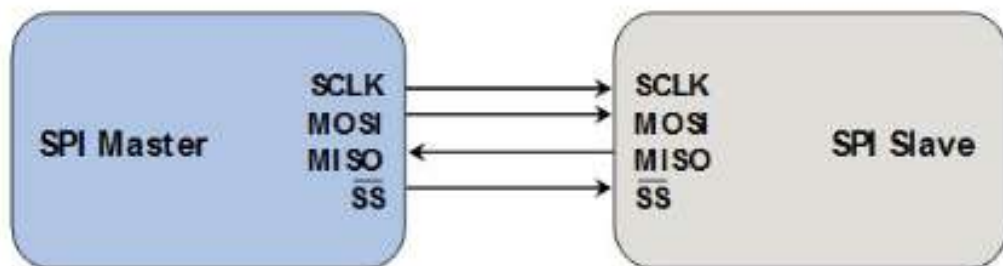


SPI



可知SPI匯流排傳輸只需要4條線就能完成，這四條線的作用分別如下：

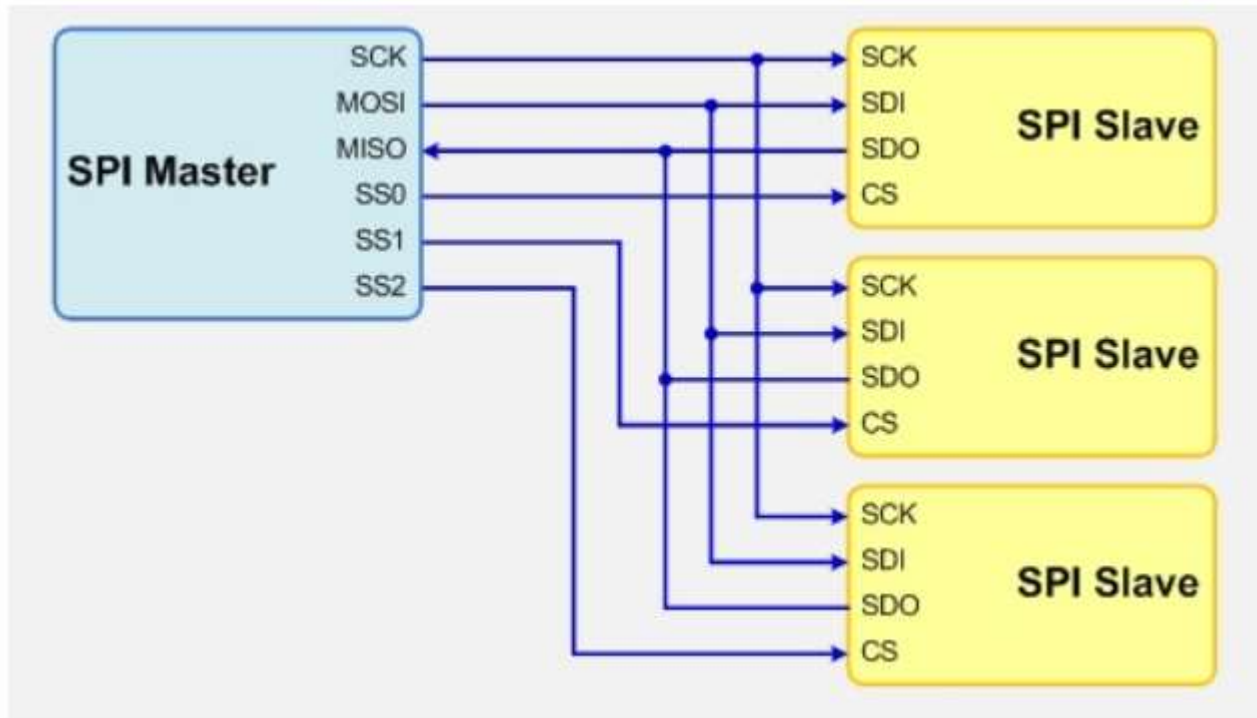
SCK(Serial Clock)：SCK是串列時鐘線，作用是Master向Slave傳送時脈訊號，控制資料交換的時機與速率；

