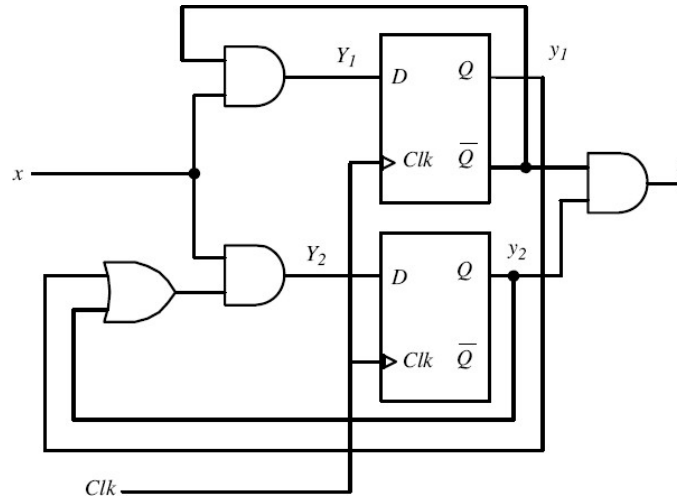


Nome e Matricula: \_\_\_\_\_

1. Para o circuito abaixo que implementa uma máquina de estados.

(a) Suponha que os 2 flipflops tenham o valor inicial 0, desenhe a tabela com descrição da máquina de estados implementada pelo circuito



e2	e1	x	p2=x(e1 e2)	p1=x!e1	z	data
0	0	0	0	0	0	0
		1	0	1	0	2
0	1	0	0	0	0	0
		1	1	0	0	4
1	0	0	0	0	1	1
		1	1	1	1	7
1	1	0	0	0	0	0
		1	1	0	0	4

(b) Desenhe o diagrama de estados da máquina.

(c) Descreva em Verilog com os dois formatos (case, equações) a máquina abaixo.

```
module ff ( input data, input c, input r, output q);
```

```
reg q;
```

```
always @(posedge c or negedge r)
```

```
begin
```

```
if(r==1'b0) q <= 1'b0;
```

```
else q <= data;
```

```
end
```

```
endmodule //End
```

```
module statem(input clk, input reset, input x, output z);
```

```
reg [1:0] state;
```

```
parameter zero=2'd0, one=2'd1, two=2'd2, three=2'd3 ;
```

```
assign z = (state == two)? 1:0;
```

```
always @(posedge clk or negedge reset)
```

```

begin
  if (reset==0)
    state = zero;
  else
    case (state)
      zero: if ( x == 1 ) state = one;
      one:  if ( x == 1 ) state = two;
            else state = zero;
      two:  if ( x == 1 ) state = three;
            else state = zero;
      three: if ( x == 1 ) state = two;
             else state = zero;

    endcase
  end
endmodule

module statePorta(input clk, input res, input x, output z);
wire [1:0] e; wire [1:0] p;
assign z = ~e[0] & e[1];
assign p[0] = ~e[0] & x;
assign p[1] = x & (e[0] | e[1]);
ff e0(p[0],clk,res,e[0]);
ff e1(p[1],clk,res,e[1]);
endmodule

```

2. A com 2 bits. Se A=0 sequencia 0,1,0,3 → 0,1,0,3, Se A=1 sequencia 1,3 → 1,3, Se A=2 sequencia 0,1,0 → 0,1,0 → ..... Se A=3 sequencia 1,0 → 1,0 ....  
Fazer a tabela, diagrama de estados e preencher a memória. Não é preciso fazer código em Verilog.

3. Suponha a tabela abaixo com o conteúdo da memória de uma máquina de estados, onde os 2 primeiros bits são o próximo estado e os três últimos são as saídas. O endereço da memória é a concatenação do estado atual E1, E0 e a entrada A. Por exemplo, na linha 2 (010 em binário), o estado é 01 e a entrada A tem o valor 0. Desenhe o diagrama de estados da máquina. O estado inicial é 00.

p1	p0	s2	s1	s0	Numero da linha
0	0	1	0	1	0
0	1	1	0	1	1
0	0	1	1	1	2 (010) onde E1E0 = 01, A = 0
1	1	1	1	1	3
1	0	1	0	1	4
0	1	1	0	1	5
0	0	0	0	1	6
0	1	0	0	1	7

```

module stateMem(input clk,input res, input x, output z);
reg [2:0] StateMachine [0:7];
initial

```

```
begin
StateMachine[0] = 4'd0; StateMachine[1] = 4'd2; StateMachine[2] = 4'd0; StateMachine[3] = 4'd4;
StateMachine[4] = 4'd1; StateMachine[5] = 4'd7; StateMachine[6] = 4'd0; StateMachine[7] = 4'd4;
end
wire [2:0] address; wire [2:0] dout;
assign address[0] = x;
assign dout = StateMachine[address];
assign z = dout[0];
ff st0(dout[1],clk,res,address[1]);
ff st1(dout[2],clk,res,address[2]);
endmodule
```

-----Exemplos de Verilog para Questão 1 -----