



Gowin SDIO_SPI IP 用户指南

IPUG947-1.0, 2020-05-07

版权所有©2020 广东高云半导体科技股份有限公司

未经本公司书面许可，任何单位和个人都不得擅自摘抄、复制、翻译本档内容的部分或全部，并不得以任何形式传播。

免责声明

本档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除高云半导体在其产品的销售条款和条件中声明的责任之外，高云半导体概不承担任何法律或非法律责任。高云半导体对高云半导体产品的销售和 / 或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。高云半导体对档中包含的文字、图片及其它内容的准确性和完整性不承担任何法律或非法律责任，高云半导体保留修改档中任何内容的权利，恕不另行通知。高云半导体不承诺对这些档进行适时的更新。

版本信息

日期	版本	说明
2020/05/07	1.0	初始版本。

目录

目录	i
图目录	iii
表目录	iv
1 关于本手册	1
1.1 手册内容	1
1.2 相关文档	1
1.3 术语、缩略语	1
1.4 技术支持与反馈	1
2 概述	2
3 特征与性能	3
3.1 特性	3
3.2 性能	3
3.3 资源占用	3
4 系统框图	4
4.1 系统框图	4
5 端口列表	5
5.1 系统信号	5
5.2 SDIO 侧信号	6
5.3 SPI 侧信号	6
6 结构及功能描述	7
6.1 整体结构	7
6.2 初始化流程	8
6.3 SPI 端口写操作流程	8
6.3.1 禁用中断	8
6.3.2 启用中断	8
6.4 SPI 端口读操作流程	9
6.4.1 禁用中断	9
6.4.2 启用中断	9
6.5 SPI 接收 FIFO 读操作	10
6.5.1 禁用中断	10

6.5.2 启用中断	10
6.6 寄存器.....	11
6.6.1 寄存器概述.....	11
6.6.2 寄存器描述.....	11
6.7 接口时序	14
6.7.1 SPI 接口时序.....	14
6.7.2 SDIO 总线接口.....	14
7 参数配置	15
7.1 概述	15
7.2 参数	15
8 界面配置	16
8.1 概述	16
8.2 应用框图	16
8.3 GUI 调用.....	17

图目录

图 4-1 系统框图.....	4
图 6-1 整体结构	7
图 6-2 SPI 写接口时序	14
图 6-3 SPI 读接口时序	14
图 8-1 应用框图.....	16

表目录

表 1-1 术语、缩略语	1
表 3-1 性能参考.....	3
表 5-1 系统信号	5
表 5-2 SDIO 侧信号.....	6
表 5-3 SPI 侧信号	6
表 6-1 寄存器总表.....	11
表 6-2 CIR	11
表 6-3 IITR.....	11
表 6-4 IER	11
表 6-5 SBDFR	12
表 6-6 RTCR.....	12
表 6-7 TFR.....	12
表 6-8 RFR.....	12
表 6-9 SPISR.....	12
表 6-10 IIR.....	13

1 关于本手册

1.1 手册内容

SDIO-SPI 用户手册主要包括功能特点、整体框图、工作原理、信号定义、参数介绍、GUI 调用、接口时序等。目的帮助用户快速了解高云半导体 SDIO- SPI 的产品特性和使用方法，加速用户产品开发。

1.2 相关文档

通过登录高云半导体网站 www.gowinsemi.com.cn 可以下载、查看 FPGA 产品文档。

SDIO IP 使用见 [IPUG905](#), *Gowin SDIO Slave Controller 用户指南*。

1.3 术语、缩略语

表 1-1 中列出了本手册中出现的相关术语、缩略语及相关释义。

表1-1术语、缩略语

术语、缩略语	全称	含义
FPGA	Field Programmable Gate Array	现场可编程门阵列
SDIO	Secure Digital Input and Output	安全数字输入输出
SPI	Serial Peripheral Interface	串行外设接口

1.4 技术支持与反馈

高云半导体提供全方位技术支持，在使用过程中如有任何疑问或建议，可直接与公司联系：

网址：<http://www.gowinsemi.com.cn>

E-mail: support@gowinsemi.com

Tel: +86 755 8262 0391

2 概述

SDIO（Secure Digital Input and Output）在 SD 标准上定义了一种外设接口。

SPI（Serial Peripheral Interface）是一种高速、全双工、同步的通信总线。

为了更好地服务 FPGA 用户，降低用户的系统开发难度，提高产品开发速度，设计一款 SDIO-SPI IP，实现 SDIO 与 SPI 间的接口转换功能。

SDIO-SPI 通过软件界面调用后，可通过 Gowin 软件综合、布局布线适用到 GW1N、GW1NR、GW1NS、GW1NZ、GW2A、GW2AR 等系列 FPGA 产品。

