Lógica Digital (1001351)

Circuitos Sequenciais: Máquinas de Estados Finitos

Prof. Edilson Kato
kato@ufscar.br

Prof. Maurício Figueiredo
mauricio@ufscar.br

Prof. Ricardo Menotti
menotti@ufscar.br

Prof. Roberto Inoue
rsinoue@ufscar.br

Departamento de Computação
Universidade Federal de São Carlos

Atualizado em: 31 de maio de 2019

figure6.29.v

```
1  module simple (Clock, Resetn, w, z);
2    input Clock, Resetn, w;
3    output z;
4    reg [2:1] y, Y;
5    parameter [2:1] A = 2'b00, B = 2'b01, C = 2'b10;
6    // Define the next state combinational circuit
7    always @(w, y)
8      case (y)
9        A: if (w)   Y = B;
10          else     Y = A;
11       B: if (w)   Y = C;
12          else     Y = A;
13       C: if (w)   Y = C;
14          else     Y = A;
15       default:    Y = 2'bxx;
16     endcase
17   // Define the sequential block
18   always @(negedge Resetn, posedge Clock)
19     if (Resetn == 0)   y <= A;
20     else  y <= Y;
21   // Define output
22   assign z = (y == C);
23 endmodule
```

figure6.33.v

```
1  module simple (Clock, Resetn, w, z);
2    input Clock, Resetn, w;
3    output reg z;
4    reg [2:1] y, Y;
5    parameter [2:1] A = 2'b00, B = 2'b01, C = 2'b10;
6    // Define the next state combinational circuit
7    always @(w, y)
8    begin
9      case (y)
10       A: if (w)   Y = B;
11            else   Y = A;
12       B: if (w)   Y = C;
13            else   Y = A;
14       C: if (w)   Y = C;
15            else   Y = A;
16       default:    Y = 2'bxx;
17     endcase
18     z = (y == C);  //Define output
19    end
20   // Define the sequential block
21   always @(negedge Resetn, posedge Clock)
22     if (Resetn == 0)  y <= A;
23     else  y <= Y;
24 endmodule
```

figure6.34.v

```
1  module simple (Clock, Resetn, w, z);
2    input Clock, Resetn, w;
3    output z;
4    reg [2:1] y;
5    parameter [2:1] A = 2'b00, B = 2'b01, C = 2'b10;
6    // Define the sequential block
7    always @(negedge Resetn, posedge Clock)
8      if (Resetn == 0)  y <= A;
9      else
10       case (y)
11         A: if (w)  y <= B;
12              else  y <= A;
13         B: if (w)  y <= C;
14              else  y <= A;
15         C: if (w)  y <= C;
16              else  y <= A;
17         default:  y <= 2'bxx;
18       endcase
19   // Define output
20   assign z = (y == C);
21 endmodule
```

figure6.35.v

```
1  module control (Clock, Resetn, w, R1in, R1out, R2in, R2out, R3in, R3out,Done);
2    input Clock, Resetn, w;
3    output R1in, R1out, R2in, R2out, R3in, R3out, Done;
4    reg [2:1] y, Y;
5    parameter [2:1] A = 2'b00, B = 2'b01, C = 2'b10, D = 2'b11;
6    // Define the next state combinational circuit
7    always @(w, y)
8      case (y)
9        A: if (w)  Y = B;
10             else  Y = A;
11       B:  Y = C;
12       C:  Y = D;
13       D:  Y = A;
14     endcase
15   // Define the sequential block
16   always @(negedge Resetn, posedge Clock)
17     if (Resetn == 0)  y <= A;
18     else  y <= Y;
19   // Define outputs
20   assign R2out = (y == B);
21   assign R3in = (y == B);
22   assign R1out = (y == C);
23   assign R2in = (y == C);
24   assign R3out = (y == D);
25   assign R1in = (y == D);
26   assign Done = (y == D);
27 endmodule
```

figure6.36.v

```
1  module mealy (Clock, Resetn, w, z);
2    input Clock, Resetn, w;
3    output reg z;
4    reg y, Y;
5    parameter A = 1'b0, B = 1'b1;
6    // Define the next state and output combinational circuits
7    always @(w, y)
8      case (y)
9        A:  if (w) begin
10             z = 0;
11             Y = B;
12             end
13         else begin
14           z = 0;
15           Y = A;
16            end
17       B:  if (w) begin
18             z = 1;
19             Y = B;
20             end
21         else begin
22           z = 0;
23           Y = A;
24         end
25     endcase
26   // Define the sequential block
27   always @(negedge Resetn, posedge Clock)
28     if (Resetn = = 0) y <= A;
29     else   y <= Y;
30 endmodule
```

Bibliografia

- Brown, S. & Vranesic, Z. - Fundamentals of Digital Logic with Verilog Design, 3rd Ed., Mc Graw Hill, 2009

Lógica Digital (1001351)

Circuitos Sequenciais: Máquinas de Estados Finitos

Prof. Edilson Kato
kato@ufscar.br

Prof. Maurício Figueiredo
mauricio@ufscar.br

Prof. Ricardo Menotti
menotti@ufscar.br

Prof. Roberto Inoue
rsinoue@ufscar.br

Departamento de Computação
Universidade Federal de São Carlos

Atualizado em: 31 de maio de 2019