MOSI(Master Out Slave in)：在SPI Master上也被稱為Tx-channel，作用是SPI主機給SPI從機發送資料；

CS/SS(Chip Select/Slave Select)：作用是SPI Master選擇與哪一個SPI Slave通信，低電平表示從機被選中(低電平有效)；

MISO(Master In Slave Out)：在SPI Master上也稱為Rx-channel，作用是SPI主機接收SPI從機傳送過來的資料；

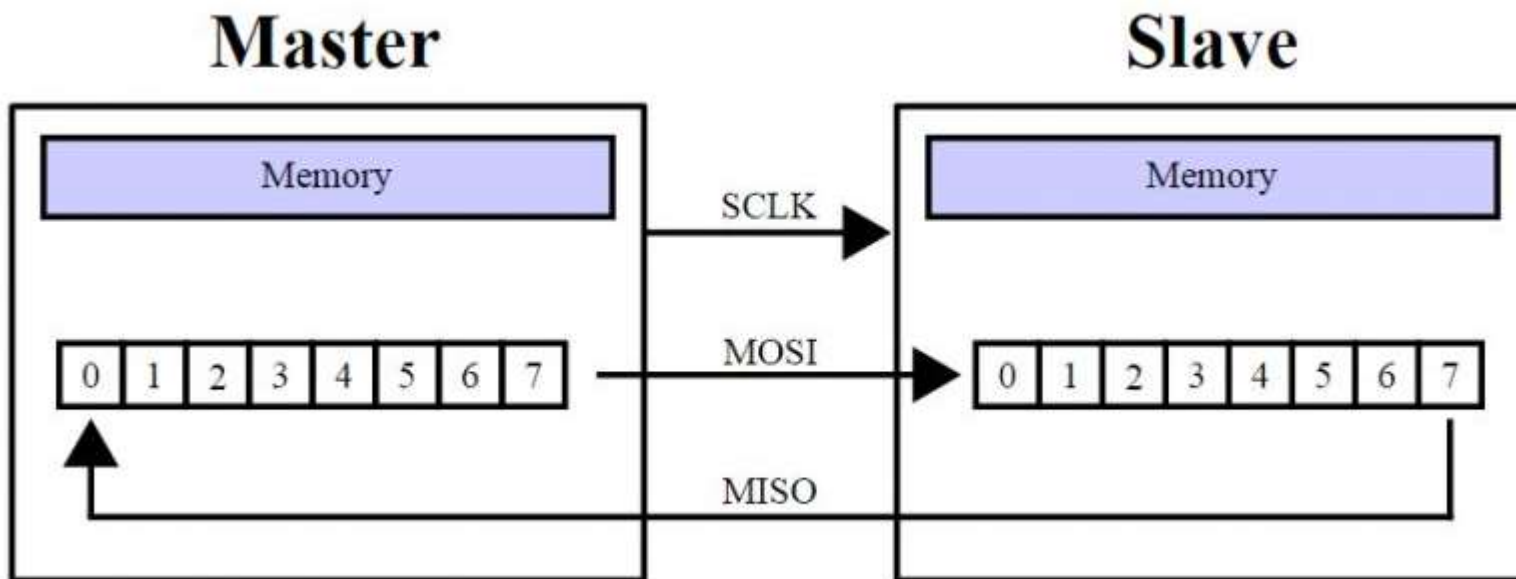
SPI 特點



採用主從模式（Master-Slave）的控制方式，支援單Master多Slave。

