



# **SPI IP 用户指南**

上海安路信息科技股份有限公司

IPUG108 (v1.0) 2021 年 7 月

**Confidential**



# 目 录

目 录 .....	I
1 IP 概述 .....	1
1.1 SPI 协议简介 .....	1
1.2 器件支持 .....	2
2 IP 详情 .....	3
2.1 IP 例化 .....	3
2.1.1 配置 GUI .....	3
2.1.2 生成文件 .....	5
2.2 端口 .....	6
2.2.1 SPI Master 端口 .....	6
2.2.2 SPI Slave 端口 .....	7
2.3 用户可存取寄存器 .....	8
2.3.1 SPI Master 的 UAR .....	8
2.3.2 SPI Slave 的 UAR .....	9
2.4 时序 .....	10
2.4.1 写 UAR .....	10
2.4.2 读 UAR .....	11
2.4.3 SPI Master IP 发起 SPI 传输 .....	11
2.4.4 TRDY 和 TOE .....	12
2.4.5 RPRDY, RRDY 和 ROE .....	12
2.4.6 “部分接收阈值” 功能 .....	13
2.5 资源占用和时序性能 .....	14
3 版本信息 .....	15



---

免责声明.....	15
-----------	----

# 1 IP 概述

SPI，即串行外设接口（Serial Peripheral Interface）是一种同步串行通信协议，适用于短距离通信。该协议自 Motorola 公司于 20 世纪 80 年代提出至今，已经广泛应用在嵌入式系统当中，成为许多传感器，ADC，DAC，存储器以及控制器件的通信接口。值得注意的是，SPI 是一种约定俗称的协议，并无正式的协议标准，造成该协议的形式多样，不同的器件厂商也会定义自己的协议版本。安路的 SPI IP 核具有极高的灵活性，用户可通过配置 SPI IP 核的主/从类型，时钟极性，时钟相位，数据宽度，数据传输顺序，从设备数量以及部分接收阈值等，使 SPI IP 核兼容市场上常见的 SPI 接口。

## 1.1 SPI 协议简介

SPI 器件通信的拓扑通常为一主 (Master) 一从 (Slave) 或一主多从，工作模式为全双工模式，读写操作只能由主设备发起。

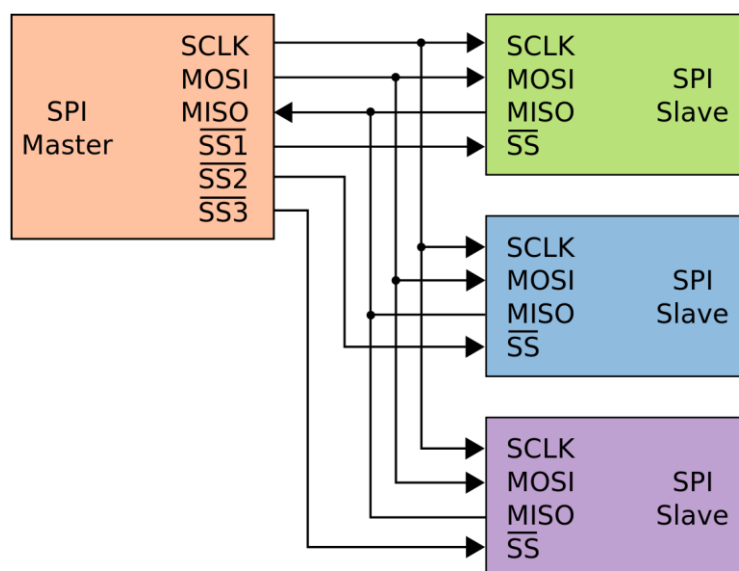


图 1-1 SPI 主设备与多个独立的从设备相连

SPI 端口包含 2 条数据信号线 MISO 和 MOSI，1 条同步时钟信号线 SCLK 以及 1 条或多条从设备片选信号线：

- MOSI (Master Output Slave Input) - 主设备输出/从设备输入信号；
- MISO (Master Input Slave Output) - 主设备输入/从设备输出信号；
- SCLK (Serial Clock) - 串行时钟信号，由主设备产生，用来同步数据比特；
- SS (Slave Select) /CS (Chip Select) - 从设备片选信号，低电平有效，由主设备控制；

SPI 协议中有时钟极性 (CPOL) 和时钟相位 (CPHA) 两个参数，通过对这两个参数的设置可以产生 4 种时序模式：

- CPOL (Clock Polarity) — 可被设置为 0 或 1。当 CPOL=0 时，SCLK 在空闲状态下为低电平。当 CPOL=1 时，SCLK 在空闲状态下为高电平。



- CPHA (Clock Phase) :---可被设置为 0 或 1。当 CPHA=0 时，数据在 SCLK 的前沿被锁存，数据在 SCLK 的后沿被更新。当 CPHA=1 时，数据在 SCLK 的后沿被锁存，数据在 SCLK 的前沿被更新。

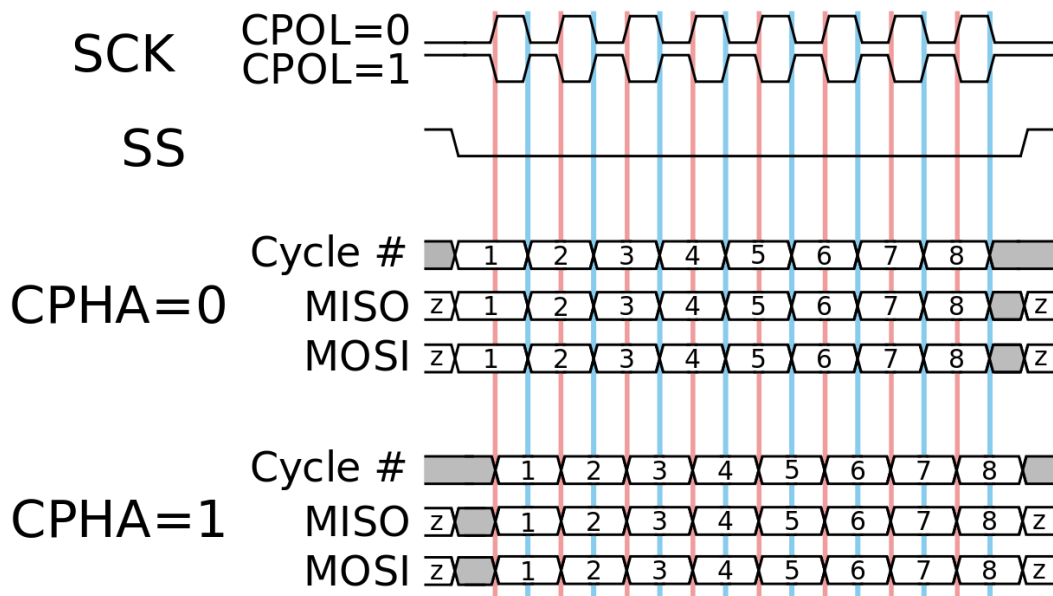


图 1-2 SPI 时序图，红线代表时钟的前沿，蓝线代表时钟的后沿

## 1.2 器件支持

SPI IP 核支持安路全系列 FPGA 器件。



## 2 IP 详情

### 2.1 IP 例化

#### 2.1.1 配置 GUI

用户可在 TD 的 IP Generator 中来配置并生成 IP 核。

Master/Slave			
Type:	Master ▼		
Number of slave select(SS_n) signals:	1 ▼		
Timing			
System clock(SYS_Clk) rate:	<input type="text"/>	Hz	
SPI clock(SCLK) rate:	<input type="text"/>	Hz	Actual SCLK rate: <input type="text"/> Hz
SS_n setup time:	<input type="text"/>	ns	Actual SS_n setup time: <input type="text"/> ns
SS_n hold time:	<input type="text"/>	ns	Actual SS_n hold tiime: <input type="text"/> ns
SCLK polarity:	1 ▼		
SCLK phase:	1 ▼		
Data			
Width:	8 ▼		
Shift direction:	MSB First ▼		
RxPartial Threshold Function			
<input type="checkbox"/> Enable RxPartial threshold			
RxPartial threshold:	1 ▼		

图 2-1 SPI IP 配置 GUI

表 2-1 SPI IP 配置 GUI 中的组件说明



组件名称	有效条件	取值范围	描述
Type	任何条件	Master, Slave	配置该 IP 作为主设备或从设备
Number of slave select (SS_n) signals	Type = Master	1, 2, 3 ....32	从设备片选信号线数量
System clock(Sys_Clk) rate	Type = Master	2~200000000	IP 内部逻辑工作的参考输入时钟频率
SPI clock (SCLK) rate	Type = Master	不超过 Sys_Clk 频率的 1/2	SCLK 时钟频率
Actual SCLK rate	Type = Master	NA	该组件为只读，在用户设定 System clock(Sys_Clk) rate 和 SPI clock (SCLK) rate 后，IP Generator 会告知用户实际能达到的 SCLK 时钟频率
SS_n setup time <sup>(1)</sup>	Type = Master	正整数	SS_n 的建立时间
Actual SS_n setup time	Type = Master	NA	该组件为只读，在用户设定 System clock(Sys_Clk) rate 和 SS_n setup time 后，IP Generator 会告知用户实际能达到的 SS_n 建立时间
SS_n hold time <sup>(1)</sup>	Type = Master	正整数	SS_n 的保持时间
Actual SS_n hold time	Type = Master	NA	在用户设定 System clock(Sys_Clk) rate 和 SS_n hold time 后，IP Generator 会告知用户实际能达到的 SS_n 建立时间
SCLK polarity <sup>(2)</sup>	Type = Slave	0, 1	SCLK 时钟极性
SCLK phase <sup>(2)</sup>	Type = Slave	0, 1	SCLK 时钟相位
Width	任何条件	8, 9, 10 ....32	SPI 每轮传输的数据位数
Shift direction <sup>(2)</sup>	Type = Slave	MSB First, LSB First	数据传输顺序
Enable RxPartial threshold <sup>(3)</sup>	Type = Slave	勾选, 不勾选	使能“部分接收阈值”功能



