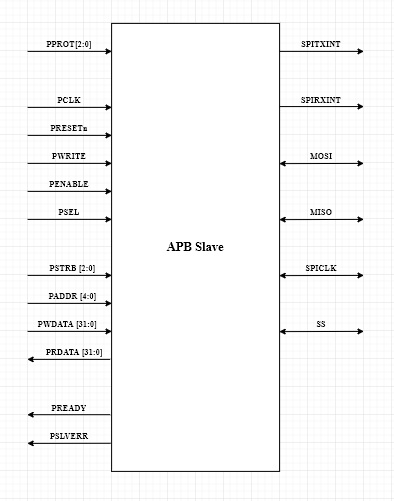
**VLSI Technology**

**4.4.2020**

**SPI Detail Description**

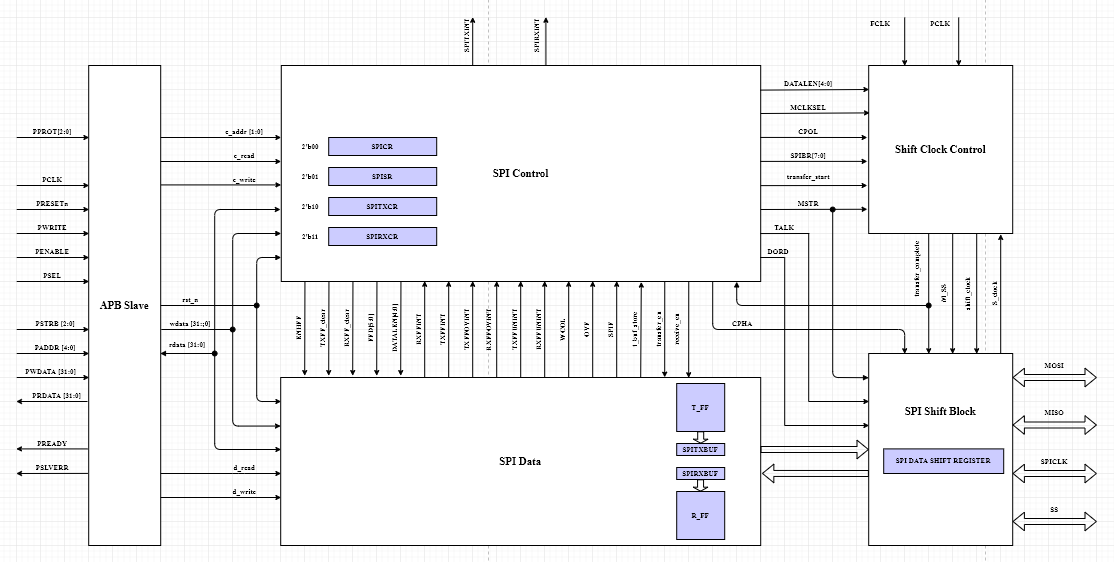
1. **Sơ đồ tín hiệu giao tiếp:**

****

**Bảng tín hiệu giao tiếp SPI**

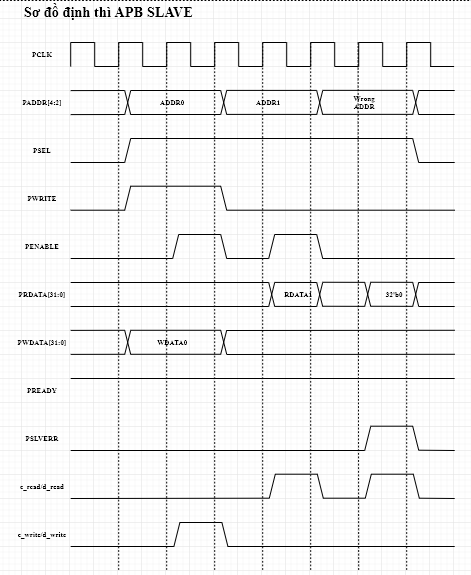
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Tín hiệu** | **Số bit** | **Chiều** | **Mô tả** |
|  | PCLK | 1 | Ngõ vào | Tín hiệu xung clock từ giao thức APB |
|  | PRESETn | 1 | Ngõ vào | Tín hiệu reset từ giao thức APB, tích cực thấp |
|  | PWRITE | 1 | Ngõ vào | Tín hiệu giúp phân biệt truy cập:  PWRITE = 0: đọc  PWRITE = 1: ghi |
|  | PSEL | 1 | Ngõ vào | Tín hiệu từ giao thức APB chọn ngoại vi SPI |
|  | PSTRB | 4 | Ngõ vào | Tín hiệu báo vị trí hợp lệ của các byte trên dữ liệu ghi,  Yêu cầu PSTRB luôn bằng 4’b1111 |
|  | PPROT | 3 | Ngõ vào | Không hỗ trợ |
|  | PENABLE | 1 | Ngõ vào | Tín hiệu báo pha ENABLE của một giao thức đọc ghi |
|  | PADDR | 5 | Ngõ vào | Tín hiệu chứa địa chỉ các thanh ghi SPI từ giao thức APB |
|  | PWDATA | 32 | Ngõ vào | Tín hiệu chứa dữ liệu cần truyền |
|  | PRDATA | 32 | Ngõ ra | Tín hiệu chứa dữ liệu đọc được |
|  | PREADY | 1 | Ngõ ra | Tín hiệu báo SPI sẵn sàng nhận hoặc truyền dữ liệu ở cạnh lên PCLK tiếp theo |
|  | PSLVERR | 1 | Ngõ ra | Tín hiệu báo có lỗi đọc ghi trong một APB transfer |
|  | MOSI | 1 | Ngõ ra hoặc vào tùy thuộc vào cài đặt Master/Slave | Master out Slave in |
|  | MISO | 1 | Ngõ vào hoặc ra tùy thuộc vào cài đặt Master/Slave | Master in Slave out |
|  | SPICLK | 1 | Ngõ ra hoặc vào tùy thuộc vào cài đặt Master/Slave | Clock của giao tiếp SPI |
|  |  | 1 | Ngõ vào | Tín hiệu chọn Slave. |
|  | SPITXINT | 1 | Ngõ ra | Tín hiệu ngắt SPI transfer |
|  | SPIRXINT | 1 | Ngõ ra | Tín hiệu ngắt SPI Receive |

