
第 20 章 串行外设接口 (SPI™)

目录

本章包括下列主题：

20.1 简介	20-2
20.2 状态和控制寄存器	20-4
20.3 工作模式	20-7
20.4 SPI 主控模式时钟频率	20-19
20.5 低功耗模式下的工作	20-20
20.6 与 SPI 模块相关的特殊功能寄存器	20-22
20.7 相关应用笔记	20-23
20.8 版本历史	20-24

20.1 简介

串行外设接口（Serial Peripheral Interface，SPI）模块是一个**同步串行接口**，可用于与其他外设或者单片机进行通信。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 系列器件在单个器件上提供一个或两个 SPI 模块，具体取决于具体不同的器件。SPI1 和 SPI2 功能相同。很多高引脚数（64 引脚或更多）封装的器件具有 SPI2 模块，而 SPI1 模块则是所有的器件都具有的。

注： 在本章中，SPI 模块统称为 SPIx，或分别称为 SPI1 和 SPI2。特殊功能寄存器也使用类似的符号表示上述类似的意思。例如，SPIxCON 指 SPI1 或 SPI2 模块的控制寄存器。

SPI 串口包含下列特殊功能寄存器（SFR）：

- SPIxBUF：地址位于 SFR 空间，用于缓冲待发送数据和已接收数据。此地址由 SPIxTXB 和 SPIxRXB 寄存器共享。
- SPIxCON：配置模块各种操作模式的控制寄存器。
- SPIxSTAT：显示各种状态条件的状态寄存器。

此外，还有一个 16 位移位寄存器 SPIxSR，此寄存器不映射到存储器空间。该寄存器可用于将数据移入和移出 SPI 端口。

存储器映射的 SFR（SPIxBUF）是 SPI 数据接收 / 发送寄存器。在内部，SPIxBUF 寄存器实际上由两个独立的寄存器（SPIxTXB 和 SPIxRXB）组成。接收缓冲寄存器 SPIxRXB 和发送缓冲寄存器 SPIxTXB 是两个单向 16 位寄存器。这两个寄存器共享名为 SPIxBUF 的 SFR 地址单元。如果用户将需要发送的数据写入了 SPIxBUF 地址单元，该数据会在内部写入 SPIxTXB 寄存器。与此类似，当用户从 SPIxBUF 读取已接收到的数据时，该数据在内部是从 SPIxRXB 寄存器读取的。这种接收和发送操作的双缓冲可以使数据在后台连续传输。发送和接收可同时进行。

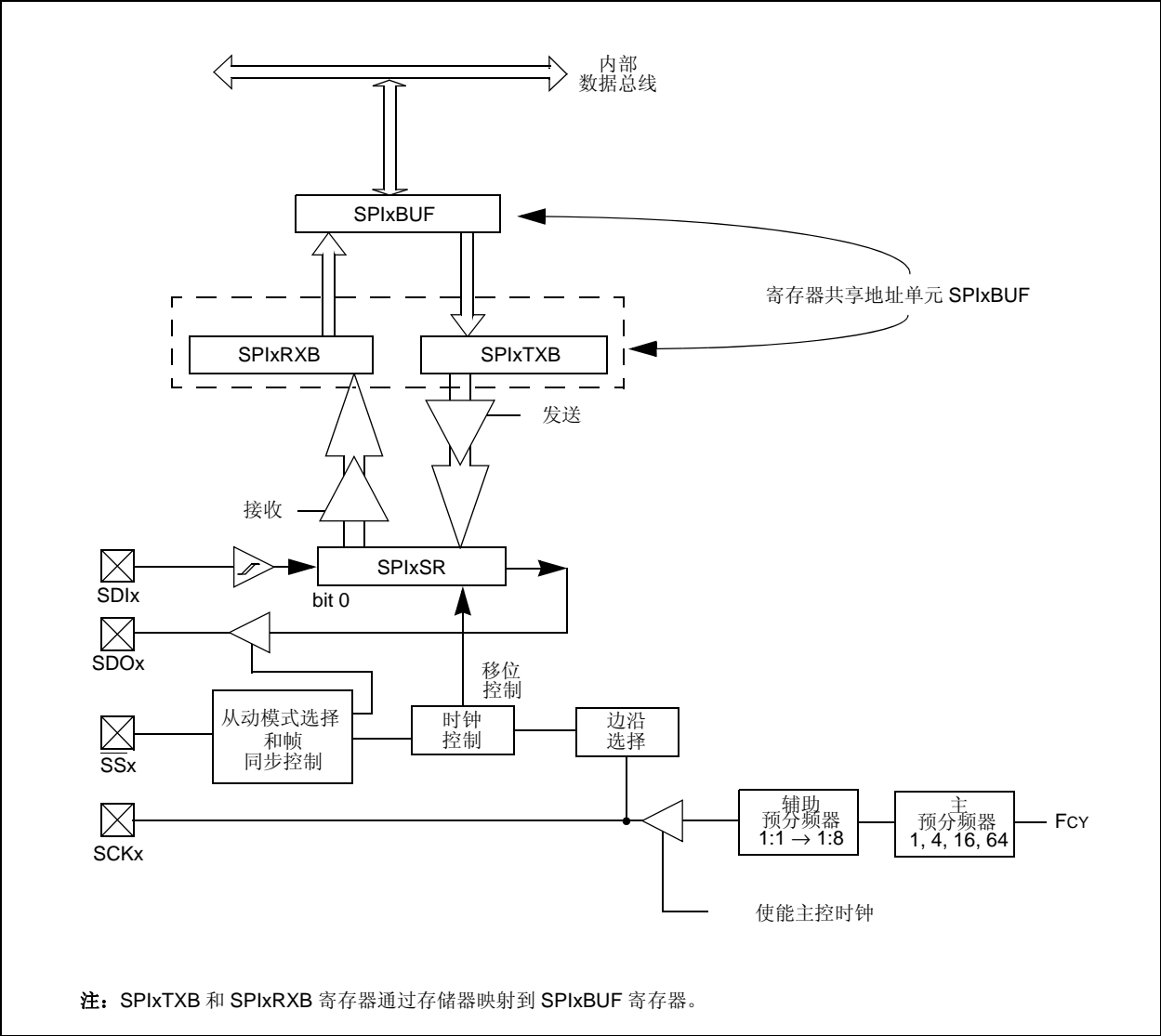
注： 用户无法直接写入 SPIxTXB 寄存器或读取 SPIxRXB 寄存器。所有的读写操作都是在 SPIxBUF 寄存器中进行的。

SPI 串行接口由以下四个引脚组成：

- SDIx：串行数据输入
- SDOx：串行数据输出
- SCKx：移位时钟输入或输出
- SSx：低电平有效从动选择或者帧同步 I/O 脉冲

注： SPI 模块可以配置为使用 3 个或 4 个引脚工作。在 3 引脚模式下，不使用 SSx 引脚。

图 20-1: SPI 模块框图



20.2 状态和控制寄存器

寄存器 20-2: **SPIxSTAT: SPI 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/W-0 HS	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

- bit 15 **SPIEN:** SPI 使能位
1 = 使能模块并将 SCKx、SDOx、SDIx 和 SSx 配置为串口引脚
0 = 禁止模块
- bit 14 **未用:** 读作 0
- bit 13 **SPISIDL:** 在空闲模式停止位
1 = 当器件进入空闲模式时模块不继续工作
0 = 在空闲模式下模块继续工作
- bit 12-7 **未用:** 读作 0
- bit 6 **SPIROV:** 接收溢出标志位
1 = 一个新字节 / 字已完全接收并丢弃。用户软件未读取 SPIxBUF 寄存器中原先的数据。
0 = 没有发生溢出
- bit 5-2 **未用:** 读作 0
- bit 1 **SPITBF:** SPI 发送缓冲器满状态位
1 = 未开始发送, SPIxTXB 满
0 = 发送开始, SPIxTXB 空
当 CPU 写 SPIxBUF 地址单元并装载 SPIxTXB 时, 该位由硬件自动置位。
当 SPIx 模块将数据从 SPIxTXB 传输到 SPIxSR 时, 该位由硬件自动清零。
- bit 0 **SPIRBF:** SPI 接收缓冲器满状态位
1 = 接收完成, SPIxRXB 满
0 = 接收未完成, SPIxRXB 空
当 SPIx 将数据从 SPIxSR 传输到 SPIxRXB 时, 该位由硬件自动置位。
当内核通过读 SPIxBUF 地址单元读 SPIxRXB 时, 该位由硬件自动清零。

