SCE 0110 - Elementos de Lógica Digital I

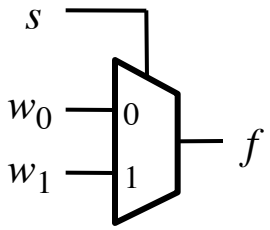
Blocos de construção de circuitos combinacionais

Prof. Dr. Vanderlei Bonato

Outline

- Multiplexadores/Multiplex/MUX
  - Seleção de sinais/canais
  - Implementação de funções lógicas de aplicação geral
    - Teorema de expansão de Shannon

Multiplexador de 2 canais

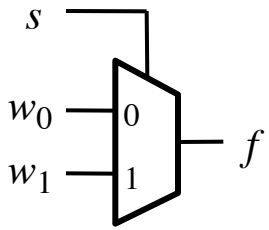


(a) Graphical symbol

s	f
0	w_0
1	w_1

(b) Truth table

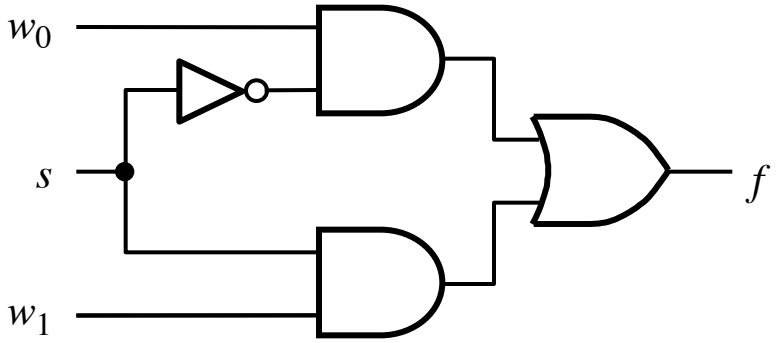
Construa o circuito lógico no modo soma dos produtos



(a) Graphical symbol

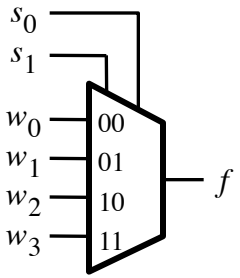
s	f
0	w_0
1	w_1

(b) Truth table



(c) Sum-of-products circuit

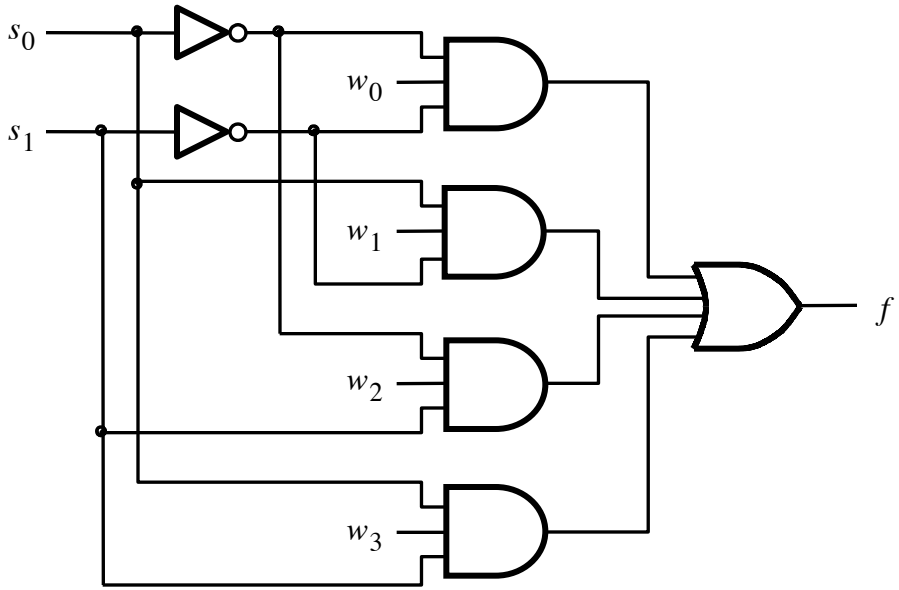
Figure 6.1. A 2-to-1 multiplexer.