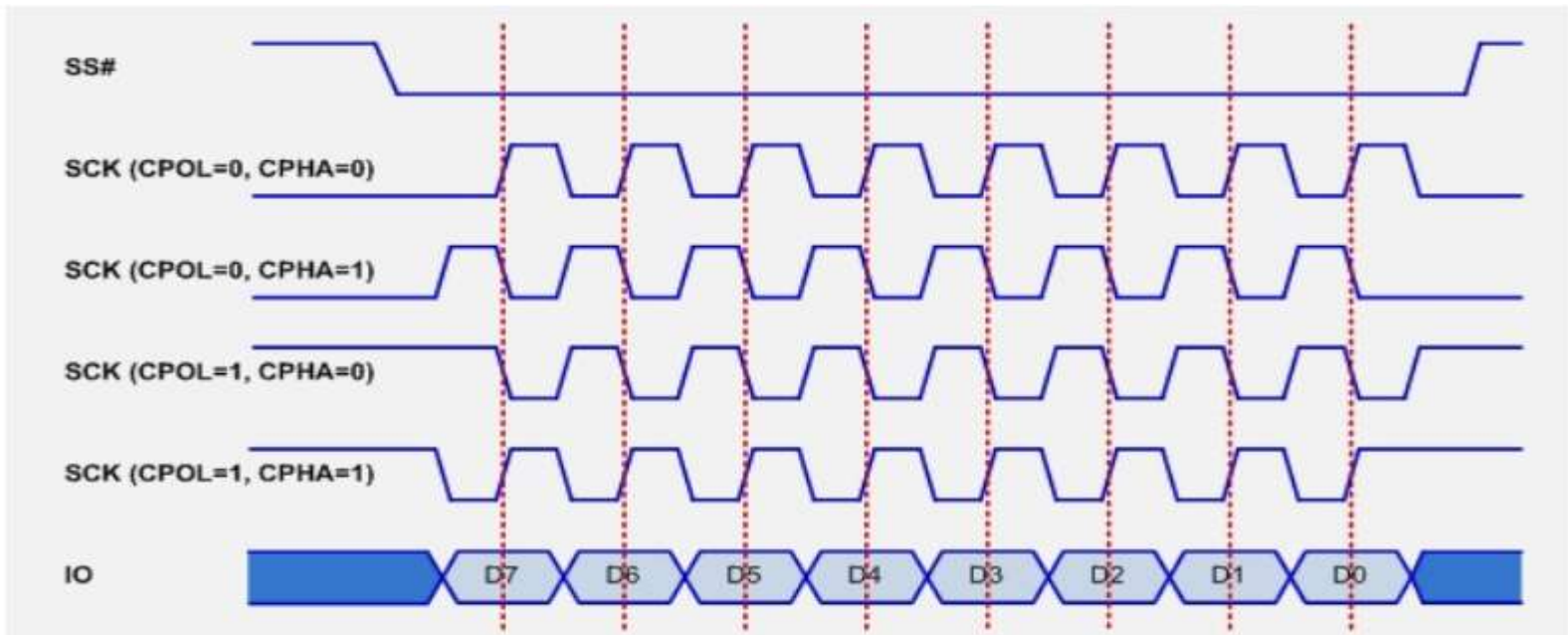
SPI規定了兩個SPI設備之間通訊必須由主設備Master來控制從設備Slave。

FPGA是主機的情況下，不管是FPGA給晶片發送數據還是從晶片接收數據，寫Verilog邏輯的時候片選信號CS與串行時脈訊號SCK必須由FPGA來產生。同時一個Master可以設定多個片選(Chip Select)來控制多個Slave。SPI協定也規定Slave設備的clock由Master透過SCK腳提供給Slave，Slave本身不能產生或控制clock，沒有clock則Slave不能正常運作。單Master多Slave的典型結構如下圖所示



