MC542

Organização de Computadores
Teoria e Prática

2006

Prof. Paulo Cesar Centoducatte

ducatte@ic.unicamp.br

www.ic.unicamp.br/~ducatte

MC542

Circuitos Lógicos

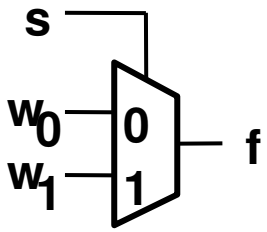
Circuitos Combinacionáis
Blocos Básicos

"Fundamentals of Digital Logic with VHDL Design" - (Capítulo 6)

Circuitos Combinacionáis: Blocos Básicos
Sumário

- Multiplexador
  - Implementação de Funções Lógicas Usando Mux
- Decodificador
- Codificador
- Codificador de Prioridade
- Conversores
- Comparador
- VHDL para Circuito Combinacional
  - Select
  - Atribuição Condicional
  - Processos
    - » If-Then-Else
    - » Case

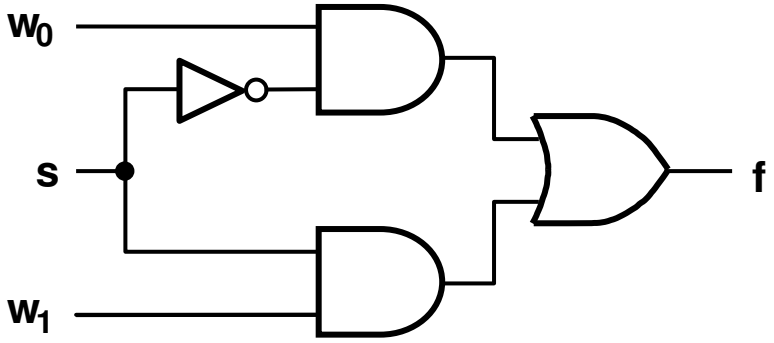
Multiplexador: 2 para 1 (2:1)