RxPartial Threshold	Type = Slave 且勾选 Enable RxPartial threshold	1,2,3...Data Width -1	设置部分接收阈值
---------------------	---	-----------------------	----------

(1) 本文对 SS\_n 的建立时间的定义为，SS\_n 下降沿到 SCLK 第一个前沿间的时间；对 SS\_n 的保持时间的定义为，SCLK 最后一个后沿到 SS\_n 的上升沿之间的时间。安路 SPI Slave IP 对 SS\_n 的建立与保持时间的要求为大于等于 1 个 Sys\_Clk 时钟周期。

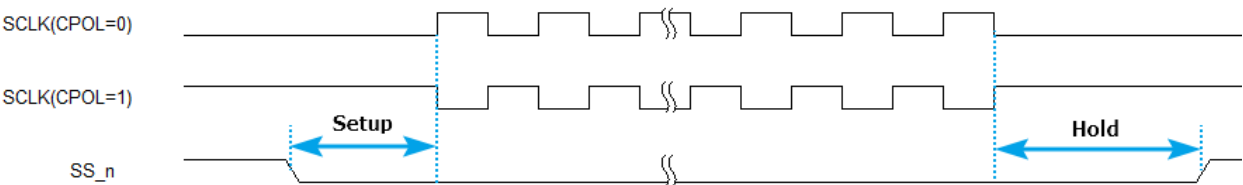


图 2-2 SS\_n 的建立时间与保持时间示意图

- (2) 当 Type = Master 时，SCLK 时钟极性，时钟相位，数据传输顺序由用户通过 UAR (User-Accessible Register) 来动态配置。详情见本文 2.3.1 节。
- (3) “部分接收阈值”功能被使能后，用户可提前读取 SPI Slave 接收到的部分数据，详情见本文 2.4.6 节。

2.1.2 生成文件

表 2-2 IP Generator 生成文件列表

文件名	描述
USERNAME_gate.v	IP 核的 Netlist 文件，用户将该文件添加到 TD 工程中后可例化该 IP 核。该 Netlist 文件可用于 Implementation 和 Gate-level simulation（需要仿真软件编译相应器件的仿真库）
USERNAME.ipc	IP 核的配置文件，用户可用 IP Generator 加载该文件来查看或更改 IP 配置

用户在使用 IP Generator 生成 IP 核后，TD 会弹出对话框如图 2-3 所示，如果用户希望将该 IP 核添加到当前 TD 工程中，需勾选 “\_gate.v” 前的选项框，如图中红框内所示，并点击 “Yes”。