图注:

R = 可读位	W = 可写位	U = 未用位, 读作 0
HC = 由硬件清零	HS = 由硬件置位	
-n = 复位时的值	1 = 置位	0 = 清零
		x = 未知

寄存器 20-2: SPIxCON: SPIx 控制寄存器

高字节:							
U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE
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 未用：读作 0
- bit 14 **FRMEN**: 帧 SPI 支持位
1 = 使能帧 SPI 支持
0 = 禁止帧 SPI 支持
- bit 13 **SPIFSD**: SSx 引脚上的帧同步脉冲方向控制位
1 = 帧同步脉冲输入（从动模式）
0 = 帧同步脉冲输出（主控模式）
- bit 12 未用：读作 0
- bit 11 **DISSDO**: SDOx 引脚禁止位
1 = 模块不使用 SDOx 引脚。该引脚由相关端口寄存器控制。
0 = SDOx 引脚由模块控制
- bit 10 **MODE16**: 字 / 字节通信选择位
1 = 通信为字宽（16 位）
0 = 通信为字节宽（8 位）
- bit 9 **SMP**: SPI 数据输入采样相位位
主控模式:
1 = 输入数据在数据输出时间末尾采样
0 = 输入数据在数据输出时间中间采样
从动模式:
当 SPI 在从动模式下使用时，必须将 SMP 清零。
- bit 8 **CKE**: SPI 时钟沿选择位
1 = 串行输出数据在有效时钟状态转变为空闲时钟状态时变化（参见 bit 6）
0 = 串行输出数据在空闲时钟状态转变为有效时钟状态时变化（参见 bit 6）
注: 在帧 SPI 模式下未使用 CKE 位。在帧 SPI 模式下（FRMEN = 1），用户应将该位编程为 0。
- bit 7 **SSEN**: 从动选择使能（从动模式）位
1 = SS 引脚用于从动模式
0 = 模块不使用 SS 引脚。引脚由端口功能控制。
- bit 6 **CKP**: 时钟极性选择位
1 = 空闲状态时钟信号为高电平；有效状态为低电平
0 = 空闲状态时钟信号为低电平；有效状态为高电平
- bit 5 **MSTEN**: 主控模式使能位
1 = 主控模式
0 = 从动模式

寄存器 20-2: SPIxCON: SPIx 控制寄存器 (续)

- bit 4-2 **SPRE<2:0>**: 辅预分频比 (主控模式) 位
(支持设置: 1:1、2:1 到 8:1 全部支持)
111 = 辅预分频比 1:1
110 = 辅预分频比 2:1
.....
000 = 辅预分频比 8:1
- bit 1-0 **PPRE<1:0>**: 主预分频比 (主控模式) 位
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

图注:			
R = 可读位	W = 可写位	U = 未用位, 读作 0	
-n = POR 值	1 = 置位	0 = 清零	x = 未知

20.3 工作模式

以下各节讨论了 SPI 模块灵活的工作模式：

- 8 位和 16 位数据发送 / 接收
- 主控模式和从动模式
- 帧 SPI 模式

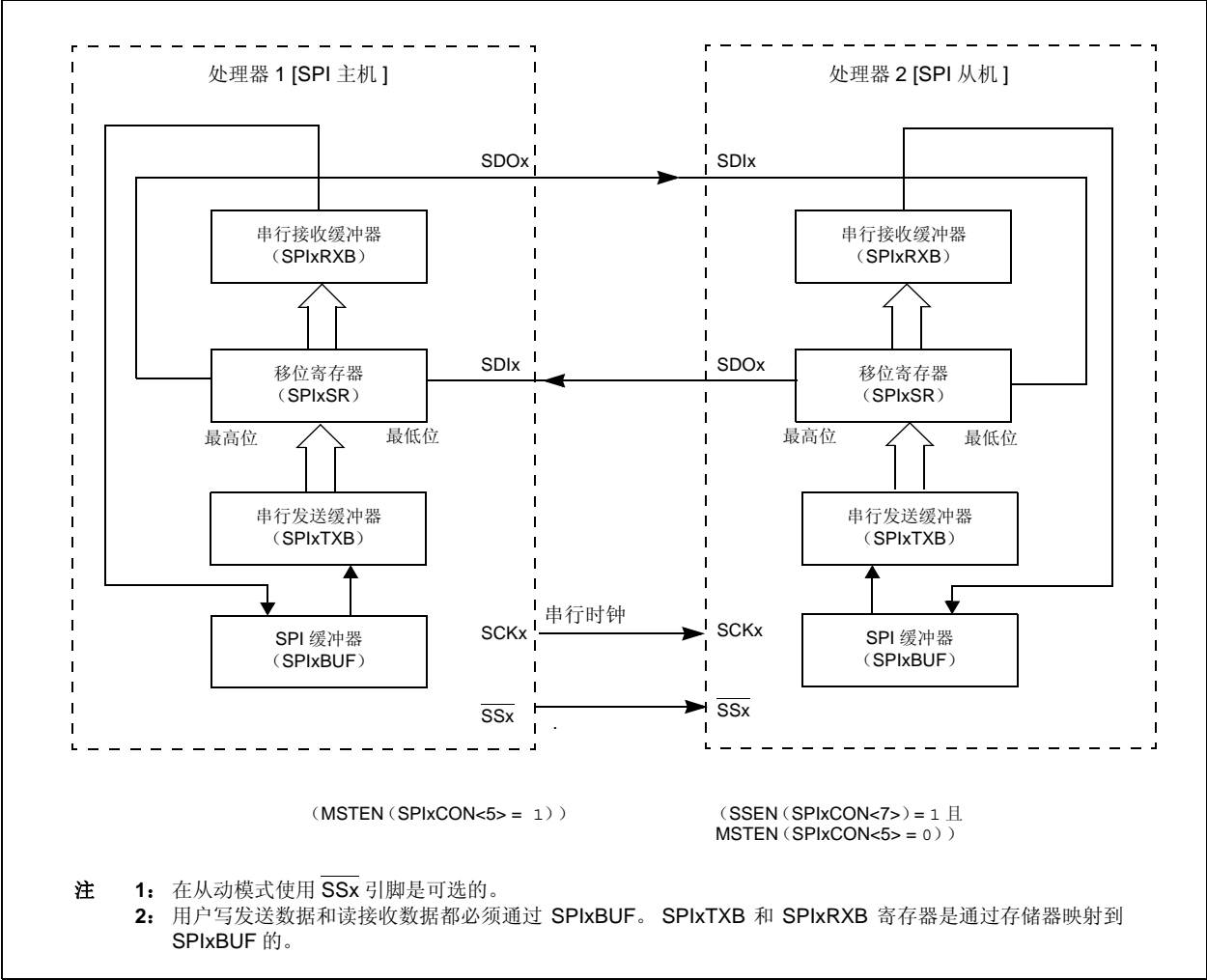
20.3.1 8 位与 16 位工作模式

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

- 当 **MODE16** (**SPIxCON<10>**) 位的值变化时，模块会复位。因此，在正常工作过程中不应该改变该位。
- 8 位工作模式下数据是从 **SPIxSR** 的 **bit 7** 发送的，而在 16 位工作模式下，则是从 **SPIxSR** 的 **bit 15** 发送的。在两种模式下，数据都会移入 **SPIxSR** 的 **bit 0**。
- 在 8 位模式下移入 / 移出数据需要 **SCKx** 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。