(a) Graphic symbol

s_1	s_0	f
0	0	w_0
0	1	w_1
1	0	w_2
1	1	w_3

(b) Truth table



(c) Circuit

Figure 6.2. A 4-to-1 multiplexer.

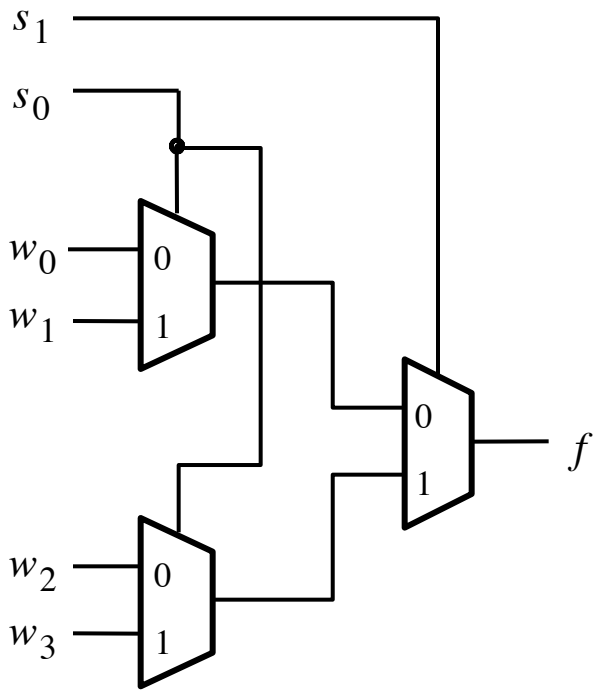


