## SPI协议

### 1 介绍

#### 1.1 概述

SPI协议是一种双工、同步、串行的通信协议。软件可以通过轮询或中断方式查询SPI设备状态。

#### 1.2 SPI协议特性

* 具有主模式和从模式；
* 双工通信；
* 从选择输出；
* 模式错误中断；
* 双缓冲数据寄存器；
* 可编程的串行时钟；
* 在等待状态下控制SPI；

#### 1.3 工作状态

SPI具有三种工作状态：

* 运行状态：SPI基本工作状态；
* 等待模式：一种可配置的低功耗状态；由SPICR2寄存器中的SPISWAI位控制。在等待状态下，如果SPISWAI位设置为0，则SPI的操作方式类似于运行状态；如果设置了SPISWAI位，SPI将进入低功率状态并关闭SPI时钟生成。如果SPI被配置为主模式，任何正在进行的传输都会停止，但在进入运行状态后会恢复。如果SPI被配置为从模式，则数据的接收和传输将继续进行，以便保持数据同步；
* 停止状态：为了降低功耗，SPI在停止状态下处于非活动状态。如果SPI被配置为主模式，任何正在进行的传输都会停止，但在CPU进入运行模式后会恢复。如果SPI被配置为从模式，数据的接收和传输继续进行，以便从保持数据同步；

### 2 SPI信号

#### 2.1 MOSI

当SPI模块被配置为主模式时，该引脚用于将数据传出；被配置为从模式时，该引脚用于接收数据。

#### 2.2 MISO

当SPI模块被配置为从模式时，该引脚用于将数据传出；当被配置为主模式时，该引脚用于接收数据。

#### 2.3 SS

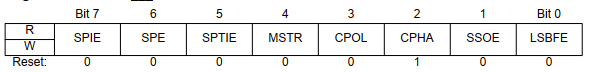
该引脚用于从输出选择信号到另一个外设；当配置为主模式时，数据传输将发生；配置为从模式时，接收该选择信号。

#### 2.4 SCK

此引脚用于输出或接收时钟信号(从模块）。

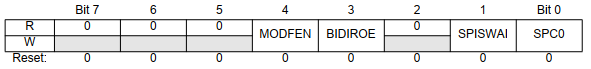
### 3 寄存器

#### 3.1 SPI控制寄存器1



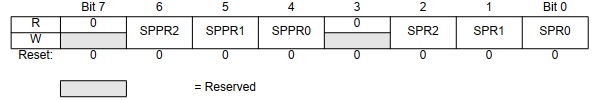
* SPIE：中断使能位；如果设置了SPIF或MODF状态标志，此位启用SPI中断请求；
* SPE：SPI模块使能位；使能SPI模块并将SPI端口引脚专用于SPI；设置为0时SPI被禁用并强制进入空闲状态，SPISR寄存器中的状态位将被重置
* SPTIE：SPI传输中断使能；如果设置了SPTEF标志，该位启用SPI中断请求；
* MSTR：SPI主从模式选择位；设置为1为主模式；
* CPOL：SPI时钟极性位；该位选择反转或非反转时钟；进行传输数据的SPI模块之间的CPOL值必须相同；在主模式下，改变改位将中止正在进行的传输并使模块系统进入空闲状态；
* CPHA：选择SPI时钟采样模式；在主模式下，改变该位将中止正在进行的传输并迫使SPI系统进入空闲状态；
* SSOE：使能SS信号输出；如果设置了MODFEN，则只有在主模式下才启用SS输出特性；在主模式下，改变该位将中止正在进行的传输并迫使SPI模块进入空闲状态；
* LSBFE：该位不影响MSB和LSB在数据寄存器中的位置；数据寄存器的读写操作总是将MSB放在第7位；在主模式下，改变该位将中止正在进行的传输并使SPI模块进入空闲状态；

#### 3.2 SPI控制寄存器2



* MODFEN：使能MODF错误检测；如果SPI处于主模式并且MODFEN被清除，则SPI模块不使用SS引脚；在从模式下，不管MODFEN的值如何，SS只能作为输入；
* BIDIROE：在双向工作模式下，该位控制SPI的MOSI和MISO输出缓冲区；在主模式下，这个位控制MOSI端口的输出缓冲区；在从模式下，控制MISO端口的输出缓冲区。在设置了SPC0的主模式下，改变该位将中止正在进行的传输并使SPI模块进入空闲状态；
* SPISWAI：在等待模式下控制时钟生成；
* SPC0：位使能双向引脚配置；在主模式下，改变该位将中止正在进行的传输并迫使SPI系统进入空闲状态；

