



Gowin SDIO_SPI IP User Guide

IPUG947-1.0E, 09/21/2020

Copyright©2020 Guangdong Gowin Semiconductor Corporation. All Rights Reserved.

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

Disclaimer

GOWINSEMI[®], LittleBee[®], Arora, and the GOWINSEMI logos are trademarks of GOWINSEMI and are registered in China, the U.S. Patent and Trademark Office, and other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders, as described at www.gowinsemi.com. GOWINSEMI assumes no liability and provides no warranty (either expressed or implied) and is not responsible for any damage incurred to your hardware, software, data, or property resulting from usage of the materials or intellectual property except as outlined in the GOWINSEMI Terms and Conditions of Sale. All information in this document should be treated as preliminary. GOWINSEMI may make changes to this document at any time without prior notice. Anyone relying on this documentation should contact GOWINSEMI for the current documentation and errata.

Revision History

Date	Version	Description
09/21/2020	1.0E	Initial version published.

Contents

Contents	0
List of Figures	2
List of Tables.....	3
1 About This Guide	4
1.1 Purpose	4
1.2 Related Documents	4
1.3 Terminology and Abbreviations.....	4
1.4 Support and Feedback.....	4
2 Overview	5
3 Features and Performance.....	6
3.1 Features	6
3.2 Performance.....	7
3.3 Resource Utilization	7
4 System Block Diagram.....	8
4.1 System Block Diagram.....	8
5 Port List	10
5.1 System Signal.....	10
5.2 SDIO Signal.....	11
5.3 SPI Signal.....	11
6 Structure and Functional Description.....	12
6.1 Architecture	12
6.2 Initialization Flow	13
6.3 SPI Port Write Operation Flow	13
6.3.1 Disable Interrupt	13
6.3.2 Enable Interrupt	13
6.4 SPI Port Read Operation Flow	14
6.4.1 Disable Interrupt	14
6.4.2 Enable Interrupt	14
6.5 SPI Receiving FIFO Read Operation.....	15
6.5.1 Disable Interrupt	15
6.5.2 Enable Interrupt	15
6.6 Register.....	16
6.6.1 Register Description.....	16

6.6.2 Register Description.....	16
6.7 Interface Timing	19
6.7.1 SPI Interface Timing.....	19
6.7.2 SDIO Bus Interface Timing.....	19
7 Parameters Configuration.....	20
7.1 Overview	20
7.2 Parameter.....	20
8 Interface Configuration	21
8.1 Overview	21
8.2 Application Diagram	21
8.3 GUI Call.....	22

List of Figures

Figure 4-1 System Block Diagram.....	8
Figure 6-1 Architecture	12
Figure 6-2 SPI Write Interface Timing	19
Figure 6-3 SPI Read Interface Timing	19
Figure 8-1 Application Diagram.....	21

List of Tables

Table 1-1 Abbreviations and Terminology	4
Table 3-1 Performance Reference	7
Table 5-1 System Signal	10
Table 5-2 SDIO Signal	11
Table 5-3 SPI Signal	11
Table 6-1 Register	16
Table 6-2 CIR	16
Table 6-3 IITR.....	16
Table 6-4 IER.....	17
Table 6-5 SBDIFR.....	17
Table 6-6 RTCR.....	17
Table 6-7 TFR.....	17
Table 6-8 RFR	17
Table 6-9 SPISR.....	18
Table 6-10 IIR.....	18

1 About This Guide

1.1 Purpose

SDIO-SPI user guide mainly includes the features of function, diagram, principle, signal definition, parameter, GUI, and interface timing. It aims to help you quickly understand the features and usage of Gowin SDIO-SPI to accelerate product development.

1.2 Related Documents

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

[*IPUG905*](#), *Gowin SDIO Slave Controller User Guide*.

1.3 Terminology and Abbreviations

The terminology and abbreviations used in this manual are as shown in Table 1-1.

Table 1-1 Abbreviations and Terminology

Terminology and Abbreviations	Meaning
FPGA	Field Programmable Gate Array
SDIO	Secure Digital Input and Output
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 by the following ways.

Website: www.gowinsemi.com/en/

E-mail: support@gowinsemi.com

2Overview

SDIO (Secure Digital Input and Output) defines a peripheral interface on the SD standard.

SPI is a high-speed, full-duplex, synchronous communication bus.

In order to reduce the difficulty of system development and improve the speed of product development, SDIO- SPI IP is designed to realize the function of interface conversion between SDIO and SPI.

After calling the SDIO-SPI, it can be applied to GW1N, GW1NR, GW1NS, GW1NZ, GW2A and GW2AR series of FPGA products by using Gowin softwares.

