

INF 250 - Lista de Exercícios

Ponto flutuante e FSM

Questões Ponto Flutuante

Q1. Converta os números a seguir para a representação binária no formato Float7 (3 bits de expoente e 4 bits de mantissa), considerando:

- a) 0.75
 - b) 0.62
 - c) 0.95
 - d) 1.12
 - e) 1.45
 - f) 1.99
 - g) 2.25
 - h) 3.10
 - i) 6.40
 - j) 12.50
-

Q2. Dados os seguintes números no formato Float7 com 3 bits de expoente e 4 mantissa, calcule a soma ($A+B$) e a multiplicação ($A*B$):

- a) $A = 0.75$, $B = 0.62$
- b) $A = 1.12$, $B = 1.85$
- c) $A = 2.50$, $B = 1.25$
- d) $A = 0.90$, $B = 1.30$
- e) $A = 3.10$, $B = 0.95$
- f) $A = 1.99$, $B = 1.05$
- g) $A = 0.45$, $B = 0.80$
- h) $A = 2.25$, $B = 2.50$
- i) $A = 1.50$, $B = 0.72$
- j) $A = 2.99$, $B = 1.40$

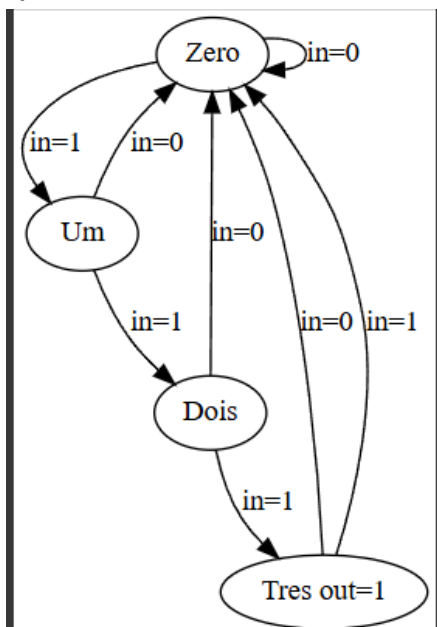
- k) $A = 3.10$, $B = 6.40$
- l) $A = 4.25$, $B = 2.75$
- m) $A = 5.50$, $B = 1.80$

- n) $A = 7.20$, $B = 3.60$
- o) $A = 6.75$, $B = 2.30$
- p) $A = 8.10$, $B = 4.05$
- q) $A = 9.50$, $B = 0.95$
- r) $A = 12.40$, $B = 3.10$
- s) $A = 10.25$, $B = 6.50$
- t) $A = 15.75$, $B = 2.20$

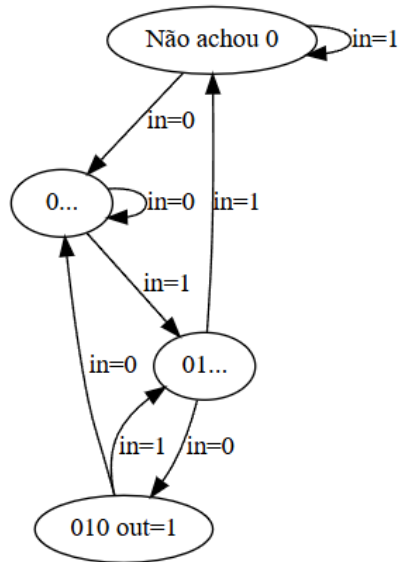
Questões FSM

Q1. Fazer os 3 tipos de código verilog e tabela do diagrama ([link](#))

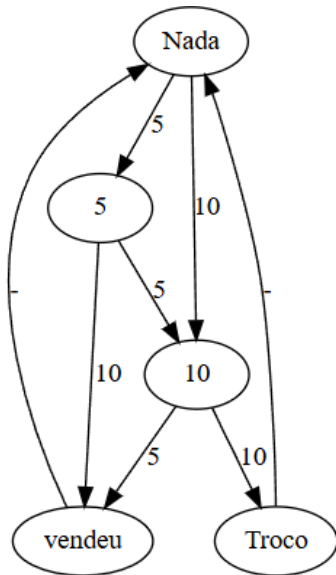
a)



b)



c)



