ii. AddrEn = 1
 - iii. TransMode = 0x1 (write-only)
 - iv. DualQuad = 0x2 (Quad I/O mode)
 - v. WrTranCnt = 3 (Total number of transfer - 1)
- d) Set SPI data register (0x2C) to 0x33221100
DATA = 0x33221100 (1~4 bytes)
- e) Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
- f) Set SPI command register (0x24) to 0x00000032
CMD = 0x32

Note!

- The erase command is to set the corresponding area data to 1. There are 4 erase commands including 4 KB sector erase, 32 KB block erase, 64 KB block erase, and chip erase. If there are other valid data in the erased area, the user needs to read the other data in the erased area before write, and then write back in time.
- The amount of write data cannot exceed the TX FIFO depth.

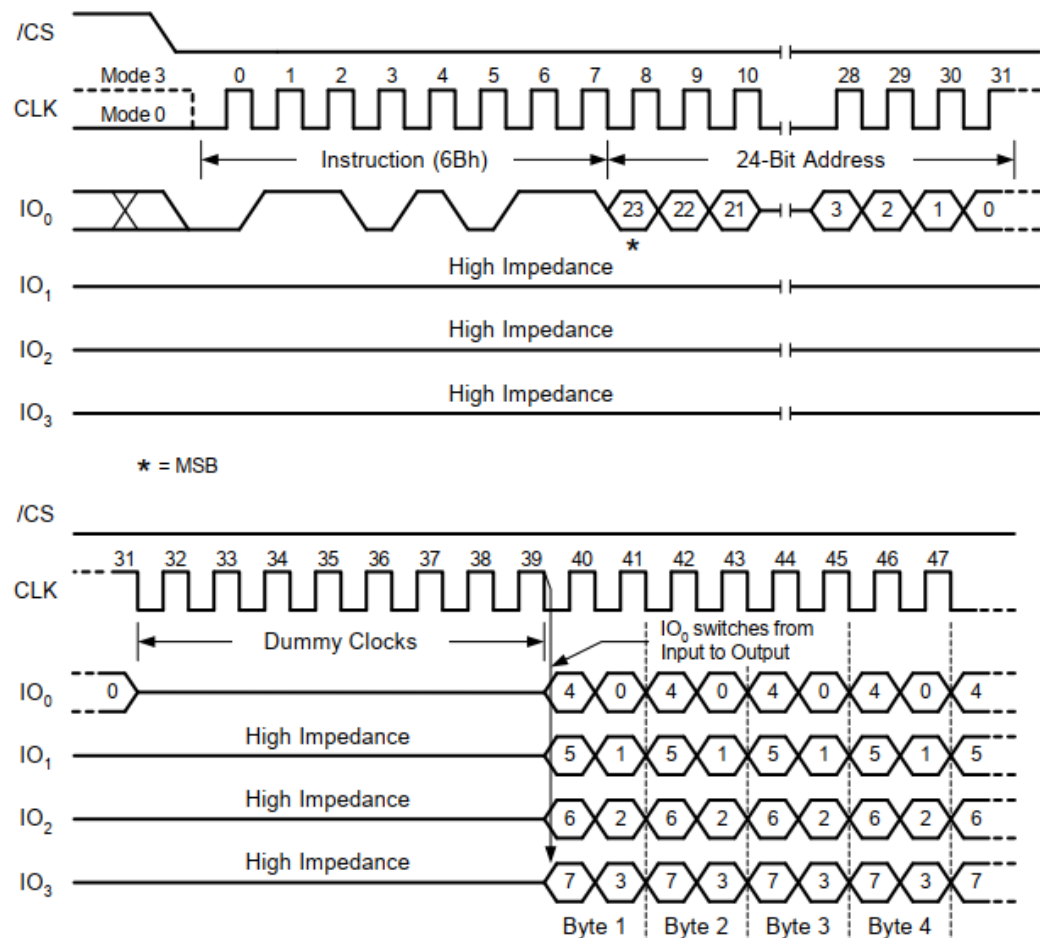
Figure 3-19 Quad Page Program Timing Diagram

3.6.18 Fast Read Quad Output (6BH) Operating Process

For example, the operating process of read 4 bytes in succession is as follows:

1. Set SPI transfer control register (0x20) to 0x69800603
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) TransMode = 0x9 (Dummy, read)
 - d) DualQuad = 0x2 (Quad I/O mode)
 - e) RdTranCnt = 3 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x0000006B
CMD = 0x6B
5. Read SPI data register (0x2C)

Figure 3-20 Fast Read Quad Output Timing Diagram

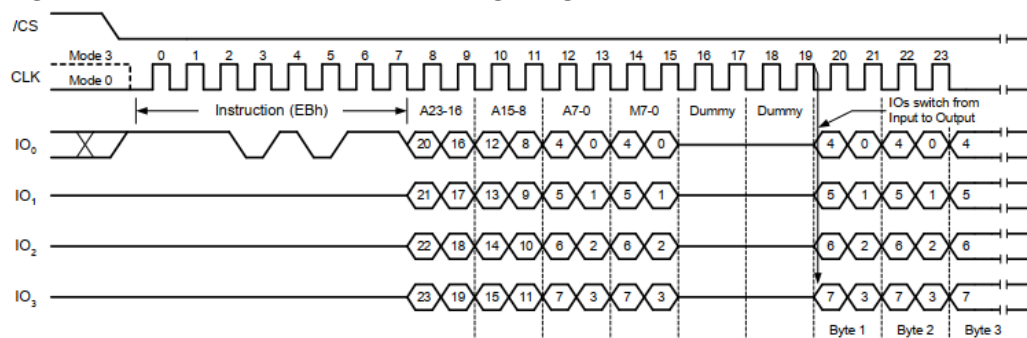


3.6.19 Fast Read Quad I/O (EBH) Operating Process

For example, the operating process of read 4 bytes in succession is as follows:

1. Set SPI transfer control register (0x20) to 79A00203
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) AddrFmt = 1
 - d) TransMode = 0x9 (Dummy, read)
 - e) DualQuad = 0x2 (Quad I/O mode)
 - f) TokenEn = 1 (Token byte enable)
 - g) DummyCnt = 1 (Number of Dummy -1)
 - h) RdTranCnt = 3 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
4. Set SPI command register (0x24) to 0x000000EB
CMD = 0xEB
5. Read SPI data register (0x2C)

Figure 3-21 Fast Read Quad I/O Timing Diagram



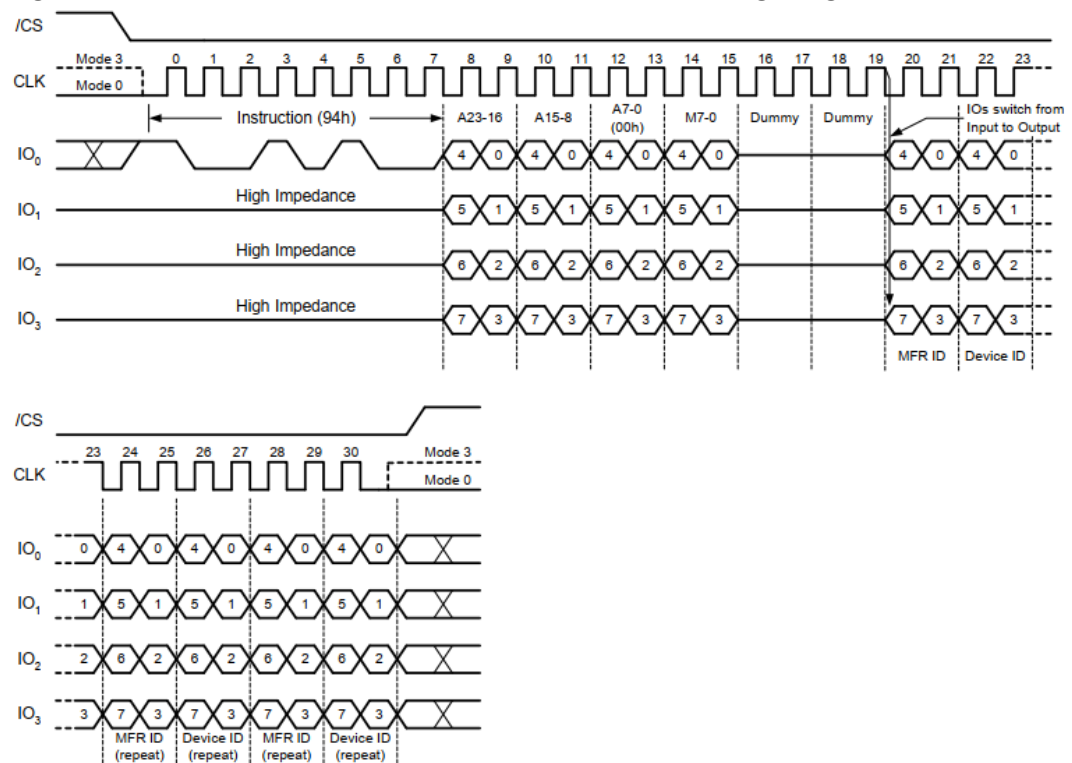
3.6.20 Read Manufacturer/Device ID Quad I/O (94H) Operating Process

The process is as follows:

1. Set SPI transfer control register (0x20) to 0x79A00201
 - a) CmdEn = 1
 - b) AddrEn = 1
 - c) AddrFmt = 1
 - d) TransMode = 0x9 (Dummy, read)
 - e) DualQuad = 0x2 (Quad I/O mode)

- f) TokenEn = 1 (Token byte enable)
 - g) DummyCnt = 1 (Number of Dummy -1)
 - h) RdTranCnt = 1 (Total number of transfer - 1)
2. Set SPI control register (0x30) to 0x00000002
RXFIFORST = 1
 3. Set SPI address register (0x28) to 0x00000000
ADDR = 0x00000000
 4. Set SPI command register (0x24) to 0x00000094
CMD = 0x94
 5. Read SPI data register (0x2C)

Figure 3-22 Read Manufacturer/Device ID Quad I/O Timing Diagram

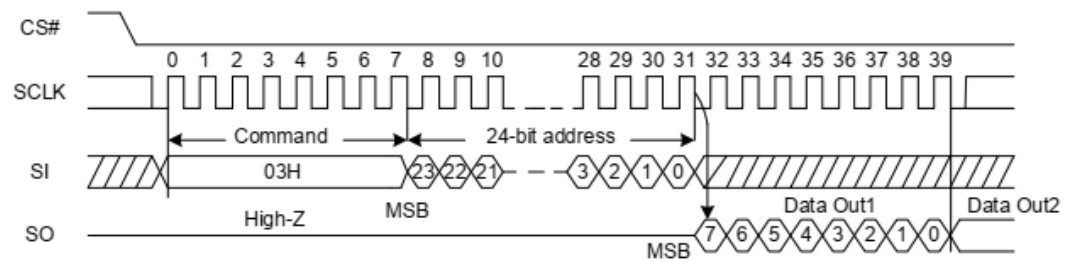


3.7 Timing Description

This section describes the timing of Gowin SPI Flash Interface (With External Flash) IP.

3.7.1 SPI Interface Timing

The timing diagram of SPI Flash read interface is as shown in Figure 3-23.

Figure 3-23 Timing Diagram of SPI Flash Read Interface

The timing diagram of SPI Flash write interface is as shown in Figure 3-24 .

