

SPI 总线协议及 SPI 时序图详解

SPI，是英语 Serial Peripheral Interface 的缩写，顾名思义就是串行外围设备接口。SPI，是一种高速的，全双工，同步的通信总线，并且在芯片的管脚上只占用四根线，节约了芯片的管脚，同时为 PCB 的布局上节省空间，提供方便，正是出于这种简单易用的特性，现在越来越多的芯片集成了这种通信协议。

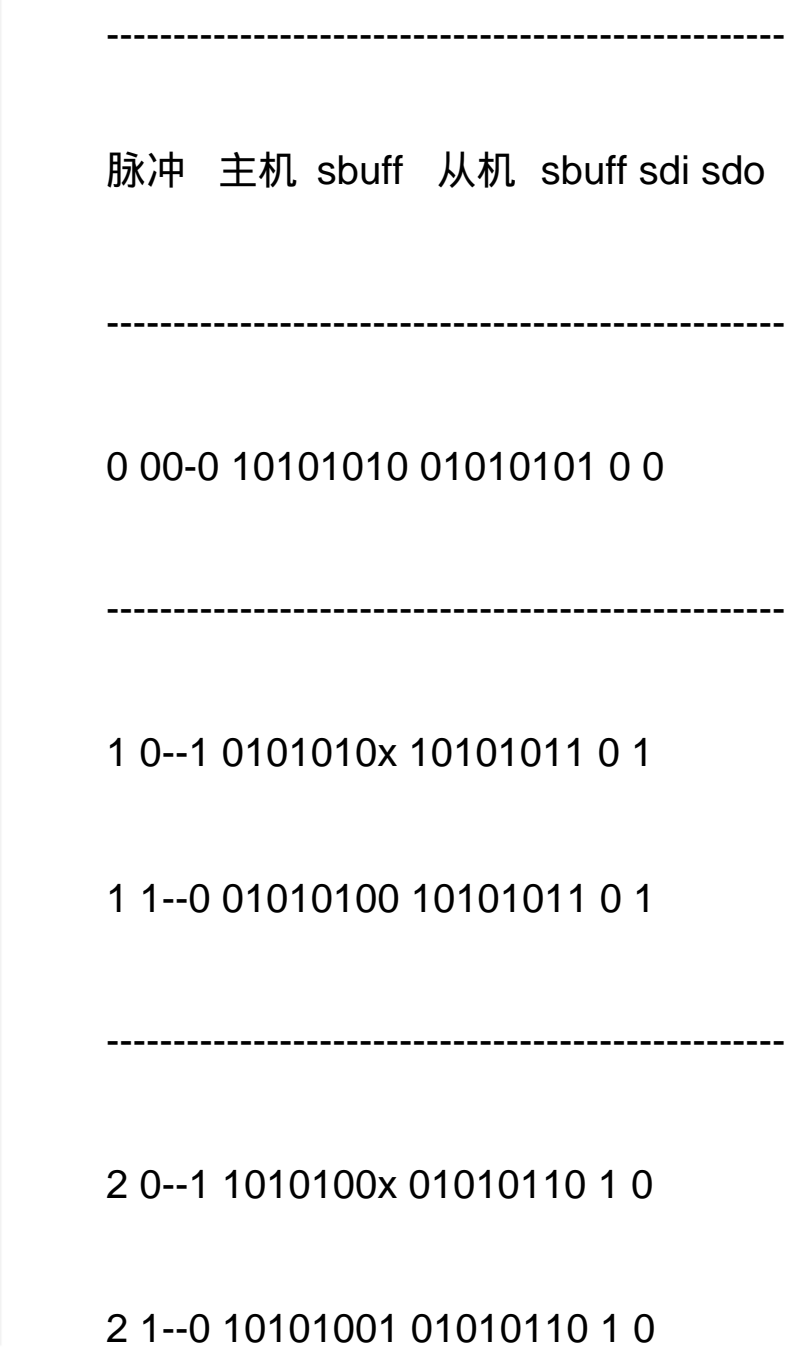
SPI 是一个环形总线结构，由 ss(cs)、sck、sdi、sdo 构成，其时序其实很简单，主要是在 sck 的控制下，两个双向移位寄存器进行数据交换。

上升沿发送、下降沿接收、高位先发送。

上升沿到来的时候，sdo 上的电平将被发送到从设备的寄存器中。

下降沿到来的时候，sdi 上的电平将被接收到主设备的寄存器中。

假设主机和从机初始化就绪：并且主机的 sbuff=0xaa (10101010)，从机的 sbuff=0x55 (01010101)，下面将分步对 spi 的 8 个时钟周期的数据情况演示一遍（假设上升沿发送数据）。