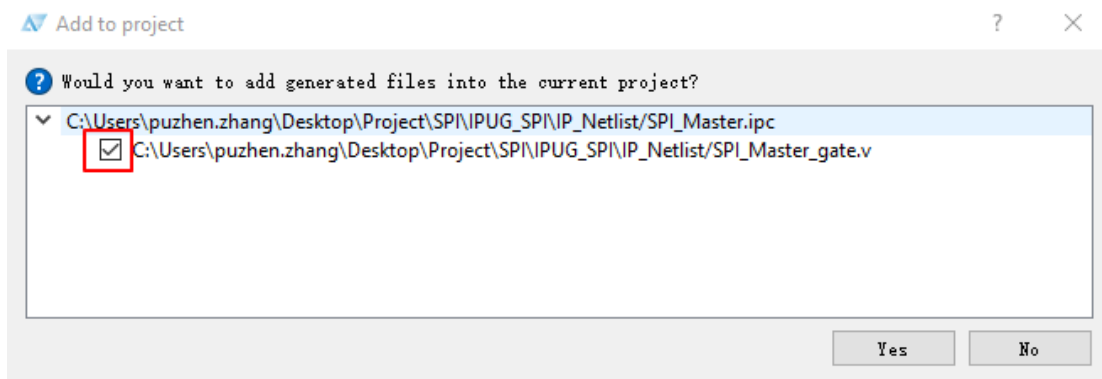


图 2-3 添加 Netlist 文件到 TD 工程

## 2.2 端口

### 2.2.1 SPI Master 端口

表 2-3 当 Type = Master 时，SPI IP 的端口信号

	端口名	位宽	方向	说明
面向片内用户逻辑的端口	Sys_Clk	1	In	SPI IP 核内部逻辑工作的输入参考时钟
	RST_n	1	In	复位信号，低电平有效
	Address	2	In	地址端口，用于寻址 SPI IP 内的寄存器
	Data_to_IP	32	In	向 SPI IP 写数据的端口
	Write	1	In	Data_to_IP 端口的写使能信号
	Data_from_IP	32	Out	从 SPI IP 读数据的端口
	Read	1	In	Data_from_IP 端口的读使能信号
	TOE	1	Out	Transmit-overflow error, 发送数据覆盖错误, 前一次被写入 TxData 寄存器里的数据没有被发送时, 如果又有新的数据被写入 TxData 寄存器, 则该信号变为高电平, 在这种情况下, 新数据被忽略。读 TxData 寄存器可清 0 该信号
	ROE	1	Out	Receive-overflow error, 接受数据覆盖错误, 当 RxData 寄存器里的数据没有被读取时, 如果 RxData 寄存器又接收到新的数据, 则该信号变为高电平, 在这种情况下, 新数据被忽略。读 RxData 寄存器可清 0 该信号



	TRDY	1	Out	Transmitter Ready, 当 TxData 寄存器可以接收新数据时, 该信号为高电平
	RRDY	1	Out	Receiver Ready, 当 RxData 寄存器的数据被更新后, 该信号变为高电平。读 RxData 寄存器可清 0 该信号
面向 SPI Slave 的端口	SCLK	1	Out	串行时钟信号
	MOSI	1	Out	主设备输出/从设备输入信号
	MISO	1	In	主设备输入/从设备输出信号
	SS_n	1~32	Out	从设备片选信号, 低电平有效

## 2.2.2 SPI Slave 端口

表 2-4 当 Type = Slave 时, SPI IP 的端口

	端口名	位宽	方向	说明
面向片内用户逻辑的端口	Sys_Clk <sup>(1)</sup>	1	In	SPI IP 核内部逻辑工作的输入参考时钟
	RST_n	1	In	复位信号, 低电平有效
	Address	2	In	地址端口, 用于寻址 SPI IP 内的寄存器
	Data_to_IP	32	In	向 SPI IP 写数据的端口
	Write	1	In	Data_to_IP 端口的写使能信号
	Data_from_IP	32	Out	从 SPI IP 读数据的端口
	Read	1	In	Data_from_IP 端口的读使能信号
	TOE	1	Out	Transmit-overflow error, 发送数据覆盖错误, 前一次被写入 TxData 寄存器里的数据没有被发送时, 如果又有新的数据被写入 TxData 寄存器, 则该信号变为高电平, 在这种情况下, 新数据被忽略。读 TxData 寄存器可清 0 该信号
	ROE	1	Out	Receive-overflow error, 接受数据覆盖错误, 当 RxData 寄存器里的数据没有被读取时, 如果 RxData 寄存器又接收到新的数据, 则该信号变为高电平, 在这种情况下, 新数据被忽略。读 RxData 寄存器可清 0 该信号
	TRDY	1	Out	Transmitter Ready, 当 TxData 寄存器可以接收新数据时, 该信号为高



面向 SPI Master 的端口				电平
	RRDY	1	Out	Receiver Ready, 当 RxData 寄存器的数据被更新后, 该信号变为高电平。读 RxData 寄存器可清 0 该信号
	RPRDY	1	Out	Receiver Partial Ready, 当 SPI Slave 接收到的数据位数达到 RxPartial Threshold 后, 该信号变为高电平。读 RxData 寄存器可清 0 该信号
面向 SPI Master 的端口	SCLK <sup>(1)</sup>	1	In	串行时钟信号
	MOSI	1	In	主设备输出/从设备输入信号
	MISO	1	Out	主设备输入/从设备输出信号
	SS_n	1	In	从设备片选信号, 低电平有效

(1) SPI Slave 的 Sys\_Clk 频率应大于等于其 SCLK 频率的 2 倍。

2.3 用户可存取寄存器

生成 SPI IP 后, 用户逻辑可通过 Address, Data\_to\_IP, Write, Data\_from\_IP, Read 等端口来读写用户可存取寄存器 (User-Accessible Register, 缩写为 UAR)。

2.3.1 SPI Master 的 UAR

表 2-5 SPI Master 的 UAR 映射

内部地址	寄存器名	读写类型	寄存器内容						
			31~6	5	4	3	2	1	0
0	TxData	读/写	需要发送的数据						
1	RxData	只读	接收到的数据						
2	Control	读/写					CPOL	CPHA	Shift_Dir
3	Slave_Select	读/写	从设备片选掩码						

TxData 寄存器

发送数据寄存器。用户的逻辑设计可在 TRDY 信号为高电平期间写 TxData 寄存器; 当 TRDY 信号为低电平时, 写入 TxData 寄存器的值被忽略, TOE 信号变为高电平。用户的逻辑设计可在任意时刻读取 TxData



寄存器。读 TxData 寄存器操作可将 TOE 信号清 0。在 Master 模式下，写 TxData 寄存器成功会发起一次 SPI 传输。

## RxData 寄存器

接收数据寄存器。用户的逻辑设计可在任意时刻读取 RxData 寄存器。当 RxData 寄存器里的数据没有被读取时，即 RRDY 为高电平时，如果 SPI IP 核又接收到新的数据，则 ROE 信号变为高电平，在这种情况下，新接收到的数据被忽略。读 RxData 寄存器操作可将 RRDY 和 ROE 信号清 0。

## Control 寄存器

控制寄存器。用户的逻辑设计可在 TRDY 信号为高电平期间写 Control 寄存器；当 TRDY 信号为低电平时，写入 Control 寄存器的值被忽略。用户的逻辑设计可在任意时刻读取 Control 寄存器。

表 2-6 Control 寄存器的位定义

#	名称	默认值	说明
0	Shift_Dir	1	移位方向。当 Shift_Dir = 0 时，LSB 先传；当 Shift_Dir = 1 时，MSB 先传
1	CPHA	1	时钟相位
2	CPOL	1	时钟极性

## Slave\_Select 寄存器

从设备选择寄存器。用户的逻辑设计可在 TRDY 信号为高电平期间写 Slave\_Select 寄存器；当 TRDY 信号为低电平时，写入 Slave\_Select 寄存器的值被忽略。用户的逻辑设计可在任意时刻读取 Slave\_Select 寄存器。

表 2-7 Slave\_Select 寄存器的位定义

#	名称	默认值	说明
31~0	Slave_Select	0xFFFFFFFF E	SPI 主设备根据该寄存器中被置 0 的位来选择对应的从设备

### 2.3.2 SPI Slave 的 UAR



表 2-8 SPI Slave 的 UAR 映射

内部地址	寄存器名	读写类型	寄存器内容						
			31~6	5	4	3	2	1	0
0	TxData	读/写	需要发送的数据						
1	RxData	只读	接收到的数据						
2	Control	读/写				SPI_EN			

## TxData 寄存器

发送数据寄存器。用户的逻辑设计可在 SS\_n 信号为高电平期间多次写 TxData 寄存器。当 SS\_n 信号为低电平时，如果在上一次 SPI 传输结束后没有对 TxData 执行过写操作，则 TxData 依然可被写入新数据；如果在上一次 SPI 传输结束后已对 TxData 执行过写操作，则 TRDY 信号为低电平，如果此时再对 TxData 进行写操作，则新写入的值被忽略，TOE 信号变为高电平。读 TxData 寄存器操作可将 TOE 信号清 0。

## RxData 寄存器

接收数据寄存器。用户的逻辑设计可在任意时刻读取 RxData 寄存器。当 RxData 寄存器里的数据没有被读取时，即 RRDY 或 RPRDY 为高电平时，如果 SPI IP 核又接收到新的数据，则 ROE 信号变为高电平，在这种情况下，新接收到的数据被忽略。读 RxData 寄存器操作可将 RRDY，RPRDY 和 ROE 信号清 0。

## Control 寄存器

控制寄存器。用户的逻辑设计可在任意时刻写或读 Control 寄存器。

表 2-9 Control 寄存器的位定义

#	Name	默认值	说明
3	SPI_EN	0	SPI Slave IP 核使能。当 SPI_EN = 0 时，Slave SPI IP 核不工作，片外输入信号被忽略，片外输出的引脚为高阻态；当 SPI_EN = 1 时，SPI Slave IP 核使能

