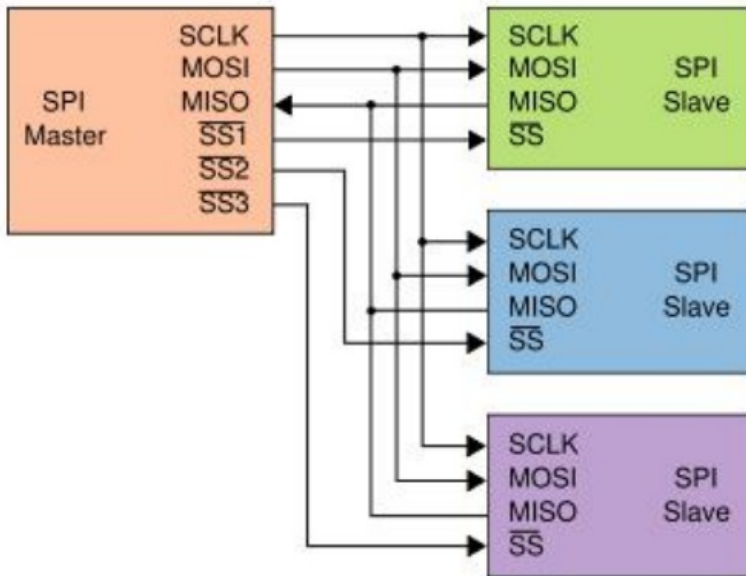


专题20-SPI驱动程序设计

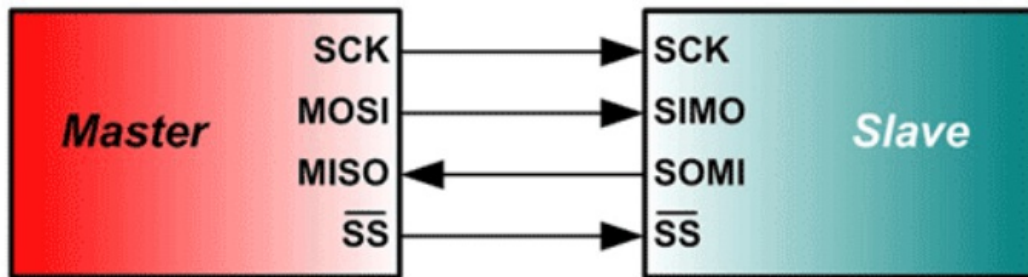
一、SPI总线介绍

1.1、总线结构

SPI(Serial Peripheral Interface)串行外设接口,是一种高速的,全双工,同步的通信总线。采用主从模式 (Master Slave) 架构,支持多个slave ,一般仅支持单Master。