Figure 3-24 Timing Diagram of SPI Flash Write Interface

3.7.2 AHB Interface Timing

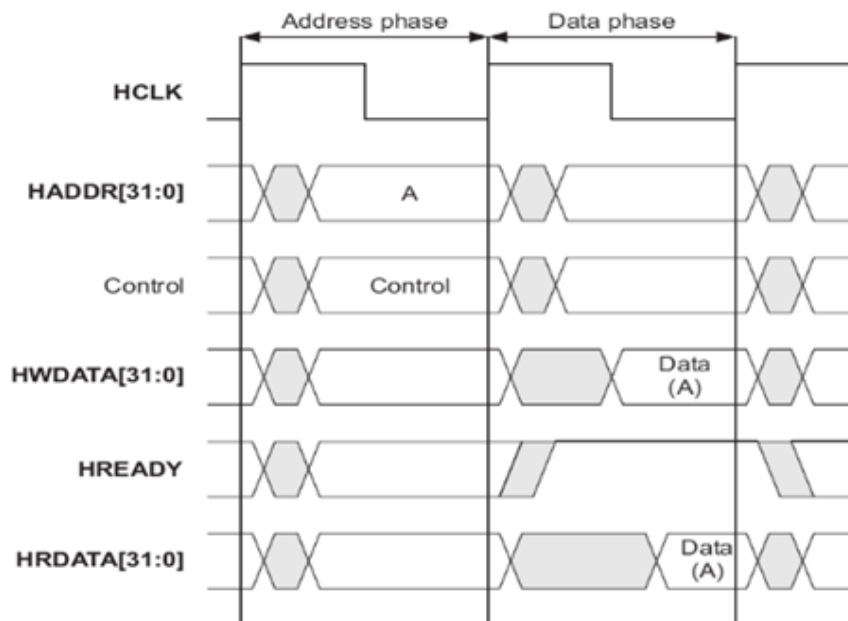
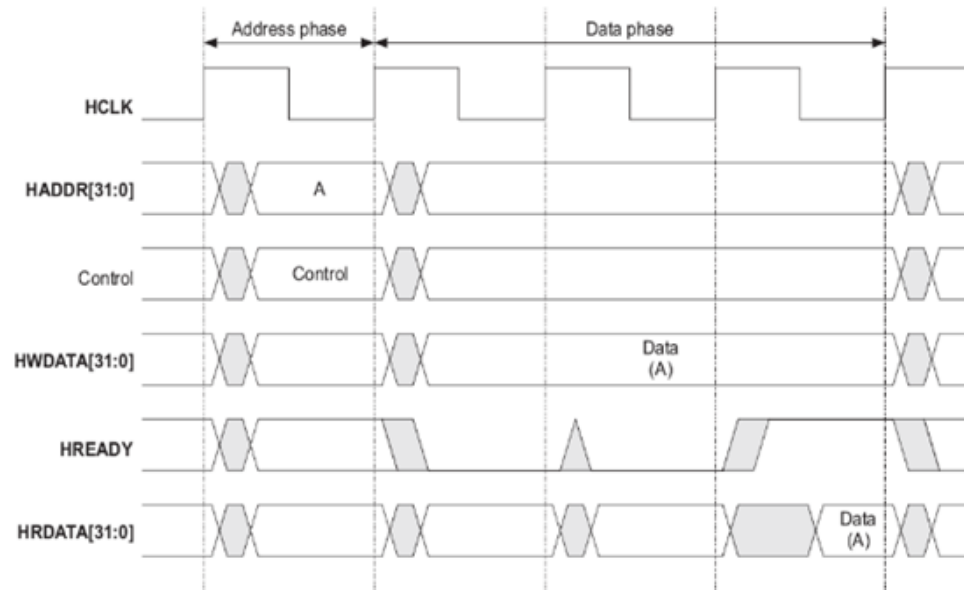
Figure 3-25 AHB Bus Transfer Interface Timing without Waiting State