3 Features and Performance

3.1 Features

- Supports data communication of SDIO and SPI;
- SDIO interface is slave mode;
- SPI interface is master mode, supporting up to 4 SPI channels;
- SPI bus clock rate can be set flexibly through register configuration;
- Supports interrupt;
- The depth of the SPI receiving FIFO is 255 Bytes;
- The depth of the SPI sending FIFO is 255 Bytes;
- All operations on the user side of SDIO Host can be realized through CMD53 interface, including clearing interrupt registers on the Host side, i.e. reading and writing the Funtion1 register via CMD53 interface to realize the parameter configuration, state query and data read/write with SPI;
- SDIO CMD53 command OP Code fixed value to 0, select Multi byte R/W to fixed address to complete the data interaction with SPI interface;
- Can be synthesized;
- The language is Verilog.

3.2 Performance

The frequency of SDIO-SPI depends on the clock frequency provided by the SDIO host and the maximum frequency supported by the IP on the selected chip.

3.3 Resource Utilization

Table 3-1 Performance Reference

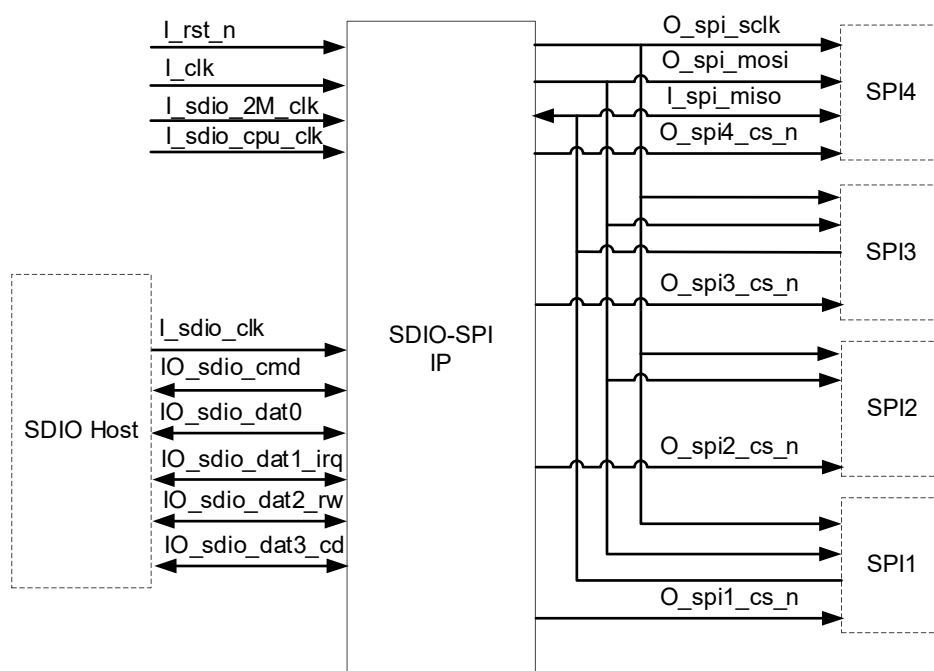
Chip Type	Programming Language	LUT4 Resources	REG Resources
GW2A-LV18PG256C7/I6	Verilog	4039	2155

4 System Block Diagram

4.1 System Block Diagram

SDIO-SPI is an interface between SDIO host and SPI terminal. It realizes data communication from SDIO to SPI. SDIO host includes clock, command, data, interrupt signal. SPI signal includes data, clock, and chip selection. SPI_UART includes reset and clock signals.

Figure 4-1 System Block Diagram



Note!

- Normally, the SPI interface side triggers the SDIO Host side to operate through interrupts; after the SDIO Host side disables all interrupts in the interrupt register, the SDIO Host can read the status and control the data sending and receiving operations through queries.

- If the interrupt function is enabled, when the SPI interface side triggers the interrupt and you receive the interrupt, you should first query the IIR status value and then execute the corresponding operation according to the instruction of the IIR (it is also OK if you skip the interrupt), and finally you can send a clear interrupt instruction to notify the SPI interface side to clear the interrupt.