3 0--1 0101001x 10101101 0 1

3 1--0 01010010 10101101 0 1

4 0--1 1010010x 01011010 1 0

4 1--0 10100101 01011010 1 0

5 0--1 0100101x 10110101 0 1

5 1--0 01001010 10110101 0 1

6 0--1 1001010x 01101010 1 0

6 1--0 10010101 01101010 1 0

7 0--1 0010101x 11010101 0 1

7 1--0 00101010 11010101 0 1

8 0--1 0101010x 10101010 1 0

8 1--0 01010101 10101010 1 0

这样就完成了两个寄存器 8 位的交换，上面的 0--1 表示上升沿、 1--0 表示下降沿， sdi、sdo 相对于主机而言的。根据以上分析，一个完整的传送周期是 16 位，即两个字节，因为，

首先主机要发送命令过去，然后从机根据主机的名准备数据，主机在下一个 8 位时钟周期才把数据读回来。

SPI 总线是 Motorola 公司推出的三线同步接口， 同步串行 3 线方式进行通信：一条时钟线 SCK，一条数据输入线 MOSI，一条数据输出线 MISO; 用于 CPU 与各种外围器件进行全双工、同步串行通讯。 SPI 主要特点有：可以同时发出和接收串行数据；可以当作主机或从机工作；提供频率可编程时钟；发送结束中断标志；写冲突保护；总线竞争保护等。

SPI 总线有四种工作方式（SP0, SP1, SP2, SP3），其中使用的最为广泛的是 SPI0 和 SPI3 方式。

SPI 模块为了和外设进行数据交换，根据外设工作要求，其输出串行同步时钟极性和相位可以进行配置，时钟极性（CPOL）对传输协议没有重大的影响。如果 CPOL=0，串行同步时钟的空闲状态为低电平；如果 CPOL=1，串行同步时钟的空闲状态为高电平。时钟相位(CPHA) 能够配置用于选择两种不同的传输协议之一进行数据传输。如果 CPHA=0，在串行同步时钟的第一个跳变沿（上升或下降）数据被采样；如果 CPHA=1，在串行同步时钟的第二个跳变沿（上升或下降）数据被采样。 SPI 主模块和与之通信的外设时钟相位和极性应该一致。