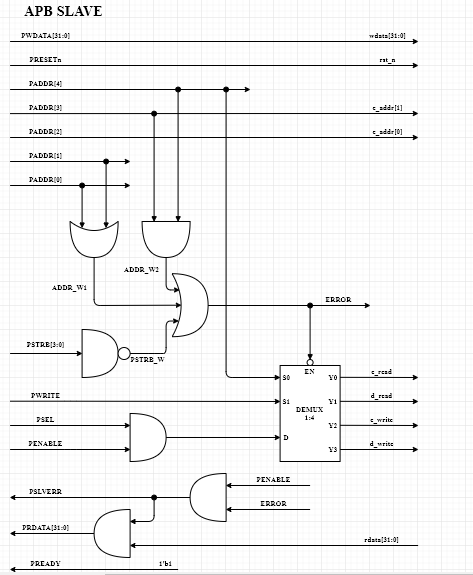
1. **Sơ đồ khối tổng quát:**

****

* Khối APB Slave: giao tiếp APB giữa bus APB và ngoại vi SPI
* Khối SPI Control: Chứa các thanh ghi cấu hình và trạng thái của SPI, điều khiển hoạt động của SPI
* Khối SPI Data: Chứa các FIFO truyền nhận (R\_FIFO, T\_FIFO), thanh ghi đệm truyền nhận (SPIRXBUF, SPITXBUF)
* Khối Shift Clock Control: chứa các chọn clock, bộ tạo clock dịch và tín hiệu SS (Slave\_Select) cho SPI,
* Khối SPI Shift Block: chứa các thanh ghi dịch, các chân bidirectional pin cho giao tiếp SPI

1. **Sơ đồ chi tiết:**
2. **APB Slave:**



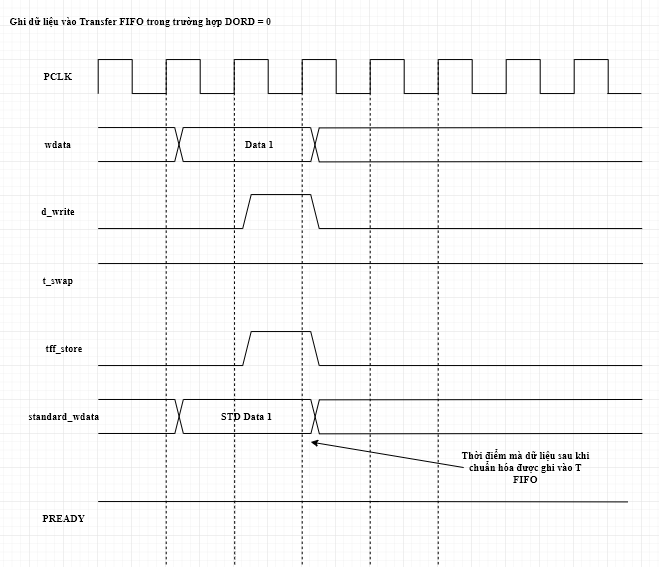


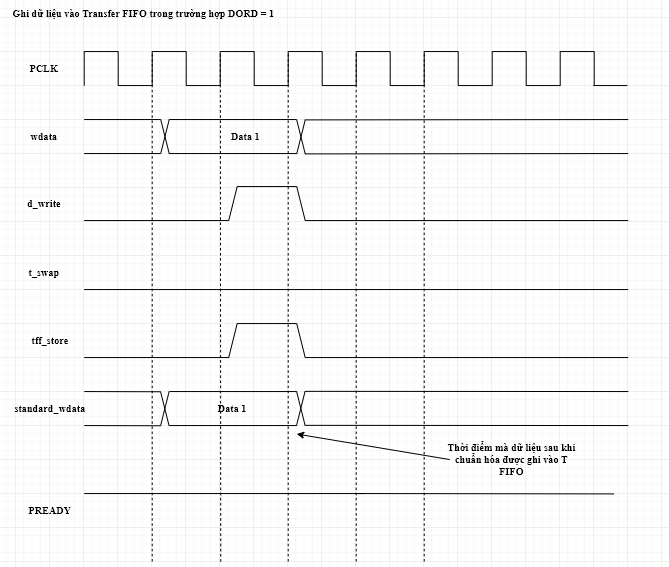
* APB Slave khi phát hiện có lỗi (Lỗi địa chỉ hoặc lỗi PSTRB) sẽ trả lời bằng một chu kỳ tích cực PSLVERR và dữ liệu trên PRDATA[31:0] = 32’b0
* Các tín hiệu c\_read, d\_read, d\_write, c\_write, chỉ được tích cực khi không có lỗi xảy ra. Mỗi chu kỳ đọc ghi của APB bus chỉ tích cực các tín hiệu này trong một chu kỳ PCLK
* PREADY luôn được đặt bằng 1