20.3.2 主控模式和从动模式

图 20-2: SPI 主 / 从连接



20.3.2.1 主控模式

遵循以下步骤将 SPI 模块设置为工作在主控模式：

1. 如果使用中断：
 - 清零相应 IFSn 寄存器中的 SPIxIF 位。
 - 置位相应 IECn 寄存器中的 SPIxIE 位。
 - 向相应的 IPCn 寄存器设置 SPIxIP 位。
2. 将所需设置写入 SPIxCON 寄存器，同时 MSTEN (SPIxCON<5>) = 1。
3. 将 SPIROV 位 (SPIxSTAT<6>) 清零。
4. 通过将 SPIEN 位 (SPIxSTAT<15>) 置位使能 SPI 工作。
5. 将待发送数据写入 SPIxBUF 寄存器。数据一写入 SPIxBUF 寄存器，发送（以及接收）就会立即开始。

在主控模式下，系统时钟被预分频，然后作为串行时钟使用。预分频基于 PPRE<1:0> (SPIxCON<1:0>) 和 SPRE<1:0> (SPIxCON<4:2>) 位的设置。串行时钟通过 SCKx 引脚输出到从动器件。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，请参阅第 20.4 节“SPI 主控模式时钟频率”。

CKP 和 CKE 位确定在哪个时钟沿发送数据。

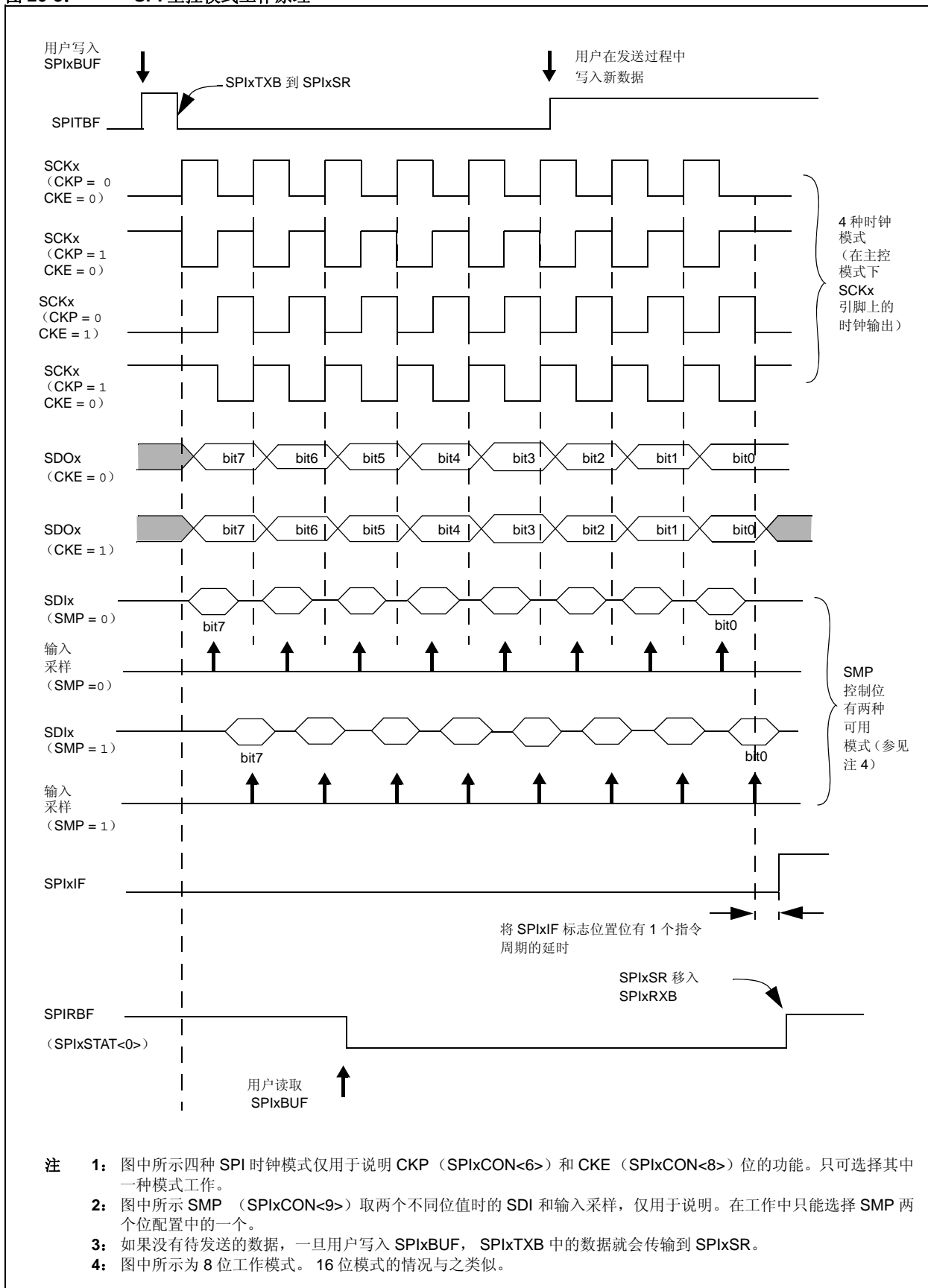
待发送数据和已接收数据都分别向 SPIxBUF 寄存器写入和从该寄存器读取。

以下内容描述了主控模式下 SPI 模块的工作原理。

1. 一旦模块被设置为主控工作模式并使能，待发送数据就会写入 SPIxBUF 寄存器。SPITBF (SPIxSTAT<1>) 位置位。
2. SPIxTXB 的内容移到移位寄存器 SPIxSR，并且模块将 SPITBF 位清零。
3. 一组 8/16 个时钟脉冲将 8/16 位发送数据从 SPIxSR 移出到 SDOx 引脚，同时将 SDIx 引脚的数据移入 SPIxSR。
4. 当传输结束后，会发生以下事件：
 - 中断标志位 SPIxIF 置位。通过将中断使能位 SPIxIE 置位可以允许 SPI 中断。SPIxIF 标志不会被硬件自动清零。
 - 另外，当正在进行的发送和接收操作结束后，SPIxSR 的内容会移到 SPIxRXB 寄存器。
 - SPIRBF (SPIxSTAT<0>) 位由模块置位，表明接收缓冲器满。一旦用户代码读 SPIxBUF 寄存器，硬件就会将 SPIRBF 位清零。
5. 当 SPI 模块需要从 SPIxSR 传输数据到 SPIxRXB 时，如果 SPIRBF 位置位（接收缓冲器满），模块会将 SPIROV (SPIxSTAT<6>) 位置位，表明产生了溢出条件。
6. 只要 SPITBF (SPIxSTAT<1>) 清零，用户软件就可以在任何时候将待发送数据写入 SPIxBUF。写入可以与 SPIxSR 移出前面写入的数据同时发生，因此可以允许连续发送。

注： 用户不能直接写 SPIxSR 寄存器。对 SPIxSR 寄存器的所有写入都是通过 SPIxBUF 寄存器进行的。

图 20-3: SPI 主控模式工作原理



20.3.2.2 从动模式

遵循以下步骤将 SPI 模块设置为从动工作模式：

1. 将 SPIxBUF 寄存器清零。
2. 如果使用中断：
 - 清零相应 IFSn 寄存器中的 SPIxIF 位。
 - 置位相应 IECn 寄存器中的 SPIxIE 位。
 - 向相应的 IPCn 寄存器设置 SPIxIP 位。
3. 将所需设置写入 SPIxCON 寄存器，同时使 MSTEN (SPIxCON<5>) = 0。
4. 将 SMP 位清零。
5. 如果 CKE 位置位，则 SSEN 位必须置位，从而使能 $\overline{\text{SSx}}$ 引脚。
6. 将 SPIROV 位 (SPIxSTAT<6>) 清零，并且
7. 通过将 SPIEN 位 (SPIxSTAT<15>) 置位使能 SPI 工作。

在从动模式下，在外部时钟脉冲出现在 SCKx 引脚时发送和接收数据。CKP (SPIxCON<6>) 和 CKE (SPIxCON<8>) 位决定数据发送发生在哪个时钟沿。