SPI 时序图详解 -SPI 接口在模式 0 下输出第一位数据的时刻

SPI 接口在模式 0 下输出第一位数据的时刻

SPI 接口有四种不同的数据传输时序，取决于 CPOL 和 CPHL 这两位的组合。图 1 中表现了这四种时序，

时序与 CPOL、CPHL 的关系也可以从图中看出。

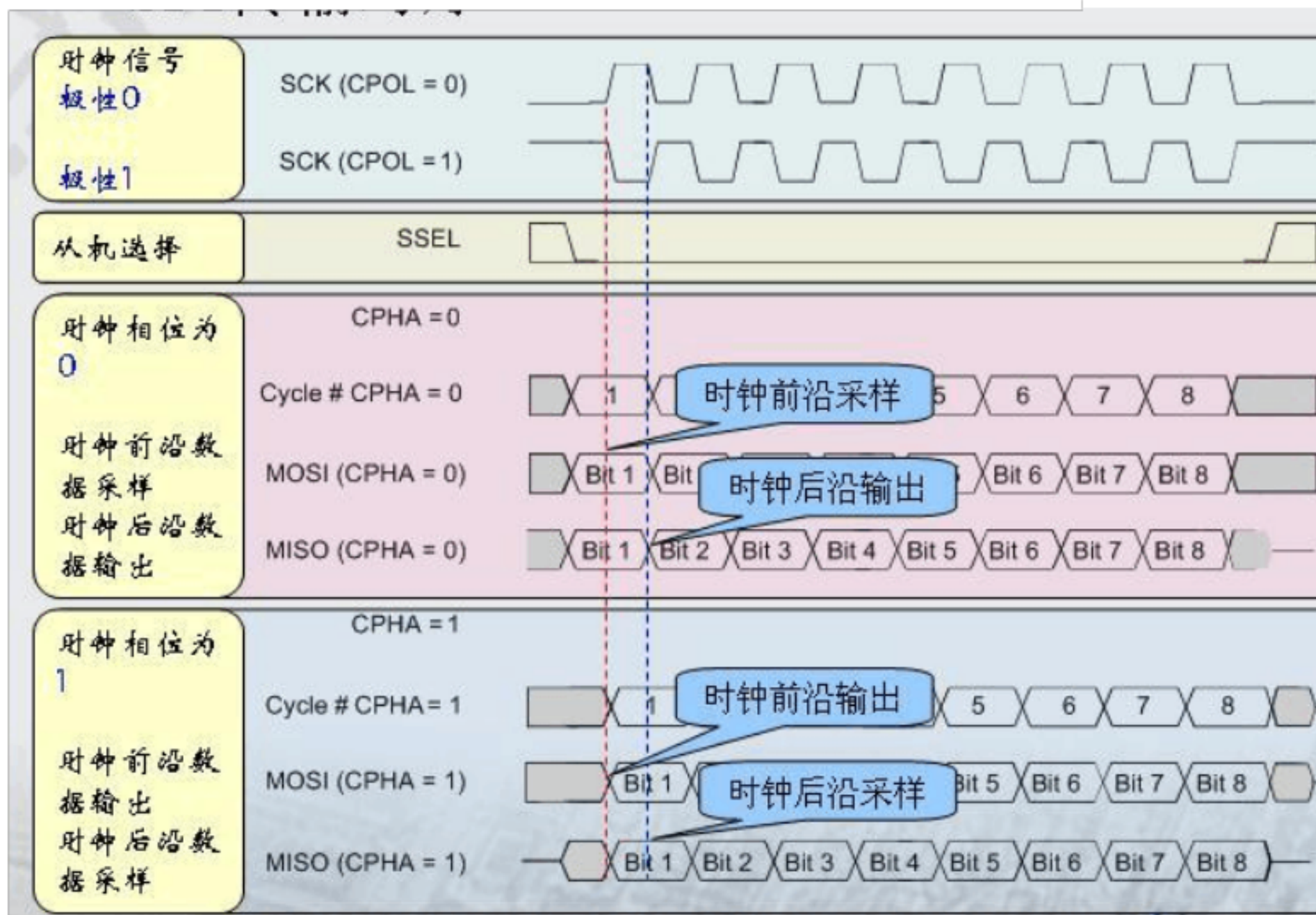


图 1

CPOL 是用来决定 SCK 时钟信号空闲时的电平， CPOL = 0 ,空闲电平为低电平， CPOL = 1 时，

空闲电平为高电平。 CPHA 是用来决定采样时刻的， CPHA=0 ，在每个周期的第一个时钟沿采样，

CPHA = 1 ，在每个周期的第二个时钟沿采样。

由于我使用的器件工作在模式 0 这种时序（ CPOL = 0 ，CPHA = 0 ），所以将图 1 简化为图 2 ，