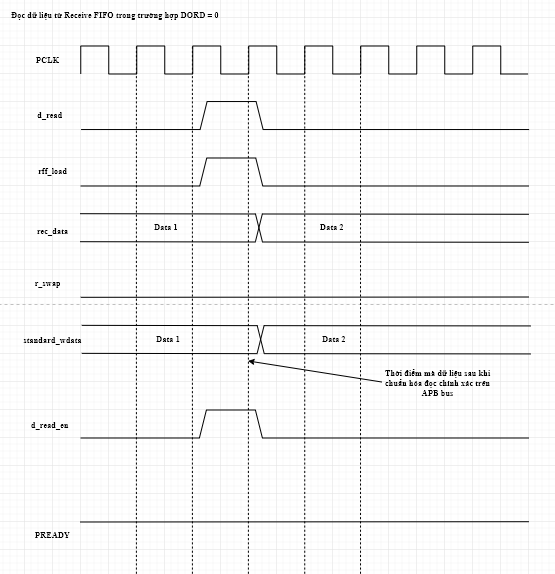
1. **SPI Data:**

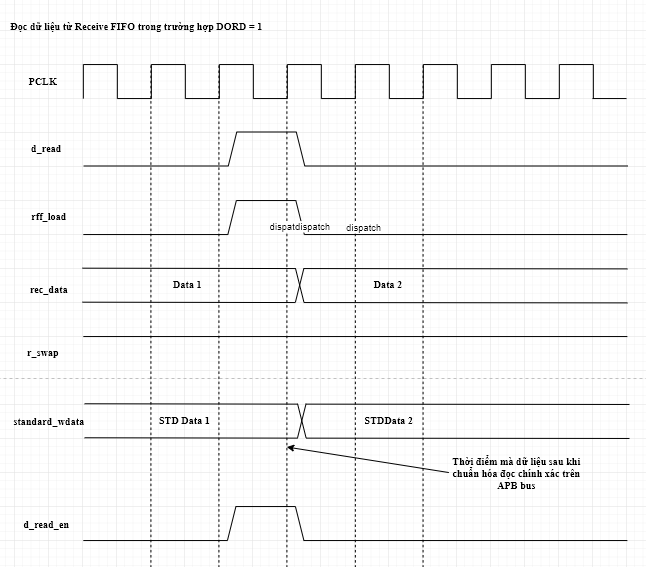
* Sơ đồ định thì các trường hợp hoạt động của SPI Data:

Note: Các tín hiệu điều khiển của SPI khi chỉ sử dụng buffer cũng tương tự như khi dùng FIFO, nên các hình minh họa bên dưới chỉ minh họa trường hợp dùng FIFO