Figure 3-26 AHB Bus Transfer Interface Timing with Waiting State



3.7.3 APB Interface Timing

Figure 3-27 APB Bus Write Interface Timing

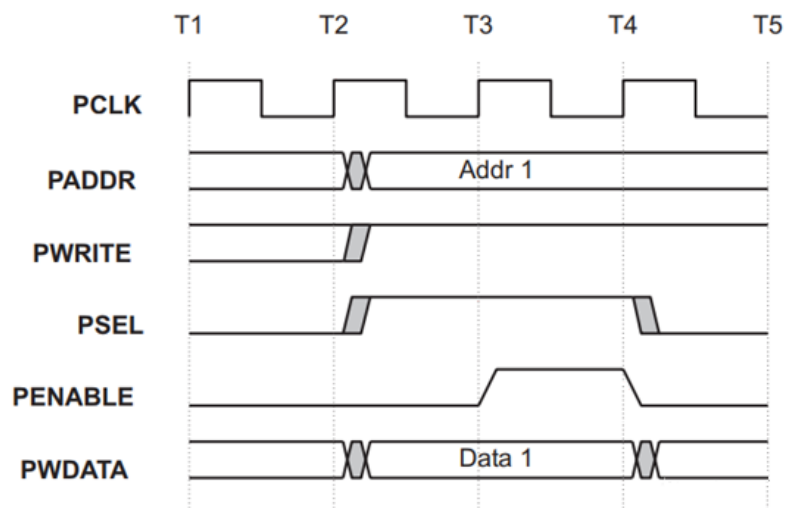
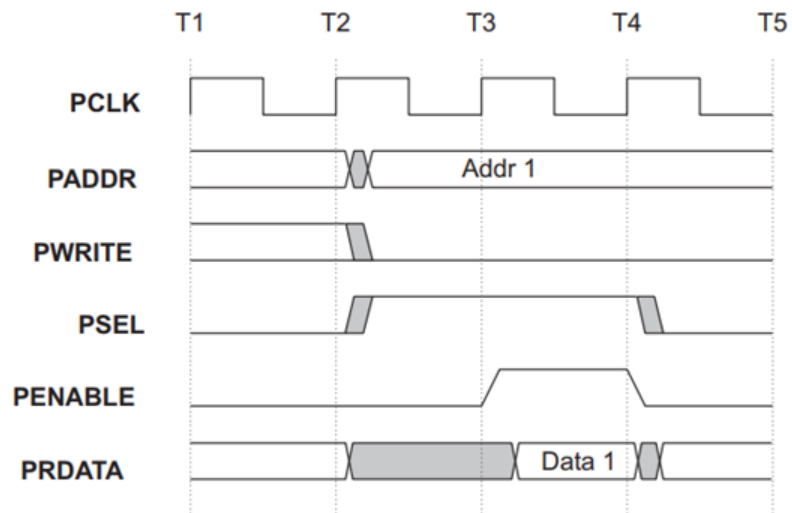