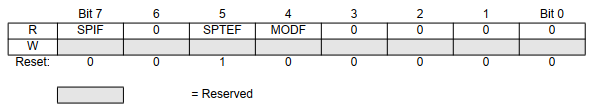
#### 3.3 SPI波特率寄存器



* SPPR2-SPPR0：SPI波特率预选位；
* SPR2-SPR0：SPI波特率选择位；

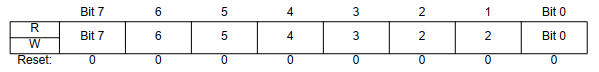
SPI时钟分频系数：

#### 3.4 SPI状态寄存器



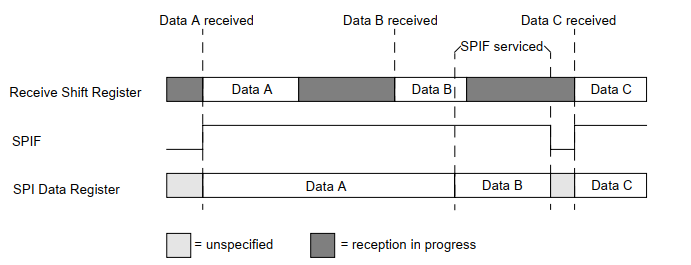
* SPIF：SPI中断标志位；在接收的数据已传输到SPI数据寄存器之后设置为1；在读取数据寄存器后读取状态寄存器清除该位；
* SPTEF：SPI传输空中断标志位；设置为1则表示数据寄存器为空；将数据放入传输数据寄存器然后读取SPISR会清除该位，否则不会被清零；
* MODF：模式错误标志位；SPI配置为主模式时，如果SS输入为低电平且MODFEN为1则设置此位；

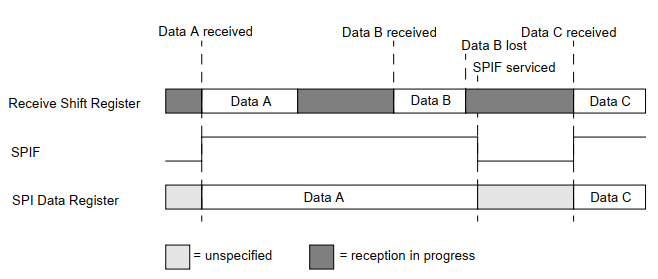
#### 3.5 SPI数据寄存器



SPI数据寄存器是SPI数据的输入和输出寄存器。对这个寄存器的写入使得一个字节数据排队和传输。当SPI模块配置为主模式，在前一个传输完成后立即传输一个排队的数据字节。SPISR寄存器中SPTEF标志位指示SPI数据寄存器可以准备接受新数据。

当设置SPIF时，SPIDR中接收的数据是有效的。如果SPIF被清除并且已经接收到一个字节数据，则接收到的字节将从接收移位寄存器传输到SPIDR并设置SPIF。如果设置了SPIF且未读取SPIDR，在接收到第二个字节时，则将数据保存在移位寄存器中，直到接收到下一个数据，但是SPIDR中的数据不会改变。





SPIF被设置且未读取SPIDR，接收移位寄存器中有一个有效的数据；在第三次传输开始之前，SPIDR被读取，则接收移位寄存器中的字节被传输到SPIDR中，SPIF保持设置。如果设置了SPIF并且接收移位寄存器中有一个有效的字节，并且SPIF在第三次传输开始后被服务，则接收移位寄存器中的数据被覆盖，并且没有被传输到SPIDR中。

### 4 功能描述

#### 4.1 概述

SPI模块允许在MCU和外围设备之间进行双工、同步、串行通信。软件可以轮询SPI状态或者中断驱动。SPI模块通过在SPI控制寄存器1中设置SPI使能位(SPE)来使能。

SPI模块的主要组成部分是SPI数据寄存器。在master和slave中的8位数据寄存器通过MOSI和MISO引脚连接，形成一个分布式16位寄存器。当执行数据传输操作时，这个16位寄存器被串行移位8位，数据在master和slave之间交换。写入master的SPI数据寄存器成为slave的输入数据，从master读取SPI数据寄存器是slave的输出数据。

