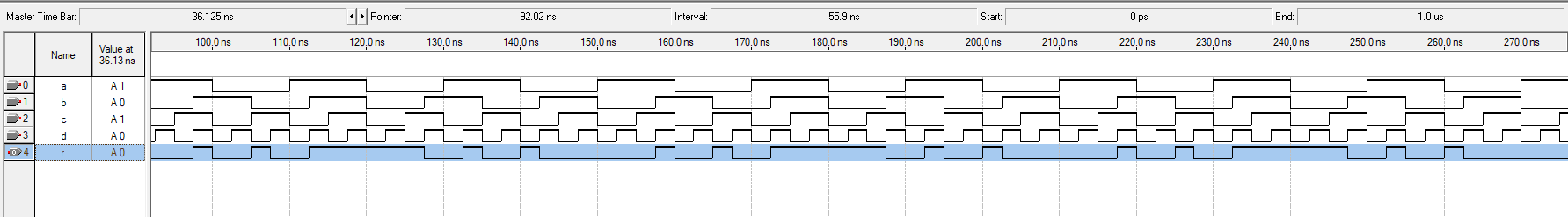
**Lab Assignment 01**

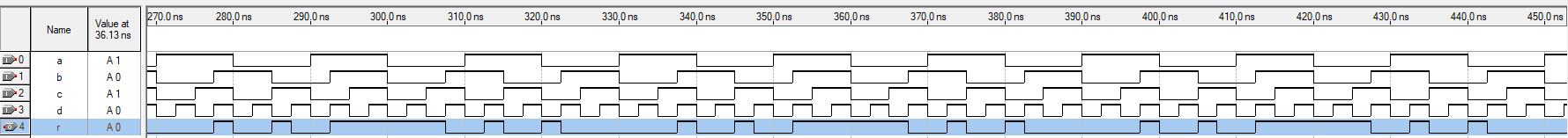
Shihab Muhtasim (21301610)

CSE460 (2) sp24

**Task 1**

a)





b)

module circuit(a,b,c,d,r);

input a,b,c,d;

output r;

wire p,q,s,t;

assign p= a^~b;

assign q= a & (c | d);

assign s= ~a & ~(c & d);

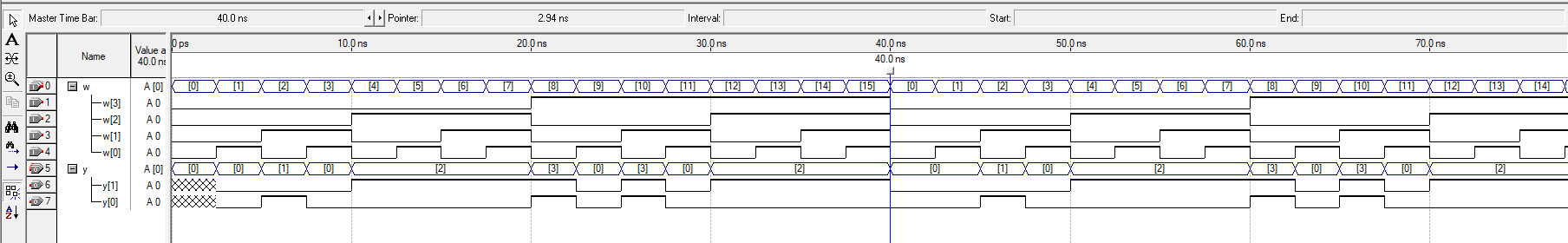
assign t= q|s;

assign r= p & t;

endmodule

**Task 2**

a)



b)

module encoder (w,y);

input [3:0]w;

output reg [1:0]y;

//2>0>3>1

always @(\*)

casex(w)

4'bx1xx : y=2;

4'bx0x1 : y=0;

4'b10x0 : y=3;