5 Port List

5.1 System Signal

Table 5-1 System Signal

No.	Signal Name	I/O	Description	Notes
1	l_rst_n	I	Reset	The I/O of all the signals takes IP as reference
2	l_clk	I	System clock for internal logic and SPI interface processing	
3	l_sdio_2M_clk	I	SDIO slave required clock	
4	l_sdio_cpu_clk	I	SDIO slave required clock	

5.2 SDIO Signal

Table 5-2 SDIO Signal

No.	Signal Name	I/O	Description	Notes
1	I_sdio_clk	I	SDIO clock input signal	The I/O of all the signals takes IP as reference
2	IO_sdio_cmd	IO	SDIO Command/ Response Signal	
3	IO_sdio_dat0	IO	● SDIO 1bit mode is dat signal input; ● SDIO 4bit mode is dat0 signal input.	
4	IO_sdio_dat1_irq	IO	● SDIO 1bit mode is interrupt signal input; ● SDIO 4bit mode is dat1 and interrupt multiplexed signal input.	
5	IO_sdio_dat2_rw	IO	● SDIO 1bit mode is Read Wait signal input; ● SDIO 4bit Mode is dat2 and Read Wait multiplexed signal input.	
6	IO_sdio_dat3_cd	IO	● SDIO 1bit mode is undefined; ● SDIO 4bit mode is dat3 signal input.	

5.3 SPI Signal

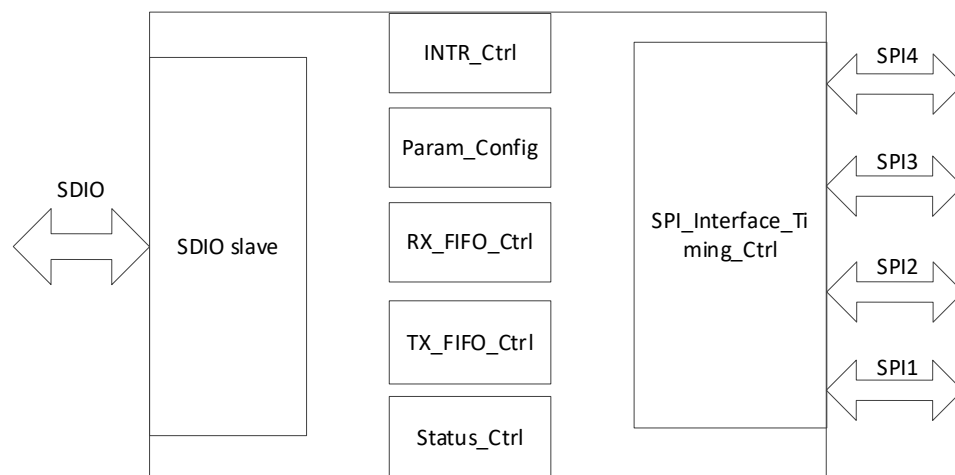
Table 5-3 SPI Signal

No.	Signal Name	I/O	Description	Notes
1	O_spi_sclk	O	SPI clock signal	The I/O of all the signals takes IP as reference
2	O_spi_mosi	O	SPI sending data	
3	I_spi_miso	I	SPI receiving data	
4	O_spi4_cs_n	O	SPI 4-channel chip-selected signal	
5	O_spi3_cs_n	O	SPI 3-channel chip-selected signal	
6	O_spi2_cs_n	O	SPI 2-channel chip-selected signal	
7	O_spi1_cs_n	O	SPI 1-channel chip-selected signal	

6 Structure and Functional Description

6.1 Architecture

Figure 6-1 Architecture



The IP design includes SDIO_slave、INTR_Ctrl、Param_Config、RX_FIFO_Ctrl、TX_FIFO_Ctrl、Status_Ctrl、SPI_Interface_Timing_Ctrl modules.

- SDIO_slave: Implements data communication with SDIO host and completes SDIO interface timing control and protocol parse;
- INTR_Ctrl: Implement interrupt control function according to IP internal status;
- Param_Config: Receives the configuration parameters from the SDIO host and delivers them to the corresponding module.
- RX_FIFO_Ctrl: Implements data buffer received from SPI interface;
- TX_FIFO_Ctrl: Implements data buffer from SDIO host;
- Status_Ctrl: Detects and counts the internal interface status and working status of IP;

- SPI_Interface_Timing_Ctrl: Realizes data communication with SPI 4~SPI 1 and completes SPI 4~SPI 1 interface timing control and protocol parse;

6.2 Initialization Flow

1. After power on, the SDIO host needs to configure the IP parameters, the recommended sequence is as follows:
 - a) SBDIFR Register
 - b) IIFR Register
 - c) IER Register (**Enable register last setting**)
2. Then SDIO host can communicate data with SPI according to read/write control operation flow.

6.3 SPI Port Write Operation Flow

6.3.1 Disable Interrupt

When the interrupt of SPI sending FIFO register is disabled, the write operation flow from SDIO Host is as follows (Take SPI 1 for an instance):

1. SDIO Host queries the SPISR register through the CMD53 interface to read the current SPI port operation state.
2. When SPI port is idle, SDIO Host writes the write operation type, SPI channel 1, and the number of bytes to be sent M to the RTCR register via the CMD53 interface.
3. SDIO Host then writes the data to be sent to the TFR register sequentially through the CMD53 interface.
4. Finally the SPI port will be sent from SPI channel 1 to the SPI slave via the preset SPI clock rate.