3 特征与性能

3.1 特性

- 实现 SDIO 与 SPI 接口之间的数据通信；
- SDIO 接口为从端模式；
- SPI 接口为主端模式，最大支持 4 路 SPI 通道；
- SPI 总线时钟速率可通过寄存器配置灵活设置；
- 支持中断方式；
- SPI 收 FIFO 深度 255Bytes；
- SPI 发 FIFO 深度 255Bytes；
- SDIO Host 端所有用户侧的操作都通过 CMD53 接口来实现，包括 Host 端清中断寄存器操作，即通过 CMD53 接口读写 Funtion1 寄存器实现与 SPI 侧的参数配置、状态查询及数据读写。
- SDIO CMD53 命令 OP Code 固定取值为 0, 选取 Multi byte R/W to fixed address 的方式完成与 SPI 接口的数据交互；
- 完全可综合；
- 设计语言为 Verilog。

3.2 性能

SDIO-SPI 的工作频率取决于 SDIO 主端提供的时钟频率和 IP 在所选芯片中支持的最大工作频率。

3.3 资源占用

表3-1 性能参考

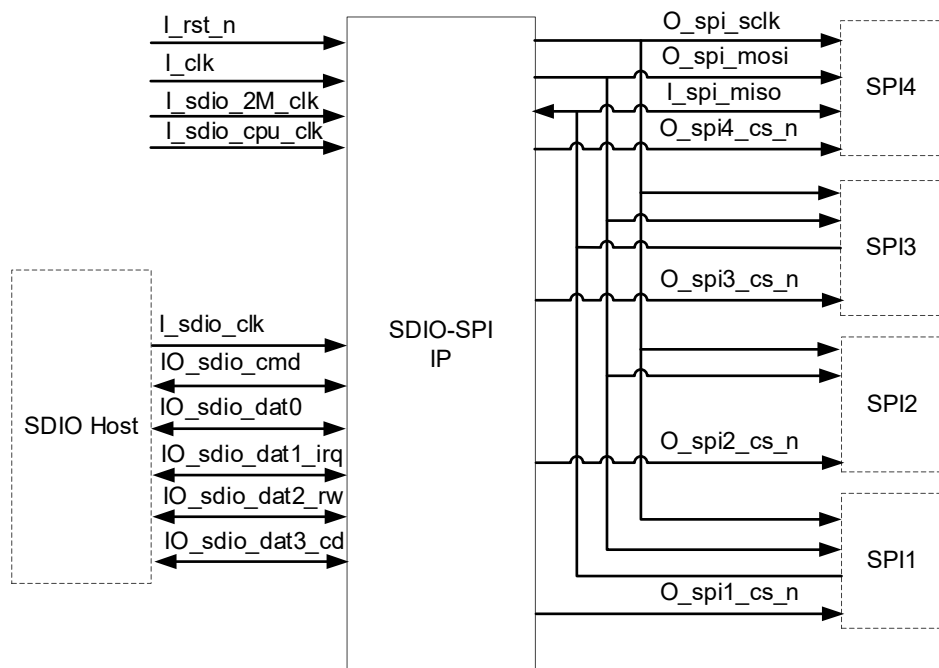
芯片型号	编程语言	LUT4资源	REG资源
GW2A-LV18PG256C7/I6	Verilog	4039	2155

4 系统框图

4.1 系统框图

SDIO-SPI 介于 SDIO 主端与 SPI 终端之间，实现 SDIO 到 SPI 的接口转换，完成数据通信功能。SDIO 主端包括时钟、命令、数据、中断等信号，SPI 侧信号包括数据、时钟、片选。本地输入包括复位和时钟信号。

图4-1 系统框图



注!

- 正常情况下，SPI 接口侧通过中断来触发 SDIO Host 端进行操作；在 SDIO Host 端禁用中断寄存器所有中断后，SDIO Host 可以通过查询来读取状态和控制数据收发操作。
- 如果启用中断功能，则 SPI 接口侧触发中断后，待用户收到中断时，需先查询 IIR 状态值，然后根据 IIR 的指示执行相应操作（如果用户跳过也可以），最后用户发送清中断指令，通知 SPI 接口侧清除本次中断。

5 端口列表

5.1 系统信号

表5-1系统信号

序号	信号名称	方向	描述	备注
1	l_rst_n	↓	复位	所有信号输入输出方向均以IP为参考
2	l_clk	↓	系统时钟，用于内部逻辑及SPI接口处理	
3	l_sdio_2M_clk	↓	SDIO从端所需时钟	
4	l_sdio_cpu_clk	↓	SDIO从端所需时钟	

5.2 SDIO 侧信号

表5-2 SDIO 侧信号

序号	信号名称	方向	描述	备注
1	l_sdio_clk	I	SDIO 时钟输入信号	所有信号输入输出方向均以IP为参考
2	IO_sdio_cmd	IO	SDIO Command/Response 信号	
3	IO_sdio_dat0	IO	SDIO 1bit 模式为dat 信号输入, SDIO 4bit 模式为 dat0 信号输入	
4	IO_sdio_dat1_irq	IO	SDIO 1bit 模式为中断信号输入, SDIO 4bit 模式为 dat1 和中断复用信号输入	
5	IO_sdio_dat2_rw	IO	SDIO 1bit 模式为Read Wait 信号输入, SDIO 4bit 模式为dat2 和Read Wait 复用信号输入	
6	IO_sdio_dat3_cd	IO	SDIO 1bit 模式未定义, SDIO 4bit 模式为dat3 信号输入	

5.3 SPI 侧信号

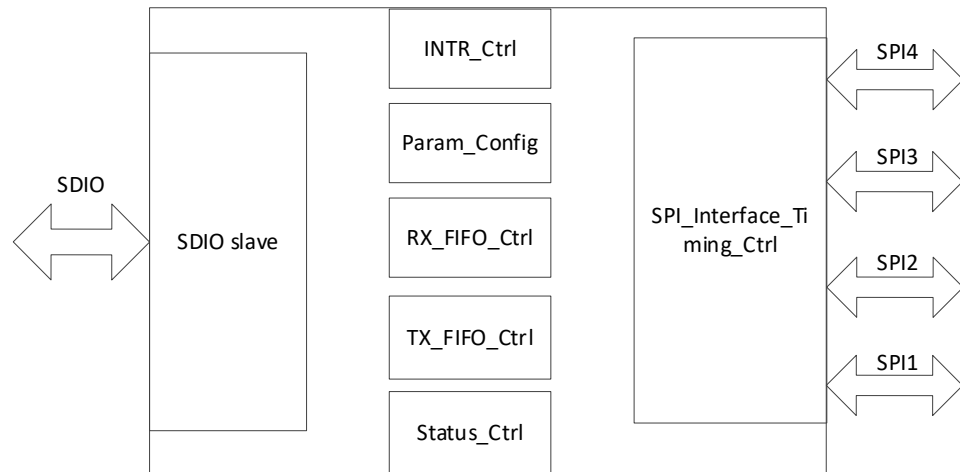
表5-3 SPI侧信号

序号	信号名称	方向	描述	备注
1	O_spi_sclk	O	SPI时钟信号	所有信号输入输出方向均以IP为参考
2	O_spi_mosi	O	SPI发送数据	
3	I_spi_miso	I	SPI接收数据	
4	O_spi4_cs_n	O	SPI4通道片选信号	
5	O_spi3_cs_n	O	SPI3通道片选信号	
6	O_spi2_cs_n	O	SPI2通道片选信号	
7	O_spi1_cs_n	O	SPI1通道片选信号	

6 结构及功能描述

6.1 整体结构

图6-1整体结构



整个 IP 设计包括 SDIO_slave、INTR_Ctrl、Param_Config、RX_FIFO_Ctrl、TX_FIFO_Ctrl、Status_Ctrl、SPI_Interface_Timing_Ctrl、模块。

- SDIO_slave: 实现与 SDIO 主端数据通信，完成 SDIO 接口时序控制和协议解析功能；
- INTR_Ctrl: 根据 IP 内部状态完成相应的中断控制功能；
- Param_Config: 接收 SDIO 主端发送来的配置参数，并下发到对应模块；
- RX_FIFO_Ctrl: 实现对从 SPI 接口接收来的数据缓存；
- TX_FIFO_Ctrl: 实现对 SDIO 主端发送来的数据缓存；
- Status_Ctrl: 实现对 IP 内部接口状态和工作状态的检测和统计；
- SPI_Interface_Timing_Ctrl: 实现与 SPI4~SPI1 端数据通信，完成 SPI4~SPI1 接口时序控制和协议解析功能。

6.2 初始化流程

1. 上电后，SDIO 主端首先需要对 IP 进行参数配置，建议顺序如下：
 - a) SBDIFR 寄存器
 - b) IITR 寄存器
 - c) IER 寄存器（使能寄存器最后设置）
2. 然后 SDIO 主端可以根据读写控制操作流程进行与 SPI 端数据通信。

6.3 SPI 端口写操作流程

6.3.1 禁用中断

禁用 SPI 发送 FIFO 寄存器中断情况，SDIO Host 主端发起写操作流程如下（以 SPI1 为例）：

1. SDIO Host 通过 CMD53 接口查询 SPISR 寄存器，读取当前 SPI 端口工作状态；
2. 当 SPI 端口为空闲时，SDIO Host 通过 CMD53 接口向 RTCR 寄存器写入写操作类型、SPI 通道 1、待发送字节 M 量；
3. 然后 SDIO Host 会通过 CMD53 接口把要发送的数据依次写入 TFR 寄存器；
4. 最后 SPI 端口会通过设定的 SPI 时钟速率由 SPI 通道 1 发送给 SPI 从端。