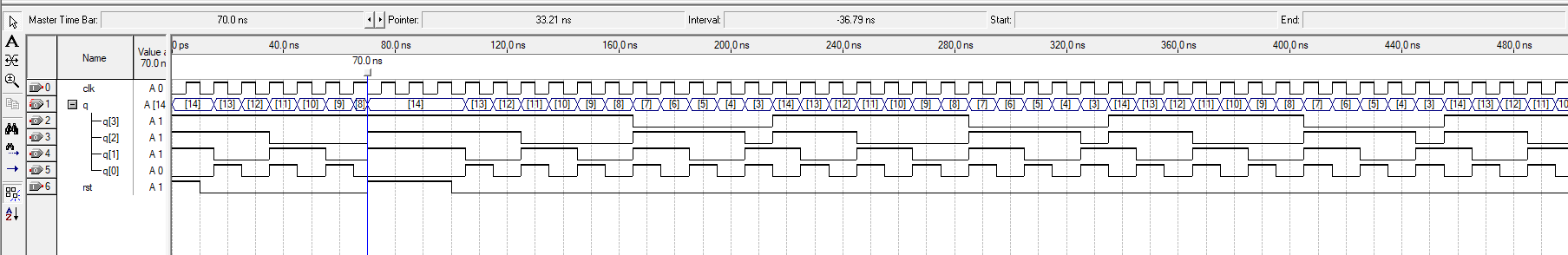
4'b0010 : y=1;

endcase

endmodule

**Task3**

a)



b)

module counter(rst,q,clk);

input rst,clk;

output reg [3:0]q;

always @(posedge clk, posedge rst)

begin

if (rst)

q<=14;

else if (q==3)

q<=14;

else

q<= q-1;

end

endmodule

**Task 4**

**Code:**

module circle(xc,yc,r,z,m,n);

input [2:0]xc,yc,r,m,n;

output reg [1:0]z;

wire [5:0] r\_sq;

wire [5:0] d;

assign d= (m-xc)\*(m-xc)+(n-yc)\*(n-yc);

assign r\_sq=r\*r;

always @(\*)

begin

if (d==r\_sq)

z=0;

else if (d>r\_sq)

z=1;