6.3.2 Enable Interrupt

When the interrupt of SPI sending FIFO register is enabled, the write operation flow from SPI is as follows (Take SPI 1 for an instance):

1. When the SPI port detects that the sending FIFO is idle, it will trigger an interrupt to notify the SDIO Host.
2. After SDIO Host receives the interrupt, it queries the IIR register, and if it is SPI sending FIFO interrupt, it writes the write operation type, SPI channel 1, and number of bytes to be sent M to the RTCR register through the CMD53 interface.

3. SDIO Host then writes the data to be sent to the TFR register sequentially through the CMD53 interface.
4. After the SDIO Host writes the data, it sends a clear interrupt command to SPI port.
5. Finally, SPI port sends data from the SPI sending FIFO to the SPI slave in turn via the preset SPI clock rate from SPI channel 1.

6.4 SPI Port Read Operation Flow

6.4.1 Disable Interrupt

When the interrupt of SPI sending FIFO register is disabled, the read operation flow from SDIO Host is as follows (Take SPI 1 for an instance):

1. SDIO Host queries the SPISR register status via the CMD53 interface.
2. When the SPI port is free, the SDIO Host writes the read operation type, SPI channel 1, the number of bytes to be sent M (for read control of the slave), and the number of bytes to be received N to the RTCR register via the CMD53 interface.
3. SDIO Host then writes the data to be sent to the TFR register sequentially.
4. SPI port will send data in SPI sending FIFO to SPI slave through SPI channel 1 via the preset SPI clock rate. At the same time, the data sent by SPI slave will be sequentially stored in SPI receiving FIFO.
5. SDIO Host queries the SPISR register in real time through the CMD53 interface. When it detects that the SPI port is in the SDIO Host idle state and the receiving FIFO is nonzero, it will read away the data in the SPI receiving FIFO through the read operation on the RFR register sequentially.

6.4.2 Enable Interrupt

When the interrupt of SPI sending FIFO register is enabled, the read operation flow from SPI port is as follows (Take SPI 1 for an instance):

1. When SPI port detects that the sending FIFO is idle, it will trigger an interrupt.
2. After SDIO Host receives interrupt, it queries the IIR register, and if it is SPI sending FIFO interrupt, it writes the read operation type, SPI channel 1, and number of bytes to be sent M to the RTCR register

through the CMD53 interface.

3. SDIO Host then writes the data to be sent to the TFR register sequentially.
4. After the SDIO Host writes the data, it sends a clear interrupt command to SPI port.
5. SPI port will send data in SPI sending FIFO to SPI slave through SPI channel 1 via the preset SPI clock rate. At the same time, the data sent by SPI slave will be sequentially stored in SPI receiving FIFO.

6.5 SPI Receiving FIFO Read Operation

6.5.1 Disable Interrupt

When the interrupt of SPI Receiving FIFO is disabled, the receiving FIFO read data operation flow from SDIO Host is as follows:

1. SDIO Host queries the SPISR register to obtain the number of valid data and SPI port status in the current receiving FIFO.
2. When the SPI port is free and there is data in the receiving FIFO, SDIO Host reads away all the data in the receiving FIFO sequentially from the RFR registers through the CMD53 interface.

6.5.2 Enable Interrupt

When the interrupt of SPI receiving FIFO is enabled, the receiving FIFO read data operation flow from SPI is as follows:

1. When SPI port detects that the receiving FIFO is non-idle and SPI port is idle, it will trigger an interrupt.
2. After the SDIO Host receives an interrupt, it queries the IIR register.
3. If SPI receiving FIFO interrupts, then query the SPISR register to obtain the number of valid data in the currently receiving FIFO.
4. And then read away all the data in the receiving FIFO sequentially from the RFR register via the CMD53 interface.
5. After the SDIO Host completes this operation, then send a clear interrupt instruction to the SPI port.

