

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



Tài liệu hướng dẫn thực hành:

THIẾT KẾ LUẬN LÝ SỐ

LƯU HÀNH NỘI BỘ

Thành phố Hồ Chí Minh – Tháng 09/2014

MỤC LỤC

| | |
|--|---|
| Tài liệu hướng dẫn thực hành: | 1 |
| THIẾT KẾ LUẬN LÝ SỐ | 1 |
| Bài 1. THIẾT KẾ MẠCH ĐẾM ĐỒNG BỘ CÓ KHẢ NĂNG NẠP GIÁ TRỊ BAN ĐẦU | 1 |
| 1.1 Mục tiêu | 1 |
| 1.2 Nội dung thực hành | 1 |
| 1.3 Sinh viên chuẩn bị trước ở nhà | 2 |
| 1.4 Hướng dẫn thực hành | 2 |
| 1.5 Bài tập ôn tập | 3 |
| Bài 2. THIẾT KẾ MẠCH TUẦN TỰ BẰNG MÔ HÌNH MÁY TRẠNG THÁI HỮU HẠN | 4 |
| 2.1 Mục tiêu | 4 |
| 2.2 Nội dung thực hành | 4 |
| 2.3 Sinh viên chuẩn bị | 5 |
| 2.4 Hướng dẫn thực hành | 5 |
| 2.5 Bài tập ôn tập | 6 |
| Bài 3. THIẾT KẾ ALU | 7 |
| 3.1 Mục tiêu | 7 |
| 3.2 Nội dung thực hành | 7 |
| 3.3 Sinh viên chuẩn bị | 7 |
| 3.4 Hướng dẫn thực hành | 8 |

| | | |
|--------|------------------------------|----|
| 3.5 | Bài tập ôn tập | 8 |
| Bài 4. | THIẾT KẾ DATAPATH ĐƠN GIẢN | 9 |
| 4.1 | Mục tiêu | 9 |
| 4.2 | Nội dung thực hành | 9 |
| 4.3 | Sinh viên chuẩn bị | 9 |
| 4.4 | Hướng dẫn thực hành | 10 |
| 4.5 | Bài tập ôn tập | 10 |
| Bài 5. | THIẾT KẾ SIMPLE CONTROL UNIT | 11 |
| 5.1 | Mục tiêu | 11 |
| 5.2 | Nội dung thực hành | 11 |
| 5.3 | Sinh viên chuẩn bị | 11 |
| 5.4 | Hướng dẫn thực hành | 12 |
| 5.5 | Bài tập ôn tập | 12 |
| Bài 6. | THIẾT KẾ SIMPLE PROCESSOR | 13 |
| 6.1 | Mục tiêu | 13 |
| 6.2 | Nội dung thực hành | 13 |
| 6.3 | Sinh viên chuẩn bị | 13 |
| 6.4 | Hướng dẫn thực hành | 13 |

NỘI QUY THỰC HÀNH

1. Sinh viên tham dự đầy đủ các buổi thực hành theo quy định của giảng viên hướng dẫn (GVHD) (6 buổi với lớp học cách tuần và 10 buổi với lớp học liên tục).
2. Sinh viên phải chuẩn bị các phần từ mục 1 tới mục 3 trong phần “Hướng dẫn thực hành” trước khi đến lớp. GVHD sẽ kiểm tra bài chuẩn bị của sinh viên trong 15 phút đầu của buổi học (nếu không có bài chuẩn bị -> tính vắng buổi học đó).
3. Sinh viên phải chạy thiết kế của mình trên kit DE2 sau khi thiết kế xong để GVHD check hoàn thành bài thực hành.
4. Sinh viên làm các bài tập ôn tập để được cộng điểm thực hành, bài tập ôn tập sẽ được GVHD check khi sinh viên có yêu cầu trong buổi học liên sau bài thực hành đó.