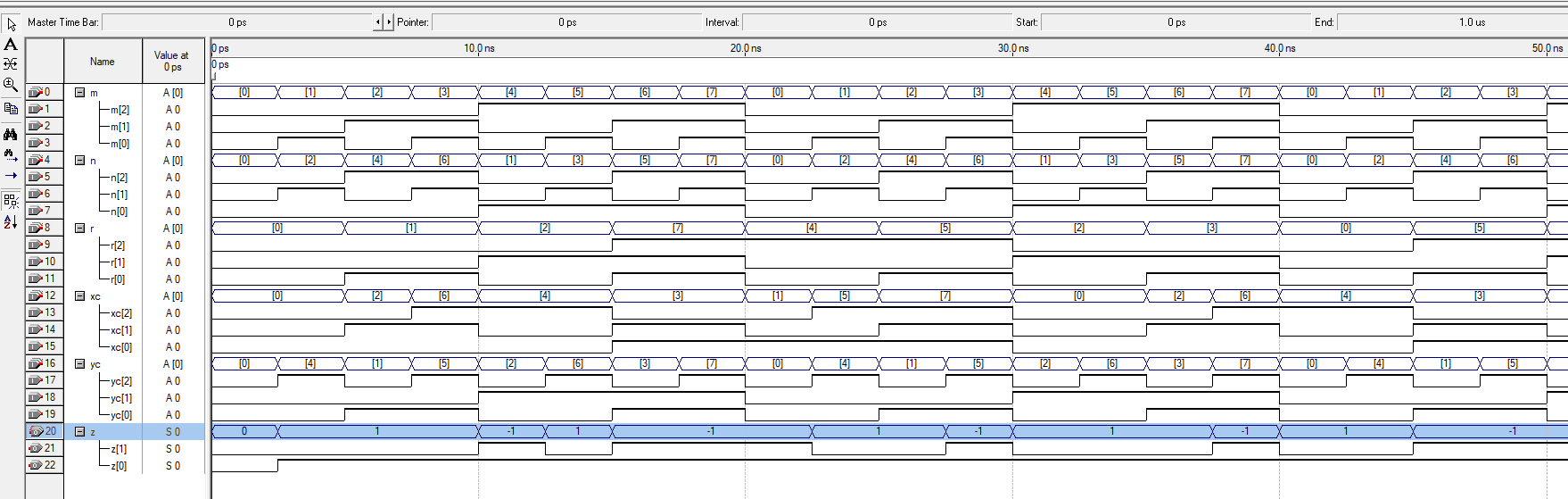
else

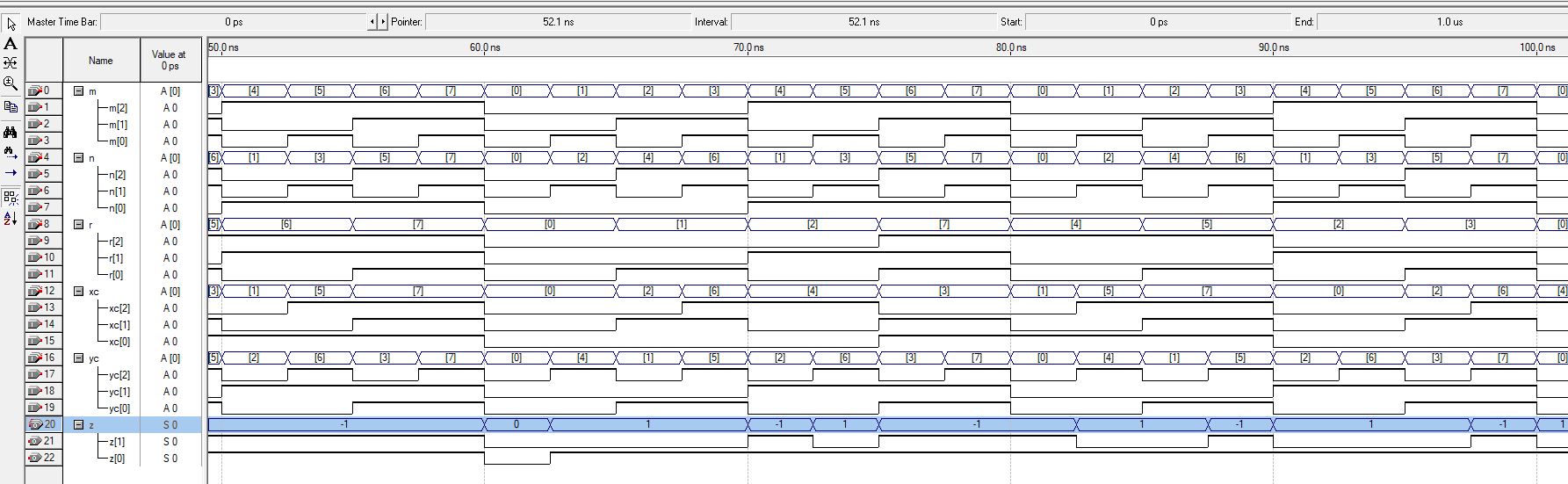
z=-1;

end

endmodule

**Diagram:**





**Validation:**

The above program takes input x,y, and r as the parameters for a circle and uses the circle equation (x-xc)\*\*2+ (y-yx)\*\*2= r\*\*2 to find the given point (m,n) fall in the circle or not. If the result of LHS using x=m and y=n is less than r square then the point is inside the circle. So, it can be seen in the figure where the LHS result was less than the value of r square then z was equal to -1 indicating m, and n are inside the circle. Similarly, LHS= r sq will mean that m,n are on the circle, and if LHS>r sq then m,n are outside the circle denoting with z=1 in the timing diagram. The timing diagram inputs are set by random clock rates and provide output z according to the timing diagram.