Figure 6.3. Using 2-to-1 multiplexers to build a 4-to-1 multiplexer.

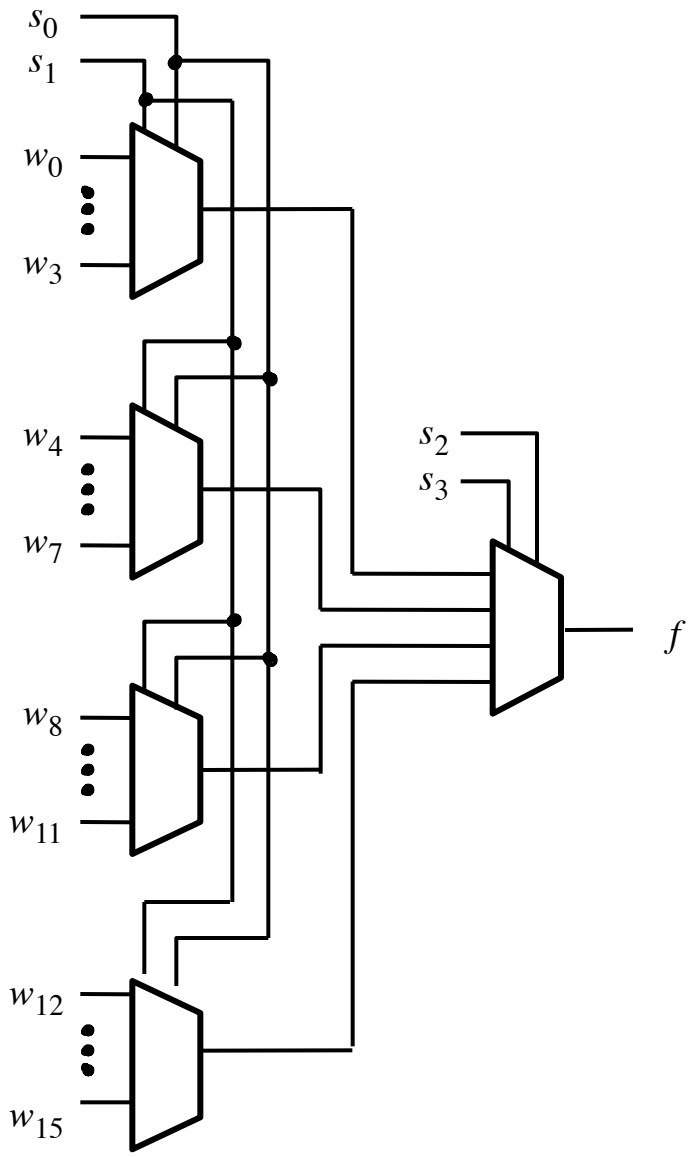
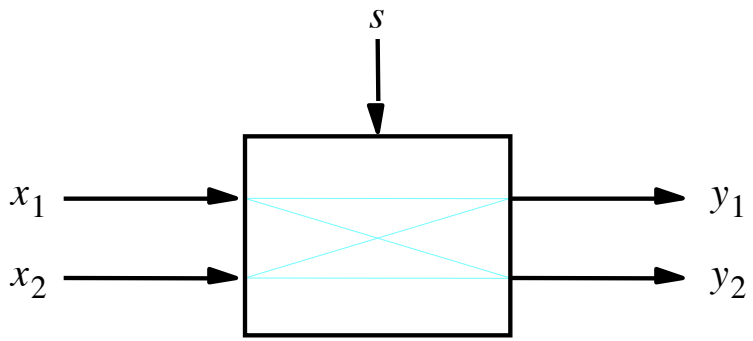
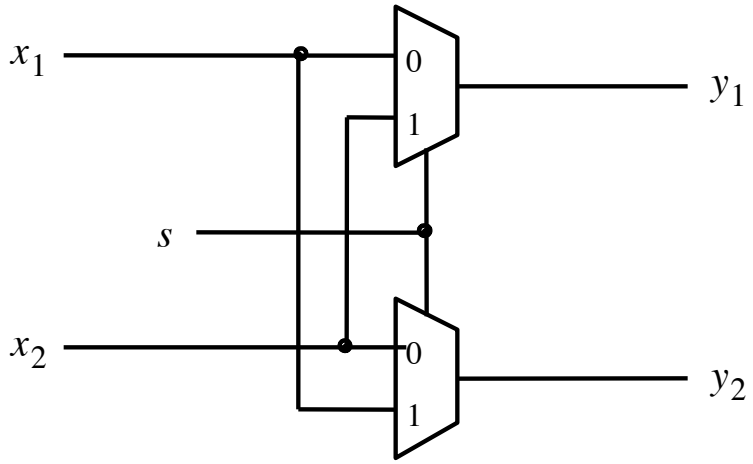