Figure 3-28 APB Bus Read Interface Timing



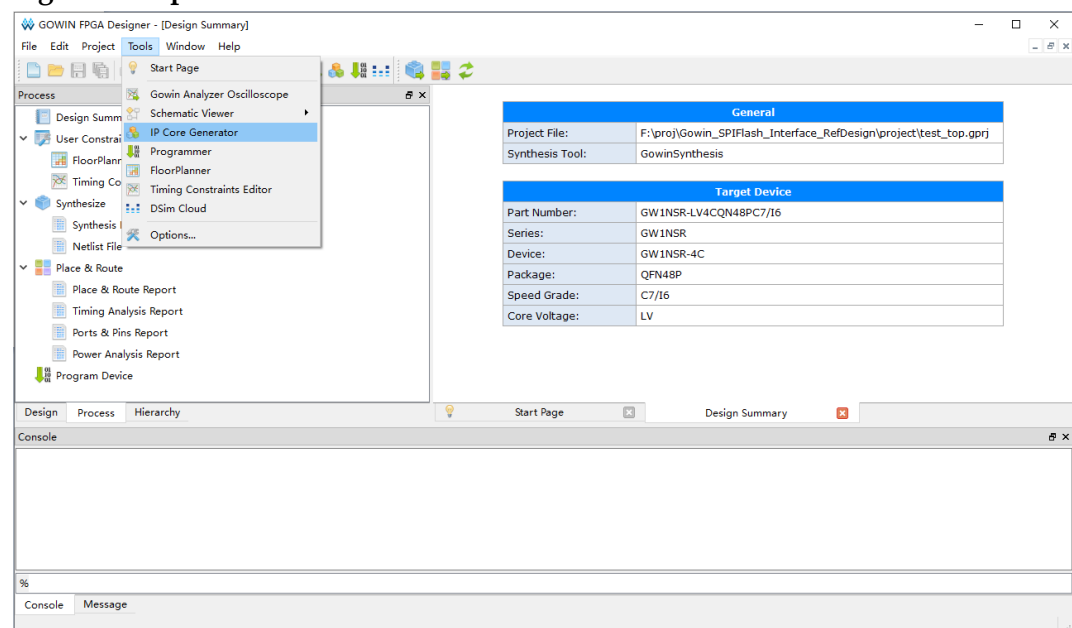
4 Interface Configuration

You can use IP core generator tool in Gowin Software to call and configure Gowin SPI Flash Interface (With External Flash) IP.

1. Open IP Core Generator

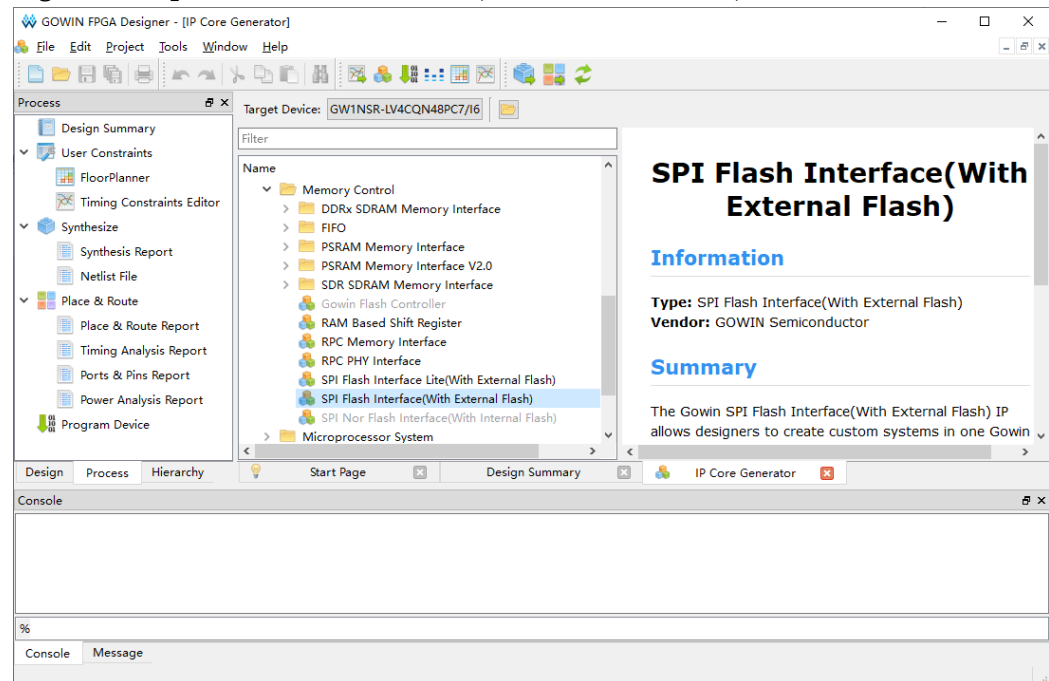
After creating the project, click the "Tools" tab in the upper left, select and open the IP Core Generator from the drop-down list, as shown in Figure 4-1.