只关注模式 0 的时序。

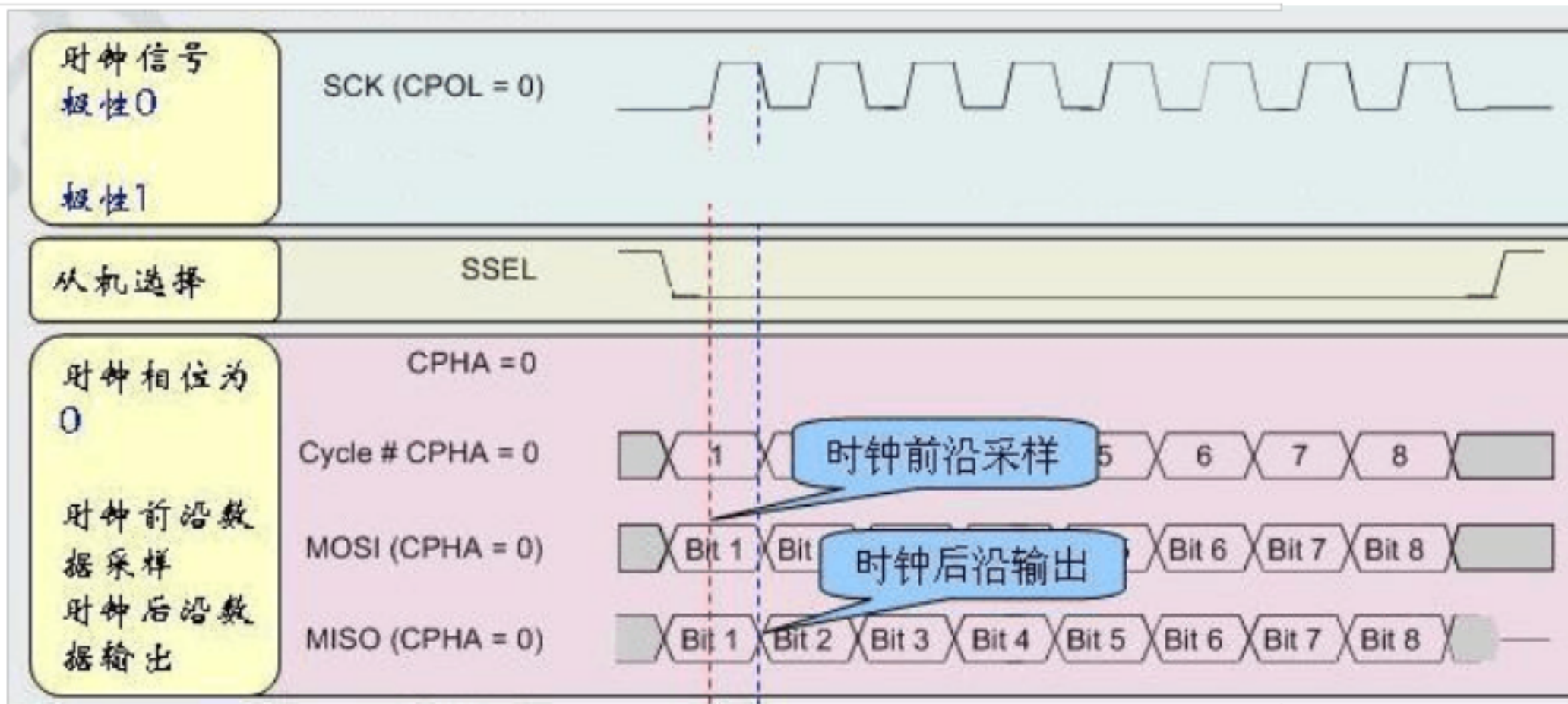


图 2

我们来关注 SCK 的第一个时钟周期，在时钟的前沿采样数据（上升沿，第一个时钟沿），

在时钟的后沿输出数据（下降沿，第二个时钟沿）。首先来看主器件，主器件的输出

口（MOSI）输出的数据 bit1，

在时钟的前沿被从器件采样，那主器件是在何时刻输出 bit1 的呢？bit1 的输出时刻实际上在 SCK 信号有效以前，

比 SCK 的上升沿还要早半个时钟周期。bit1 的输出时刻与 SSEL 信号没有关系。再来看从器件，

主器件的输入口 MISO 同样是在时钟的前沿采样从器件输出的 bit1 的，那从器件又是在何时刻输出 bit1 的呢。

从器件是在 SSEL 信号有效后，立即输出 bit1，尽管此时 SCK 信号还没有起效。关于上面的主器件

