

13.4 Modelagem de Circuitos Sequenciais em VHDL

São quatro os objetivos deste laboratório: (i) apreender o conceito de *processo* em VHDL; (ii) efetuar a modelagem em VHDL de sistemas sequenciais síncronos; (iii) escrever um modelo para uma máquina de estados finita e (iv) verificar, através de simulação, a corretude do modelo. O trabalho pode ser efetuado em duplas.

A Seção 13.4.6 contém a(s) tarefa(s) por resolver nesta aula e que deve(m) ser enviada(s) ao professor.

13.4.1 Revisão: como funciona o mecanismo de simulação

Reveja a Seção 9.2.5.

13.4.2 Comandos Sequenciais

Leia a Seção 9.4.2.

13.4.3 Processos

Leia a Seção 9.4.3.

13.4.4 Comandos Sequenciais

Leia a Seção 9.4.11.

13.4.5 Modelagem de máquinas de estado

Leia a Seção 9.4.12

Espaço em branco proposital.

13.4.6 Da tarefa:

Etapa 23 Projete uma Máquina de Moore, com uma entrada de dados D , uma saída C , uma entrada de relógio rel , e uma entrada de $reset$. Após a inicialização, sempre que o circuito detectar a sequência $\dots 01111110 \dots$ (0, seis 1, 0) a saída C produz um pulso com duração de um ciclo.

D: 001110111111001111110001010

C: 000000000000010000000100000

A sequência $\dots 011111101111110 \dots$ produz um único pulso na saída C .

Etapa 24 Copie para sua área de trabalho o arquivo com o código VHDL:

(a) `wget http://www.inf.ufpr.br/roberto/ci210/vhdl/l_processos.tgz`

(b) expanda-o com: `tar xzf l_processos.tgz`

o diretório `processos` será criado;

(c) mude para aquele diretório: `cd processos`

Ao contrário dos outros laboratórios, neste você deve escrever seu código VHDL para a máquina de estados diretamente no arquivo `tb_me.vhd`. Você deve consertar e completar o esqueleto que está no TB, acrescentando mais estados ao tipo `states` e cláusulas ao **case** do processo `U_st_transitions`. Uma vez completado seu modelo, e gerados os casos de teste que **comprovem** sua corretude – o vetor de testes deve, pelo menos, triplicar de tamanho – envie o código do `tb_me.vhd` para o seu professor, com assunto `ci1210-ME`, e os nomes dos dois autores, até as 23:59 do dia 27fev.

EOF

Histórico das Revisões:

02set2019: remoção de VHDL;

02set2019: revisão do texto;

06set2018: revisão do texto;

19set2017: revisão do texto, formatação de exemplos de código;

07set2017: revisão do texto, acréscimo de exemplos;

16set2016: revisão do texto por Marco Zanata;

12set2016: acréscimo de exemplos, melhorias no esqueleto do código VHDL;

01out2015: revisão do texto;

19set2015: primeira versão.