Figure 4-1 Open IP Core Generator



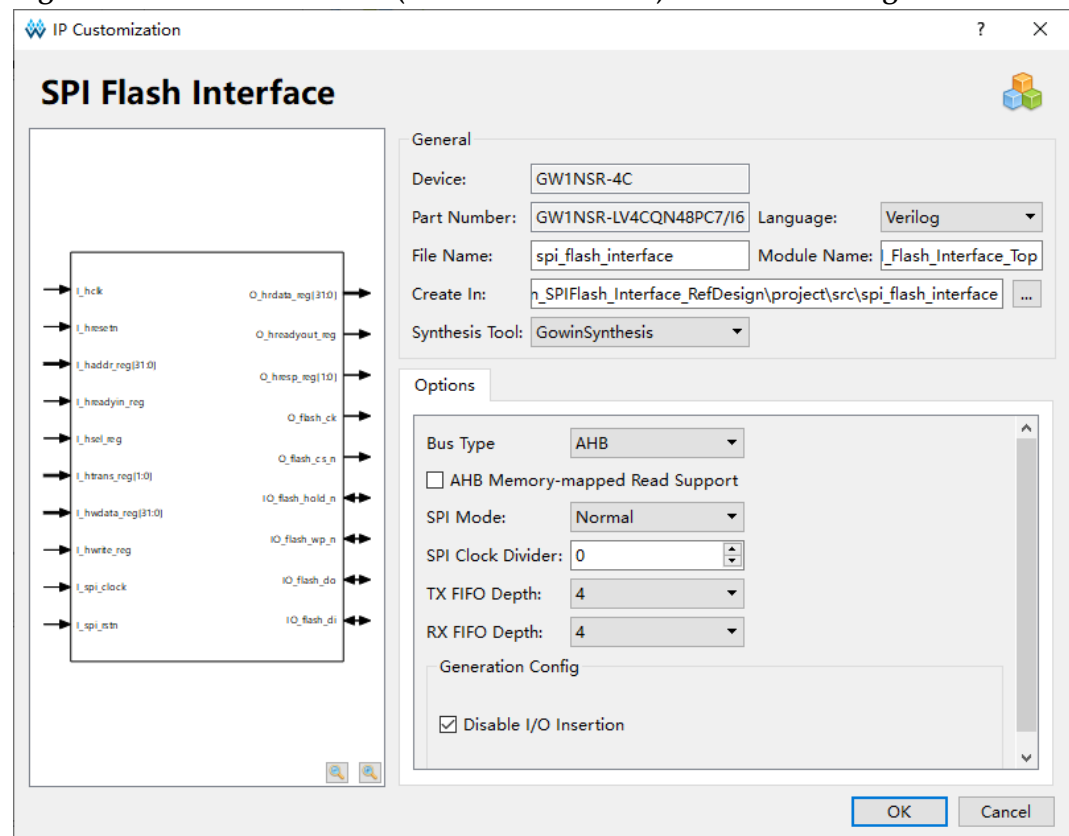
2. Open SPI Flash Interface (With External Flash) IP Core

Click "Memory Control" option, and double-click on "SPI Flash Interface (With External Flash)" to open the configuration interface, as shown in Figure 4-2.

Figure 4-2 Open SPI Flash Interface (With External Flash) IP core

3. SPI Flash Interface (With External Flash) IP Core Port Interface

On the left of the configuration interface is the port diagram of SPI Flash Interface (With External Flash) IP core, as shown in Figure 4-3.

Figure 4-3 SPI Flash Interface (With External Flash) IP Core Port Diagram

4. Configure Basic Information

See the basic information in the upper part of the configuration interface. Take GW1NSR-4C chip as an example, and select QN48P package. The "Module Name" displays the top-level file name of the generated project, and the default is "SPI_Flash_Interface_Top"; you can modify the name. The "File Name" displays the folder generated by the IP core, which contains the files required by SPI Flash Interface (With External Flash) IP core, and the default is "spi_flash_interface", and you can modify the path. "Create In" displays the path of IP core folder, and the default is "\\project path\\src\\spi_flash_interface"; you can modify the path.

Figure 4-4 Basic Information Configuration Interface

The 'General' configuration window shows the following settings:

- Device: GW1NSR-4C
- Part Number: GW1NSR-LV4CQN48PC7/I6
- Language: Verilog
- File Name: spi_flash_interface
- Module Name: _Flash_Interface_Top
- Create In: n_SPIFlash_Interface_RefDesign\project\src\spi_flash_interface
- Synthesis Tool: GowinSynthesis

5. Options

In the "Options", You can configure SPI Flash Interface (With External Flash) IP bus interface, mode, etc.