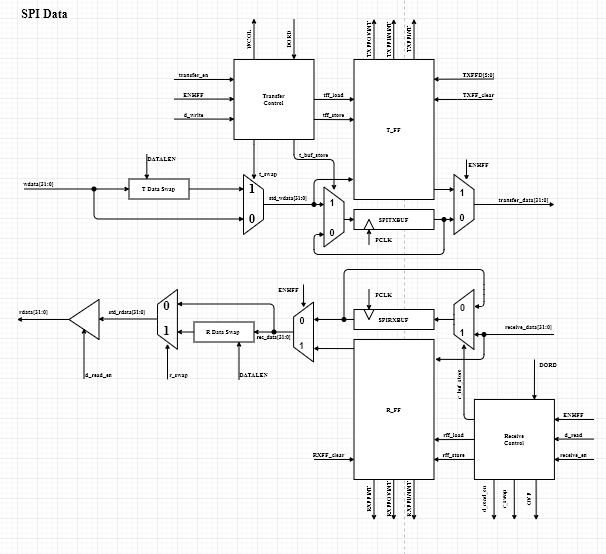


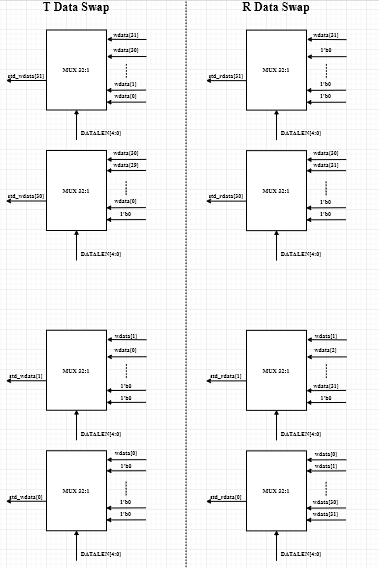


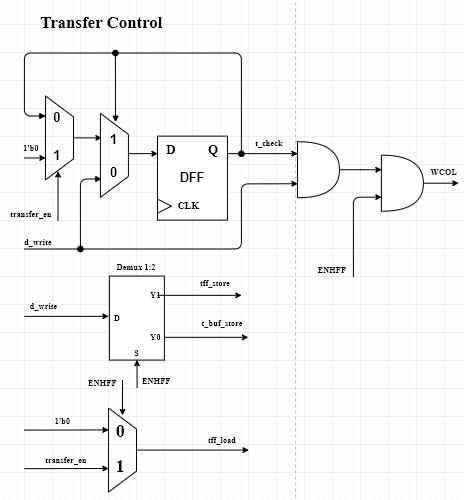


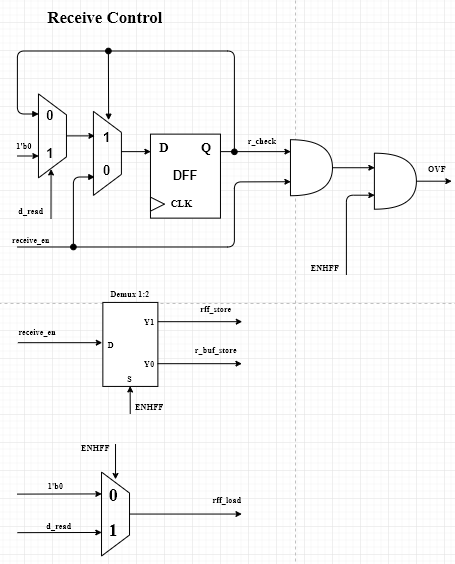


* Sơ đồ khối chi tiết các thành phần của SPI Data:





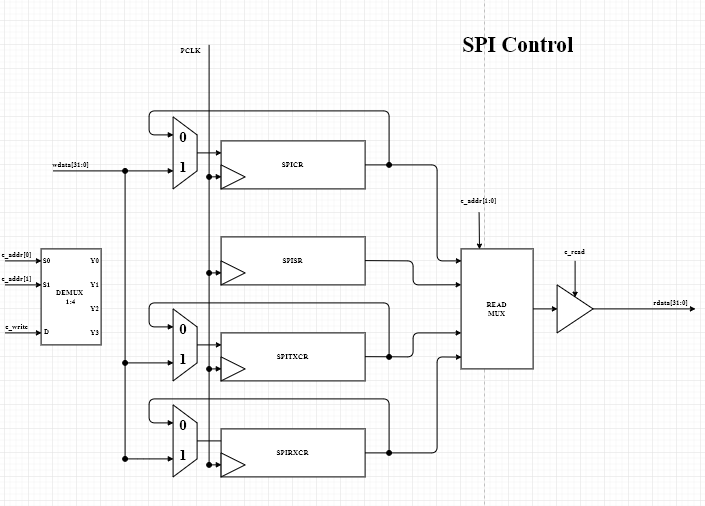


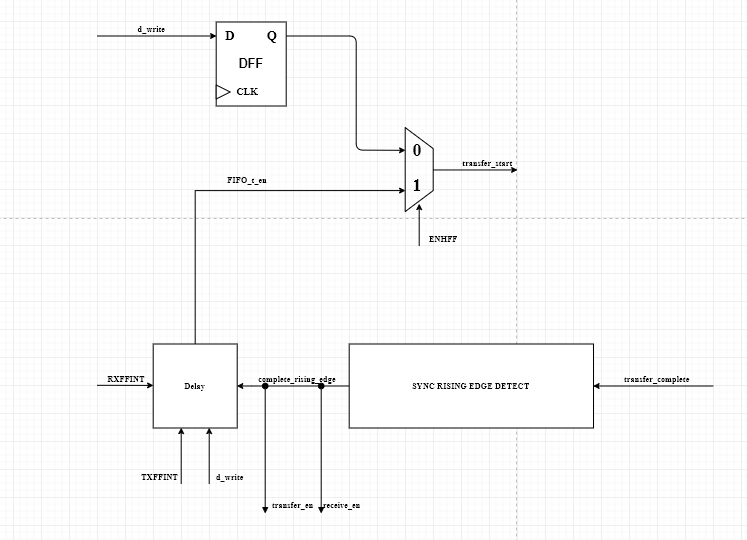
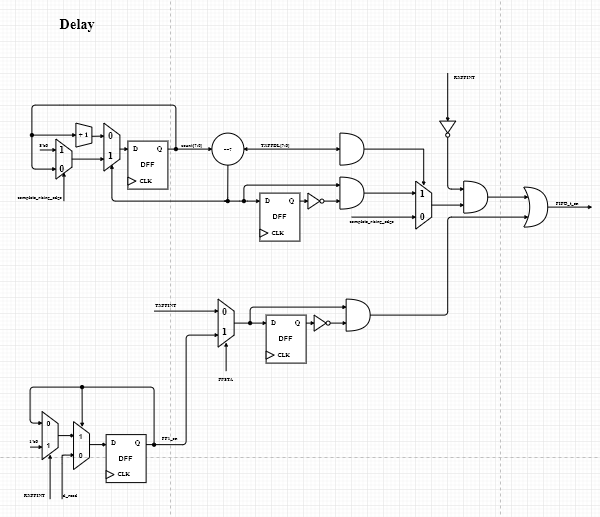


