第 35 章 串行外设接口 （SPI）（第二部分）

目录

本章包括下列主题：

35.1 简介 35-2
35.2 SPI 寄存器 35-3
35.3 工作模式 35-8
35.4 主模式时钟频率 35-21
35.5 节能模式下的工作 35-22
35.6 与 SPI1 模块相关的特殊功能寄存器 35-23
35.7 相关应用笔记 35-24
35.8 版本历史 35-25

35.1 简介

串行外设接口（Serial Peripheral Interface，SPI）模块是用于同其他外设或单片机器件进行通信的同步串行接口。这些外设可以是串行 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等。SPI 模块与 Motorola 的 SPI 和 SIOP 接口兼容。

dsPIC30F SMPS 和数字电源转换器件系列的每种器件都提供一个 SPI 模块，称为 SPI1。

SPI1 串行接口由 4 个引脚组成：

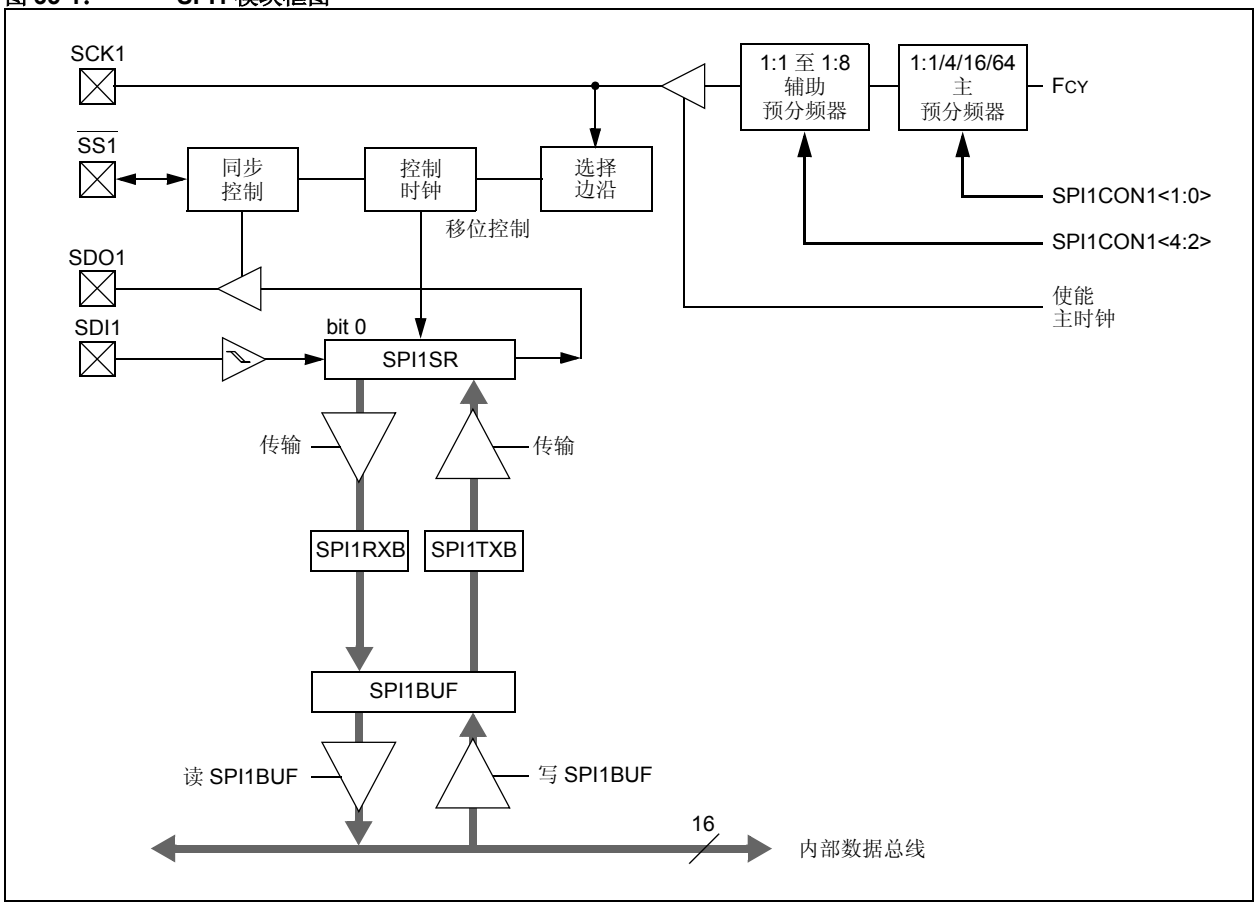
- SDI1：串行数据输入
- SDO1：串行数据输出
- SCK1：移位时钟输入或输出
- $\overline{SS1}$：低电平有效从选择或帧同步 I/O 脉冲

可将 SPI1 模块配置为使用 2、 3 或 4 个引脚进行工作。在 3 引脚模式下，不使用 $\overline{SS1}$ 引脚。在 2 引脚模式下，不使用 SDO1 和 $\overline{SS1}$ 引脚。

SPI1 模块的框图如图 35-1 所示。

注： dsPIC30F SMPS 和数字电源转换器件系列中的某些器件不包含 $\overline{SS1}$ 引脚，因而不支持帧 SPI 和从选择功能。关于可用的 SPI 引脚，请参考特定的器件数据手册。

图 35-1： SPI1 模块框图



© 2008 Microchip Technology Inc.

35.2 SPI 寄存器

SPI 模块由一个用于移入和移出数据的 16 位移位寄存器 SPI1SR 和一个缓冲寄存器 SPI1BUF 组成。控制寄存器 SPI1CON1 和 SPI1CON2 用于配置模块。另外，状态寄存器 SPI1STAT 用来表明各种状态条件。

35.2.1 状态和控制寄存器

SPI1STAT、SPI1CON1 和 SPI1CON2 寄存器提供了用于控制模块操作的接口。寄存器 35-1、寄存器 35-2 和寄存器 35-3 中给出了这些寄存器的详细说明。

- SPI1STAT：SPI1 状态和控制寄存器

  此状态寄存器用来表明各种状态条件，如接收溢出、发送缓冲区满以及接收缓冲区满。此寄存器还可用于指定空闲模式下模块的工作，且包含一个可使能和禁止模块的位。
- SPI1CON1：SPI1 控制寄存器 1

  此控制寄存器指定时钟预分频比、主 / 从模式、字 / 字节通信、时钟极性和时钟 / 数据引脚操作。
- SPI1CON2：SPI1 控制寄存器 2

  此控制寄存器使能或禁止帧 SPI 操作。此寄存器还指定帧同步脉冲的方向、极性和边沿选择。

35.2.2 SPI1BUF 寄存器

SPI1BUF 是 SPI1 的数据接收 / 发送寄存器。SPI1BUF 寄存器实际上由两个独立的寄存器组成：发送缓冲寄存器 SPI1TXB 和接收缓冲寄存器 SPI1RXB。这两个单向的 16 位寄存器共用 SPI1BUF 的 SFR 地址。如果用户应用程序将要发送的数据写入 SPI1BUF 地址，在内部该数据会被写入 SPI1TXB 寄存器。与此类似，当用户应用程序从 SPI1BUF 读取接收到的数据时，在内部会从 SPI1RXB 寄存器读取该数据。

这种方法对接收和发送操作进行了双重缓冲，允许在后台连续传输数据。发送和接收可同时进行。

此外，还有一个 16 位的移位寄存器 SPI1SR，这个寄存器未被映射到存储空间。它用于将数据移入和移出 SPI 端口。

寄存器 35-1： SPI1STAT：SPI1 状态和控制寄存器

高字节：							
R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	—	SPISIDL	—	—	—	—	—
bit 15							bit 8

低字节：							
U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

bit 15 **SPIEN**：SPI1 使能位
1 = 使能模块并将 SCK1、 SDO1、 SDI1 和 $\overline{SS1}$ 引脚配置为串行端口引脚
0 = 禁止模块

bit 14 **未实现**：读为 0

bit 13 **SPISIDL**：空闲模式停止位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作

bit 12-7 **未实现**：读为 0

bit 6 **SPIROV**：接收溢出标志位
1 = 一个新字节 / 字已被完全接收并丢弃。用户应用程序没有读取先前保存在 SPI1BUF 寄存器中的数据。
0 = 未发生溢出

bit 5-2 **未实现**：读为 0

bit 1 **SPITBF**：SPI1 发送缓冲区满状态位
1 = 发送尚未开始， SPI1TXB 为满
0 = 发送已开始， SPI1TXB 为空
当 CPU 写 SPI1BUF 存储单元并装入 SPI1TXB 时，该位由硬件自动置 1。
当 SPI1 模块将数据从 SPI1TXB 传输到 SPI1SR 时，该位由硬件自动清零。

bit 0 **SPIRBF**：SPI1 接收缓冲区满状态位
1 = 接收完成， SPI1RXB 为满
0 = 接收未完成， SPI1RXB 为空
当 SPI1 模块将数据从 SPI1SR 传输到 SPI1RXB 时，该位由硬件自动置 1。
当内核通过读取 SPI1BUF 存储单元读 SPI1RXB 时，该位由硬件自动清零。

图注：	C = 可清零位		
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

© 2008 Microchip Technology Inc.