当SPTEF设置为1时，写入SPIDR将数据放入传输数据寄存器。当传输完成并且SPIF被清除时，接收的数据被移动到接收数据寄存器中。这8位数据寄存器作为SPI接收数据寄存器用于读取，作为SPI传输数据寄存器用于写入。

SPI控制寄存器1中的时钟相位控制位(CPHA)和时钟极性控制位(CPOL)从SPI系统使用的四种时钟格式中选择一种。CPOL位选择一个非倒置时钟或倒置时钟。CPHA位通过在SCK奇数边或偶数边采样信号来适应两种根本不同的协议。

#### 4.2 主模式

当设置MSTR位时，SPI模块以主模式运行。只有主模式才可以发起数据传输。传输从写入主SPI数据寄存器开始。如果移位寄存器为空，则字节立即转移到移位寄存器。在串行时钟的控制下，字节开始在MOSI引脚上移出。

SPI波特率寄存器中的SPPR2、SPPR1和SPPR0波特率选择位与SPPR2、SPPR1和SPPR0波特率预选位一起控制波特率发生器并确定传输速度。SCK引脚是SPI时钟输出引脚。

在主模式下，串行数据输出引脚(MOSI)和串行数据输入引脚(MISO)的功能由SPC0和BIDIROE控制位决定。

如果设置了MODFEN和SSOE位，则SS引脚配置为从选择输出。在每次传输过程中，SS设置为低电平，SPI处于空闲状态时输出高电平。

如果设置了MODFEN，并且清除了SSOE，则将SS引脚配置为检测模式错误的输入。如果SS输入变低，这表明一个模式错误，另一个master试图驱动MOSI和SCK线。在这种情况下，SPI通过清除MSTR位并禁用从输出缓冲区MISO(或双向模式下的SISO)，立即切换到从模式。因此，结果是所有输出都被禁用，而SCK、MOSI和MISO是输入。当模式错误发生时，如果传输正在进行，传输将被中止，SPI将被迫进入空闲状态。

模式错误还在SPI状态寄存器(SPISR)中设置模式错误(MODF)标志。如果在设置MODF标志时设置了SPI中断使能位(SPI)，那么也会请求一个SPI中断序列。

当写入master的SPI数据寄存器时，具有半个SCK周期延迟。延迟之后，SCK在master内启动。其余的传输操作略有不同，这取决于SPI控制寄存器1中由CPHA指定的时钟格式。

#### 4.3 从模式

当MSTR位为0时，SPI模块以从模式工作。在从模式下，SCK是来自master。在从模式下，串行数据输出引脚(MISO)和串行数据输入引脚(MOSI)的功能由SPI控制寄存器2中的SPC0位和BIDIROE位决定。

SS引脚是从选择输入。在数据传输之前，从SPI的SS引脚必须低。SS必须保持在低水平，直到传输完成。如果SS为高电平，SPI将被迫进入空闲状态。

SS输入还控制串行数据输出引脚，如果SS为高电平(未选中)，则串行数据输出引脚为高阻抗；如果SS低，则SPI数据寄存器中的第一位将从串行数据输出引脚中驱动出来。同样，如果从机没有被选择(SS为高电平)，那么SCK输入将被忽略，SPI移位寄存器不会发生内部移位。

只要不超过一个slave驱动从设备的串行数据输出线，几个从设备就有可能从一个主设备接收到相同的数据传输，尽管主设备不会从所有slave接收到返回信息。

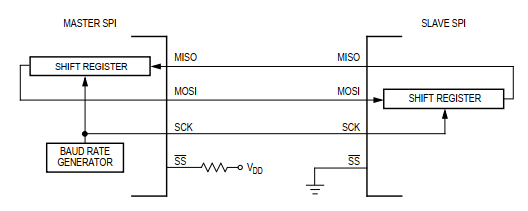
如果SPI控制寄存器1中的CPHA位被清除，则SCK输入上的奇数边会导致串行数据输入引脚处的数据被锁存。偶数边导致先前从串行数据输入引脚锁存的值移位到SPI移位寄存器的LSB或MSB。

如果设置了CPHA位，则SCK输入端的偶数边会导致串行数据输入引脚处的数据被锁存。奇数边导致先前从串行数据输入引脚锁存的值移位到SPI移位寄存器的LSB或MSB。