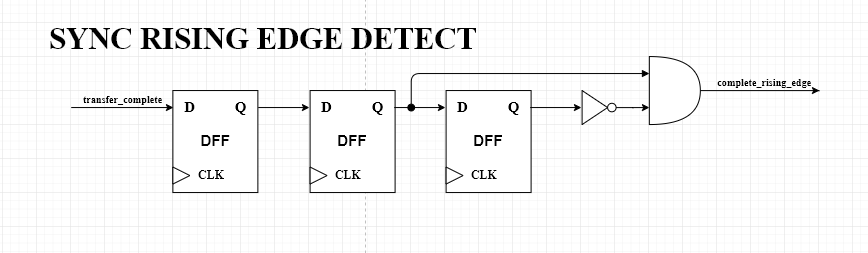
* Tín hiệu transfer\_en đến từ SPI Control giúp SPI Data biết rằng mình cần phải load một ô dữ liệu mới từ FIFO truyền lên đường truyền transfer\_data[31:0].
* Khi không sử dụng FIFO, SPITXBUF ghi đè dữ liệu mới mỗi khi có một xung d\_write tích cực mức cao.
* Tín hiệu Receive\_en đến từ SPI Control giúp SPI Data biết rằng có một dữ liệu mới được nhận và sẵn sàng được ghi vào FIFO nhận hoặc SPIRXBUF
* Khối T Data Swap được sử dụng để chuyển dữ liệu cần truyền về dạng left-justified khi ở chế độ dịch MSB trước
* Khối R Data Swap được sử dụng để chuyển dữ liệu cần đọc về dạng right-justified khi ở chế độ dịch LSB trước

1. **SPI Control:**

* Sơ đồ khối chi tiết các thành phần SPI control



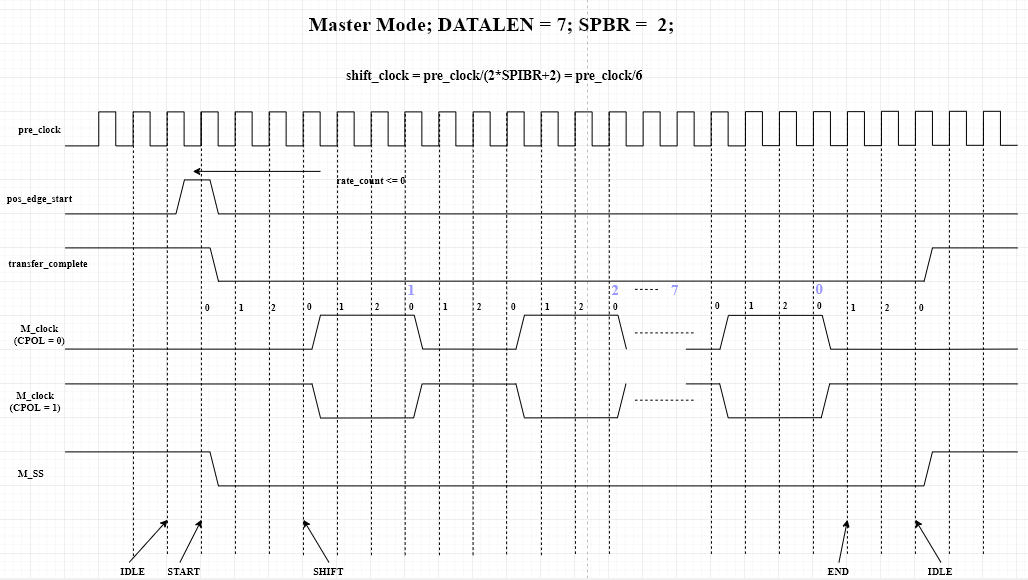


* Việc ghi từ APB vào SPISR (SPI Status Register) không làm thay đổi nội dung thanh ghi này
* Khối SYNC RISING EDGE DETECT được dùng để đồng bộ tín hiệu shift\_complete được gửi từ miền clock dịch sang. Tín hiệu tạo ra là một xung tích cực mức cao complete\_rising\_edge báo hiệu cho SPI biết là một chuỗi dữ liệu vừa được gửi xong.
* Khối Delay tạo delay giữa các lần kết thúc truyền nhận và bắt đầu sự kiện truyền mới nhằm chờ Slave kịp sử lý, đưa dữ liệu mới lên đường truyền sau khi một chu kỳ truyền nhận kết thúc