寄存器 35-2： SPI1CON1：SPI1 控制寄存器 1

高字节：							
U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE(1)
bit 15							bit 8

低字节：							
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>	
bit 7							bit 0

bit 15-13 **未实现：**读为 0

bit 12 **DISSCK：**禁止 SCK1 引脚的位 （仅限 SPI 主模式）
1 = 禁止内部 SPI 时钟，引脚用作 I/O
0 = 使能内部 SPI 时钟

bit 11 **DISSDO：**禁止 SDO1 引脚的位
1 = 模块不使用 SDO1 引脚，引脚用作 I/O
0 = SDO1 引脚由模块控制

bit 10 **MODE16：**字 / 字节通信选择位
1 = 采用字宽 （16 位）通信
0 = 采用字节宽 （8 位）通信

bit 9 **SMP：**SPI1 数据输入采样阶段位
主模式：
1 = 在数据输出时间的末端采样输入数据
0 = 在数据输出时间的中间采样输入数据
从模式：
当 SPI1 用于从模式时，必须将 SMP 清零

bit 8 **CKE：**SPI1 时钟边沿选择位 (1)
1 = 串行输出数据在时钟从工作状态转变为空闲状态时变化 （见 bit 6）
0 = 串行输出数据在时钟从空闲状态转变为工作状态时变化 （见 bit 6）

bit 7 **SSEN：**从选择使能位 （从模式）
1 = SS1 引脚用于从模式
0 = 模块不使用 SS1 引脚，此引脚由端口功能控制

bit 6 **CKP：**时钟极性选择位
1 = 时钟信号的空闲状态为高电平；工作状态为低电平
0 = 时钟信号的空闲状态为低电平；工作状态为高电平

bit 5 **MSTEN：**主模式使能位
1 = 主模式
0 = 从模式

bit 4-2 **SPRE<2:0>：**辅助预分频比位 （主模式）
111 = 辅助预分频比 1:1
110 = 辅助预分频比 2:1
•
•
•
000 = 辅助预分频比 8:1

寄存器 35-2： SPI1CON1：SPI1 控制寄存器 1 （续）

bit 1-0 **PPRE<1:0>：**主预分频比位 （主模式）
11 = 主预分频比 1:1
10 = 主预分频比 4:1
01 = 主预分频比 16:1
00 = 主预分频比 64:1

注 1： 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 （FRMEN = 1）下，用户应用程序应将该位编程为 0。

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

© 2008 Microchip Technology Inc.

寄存器 35-3： SPI1CON2：SPI1 控制寄存器 2

高字节：							
R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	—	—	—	—	—
bit 15							bit 8

低字节：							
U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	FRMDLY	—
bit 7							bit 0

bit 15 **FRMEN**：帧 SPI1 支持位
1 = 使能帧 SPI1 支持 （$\overline{SS1}$ 引脚用作帧同步脉冲输入 / 输出）
0 = 禁止帧 SPI1 支持

bit 14 **SPIFSD**：帧同步脉冲方向控制位
1 = 帧同步脉冲输入 （从器件）
0 = 帧同步脉冲输出 （主器件）

bit 13 **FRMPOL**：帧同步脉冲极性位
1 = 帧同步脉冲为高电平有效
0 = 帧同步脉冲为低电平有效

bit 12-2 **未实现**：读为 0

bit 1 **FRMDLY**：帧同步脉冲边沿选择位
1 = 帧同步脉冲与第一个位时钟一致
0 = 帧同步脉冲超前第一个位时钟

bit 0 **未实现**：用户应用程序不能将该位设置为 1

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

35.3 工作模式

以下各小节讨论了 SPI 模块灵活的工作模式：

- 8 位和 16 位数据发送 / 接收
- 主模式和从模式
- 帧 SPI 模式
- SPI1 只接收操作
- SPI1 错误处理

35.3.1 8 位和 16 位数据发送 / 接收

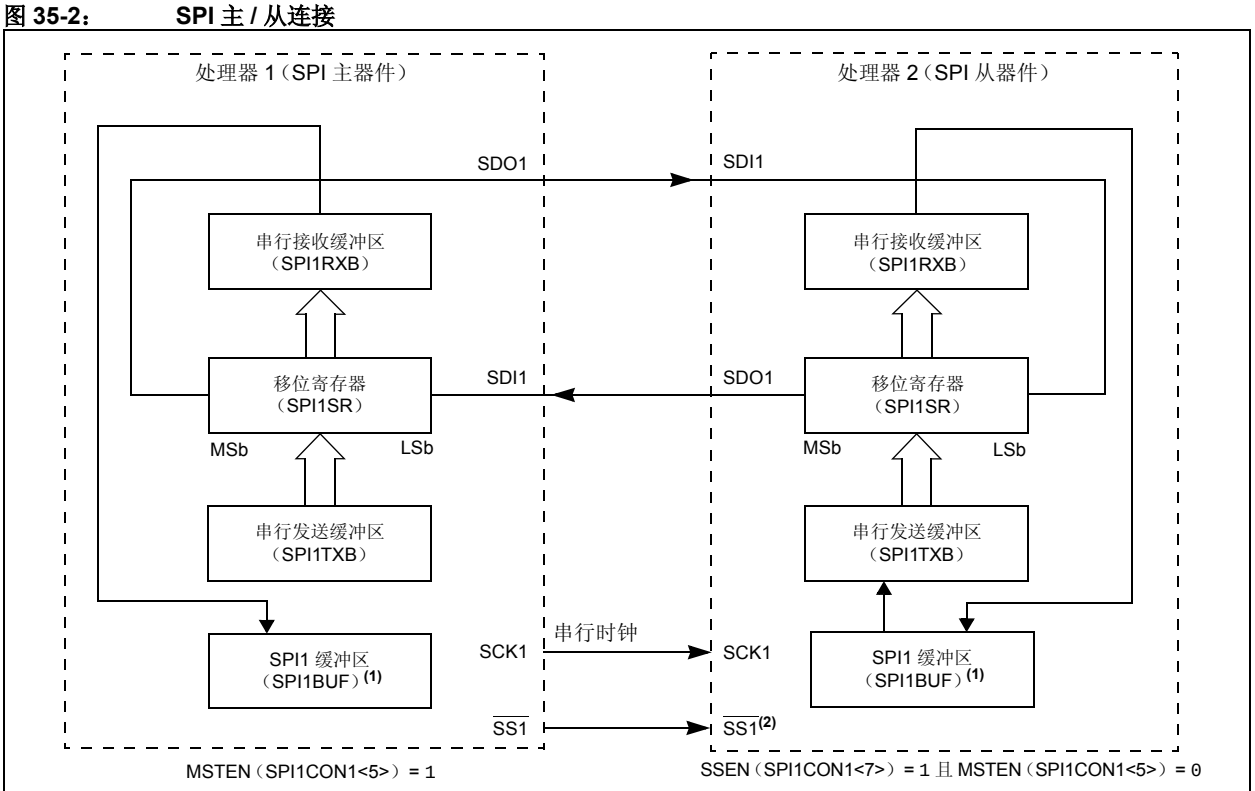
MODE16 控制位（SPI1CON1<10>）允许模块在 8 位或 16 位模式下进行通信。除了接收和发送的位数外，两种模式的功能是相同的。此外，在阅读本文时应注意以下各项：

- 当 MODE16 位的值发生变化时，模块会被复位。因此，在正常工作期间不应该改变该位。
- 8 位工作模式下数据是从 SPI1SR 寄存器的 bit 7 发送出去的，而在 16 位工作模式下，则是从 SPI1SR 的 bit 15 发送出去的。在两种模式下，接收时数据都会移入 SPI1SR 的 bit 0。
- 在 8 位模式下发送或者接收数据时，需要 SCK1 引脚上出现 8 个时钟脉冲，而在 16 位模式下则需要 16 个时钟脉冲。

35.3.2 主模式和从模式

数据的收发可看作在一个模块的移位寄存器的最高位（MSb）和另一个模块的移位寄存器的最低位（LSb）之间采取了一个直接通路，然后将数据移到相应的发送缓冲区或接收缓冲区。配置为主模块的模块根据需要为从器件提供串行时钟信号和同步信号。图 35-2 给出了主从模块的连接。

图 35-2： SPI 主 / 从连接



注 1： 用户应用程序必须向 SPI1BUF 写入发送数据，并从 SPI1BUF 读取接收到的数据。SPI1TXB 和 SPI1RXB 寄存器被存储器映射到 SPI1BUF。

2： 可选择在从模式下使用 SS1 引脚。

© 2008 Microchip Technology Inc.