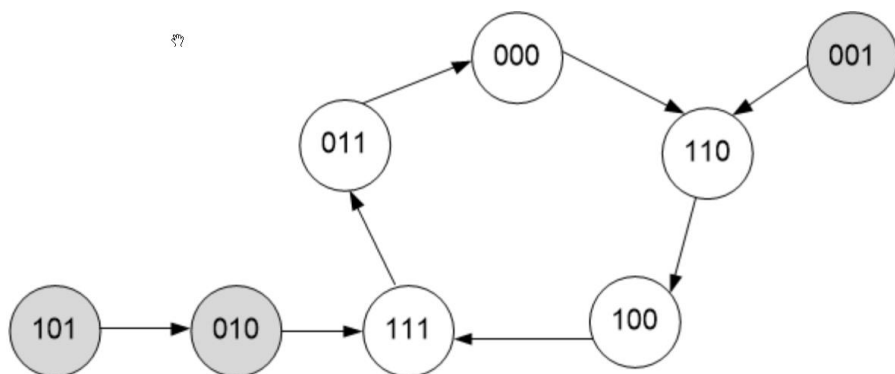
Bài 1. THIẾT KẾ MẠCH ĐẾM ĐỒNG BỘ CÓ KHẢ NĂNG NẠP GIÁ TRỊ BAN ĐẦU

1.1 Mục tiêu

Mục tiêu của bài này là giúp sinh viên thiết kế một mạch đếm đồng bộ. Sau đó, sẽ hiện thực thiết kế trên kit DE2 với giá trị ngõ ra của bộ đếm được hiển thị giá trị đếm trên LED đơn và LED 7 đoạn.




1.2 Nội dung thực hành

Trong Lab này, sinh viên sử dụng các FF-T để thiết kế một mạch đếm đồng bộ theo một chu trình đếm cho trước, với giá trị ban đầu của mạch đếm được nạp vào thông qua các chân Preset và Clear của các Flip-Flop (FF), có chu trình đếm như sơ đồ trong Hình 1-1.








Hình 1-1 Sơ đồ chuyển trạng thái của bộ đếm

1.3 Sinh viên chuẩn bị trước ở nhà

-  Lập bảng chuyển trạng thái và bảng kích thích cho mạch đếm theo sơ đồ ở Hình 1-1.
-  Tìm phương trình ngõ vào của các FF-T.
-  Vẽ mạch theo các phương trình có vào, bao gồm cả mạch nạp dữ liệu ban đầu cho bộ đếm thông qua các ngõ vào Preset và Clear bất đồng bộ.

1.4 Hướng dẫn thực hành

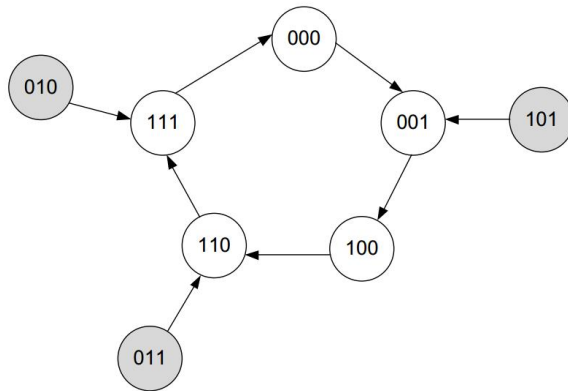
1. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab1-MSSV
2. Thực thi mạch đã thiết kế trong phần chuẩn bị. Với:
 -  KEY[0]: xung Clock của mạch
 -  KEY[1]: chân cho phép nạp (LE: Load Enable)
 -  SW[2:0]: giá trị dữ liệu ban đầu cần nạp vào
 -  LEDG[2:0]: hiển thị kết quả ngõ ra của bộ đếm
 -  Thêm bộ giải mã BCD_to_LED7 vào mạch để hiển thị kết quả ra Led 7 đoạn.
3. Gán pin cho mạch trên
4. Biên dịch để tổng hợp và tạo file .sof

5. Nạp file thực thi lên Kit DE2 và kiểm tra hoạt động của mạch

1.5 Bài tập ôn tập

1. Sử dụng FF-D để thiết kế mạch đếm đồng bộ có khả năng nạp giá trị bang đầu theo sơ đồ trong Hình 1-2 và hiện thực thiết kế chạy trên Kit De2.