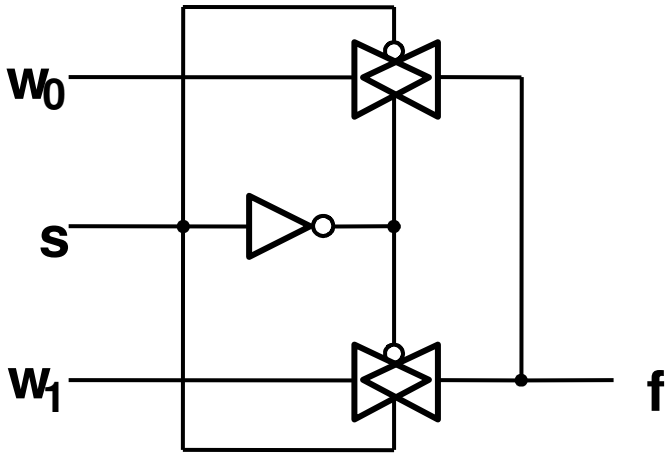
Símbolo Gráfico

s	f
0	w_0
1	w_1

Tabela Verdade

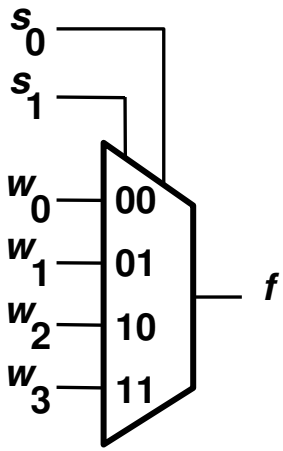


Suma-de-produtos

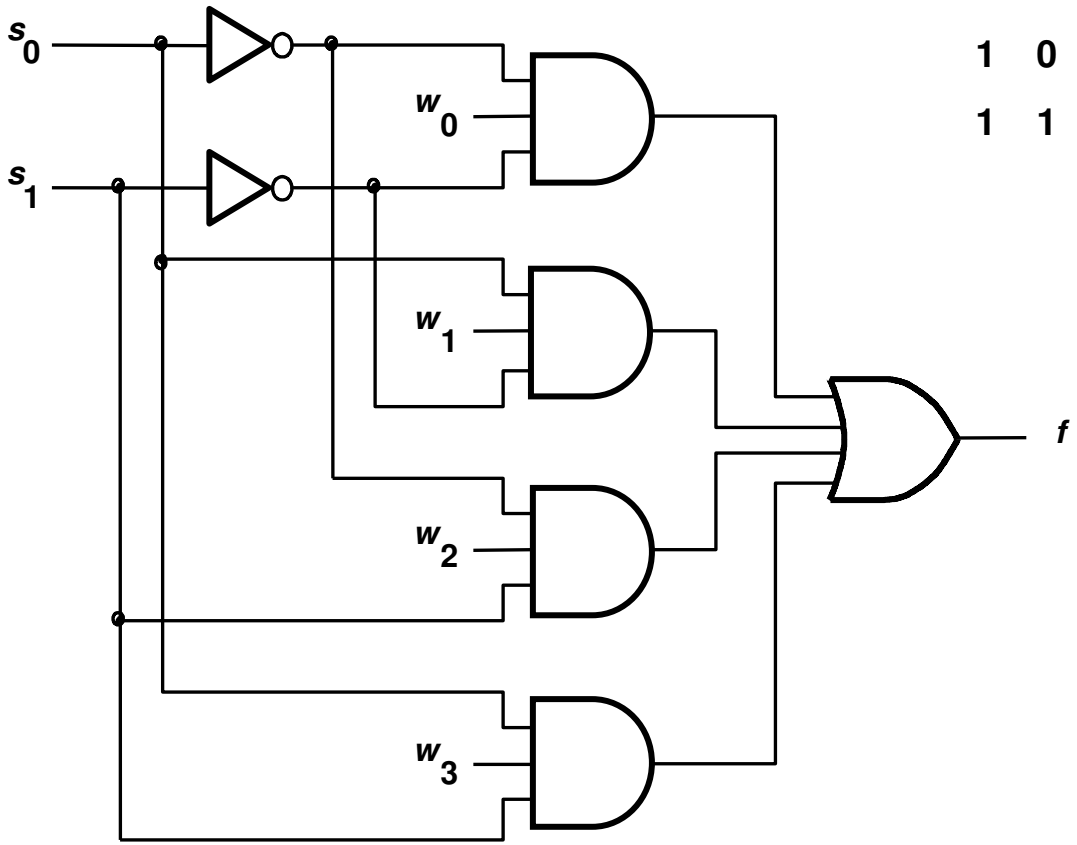


transmission gates

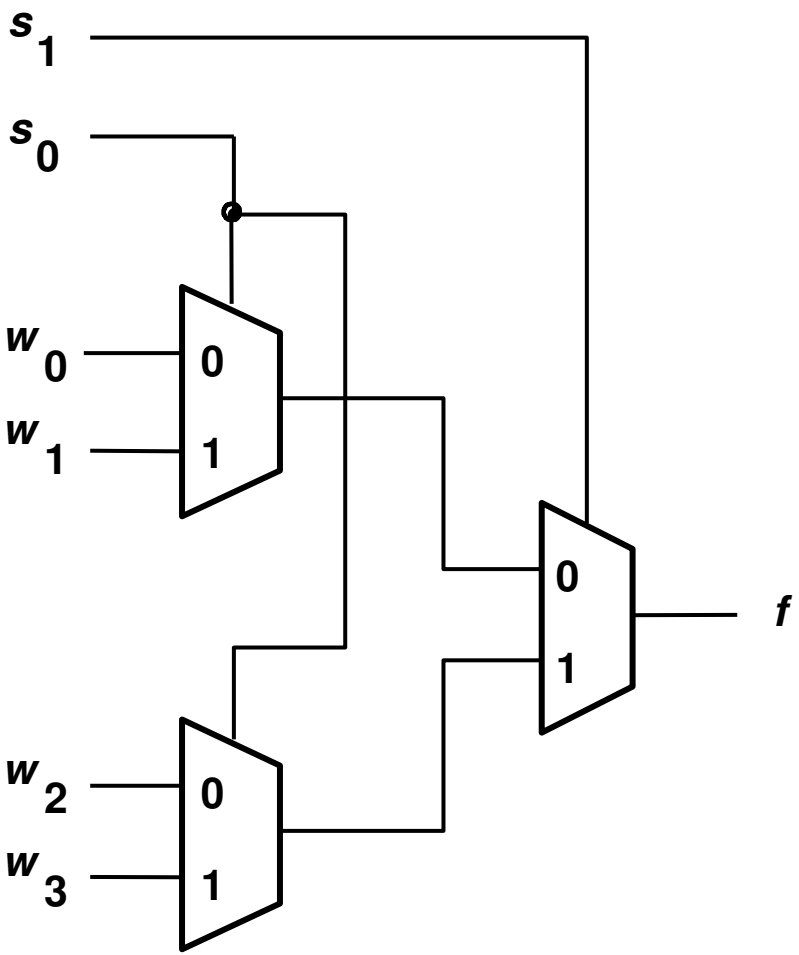
Multiplexador: 4 para 1 (4:1)



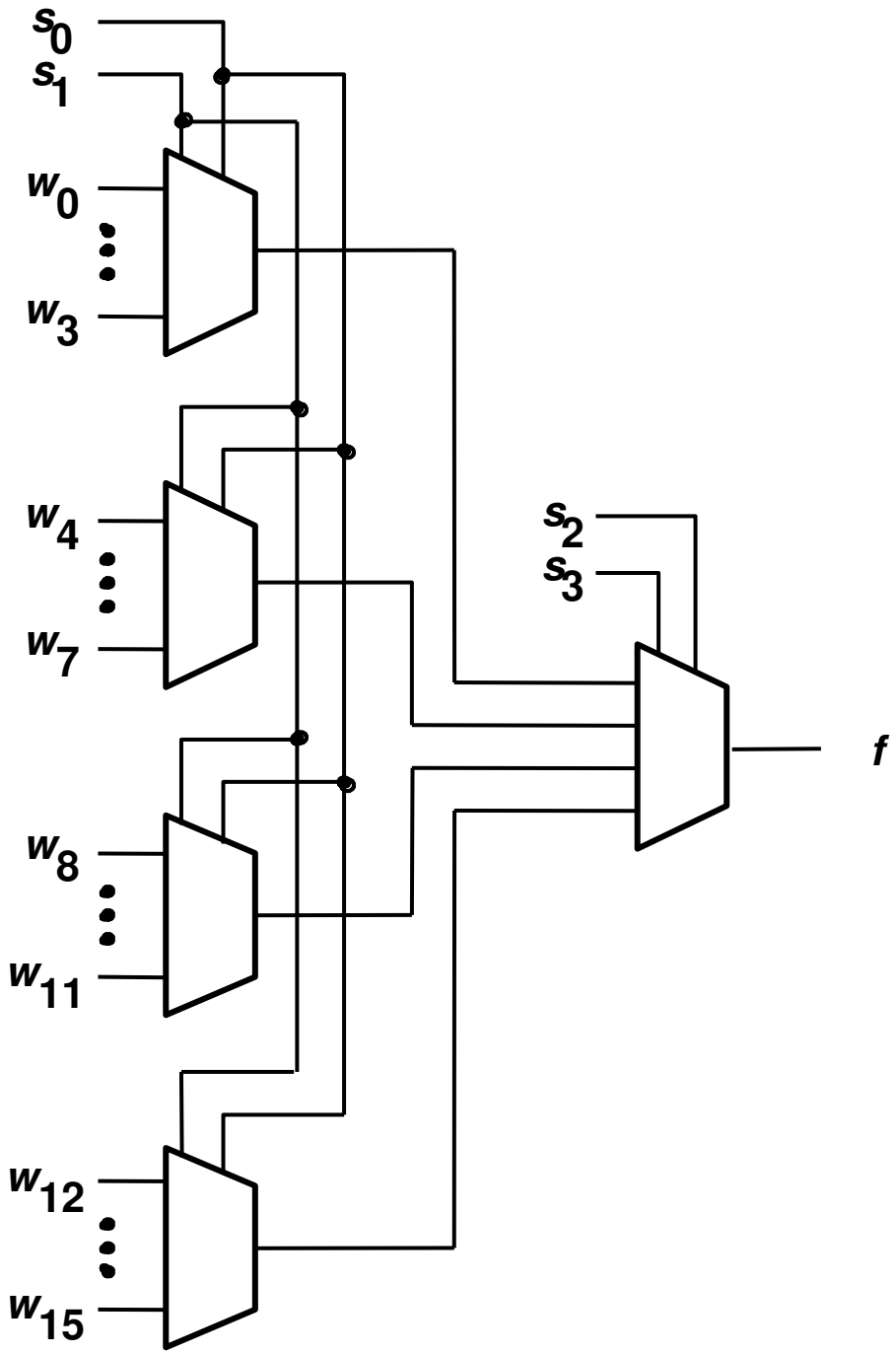
s_1	s_0	f
0	0	w_0
0	1	w_1
1	0	w_2
1	1	w_3