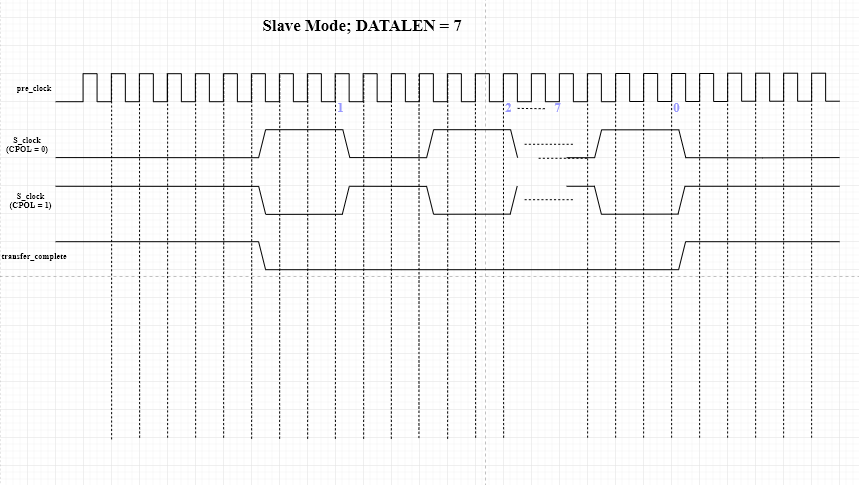
1. **Shift Control Clock:**

* Sơ đồ định thì cho một hoạt động của khối Shift Control Clock

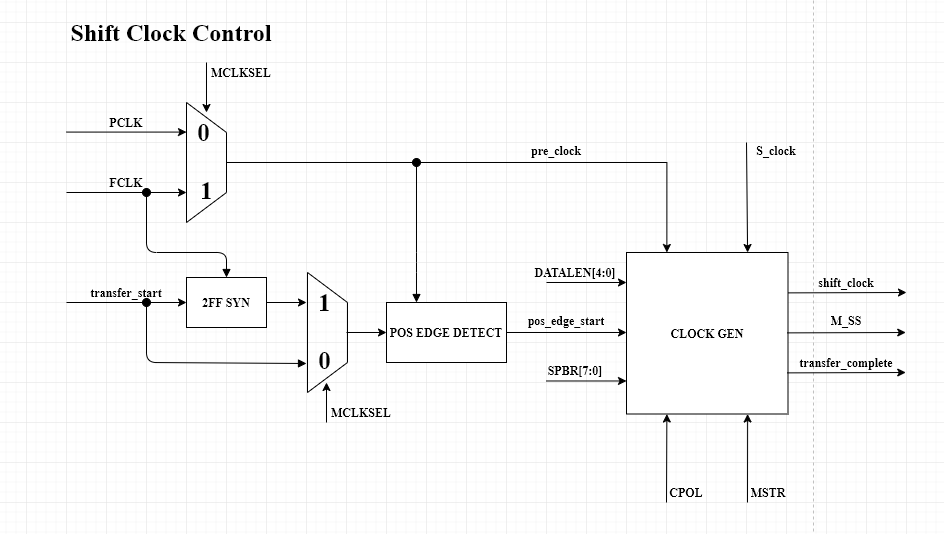
1. Minh họa cho ví dụ hoạt động ở chế độ Master, độ dài dữ liệu là 8 bit, tốc độ truyền là src\_clock/6

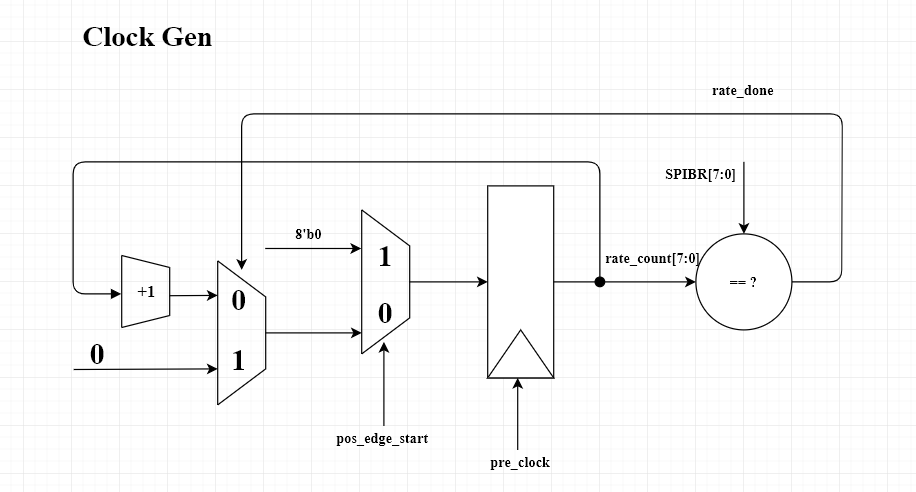


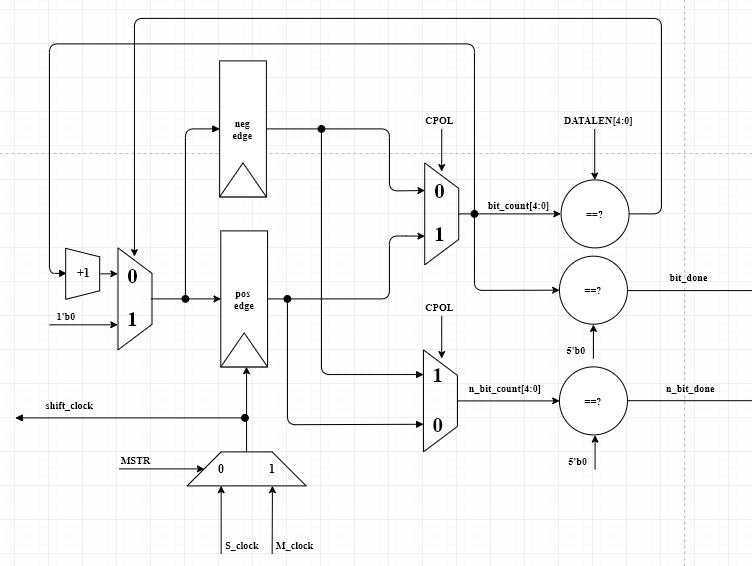
1. Minh họa cho ví dụ hoạt động ở chế độ Slave