6.6 Register

6.6.1 Register Description

SDIO- SPI registers include the following registers:

Table 6-1 Register

No.	Name	Address (17-bit width)	Default (32-bit width)	Type	Description
1	CIR	0x10000	0x00000000	Write-only	Clearing Interrupt Registers
2	IITR	0x10001	0x00000000	Read/ Write	Interrupt Interval Time Register
3	IER	0x10002	0x00000000	Read/ Write	Interrupt Enable Register
4	SBD FR	0x10003	0x00000000	Read/ Write	SPI Baud Rate Frequency Division Register
5	RTCR	0x10004	0x00000000	Read/ Write	Transceiver Control Register
6	TFR	0x10005	-	Write-only	Sending FIFO Register
7	RFR	0x00000	-	Read-only	Receiving FIFO Register
8	SPISR	0x00001	0x00000000	Read-only	SPI Status Register
9	IIR	0x00002	0x00000000	Read-only	Interrupt Indicator Register

6.6.2 Register Description

Table 6-2 CIR

Address	Bit(s)	Default	Type	Description
0x10000	Bit [31:1]	-	-	Reserved
	Bit[0]	0	Write	Bit0: 1 Clear interrupts; When Host's query to clear the interrupt register is received and bit0 sent is 1, SDIO slave pulls down the fun1_interrupt pin to interrupts.

Table 6-3 IITR

Address	Bit(s)	Default	Type	Description
0x10001	Bit [31:16]	-	-	Reserved
	Bit [15:0]	0x0000	Read/ Write	Interrupt trigger interval: Range from 1 to 65535

Table 6-4 IER

Address	Bit	Default	Type	Description
0x10002	Bit [31:2]	-	-	Reserved
	Bit[1]	0	Read/ Write	The interrupt of SPI sending FIFO enabled 0: disable; 1: enable;
	Bit[0]	0	Read/ Write	The interrupt of SPI receiving FIFO enabled 0: disable; 1: enable;

Table 6-5 SBDFFR

Address	bit	Default	Type	Description
0x10003	Bit [31:8]	-	-	Reserved
	Bit [7:0]	0x00	Read/ Write	SPI clock rate Range from 0 to 255; SPI Clock Rate = Input Clock Frequency / (4*(Frequency Division Register value + 1))

Table 6-6 RTCR

Address	bit	Default	Type	Description
0x10004	Bit [31:24]	0x00	Read/ Write	Bit [31:26]: Reserved Bit [25:24]: SPI transceiver control 01: SPI operates to send data 10: SPI operates receive data 00: No operation Others: reserved
	Bit [23:16]	0x00	Read/ Write	Bit [23:20]: Reserved Bit[19:16]: SPI channel selection 1000: Corresponds to SPI channel 4. 0100: corresponds to SPI channel 3. 0010: corresponds to SPI channel 2. 0001: Corresponds to SPI channel 1. 0000: No channel selection Others: reserved
	Bit [15:8]	0x00	Read/ Write	Bit [15:8]: the number of data sent by SPI ranges from 0 to 255 (Byte);
	Bit [7:0]	0x00	Read/ Write	Bit [7:0]: the number of data received by SPI Ranges from 0 to 255 (Byte);

Table 6-7 TFR

Address	bit	Default	Type	Description
0x10005	Bit [7:0]	-	Write	The maximum depth of SPI sending FIFO register is 255 bytes.

Table 6-8 RFR

Address	bit	Default	Type	Description
0x00000	Bit [7:0]	-	Read- only	The maximum depth of SPI receiving FIFO register is 255 bytes.

Table 6-9 SPISR

Address	bit	Default	Type	Description
0x00001	Bit [31:24]	0x00	Read-only	Reserved
	Bit [23:16]	0x00	Read-only	Bit[23:16]: the number of valid data in SPI sending FIFO (0~255 byte)
	Bit [15:8]	0x00	Read-only	Bit[15:8]: the number of valid data in SPI receiving FIFO (0~255 byte)
	Bit [7:0]	0x00	Read-only	Bit [7:1]: Reserved Bit [0]: 1: SPI port is in transceiver state 0: SPI port is idle

Table 6-10 IIR

Address	bit	Default	Type	Description
0x00002	Bit [31:3]	0x00	Read-only	Reserved
	Bit [2:1]	0x00	Read-only	SPI Interrupt Type. Bit [2:1]: 10: SPI port idle interrupt, low priority (Triggered when there is no read/write operation on the SPI port). 01: SPI receiving FIFO interrupt, high priority (Triggered when there is data in the receiving FIFO and the SPI port is free). Others: reserved
	Bit [0]	0x00	Read-only	Interrupt status 1: Interrupt waiting 0: No interrupt waiting