6.3.2 启用中断

启用 SPI 发送 FIFO 寄存器中断情况，SPI 端发起写操作流程如下（以 SPI1 为例）：

1. 当 SPI 端口检测到发送 FIFO 为空闲时，触发中断通知 SIDO Host；
2. SDIO Host 收到中断后，查询 IIR 寄存器，若为 SPI 发送 FIFO 中断，则通过 CMD53 接口向 RTCR 寄存器写入写操作类型、SPI 通道 1、待发送字节数 M；
3. 接下来 SDIO Host 会通过 CMD53 接口把要发送的数据依次写入 TFR 寄存器；
4. 待 SDIO Host 写完数据后，再发送清中断指令给 SPI 端口。
5. 最后 SPI 端口会通过设定的 SPI 时钟速率由 SPI 通道 1 把 SPI 发送 FIFO 中数据依次发送给 SPI 从端。

6.4 SPI 端口读操作流程

6.4.1 禁用中断

禁用 SPI 发送 FIFO 寄存器中断情况，SDIO Host 端发起读操作流程如下（以 SPI1 为例）：

1. SDIO Host 通过 CMD53 接口查询 SPISR 寄存器状态；
2. 当 SPI 端口为空闲时，SDIO Host 通过 CMD53 接口向 RTCR 寄存器写入读操作类型、SPI 通道 1、待发送字节数 M（用于对从端的读控制）和待接收字节数 N；
3. 然后 SDIO Host 把要发送的数据依次写入 TFR 寄存器；
4. SPI 端口会通过设定的 SPI 时钟速率由 SPI 通道 1 把 SPI 发送 FIFO 中数据发送给 SPI 从端，同时把 SPI 从端发送来的数据依次存入 SPI 接收 FIFO；
5. SDIO Host 通过 CMD53 接口实时查询 SPISR 寄存器，当检测到 SPI 端口处于 SDIO Host 空闲状态且接收 FIFO 非零时，会通过 RFR 寄存器的读操作依次读走 SPI 接收 FIFO 中的数据。

6.4.2 启用中断

启用 SPI 发送 FIFO 寄存器中断情况，SPI 端口发起读操作流程如下（以 SPI1 为例）：

1. 当 SPI 端口检测到发送 FIFO 为空闲时，触发中断；
2. SDIO Host 收到中断后，查询 IIR 寄存器，若为 SPI 发送 FIFO 中断，则通过 CMD53 接口向 RTCR 寄存器写入读操作类型、SPI 通道 1、待发送字节数 M（用于对从端的读控制）和待接收字节数 N；
3. 然后 SDIO Host 把要发送的数据依次写入 TFR 寄存器；
4. 待 SDIO Host 写完数据后，再发送清中断指令给 SPI 端口。
5. SPI 端口会通过设定的 SPI 时钟速率由 SPI 通道 1 把 SPI 发送 FIFO 中数据发送给 SPI 从端，同时把 SPI 从端接收来的数据依次存入 SPI 接收 FIFO。

6.5 SPI 接收 FIFO 读操作

6.5.1 禁用中断

禁用 SPI 接收 FIFO 中断情况，SDIO Host 端发起接收 FIFO 读数据操作流程如下：

1. SDIO Host 通过查询 SPISR 寄存器，获取当前接收 FIFO 中的有效数据数量及 SPI 端口状态；
2. 当 SPI 端口空闲且接收 FIFO 中有数据时，SDIO Host 会通过 CMD53 接口从 RFR 寄存器依次读走接收 FIFO 中的所有数据。

6.5.2 启用中断

启用 SPI 接收 FIFO 中断情况，SPI 端发起接收 FIFO 读数据操作流程如下：

1. 当 SPI 端口检测到接收 FIFO 为非空闲且 SPI 端口为空闲时，触发中断；
2. SDIO Host 收到中断后，查询 IIR 寄存器；
3. 若为 SPI 接收 FIFO 中断，则再通过查询 SPISR 寄存器，获取当前接收 FIFO 中的有效数据数量；
4. 然后再通过 CMD53 接口从 RFR 寄存器依次读走接收 FIFO 中的所有数据；
5. 待 SDIO Host 完成本次操作后，再发送清中断指令给 SPI 端口。