35.3.2.1 主模式

在主模式下，系统时钟被预分频，然后被用作串行时钟。预分频基于 PPRE<1:0>（SPI1CON1<1:0>）和 SPRE<2:0>（SPI1CON1<4:2>）位的设置。串行时钟通过 SCK1 引脚输出到从器件。仅当有待发送数据时才会产生时钟脉冲。如需了解更多信息，请参见**第 35.4 节“主模式时钟频率”**。 CKP 和 CKE 位决定数据传输发生在时钟的哪个边沿。

将待发送数据写入 SPI1BUF 寄存器或从该寄存器读取已接收数据。

下面描述了主模式下 SPI1 模块的工作原理：

1. 一旦模块被设置为主工作模式并使能，待发送数据就会被写入 SPI1BUF 寄存器。SPITBF（SPI1STAT<1>）位被置 1。
2. SPI1TXB 的内容被移到移位寄存器 SPI1SR，且模块将 SPITBF 位清零。
3. 一组 8/16 个时钟脉冲将 8/16 位发送数据从 SPI1SR 移出到 SDO1 引脚，同时将 SDI1 引脚的数据移入 SPI1SR。

   当传输结束时，会发生以下事件：

   a) 中断标志位 SPI1IF 置 1。通过将中断允许位 SPI1IE 置 1 可允许 SPI1 中断。 SPI1IF 标志不会被硬件自动清零。

   b) 当正在进行的发送和接收操作结束时， SPI1SR 寄存器的内容被移到 SPI1RXB 寄存器。

   c) SPIRBF （SPI1STAT<0>）位被模块置 1，表明接收缓冲区满。

   d) 一旦用户应用程序读取了 SPI1BUF 寄存器，硬件就会将 SPIRBF 位清零。
4. 当 SPI1 模块需要从 SPI1SR 传输数据到 SPI1RXB 时，如果 SPIRBF 位置 1 （接收缓冲区满），模块会将 SPIROV （SPI1STAT<6>）位置 1，表明产生了溢出条件。
5. 只要 SPITBF（SPI1STAT<1>）为零，用户应用程序就可以在任何时候将待发送数据写入 SPI1BUF。写入操作可以与 SPI1SR 移出先前写入的数据同时发生，因此可以允许连续发送。

注：	用户应用程序不能直接写 SPI1SR 寄存器。对 SPI1SR 寄存器的所有写入操作都是通过 SPI1BUF 寄存器进行的。

要将 SPI1 模块设置为工作在主模式，执行下列步骤：

1. 如果使用中断：
   a) 清零 IFS0 寄存器中的 SPI1IF 位。
   b) 置 1 IEC0 寄存器中的 SPI1IE 位。
   c) 写 IPC2 寄存器中的 SPI1IP 位来设置中断优先级。
2. 将需要的设置写入 SPI1CON1 寄存器，且设置 MSTEN （SPI1CON1<5>）= 1。
3. 将 SPIROV 位 （SPI1STAT<6>）清零。
4. 通过将 SPIEN 位 （SPI1STAT<15>）置 1 使能 SPI1 模块工作。
5. 将待发送数据写入 SPI1BUF 寄存器。发送 （和接收）在数据写入 SPI1BUF 寄存器时立即开始。

例 35-1 给出了为主模式配置 SPI 寄存器的代码片段。

例 35-1： SPI 配置——主模式

```
/* Following code snippet shows SPI register configuration for MASTER mode*/

IFS0bits.SPI1IF = 0;       //Clear the Interrupt Flag
IEC0bits.SPI1IE = 0;       //disable the Interrupt
                           // SPI1CON1 Register Settings
SPI1CON1bits.DISSCK = 0;   //Internal Serial Clock is Enabled.
SPI1CON1bits.DISSDO = 0;   //SDO1 pin is controlled by the module.
SPI1CON1bits.MODE16 = 1;   //Communication is word-wide (16 bits).
SPI1CON1bits.SMP = 0;      //Input Data is sampled at the middle of data
output time.
SPI1CON1bits.CKE = 0;      //Serial output data changes on transition from
                           //Idle clock state to active clock state
SPI1CON1bits.CKP = 0;      //Idle state for clock is a low level; active
                           //state is a high level
SPI1CON1bits.MSTEN = 1;    //Master Mode Enabled
SPI1STATbits.SPIEN = 1;    //Enable SPI Module
SPI1BUF = 0x0000;          //Write data to be transmitted
                           //Interrupt Controller Settings
IFS0bits.SPI1IF = 0;       //Clear the Interrupt Flag
IEC0bits.SPI1IE = 1;       //Enable the Interrupt
```

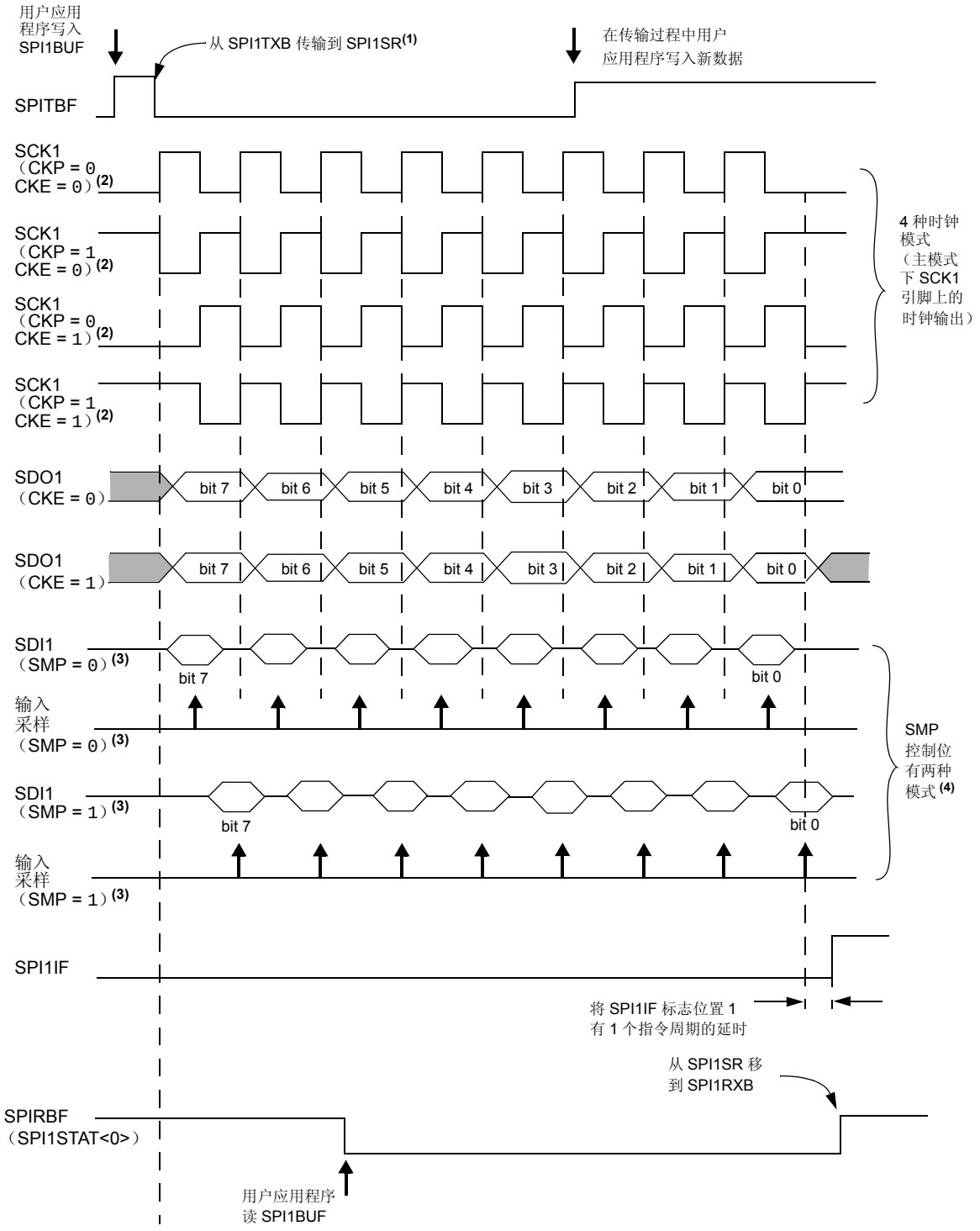
35.3.2.1.1 主模式下的外部时钟

在主模式下，也可将模块配置为依靠外部数据时钟工作。SPI1 时钟操作由 DISSCK 位（SPI1CON1<12>）控制。当此位置 1 时，禁止内部数据时钟，且当 SCK1 引脚上出现外部时钟脉冲时传输数据。主模式工作的所有其他方面都与上文所述相同。

注： DISSCK 位仅在 SPI 主模式下可用。

© 2008 Microchip Technology Inc.

图 35-3： SPI1 主模式时序



注 1： 如果没有待发送的数据，一旦用户应用程序写入 SPI1BUF， SPI1TXB 中的数据就会传输到 SPI1SR。

2： 图中所示四种 SPI1 时钟模式仅用于说明 CKP （SPI1CON1<6>）和 CKE （SPI1CON1<8>）位的功能。只可选择四种工作模式之一。

3： 图中所示 SMP 位 （SPI1CON1<9>）取两个不同值时的 SDI1 和输入采样，仅用于说明目的。在工作中只能选择 SMP 位的两种配置之一。

4： 图中所示为 8 位模式下的操作。除时钟脉冲数外， 16 位模式下的操作与此类似。

35.3.2.2 从模式

在从模式下，当 SCK1 引脚上出现外部时钟脉冲时发送和接收数据。 CKP （SPI1CON<6>）和 CKE （SPI1CON<8>）位决定在哪个时钟边沿发生数据传输。将待发送数据写入 SPI1BUF 寄存器或从该寄存器读取已接收数据。从模式下模块的其余操作与在上述主模式下的操作相同。

要将 SPI1 模块设置为工作在从模式，需执行下列步骤：

1. 清零 SPI1BUF 寄存器。
2. 如果使用中断：
   a) 清零 IFS0 寄存器中的 SPI1IF 位。
   b) 置 1 IEC0 寄存器中的 SPI1IE 位。
   c) 写 IPC2 寄存器中的 SPI1IP 位来设置中断优先级。
3. 将需要的设置写入SPI1CON1和SPI1CON2寄存器，且使MSTEN（SPI1CON1<5>）= 0。
4. 将 SMP 位清零。
5. 如果 CKE 位置 1，则必须将 SSEN 位 （SPI1CON1<7>）置 1 来使能 $\overline{SS1}$ 引脚。
6. 将 SPIROV 位 （SPI1STAT<6>）清零。
7. 通过将 SPIEN 位 （SPI1STAT<15>）置 1 使能 SPI1 工作。

例 35-2 给出了为从模式配置 SPI 寄存器的代码片段。

例 35-2： SPI 配置——从模式

```
/* Following code snippet shows SPI register configuration for SLAVE mode*/

IFS0bits.SPI1IF = 0;           //Clear the Interrupt Flag
IEC0bits.SPI1IE = 0;           //Disable The Interrupt
                               // SPI1CON1 Register Settings

SPI1CON1bits.DISSCK = 0;       //Internal Serial Clock is Enabled.
SPI1CON1bits.DISSDO = 0;       //SDO1 pin is controlled by the module.
SPI1CON1bits.MODE16 = 1;       //Communication is word-wide (16 bits).
SPI1CON1bits.SMP = 0;          //Input Data is sampled at the middle of data
                               //output time.
SPI1CON1bits.CKE = 0;          //Serial output data changes on transition
                               //from Idle clock state to active clock state
SPI1CON1bits.CKP = 0;          //Idle state for clock is a low level; active
                               //state is a high level
SPI1CON1bits.MSTEN = 0;        //Master Mode disabled
SPI1STATbits.SPIROV=0;         //No Receive Overflow Has Occurred
SPI1STATbits.SPIEN = 1;        //Enable SPI Module

                               //Interrupt Controller Settings
IFS0bits.SPI1IF = 0;           //Clear the Interrupt Flag
IEC0bits.SPI1IE = 1;           //Enable The Interrupt
```

35.3.2.2.1 从选择同步

$\overline{SS1}$ 引脚允许同步从模式。如果 SSEN 位（SPI1CON1<7>）置 1，则仅当 $\overline{SS1}$ 引脚被驱动为低电平状态时才能使能从模式下的发送和接收 （见图 35-5）。为确保 $\overline{SS1}$ 引脚能正常用作输入引脚，不能使用端口输出或与 $\overline{SS1}$ 引脚功能复用的其他外设输出。如果 SSEN 位置 1 且 $\overline{SS1}$ 引脚被驱动为高电平， SDO1 引脚将不再被驱动并且将变为三态，即使模块在发送过程中也是如此。

下次 $\overline{SS1}$ 引脚被驱动为低电平时，将使用 SPI1TXB 寄存器中保存的数据重新尝试被中止的发送。如果 SSEN 位没有被置 1，则 $\overline{SS1}$ 引脚不影响从模式下模块的工作。

注： 为满足模块的时序要求，当 CKE = 1 时，在从模式下必须使能 $\overline{SS1}$ 引脚 （详细信息请参见图 35-6）。

© 2008 Microchip Technology Inc.

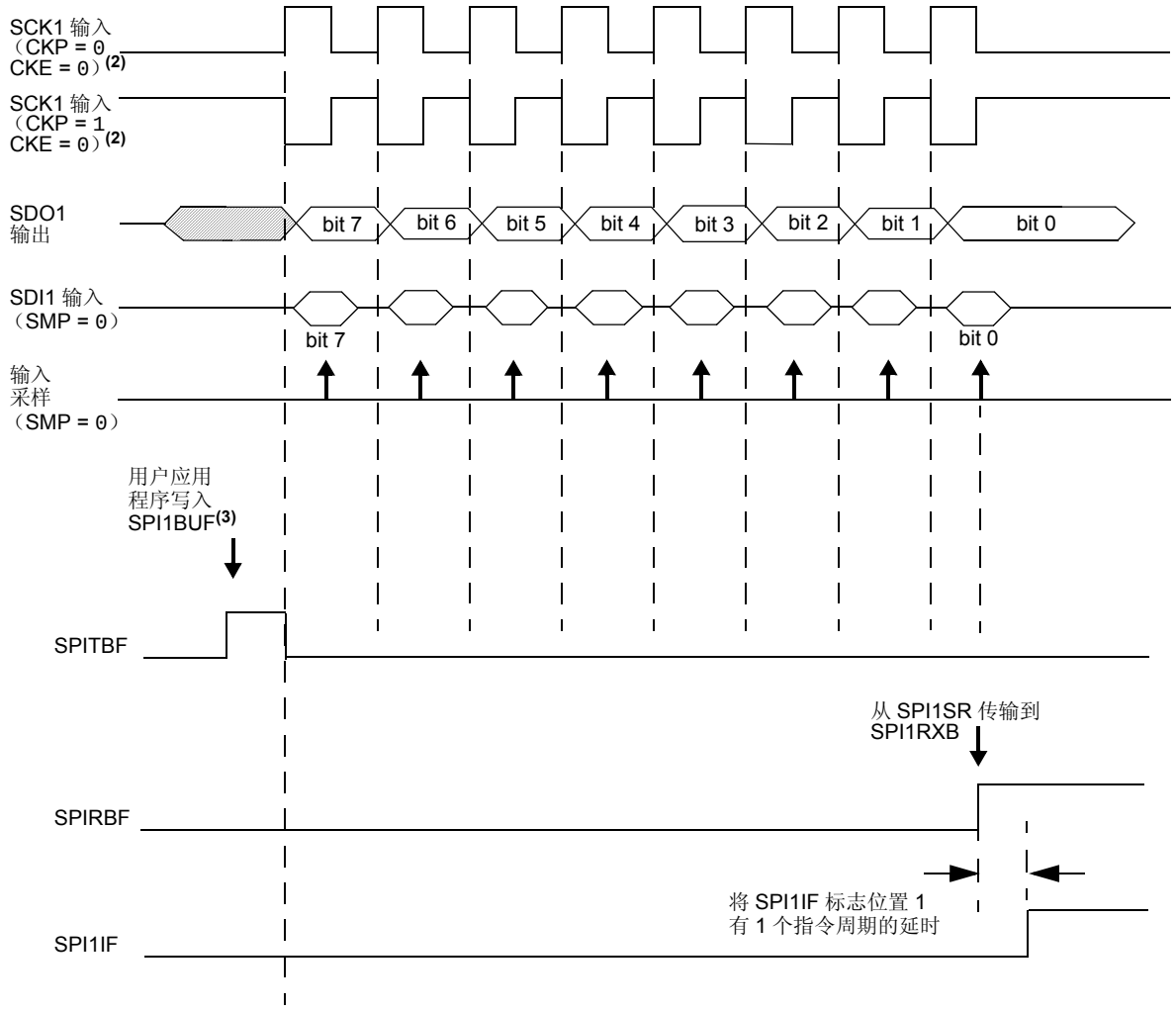
35.3.2.2.2 SPITBF 状态标志的操作

SPITBF 位 （SPI1STAT<1>）的功能在从工作模式下与主模式下有所不同。

如果 SSEN （SPI1CON1<7>）清零，当用户应用程序装载数据到 SPI1BUF 时， SPITBF 将置 1。它在模块将 SPI1TXB 中的数据传输到 SPI1SR 时清零。这与主模式下 SPITBF 位的功能类似。

如果 SSEN 位置 1， SPITBF 在用户应用程序装载 SPI1BUF 时置 1。但是，它只有在 SPI1 模块完成数据发送时才被清零。当 SS1 引脚变为高电平时，发送将被中止，但是可能在将来某个时间重新尝试发送。每个数据字都保存在 SPI1TXB 中，直到所有位都被发送到接收器中。

图 35-4： SPI1 从模式时序 （从选择引脚禁止） (1)

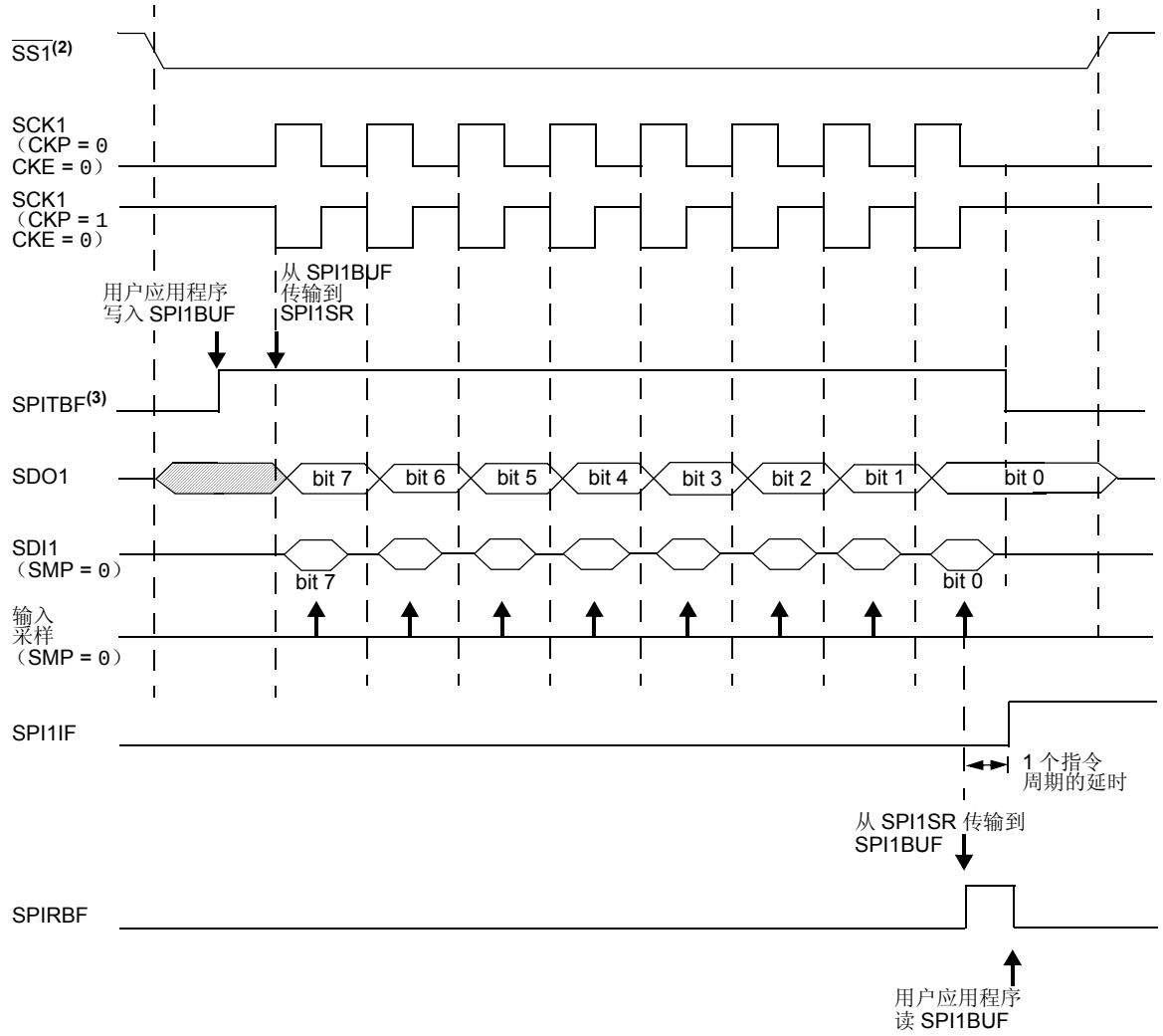


注 1： 图中所示为 8 位模式下的操作； 16 位模式下的操作与此类似。

2： 图中所示两种 SPI1 时钟模式仅用于说明 CKP（SPI1CON1<6>）和 CKE（SPI1CON1<8>）位的功能。可选择 CKP 和 CKE 位的任意组合来配置模块的工作。

3： 如果没有待发送的数据或正在进行发送，一旦用户应用程序写入SPI1BUF，SPI1BUF中的数据就会传输到SPI1SR。

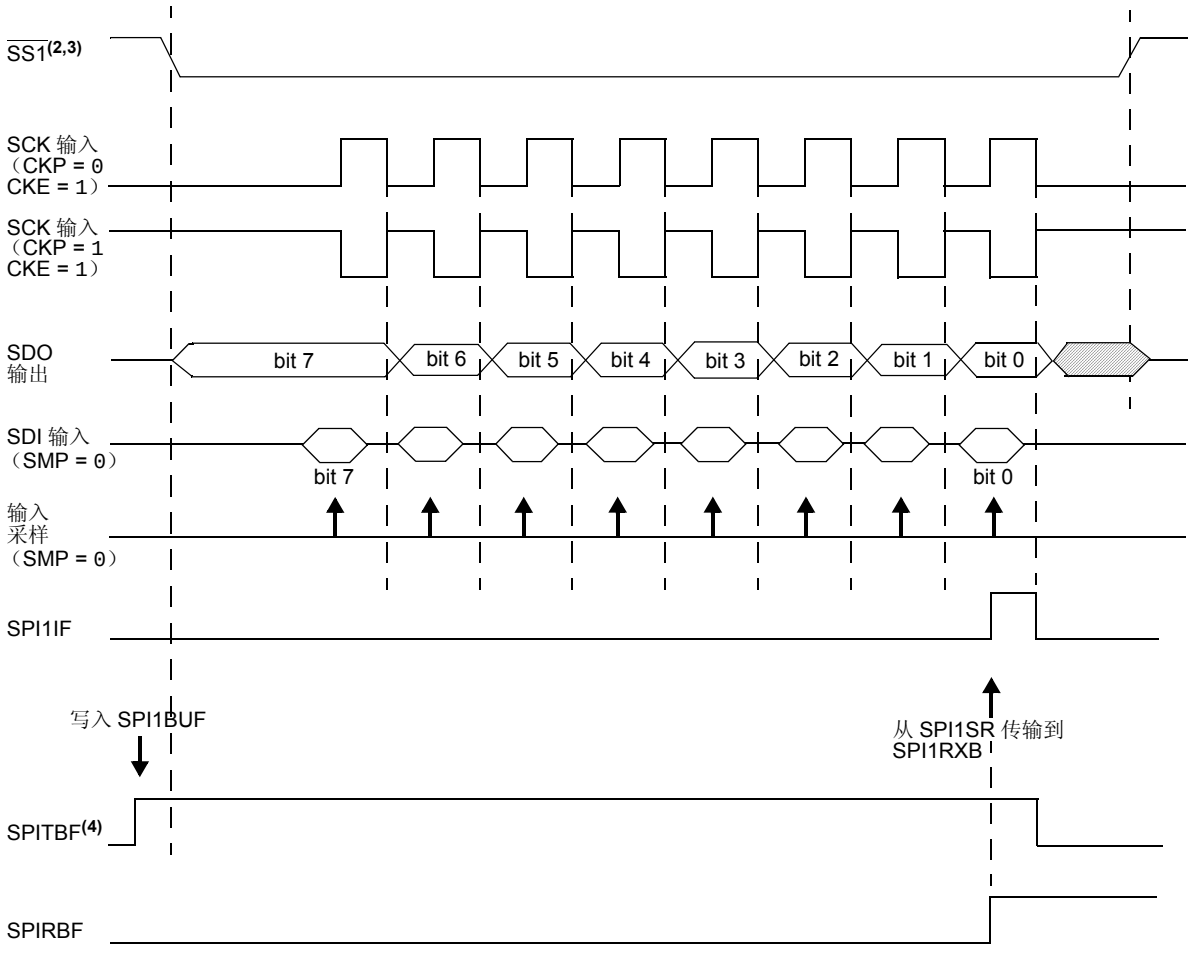
图 35-5： SPI1 从模式时序 （从选择引脚使能） (1)



注 1： 图中所示为 8 位模式下的操作； 16 位模式下的操作与此类似。
2： 当 SSEN 位 （SPI1CON1<7>）被置为 1 时，要使能从模式下的发送和接收， $\overline{SS1}$ 引脚必须被驱动为低电平。
3： 发送数据保存在 SPI1TXB 中， SPITBF 保持置 1 直到发送完所有位。

© 2008 Microchip Technology Inc.

图 35-6： SPI1 从模式时序 （CKE = 1）(1)



注 1： 图中所示为 8 位模式下的操作； 16 位模式下的操作与此类似。
2： 当 CKE = 1 时 SS1 引脚必须用于从模式工作。
3： 当 SSEN 位 （SPI1CON1<7>）被置为 1 时，要使能从模式下的发送和接收， SS1 引脚必须被驱动为低电平。
4： 发送数据保存在 SPI1TXB 中， SPITBF 保持置 1 直到发送完所有位。

35.3.3 帧 SPI 模式

当 SPI1 模块工作在主模式或从模式时，模块支持一个基本的帧 SPI 协议。模块使用四个控制位来配置帧 SPI 操作：

- FRMEN （SPI1CON2<15>）可使能帧 SPI 模式并使 SS1 引脚用作帧同步脉冲输入或输出引脚。 SSEN （SPI1CON1<7>）位的状态被忽略。
- SPIFSD 位 （SPI1CON2<14>）决定 SS1 引脚的输入 / 输出方向 （即模块是接收还是产生帧同步脉冲）。
- FRMPOL 位 （SPI1CON2<13>）选择单个 SPI 数据帧的帧同步脉冲的极性 （高电平有效还是低电平有效）。
- FRMDLY 位 （SPI1CON2<1>）选择同步脉冲是与第一个串行时钟脉冲一致还是在其之前出现。

