Chirag solanki

15035

**Answer 1**

module moore1011

(input clk, rst, inp,

output reg outp);

reg [2:0] state;

parameter S0=0, S1=1, S2=2, S3=3,S4=4;

always @(posedge clk, posedge rst)

if(rst==1)

state<=S0;

else

case(state)

S0: if(inp)

state<=S1;

else

state<=S0;

S1: if(inp)

state<=S1;

else

state<=S2;

S2: if(inp)

state<=S3;

else

state<=S0;

S3: if(inp)

state<=S4;

else

state<=S1;

S4: if(inp)

state<=S0;

else

state<=S2;

endcase

///output logic

always @(state)

case(state)

S0:

outp<=0;

S1:

outp<=0;

S2:

outp<=0;

S3:

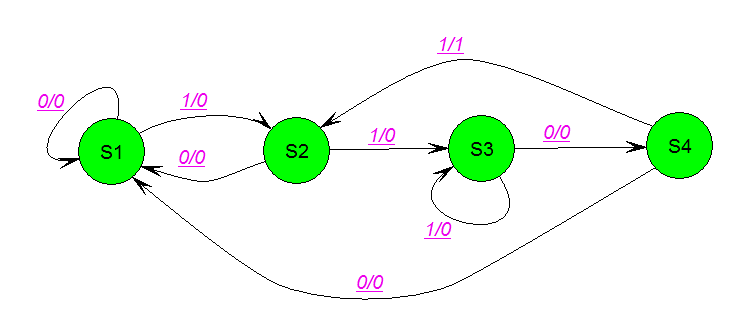
outp<=0;

S4:

outp<=1;

endcase

endmodule



Answer 3

module bitDiv7(

clk,

rst,

in\_i,

flag\_o

);

Input clk;

Input rst;

Input in\_i;

Output flag\_o;

Reg flag\_o;

Parameter pIdle= 4’b0000;

Parameter pRem0= 4’b0001;

Parameter pRem1= 4’b0010;

Parameter pRem2= 4’b0011;

Parameter pRem3= 4’b0100;

Parameter pRem4= 4’b1010;

Parameter pRem5= 4’b0011;

Parameter pRem6= 4’b1110;

reg[2:0] curSt\_r;

reg[2:0] nxtSt\_r;

always@(posedge clk) begin

if(rst)curSt\_r <= #1 pIdle;

elsecurSt\_r <= #1 nxtSt\_r;

end

always @ (\*)begin

flag\_o = 1’b0;

case (curSt\_r)

pIdle: begin

if(rst) nxtSt\_r = pIdle;

else begin

if(in\_i)nxtSt\_r = pRem1;

elsebegin

flag\_o = 1’b1;

nxtSt\_r = pRem0;

end

end

pRem0: begin

if(in\_i)nxtSt\_r = pRem1;

elsebegin

flag\_o = 1’b1;

nxtSt\_r = pRem0;

end

end

pRem1: begin

if(in\_i)nxtSt\_r = pRem3;

elsenxtSt\_r = pRem2;

end

pRem2: begin

if(in\_i)begin

flag\_o = 1’b1;

nxtSt\_r = pRem0;

end

elsenxtSt\_r = pRem4;

end

pRem3: begin

if(in\_i)nxtSt\_r = pRem2;

elsenxtSt\_r = pRem1;

end

pRem4: begin

if(in\_i)nxtSt\_r = pRem4;

elsenxtSt\_r = pRem3;

end

pRem5: begin

if(in\_i)nxtSt\_r = pRem5;

elsenxtSt\_r = pRem4;

end

pRem6: begin

if(in\_i)nxtSt\_r = pRem6;

elsenxtSt\_r = pRem5;

end

default:nxtSt\_r = pIdle;

endcase

end

endmodule