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

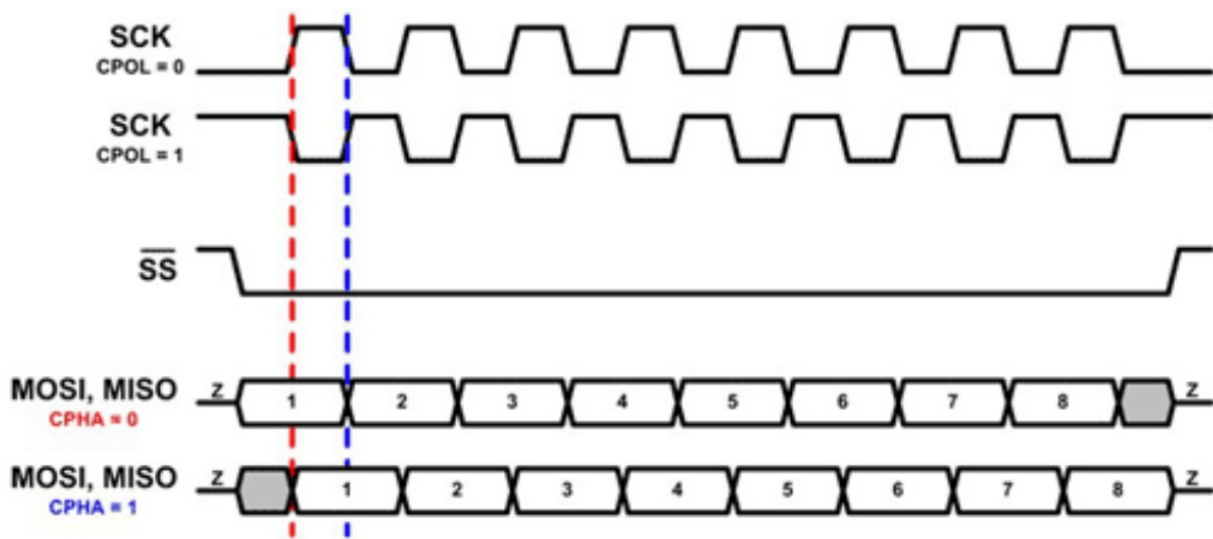


1.2、数据传输过程

主节点通过 MOSI 线输出数据,从节点在 SIMO 处从主节点读取数据。同时,也在通过 SOMI 输出 MSB (最高位),主节点会在 MISO处读取从节点的数据,整个过程将一直持续,直至交换完所有数据。



1.3、总线时序



CPOL极性：决定时钟空闲时为高电平还是低电平

CPOL=0：CLK空闲时是低电平，CLK有效时是高电平

CPOL=1：CLK空闲时是高电平，CLK有效时是低电平

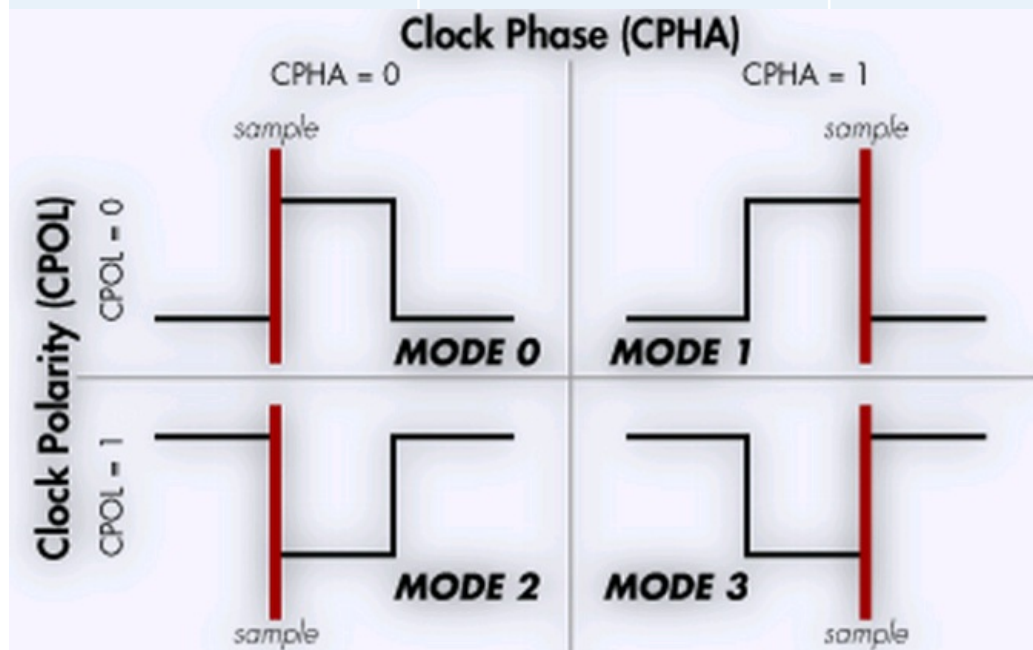
CPHA相位：决定何时进行数据采样(读取)

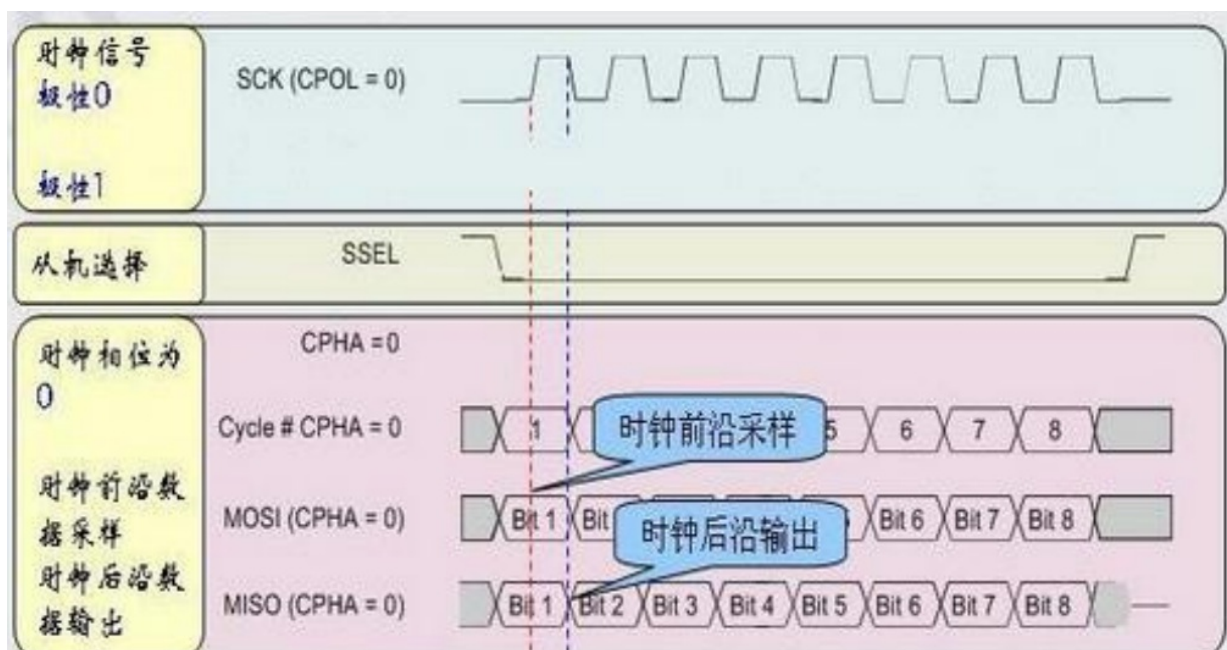
CPHA=0:第一个边沿采样

CPHA=1:第二个边沿采样

根据CPOL和CPHA的不同组合，SPI被分为4种模式

类型	CPOL	CPHA
SPI0	0	0
SPI1	0	1
SPI2	1	0
SPI3	1	1

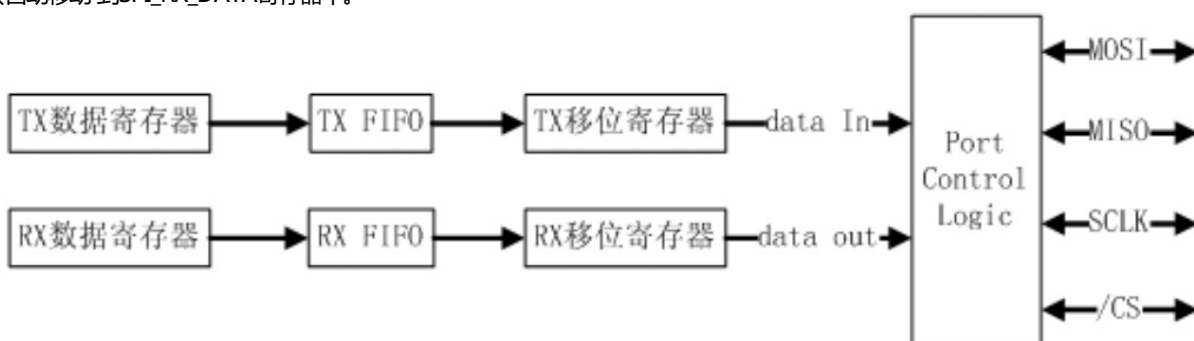




二、SPI裸机驱动设计

2.1、SPI控制器工作流程

SPI有两个通道，分别为TX通道和RX通道，CPU要写数据到FIFO中，先写数据到SPI_TX_DATA寄存器中，这样此寄存器中的内容就会自动移动到发送FIFO中；同样的道理，如果CPU要从接收FIFO中读取数据，就访问寄存器SPI_RX_DATA，紧接着，接收FIFO的数据就会自动移动到SPI_RX_DATA寄存器中。