## 2.4 时序

### 2.4.1 写 UAR

以写 Control 寄存器为例。

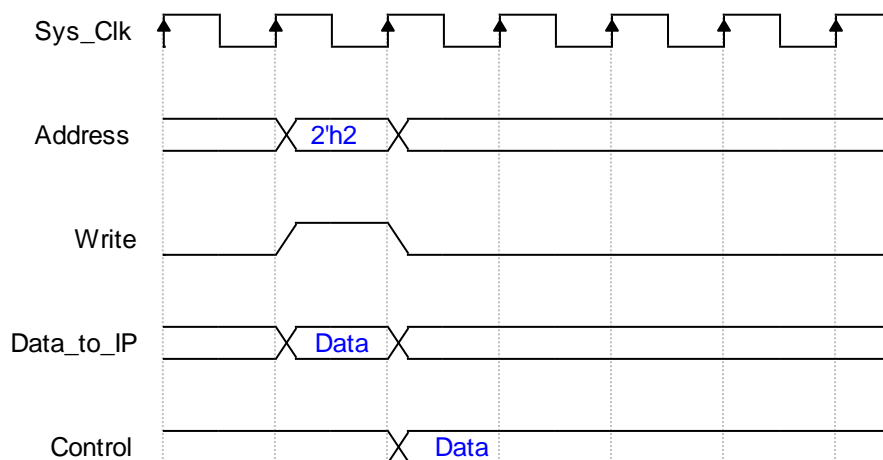


图 2-4 写 Control 寄存器时序

## 2.4.2 读 UAR

以读 Control 寄存器为例，其中 User's Reg 为用户逻辑中的寄存器。

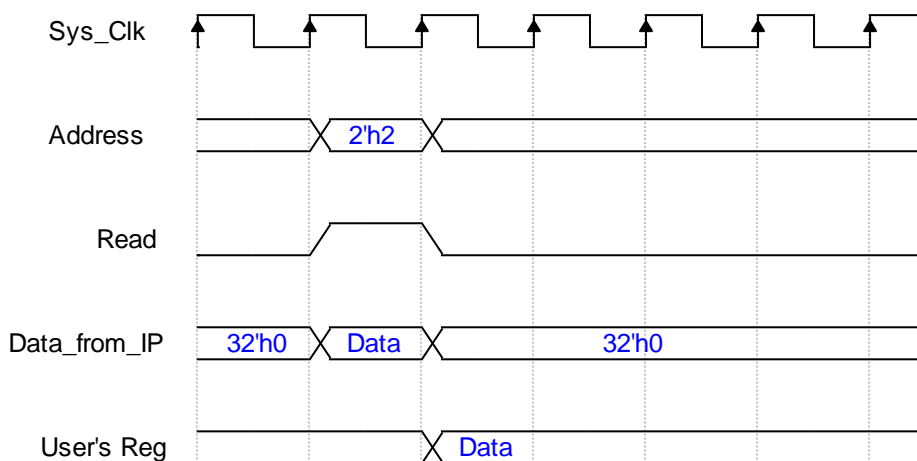


图 2-5 读 Control 寄存器时序

## 2.4.3 SPI Master IP 发起 SPI 传输

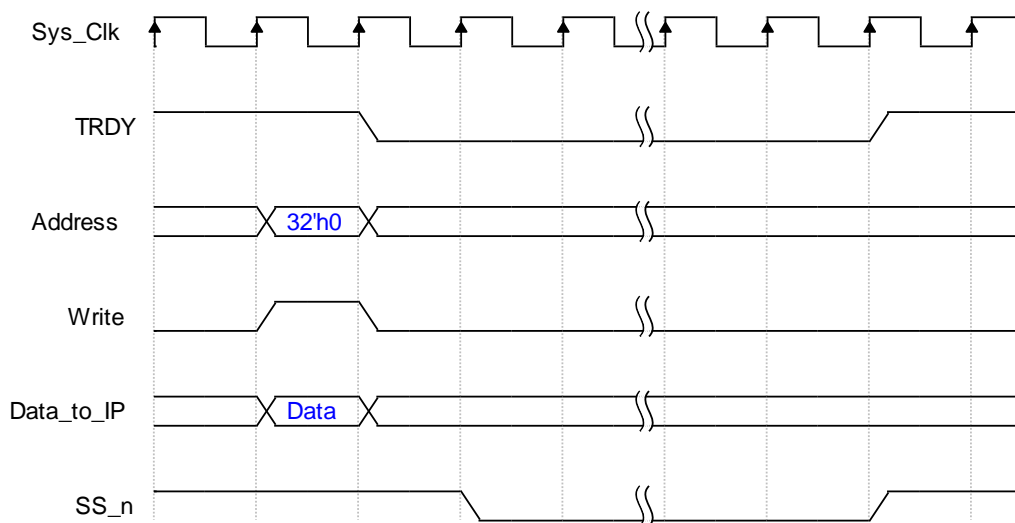


图 2-6 SPI Master IP 发起 SPI 传输的时序

#### 2. 4. 4 TRDY 和 TOE

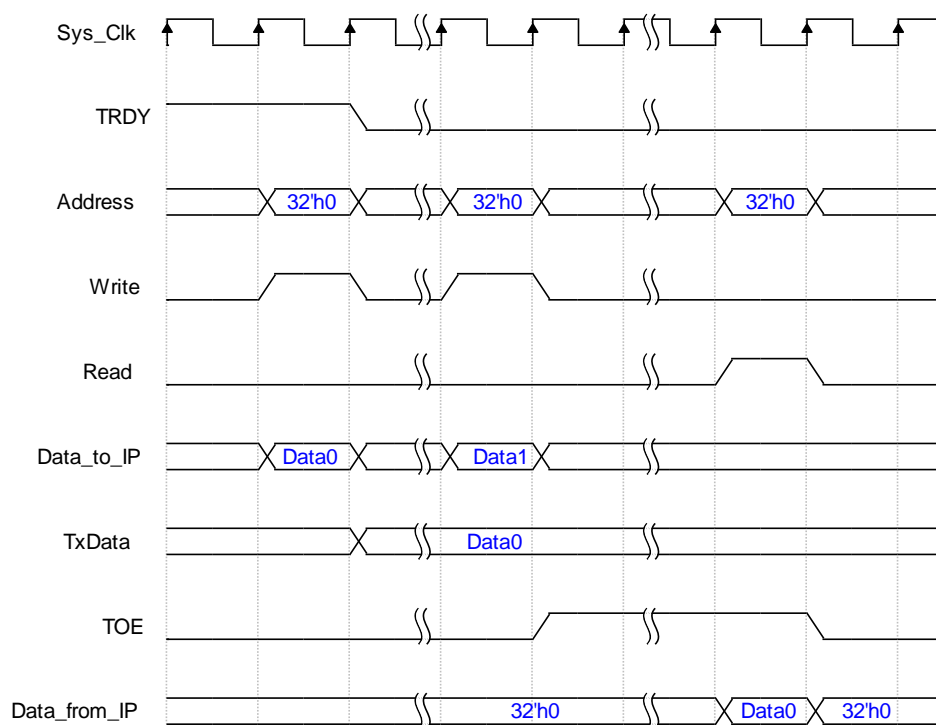