待发送数据和已接收数据分别向 SPIxBUF 寄存器写入和从该寄存器读取。

模块在该模式下的其余工作与在主控模式下相同。

从动模式还提供了一些其他功能，它们是：

从动模式选择同步： $\overline{\text{SSx}}$ 引脚允许同步从动模式。如果 SSEN (SPIxCON<7>) 位置位，只有 $\overline{\text{SSx}}$ 引脚驱动为低电平状态时才会使能从动模式下的发送和接收。为了使 $\overline{\text{SSx}}$ 引脚能作为输入引脚使用，不能驱动端口输出或其他外设输出。如果 SSEN 位置位且 $\overline{\text{SSx}}$ 引脚驱动为高电平，SDOx 引脚将不再被驱动并将呈现为三态，即使模块处于发送过程中也是如此。在下次 $\overline{\text{SSx}}$ 引脚驱动为低电平时，使用保存在 SPIxTXB 寄存器的数据重试上次中止的发送。如果 SSEN 位没有置位， $\overline{\text{SSx}}$ 引脚不会影响从动模式下的模块工作。

SPITBF 状态标志工作原理：SPITBF (SPIxSTAT<1>) 位的功能在从动工作模式下是与主控模式不同的。以下描述了从动工作模式下 SPITBF 的各种设置所对应的功能：

1. 如果 SSEN (SPIxCON<7>) 清零，SPITBF 将在用户代码装入 SPIxBUF 时置位。它将在模块将 SPIxTXB 中的数据传送到 SPIxSR 时清零。这与主控模式下 SPITBF 位的功能类似。
2. 如果 SSEN (SPIxCON<7>) 置位，SPITBF 将在用户代码载入 SPIxBUF 时置位。但是，它只有在 SPIx 模块完成数据发送后才会清零。当 $\overline{\text{SSx}}$ 引脚变为高电平时，发送将被中止并可能在一段时间以后重试。每个数据字都保存在 SPIxTXB 中，直到所有的位都被发送到接收器为止。