87



Hình 1-2 Sơ đồ chuyển trạng thái bài tập 1

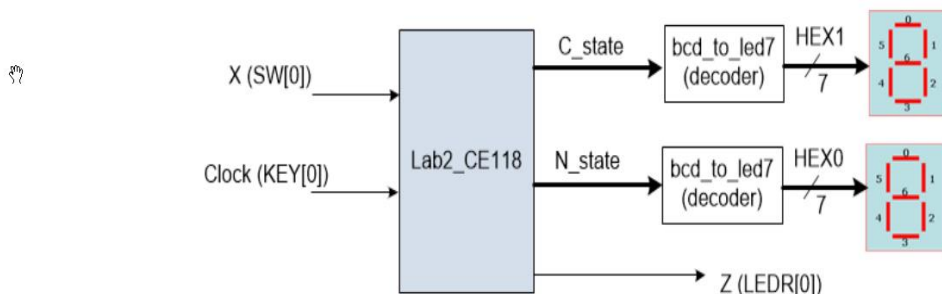
Bài 2. THIẾT KẾ MẠCH TUẦN TỰ BẰNG MÔ HÌNH MÁY TRẠNG THÁI HỮU HẠN

2.1 Mục tiêu

Sau khi thực hiện xong bài thực hành, sinh viên sẽ có khả năng thiết kế một mạch tuần tự bằng mô hình máy trạng thái hữu hạn (FSM). Mạch sẽ được thiết kế, thử nghiệm và mô phỏng trên phần mềm Quartus II

2.2 Nội dung thực hành

Sinh viên sử dụng FF-D để thiết kế một mạch tuần tự theo mô hình máy trạng thái kiểu Moore có sơ đồ khối như trong Hình 2-1 bên dưới. Hệ tuần tự này có chức năng phát hiện 2 bit ngõ vào (X) liên tiếp khác nhau thì ngõ ra $Z = 1$.







Với C_state : trạng thái hiện tại





N_state : trạng thái kế tiếp

Hình 2-1 Sơ đồ khối của mạch

2.3 Sinh viên chuẩn bị

-  Vẽ lưu đồ chuyển trạng thái của mạch tuần tự trên
-  Lập bảng kích thích của mạch
-  Tìm phương trình ngõ vào của các FF-D và ngõ ra Z của mạch
-  Vẽ mạch trên Quartus II

2.4 Hướng dẫn thực hành

1. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab2-MSSV
2. Thực thi mạch đã thiết kế trong phần chuẩn bị. Với:
 -  KEY[0]: xung Clock của mạch
 -  SW[1]: tín hiệu ngõ vào của mạch
 -  LEDR[0]: hiển thị kết quả ngõ ra của mạch
 -  HEX0[6:0], HEX1[6:0]: hiển thị kết quả lần lượt của trạng thái kế tiếp và trạng thái hiện tại.
3. Gán pin cho mạch trên
4. Biên dịch để tổng hợp và tạo file .sof
5. Nạp file thực thi lên Kit DE2 và kiểm tra hoạt động của mạch

2.5 Bài tập ôn tập

1. Thực hiện lại thiết kế mạch bên trên theo mô hình máy trạng thái kiểu Mealy.

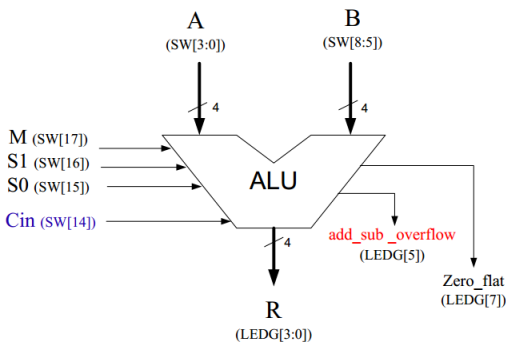
Bài 3. THIẾT KẾ ALU

3.1 Mục tiêu

Bài lab sẽ giúp sinh viên thiết kế được một ALU 4-bit đơn giản trên phần mềm LogiSim theo một clip hướng dẫn. Sau đó sinh viên sẽ tự thiết kế lại và kiểm chứng hoạt động của ALU này trên phần mềm Quartus II và kit DE 2.