SPI1 模块支持两种帧工作模式。在帧主模式下，SPI1 模块产生帧同步脉冲并在 $\overline{SS1}$ 引脚上为其他器件提供此脉冲。在帧从模式下， SPI1 模块使用 $\overline{SS1}$ 引脚上接收到的帧同步脉冲。

注： 在所有帧 SPI 模式下必须使用 $\overline{SS1}$ 和 SCK1 引脚。

支持帧 SPI 模式和非帧主 / 从模式的以下四种组合：

- SPI 主 – 帧主
- SPI 主 – 帧从
- SPI 从 – 帧主
- SPI 从 – 帧从

这四种模式决定 SPI1 模块是否产生串行时钟和帧同步脉冲。

当 FRMEN（SPI1CON2<14>）= 1 且 MSTEN（SPI1CON1<5>）= 1 时， SCK1 引脚成为输出引脚，且 SCK1 上的 SPI 时钟成为自由运行时钟。

当 FRMEN = 1 且 MSTEN = 0 时，SCK1 引脚成为输入引脚。假设提供给 SCK1 引脚的时钟源是自由运行的时钟。

时钟的极性由 CKP（SPI1CON1<6>）位选择。 CKE（SPI1CON1<8>）位在帧 SPI 模式下不使用，应该由用户应用程序编程为 0。

当 CKP = 0 时，帧同步脉冲输出和 SDO1 数据输出在 SCK1 引脚上时钟脉冲的上升沿发生变化。在串行时钟的下降沿，在 SDI1 输入引脚上采样输入数据。

当 CKP = 1 时，帧同步脉冲输出和 SDO1 数据输出在 SCK1 引脚上时钟脉冲的下降沿发生变化。在串行时钟的上升沿，在 SDI1 输入引脚上采样输入数据。

35.3.3.1 帧主模式和帧从模式

当 SPIFSD（SPI1CON2<13>）= 0 时， SPI1 模块处于帧主工作模式。在此模式下，当用户应用程序将发送数据写入 SPI1BUF 单元（从而将发送数据写入 SPI1TXB 寄存器）时，模块发出帧同步脉冲。在帧同步脉冲的末尾，SPI1TXB 中的数据被传输到 SPI1SR，同时开始发送 / 接收数据。

当 SPIFSD = 1 时，模块处于帧从模式。在此模式下，帧同步脉冲由外部时钟源产生。当模块采样到帧同步脉冲时，它把 SPI1TXB 寄存器的内容传输到 SPI1SR，同时开始发送 / 接收数据。在接收到帧同步脉冲前，用户应用程序必须确保 SPI1BUF 中装入了正确的发送数据。

注： 无论数据是否写入 SPI1BUF，接收到帧同步脉冲时都将启动发送。如果新数据尚未写入，将发送 SPI1TXB 中的原有数据。

© 2008 Microchip Technology Inc.

35.3.3.2 SPI 主 – 帧主模式

在 SPI 主 / 帧主模式下， SPI1 模块同时产生时钟信号和帧同步信号，如图 35-7 所示。通过将 MSTEN 和 FRMEN 位设置为 1 并将 SPIFSD 位设置为 0 来使能这种配置。

在此模式下，无论模块是否正在进行发送，都将在 SCK1 引脚上连续输出串行时钟。当写入 SPI1BUF 时， SS1 引脚将在适当的 SCK1 时钟发送边沿被驱动为有效状态 （由 FRMPOL 位确定）并在一个数据帧期间保持有效。如果 FRMDLY 控制位（SPI1CON2<1>）清零，将在发送数据前产生帧同步脉冲，如图 35-8 所示。如果 FRMDLY 位置 1，帧同步脉冲将与数据发送的开始一致，如图 35-9 所示。模块在 SCK1 的下一个发送边沿开始发送数据。

图 35-7： SPI 主 – 帧主模式连接图

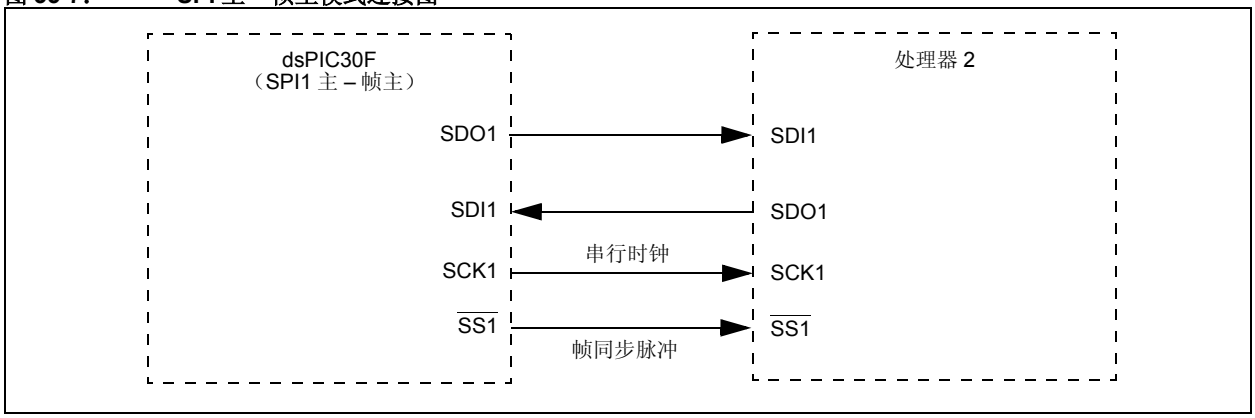


图 35-8： SPI 主 – 帧主模式时序 （FRMDLY = 0）

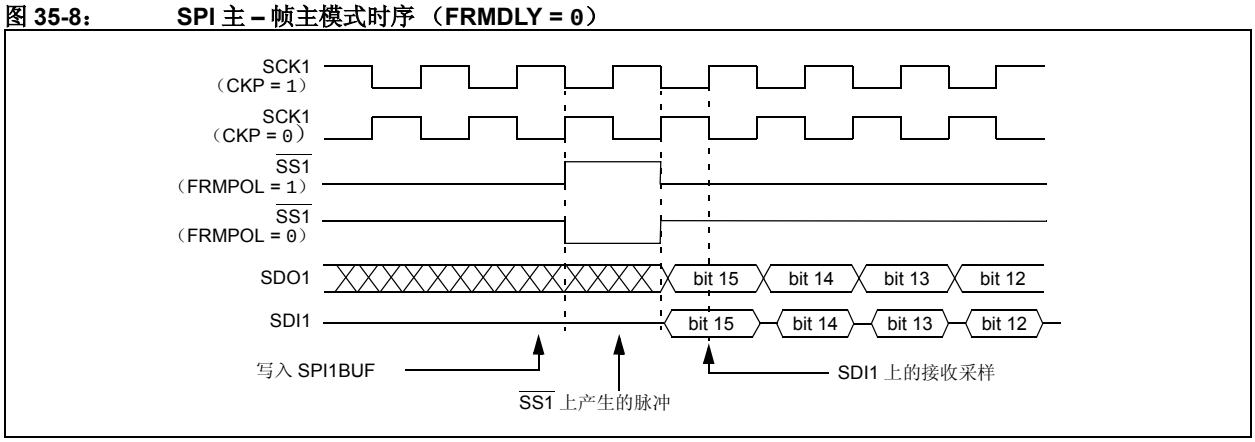
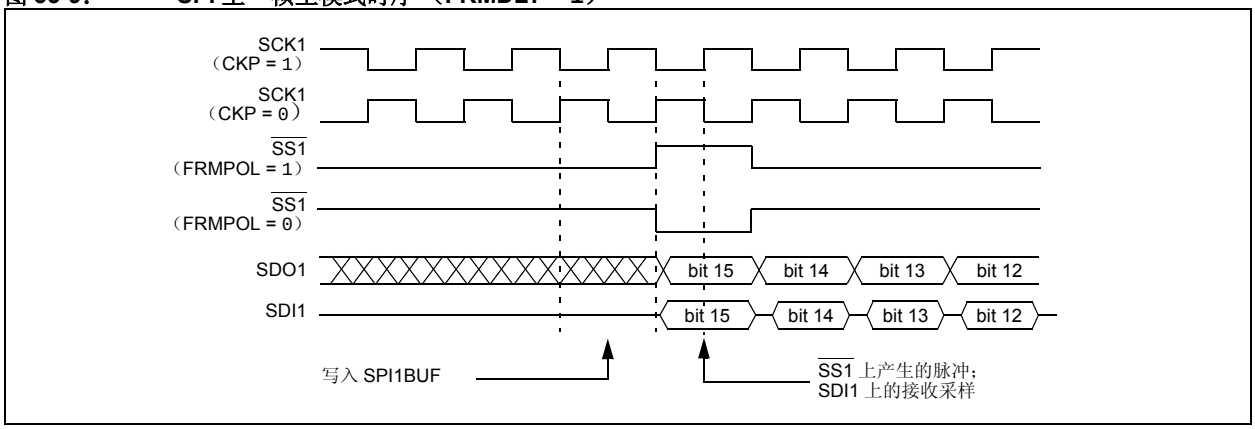


