



Tecnologia dos Computadores (2018/2019)

-- Trabalho #7 --

VHDL para Circuitos Combinacionais

Duração: 1 semana

Objetivo

O VHDL é uma linguagem de programação que permite descrever e implementar *hardware*. Neste trabalho pretende-se introduzir a linguagem para descrição de circuitos puramente combinacionais (i.e. que não dependem de memória).

Um outro conceito fundamental em sistemas digitais é o conceito de hierarquia: um determinado circuito pode ser constituído por vários sub-circuitos. Este trabalho também abordará este tema, tanto ao nível de VHDL como do programa Quartus II.

Introdução

Nas aulas teóricas vimos que um circuito votador de três entradas (A, B, C) é descrito pela seguinte equação: $Y = AB + AC + BC$. Na figura seguinte podem ver-se duas representações deste circuito: como um bloco abstrato, com entradas e saídas representadas, e a nível de implementação interna, através do seu diagrama lógico.

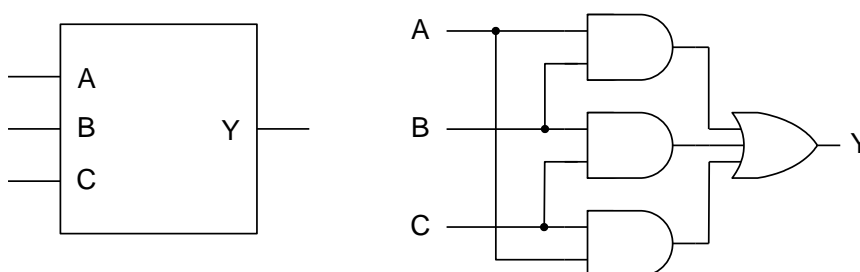


Figura 1 - Circuito votador

Neste trabalho iremos fazer diversas experiências com este circuito simples, tanto a nível de simulação, como a nível de implementação.

Nas diferentes fases do trabalho, crie pastas diferentes guardando cada parte numa pasta separada. Terá de mostrar ao professor cada uma das partes. Sempre que passar à fase seguinte, reaproveite o circuito que já criou da fase anterior (i.e. copie os ficheiros).

Trabalho

Fase 1 (trab07a) – Votador Simples

Crie o circuito acima mencionado. Chame ao circuito **VOTADOR** (i.e. “votador.bdf”). O resultado deverá ser semelhante ao apresentado na figura seguinte.

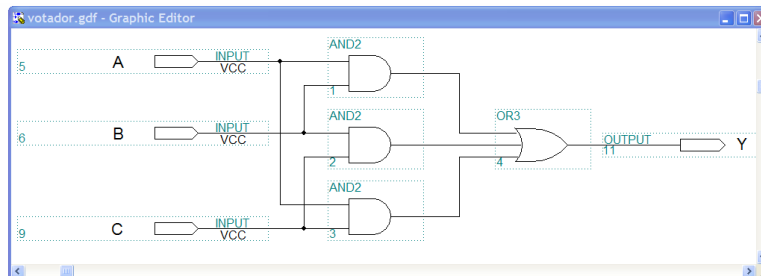


Figura 2 – Esquemático do circuito

De seguida, teste o circuito criado na placa DE2. Para isso, terá de associar as 3 entradas do circuito a 3 interruptores da placa DE2, e a saída a um LED. Consulte o manual da placa, como fez no trabalho laboratorial anterior.

Compile, programe (“Tools->Programmer”, escolhendo a opção “Program/Configure” e clicando “Start”) e experimente o dispositivo.

Nota: Sempre que desenhar um circuito, pode criar um símbolo para ele em termos de bloco de componente, i.e. onde apenas são representadas entradas e saídas. Esse símbolo pode ser utilizado para construir projetos mais sofisticados. Se aceder ao menu “File->Create/Update->Create Symbol Files for Current File”, o símbolo será criado (em “Edit->Insert Symbols” na pasta “Project” surge o símbolo votador).