Mux 4:1 a partir de Mux 2:1

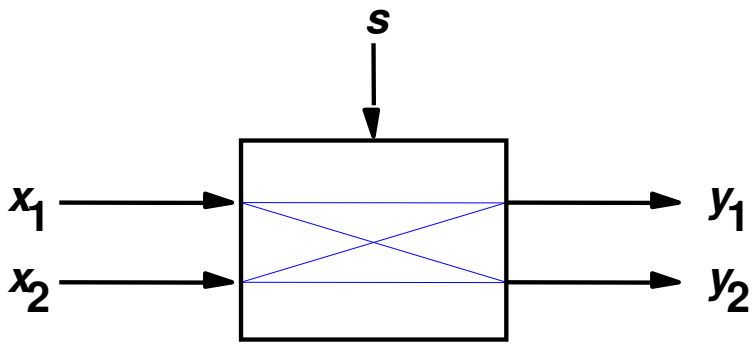


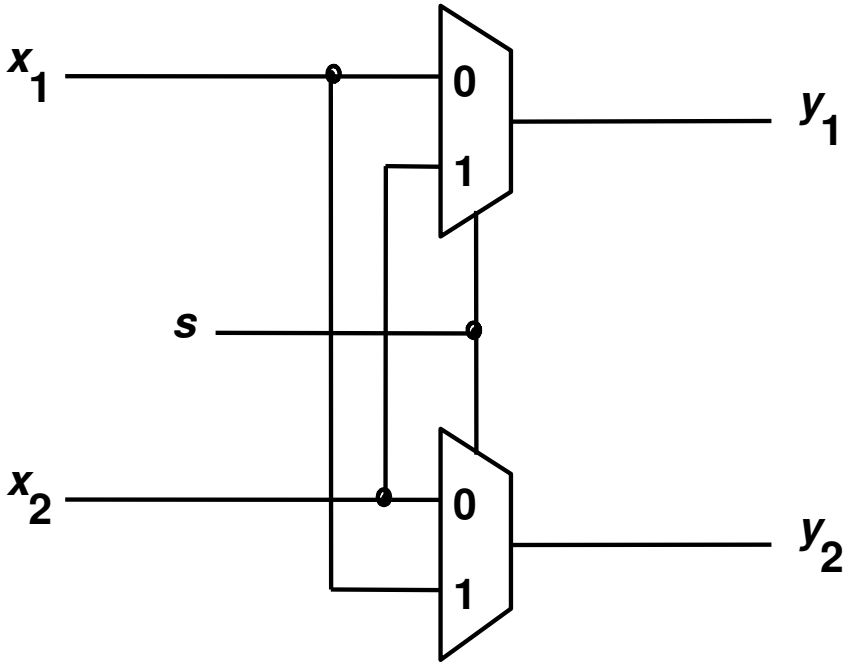
Mux 16:1



Exemplo de Uso de Mux

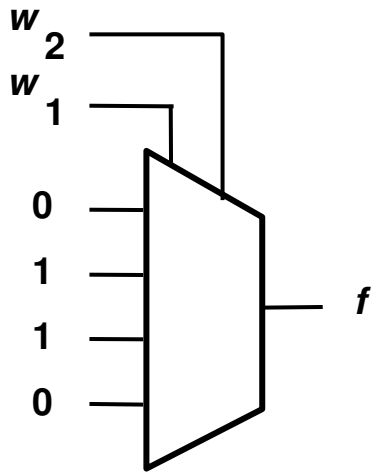
2x2 crossbar switch





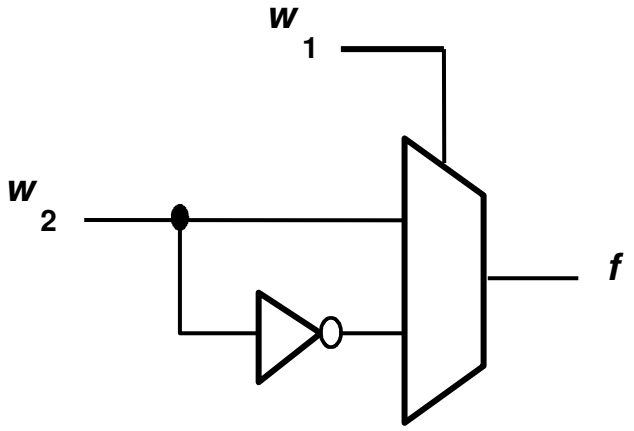
Síntese de Funções Lógicas Usando MUX

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



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

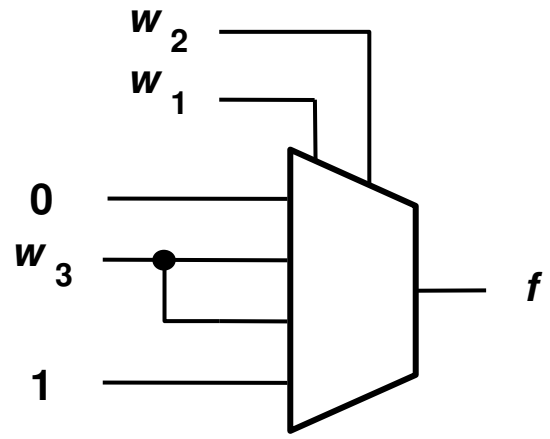
w_1	f
0	w_2
1	$\overline{w}_2$



Exemplo

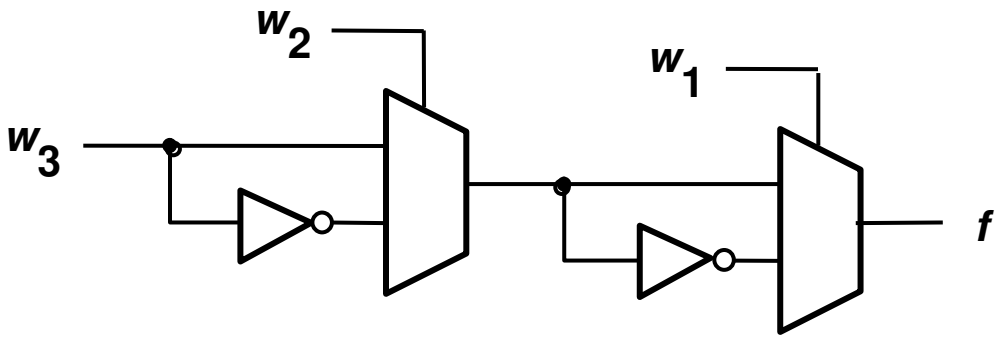
w_1	w_2	w_3	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

