

Digital Design Practical Exercise 3:

Sequential Logic Modeling

Yêu cầu báo cáo:

Trong báo cáo nên có các phần sau:

- Source code VHDL của các phần thực hành (những phần phải viết code)
- Kết quả chạy mô phỏng, giải thích kết quả (dưới dạng dễ hiểu và dễ thấy)
- Trả lời các câu hỏi trong phần hướng dẫn
- Liên kết đến các phần lý thuyết đã học và những kiến thức mà các bạn thấy

Chuẩn bị

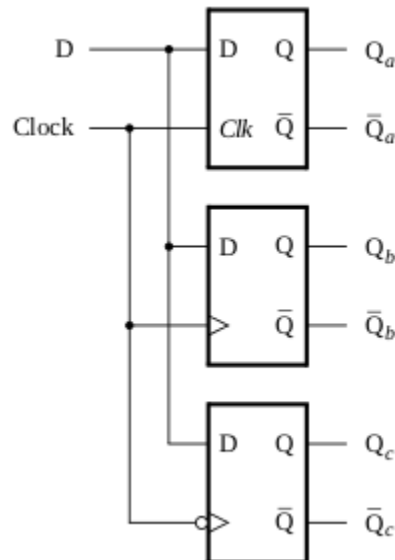
Tạo thư mục lab3 trong ổ C.

Chuyển thư mục làm việc trong modelsim đến thư mục này bằng cách dùng lệnh:

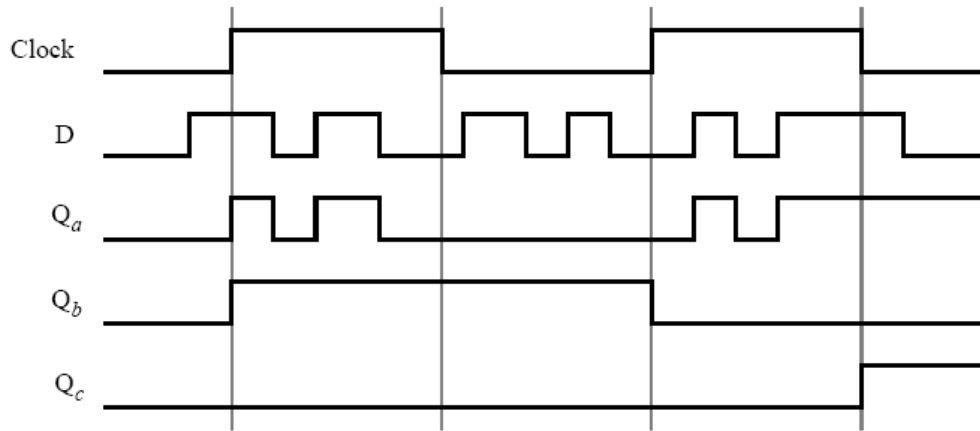
```
cd C:/lab3
```

Phần 1: Các loại flip-flop

Cho mạch điện như trong Hình 1. Hình 2 thể hiện quan hệ giữa input và output của mạch điện này với các đầu ra Q_a , Q_b , Q_c .



Hình 1. Mạch điện của ba phần tử bộ nhớ khác nhau.



Hình 2. Lược đồ thời gian của mạch điện trong Hình 1.

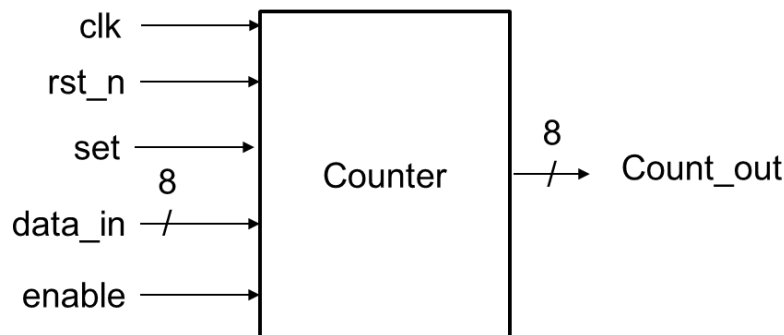
- Hoàn thành kịch bản kiểm tra của các flip-flop ở Hình 1 trong tập tin `flipflops_tb.vhd` bằng cách tạo ra dạng sóng của tín hiệu *d* và tín hiệu *clk* ; biên dịch và chạy mô phỏng thiết kế.
- Nhận xét về sự khác nhau trong tín hiệu đầu ra theo quan hệ với tín hiệu đầu vào và tín hiệu xung nhịp đồng hồ *clk*. Các phần tử trong mạch điện ở Hình 1 thuộc loại nào? Gọi tên các mạch điện đó.
- Đọc code của tập tin `flipflops.vhd` từ đó trả lời các câu hỏi sau:
 - Danh sách nhạy cảm (sensitivity list) là gì?
 - Khi nào thì một process được chạy trong ngôn ngữ VHDL. Nêu ứng dụng của process trong VHDL để miêu tả các mạch điện tổ hợp và các mạch tuần tự.