图 2-7 TRDY 和 TOE 信号时序

#### 2. 4. 5 RPRDY, RRDY 和 ROE

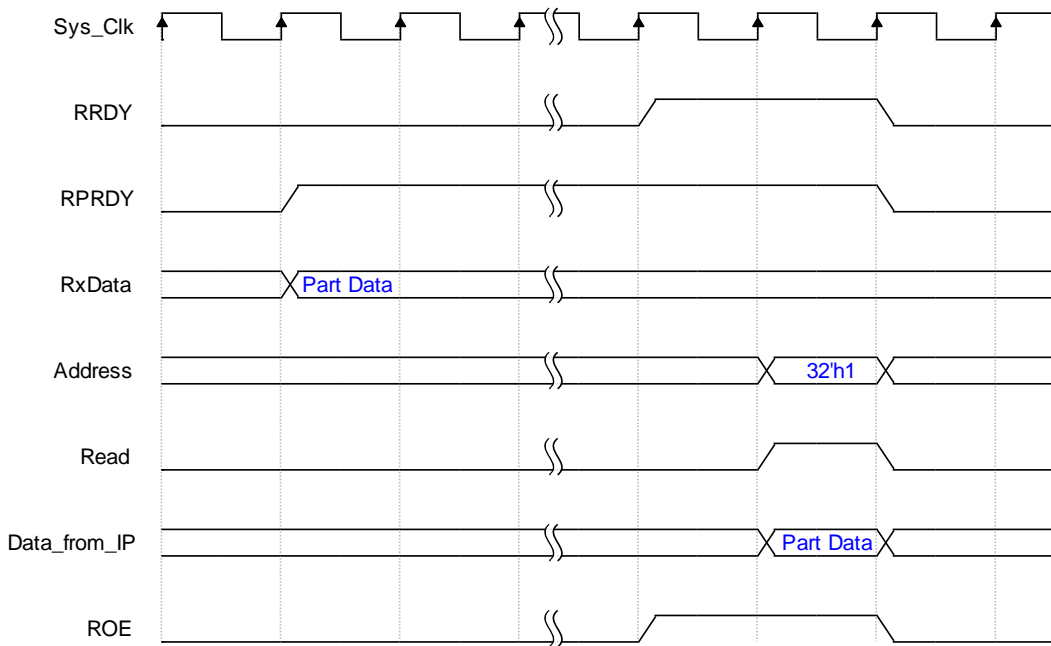


图 2-8 RPRDY, RRDY 和 ROE 信号时序

## 2.4.6 “部分接收阈值” 功能

在一些 SPI 的上层协议中，规定了 SPI 通信数据的格式。例如图 2-9 中，SPI Master 发送的前若干位为地址位和读写标志位，后若干位为数据位。安路 SPI Slave IP 的“部分接收阈值”功能专为兼容这种上层协议而设计，用户通过在 GUI 中设置部分接收阈值 (RxPartial threshold)，可令 SPI Slave IP 接收到的数据位数达到该阈值后立即推送到 RxData 寄存器，并将 RPRDY 置 1。用户逻辑在读取 RxData 寄存器后可进行寻址等操作，并将需要发送的数据写入 TxData 寄存器。