和从器件输出 bit1 位的时刻，可以从图 3、4 中得到验证。

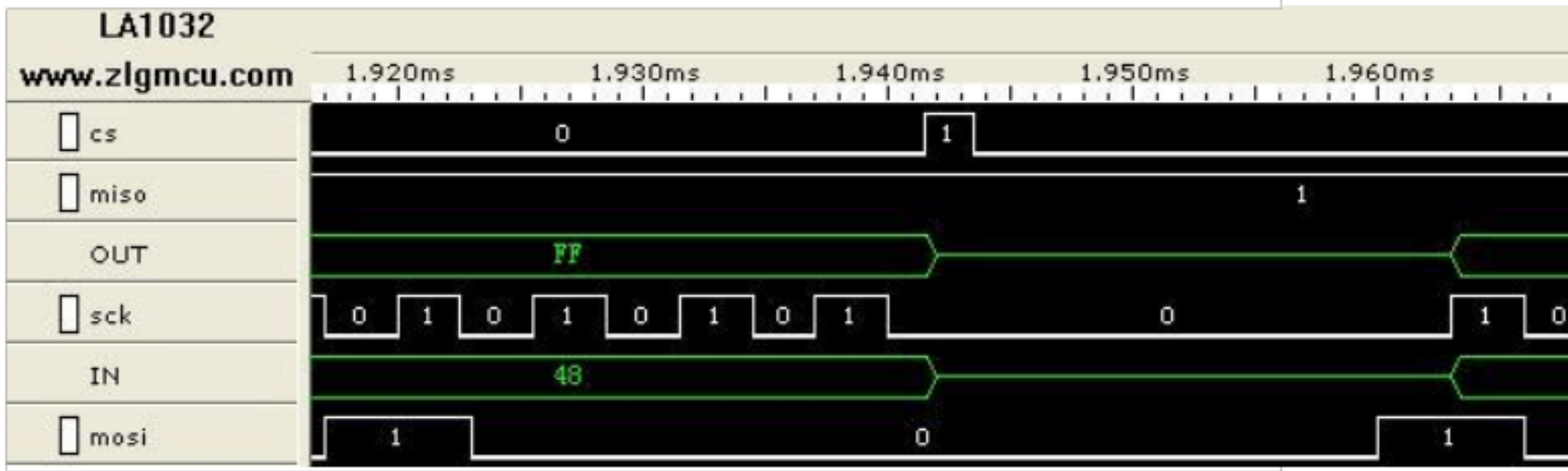


图 3

注意图 3 中，CS 信号有效后（低电平有效，注意 CS 下降沿后发生的情况），故意用延时程序

延时了一段时间，之后再向数据寄存器写入了要发送的数据，来观察主器件输出 bit1 的情况（MOSI）。

可以看出，bit1（值为 1）是在 SCK 信号有效之前的半个时钟周期的时刻开始输出的（与 CS 信号无关），

到了 SCK 的第一个时钟周期的上升沿正好被从器件采样。

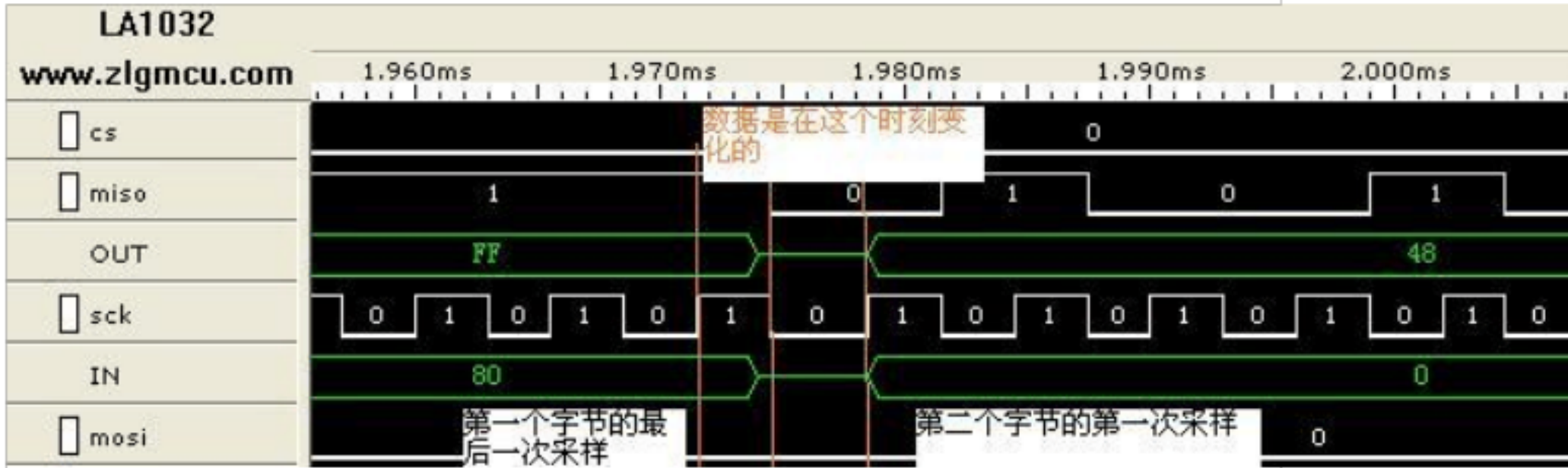


图 4

图 4 中，注意看 CS 和 MISO 信号。我们可以看出，CS 信号有效后，从器件立刻输出了 bit1（值为 1）。