Figure 6.4. A 16-to-1 multiplexer.



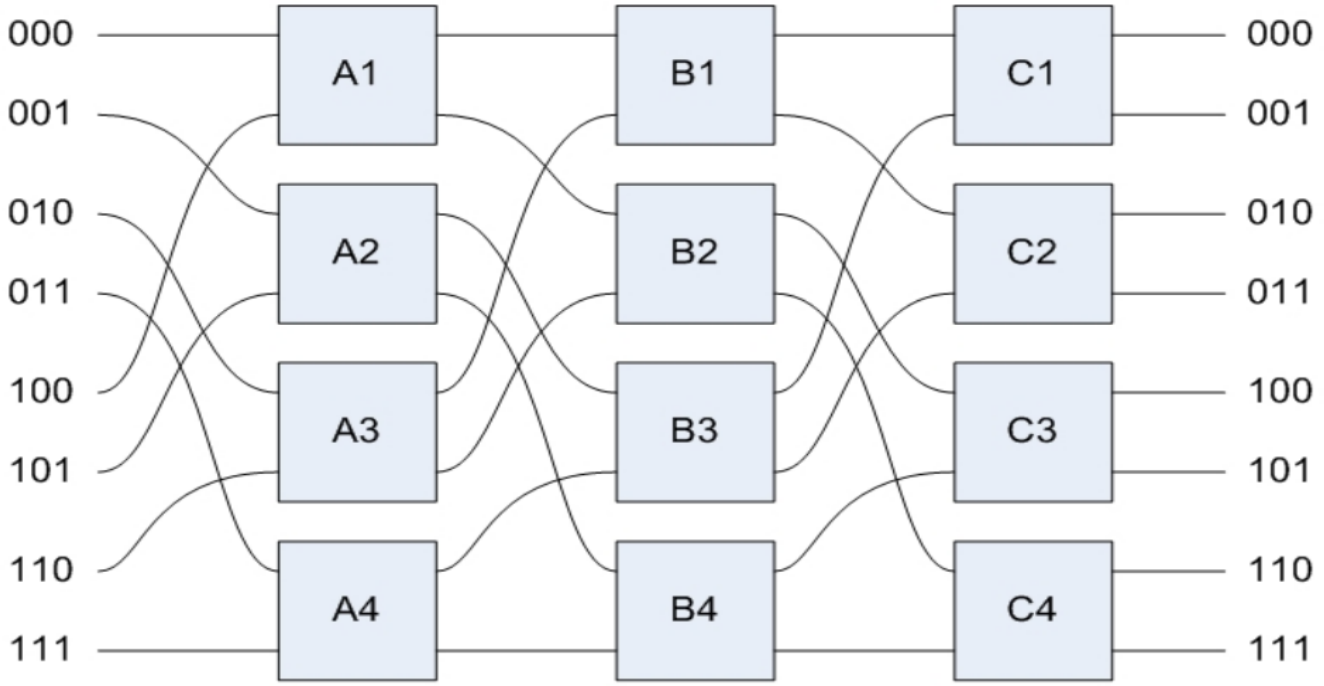
(a) A 2x2 crossbar switch



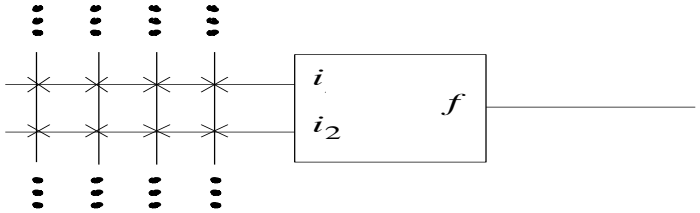
(b) Implementation using multiplexers

Figure 6.5. A practical application of multiplexers.

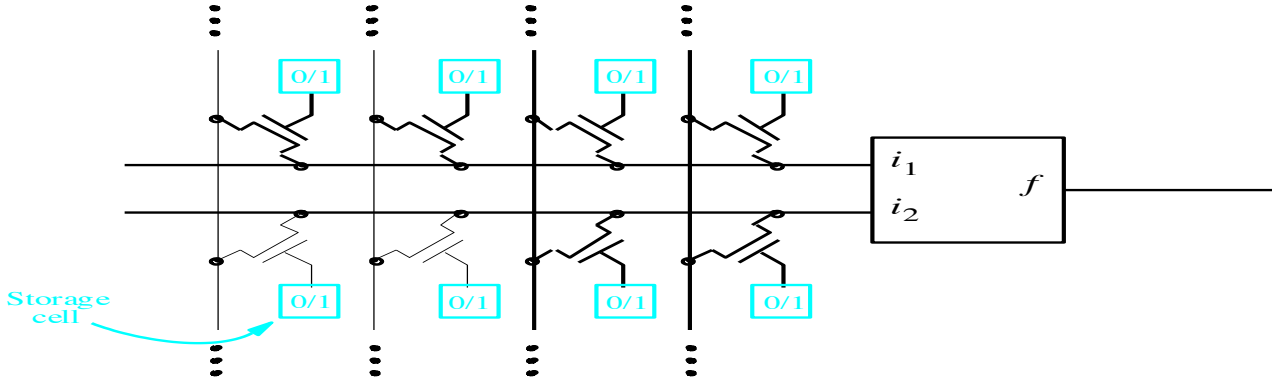
Exemplo de uso de comutadores com rede Ômega