6.6 寄存器

6.6.1 寄存器概述

SDIO-SPI 寄存器包括如下：

表6-1寄存器总表

序号	寄存器名称	寄存器地址 (17bit位宽)	默认值 (32bit位宽)	操作类型	描述
1	CIR	0x10000	0x00000000	只写	清中断寄存器
2	IITR	0x10001	0x00000000	读/写	中断间隔时间寄存器
3	IER	0x10002	0x00000000	读/写	中断使能寄存器
4	SBDIFR	0x10003	0x00000000	读/写	SPI波特率分频寄存器
5	RTCR	0x10004	0x00000000	读/写	收发控制寄存器
6	TFR	0x10005	-	只写	发送FIFO寄存器
7	RFR	0x00000	-	只读	接收FIFO寄存器
8	SPISR	0x00001	0x00000000	只读	SPI状态寄存器
9	IIR	0x00002	0x00000000	只读	中断指示寄存器

6.6.2 寄存器描述

表6-2 CIR

寄存器地址	比特位	默认值	操作类型	描述
0x10000	Bit[31:1]	-	-	保留
	Bit[0]	0	写	Bit0: 1 清中断； 当收到Host端查询清中断寄存器且发送的bit0为1时，SDIO从端拉低fun1_interrupt管脚，以清除中断。

表6-3 IITR

寄存器地址	比特位	默认值	操作类型	描述
0x10001	Bit[31:16]	-	-	保留
	Bit[15:0]	0x0000	读/写	中断触发间隔时间：取值范围1~65535。

表6-4 IER

寄存器地址	Bit位	默认值	操作类型	描述
0x10002	Bit[31:2]	-	-	保留
	Bit[1]	0	读/写	SPI发送FIFO中断使能 0: 禁用, 1: 使能;
	Bit[0]	0	读/写	SPI接收FIFO中断使能 0: 禁用, 1: 使能;

表6-5 SBDPR

寄存器地址	Bit位	默认值	操作类型	描述
0x10003	Bit[31:8]	-	-	保留
	Bit[7:0]	0x00	读/写	SPI时钟速率：取值范围0~255； SPI时钟速率=输入时钟频率/（4* （分频寄存器值+1））

表6-6 RTCR

寄存器地址	Bit位	默认值	操作类型	描述
0x10004	Bit[31:24]	0x00	读/写	Bit[31:26]:保留 Bit[25:24]: SPI收发控制 01: SPI为发送数据操作 10: SPI为接收数据操作 00: 无操作 其他: 保留
	Bit[23:16]	0x00	读/写	Bit[23:20]:保留 Bit[19:16]: SPI通道选择 1000: 对应SPI通道4; 0100: 对应SPI通道3; 0010: 对应SPI通道2; 0001: 对应SPI通道1; 0000: 无通道选择 其他: 保留
	Bit[15:8]	0x00	读/写	Bit[15:8]: SPI发送数据个数, 取值范围0~255（单位byte）
	Bit[7:0]	0x00	读/写	Bit[7:0]: SPI接收个数, 取值范围0~255（单位byte）

表6-7 TFR

寄存器地址	Bit位	默认值	操作类型	描述
0x10005	Bit[7:0]	-	写	SPI发送FIFO寄存器，最大深度 255bytes

表6-8 RFR

寄存器地址	Bit位	默认值	操作类型	描述
0x00000	Bit[7:0]	-	只读	SPI接收FIFO寄存器，最大深度 255bytes

表6-9 SPISR

寄存器地址	Bit位	默认值	操作类型	描述
0x00001	Bit[31:24]	0x00	只读	保留
	Bit[23:16]	0x00	只读	Bit[23:16]: SPI发送FIFO中的有效 数据个数（单位Byte，0~255）

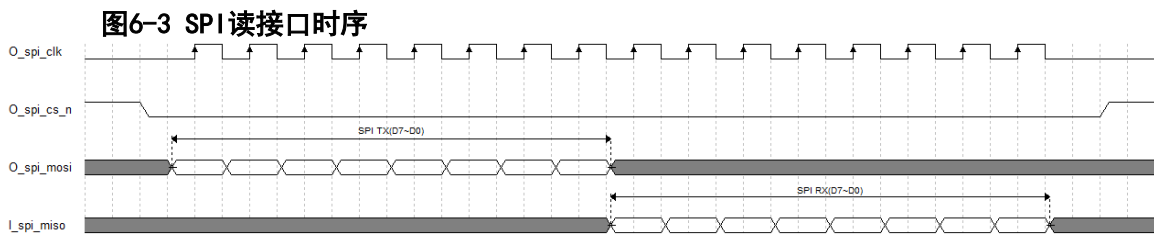
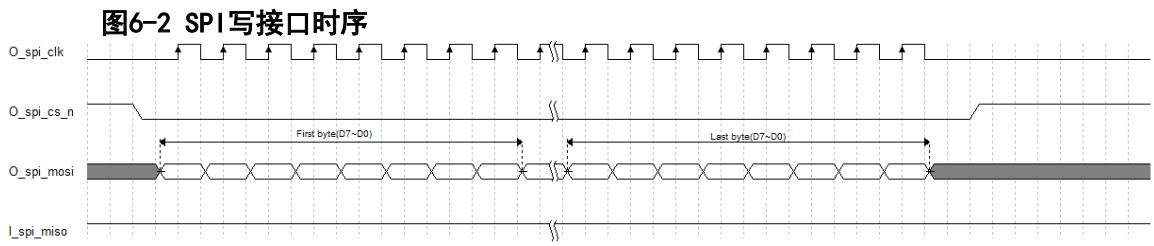
	Bit[15:8]	0x00	只读	Bit[15:8]: SPI接收FIFO中的有效数据个数(单位Byte, 0~255)
	Bit[7:0]	0x00	只读	Bit[7:1]: 保留 Bit[0]: 1: SPI端口处于收发状态 0: SPI端口处于空闲状态