通常我们进行的 spi 操作都是 16 位的。图 5 记录了第一个字节和第二个字节间的相互衔接的过程。

第一个字节的最后一位在 SCK 的上升沿被采样，随后的 SCK 下降沿，从器件就输出了第二个字节的第一位。



SPI 总线协议介绍（接口定义 ,传输时序）

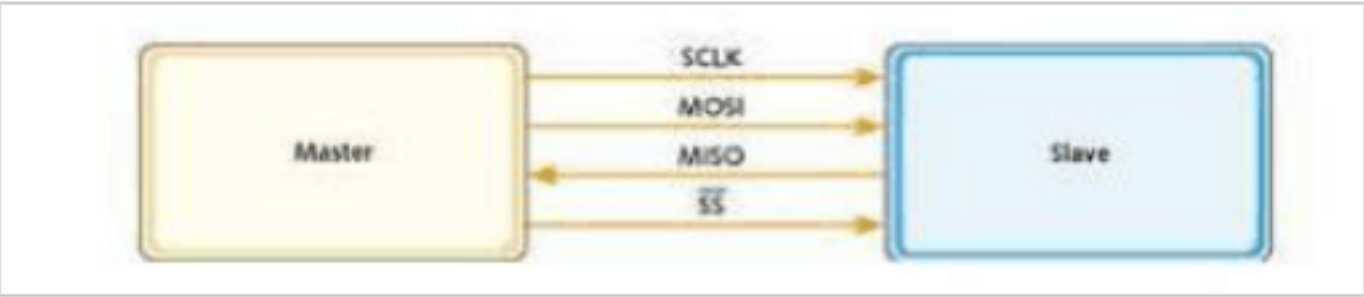
一、技术性能

SPI 接口是 Motorola 首先提出的全双工三线同步串行外围接口，采用主从模式（Master Slave）架构；支持多 slave 模式应用，一般仅支持单 Master。

时钟由 Master 控制，在时钟移位脉冲下，数据按位传输，高位在前，低位在后（MSB first）；SPI 接口有 2 根单向数据线，为全双工通信，目前应用中的数据速率可达几 Mbps 的水平。