Phần 2: Mô hình hóa và mô phỏng bộ đếm

Trong Hình 3 là các đầu vào và đầu ra của một bộ đếm bao gồm các tín hiệu:

STT	Tên tín hiệu	Chế độ	Độ rộng bit	Miêu tả
1	Clk	In	1	Tín hiệu xung nhịp đồng hồ của bộ đếm. Đầu ra của bộ đếm thay đổi ở sườn lên của tín hiệu clk.
2	Rst_n	In	1	Tín hiệu reset. - Khi <code>rst_n = '0'</code> bộ đếm ở chế độ reset. - Khi <code>rst_n = '1'</code> , bộ đếm ở chế độ hoạt động.
3	Set	In	1	- Khi <code>set = '1'</code> , giá trị đầu ra của bộ đếm bằng <code>data_in</code> . - Khi <code>set = '0'</code> , các chế độ hoạt động khác.
4	Data_in	In	8	Giá trị được nạp vào các thanh ghi của bộ đếm khi <code>set = '1'</code>
5	Enable	In	1	Tín hiệu cho phép đếm - Khi <code>enable = '1'</code> giá trị của các thanh ghi tăng 1 đơn vị sau mỗi sườn lên của tín hiệu clk.

				- Khi <i>enable</i> = '0' và các tín hiệu khác ở chế độ không kích hoạt, các thanh ghi giữ nguyên giá trị.
6	Count_out	Out	8	Giá trị đầu ra của bộ đếm



Hình 3. Các chân vào ra của bộ đếm 8 bit.

Trong phần này, chúng ta sẽ mô hình hóa một bộ đếm tiến sử dụng ngôn ngữ VHDL thực hiện các chức năng như đã được miêu tả trong phần trên.

Gợi ý:

Có thể dùng các phép cộng bằng cách dùng kiểu dữ liệu unsigned trong package numeric_std của thư viện IEEE.

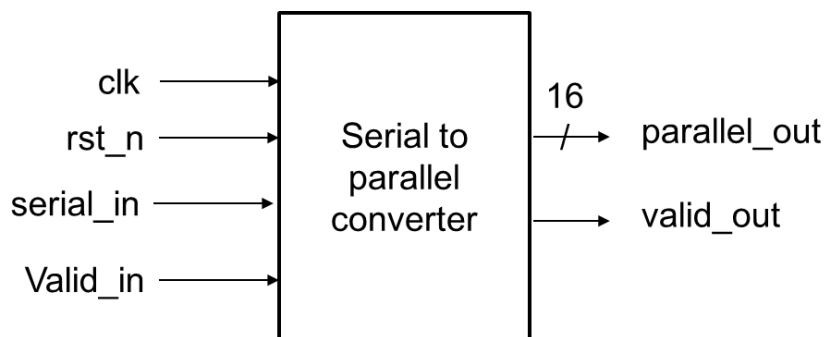
1. Mô hình hóa hành vi của bộ đếm được miêu tả ở trên dùng ngôn ngữ vhdl. Tạo kịch bản kiểm tra cho bộ đếm này, chạy mô phỏng và kiểm tra lại kết quả mô phỏng.
Gợi ý: có thể dùng lệnh *force* hoặc viết *testbench*.
2. Các tín hiệu điều khiển *rst_n*, *set* trong thiết kế của các bạn là tín hiệu điều khiển đồng bộ (synchronous) hay là bất đồng bộ (asynchronous). Viết miêu tả của các tín hiệu này theo cách khác (đồng bộ hoặc bất đồng bộ).
3. **Bonus:** thay đổi thiết kế thực hiện trong mục 1 để cho phép đếm tiến và đếm lùi.

Phần 3: Ứng dụng bộ đếm và thanh ghi dịch

Trong phần này, chúng ta sẽ thiết kế một bộ chuyển đổi dữ liệu từ dạng tuần tự sang dạng song song (serial to parallel converter). Bộ chuyển đổi dữ liệu từ tuần tự sang song song nhận đầu vào là 1 bit dữ liệu và sẽ đưa ra đầu ra 16-bit dữ liệu mỗi khi nhận đủ 16 giá trị đầu vào. Các tín hiệu vào ra của bộ chuyển đổi tuần tự sang song song được miêu tả như sau:

STT	Tên tín hiệu	Chế độ	Độ rộng (bit)	Miêu tả
1	Clk	In	1	Tín hiệu xung nhịp của cả hệ thống
2	Rst_n	In	1	Tín hiệu reset. <ul style="list-style-type: none"> - Khi <i>rst_n</i> = '0', mạch điện ở chế độ reset, các thanh ghi được nạp giá trị mặc định. - Khi <i>rst_n</i> = '1', mạch điện ở chế độ hoạt động