注：要符合模块的时序要求，当 CKE = 1 时，在从动模式下必须使能 $\overline{\text{SSx}}$ 引脚（详情请参见图 20-6）。

图 20-4: SPI 从动模式工作原理：禁止从动选择引脚

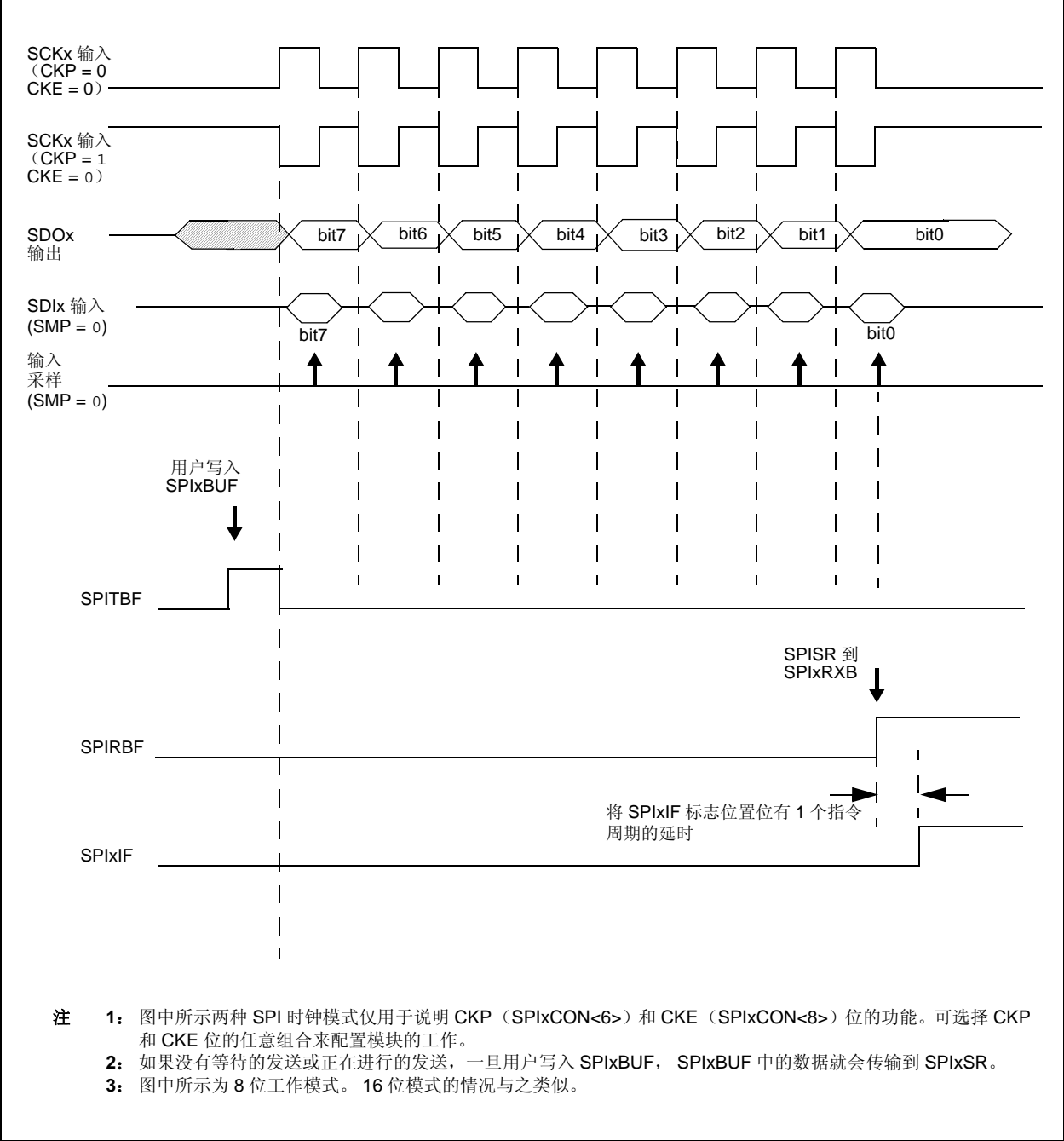


图 20-5: 从动选择引脚使能时的 SPI 从动模式工作原理

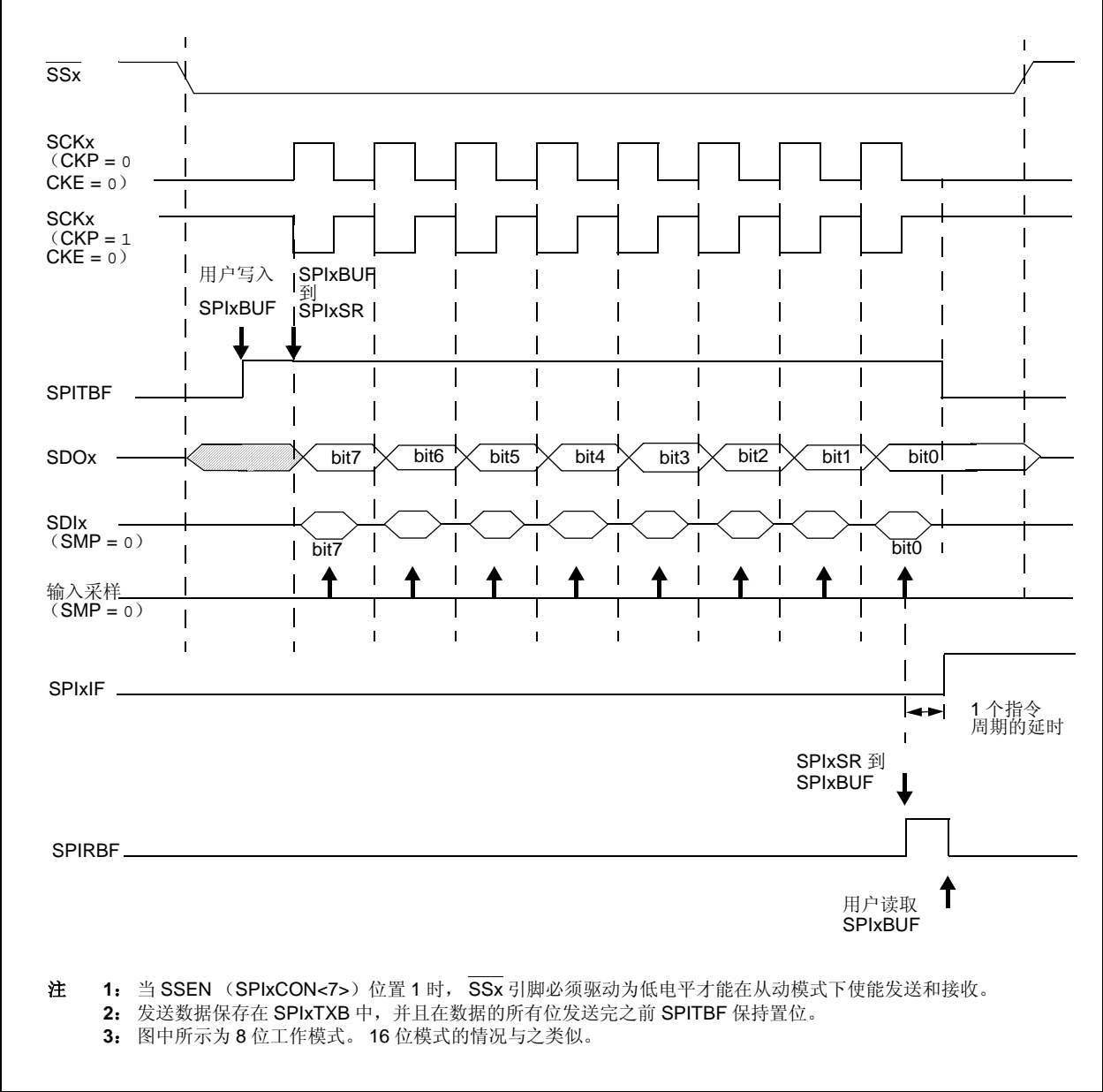
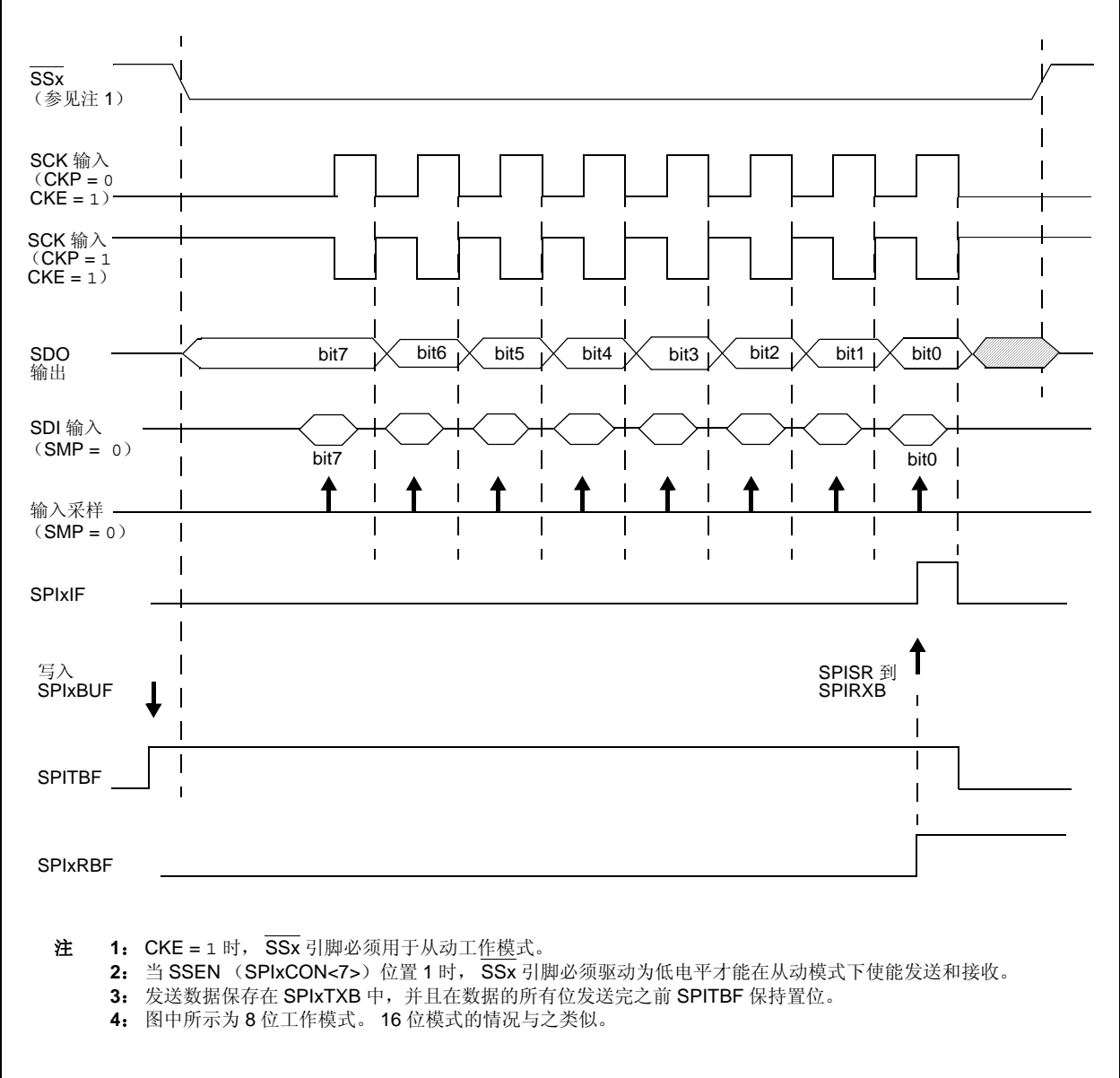


图 20-6: SPI 模式时序 (从动模式且 CKE = 1)



20.3.3 SPI 错误处理

当新的数据字移入 SPIxSR 并且用户软件未读取 SPIxRXB 的原先内容时, SPIROV 位 (SPIxSTAT<6>) 将置位。模块不会将接收到的数据从 SPIxSR 传输到 SPIxRXB。在 SPIROV 位清零前, 禁止接收后续数据。SPIROV 位不会被模块自动清零, 它必须由用户软件清零。

20.3.4 SPI 仅启用接收功能的工作原理

将控制位 DISSDO (SPIxCON<11>) 置位, 禁止 SDOx 引脚的发送功能。这样可以使 SPIx 模块配置为仅接收的工作模式。如果 DISSDO 位置位, SDOx 引脚将由相应端口功能控制。

DISSDO 功能适用于所有 SPI 工作模式。

20.3.5 帧 SPI 模式

当工作在主控模式或从动模式时, 模块支持一个非常基本的帧 SPI 协议。在 SPI 模块中提供了以下功能来支持帧 SPI 模式:

- 控制位 FRMEN (SPIxCON<14>) 可启用帧 SPI 模式并使 \overline{SSx} 引脚作为帧同步脉冲输入或输出引脚使用。SSEN (SPIxCON<7>) 的状态会被忽略。
- 控制位 SPIFSD (SPIxCON<13>) 决定 \overline{SSx} 引脚的输入 / 输出方向 (即模块是接收还是产生帧同步脉冲)。
- 帧同步脉冲在一个 SPI 时钟周期内为高电平有效脉冲。

SPI 模块支持以下两种帧 SPI 模式:

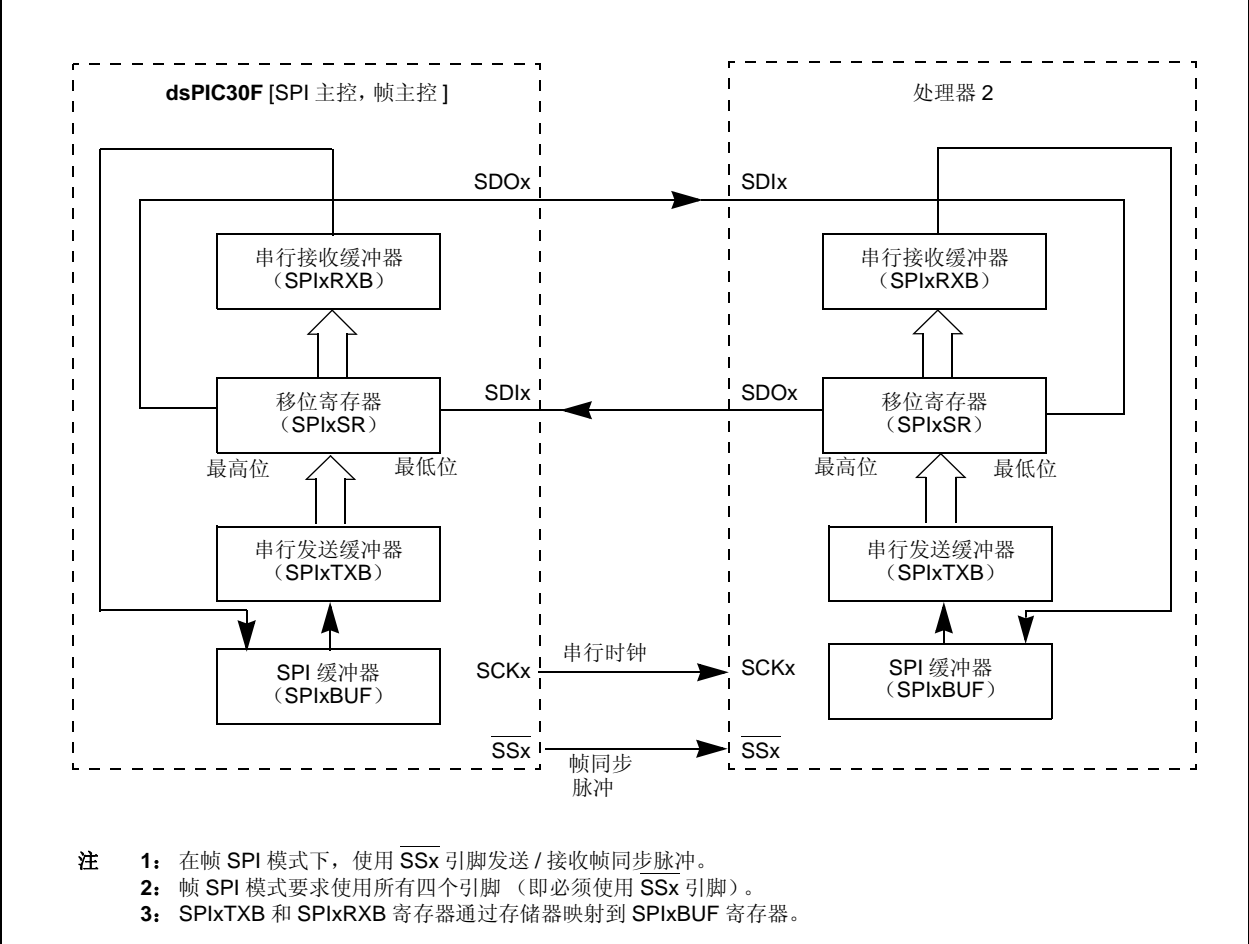
- 帧主控模式: SPI 模块产生帧同步脉冲并在 \overline{SSx} 引脚为其他器件提供此脉冲。
- 帧从动模式: SPI 模块使用在 \overline{SSx} 引脚接收到的帧同步脉冲。

主控模式和从动模式都支持帧 SPI 模式。因此, 用户可以使用以下四种帧 SPI 配置:

- SPI 主控模式和帧主控模式
- SPI 主控模式和帧从动模式
- SPI 从动模式和帧主控模式
- SPI 从动模式和帧从动模式

这四种模式决定 SPIx 模块是否产生串行时钟和帧同步脉冲。

图 20-7: SPI 主控模式、帧主控模式框图



20.3.5.1 在帧 SPI 模式下的 SCKx

当 $FRMEN$ ($SPIxCON<14>$) = 1 且 $MSTEN$ ($SPIxCON<5>$) = 1 时， $SCKx$ 引脚为输出引脚，且 $SCKx$ 上的 SPI 时钟成为自由运行时钟。

当 $FRMEN$ = 1 且 $MSTEN$ = 0 时， $SCKx$ 引脚成为输入引脚。假设提供给 $SCKx$ 引脚的源时钟信号是自由运行时钟信号。

时钟的极性由 CKP ($SPIxCON<6>$) 位选择。 CKE ($SPIxCON<8>$) 位在分帧 SPI 模式下未使用，应该在用户软件中编程为 0。

当 CKP = 0 时，帧同步脉冲输出和 $SDOx$ 数据输出在 $SCKx$ 引脚的时钟脉冲上升沿变化。在串行时钟的下降沿，在 $SDIx$ 输入引脚上采样输入数据。

当 CKP = 1 时，帧同步脉冲输出和 $SDOx$ 数据输出在 $SCKx$ 引脚的时钟脉冲下降沿变化。在串行时钟的上升沿，在 $SDIx$ 输入引脚上采样输入数据。

20.3.5.2 在帧 SPI 模式下的 SPIx 缓冲器

当 SPIFSD (SPIxCON<13>) = 0 时, SPIx 模块处于帧主控工作模式。在此模式下, 当用户软件将发送数据写入 SPIxBUF 地址单元 (从而可将发送数据写入 SPIxTXB 寄存器) 时, 模块启动帧同步脉冲。在帧同步脉冲的末尾, SPIxTXB 的数据被传输到 SPIxSR, 同时开始发送/接收数据。

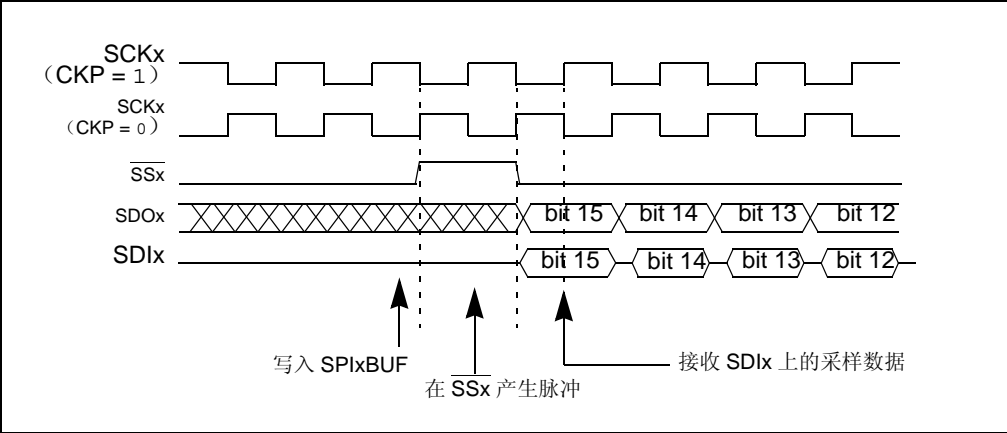
当 SPIFSD (SPIxCON<13>) = 1 时, 模块处于帧从动模式。在此模式下, 帧同步脉冲由外部时钟源提供。当模块采样帧同步脉冲时, 它将把 SPIxTXB 寄存器的内容传输到 SPIxSR, 同时开始发送/接收数据。在接收到帧同步脉冲前, 用户必须确保在 SPIxBUF 中装入了要发送的正确数据。

注: 无论数据是否写入 SPIxBUF, 接收到帧同步脉冲的同时都将开始发送。如果新数据尚未写入, 将发送 SPIxTXB 的原有数据。

20.3.5.3 SPI 主控模式和帧主控模式

通过将 MSTEN (SPIxCON<5>) 和 FRMEN (SPIxCON<14>) 位置 1 并将 SPIFSD (SPIxCON<13>) 位清零可启用此帧 SPI 模式。在此模式下, 无论模块是否正在发送, 串行时钟都将在 SCKx 引脚连续输出。当写入 SPIxBUF 时, SSx 引脚将在 SCKx 时钟的下一个发送沿驱动为高电平。SSx 引脚在一个 SCKx 时钟周期内将为高电平。如图 20-8 所示, 模块将在 SCKx 的下一个发送沿开始发送数据。图 20-7 所示为表明此工作模式的信号方向的连接图。

图 20-8: SPI 主控模式、帧主控模式



20.3.5.4 SPI 主控模式和帧从动模式

通过将 MSTEN、FRMEN 和 SPIFSD 位置 1 可启用此帧 SPI 模式。 $\overline{\text{SSx}}$ 引脚为输入引脚，并在 SPI 时钟的采样沿对其进行采样。如图 20-9 所示，当采样到高电平时，在紧接着的 SPI 时钟发送沿就会发送数据。当发送完成时中断标志 SPIxIF 将置位。在 $\overline{\text{SSx}}$ 引脚接收到信号前，用户必须确保在 SPIxBUF 中装入了正确的待发送数据。图 20-10 所示为表示此工作模式的信号方向的连接图。

