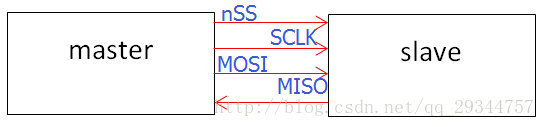
# SPI通信协议（3线）

SPI(Serial peripheral interface)即串行外围设备接口，是由Motorola首先在其MC68HCxx系列单片机上定义的，基于高速全双工总线的通讯协议。(又是高速，而且全双工，确实强大)被广泛应用于ADC、LCD等设备与MCU之间。

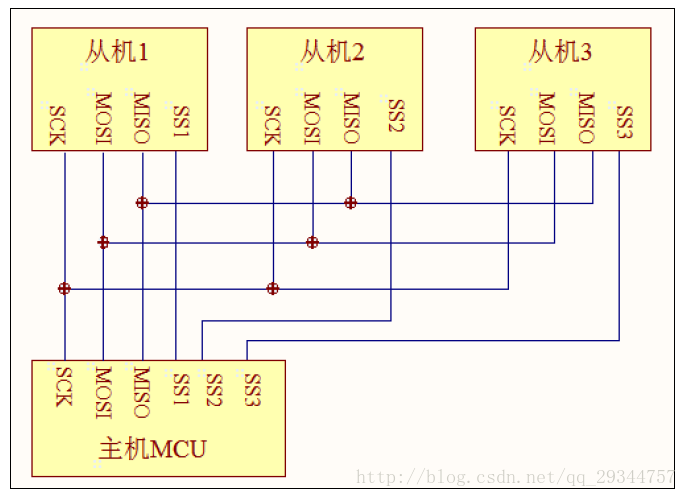
跟前面学习I2C、USART一样，学习一种协议，还是从两个层面分析：物理层和协议层。

**1. SPI物理层**

SPI通讯需要使用4条线：3条总线和1条片选



SPI还是遵循主从模式，3条总线分别是SCK、MOSI和MISO，片选线为nSS(低电平有效)，SPI协议适用于一主多从的工作场景：



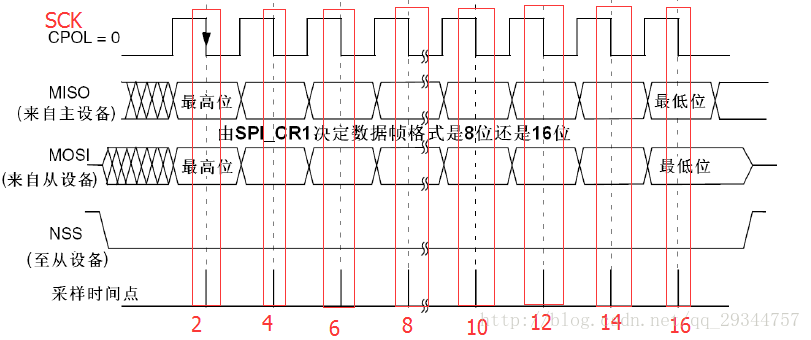
(1) nSS(Slave Select)：片选信号线，用于选中SPI从设备。每个从设备独立拥有这条nSS信号线，占据主机的一个引脚。设备的其他总线是并联到SPI主机的，即无论多少个从设备，都共同使用这3条总线。当从设备上的nSS引脚被置拉低时表明该从设备被主机选中。

(2) SCK(Serial Clock)：时钟信号线，通讯数据同步用。时钟信号由通讯主机产生，它决定了SPI的通讯速率。

(3) MOSI(Master Ouput Slave Input)：主机(数据)输出/从设备(数据)输入引脚，即这条信号线上传输从主机到从机的数据。

(4) MISO(Master Input Slave Ouput)：主机(数据)输入/从设备(数据)输出引脚，即这条信号线上传输从机从到主机的数据主从机通过两条信号线来传输数据，那么自然是全双工通讯的了。之前的I2C通讯，数据只在一条SDA线上传输，主从机数据交互只能采用半双工。

**2. SPI协议层**



如上为SPI通讯时序图，nSS、SCK、MOSI信号均由主机产生，MISO信号由从机产生。在nSS为低电平的前提下，MOSI和MISO信号才有效，在每个时钟周期MOSI和MISO传输一位数据。

跟I2C通讯类似，SPI通讯也需要通讯的起始/结束信号，有效数据和同步时钟。

**2.1 通讯的起始/结束信号**

图中的nSS信号由高电平变为低电平即为SPI通讯的起始信号，反过来，nSS信号由低电平变为高电平即为SPI通讯的结束信号。**这个可比I2C简单得多吧**。当从机检测到自身的nSS引脚被拉低时就知道自己被主机选中，准备和主机进行通讯。、

**2.2 有效数据的采集**

SPI通讯的数据采集是个相对复杂的环节，先不说其他，以上图为例：

图中红色框框即为有效数据被采集的时间点，"CPOL = 0"所在的脉冲信号表示的是用于进行数据同步的SCK，MOSI和MISO线上的数据在每个SCK时钟周期传输一位数据，注意，数据的输入/输出是可以同时进行的。