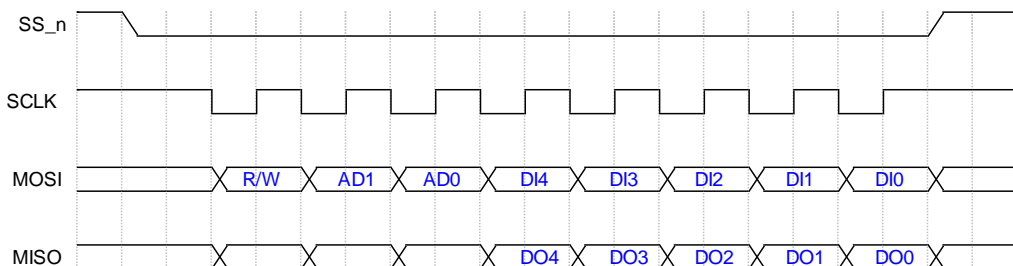


图 2-9 一些 SPI 上层协议中的数据格式

在上述场景中使用“部分接收阈值”功能时应注意，需要发送的数据应在 SCLK 的发送边沿到达之前准备就绪，这就要求 Sys\_Clk 频率远高于 SCLK 频率。如果用户逻辑对 RPRDY 信号的响应时序如图 2-10 所示，则 Sys\_Clk 频率应大于等于 SCLK 频率的 18 倍。



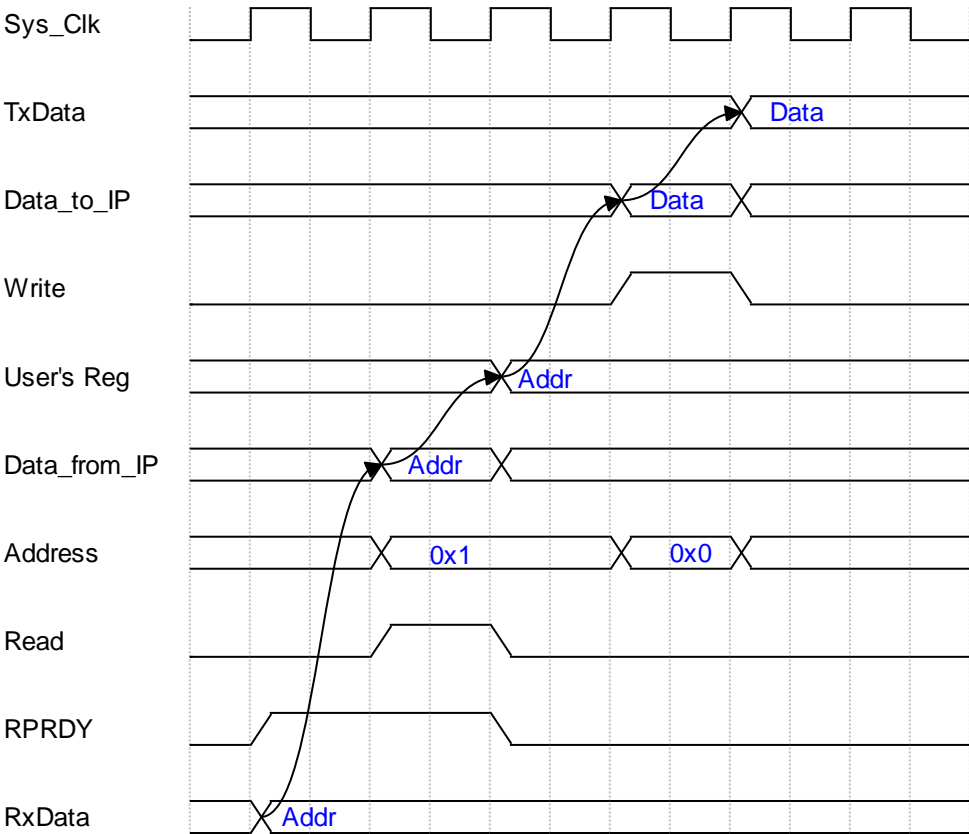


图 2.10 用户逻辑对 RPRDY 信号的响应时序

2.5 资源占用和时序性能

用户可根据以下案例中的数据来评估 SPI IP 核的资源占用情况和时序性能。该案例采用 TD4. 6. 4 进行编译，器件型号为安路 EAGLE 系列 EG4X20BG256。

表 2-10 SPI IP 的资源占用情况和时序性能

Type	CPOL	CPHA	Shift_Dir	LUT 占用	Reg 占用	Sys_Clk 最高频率
Master	1	1	1	492	257	200MHz
Slave	1	1	1	272	150	200MHz



### 3 版本信息

日期	版本	修订记录
2021/07/05	1.0	初版制定

**版权所有©2021 上海安路信息科技股份有限公司**

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、翻译本文档内容的部分或全部，并不得以任何形式传播。

### 免责声明

本文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。除安路科技在其产品的销售条款和条件中声明的责任之外，安路科技概不承担任何法律或非法律责任。安路科技对安路科技产品的销售和/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。安路科技对文档中包含的文字、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，安路科技保留修改文档中任何内容的权利，恕不另行通知。安路科技不承诺对这些文档进行适时的更新。