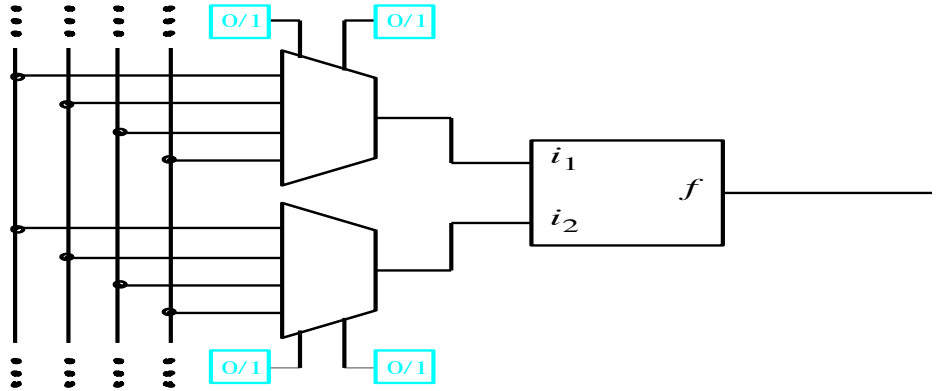
Na rede ômega, se os endereços de saída forem determinados de modo seqüencial a partir de zero e se o comutador for controlado com 0 para uma conexão direta e 1 cruzada, a comutação pode ser feita simplesmente de acordo com o resultado de um XOR entre o endereço de origem e destino. Ex. de 010 para 110 = 100



(a) Part of the FPGA in Figure 3.39

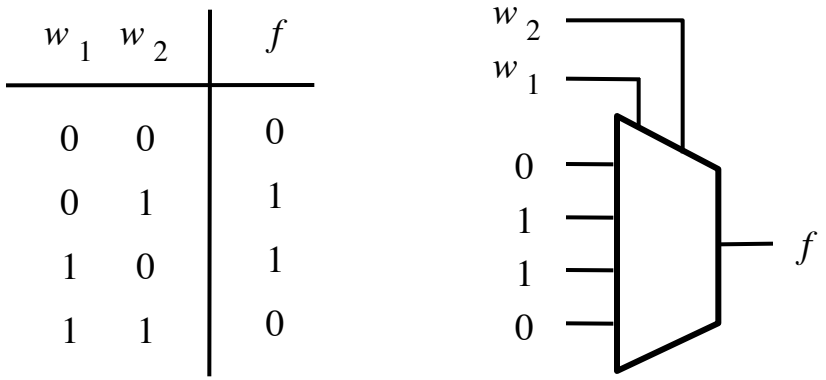


(b) Implementation using pass transistors

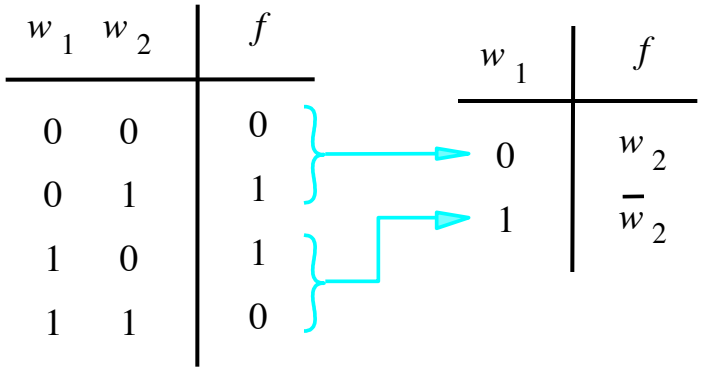


(c) Implementation using multiplexers

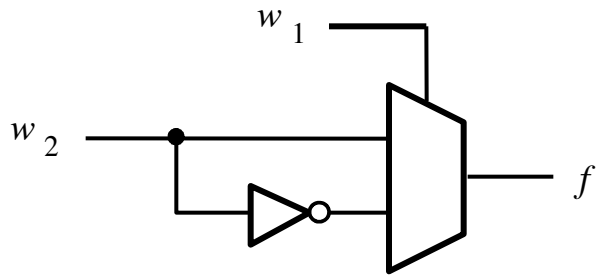
Figure 6.6. Implementing programmable switches in an FPGA.