图 20-9: SPI 主控模式，帧从动模式

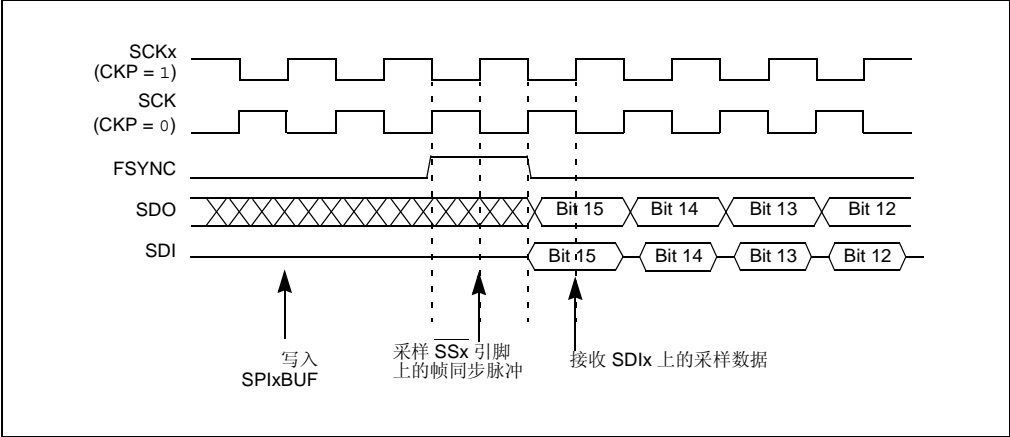
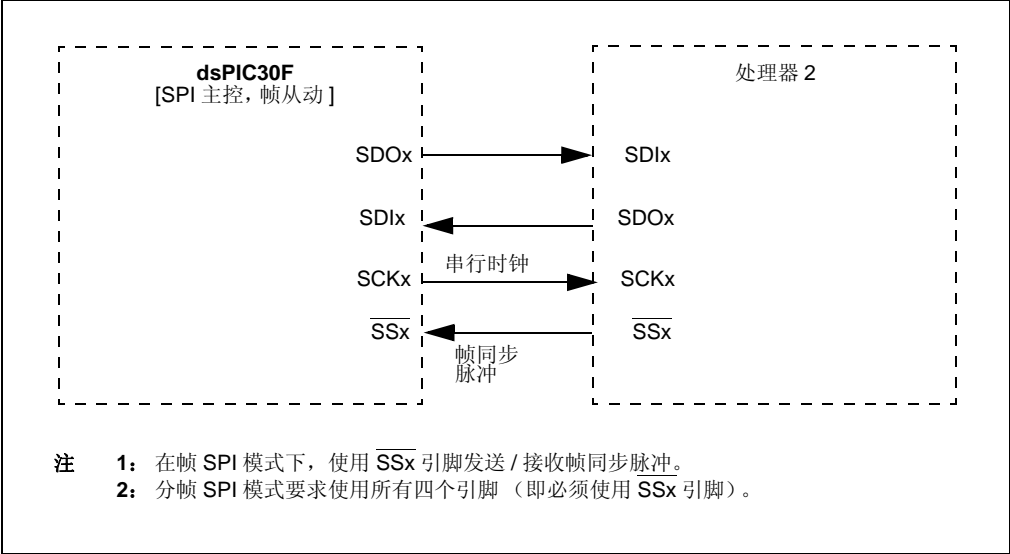


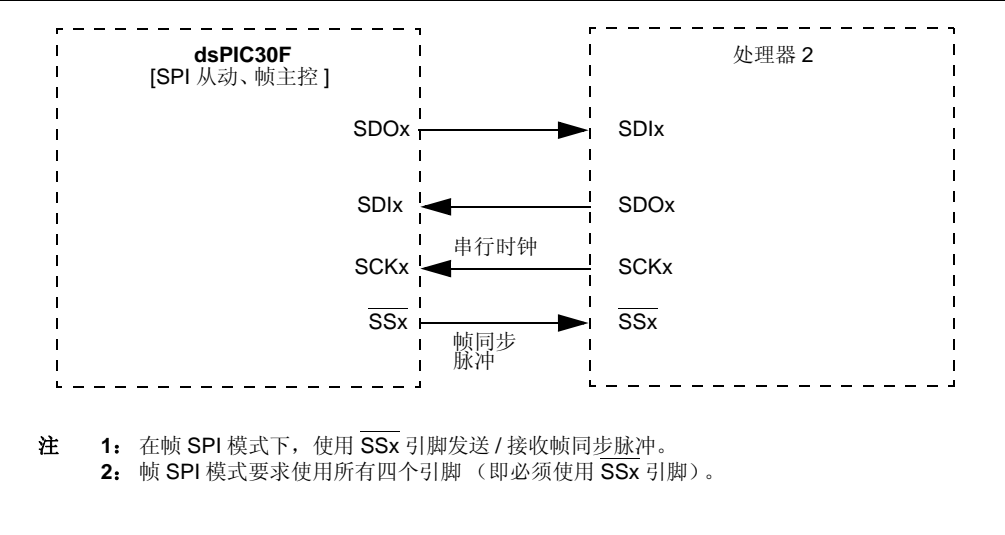
图 20-10: SPI 主控模式、帧从动模式连接图



20.3.5.5 SPI 从动模式和帧主控模式

通过将 MSTEN (SPIxCON<5>) 位清零、FRMEN (SPIxCON<14>) 位置 1 以及 SPIFSD (SPIxCON<13>) 位清零, 可使能此帧 SPI 模式。在从动模式下, 将继续使用输入 SPI 时钟。当 SPIFSD 位为低时, SSx 引脚是输出引脚。因此, 当写入 SPIBUF 时, 模块将在 SPI 时钟的下一个发送沿把 SSx 引脚驱动为高电平。SSx 引脚在一个 SPI 时钟周期内将保持驱动为高电平。将在下一个 SPI 时钟发送沿开始发送数据。图 20-11 所示为表示此工作模式的信号方向的连接图。

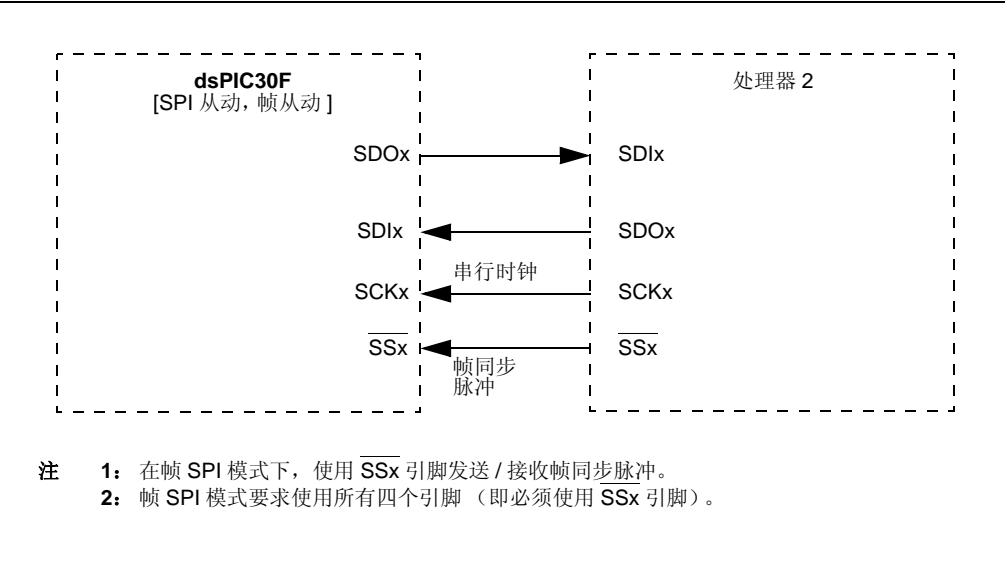
图 20-11: SPI 从动模式、帧主控模式连接图



20.3.5.6 SPI 从动模式和帧从动模式

通过将 MSTEN (SPIxCON<5>) 位清零、FRMEN (SPIxCON<14>) 位置 1 以及 SPIFSD (SPIxCON<13>) 位置 1, 可使能此帧 SPI 模式。因此, SCKx 和 SSx 引脚都将是输入引脚。将在 SPI 时钟的采样沿采样 SSx 引脚。当采样到 SSx 引脚上为高电平时, 将在下一个 SCKx 发送沿发送数据。图 20-12 所示为表示此工作模式的信号方向的连接图。