3.2 Nội dung thực hành

Thiết kế bộ ALU 4-bit có sơ đồ khối và các chức năng như Hình 3-1 bên dưới. Trong đó, lệnh Add và Subtract được thực hiện trên 2 số có dấu 4-bit. Cờ báo add_sub_overflow sẽ được bật lên 1 khi mạch phát hiện có overflow xảy ra.



| M | S ₁ | S ₀ | ALU Operations |
|---|----------------|----------------|----------------|
| 0 | 0 | 0 | Complement A |
| 0 | 0 | 1 | AND |
| 0 | 1 | 0 | EX-OR |
| 0 | 1 | 1 | OR |
| 1 | 0 | 0 | Decrement A |
| 1 | 0 | 1 | Add |
| 1 | 1 | 0 | Subtract |
| 1 | 1 | 1 | Increment A |

3.3 Sinh viên chuẩn bị

 Download phần mềm LogiSim:

<http://sourceforge.net/projects/circuit/>

- ✚ Xem trước các clip để chuẩn bị các thao tác

<https://www.youtube.com/watch?v=dYZ-Hwbcnq4>

- ✚ Xem tài liệu tham khảo và hướng dẫn sử dụng LogiSim

<http://www.cburch.com/logisim/docs/2.7/en/html/guide/index.html>

- ✚ Vẽ mạch thiết kế ALU 4-bit trong Hình 3-1 và mô phỏng trên phần mềm LogiSim

- ✚ Vẽ mạch thiết kế ALU 4-bit trong Hình 3-1 trên phần mềm Quartus II

3.4 Hướng dẫn thực hành

1. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab3-MSSV.
2. Thực thi thiết kế ALU 4-bit trong Hình 3-1 và mô phỏng trên phần mềm LogiSim.
3. Thực thi thiết kế ALU 4-bit trong Hình 3-1 và mô phỏng trên phần mềm Quartus II.

3.5 Bài tập ôn tập

1. Dựa thiết kế ALU 4-bit trong Hình 3-1 đề xuất một mô hình cho ALU 8-bit

Bài 4. THIẾT KẾ DATAPATH ĐƠN GIẢN

4.1 Mục tiêu

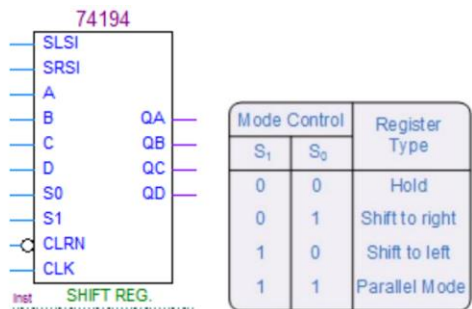
Sinh viên sẽ thực hiện thiết kế một DATAPATH đơn giản để thực hiện phép toán cộng, trừ, tăng giá trị 1 đơn vị và or (với các toán hạng là input được nhập từ ngõ vào) trên Quartus II.

4.2 Nội dung thực hành

Sinh viên dựa vào yêu cầu của bài thực hành và các lý thuyết đã học để xác định các thành phần của DATAPATH. Sau đó, sử dụng lại ALU 4-bit đã thiết kế trong bài Lab 3 và bộ Shifter (sử dụng chip 74194 trong Quartus II) như Hình 4-1 và các phần phần cần thiết khác để thiết kế DATAPATH theo yêu cầu.

| M | S ₁ | S ₀ | ALU Operations |
|---|----------------|----------------|----------------|
| 0 | 0 | 0 | Complement A |
| 0 | 0 | 1 | AND |
| 0 | 1 | 0 | EX-OR |
| 0 | 1 | 1 | OR |
| 1 | 0 | 0 | Decrement A |
| 1 | 0 | 1 | Add |
| 1 | 1 | 0 | Subtract |
| 1 | 1 | 1 | Increment A |

Hình 1. Bảng hoạt động của ALU



Hình 2. Shifter chip và bảng hoạt động

4.3 Sinh viên chuẩn bị

✚ Viết giải thuật với mã giả để thực hiện yêu cầu bài Lab

✚ Xác định các thành phần cần thiết của DATAPATH



4.4 Hướng dẫn thực hành

2. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab4-MSSV.
3. Kết nối các thành phần đơn lẻ để tạo thành DATAPATH hoàn chỉnh trên phần mềm Quartus II.