(a) Implementation using a 4-to-1 multiplexer

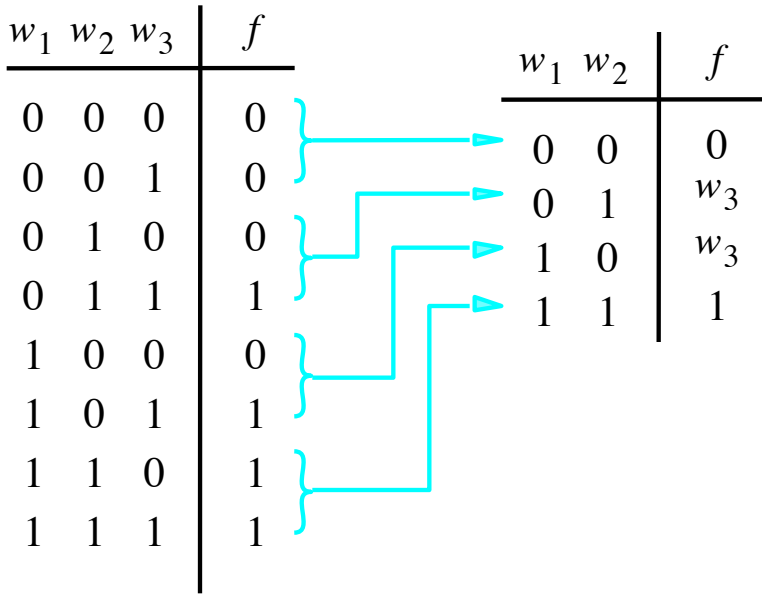


(b) Modified truth table

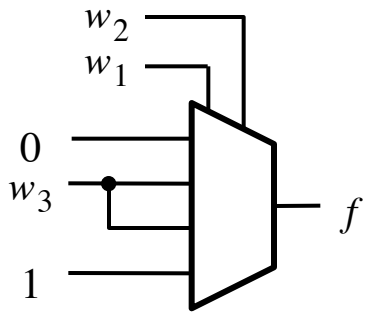


(c) Circuit

Figure 6.7. Synthesis of a logic function using multiplexers.



(a) Modified truth table

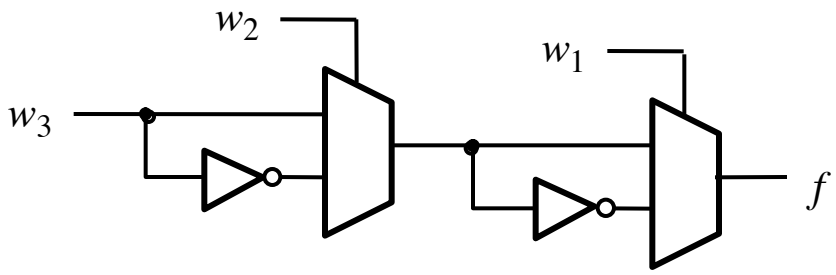


(b) Circuit

Figure 6.8. Implementation of the three-input majority function using a 4-to-1 multiplexer.

w_1	w_2	w_3	f	
0	0	0	0	
0	0	1	1	$w_2 \oplus w_3$
0	1	0	1	
0	1	1	0	
1	0	0	1	
1	0	1	0	$\overline{w_2 \oplus w_3}$
1	1	0	0	
1	1	1	1	

(a) Truth table

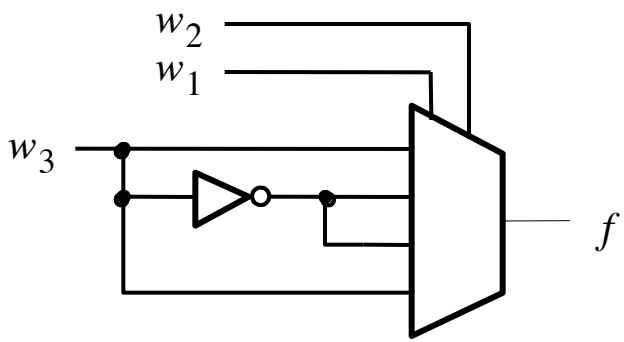


(b) Circuit

Figure 6.9. Three-input XOR implemented with 2-to-1 multiplexers.

w_1	w_2	w_3	f	
0	0	0	0	w_3
0	0	1	1	
0	1	0	1	$\bar{w}_3$
0	1	1	0	
1	0	0	1	$\bar{w}_3$
1	0	1	0	
1	1	0	0	w_3
1	1	1	1	

