|  |  |  |  |
| --- | --- | --- | --- |
| Họ tên sinh viên | MSSV | Lớp (thứ - tiết) |  |
|  |  |  |

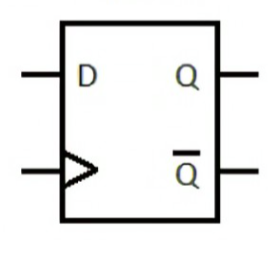
Chú ý: Sinh viên thay bằng QR code của mã số sinh viên (ví dụ: 23119012), có thể tham khảo tại https://barcode.tec-it.com)

Quick question : chapter 5

*Lưu ý: Trong mỗi thiết kế yêu cầu sinh viên thực hiện*

* *Sơ đồ khối (nguyên lý, cấu trúc)*
* *Bảng trạng thái*
* *Mô tả bằng ngôn ngữ Verilog cho module cần thiết kế,*
* *Mô tả Verilog cho module dùng để kiểm tra thiết kế*
* *Kết quả mô phỏng quá trình kiểm tra, có phân tích*
* *Module test được đặt tên theo cú pháp: tensv\_testbench\_tenmodule, ví dụ để test module encoder, sinh viên Nguyen Van An phải đặt tên module test như sau: An\_testbench\_encoder. Các kết quả mô phỏng phải được chụp màn hình bao gồm cả tên của module test trong đó có tên sinh viên thì mới hợp lệ*

1. Thiết kế và mô phỏng kiểm chứng mạch Flip Flop D



Bảng trạng thái Q(n+1) = D(n)

|  |  |  |
| --- | --- | --- |
| Clk | D | Q(n+1) |
| 0 | x | Q(n) |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