由图可见，在SCK为奇数（更正：这里应该是偶数）边沿(在这里该边沿为下降沿)时，数据得到有效采样，也就是说，**在这个时刻，MISO和MOSI的数据有效**，高电平表示数据1，低电平表示数据0，**在其它时刻数据并无效**，可以理解为为下一次MISO和MOSI的数据传输做准备。

数据在传输中，**高位在先还是低位在先**，SPI协议并无明确规定，但是数据要在主从机中正确传输，自然双方要先约定好，一般会采用高位在先(MSB)方式传输。

这里需要再提及的概念是时钟极性(CPOL)和时钟相位(CPHA)。

**时钟极性**(CPOL)指通讯设备处于空闲状态(SPI开始通讯前、nSS线无效)时，SCK的状态。

CPOL = 0：SCK在空闲时为低电平

CPOL = 1：SCK在空闲时为高电平

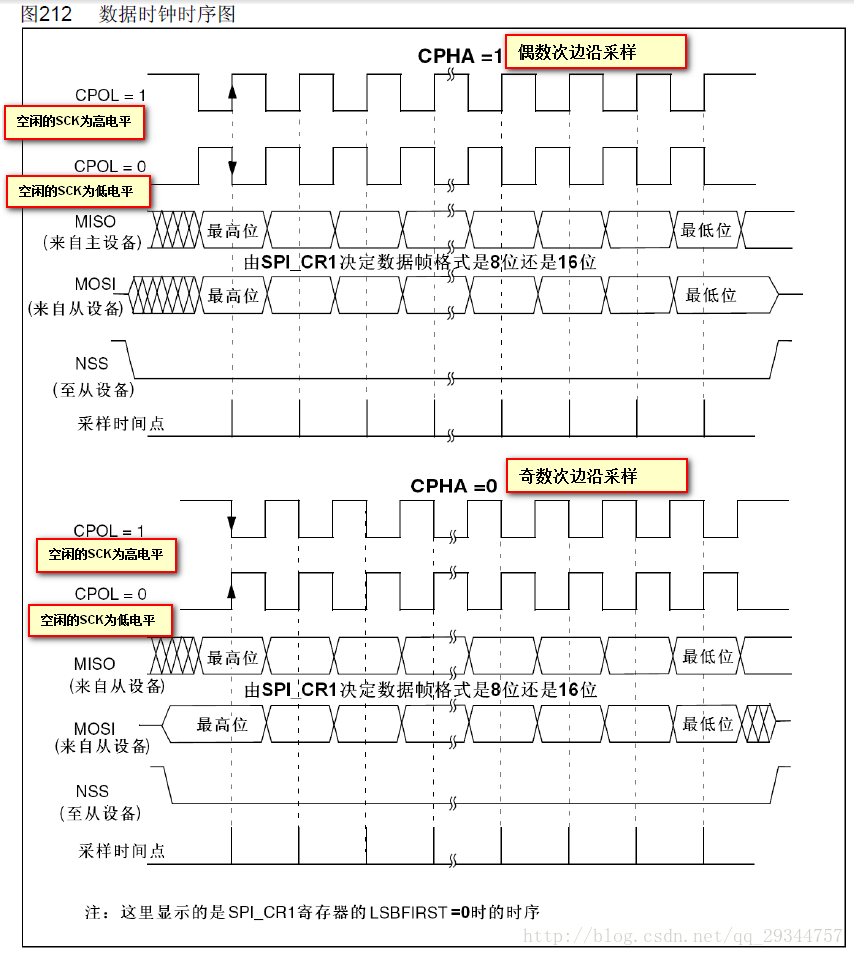
**时钟相位**(CPHA)指数据的采样时刻位于SCK的偶数边沿采样还是奇数边沿采样。

CPHA = 0：在SCK的奇数边沿采样

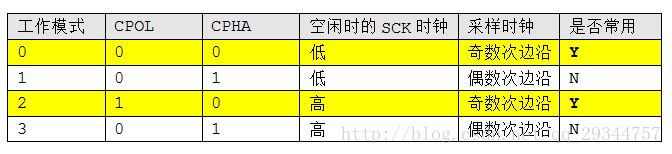
CPHA = 1：在SCK的偶数边沿采样

那么这样说来，SPI的采样时刻并非由上升沿/下降沿决定的。注意的是，在数据采样时刻，MOSI和MOSI的电平为有效电平，数据不能在这个时刻进行切换注意的是，在数据采样时刻，MOSI和MOSI的电平为有效电平，数据不能在这个时刻进行切换。在非采样时刻MOSI和MISO上的信号才能切换。

完整的时序图如下：



所以说，SPI有4中工作模式：



更正：工作模式3的CPOL应为1。

注意要让主机和从机需要在相同的工作模式下，这样才可以实现正常通讯。