2.2、6410裸机驱动分析

三、Linux SPI子系统

3.1、SPI子系统模型

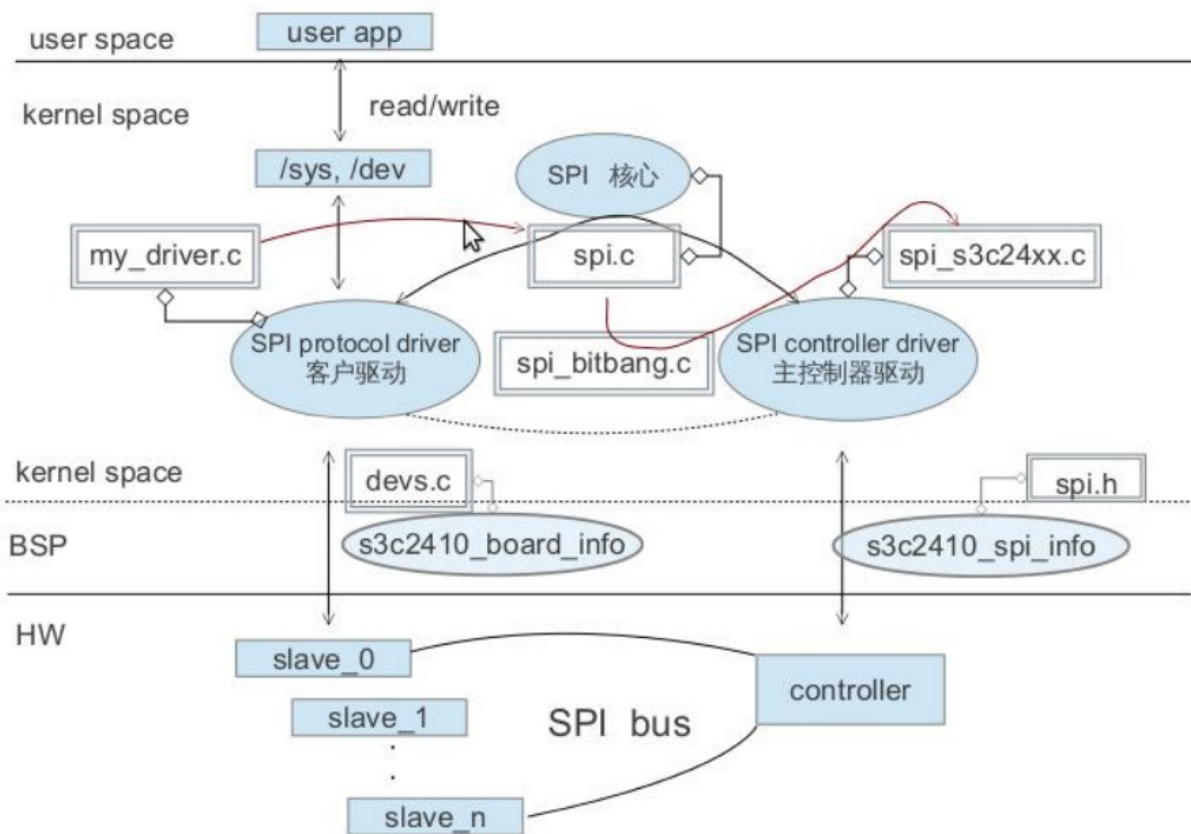


Figure 1 Linux SPI 子系统

1. SPI核心

SPI控制器驱动和设备驱动之间的纽带，它提供了SPI控制器驱动和设备驱动的注册、注销方法等。

2. SPI控制器驱动

对SPI控制器的驱动实现。

3. SPI设备驱动

对SPI从设备的驱动实现，如spi flash

3.2、SPI控制器驱动分析

四、Linux SPI驱动设计