Fase 2 (trab07b) – Conceito de Hierarquia

Após copiar o conteúdo da pasta anterior para uma nova pasta, crie um novo esquemático chamado “votador6.bdf”, sendo este a base de um novo projeto.

Neste momento, como criou um símbolo para o circuito anterior, pode utilizá-lo no novo projeto. Ao fazer “Edit->Insert Symbol”, poderá ver um componente “votador”, na pasta “Project”.

Suponha agora a seguinte situação. Em diversos sistemas de votação, os votos são agrupados por zonas. Um certo conjunto de pessoas vota nessa zona, e o total de votos dessas pessoas é contabilizado como um único voto. A votação global corresponde à maioria dos votos agrupados. Por exemplo, os alunos na Universidade de Coimbra têm um certo conjunto de assentos no Senado da Universidade. Os alunos tomam decisões e fazem votações em assembleia magna, indo depois os seus representantes participar em votações do Senado. Trata-se de um sistema a dois níveis, tal como é mostrado na figura seguinte.

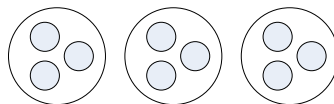


Figura 3 – Cada grupo de três conta apenas com um voto

Assim, utilizando o componente votador, implemente agora um sistema de votação de dois níveis, tal como é mostrado na Figura 4. Note que neste caso, como cada grupo de três conta apenas com um voto, podem existir empates. Assim, o novo circuito tem três saídas¹:

- MAIOR1: que está a 1 quando a maioria é 1.

¹ MAIOR é a abreviatura de “MAIORIA”, neste trabalho.

- EMPATE: que está a 1 quando existe um empate
- MAIOR0: que está a 1 quando a maioria é 0.

Faça a tabela verdade (terá de a mostrar ao professor) e aplique os teoremas necessários provando que o circuito mostrado cumpre a função que se pretende.

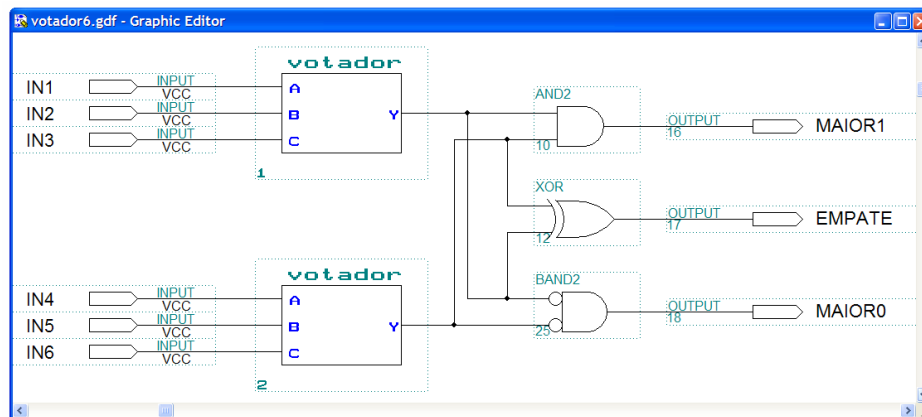


Figura 4 – Circuito votador de 6 entradas

Note que se clicar duas vezes em cima de um dos circuitos votador de três entradas, terá acesso ao circuito que está lá dentro. Deste modo, pode navegar nos diversos níveis do circuito. Note também que BAND corresponde a um AND com entradas negadas.

Na fase seguinte deste projeto, irá ligar as saídas a LEDs e as entradas a interruptores, à sua escolha.

Após compilar o circuito anterior sem problemas, atribua os pinos a cada uma das suas entradas ou saídas (consulte o manual da placa, como fez no trabalho laboratorial anterior), de forma a poder confirmar o funcionamento correto do seu circuito.

Compile, programe ("Tools->Programmer", escolhendo a opção "Program/Configure" e clicando "Start") e experimente o dispositivo.

Fase 3 (trab07c) – Circuito de Votação em VHDL

Se reparar, no esquema anterior ficou com dois votadores de três entradas e uma saída implementados como um bloco, e com um conjunto de lógica, que na prática corresponde a um votador para duas entradas. Ou seja, o AND, o XOR e o BAND poderiam (e deveriam) estar também num bloco separado, tal como mostra a figura seguinte.

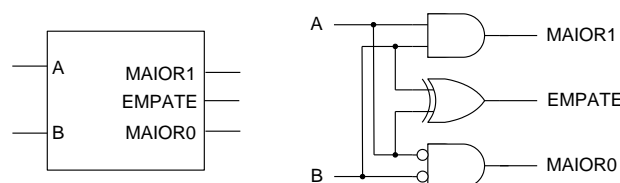


Figura 5 – Votador de duas entradas e três saídas

Nesta fase do trabalho irá proceder a essa alteração. Para começar, apague o conjunto de lógica que representa o votador.

Agora irá descrever o circuito da Figura 5 não utilizando portas, mas sim usando a linguagem VHDL. Para tal, no projeto em que está a trabalhar, faça "File->New->VHDL

File". Guarde o ficheiro como "votador2.vhd". Isto quer dizer que vai implementar um novo circuito chamado "votador2".

Um ficheiro VHDL possui duas partes: a *ENTITY*, que define os portos de entrada e saída do circuito; e a *ARCHITECTURE*, que define a arquitetura (no nosso caso as equações) que descrevem o circuito.

Consideremos o caso do circuito votador simples, de três entradas e uma saída - a sua descrição em VHDL está representada na Figura 6.

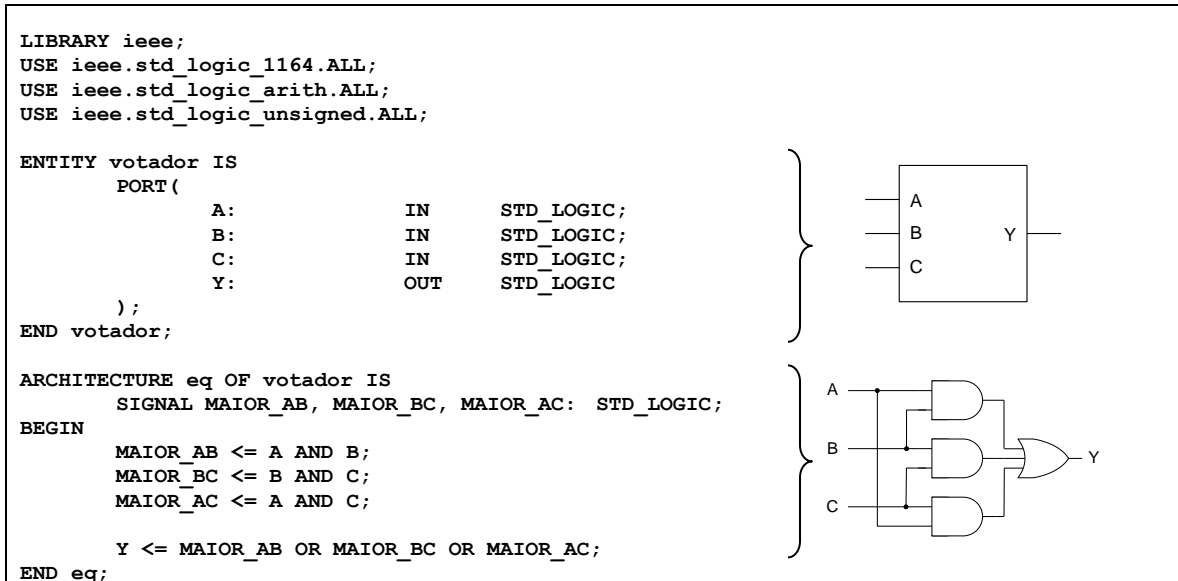


Figura 6 - Exemplo de descrição VHDL - circuito votador simples, com 3 entradas e 1 saída.

O ficheiro começa por incluir as bibliotecas necessárias. Depois é dito que o circuito votador tem quatro portos: três de entrada (A, B, C) e um de saída (Y). Na arquitetura (ARCHITECTURE) começa-se por definir três sinais (SIGNAL) auxiliares (MAIOR_AB, MAIOR_BC, MAIOR_AC). Em seguida escrevem-se as equações que representam o circuito.

Escreva agora o ficheiro VHDL correspondente ao circuito votador com duas entradas e três saídas.

Antes de compilar, deve indicar que este é agora o ficheiro principal (de topo) de compilação. Para tal, faça "Project -> Set as Top-Level Entity". Compile o ficheiro "votador2.vhd", e observe que não contém erros. Agora necessita de criar um símbolo para o circuito: tendo o ficheiro aberto, faça "File -> Create/Update -> Create Symbols for Current File". Volte novamente ao circuito "votador6.bdf", coloque-o como entidade de topo e introduza agora o votador2 que acabou de criar. Note que ao fazer "Symbol->Enter Symbol" irá perceber que possui um novo componente na pasta "Project". Interligue o "votador2" ao seu circuito, compile e teste o resultado programando novamente a placa. O circuito deverá ter o aspecto indicado abaixo. Depois de reprogramar o circuito, volte a testá-lo. Verá que funciona.

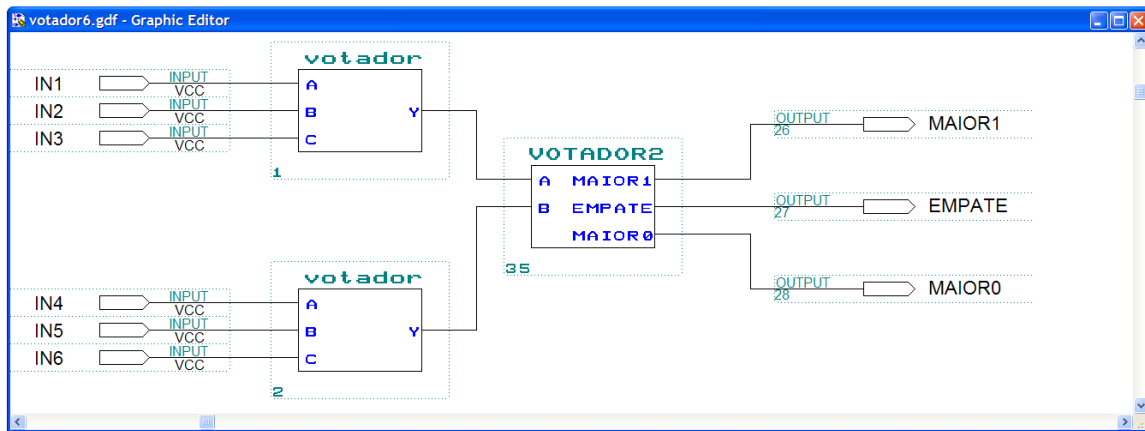


Figura 7 – Circuito de votação final

Antes de dar o trabalho por concluído ou de sair da aula, chame o docente e apresente o resultado das várias tarefas que realizou neste trabalho.

Tarefas		
1	Votador simples (implementação e teste)	<input type="checkbox"/>
2	Tabela de verdade do circuito votador hierárquico	<input type="checkbox"/>
3	Implementação e teste do circuito votador hierárquico	<input type="checkbox"/>
4	Programação de votador para 2 entradas	<input type="checkbox"/>
5	Teste do circuito votador hierárquico modificado	<input type="checkbox"/>