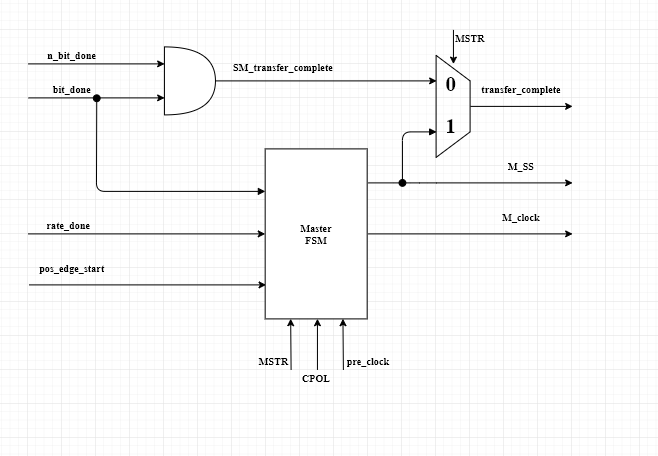


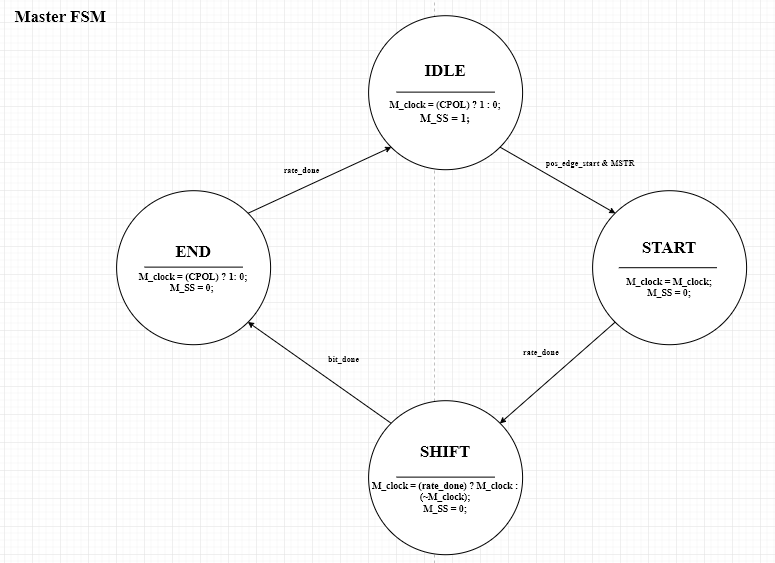
* Sơ đồ khối chi tiết của Shift Clock Control





****

****

****

* Các vấn đề ở khối Shift Clock Control:
  + Khi nguồn clock được chọn để tạo clock dịch là PCLK thì tín hiệu transfer\_start được gửi từ SPI Control sang không cần đồng bộ. Tuy nhiên nếu nguồn clock được chọn là FCLK thì tín hiệu transfer\_start cần được đồng bộ. Vấn đề ở đây là tín hiệu transfer\_start chỉ được tích cực trong một chu kỳ PCLK do đó để đồng bộ được thì clock FCLK phải thỏa điều kiện:

TPCLK ≥ 2TFCLK

* FCLK≥ 2PCLK (1)
  + Tương tự cho việc đồng bộ tín hiệu transfer\_complete được gửi từ khối Shift Clock Control sang khối SPI Control (tín hiệu này chỉ bằng 0 khi việc dịch dữ liệu đang diễn ra), để thỏa mãn yêu cầu đồng bộ tần số dịch phải thỏa điều kiện:

1. Chế độ Master:

(DATALEN + 1.5)Tshift\_clock ≥ 2TPCLK

=> shift\_clock ≤

=> ≤ (DATALEN + 1.5)PCLK

Với clock nguồn là PCLK thì công thức trên luôn thỏa. Tuy nhiên khi clock nguồn là FCLK:

FCLK ≤ (DATALEN + 1.5)PCLK (2)