当设置CPHA时，第一个边用于将第一个数据位放到串行数据输出引脚上。当CPHA被清除的并且SS输入为低电平(从选择)，SPI数据的第一个比特被驱动到串行数据输出引脚。在第8个移位之后，传输完成，接收到的数据被传输到SPI数据寄存器中并设置SPIF标志。

#### 4.4 传输形式

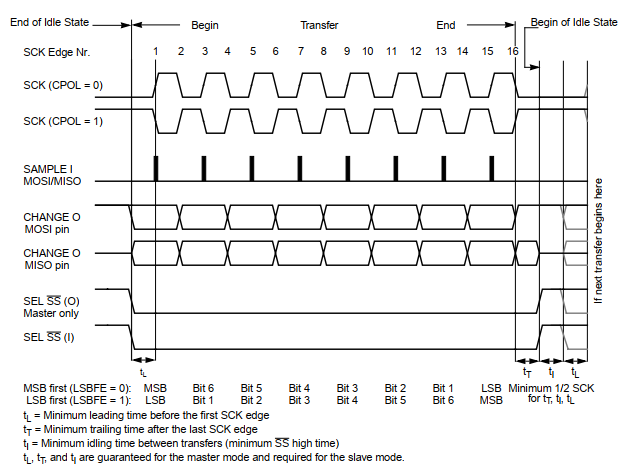
在SPI传输过程中，数据同时传输(串行移出)和接收(串行移出)。串行时钟(SCK)同步两条串行数据线上信息的移位和采样。从选择线允许选择单个slave，未选择的slave不会干扰SPI总线活动。



##### 4.4.1 传输时钟

CPOL位指定一个高时钟或低时钟，对传输格式没有显著影响。CPHA位选择两种根本不同的传输格式。master和slave的时钟相位和极性应该相同。在某些情况下，在传输之间改变相位和极性，以允许主设备与具有不同外围设备通信。

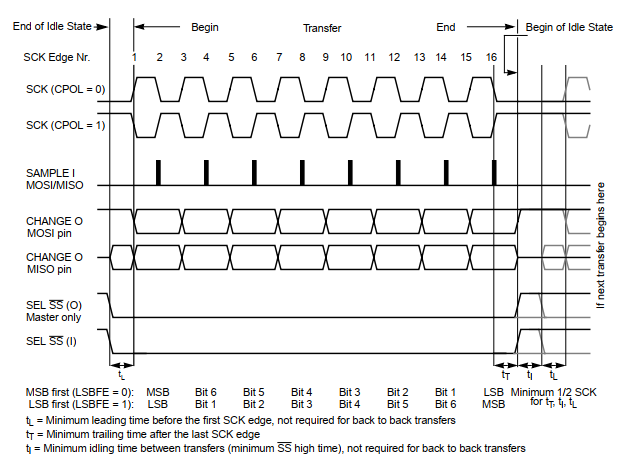
##### 4.4.2 CPHA设置为0



SCK的第一个边沿用于采样第一个数据位。在一些外设中，一旦slave被选中，从机数据的第一个比特就在从机的数据输出引脚处可用。在这种格式中，第一个SCK边在SS变低后的半个周期内发出。半个SCK周期后，第二个时钟沿出现在SCK信号线上。当这第二个时钟沿发生时，先前从串行数据输入引脚锁存的值将移位到移位寄存器的LSB或MSB。

在这第二个时钟边沿之后，SPI数据的下一位从串行数据输出引脚传输。这个过程持续到SCK线上总共16条边，数据在奇数边上被锁存，在偶数边上被移位。数据接收是双重缓冲的。数据在传输过程中被连续移位到SPI移位寄存器中，并在最后一位移位后被转移到SPI数据寄存器中。

##### 4.4.3 CPHA设置为1



一些外设需要第一个SCK边在第一个数据位在数据输出引脚可用之前，第二个时钟沿数据进行传输。在这种格式中，第一个SCK边沿是通过在8周期传输操作开始时设置CPHA位来发出的。

SCK的第一个边发生在半个SCK时钟周期同步延迟之后。第一个边沿slave将第一个数据位传输到master的串行数据输入引脚。半个SCK周期后，第二个时钟沿出现在SCK引脚上。这是master和slave的锁存边。当第三条边出现时，先前从串行数据输入引脚锁存的值将移位到SPI移位寄存器的LSB或MSB。在这条边之后，master数据的下一位从串行数据输出引脚输出。这个过程持续到SCK线上总共16个时钟沿，数据在偶数沿被锁存，移位发生在奇数沿上。数据接收是双缓冲的，数据在传输过程中被连续移位到SPI移位寄存器，在最后一位移位后被转移到SPI数据寄存器。

#### 4.5 SPI波特率

在SPI波特率寄存器(SPPR2，SPPR1、SPPR0、SPR2、SPR1和SPR0)确定SPI模块时钟的分频系数，该系数产生SPI波特率。SPI时钟频率是由SPPR (SPPR2-SPPR0)和SPR (SPR2-SPR0)共同决定。当所有位都为零(默认情况)时，SPI模块时钟被2除。

#### 4.6 特性

##### 4.6.1 SS输出

在数据传输时将SS引脚设置为低电平以选择外部设备，在空闲时驱动SS引脚为高电平以取消选择外部设备。SS输出引脚连接到外部设备的SS输入引脚。在正常的SPI操作中，SS输出仅在主模式下可用。

##### 4.6.2 引脚双向通信

当在SPI控制寄存器2中设置SPC0位时，选择双向模式(正常模式和双向模式)。在这种模式下，SPI仅使用一个串行数据引脚用于与外部设备进行通信。MSTR位决定使用哪个引脚。MOSI引脚成为主模式的串行数据I/O (MOMI)引脚，MISO引脚成为从模式的串行数据I/O (SISO)引脚。此时主模式下的MISO引脚和从模式下的MOSI引脚不被SPI使用。

每个串行I/O引脚的方向取决于BIDIROE位。如果引脚配置为输出，则移位寄存器中的串行数据将在引脚上输出。同一引脚也是移位寄存器的串行输入。SCK是主模式的输出，从模式的输入。SS是主模式的输入或输出，总是从模式的输入。双向模式不影响SCK和SS功能。

#### 4.7 错误

如果在SPI配置为主模式时SS输入为低电平，则表明系统错误。可能有多个master同时驱动MOSI和SCK线路。

在特殊情况下，SPI处于主模式，并且MODFEN位被清除，SPI模块不使用SS引脚。模式错误检查被抑制，MODF仍然被清除。如果SPI配置为从模式，则SS引脚是专用的输入引脚。模式错误在从模式下不会发生。

如果发生模式故障错误，SPI将切换到从模式，除非从输出缓冲区被禁用。SCK, MISO和MOSI引脚成为高阻抗输入，以避免与另一个输出驱动冲突。正在进行的传输被中止，SPI被迫进入空闲状态。

如果在主模式下配置的SPI模块在双向引脚下发生模式错误异常，则MOMI(双向引脚的MOSI)输出使能将被清除。从模式配置的SPI模块在双向引脚下没有模式错误。

#### 4.8 低功耗

##### 4.8.1 运行状态

在运行状态下，SPI系统使能(SPE)位在SPI控制寄存器清除，SPI系统处于低功耗的禁用状态。SPI寄存器仍然可以被访问，但该模块核心的时钟被禁用。

##### 4.8.2 等待状态

等待模式下的SPI操作取决于SPI控制寄存器2中SPISWAI位：

* 如果SPISWAI设置为0，表示处于等待模式时SPI工作正常；
* 如果设置了SPISWAI，当处于等待模式时，SPI产生时钟停止，SPI模块进入省电状态；

##### 4.8.3 停止状态

停止模式取决于系统。当模块时钟被禁用(保持高或低)时，SPI进入停止状态。如果SPI处于主模式并在进入停止模式时交换数据，则传输将被停止，直到退出停止状态。在从模式下，SPI模块将与master保持同步。停止模式不依赖于SPISWAI位。

##### 4.8.4 重启状态

* 如果数据传输发生在重置后的从模式并且没有写入SPIDR，将传输垃圾数据或在重置前从master接收到的最后一个字节数据；
* 在重置后从SPIDR读取将始终读取到零；

##### 4.8.5 中断状态

SPI仅在启用SPI时(SPE设置为1)才发起中断请求。

当master检测到SS引脚上的错误时，就会发生MODF。主模式的SPI模块必须配置MODF特性。一旦设置MODF，当前传输将被停止。发生MODF中断时状态寄存器MODF标志设置为1。清除标志也将清除中断。当设置MODF时，这个中断将保持活动状态。

当接收到新数据并将其复制到SPI数据寄存器时会发生SPIF中断。一旦设置了SPIF，直到读取数据后才会清除。

SPTEF发生在SPI数据寄存器准备接受新数据时。