



SPI 主机端 数据表 SPIM V 1.1

Copyright © 2005-2010 Cypress Semiconductor Corporation. All Rights Reserved.

资源	API 存储器 (字节)		引脚
	闪存	RAM	
CY7C639/638/633/602/601xx, CYRF69xx3			
MOSI/MISO/SCLK	38	0	3 - 4
SDIO/SCK	38	0	2 - 3

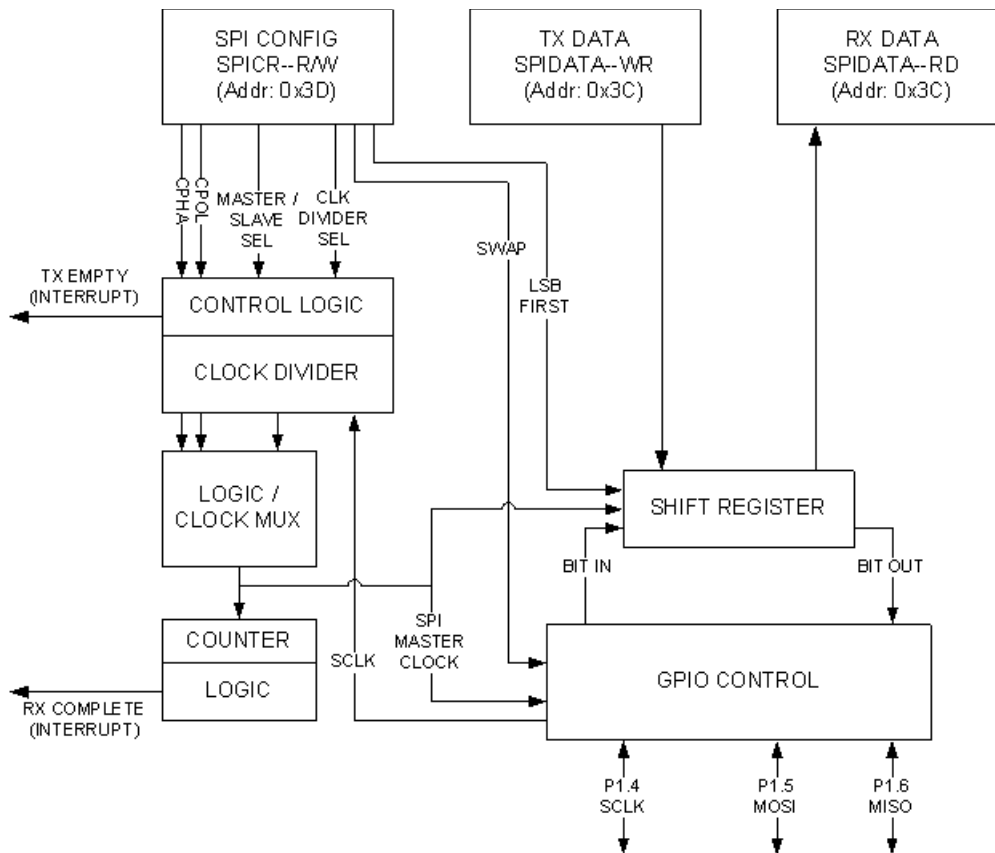
特性与概述

- 支持串行外设互连 (SPI) 主机端协议
- 支持 SPI 时钟模式 0、1、2 和 3
- SPI 完成为条件的可编程中断
- 可独立选择 SPI 从器件

SPIM 用户模块是一种串行外设互连主机端。它执行全双工同步 8-bit 数据传输。可指定 SCLK 相位、SCLK 极性和最低有效位优先，以适应大多数 SPI 时钟模式。通过用户提供的软件进行控制，您可以配置从器件选择信号以控制一个或更多 SPI 从器件。

SPI 用户模块还支持两线串行器件，例如采用 SDIO/SCK 接口的光电鼠标传感器。

Figure 1. SPIM 框图



参数和资源

SPI 引脚

SPI 引脚参数可用于选择 SPI 模块使用的引脚。该参数允许两线 (SDIO/SCK) 或三线 (MOSI/MISO/SCLK) 操作。如果所选器件支持 5V 或 3.3V 电压下的操作，则还可提供电压选择参数。

通过 SPI 引脚 参数可对 P14CR、P15CR 和 P16CR 配置寄存器进行编程。

SPI 引脚 选择	电压选项	
	CY7C639xx CY7C638xx CY7C633xx	CY7C601xx CY7C602xx
MOSI (P1.5) / MISO (P1.6) / SCLK (P1.4)	5V 或 3.3V	不适用
SDIO (P1.5) / SCK (P1.4)	5V 或 3.3V	不适用