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



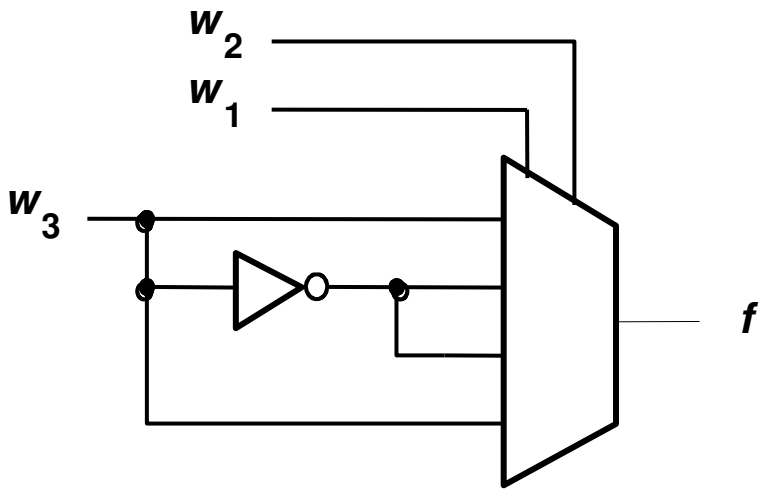
Exempo

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



Exemplo

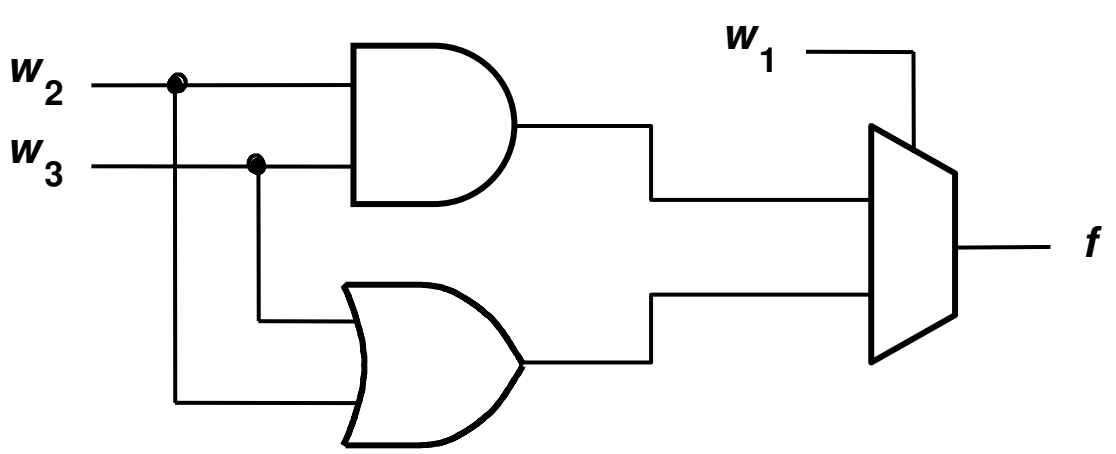
w_1	w_2	w_3	f	
0	0	0	0	w_3
0	0	1	1	
0	1	0	1	$\bar{w}_3$
0	1	1	0	
1	0	0	1	$\bar{w}_3$
1	0	1	0	
1	1	0	0	w_3
1	1	1	1	



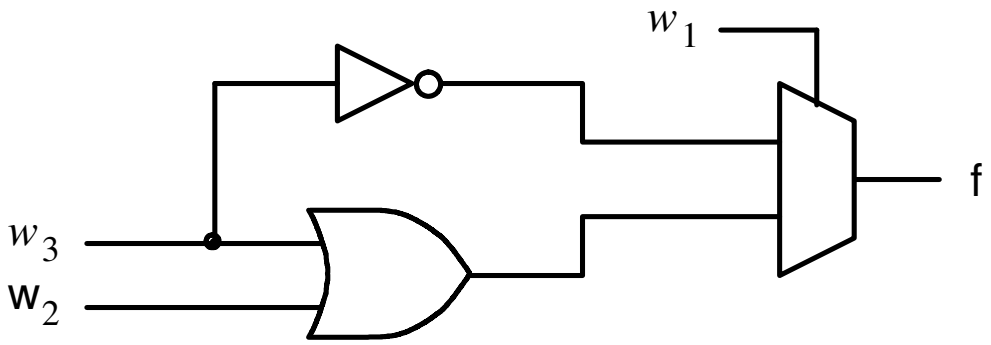
Exemplo
Maioria de uns

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

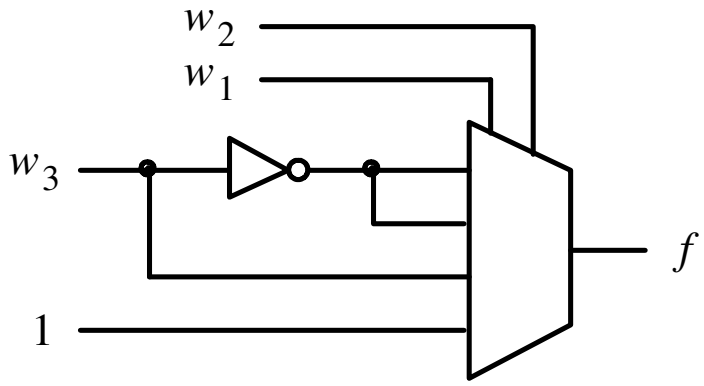
w_1	f
0	$w_2 w_3$
1	$w_2 + w_3$



Exemplo
Maioria de uns

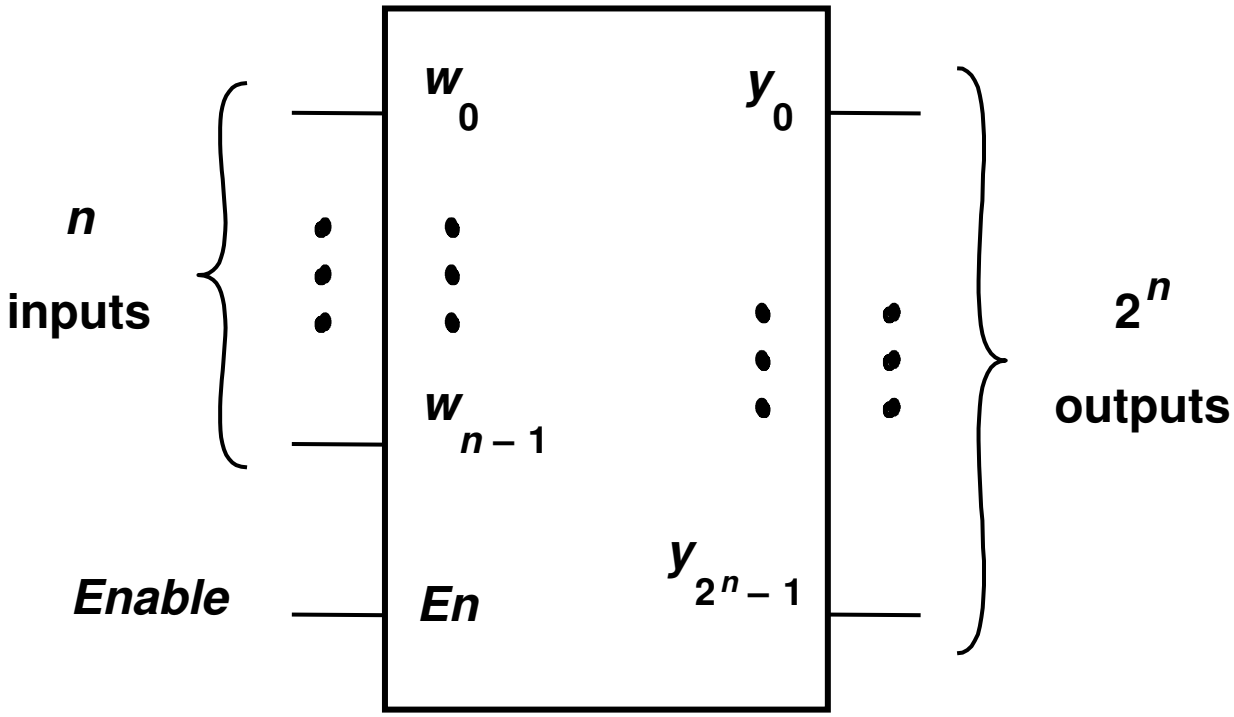


(a) Using a 2-to-1 multiplexer



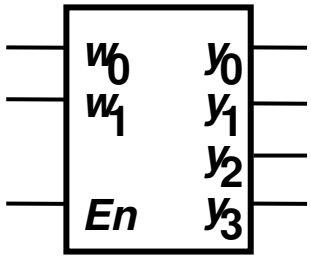
(b) Using a 4-to-1 multiplexer

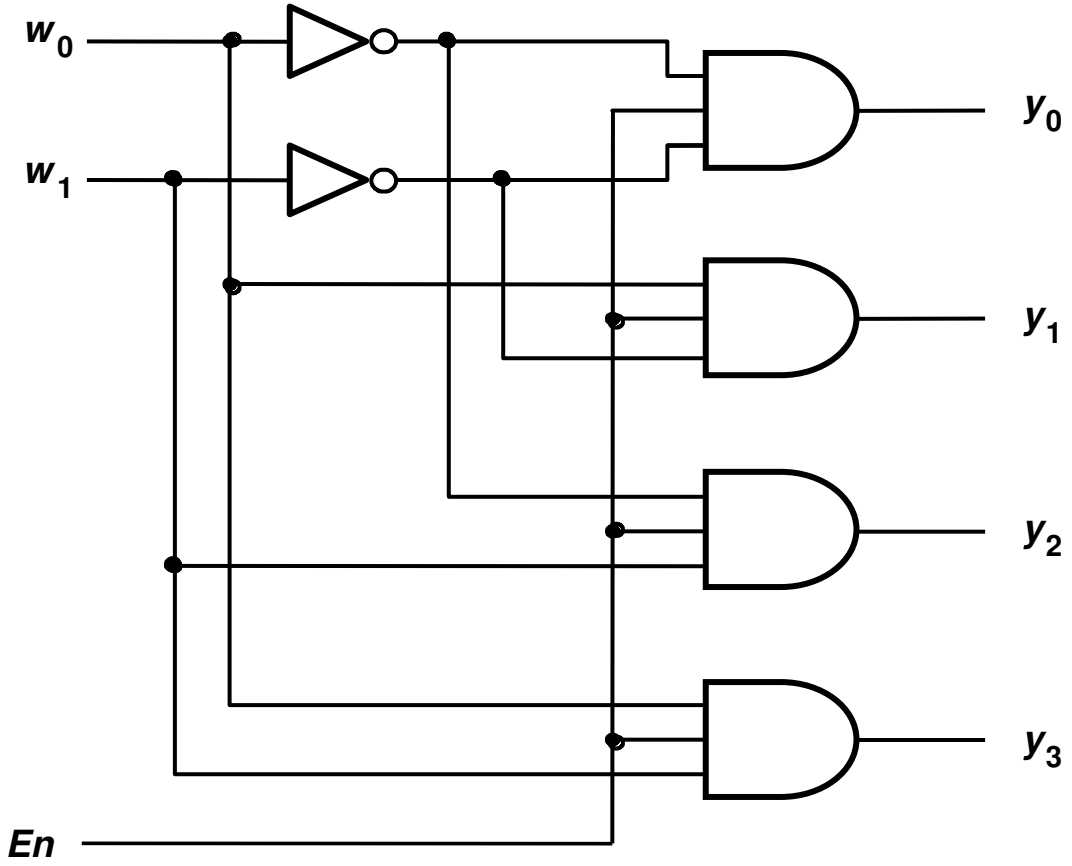
Decodificador



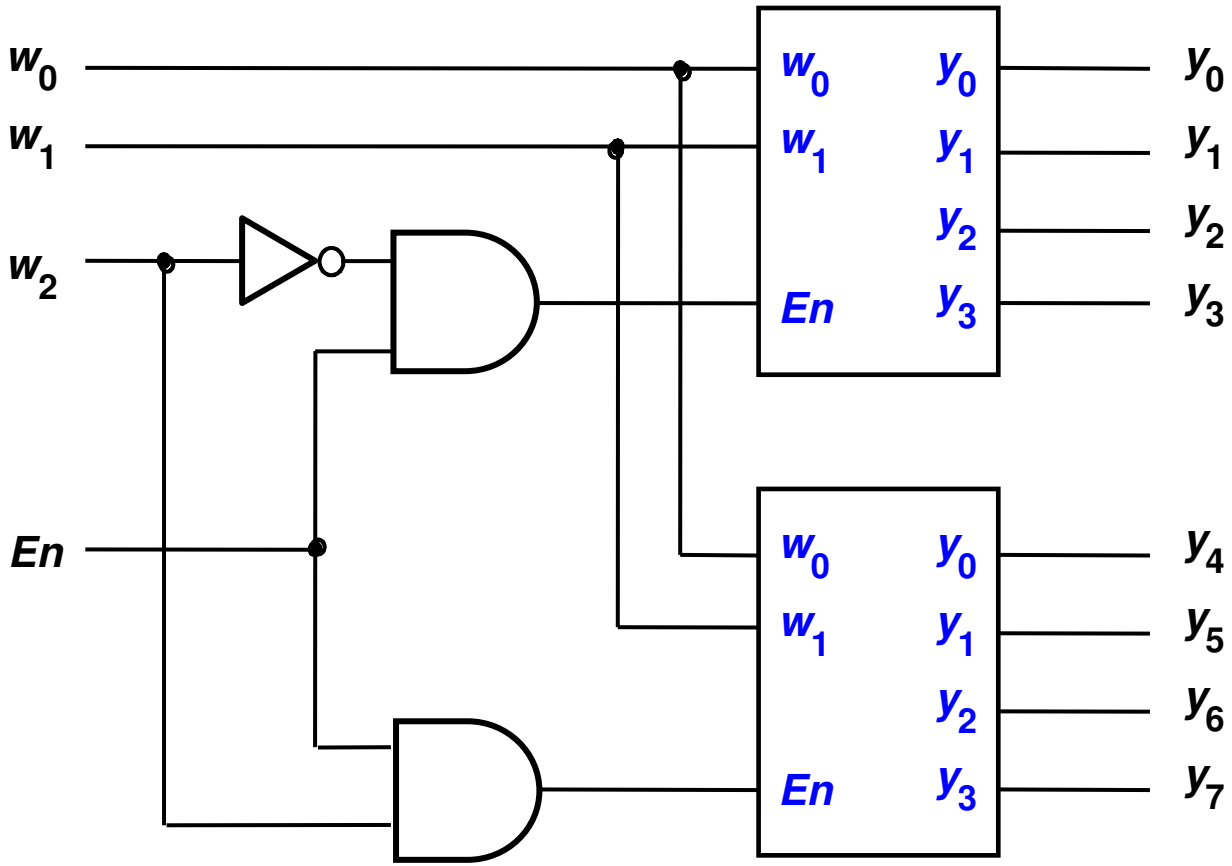
Decodicifador 2:4

En	w_1	w_0	y_0	y_1	y_2	y_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

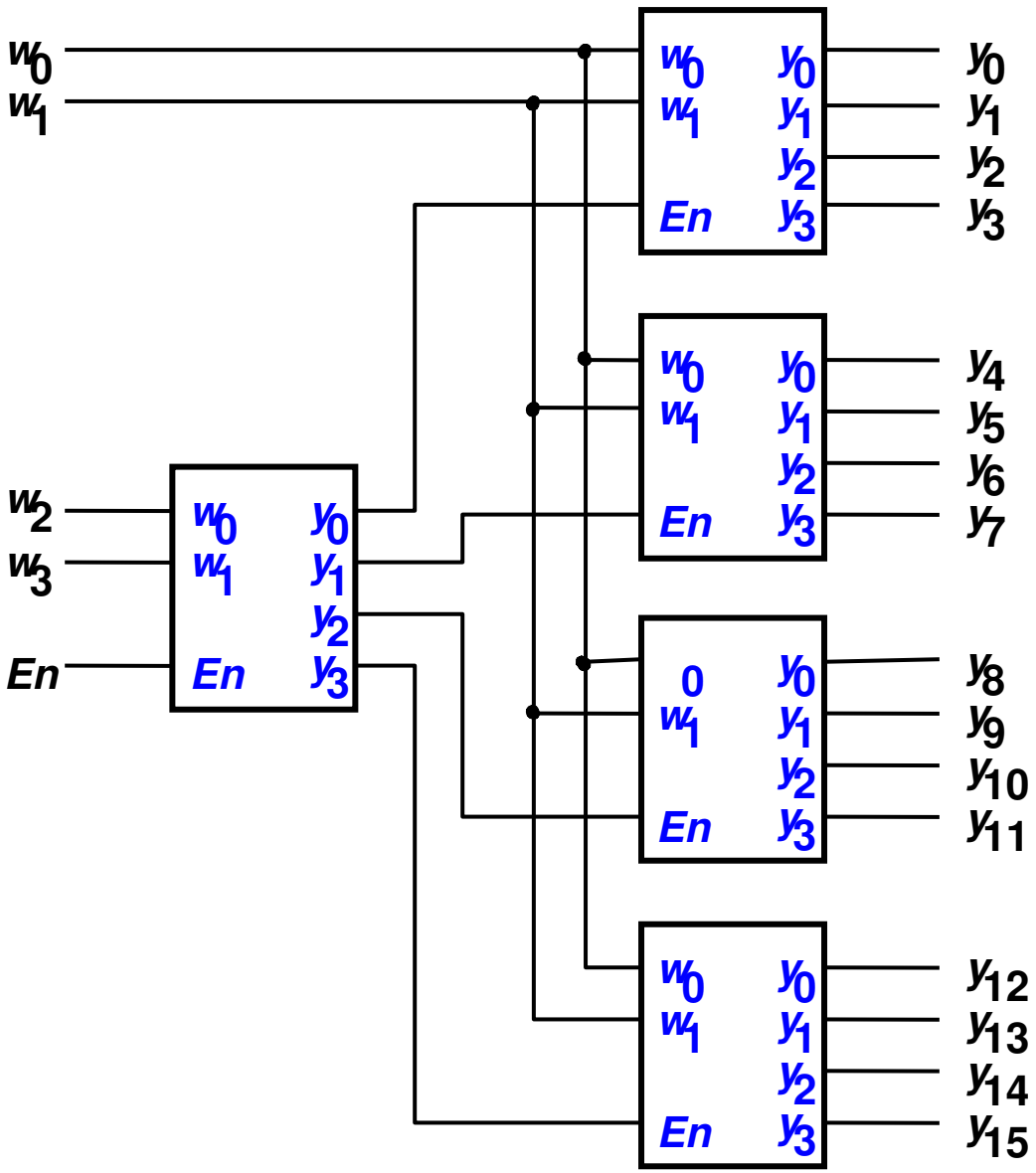




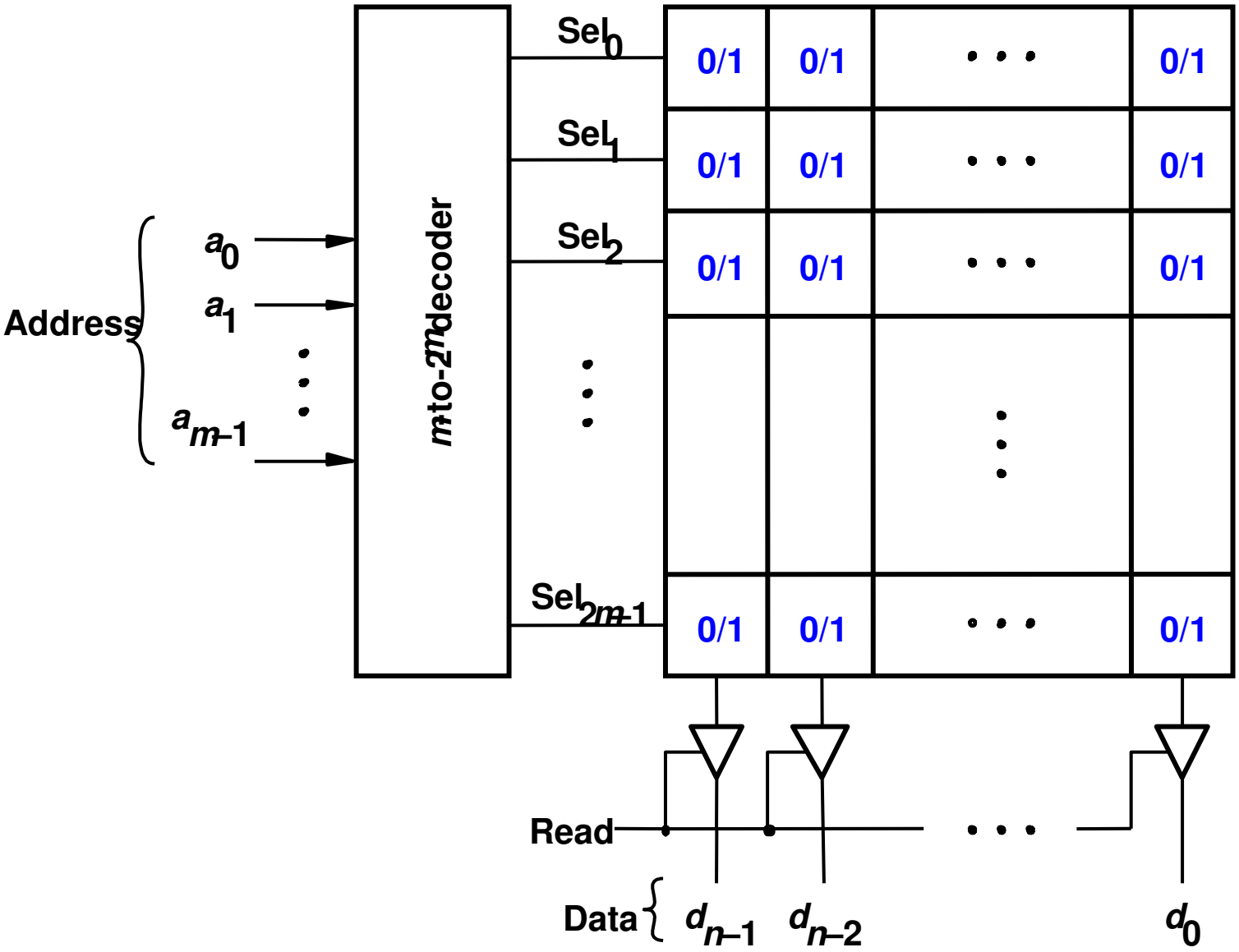
Decodicifador 3:8 Usando 2:4



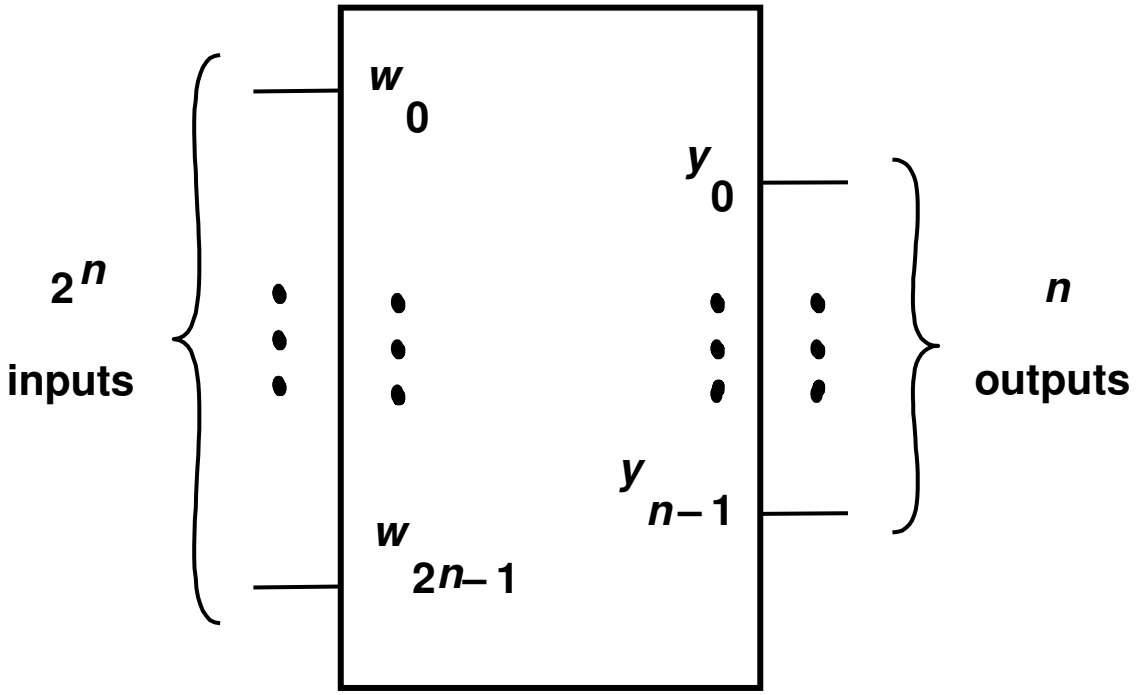
Decodicifador 4:16 Usando 2:4



Memória 2^m x n read-only memory (ROM)



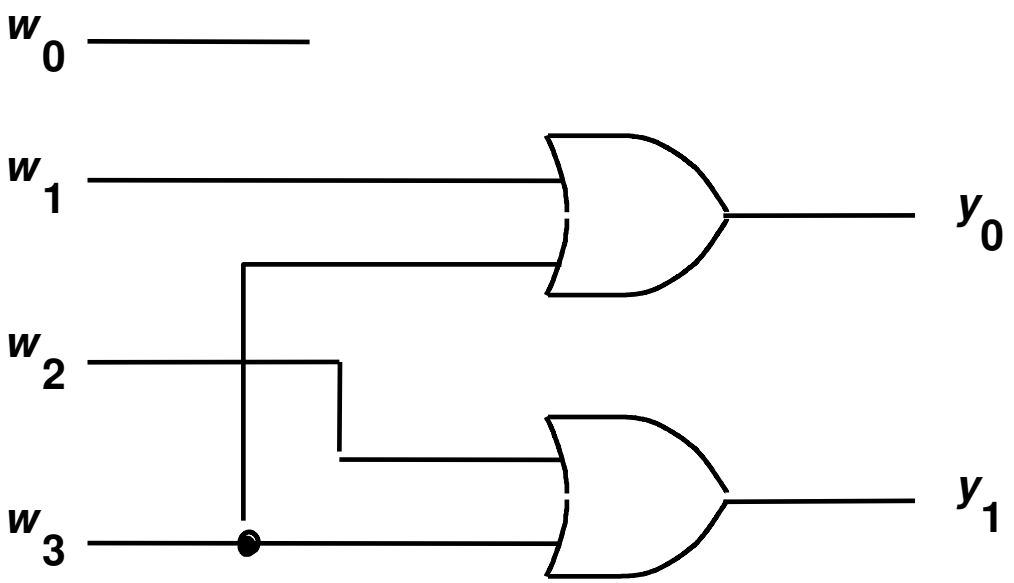
Codificador



2^n-to-n binary encoder

Codificador Binário 4:2

w_3	w_2	w_1	w_0	y_1	y_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

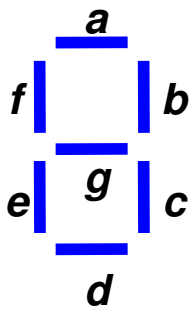


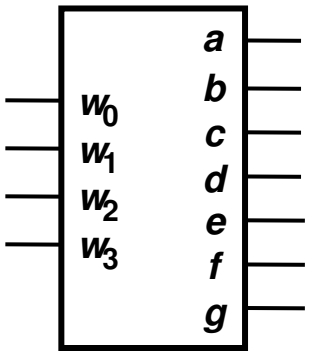
Codificador de Prioridade

w_3	w_2	w_1	w_0	y_1	y_0	z
0	0	0	0	d	d	0
0	0	0	1	0	0	1
0	0	1	x	0	1	1
0	1	x	x	1	0	1
1	x	x	x	1	1	1

Exercício: Qual o Circuito Lógico?