Q2 Fazer o diagrama de estados, fazer a tabela, fazer a descrição com os 3 tipos de verilog ([link](#))([solução](#)).

para uma máquina com 2 entradas A e B e uma saída L, onde L irá piscar 0,1...se A mudar de 0 para 1 e depois irá parar de piscar quando mudar novamente de 0 para 1. Com relação a B, se a saída L estiver Ligada, B=1 mudara o padrão de piscar para 0, 1, 1, 0, 1,1,...ou seja,

fica um clock em 0 e dois clocks em 1. Se B=0, piscará 0,1,0,1....alternando.

Q3. Montar a Tabela para o circuito abaixo e depois montar o diagrama ([link](#))([solução](#)).

para uma máquina com 1 entrada A e três saídas B,C,D, se A mudar de 0 para 1, B fica ligado, C irá piscar 0,1,0,1 e D desligado. Se A mudar de 1 para 0, C e D irão alternar, se C está ligado, D será desligado e B ficará desligado.

Q4. Montar a Tabela para o circuito abaixo e depois montar o diagrama ([link](#))([solução](#)).

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

Q5. Montar a descrição comportamental em Verilog (case, if, ...) do item anterior ([link](#)) ([solução](#)).

Q6 . Montar a Tabela para o circuito abaixo e depois montar o diagrama ([link](#))([solução](#)).

```
module statePorta(input clk, input res, output A, output B);
wire [2:0] e;
wire [2:0] p;
wire [2:0] state;
assign state = e;
wire x,y;
assign p[2] = ~e[2] & ~e[0];
assign x = ~e[2] & e[1];
assign p[1] = ~e[0] & x;
assign y = e[2] & e[1];
assign A = ~e[0] & ~y;
assign p[0] = 1'b0;
```

```
assign B = p[1];
ff  e0(p[0],clk,res,e[0]);
ff  e1(p[1],clk,res,e[1]);
ff  e2(p[2],clk,res,e[2]);
endmodule
```

Q7. Dada as entradas A e B, as saídas C e D, fazer o diagrama de estados, a tabela, o circuito Verilog com equações, com memória e comportamental ([solução](#))l.

A máquina tem o seguinte comportamento:

- A = 0, alternar C e D, piscando, C = 1, D = 0 e vice-versa.
- A = 1 e B = 1, C = 1, D = 0, fica 2 ciclos, depois alterna com C = 0, D = 1, dois ciclos e repete.
- A = 1 e B = 0, terá a saída fixa em C = 1 e D = 0.

Q8. Dada a entrada A, as saídas B, C e D, fazer o diagrama de estados, a tabela, o circuito Verilog com equações, com memória e comportamental ([solução](#)).

A máquina tem o seguinte comportamento:

- A = 0, as saídas B = 1, C e D apagadas, depois C = 1 com B e D apagados e depois D = 1 e B e C apagados. Depois repete.
- A = 1, as saídas B = 1, C e D apagadas, 2 ciclos, depois C = 1, B e D apagados 2 ciclos e repete.

Q9. Suponha uma máquina de vendas de refrigerante. Cada refrigerante custa 15 unidades. Existem duas moedas: 5 e 10 unidades. Um sensor M5 indica a detecção de uma moeda de 5 e outro M10 da moeda de 10. A seguir iremos ilustrar passo a passo a metodologia para o projeto de uma máquina de estado. Os passos serão ([link](#)):

1. montar o diagrama de estados
2. montar a tabela do diagrama (com estados simbólicos)

3. codificar os estados, gerando uma tabela verdade
4. projetar as funções lógicas para as funções de saída e transição de estado