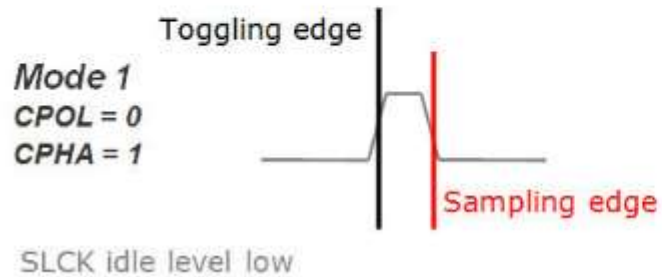
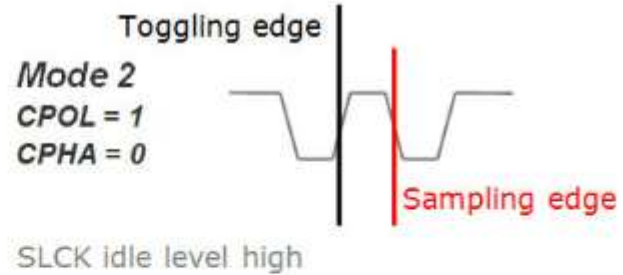
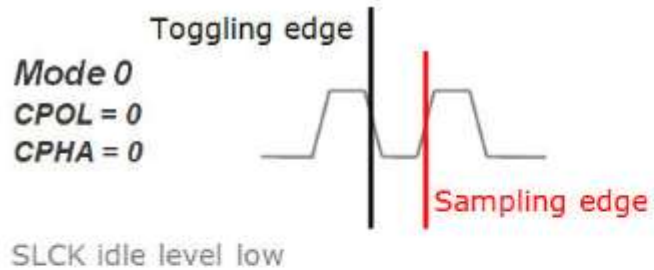
SPI匯流排在傳輸資料的同時也傳輸了時脈訊號，所以SPI協定是一種同步（Synchronous）傳輸協定。Master會根據將要交換的資料產生對應的時脈脈衝，組成時脈訊號，時脈訊號透過時脈極性(CPOL)和時脈相位(CPHA)控制兩個SPI裝置何時交換資料以及何時對接收資料進行取樣，保證資料在兩個裝置之間是同步傳輸的。