4.5 Bài tập ôn tập

1. Nếu thêm lệnh so sánh 2 số thì DATAPATH cần được thiết kế thêm những gì? Thiết kế lại DATAPATH thực hiện thêm lệnh so sánh 2 số.

Bài 5. THIẾT KẾ SIMPLE CONTROL UNIT






5.1 Mục tiêu

Sinh viên sẽ thực hiện thiết kế một Control Unit đơn giản để thực hiện phép toán phép toán cộng, trừ, tăng giá trị 1 đơn vị và or (với các toán hạng là input được nhập từ ngõ vào) trên Quartus II.

5.2 Nội dung thực hành

Dựa vào simple DATAPATH đã được thiết kế trong Lab 4 và các lý thuyết liên quan, sinh viên sẽ tiến hành thiết kế Control Unit phù hợp với yêu cầu của bài Lab bằng mạch Schematic.

5.3 Sinh viên chuẩn bị

-  Viết giải thuật với mã giả để thực hiện yêu cầu bài Lab
-  Lập bảng Control Word cho DATAPATH
-  Xác định lưu đồ chuyển trạng thái của Control Unit
-  Xây dựng bảng chuyển trạng thái sử dụng FF-D
-  Lập bảng các giá trị ngõ ra của Control Unit để thực hiện giải thuật trên

5.4 Hướng dẫn thực hành

1. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab5-MSSV.
2. Thiết kế Control Unit hoàn chỉnh trên phần mềm Quartus II

5.5 Bài tập ôn tập

1. Nếu thêm lệnh so sánh 2 số thì Control Unit cần được thiết kế thêm những gì? Thiết kế lại Control Unit thực hiện thêm lệnh so sánh 2 số.

Bài 6. THIẾT KẾ SIMPLE PROCESSOR

6.1 Mục tiêu

Sinh viên sẽ thực hiện thiết kế một Processor đơn giản để thực hiện phép toán phép toán cộng, trừ, tăng giá trị 1 đơn vị và or (với các toán hạng là input được nhập từ ngõ vào) bằng Schematic trên Quartus II.

6.2 Nội dung thực hành

Thiết kế một Processor đơn giản để thực hiện phép toán phép toán cộng, trừ, tăng giá trị 1 đơn vị, so sánh 2 số và or bằng cách sử dụng DATAPATH trong Lab 4, Control Unit trong Lab5 và kết hợp với một số cổng logic khác.

6.3 Sinh viên chuẩn bị

Mạch kết nối hoàn chỉnh của simple Processor từ simple DATAPATH (lab 4) và simple Control Unit (lab 5).

6.4 Hướng dẫn thực hành

1. Tạo một project mới trên phần mềm Quartus II, đặt tên: E/CE118-Lab/Lab6-MSSV.
2. Hiện thực và thử nghiệm Simple Processor trên Kit DE2.

TÀI LIỆU THAM KHẢO

1. Tài liệu hướng dẫn từ Altera:

ftp://ftp.altera.com/up/pub/Altera_Material/11.0/Tutorials/Schematic/Quartus_II_Introduction.pdf

2. Một số clip hướng dẫn:

<http://www.youtube.com/watch?v=gBknFw511s0&list=PL6uEPJDfhWzn0j06UwSII3S2H68GCuRnm>

<http://www.youtube.com/watch?v=fQKTbyQHAjE>

3. Một số trang web về mạch số:

<http://users.ece.gatech.edu/~hamblen/DE2/>

<http://www.youtube.com/watch?v=BubMdhBYXhw>

<http://voer.edu.vn/m/mach-so-sanh/fa3044bd>

4. Chức năng mạch Mux:

<http://en.wikipedia.org/wiki/Multiplexer>