二、接口定义

SPI 接口共有 4 根信号线，分别是：设备选择线、时钟线、串行输出数据线、串行输入数据线。



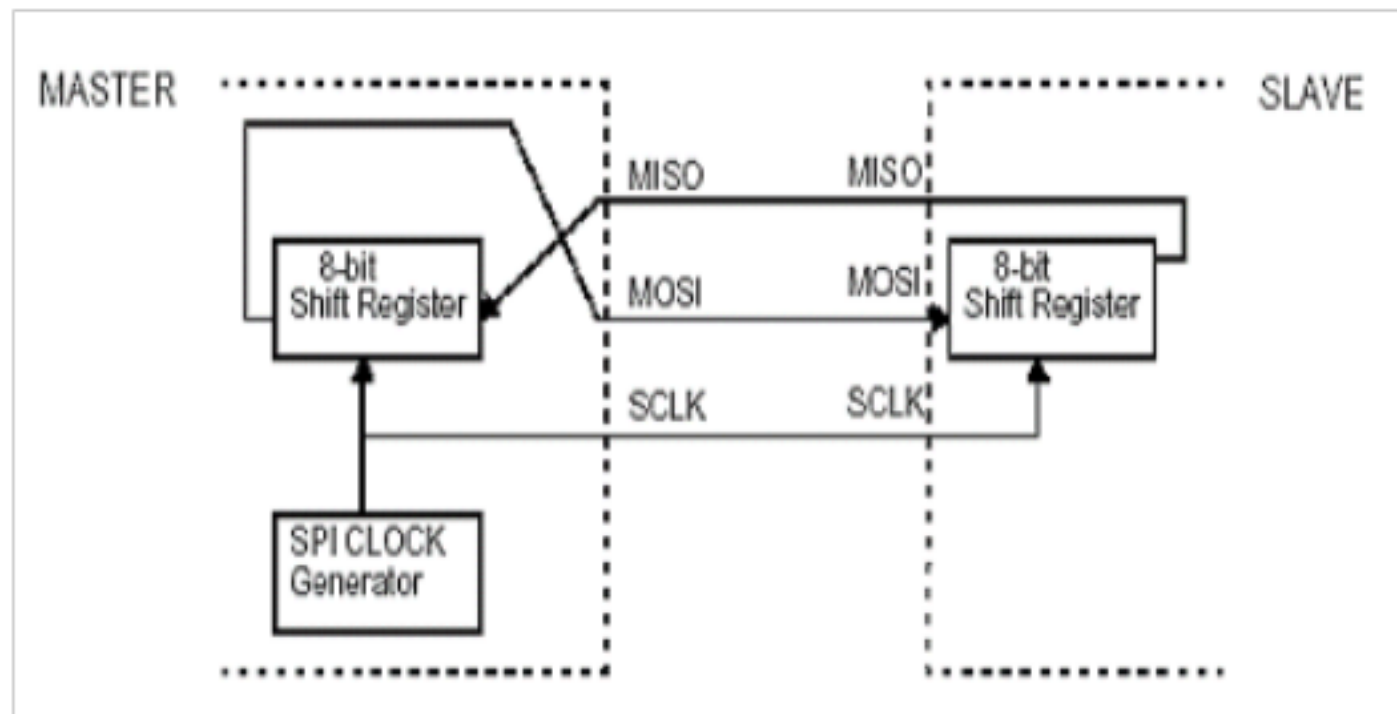
（1）MOSI：主器件数据输出，从器件数据输入

（2）MISO：主器件数据输入，从器件数据输出

(3) SCLK : 时钟信号 , 由主器件产生

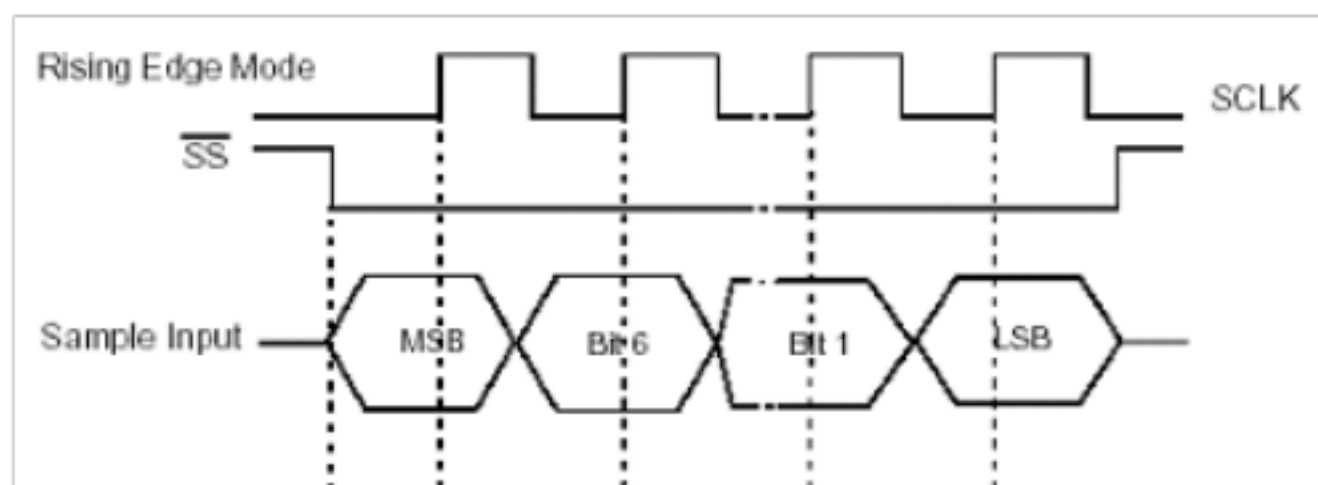
(4) /SS : 从器件使能信号 , 由主器件控制

三、内部结构



四、传输时序

SPI 接口在内部硬件实际上是两个简单的移位寄存器 , 传输的数据为 8 位 , 在主器件产生的从器件使能信号和移位脉冲下 , 按位传输 , 高位在前 , 低位在后。如下图所示 , 在 SCLK 的下降沿上数据改变 , 上升沿一位数据被存入移位寄存器。



SPI 接口没有指定的流控制 , 没有应答机制确认是否接收到数据。

SPI 接口时钟配置：

在主设备这边配置 SPI 接口时钟的时候一定要弄清楚从设备的时钟要求，因为主设备这边的时钟极性和相位都是以从设备为基准的。因此在时钟极性的配置上一定要搞清楚从设备是在时钟的上升沿还是下降沿接收数据，是在时钟的下降沿还是上升沿输出数据。但要注意的，由于主设备的 SDO 连接从设备的 SDI，从设备的 SDO 连接主设备的 SDI，从设备 SDI 接收的数据是主设备的 SDO 发送过来的，主设备 SDI 接收的数据是从设备 SDO 发送过来的，所以主设备这边 SPI 时钟极性的配置（即 SDO 的配置）跟从设备的 SDI 接收数据的极性是相反的，跟从设备 SDO 发送数据的极性是相同的。下面这段话是 Sychip Wlan8100 Module Spec 上说的，充分说明了时钟极性是如何配置的：

The 81xx module will always input data bits at the rising edge of the clock, and the host will always output data bits on the falling edge of the clock.

意思是：主设备在时钟的下降沿发送数据，从设备在时钟的上升沿接收数据。因此主设备这边 SPI 时钟极性应该配置为下降沿有效。

又如，下面这段话是摘自 LCD Driver IC SSD1289：

SDI is shifted into 8-bit shift register on every rising edge of SCK in the order of data bit 7, data bit 6 data bit 0.