Figure 4-5 Options

The 'Options' configuration window shows the following settings:

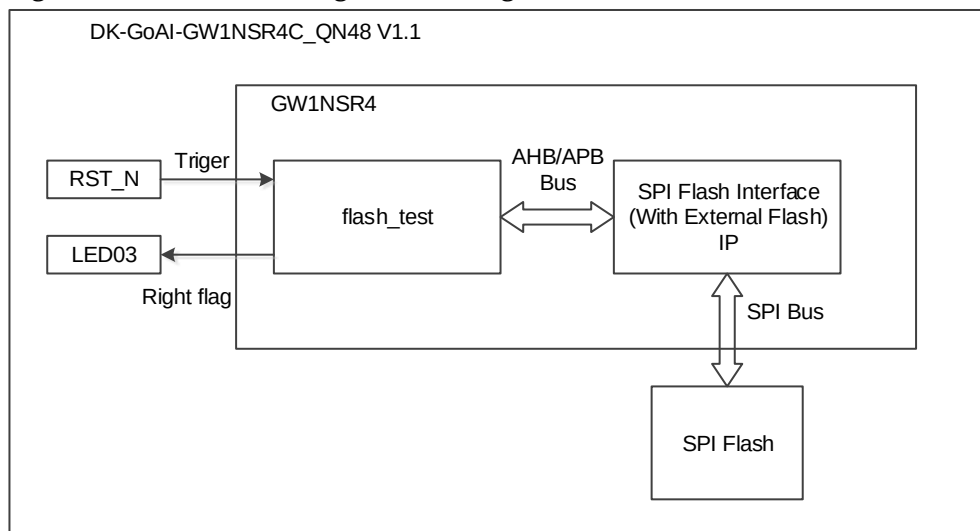
- Bus Type: AHB
- ☐ AHB Memory-mapped Read Support
- SPI Mode: Normal
- SPI Clock Divider: 0
- TX FIFO Depth: 4
- RX FIFO Depth: 4
- Generation Config:
 - ☒ Disable I/O Insertion

5 Reference Design

This chapter is intended to introduce the usage and structure of the reference design of SPI Flash Interface (With External Flash) IP. See SPI Flash Interface [Reference Design](#) for details at Gowinsemi website.

Take DK-GoAI-GW1NSR4C_QN48_V1.1 as an example, the block diagram of reference design is as shown in Figure 5-1. For the details of DK-GoAI-GW1NSR4C_QN48_V1.1, you can click [here](#).

Figure 5-1 Reference Design Block Diagram



In the reference design, the flash_test module simulates the AHB or APB bus master device; when pressing the key RST_N, it triggers flash_test to send write and read commands; perform continuous write and read operations to the SPI Flash through the SPI Flash Interface (With External Flash) IP, and then compare the read data with the write data to see whether it is correct, if it is correct, flash_test sends the Right flag signal to LED03 for indication; if the light is on, it indicates that the read/write is normal.

6 File Delivery

The file delivery for Gowin SPI Interface IP includes the documentation, the design source code, and the reference design.

6.1 Documentation

The folder contains the user guide in PDF.

Table 6-1 Documents List

Name	Description
IPUG1015, Gowin SPI Flash Interface (With External Flash) IP User Guide	Gowin SPI Flash Interface (With External Flash) IP User Guide, i.e., this manual.

6.2 Design Source Code (Encryption)

The encryption code folder contains the RTL encryption code of Gowin SPI Flash Interface (With External Flash) IP used for the GUI, to generate the IP cores as needed.

Table 6-2 SPI Flash Interface Design Source Code List

Name	Description
spi_flash_interface.v	The top-level file of the IP core, which provides users with interface information, encrypted.

6.3 Reference Design

The RefDesign folder contains the netlist file of Gowin SPI Flash Interface (With External Flash) IP, user reference design, constraints file, top-level file and the project folder, etc.

Table 6-3 RefDesign Folder Contents List

Name	Description
flash_test_top.v	The top module of reference design
flash_test.v	Test stimulus generation module
test_top.cst	Project physical constraints file
test_top.sdc	Project timing constraints file
spi_flash_interface	SPI Flash Interface (With External Flash) IP folder