1. Chế độ Slave:

(DATALEN + 0.5)Tshift\_clock ≥ 2TPCLK

=> shift\_clock ≤

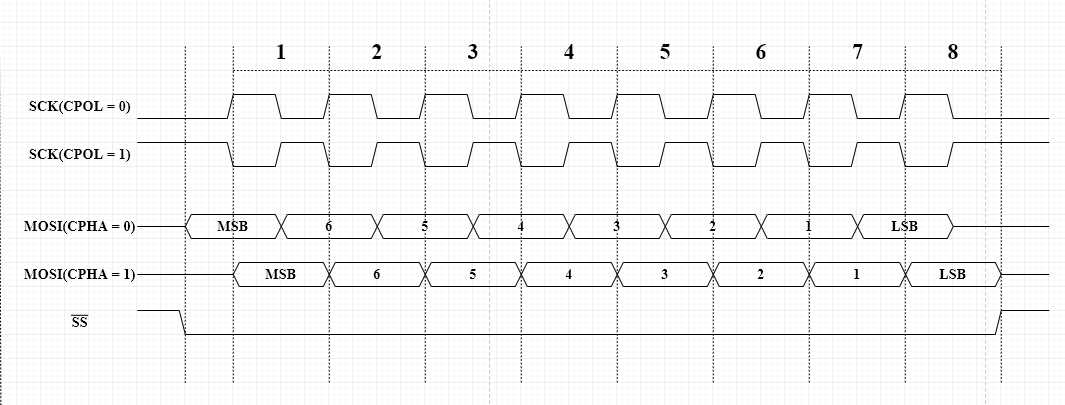
=> S\_clock ≤ (3)

Từ (1) (2) (3) ta có các điều kiện cho các nguồn clock:

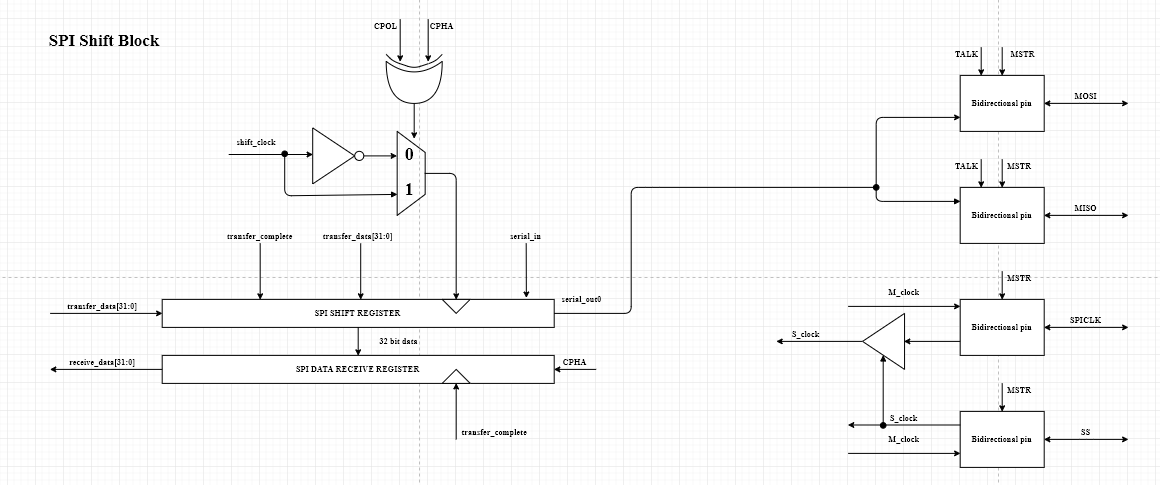
* 2PCLK ≤ FCLK ≤ (DATALEN + 1.5)PCLK(2\*SPBR+2)
* S\_clock ≤

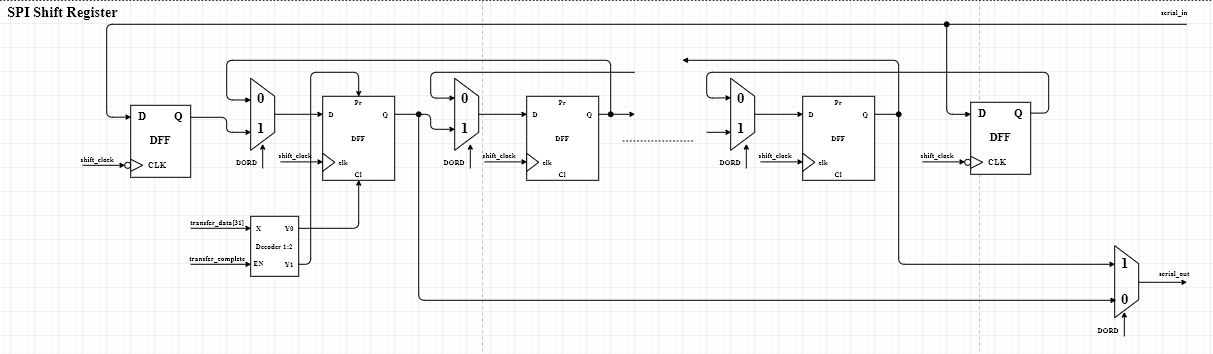
1. **SPI Shift Block:**

* Sơ đồ định thì khi SPI hoạt động ở chế độ Master, truyền 8 bit



* Sơ đồ chi tiết khối SPI Shift Block





* Tín hiệu S\_clock chỉ được truyền vào SPI khi S\_clock bằng 0
* Dữ liệu nhận được sau một chu kỳ truyền nhận sẽ được truyền song song vào thanh ghi SPI DATA RECEIVE REGISTER nhờ xung cạnh lên của tín hiệu transfer\_complete
* Khi transfer\_complete bằng 1 dữ liệu từ transfer\_data[31:0] sẽ được đưa vào thanh ghi dịch