

7/10/2013

Monday, October 07, 2013 1:34 PM

3.2.2. Mô hình hành vi

- a) Khối lệnh always
- b) Câu lệnh gán tuần tự (blocking)
- c) Câu lệnh điều kiện if/else
- d) Câu lệnh lựa chọn case

- Cú pháp

case/casex/casez (biểu_thức)

giá_trị_1: khối_lệnh_1

giá_trị_2: khối_lệnh_2

...

default: khối_lệnh_n;

endcase

- Hoạt động

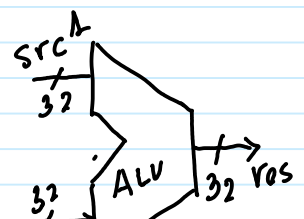
- Giá trị của biểu_thức sẽ được so sánh với các giá trị lựa chọn giá_trị_1, giá_trị_2, ... tùy từng loại case:
 - case: so sánh sử dụng phép toán === (phân biệt giữa các giá trị 0, 1, x, z);
 - casex: các giá trị x, z, ? trong giá trị lựa chọn sẽ được coi là bằng với bit giá trị 0 và 1. Ví dụ: 0x? có thể coi là bằng 000, 001, 010, 011
 - casez: các giá trị z, ? trong giá trị lựa chọn sẽ được coi là bằng với bit giá trị 0 và 1.
 - Khối lệnh có giá trị lựa chọn bằng giá trị biểu thức sẽ được thực hiện.
 - Nếu không có giá trị lựa chọn bằng giá trị biểu thức, khối lệnh n tương ứng với nhánh default sẽ được thực hiện
 - Nếu có nhiều hơn 1 khối lệnh có giá trị lựa chọn bằng giá trị biểu thức thì khối lệnh đứng trước sẽ được thực hiện (có mức ưu tiên cao hơn)

- Tổng hợp:

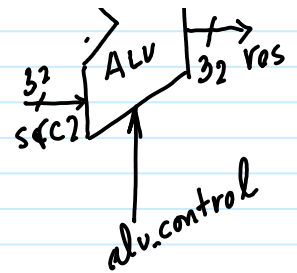
- Các khối lệnh của tất cả các nhánh lựa chọn đều được tổng hợp và được ghép nối qua khối mux
- casex, casez sẽ được tổng hợp thành mạch có mức ưu tiên khi có 2 nhánh lựa chọn cùng đúng
- nếu không có default và có biến không được gán giá trị trong mọi nhánh giá trị thì sẽ tổng hợp thành mạch tuần tự sử dụng chốt

- Ví dụ: Bộ tính toán số học và logic cho MIPS32

ALUOp	Func	ALUControl
00	X	010 (add)
X1	X	110 (subtract)
1X	100000 (add)	010 (add)
1X	100010 (sub)	110 (subtract)



1X	100000 (add)	010 (add)
1X	100010 (sub)	110 (subtract)
1X	100100 (and)	000 (and)
1X	100101 (or)	001 (or)
1X	101010 (slt)	111 (set less than)



- Ví dụ (Bài tập về nhà)
 - Bộ decoder38, encoder83, encoder83_priority: Xây dựng bằng case
- Chú ý:
 - Tham khảo thêm: Clifford Cummings: "full_case parallel_case, the Evil Twins of Verilog Synthesis"
 - Lời khuyên: chỉ sử dụng case (hạn chế dùng casex và casez), các giá trị lựa chọn trong các nhánh loại trừ nhau
- e) Lệnh lặp tĩnh (Số lần lặp cố định không phụ thuộc biến số)
 - Cú pháp
 - for
 - for (gán_khởi_tạo;biểu_thức_điều_kiện;gán_chỉ_số)
 - lệnh/khối_lệnh;
 - trong đó: gán_khởi_tạo là phép gán giá trị đầu cho chỉ số; biểu_thức_điều_kiện sử dụng để kiểm tra điểm kết thúc của lệnh gán; gán_chỉ_số là phép gán thay đổi giá trị chỉ số
 - repeat
 - repeat (số_lần_lặp)
 - lệnh/khối_lệnh;
 - Hoạt động
 - Với lệnh for:
 - Lệnh gán_khởi_tạo sẽ được thực hiện 1 lần ở bắt đầu vòng lặp
 - Biểu_thức_điều_kiện sẽ được kiểm tra mỗi lần lặp
 - Nếu biểu_thức_điều_kiện đúng thì các trong khối_lệnh sẽ được thực hiện
 - Lệnh gán_chỉ_số sẽ được thực hiện sau mỗi lần lặp
 - Với lệnh repeat: các lệnh trong khối_lệnh sẽ được thực hiện số_lần_lặp lần
 - Tổng hợp
 - Các lệnh lặp sẽ được trải (unroll) số lần bằng số lần lặp và sau đó được tổng hợp như một khối_lệnh thủ tục thông thường
 - Ví dụ: mạch majority