SPI 總線輸出模式

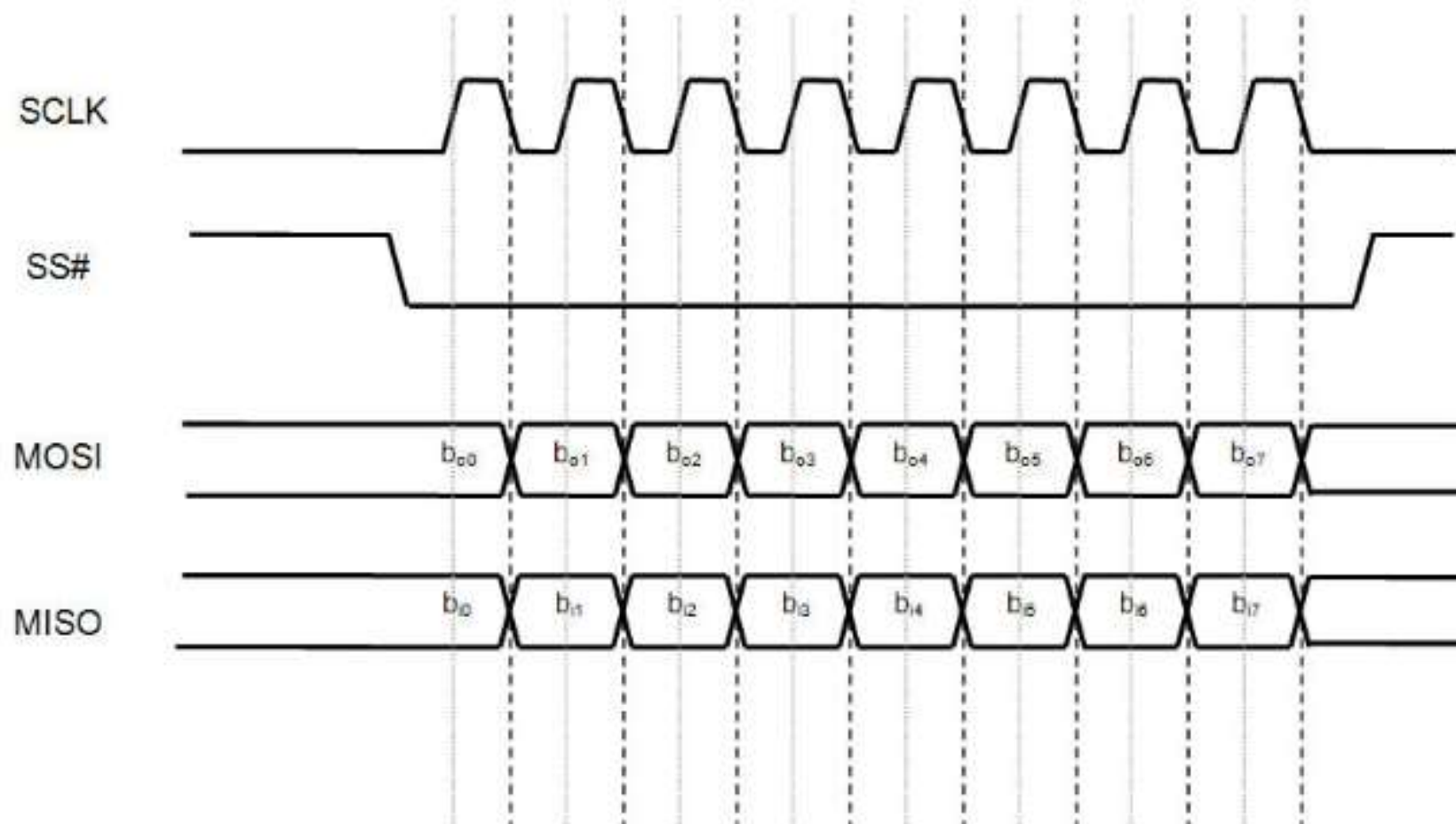


總線的四種模式，CPOL 為SCK 的閒置電平(0 = L,1 = H) CPHA為上升緣或下降緣的判定(0 =上升緣，1 = 下降緣)

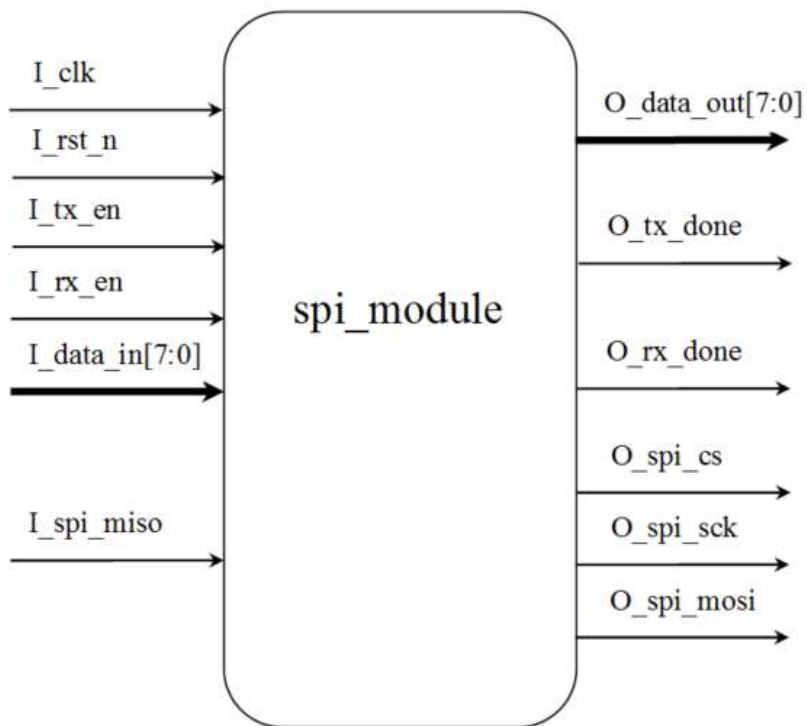
1. CPOL = 0, CPHA = 0。SCK 空閒L，DATA SCK POS採樣，neg換下筆資料
2. CPOL = 0, CPHA = 1。SCK 空閒L，DATA SCK neg採樣，POS換下筆資料
3. CPOL = 1, CPHA = 0。SCK 空閒H，DATA SCK neg採樣，POS換下筆資料
4. CPOL = 1, CPHA = 1。SCK 空閒H，DATA SCK POS採樣，neg換下筆資料