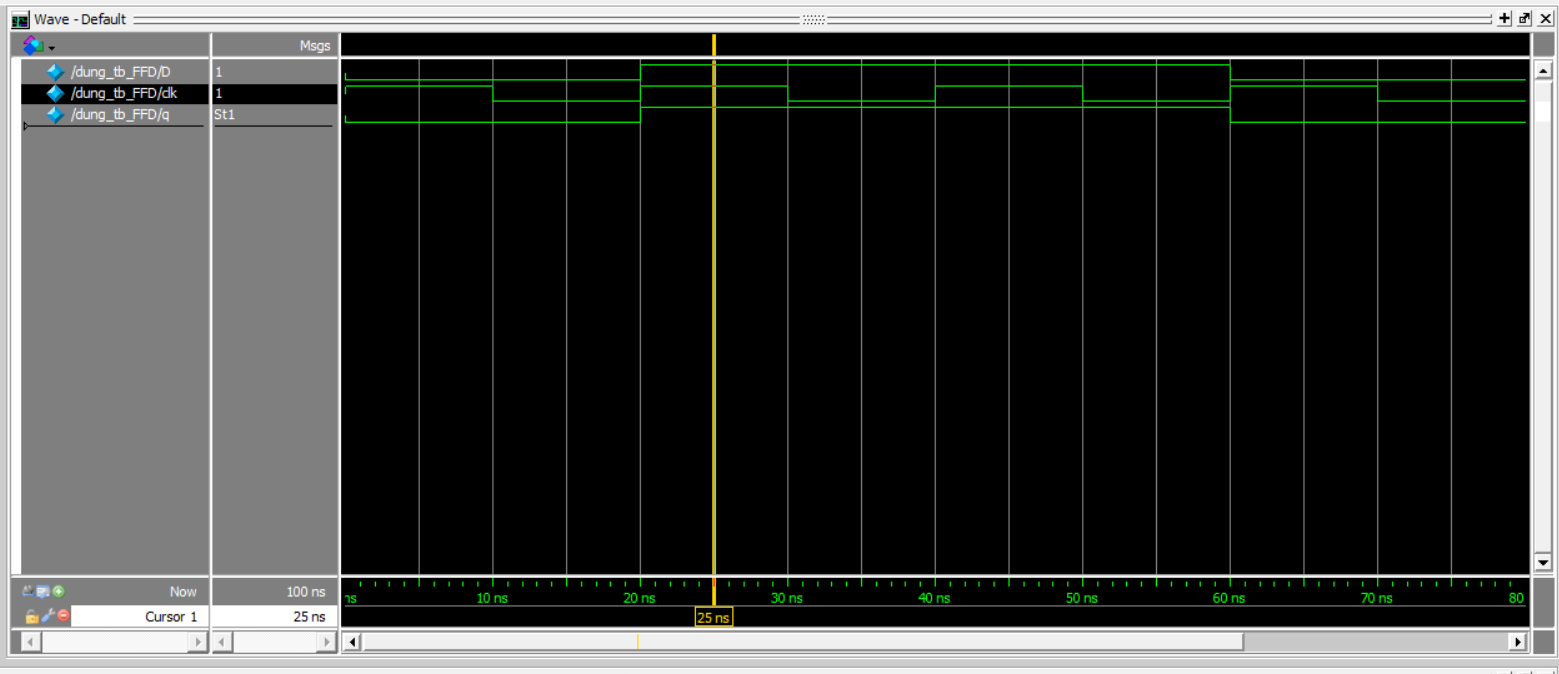
Mô tả bằng ngôn ngữ Verilog

|  |
| --- |
| module FF\_D(input wire d,clk, output reg q);  always @(posedge clk) // positive edge  begin      q <= d // Non-Blocking Assignment  end  endmodule |

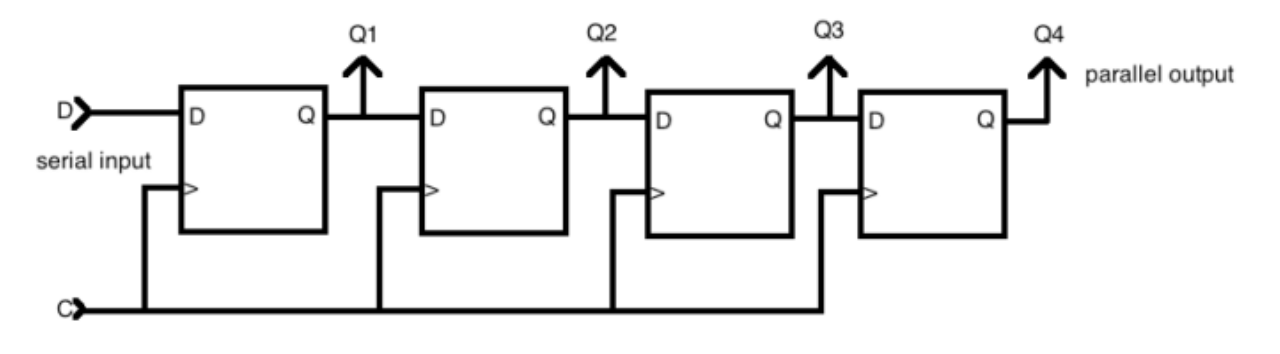
Mô tả Verilog cho module dùng để kiểm tra thiết kế

|  |
| --- |
| `timescale 1ns/1ns // define timescale  module dung\_tb\_FFD();  reg D;  reg clk;  wire q;  initial begin  clk = 1;  D = 0;  end  always forever #20 clk = ~clk;  always forever #40 D = ~D;  FF\_D m0(D,clk,q);  endmodule |

Kết quả mô phỏng



1. Thiết kế và mô phỏng kiểm chứng thanh ghi dịch 8 bit sử dụng FlipFlop D



1. Thiết kế và mô phỏng kiểm chứng mạch đếm đồng bộ 8 bit sử dụng FlipFlop T

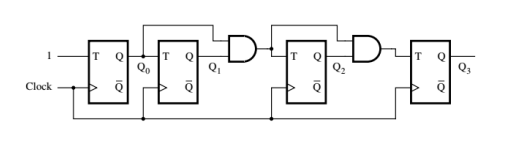


Figure : Mạch đếm đồng bộ 8 bit này sẽ sử dụng T-Flip-Flop (T-FF) và sẽ đếm từ 00000000 đến 11111111 (từ 0 đến 255 trong hệ thập phân). Mỗi xung đồng hồ sẽ làm tăng giá trị của bộ đếm lên 1.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Clock Pulse No | Q7 | Q6 | Q5 | Q4 | Q3 | Q2 | Q1 | Q0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 8 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 10 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 11 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 12 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 14 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 15 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 16 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 255 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |