

Experimento 4

Circuitos Combinacionais : Comparador de Palavras

Isaac Lopes, 12/0120801

Lucas Mafra Chagas, 12/0126443

Marcelo Giordano Martins Costa de Oliveira, 12/0037301

¹Dep. Ciência da Computação – Universidade de Brasília (UnB)
CiC 116351 - Circuitos Digitais - Turma C

{giordano.marcelo, chagas.lucas.mafra, isaaclopinho}@gmail.com

Abstract. *This report is the description and design of a comparator of binary words, using circuit synthesis techniques combinational seen, as truth table, simplification using Karnaugh map, and implementation of the simplified boolean function with logic gates.*

Resumo. *Esse relatório visa a descrição e projeto de um comparador de palavras binárias, usando técnicas de síntese de circuitos combinacionais já vistas, como tabela verdade, simplificação por mapa de Karnaugh, e implementação da função booleana simplificada com portas lógicas.*

Objetivos

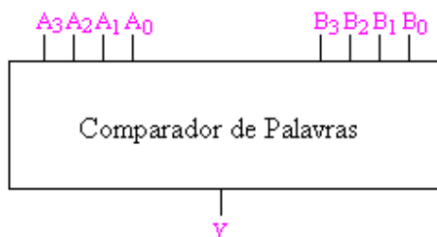
Os objetivos do presente relatório são de usar o sistema Quartus II para a implementação de circuitos com comparadores de palavras binárias com as técnicas utilizadas em relatórios passados de simplificação e montagem.

Materiais

- software Quartus-II v13.0

Introdução

Um comparador é um circuito combinatório operativo. Ele permite comparar grandezas de dois números binários. Um comprimento de uma palavra binária é o número de bits que a compõem.



Um comparador tem como saída 1 se os comprimentos forem iguais, caso contrário a saída será 0. Ele também pode possuir três saídas:

- se $A=B$
- se $A<B$
- se $A>B$

Para uma boa realização de um circuito combinacional é necessário seguir alguns passos, como os mostrados abaixo:

- Descrever sistema;
- Elaborar tabela da verdade;
- Obter funções booleanas a partir da tabela verdade;
- Simplificar funções booleanas obtidas ;
- Elaborar diagrama lógico.

Dessa maneira, fica mais intuitivo a realização da montagem e análise do circuito.

Procedimentos

Parte 1

- Complete a tabela da verdade abaixo do circuito XNOR e obtenha a sua função booleana Z_i .
- Modifique a função Z_i obtida no item a de forma a usar apenas portas NAND de 2 entradas.
- Desenhe, como um subcircuito, o diagrama lógico parcial da comparação do par de bits (A_i , B_i), realize a simulação funcional e verifique a tabela da verdade do item a.
- Desenhe o circuito total e faça a simulação funcional do diagrama lógico total, verificando a tabela verdade obtida.
- Realize a simulação temporal do circuito e comente os resultados obtidos justificando-os.

Parte 2

- Elabore a tabela da verdade parcial e obtenha as funções booleanas parciais do circuito Comparador de 1 bit.
- Minimize as expressões obtidas no item a.
- Desenhe o subcircuito e faça a simulação funcional do diagrama lógico parcial do Comparador de 1 bit, comparando com a tabela verdade do item a.
- Desenhe o circuito e realize a simulação funcional do diagrama lógico total do Comparador de 2 bits, escrevendo a tabela verdade obtida.
- Realize a simulação temporal do circuito do Comparador de 2 bits e comente os resultados.

Análise dos Resultados

Análise: Parte 1

A partir da tabela verdade 1 obtivemos a equação para montar o circuito comparador de 1 bit.

Table 1. Tabela XNOR(Comparador de 1 bit)

A	B	Zi
0	0	1
0	1	0
1	0	0
1	1	1

Com a simplificação da tabela 1 obtivemos a seguinte equação:

$$Z = \overline{A} \overline{B} + AB = \overline{A \oplus B} = \overline{AB} + \overline{AB} = \overline{AB} \overline{AB}$$

O diagrama da função encontrada está representado na figura 1 abaixo:

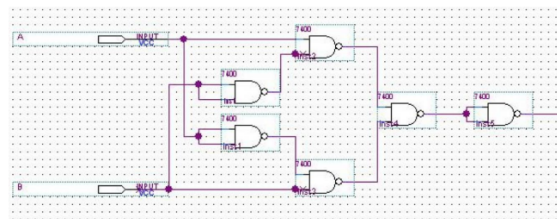


Figure 1. Diagrama Lógico do Comparador de 1 bit

Já um comparador de 3 bits pode ser implementado pela figura 2:

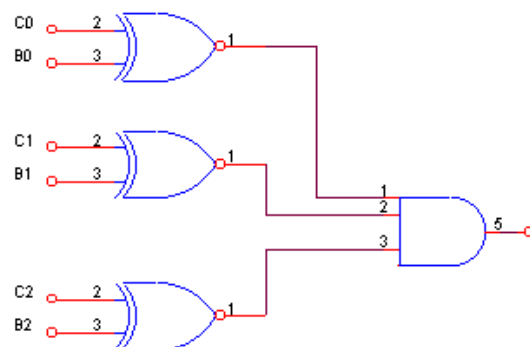


Figure 2. Circuito Comparador de 3 bits.

Como no experimento é pedido apenas com portas NAND, o circuito, na figura 3 e sua wave form na figura 4, foi implementado da seguinte maneira:

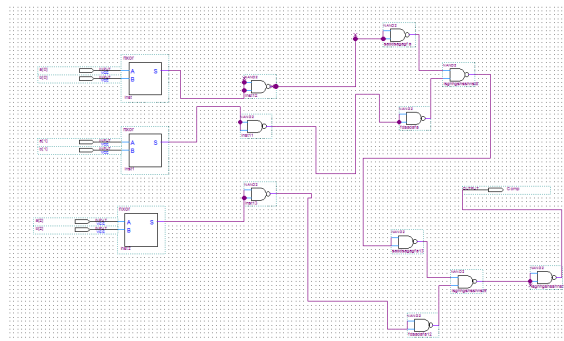


Figure 3. Diagrama total do Circuito comparador de 3 bits

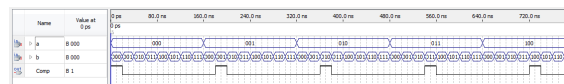


Figure 4. Forma de onda do circuito comparador de 3 bits

Analise: Parte 2

Preenchendo-se a tabela verdade parcial do comparador de 1 bit, tal que $Y1 = 1 \leftrightarrow A > B$, $Y2 = 1 \leftrightarrow A = B$ e $Y3 = 1 \leftrightarrow A < B$, obtivemos a equação necessária para montar o circuito comparador de igualdade, minoridade e maioridade de 1 bit.

Table 2. Comparador de 1 bit

A	B	Y1= $A > B$	Y2= $A = B$	Y3= $A < B$
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

Sendo,

$$Y1 = \overline{A}B$$

$$Y2 = \overline{A \oplus B}$$

$$Y3 = A\overline{B}$$

Através de Y1, Y2 e Y3, montamos o circuito comparador de 1 bit, como mostrado na figura 5 abaixo:

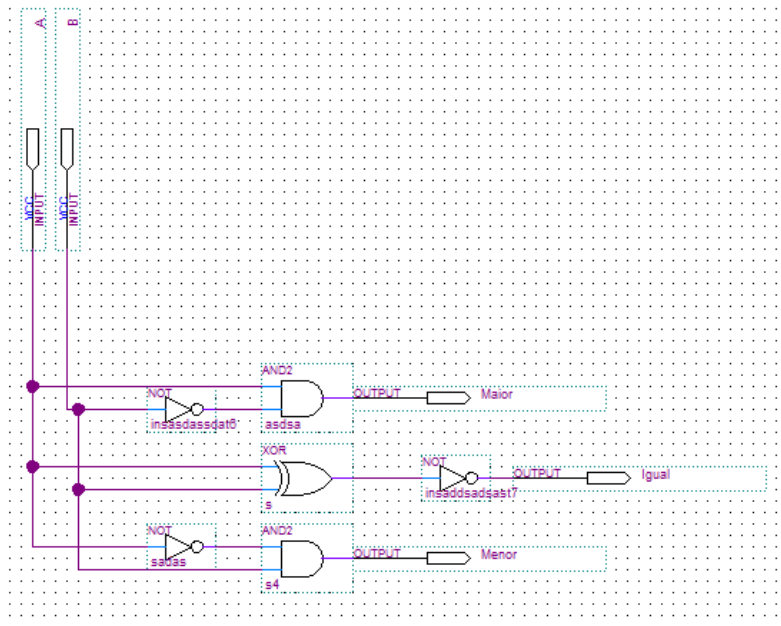


Figure 5. Circuito Comparador de 1 bit

E, finalmente, utilizando o comparador de 1 bit, implementamos o circuito comparador de 2 bits representados pelas figuras 6 e 7:

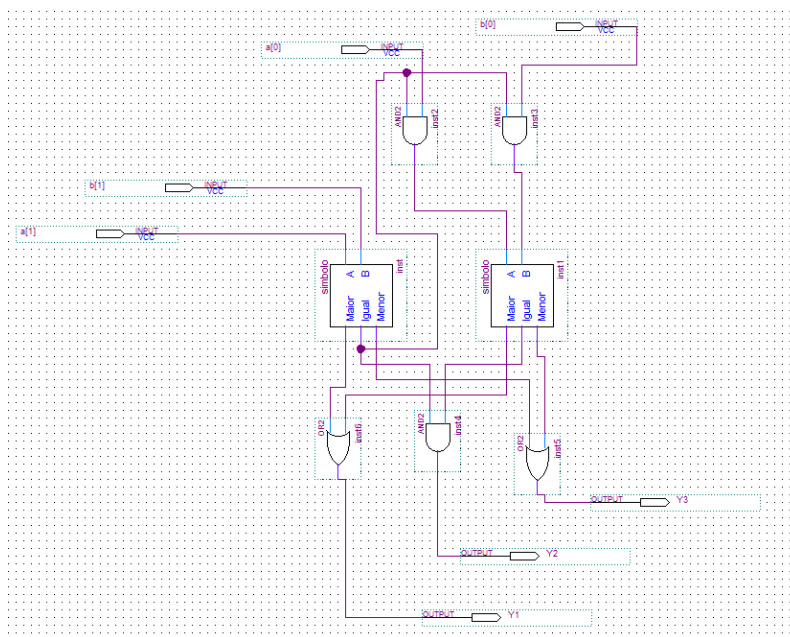


Figure 6. Circuito Comparador de 2 bit

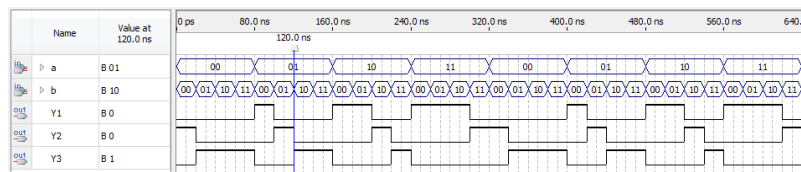


Figure 7. Wave Form do Comparador de 2 bit

Conclusão

Foi estudado nesse relatório o uso de comparadores de bits, bem como o uso da ferramenta Quartus II para a implementação de circuitos. Foram usados métodos conhecidos da síntese de circuitos combinacionais já vistos em aulas teóricas. Pode-se afirmar que, com a semelhança dos resultados nas tabelas verdades com as tabelas do Quartus II que foi um experimento de bastante sucesso.

Auto-Avaliação

1. b
2. d
3. d
4. b
5. b