(a) Truth table



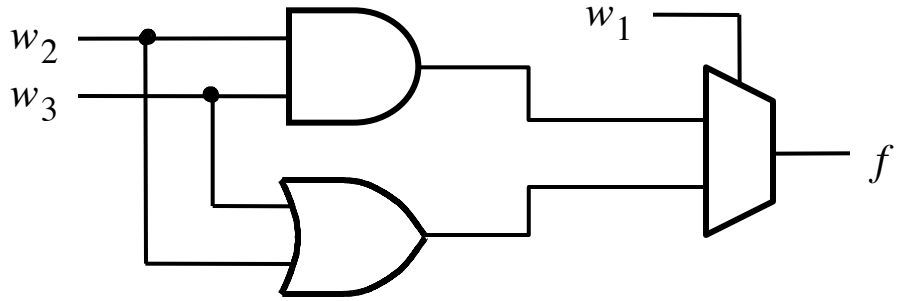
(b) Circuit

Figure 6.10. Three-input XOR function implemented with a 4-to-1 multiplexer.

w_1	w_2	w_3	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

w_1	f
0	w_2w_3
1	$w_2 + w_3$

(b) Truth table



(b) Circuit

Figure 6.11. The three-input majority function implemented using a 2-to-1 multiplexer.

Teorema da expansão de Shannon

- Qualquer função booleana $f(w_1,...,w_n)$ pode ser escrita na forma:

$$f(w_1,...,w_n) = \overline{w_1}.f(0,w_2,...,w_n) + w_1.f(1,w_2,...,w_n)$$

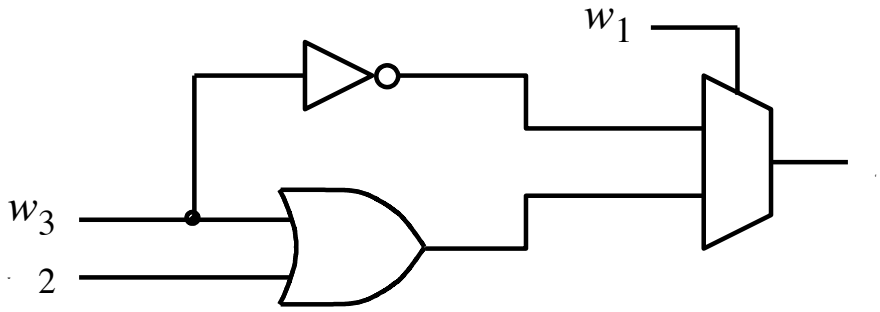
- A expansão pode ser feita para qualquer variável *n* e também para mais de uma variável ao mesmo tempo

Teorema da expansão de Shannon

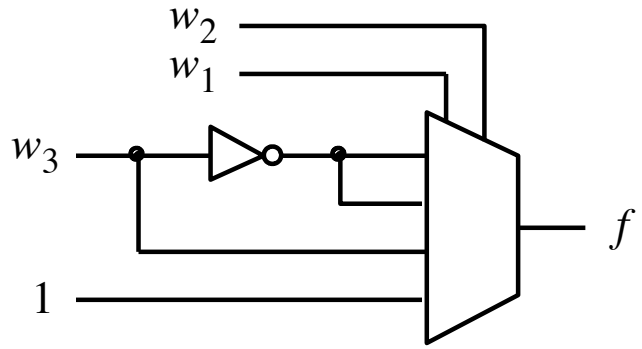
- Implemente a função abaixo usando um MUX de 2 canais/entradas e uma saída (2 para 1) e usando w_1 para controle do MUX

  $f = \overline{w}_1\overline{w}_3 + w_1w_2 + w_1w_3$

- Implemente a mesma função usando um MUX de 4 para 1 com controle através de w_1 e w_2



(a) Using a 2-to-1 multiplexer



(b) Using a 4-to-1 multiplexer

Figure 6.12. The circuits synthesized in Example 6.6.