模式0



設計目標



I_clk是系統時鐘；

I_rst_n是系統重設；

I_tx_en是主機給從機發送資料的啟用訊號，當I_tx_en為1時主機才能傳送資料給從機；

I_rx_en是主機從從機接收資料的啟用訊號，當I_rx_en為1時主機才能從從機接收資料；

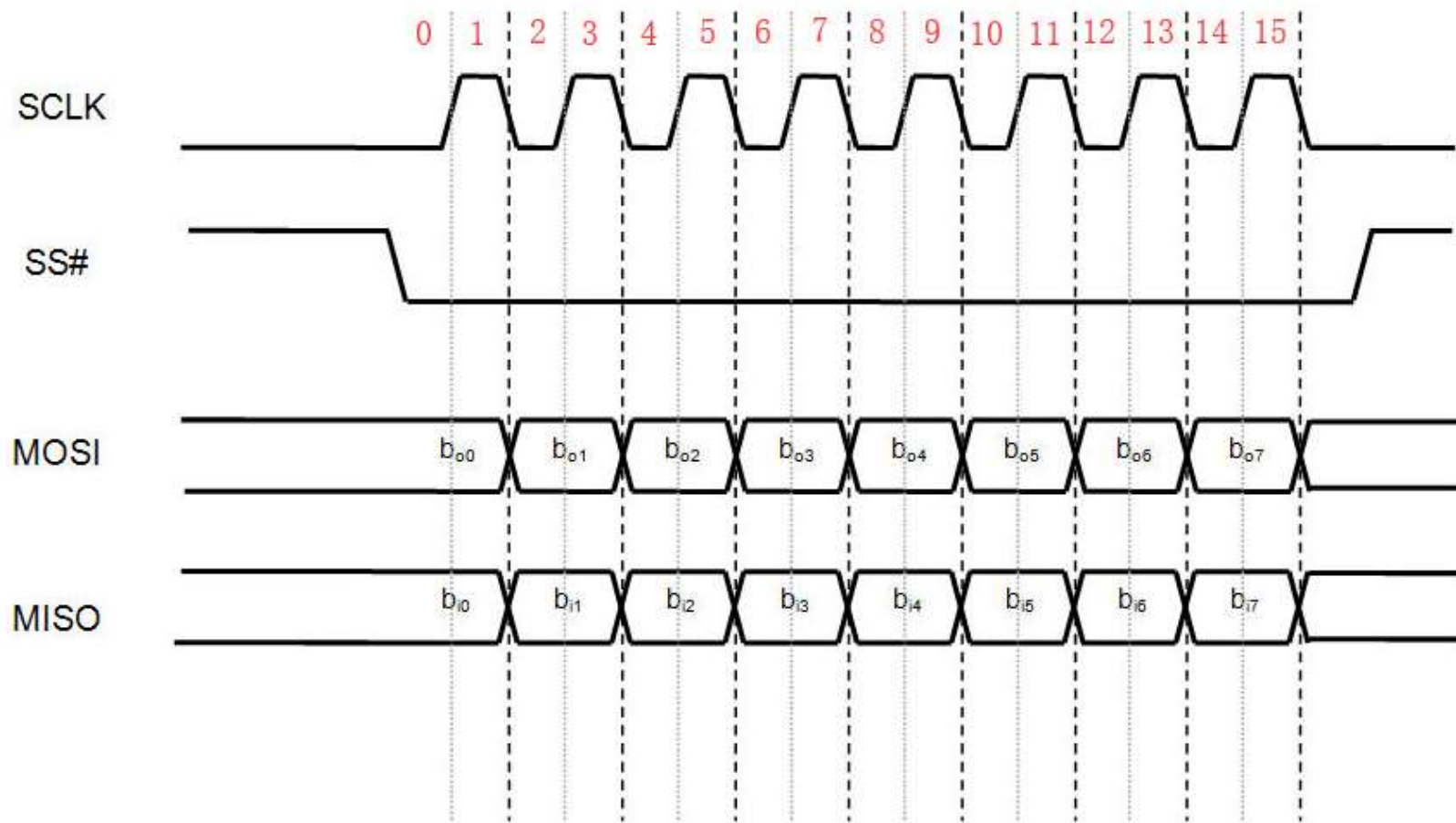
I_data_in是主機要傳送的平行資料；

O_data_out是把從機接收回來的串列資料並行化以後的平行資料；

O_tx_done是主機給從機發送資料完成的標誌位，發送完成後會產生一個高脈衝；

O_rx_done是主機從從機接收資料完成的標誌位，接收完成後會產生一個高脈衝；

I_spi_miso、O_spi_cs、O_spi_sck和O_spi_mosi是標準SPI匯流排協定規定的四根線；



FPGA 去控制或讀寫QSPI FLASH，FPGA 是 MASTER，QSPI 是SLAVER
發送: 當FPGA通過SPI 總線往QSPI FLASH中發送8BIT數據，

首先FPGA將CS/SS 信號設為0，表示開始準備發送訊號，整個發送過程的信號可以分為16個狀態。

狀態0：SCK為0，MOSI為要傳送的資料的最高位，即I_data_in[7]

狀態1：SCK為1，MOSI維持不變

狀態2：SCK為0，MOSI為要傳送的資料的次高位，即I_data_in[6]

狀態3：SCK為1，MOSI維持不變

狀態4：SCK為0，MOSI為要傳送的資料的下一位，即I_data_in[5]

狀態5：SCK為1，MOSI維持不變

狀態6：SCK為0，MOSI為要傳送的資料的下一位，即I_data_in[4]

狀態7：SCK為1，MOSI維持不變

狀態8：SCK為0，MOSI為要傳送的資料的下一位，即I_data_in[3]

狀態9：SCK為1，MOSI維持不變

狀態10：SCK為0，MOSI為要傳送的資料的下一位，即I_data_in[2]

狀態11：SCK為1，MOSI維持不變

狀態12：SCK為0，MOSI為要傳送的資料的下一位，即I_data_in[1]

狀態13：SCK為1，MOSI維持不變

狀態14：SCK為0，MOSI為要傳送的資料的最低位，即I_data_in[0]

狀態15：SCK為1，MOSI維持不變

一個位元組資料發送完畢以後，產生一個發送完成標誌位O_tx_done並把CS/SS訊號拉高完成一次發送。觀察上面的狀態可以發現狀態編號為奇數的狀態要做的操作其實是一模一樣的，所以寫程式的時候為了精簡程式碼，可以把狀態號為奇數的狀態全部整合在一起。

- 接收：當FPGA透過SPI匯流排從QSPI Flash接收一個位元組(8-bit)的資料時，首先FPGA把CS/SS片選訊號設定為0，表示準備開始接收數據，整個接收資料流程其實也可以分為16個狀態，但是與發送過程不同的是，為了確保接收到的數據準確，必須在數據的正中間採樣，也就是說模式0時序圖中灰色實線的地方才是代碼中鎖存數據的地方，所以接收過程的每個狀態執行的操作為：

- 狀態0：SCK為0，不鎖存MISO上的數據
- 狀態1：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[7]
- 狀態2：SCK為0，不鎖存MISO上的數據
- 狀態3：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[6]
- 狀態4：SCK為0，不鎖存MISO上的數據
- 狀態5：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[5]
- 狀態6：SCK為0，不鎖存MISO上的數據
- 狀態7：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[4]
- 狀態8：SCK為0，不鎖存MISO上的數據
- 狀態9：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[3]
- 狀態10：SCK為0，不鎖存MISO上的數據
- 狀態11：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[2]
- 狀態12：SCK為0，不鎖存MISO上的數據
- 狀態13：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[1]
- 狀態14：SCK為0，不鎖存MISO上的數據
- 狀態15：SCK為1，鎖存MISO上的數據，即把MISO上的數據賦值給O_data_out[0]

一個位元組資料接收完畢以後，產生一個接收完成標誌位O_rx_done並把CS/SS訊號拉高完成一次資料的接收。透過觀察上面的狀態可以發現狀態編號為偶數的狀態要做的操作其實是一模一樣的，所以寫程式的時候為了精簡程式碼，可以把狀態號為偶數的狀態全部整合在一起。而這一點剛好與發送過程的狀態剛好相反。