3	Serial_in	In	1	Tín hiệu đầu vào
4	Valid_in	In	1	Tín hiệu thông báo tín hiệu đầu vào <i>serial_in</i> có chứa dữ liệu. - Nếu <i>valid_in</i> = '1' có dữ liệu đầu vào (dữ liệu của tín hiệu <i>serial_in</i> cần được lưu vào thanh ghi). - Nếu <i>valid_in</i> = '0', không có dữ liệu đầu vào (không lưu giá trị của <i>serial_in</i> vào thanh ghi).
5	Parallel_out	Out	16-bit	16 bit đầu ra song song
6	Valid_out	Out	1	Tín hiệu thông báo tín hiệu đầu ra có chứa dữ liệu. - <i>valid_out</i> = '1' mỗi khi có 16-bit dữ liệu đầu ra. - <i>valid_out</i> = '0', không có dữ liệu đầu ra.

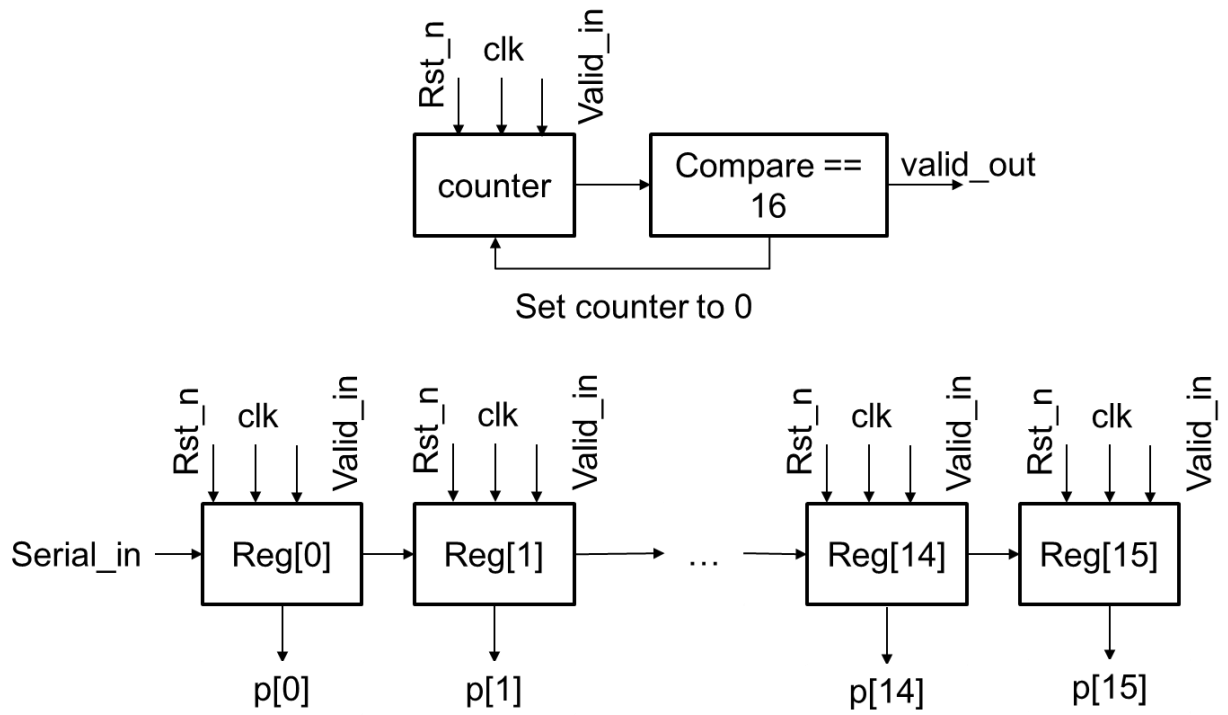


Hình 4. Các chân vào ra của bộ chuyển đổi tuần tự sang song song 16 bit.

1. Mô hình hóa hành vi của bộ chuyển đổi serial-to-parallel converter dùng ngôn ngữ VHDL. Viết kịch bản kiểm tra cho bộ chuyển đổi.
2. Tín hiệu *rst_n* có vai trò gì trong thiết kế này.
3. **Bonus:** Thiết kế bộ chuyển tuần tự sang song song cho n-bit tín hiệu đầu vào với n là 1 số nguyên dương có thể thay đổi tại thời điểm sử dụng thiết kế.

Gợi ý:

Sơ đồ khối của bộ chuyển đổi được trình bày trong Hình 5.



Hình 5. Sơ đồ khối của bộ chuyển đổi tuần tự sang song song.