图 35-9： SPI 主 – 帧主模式时序 （FRMDLY = 1）



35.3.3.3 SPI 主 – 帧从模式

在 SPI 主 / 帧从模式下，模块产生时钟信号，但是使用从模块的帧同步信号进行数据传输，（如图 35-10 所示）。通过将 MSTEN、 FRMEN 和 SPIFSD 位设置为 1 来使能此模式。

在此模式下， $\overline{SS1}$ 引脚为输入引脚，并且在 SPI1 时钟的采样沿对其进行采样。当采样到有效电平状态时，在后续的 SPI1 时钟发送沿就会发送数据。当发送完成时中断标志 SPI1IF 置 1。在 $\overline{SS1}$ 引脚上接收到信号前，用户应用程序必须确保 SPI1BUF 中装入了正确的发送数据。

图 35-10： SPI 主 – 帧从模式连接图

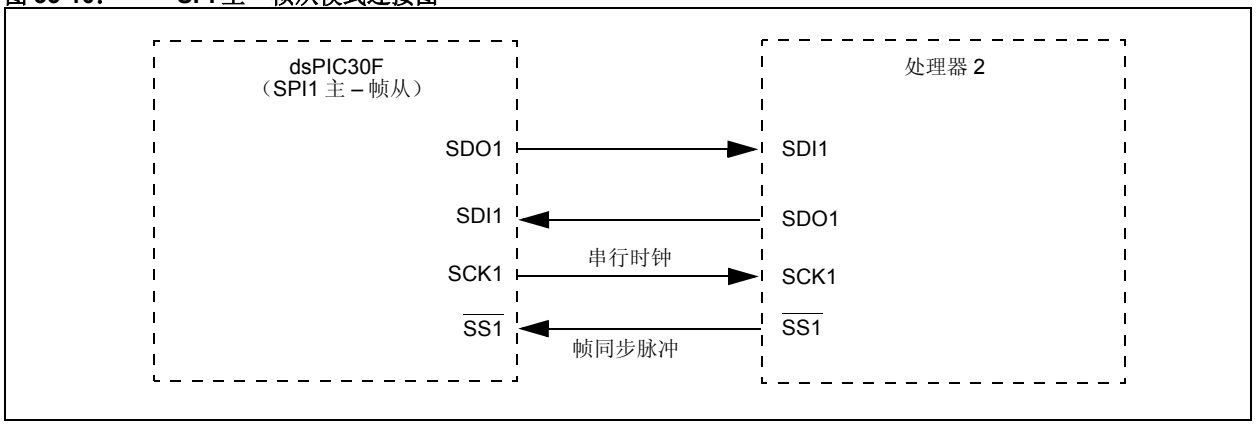


图 35-11： SPI 主 – 帧从模式时序 （FRMDLY = 0）

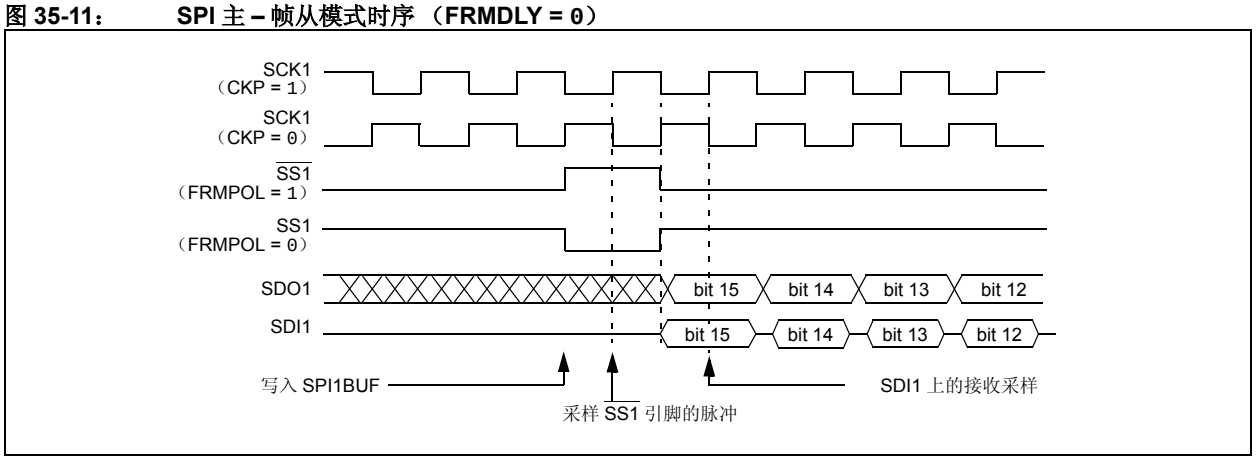
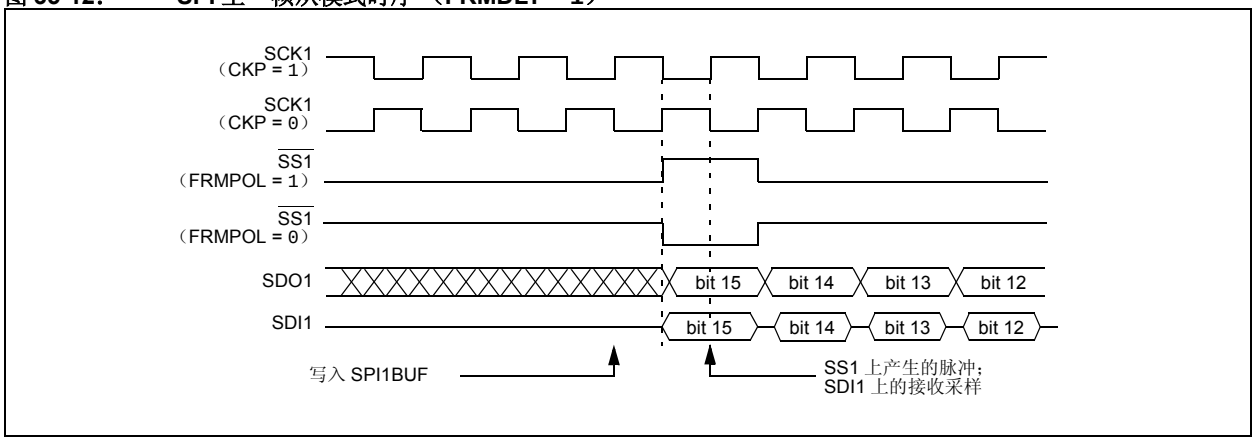


图 35-12： SPI 主 – 帧从模式时序 （FRMDLY = 1）



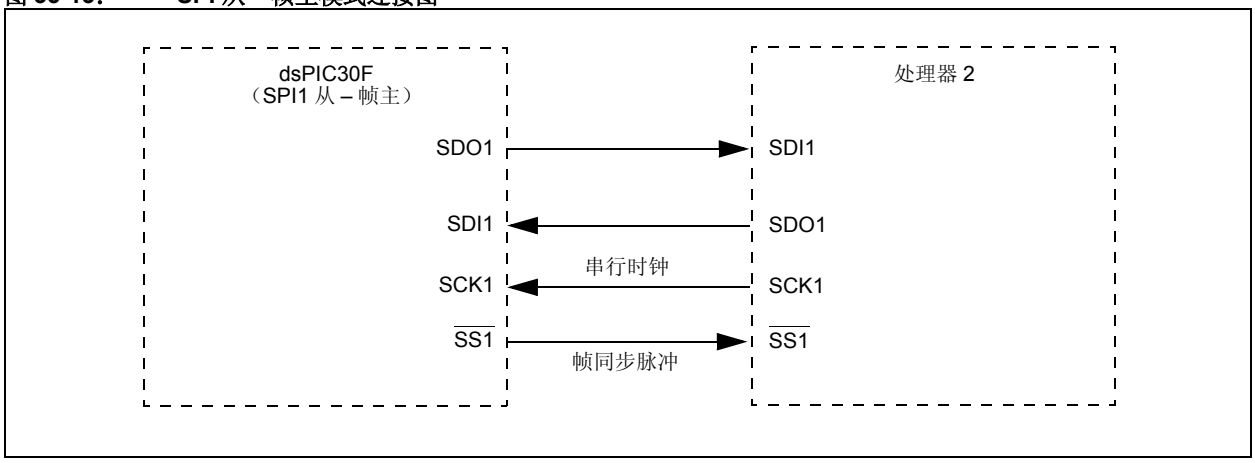
© 2008 Microchip Technology Inc.