6.7 Interface Timing

6.7.1 SPI Interface Timing

Figure 6-2 SPI Write Interface Timing

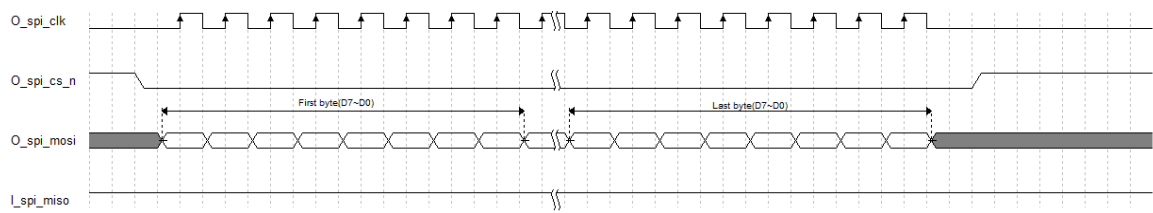
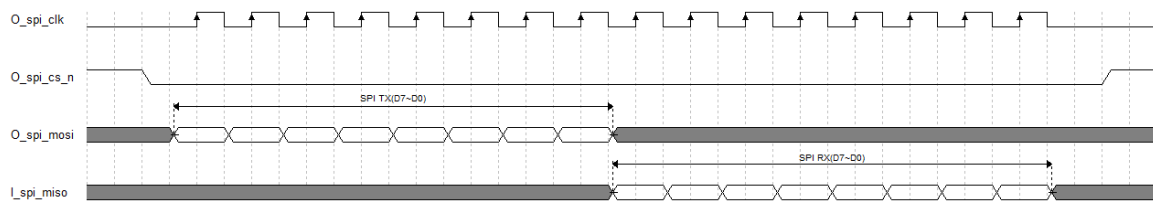


Figure 6-3 SPI Read Interface Timing



6.7.2 SDIO Bus Interface Timing

It is compliant with standard SDIO bus protocol and timing.

7 Parameters Configuration

7.1 Overview

GUI parameter definition for SDIO-SPI

7.2 Parameter

SPI 4 channel	SPI 4 channel select : Yes or No Default: No
SPI 3 channel	SPI 3 channel select : Yes or No Default: No
SPI 2 channel	SPI 2 channel select : Yes or No Default: No
SPI 1 channel	SPI 1 channel select : Yes or No Default: No

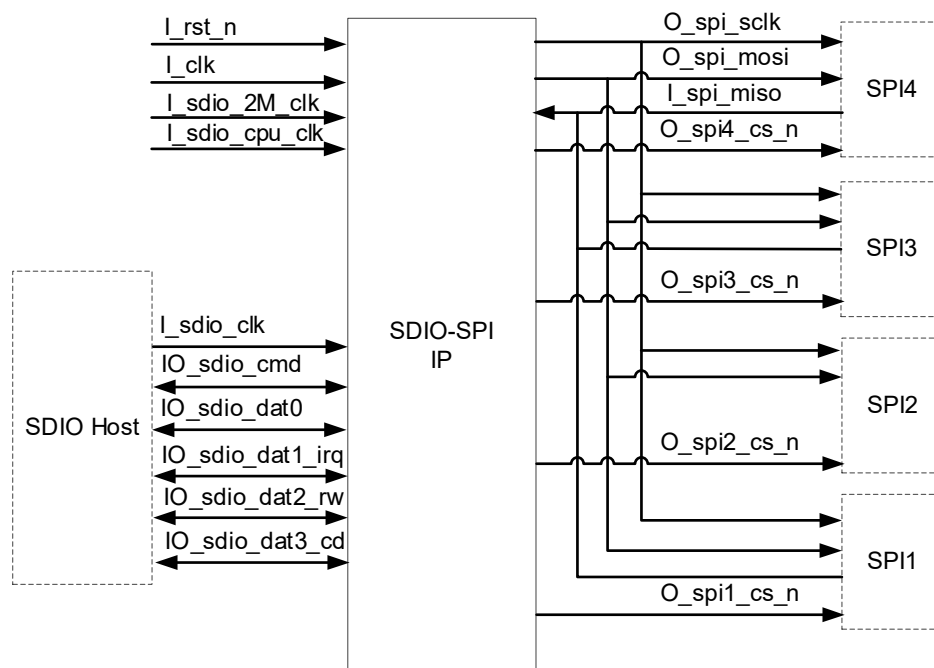
8 Interface Configuration

8.1 Overview

It mainly introduces how to set parameters and generate usable SDIO-SPI Module through GUI interface.