表6-10 IIR

寄存器地址	Bit位	默认值	操作类型	描述
0x00002	Bit[31:3]	0x00	只读	保留
	Bit[2:1]	0x00	只读	SPI中断类型。 Bit[2: 1]: 10: SPI端口空闲中断, 低优先级(当SPI端口无读写操作时触发); 01: SPI接收FIFO中断, 高优先级(当接收FIFO中有数据且SPI端口空闲时触发); 其他: 保留
	Bit[0]	0x00	只读	中断状态。 1: 中断等待中; 0: 无中断等待

6.7 接口时序

6.7.1 SPI 接口时序



6.7.2 SDIO 总线接口

符合标准 SDIO 总线协议和时序。

7 参数配置

7.1 概述

SDIO-SPI 的 GUI 参数定义。

7.2 参数

SPI4 channel	SPI4 channel select : Yes or No Default: No
SPI3 channel	SPI3 channel select : Yes or No Default: No
SPI2 channel	SPI2 channel select : Yes or No Default: No
SPI1 channel	SPI1 channel select : Yes or No Default: No

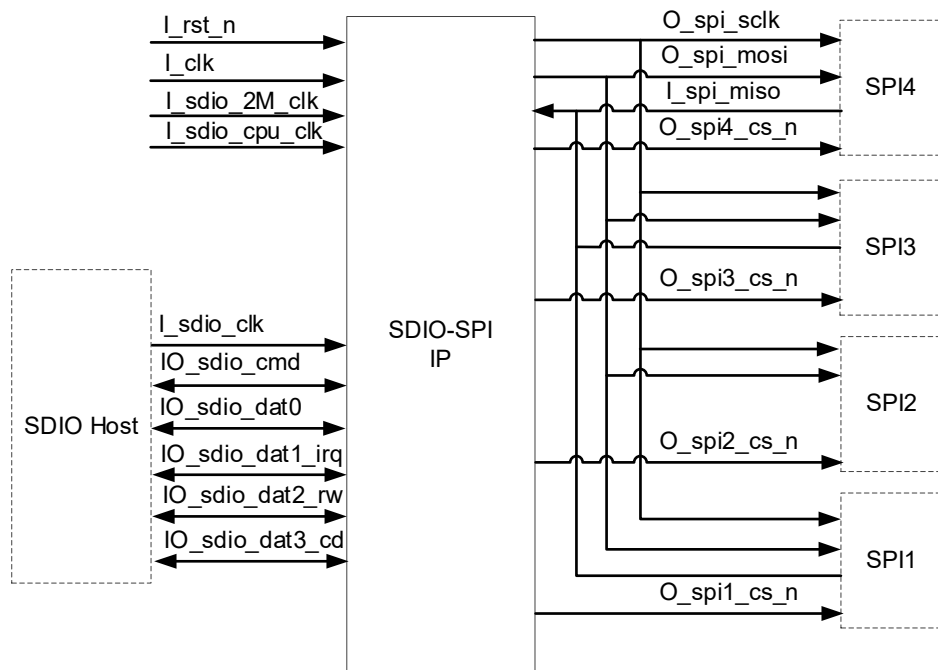
8 界面配置

8.1 概述

主要介绍如何通过 GUI 界面设置参数、生成可用 SDIO-SPI Module。