意思是：从设备 SSD1289 在时钟的上升沿接收数据，而且是按照从高位到低位的顺序接收数据的。因此主设备的 SPI 时钟极性同样应该配置为下降沿有效。

时钟极性和相位配置正确后，数据才能够被准确的发送和接收。因此应该对照从设备的 SPI 接口时序或者 Spec 文档说明来正确配置主设备的时钟。

SPI，是英语 Serial Peripheral interface 的缩写，顾名思义就是串行外围设备接口。SPI，是一种高速的，全双工，同步的通信总线，并且在芯片的管脚上只占用四根线，节约了芯片的管脚，同时为 PCB 的布局上节省空间，提供方便，正是出于这种简单易用的特性，现在越来越多的芯片集成了这种通信协议，比如 P89LPC900.

SPI，是一种高速的，全双工，同步的通信总线，其工作模式有两种：主模式和从模式，无论那种模式，都支持

3Mbit/s 的速率，并且还具有传输完成标志和写冲突保护标志。到目前为止，我使用过的具有 SPI 总线的器件，就是存储芯片 Eprom：at25128，在使用过程中，发现的确是有这种总线的优点。下面以 P89LPC900 单片机的 SPI 总线来解释 SPI 总线的通用使用规则。

LPC900 单片机的 SPI 接口主要由 4 个引脚构成：SPICLK、MOSI、MISO 及 /SS，其中 SPICLK 是整个 SPI 总线的公用时钟，MOSI、MISO 作为主机，从机的输入输出的标志，MOSI 是

主机的输出，从机的输入，MISO 是主机的输入，从机的输出。/SS 是从机的标志管脚，在互相通信的两个 SPI 总线的器件，/SS 管脚的电平低的是从机，相反 /SS 管脚的电平高的是主机。在一个 SPI 通信系统中，必须有主机。

SPI 总线可以配置成单主单从，单主多从，互为主从。今以互为主从模式作为讲解：

要进行 SPI 互为主从操作，必须遵照以下步骤：

1 对 A、B 进行初始化，均设为主机（需要进行以下操作）。

a) SPI 端口初始化为双向。

b) SPCTL 配置为 0x50，SSIG=0，SPEN=1，MSTR=1。

c) 清除 SPSTAT 中的 SPIF 及 WCOL 标志位为 0。

d) 如果需要使用 SPI 中断，可使能相应中断位。

2 将 A 上一个引脚连接到 B 的 /SS 引脚上，然后拉低 /SS，可将 B 强行置为从机模式，同时 B 机会发生以下变化：

a) B 机的 MSTR 位自动清 0。

b) B 机的 MOSI 及 SPICLK 强行变为输入模式，MISO 则变为输出模式。

c) B 机 SPIF 位置位。

d) 如果 SPI 中断使能，B 机将执行 SPI 中断服务程序。

3 B 机可设置为查询接收或中断接收方式，以时刻准备接收由 A 机发送过来的数据，要使 B 机恢复为主机，必须完整执行步骤 1。本示例中，通过两块 DP932 实验板构成了 SPI 互为主从测试系统。

程序中应注意的问题：

1 程序中应注意对首次拉低 SS 引脚进行处理：当 A 机首次通过 B_SS 将 B 机设置为从机后，从机的 SBIF 位会置位（会被认为完成一次传输），如果这之前，使能了 SPI 中断，

则从机则会执行相应的中断服务程序（本示例程序中，当 B 机的 SS 引脚被拉为低电平，B 机的 SBIF 首次置位进行处理）。

2 关于从机恢复为主机的问题：互为主从模式中，当 B 机被 A 机设置为从机后，CPCTL 寄存器中 MSTR 位被清除为 0，且 SPIF 被置 1，MOSI 和 SPICLK 强制变为输入模式，MISO 强制变为输出模式。要想恢复为主机，必须执行以下操作：

a) 将 MSTR 位置 1，SPIF 位清 0。

b) 将 MOSI，SPICLK，MISO 及 SS 重新恢复为准双向口。

c) 在 a)、b)之前，需要注意将 B_SS 拉高，如果其一直为低电平，即使完成 a)、b)操作，也会将 B 机重新设置为从机。

3 在 SPI 总线的使用过程中，可以通过 DORD（SPI 数据顺序选择位），CPOL（SPI 时钟极性选择位），CPHA（SPI 时钟相位选择位）控制主/从机传输格式。对于本实验，可以忽略这些位的影响，但是在使用一些其他 SPI 器件时，必须根据从器件数据手册的要求，对 SPI 数据的传输顺序，SPI 的时钟极性，及 SPI 的时钟相位进行正确的设置。

4 一些 SPI 的应用系统，由于硬件的设计并不是很合理，所以有时 SPI 通信不正常（传输数据出错，或其他情况），你可以试着降低 SPI 总线的传输速率，或者调节一些 SPI 时钟极性及相位，以使传输稳定。