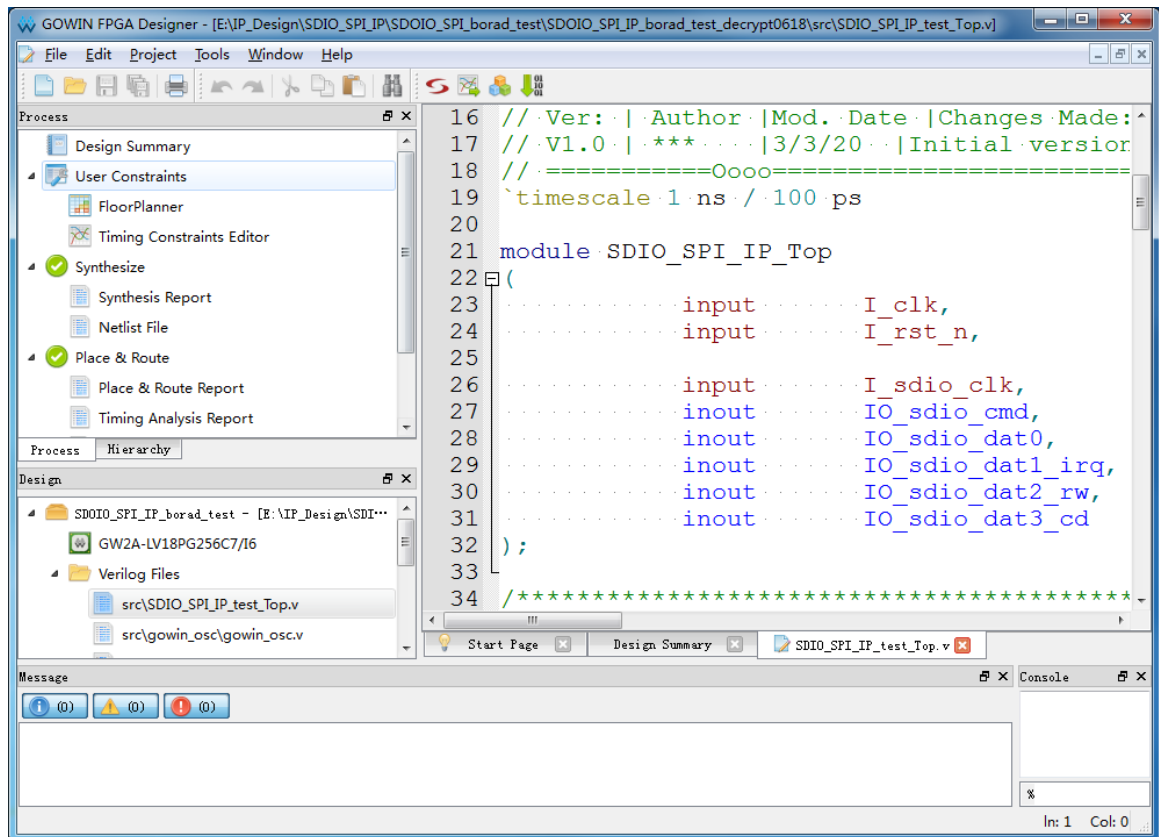
8.2 Application Diagram

Figure 8-1 Application Diagram

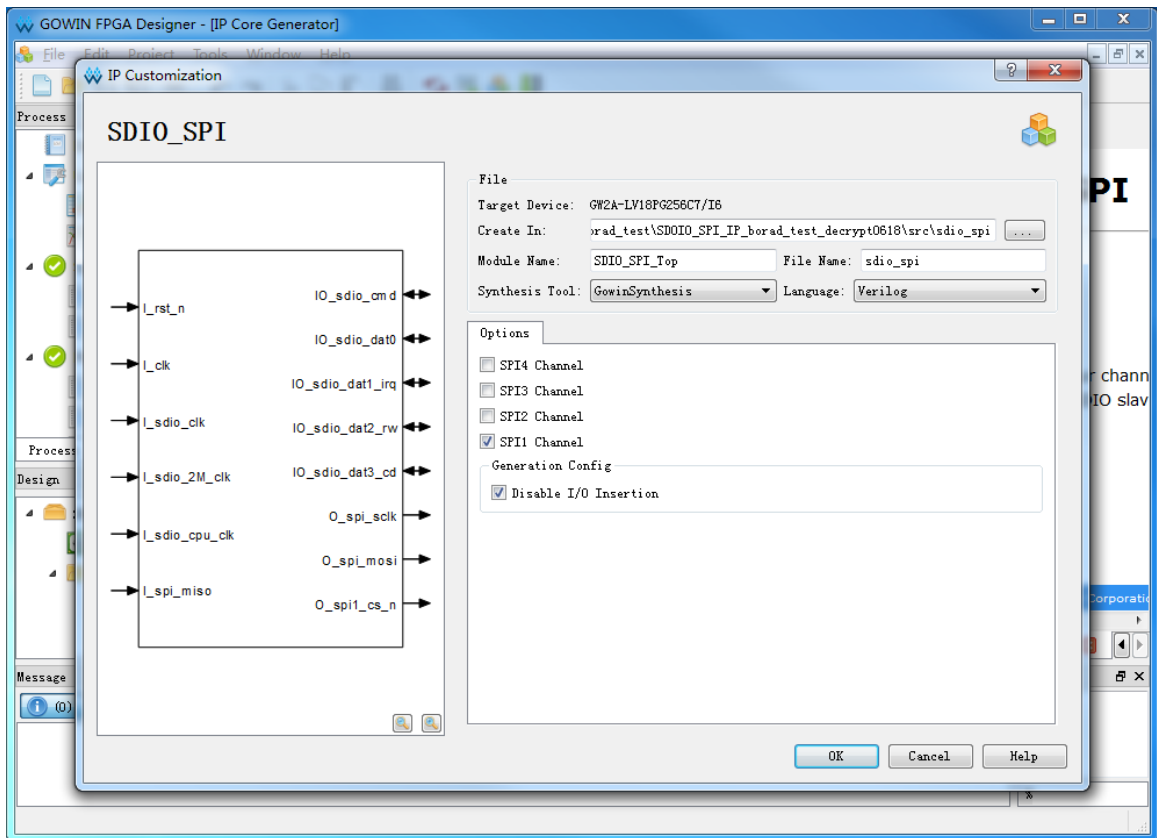


8.3 GUI Call

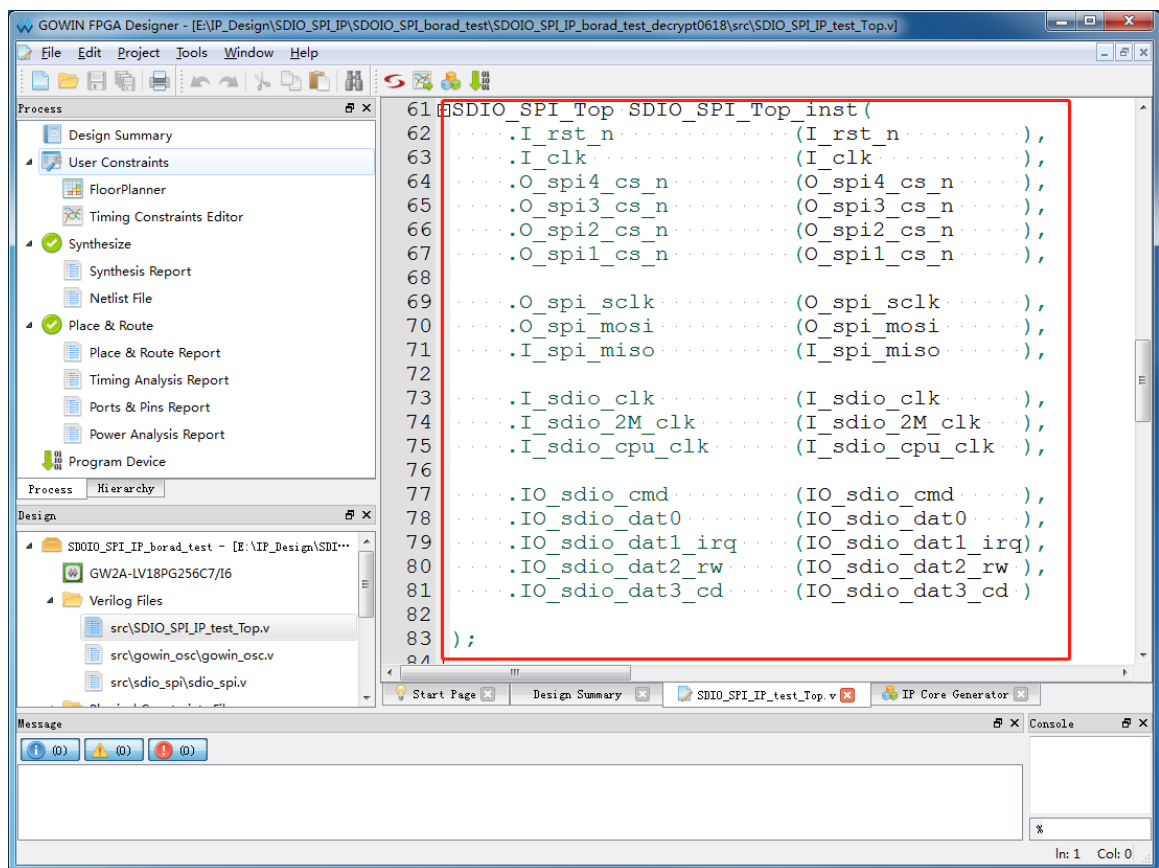
1. Start the Gowin Software and open an project, as shown below:



2. Click "IP Core Generator" from the Tools menu, as shown below:
Configure SDIO_SPI and click "OK" to generate SDIO_SPI Module.



3. Instantiate SDIO_SPI in the user application, as shown below .



Synthesize, Place & Route, download bitstream file to FPGA chip to implement SDIO-SPI function.