35.3.3.4 SPI 从 – 帧主模式

在 SPI 从 / 帧主模式下，模块作为 SPI 从模式工作，从其他 SPI 模块获取时钟；但是此模块产生帧同步信号来控制数据传输 （如图 35-13 所示）。通过将 MSTEN 位设置为 0、 FRMEN 位设置为 1 并将 SPIFSD 位设置为 0 来使能此模式。

在从模式下，输入 SPI 时钟是连续的。当 SPIFSD 位为低时，$\overline{SS1}$ 引脚是输出引脚。因此，当写入 SPI1BUF 时，模块在适当的 SPI 时钟发送沿将 $\overline{SS1}$ 引脚驱动为有效电平状态并保持一个 SPI 时钟周期。在适当的 SPI 时钟发送沿开始发送数据。

图 35-13： SPI 从 – 帧主模式连接图

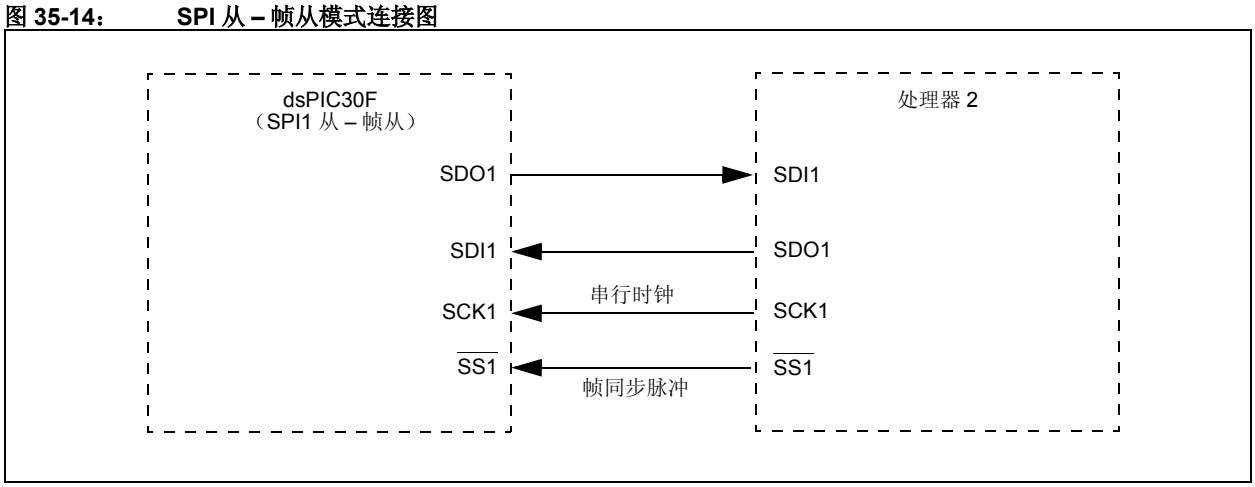


35.3.3.5 SPI 从 – 帧从模式

在 SPI 从 / 帧从模式下，模块同时从主模块获取时钟信号和帧同步信号 （如图 35-14 所示）。通过将 MSTEN 位设置为 0、 FRMEN 位设置为 1 并将 SPIFSD 位设置为 1 来使能此模式。

在此模式下，SCK1 和 $\overline{SS1}$ 引脚都是输入引脚。$\overline{SS1}$ 引脚在 SPI 时钟的采样沿进行采样。当 $\overline{SS1}$ 上采样到有效电平状态时，将在适当的 SCK1 发送沿发送数据。

图 35-14： SPI 从 – 帧从模式连接图



35.3.4 SPI1 只接收操作

将 DISSDO 控制位 （SPI1CON1<11>）置 1 将禁止 SDO1 引脚上的发送。这允许将 SPI1 模块配置为只接收工作模式。如果 DISSDO 位置 1， SDO1 引脚将由相应的端口功能控制。

DISSDO 功能可应用于所有的 SPI 工作模式。

35.3.5 SPI1 错误处理

如果一个新的数据字已经移入 SPI1SR，但是先前的 SPI1BUF 内容尚未被读取，那么 SPIROV 位（SPI1STAT<6>）将被置 1。SPI1SR 中接收到的任何数据都不进行传输，同时禁止后续的数据接收，直到 SPIROV 位清零。模块不会自动清零 SPIROV 位；它必须由用户应用程序来清零。

只要 SPIROV、 SPIRBF （SPI1STAT<0>）或 SPITBF （SPI1STAT<1>）位置 1， SPI1 中断标志位 （SPI1IF）就会被置 1。硬件不能清零此中断标志，它必须由软件来复位。仅当 IEC0 控制寄存器中相应的 SPI1IE 位置 1 时，才会产生实际的 SPI1 中断。

© 2008 Microchip Technology Inc.

35.4 主模式时钟频率

在主模式下，提供给 SPI1 模块的时钟的周期就是指令周期（TCY）。然后此时钟信号由主预分频器（由 PPRE<1:0>（SPI1CON1<1:0>）指定）和辅助预分频器（由 SPRE<2:0>（SPI1CON1<4:2>）指定）进行预分频。经过预分频的指令时钟就成为串行时钟并通过 SCK1 引脚提供给外部器件。

注：	注意 SCK1 信号时钟在正常 SPI 模式下不是自由运行的。它仅在 SPI1BUF 装入数据时运行 8 或 16 个脉冲时间。但是在帧模式下，它会连续运行。

公式 35-1 可用于根据主预分频比和辅助预分频比设置来计算 SCK1 时钟频率。

公式 35-1： SPI 时钟频率

$$FSCK = \frac{FCY}{\text{主预分频比} * \text{辅助预分频比}}$$

表 35-1 列出了部分 SPI 时钟频率示例 （单位为 kHz）。

注：	并不支持所有时钟频率。如需更多信息，参见特定器件数据手册的 SPI 模块时序规范。

表 35-1： SCK1 频率示例 (1)

FCY = 30 MHz		**辅助预分频比设置**				
		1:1	**2:1**	**4:1**	**6:1**	**8:1**
主预分频比设置	1:1	30000	15000	7500	5000	3750
	4:1	7500	3750	1875	1250	938
	16:1	1875	938	469	313	234
	64:1	469	234	117	78	59
FCY = 5 MHz						
主预分频比设置	1:1	5000	2500	1250	833	625
	4:1	1250	625	313	208	156
	16:1	313	156	78	52	39
	64:1	78	39	20	13	10

注 1： 表中 SCK1 频率的单位为 kHz。

35.5 节能模式下的工作

dsPIC30F SMPS 和数字电源转换器件系列有三种功耗模式：正常工作模式（满功耗）和通过 PWRSAV 指令启用的两种节能模式。根据不同的 SPI 模式选择，进入节能模式也可能影响 SPI1 模块的工作。

35.5.1 休眠模式

当器件进入休眠模式时，系统时钟会被禁止。进入休眠模式的结果依赖于启用休眠模式时，SPI1 模块被配置为哪种模式（主模式或从模式）。

35.5.1.1 主模式工作

以下为将 SPI1 模块配置为主模式工作时进入休眠模式的结果：

- SPI1 模块的波特率发生器停止并复位。
- 在休眠模式下发送器和接收器停止。在唤醒时，发送器和接收器不会继续进行部分完成的传输。
- 如果 SPI1 模块在发送或者接收过程中进入休眠模式，那么发送和接收将会中止。因为在发送或接收未完成时没有自动的方式能防止 SPI1 模块进入休眠模式，因此用户应用程序必须将进入休眠模式与 SPI1 模块工作同步以防止传输中止。

35.5.1.2 从模式工作

因为在从模式下， SCK1 的时钟脉冲由外部提供，所以模块在休眠模式下将继续工作。它将在进入休眠模式的转变期间内完成所有事务处理。完成事务处理后，SPIRBF 标志将置 1。因此 SPI1IF 位将被置 1。如果允许了 SPI1 中断（SPI1IE = 1），器件将从休眠模式唤醒。如果 SPI1 中断的优先级高于当前的 CPU 优先级，将从 SPI1 中断向量地址处恢复代码执行。否则，将继续从先前进入休眠模式时执行的 PWRSAV 指令后下一条指令处执行代码。如果此模块作为从器件工作，那么在进入休眠模式时它将不会复位。

当 SPI1 模块进入或退出休眠模式时，寄存器内容不受影响。

35.5.2 空闲模式

当器件进入空闲模式时，系统时钟源保持工作。 SPISIDL 位（SPI1STAT<13>）选择模块在空闲模式下是停止工作还是继续工作。

如果 SPISIDL = 1，SPI1 模块将在进入空闲模式时停止通信。其工作方式将和处于休眠模式时相同。如果 SPISIDL = 0（默认选择），模块将在空闲模式下继续工作。

© 2008 Microchip Technology Inc.

35.6 与 SPI1 模块相关的特殊功能寄存器

表 35-2： SPI1 寄存器映射

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>			PPRE<1:0>		0000
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI1BUF	0246	SPI1 发送和接收缓冲寄存器																0000

© 2008 Microchip Technology Inc.

35.7 相关应用笔记

本节列出了与手册的这一章相关的应用笔记。这些应用笔记可能并非专门针对 dsPIC30F SMPS 和数字电源转换器件系列而写的，但其概念是相近的，通过适当修改并受到一定的限制即可使用。目前与串行外设接口 （SPI）模块相关的应用笔记包括：

标题	应用笔记文档编号
Interfacing SPI™ Serial EEPROMs to PIC18 Devices	AN1006
Interfacing Microchip's MCP41XXX and MCP42XXX Digital Potentiometers to a PICmicro® Microcontroller	AN746
Interfacing Microchip's MCP3201 Analog-to-Digital Converter to the PICmicro® Microcontroller	AN719

注： 如需获得更多关于 dsPIC30F SMPS 和数字电源转换器件系列的应用笔记和代码示例，请访问 Microchip 网站 （www.microchip.com）。

© 2008 Microchip Technology Inc.

35.8 版本历史

版本 A （2007 年 2 月）

这是本文档的初始版本。

注：

© 2008 Microchip Technology Inc.