

PSI- 3451: Folha de Respostas- Aula 2

Nome do Aluno:

No. USP:

Observação: associe o(s) nome(s) dos arquivos subidos à enumeração as seções (itens) da apostila, replicados neste questionário.

Observação: todas as curvas devem estar com boa resolução e os nomes de sinais visíveis e identificáveis.

A Tabela da Verdade abaixo expressa a computação de um somador completo e deve ser usada como referência para esta atividade.

Caso	a_in	b_in	c_in	z_out	c_out
A	0	0	0	0	0
B	0	0	1	1	0
C	0	1	1	0	1

Você deverá seguir os casos abaixo para as suas simulações:

Transições Dist. Hamming 1	Uma Transição Dist. Hamming 2
A → B → C	B → A → C

Item 1) Captura e simulação do somador completo full_adder_1 (com atrasos inerciais)

a) Fazer upload de curva de simulação com as transições de distância de Hamming 1 (jpg ou bmp), fazendo as seguintes anotações na própria curva:

a.1) identificação dos valores dos tempos de atraso, na atualização dos sinais z_out, c_out, aux_and_1, aux_and_2, aux_and_3 e aux_xor (observar como a transição de um sinal afeta um outro).

Na transição de c_in, o tempo para o efeito aparecer em z_out é ns

a.2) identificação das dependências aos sinais das expressões, na atualização dos sinais z_out, c_out, aux_and_1, aux_and_2, aux_and_3 e aux_xor (estudar nas expressões como a transição de um sinal afeta um outro).

b) Fazer upload de curva de simulação com uma transição de distância de Hamming 2 (jpg ou bmp), fazendo as seguintes anotações:

b.1) identificação dos valores dos tempos de atraso, na atualização dos sinais `z_out`, `c_out`, `aux_and_1`, `aux_and_2`, `aux_and_3` e `aux_xor` (observar como a transição de um sinal afeta um outro).

b.2) na transição com distância de Hamming 2, identificação da mudança em `aux_xor` e da dependência do `z_out` aos valores de suas entradas (na expressão)

Item 2) Captura e simulação do somador completo `full_adder_1` com atraso *delta*

a) Fazer upload de curva de simulação com as transições de distância de Hamming 1 (jpg ou bmp). Anotar as dependências nas transições de atualização dos sinais `z_out`, `c_out`, `aux_and_1`, `aux_and_2`, `aux_and_3` e `aux_xor` (estudar nas expressões como a transição de um sinal afeta um outro).

b) Comente no quadro abaixo se há diferença no resultado final da simulação na comparação com o item 1.a. Comente as diferenças nos tempos de atraso.

Sim há diferença, na primeira simulação observamos um atraso expressivo (de 8ns para as saídas e menor para as portas q antecedem a saída). Já na simulação do item 2 o atraso não pode ser observado mesmo com zoom, isso pq ele é um atraso delta infinitesimal, ocorre tbem q não se pode notar a propagação do atraso, já q somar infinitesimais será infinitesimal.

Item 3) Captura e simulação do somador completo `full_adder_2` no modelo estrutural

a) Fazer upload de curva de simulação com as transições de distância de Hamming 1 (jpg ou bmp), fazendo as seguintes anotações:

a.1) identificação dos valores dos tempos de atraso, na atualização dos sinais `z_out`, `c_out`, `aux_and_1`, `aux_and_2`, `aux_and_3` e `aux_xor`

a.2) identificação das dependências aos sinais das expressões, na atualização dos sinais `z_out`, `c_out`, `aux_and_1`, `aux_and_2`, `aux_and_3` e `aux_xor`

b) Responder no quadro abaixo: b1) Os atrasos das portas no VHDL (generics) são os mesmos do modelos dataflow do item 1? b2) Os tempos de atraso vistos na simulação são iguais aos do item 1.a? Justifique.

A simulação não mostrou o comportamento apresentado na apostila de conceitos, isso se deve ao fato de as portas lógicas não terem sido instanciadas com os valores da apostila. No caso, os atrasos instanciados na descrição vhdI foram: xor2 com 13ns and2 com 10n e or3 com 15ns.

assim para z_out temos: $\text{and2} + \text{or3} = 25\text{ns}$

c_out temos: $2 \times \text{xor2} = 26\text{ns}$.

Na seção 1 tínhamos atrasos menores devido ao atraso descrito no código, porém o comportamento do circuito foi o mesmo.

Item 4) Captura, compilação e simulação do somador no modelo estrutural com alterações nas instâncias (sem generic)

Sobre a simulação realizada, responda no quadro abaixo: o comportamento temporal é similar ao do item 1.a? Explique/justifique a partir do que ocorreu do item 3) para o item 4).

Sim, o comportamento temporal é similar ao item 1, isso se deve ao valor determinado para os atrasos serem iguais.

Esse tempo de atraso se deve ao instanciamento das portas lógicas no código, como não há especificação para o generic, é usado o valor definido como padrão, no caso os valores descritos acima.

Item 5) Projeto, captura, compilação e simulação do somador de 4 bits no modelo estrutural

a) Inclua no quadro abaixo o código VHDL da entity e a arquitetura do somador ripple-carry

```
entity rc_adder_1 is
generic
(
WIDTH: natural := 4
);
port
(
a_i, b_i:in STD_LOGIC_VECTOR (WIDTH-1 downto 0);
z_out:out STD_LOGIC_VECTOR (WIDTH-1 downto 0);
c_i:in STD_LOGIC;
c_o:out STD_LOGIC
);

end rc_adder_1;

architecture structural of rc_adder_1 is

component full_adder_2
port
(
a_in, b_in, c_in:in STD_LOGIC;
z_out, c_out:out STD_LOGIC
);

end component;

component full_adder_1
port
(
a_in, b_in, c_in:in STD_LOGIC;
z_out, c_out:out STD_LOGIC
```

b) Fazer upload de curva de simulação com todos os sinais intermediários, evidenciando a ação do ripple carry na soma de a-i=1010 b_i= 0101 e c_i= 1.