CY7C601xx 和 CY7C602xx 器件对 SPI 引脚不提供双电压支持。

BitOrder

BitOrder 参数可用于为 SPI 数据流选择位顺序。相应的选项是 LSBFirst 和 MSBFirst。

通过 BitOrder 参数可对 SPI 配置寄存器 (SPICR) 的“最低有效位优先”字段进行编程。

CPOL

CPOL 参数可用于为 SPI 时钟选择空闲状态。

通过 CPOL 参数可对 SPI 配置寄存器 (SPICR) 的 CPOL 字段进行编程。

CPHA

CPHA 参数可用于选择 SPI 数据采样所在的 SPI 时钟相位。

通过 CPHA 参数可对 SPI 配置寄存器 (SPICR) 的 CPHA 字段进行编程。

ClockDivider

ClockDivider 参数可用于选择 SPI 时钟的 ClockDivider。ClockDivider 应用于 CPUCLK 来生成 SPI 时钟。

ClockDivider	SCLK 选择	两种不同 CPUCLK 频率下的 SCLK 频率	
		12 MHz	24 MHz
6	00	2 MHz	4 MHz
12	01	1 MHz	2 MHz
48	10	250 kHz	500 kHz
96	11	125 kHz	250 kHz

通过 ClockDivider 参数可对 SPI 配置寄存器 (SPICR) 的“SCLK 选择”字段进行编程。

应用程序编程接口

应用程序编程接口 (API) 例程作为用户模块的一部分提供, 使设计人员能够在较高的层级处理模块。本节具体说明了每个函数对应的接口以及 include 文件所提供的相关常量。

SPIM_Start

说明:

启动 SPIM 用户模块。提供此函数是为了保持 API 一致性。

C 语言原型:

```
void SPIM_Start(void)
```

汇编语言:

```
lcall SPIM_Start
```

参数:

无

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_Stop**说明:**

禁用 SPIM 模块。提供此函数是为了保持 API 一致性。

C 语言原型:

```
void SPIM_Stop(void)
```

汇编语言:

```
lcall SPIM_Stop
```

参数:

无

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_EnableInt**说明:**

启用 SPIM TX 缓冲区已满和 RX 完成中断。

注: 您可以添加自定义 ISR 中断处理代码在项目 lib 目录下的 SPIMINT.asm 文件中。

C 语言原型:

```
void SPIM_EnableInt(void)
```

汇编语言:

```
lcall SPIM_EnableInt
```

参数:

无

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_DisableInt**说明:**

禁用 SPIM TX 缓冲区已满和 RX 完成中断。

C 语言原型:

```
void SPIM_DisableInt(void)
```

汇编语言:

```
lcall SPIM_DisableInt
```

参数:

无

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_SetMOSI**说明:**

将输出引脚设置为 “主出从入”。

C 语言原型:

```
void SPIM_SetMOSI(BYTE bPin)
```

汇编语言:

```
mov A, bPin  
lcall SPIM_SetMOSI
```

参数:

BYTE bPin: 0x00--SPIM_MOSI_P15, 0x01--SPIM_MOSI_P16. 该参数将在累加器中传递。

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_SetMISO**说明:**

将输出引脚设置为 “主入从出”。

C 语言原型:

```
void SPIM_SetMISO(BYTE bPin)
```

汇编语言:

```
mov A, bPin  
lcall SPIM_SetMISO
```

参数:

BYTE bPin: 0x00--SPIM_MISO_P16, 0x01--SPIM_MISO_P15. 该参数将在累加器中传递。

返回值:

无

副作用:

此函数可能会修改 A 和 X 寄存器。

SPIM_bIO

说明:

传输单个数据字节并返回从从器件中接收到的数据字节。

C 语言原型:

```
BYTE SPIM_bIO(BYTE bTxData)
```

汇编语言:

```
mov    A, bRxData  
lcall  SPIM_bIO
```

参数:

bTxData - 要传输的数据

返回值:

从从器件 SPI 中接收到的数据字节将在累加器中返回。

副作用:

调用此函数之后将会清除状态位。此函数可能会修改 A 和 X 寄存器。

示例固件源代码

在 C 语言中，使用 Start API 开始操作，并在完成时调用 Stop API。

```
#include "SPIM.h"
```

```
SPIM_Start();  
... // (application processing)  
SPIM_Stop();
```

以下 C 语言代码片段显示了使用 MOSI/MISO/SCLK 选择从 SPI EEPROM 中读取数据的过程:

```
SPIM_Start();  
SPIM_SetMOSI(SPIM_MOSI_P16); // Set MOSI to P1.6/MISO P1.5  
EEPROM_Reset(); // Reset the EEPROM (user provided function)  
SPI_SELECT(); // Assert slave select (user provided function)  
SPIM_bIO(EEP_READ); // Send a read command  
SPIM_bIO(0x00); // Address 0:0  
SPIM_bIO(0x00);  
  
for (i = 0; i < 32; ++i)  
{  
    aRD[i] = SPIM_bIO(0); // Read some data  
}  
SPI_UNSELECT(); // Deassert the slave select (user provided function)
```

以下 C 语言代码片段显示了使用 SDIO/SCK 选择从 SDIO 光学传感器中读取寄存器 0x01 的过程：

```
SPIM_Start();  
SPIM_SetMOSI(SPIM_MOSI_P15); // Set MOSI to P1.5  
SPIM_bIO((0x01)); // Write the register number  
timer_delay_100_usec(); // delay (user provided function)  
SPIM_SetMISO(SPIM_MISO_P15); // Reverse the SDIO line  
data = SPIM_bIO(0); // Read the register data
```

Copyright © 2005-2010 Cypress Semiconductor Corporation. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC Designer™ and Programmable System-on-Chip™ are trademarks and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.