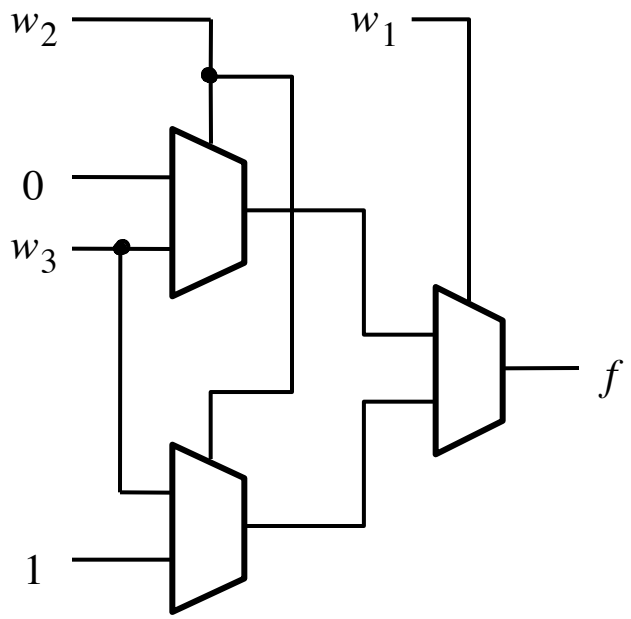
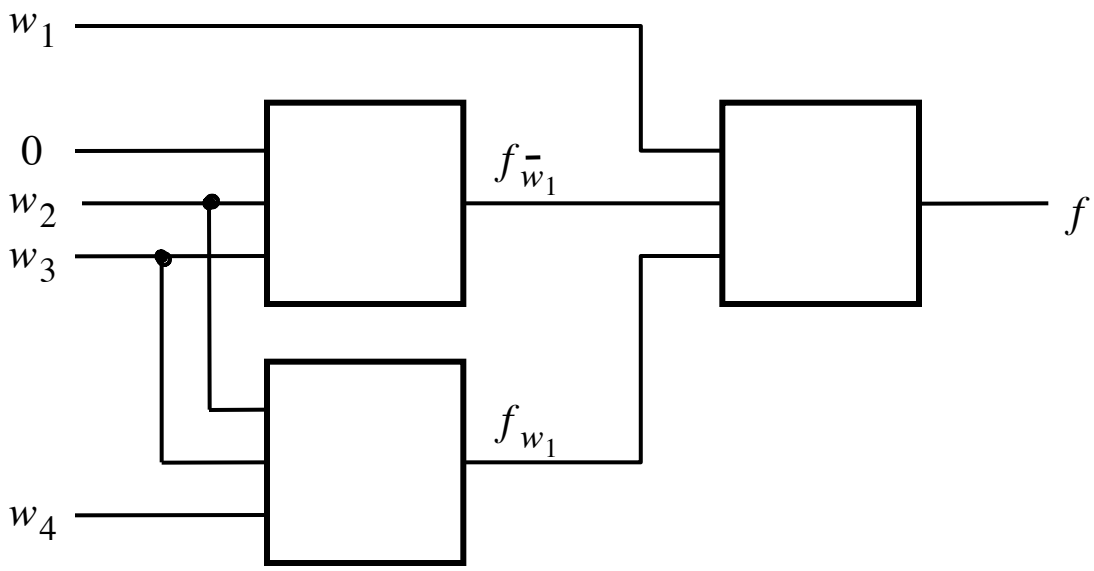
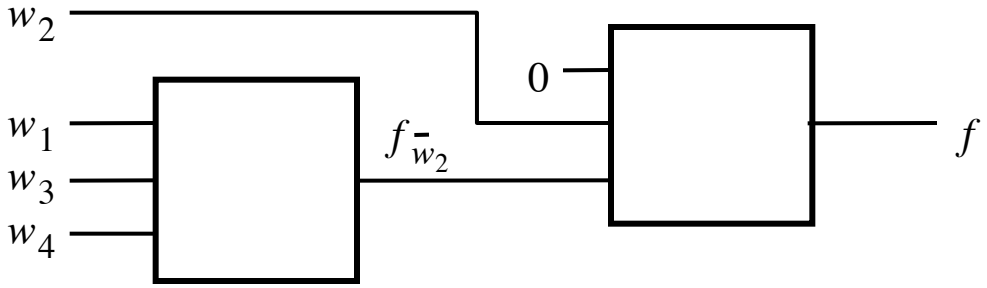


Figure 6.13. The circuit synthesized in Example 6.7



(a) Using three 3-LUTs



(b) Using two 3-LUTs

Figure 6.14. Circuits synthesized in Example 6.8