图 20-12: SPI 从动模式、帧从动模式连接图



20.4 SPI 主控模式时钟频率

在 主控模式下，提供给 SPI 模块的时钟周期就是指令周期（Tcy）。然后将此时钟信号由主预分频器（由 PPRE<1:0>（SPIxCON<1:0>）指定）和辅助预分频器（由 SPRE<2:0>（SPIxCON<4:2>）指定）预分频。经过预分频的指令时钟就变为串行时钟并通过 SCKx 引脚提供给外部器件。

注： 注意 SCKx 信号时钟在正常 SPI 模式下不是自由运行的。它仅在 SPIxBUF 加载了数据后运行 8 或 16 个脉冲时间。但是在分帧模式下，它会连续运行。

公式 20-1 可用于计算 SCKx 时钟频率主预分频器和辅助预分频器设置值的函数。

公式 20-1:

$$F_{SCK} = \frac{F_{CY}}{\text{主预分频比} \times \text{辅助预分频比}}$$

下表列出了部分范例 SPI 时钟频率（单位为 kHz）：

表 20-1: 范例 SCKx 频率

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： 表中 SCKx 频率的单位为 kHz。

注： 并不支持所有的时钟速率。如需更多信息，参见特定器件数据手册的 SPI 时序规范。

20.5 低功耗模式下的工作

dsPIC30FXXXX 系列器件具有三种能耗模式：

- 工作模式：内核与外设均处于运行状态。
- 低功耗模式：通过执行 PWRSAV 指令可进入该模式。dsPIC30F 系列器件支持两种低功耗模式。在 PWRSAV 指令中可以通过参数来指定具体的模式。这两种模式为：

- 休眠模式：器件时钟源和整个器件都关闭。可通过以下指令实现。

```
;include device p30fxxxx.inc file  
PWRSAV #SLEEP_MODE
```

- 空闲模式：器件时钟处于工作状态，CPU 和所选外设关闭。

```
;include device p30fxxxx.inc file  
PWRSAV #IDLE_MODE
```

20.5.1 休眠模式

当器件进入休眠模式后，系统时钟会被禁止。

20.5.1.1 主控模式下的工作

以下为将 SPIx 模块配置为主控工作模式时进入休眠模式的后果：

- SPIx 模块的波特率发生器停止并复位。
- 如果 SPIx 模块在发送 / 接收的过程中进入休眠模式，则发送 / 接收将被中止。因为在发送或接收未完成时没有自动的方式能防止 SPIx 模块进入休眠模式，因此用户软件必须将进入休眠与 SPI 模块工作同步以防止传输中止。
- 在休眠模式下发送器和接收器将停止工作。发送器或接收器在被唤醒后不会继续尚未完成的传输。

20.5.1.2 从动模式下的工作

因为在从动模式下，SCKx 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。它将在进入到休眠的过渡时间内完成所有事务处理（Transaction）。完成事务处理后，SPIRBF 标志将置位。从而将 SPIxIF 位置位。如果允许 SPI 中断（SPIxIE = 1），器件将从休眠模式唤醒。如果 SPI 中断的优先级高于当前 CPU 优先级，将从 SPIx 中断向量地址处恢复代码执行。否则，将继续执行在进入休眠模式之前执行的 PWRSAV 指令后的代码。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 SPIx 模块进入或退出休眠模式时，寄存器内容不受影响。

20.5.2 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（SPIxSTAT<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，SPI 模块将在进入空闲模式时停止通信。其工作状态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。

表 20-2: 与 SPI 模块相关的引脚

引脚名称	引脚类型	缓冲器类型	说明
SCK1	I/O	CMOS	SPI1 模块时钟输入或输出
SCK2	I/O	CMOS	SPI2 模块时钟输入或输出
SDI1	I	CMOS	SPI1 模块数据接收引脚
SDI2	I	CMOS	SPI2 模块数据接收引脚
SDO1	O	CMOS	SPI1 模块数据发送引脚
SDO2	O	CMOS	SPI2 模块数据发送引脚
SS1	I/O	CMOS	SPI1 模块从动选择控制引脚 1) 置位 SSEN（SPI1CON<7> 时，用于在从动模式下使能发送 / 接收 2) 在 FRMEN 和 SPIFSD（SPI1CON<14:13>）置为 11 或 10 时，作为帧同步 I/O 脉冲
SS2	I/O	CMOS	SPI2 模块从动选择控制引脚 1) 置位 SSEN（SPI2CON<7>）时，用于在从动模式下使能发送 / 接收 2) 在 FRMEN 和 SPIFSD（SPI2CON<14:13>）置为 11 或 10 时，作为帧同步 I/O 脉冲

图注： CMOS = CMOS 兼容的输入或输出， ST = CMOS 电平的施密特触发输入，
I = 输入， O = 输出

20.6 与 SPI 模块相关的特殊功能寄存器

表 20-3: 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	复位状态
SPI1STAT	0220	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI1CON	0222	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI1BUF	0224	SPI1TXB 和 SPI1RXB 寄存器共享的发送和接收缓冲器地址单元																0000 0000 0000 0000

表 20-4: SPI2 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	复位状态
SPI2STAT	0226	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI2CON	0228	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI2BUF	022A	SPI2TXB 和 SPI2RXB 寄存器共享的发送和接收缓冲器地址单元																0000 0000 0000 0000

表 20-5: SPI 模块相关的中断寄存器

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	复位状态
INTCON1	0080	NSTDIS	—	—	—	—	OVATE	OVBT	COVTE	—	—	—	SWTRAP	OVRFLOW	ADDRERR	STKERR	—	0000 0000 0000 0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	LEV8F	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000 0000 0000 0000
IFS0	0084	CNIF	MI2CIF	SI2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0	0000 0000 0000 0000
IFS1	0086	IC6IF	IC5IF	IC4IF	IC3IF	C1IF	SPI2IF	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	IC8IF	IC7IF	INT1IF	0000 0000 0000 0000
IEC0	008C	CNIE	MI2CIE	SI2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IEC1	008E	IC6IE	IC5IE	IC4IE	IC3IE	C1IE	SPI2IE	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	IC8IE	IC7IE	INT1IE	0000 0000 0000 0000
IPC2	0098	—	ADIP<2:0>			—	U1TXIP<2:0>			—	U1RXIP<2:0>			—	SPI1IP<2:0>			0100 0100 0100 0100
IPC6	00A0	—	C1IP<2:0>			—	SPI2IP<2:0>			—	U2TXIP<2:0>			—	U2RXIP<2:0>			0100 0100 0100 0100

20.7 相关应用笔记

本节列出了与手册的本章内容相关的应用笔记。这些应用笔记可能并不是专为 dsPIC30F 系列产品编写的，但是概念是相关的，通过适当修改即可使用，但使用中可能会受到一定限制。当前与串行外设接口（SPI）模块相关的应用笔记有：

标题	应用笔记编号
Interfacing Microchip's MCP41XXX/MCP42XXX Digital Potentiometers to a PICmicro® Microcontroller	AN746
Interfacing Microchip's MCP3201 Analog-to-Digital Converter to the PICmicro® Microcontroller	AN719

注： 如需获取更多 dsPIC30F 系列器件的应用笔记和代码示例，请访问 Microchip 网站（www.microchip.com）。

20.8 版本历史

版本 A

这是本文档的初始发行版。

版本 B

此版本反映了 dsPIC30F 串行外设接口（SPI）模块的编辑修订及技术内容修改。

版本 C

本章未包含任何技术内容修订，然而我们谨更新此节以反映本手册已更新到版本 C。