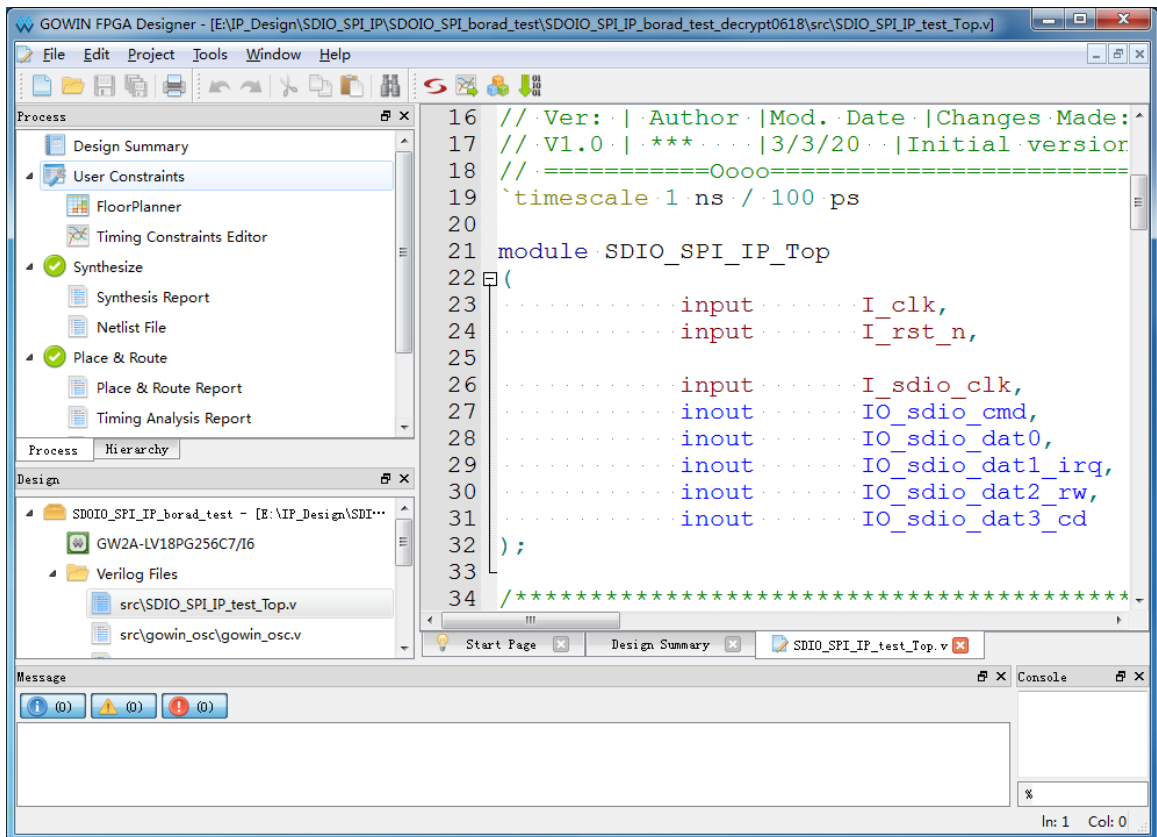
8.2 应用框图

图8-1 应用框图



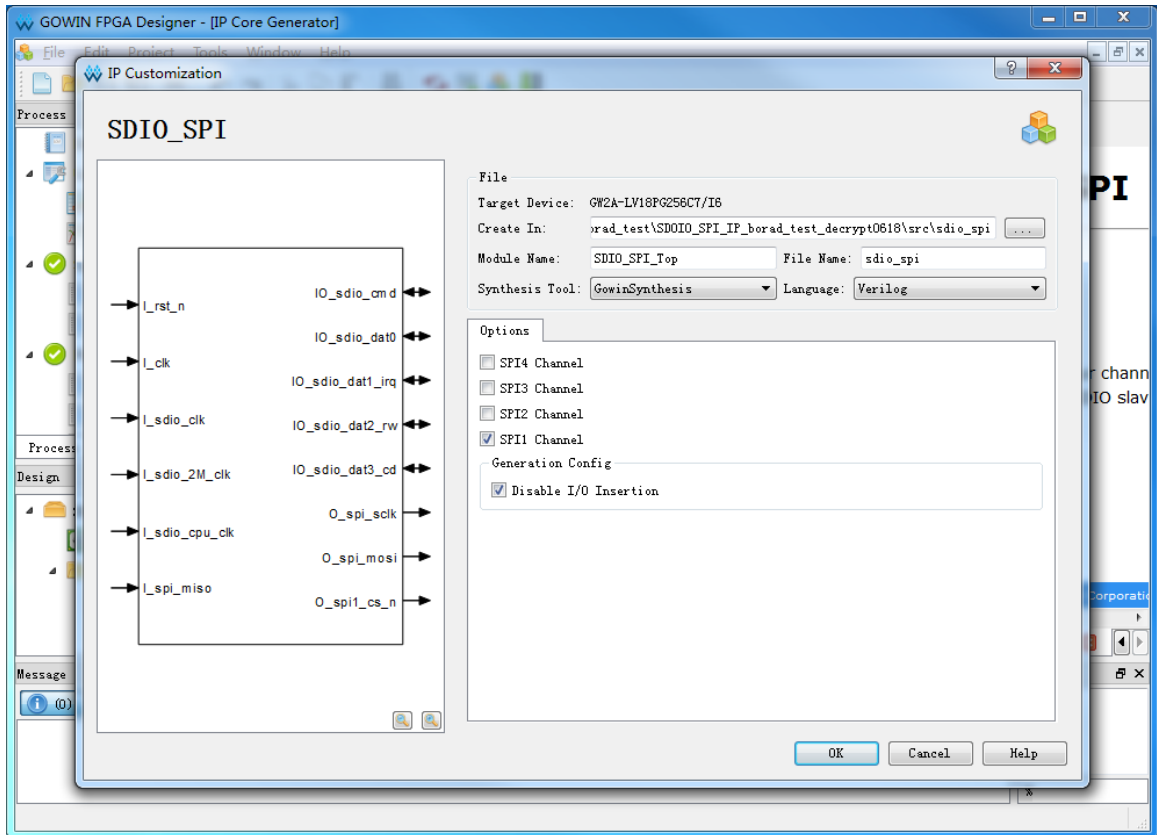
8.3 GUI 调用

1. 启动 GowinYunYuan 软件后，打开工程，如下图：

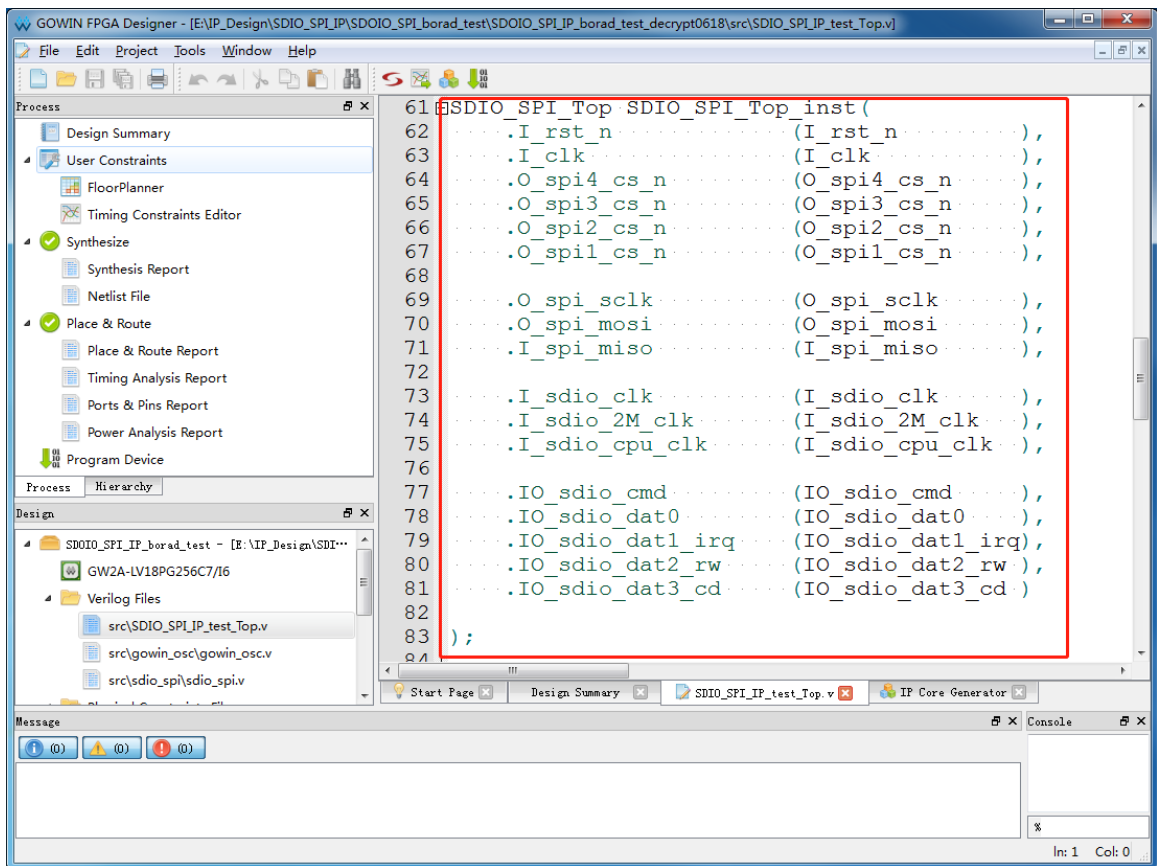


2. 在 Tools 菜单下点击“IP Core Generator”，如下图：

调用的 SDIO_SPI ， 然后点击“OK”后生成 SDIO_SPI Module。



3. 在用户程序中例化 SDIO_SPI，如下图：



综合、P&R、下载 bitstream 文件到 FPGA 芯片后可实现 SDIO- SPI 功能。