Conversor BCD para Código Display de 7 segmentos

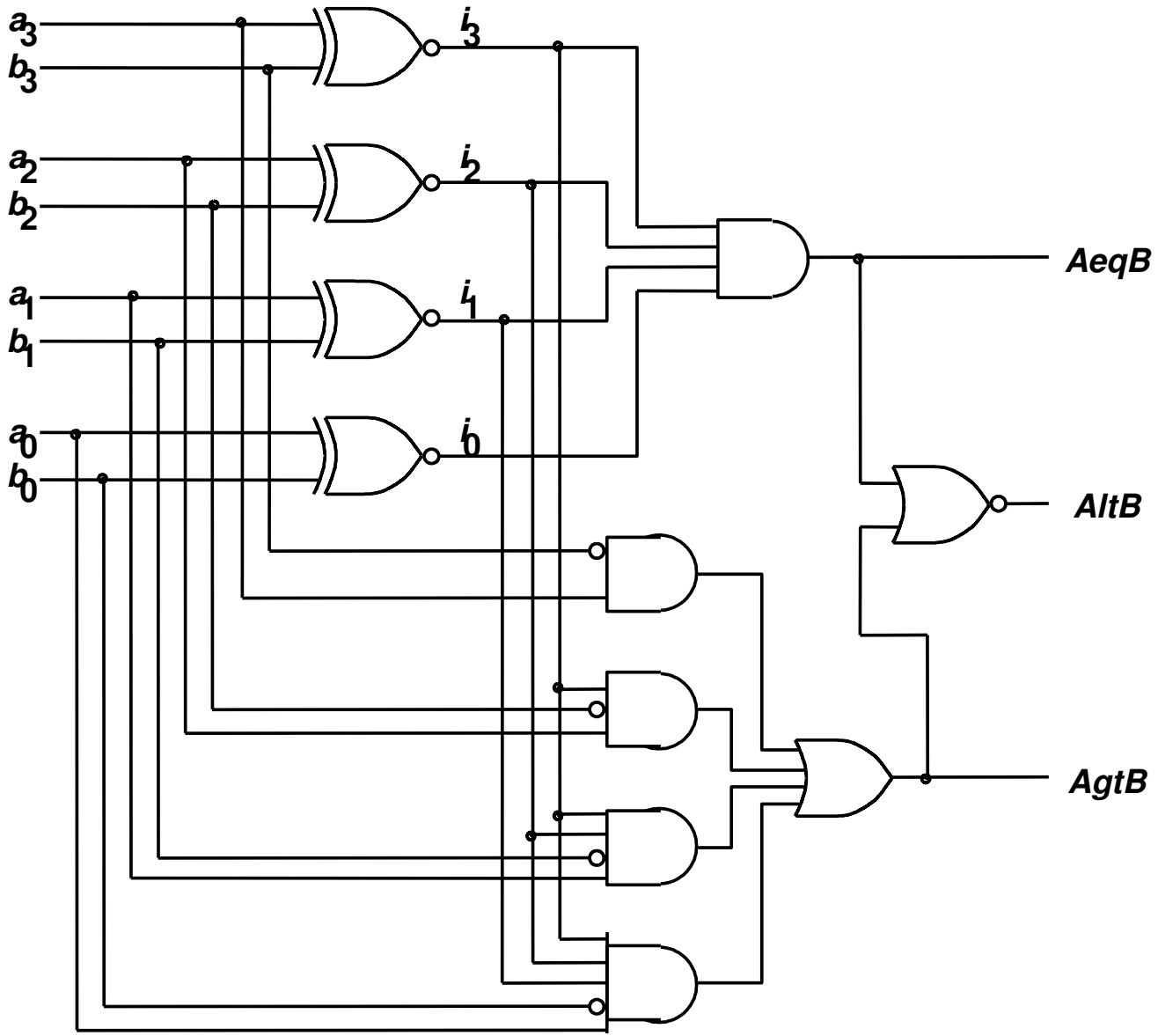




Display 7-segmentos

w_3	w_2	w_1	w_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

Comparador de 4 bits



Circuito Combinacional
VHDL

- **Comandos de Atribuição:**
  - Mux 2:1 - Comando Select

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
        PORT ( w0, w1, s  : IN    STD_LOGIC ;
                    f           : OUT   STD_LOGIC ) ;
END mux2to1 ;

ARCHITECTURE Behavior OF mux2to1 IS
BEGIN
        WITH s SELECT
                    f <=    w0 WHEN '0',
                                w1 WHEN OTHERS ;
END Behavior ;
```

Circuito Combinacional
VHDL

- **Mux 4:1 – Comando Select**

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux4to1 IS
      PORT (  w0, w1, w2, w3     : IN      STD_LOGIC ;
              s                  : IN      STD_LOGIC_VECTOR(1 DOWNTO 0) ;
              f                  : OUT     STD_LOGIC ) ;
END mux4to1 ;

ARCHITECTURE Behavior OF mux4to1 IS
BEGIN
      WITH s SELECT
            f <= w0 WHEN "00",
                 w1 WHEN "01",
                 w2 WHEN "10",
                 w3 WHEN OTHERS ;
END Behavior ;
```

Circuito Combinacional
VHDL

- **Mux 4:1 - Como um Componente**

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;


PACKAGE mux4to1_package IS
    COMPONENT mux4to1
        PORT (  w0, w1, w2, w3    : IN     STD_LOGIC ;
                s                 : IN     STD_LOGIC_VECTOR(1 DOWNTO 0) ;
                f                 : OUT    STD_LOGIC ) ;
    END COMPONENT ;
END mux4to1_package ;
```

- **Mux 16:1 usando o Componente Mux 4:1**

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
LIBRARY work ;
USE work.mux4to1_package.all ;

ENTITY mux16to1 IS
    PORT ( w    : IN     STD_LOGIC_VECTOR(0 TO 15) ;
           s    : IN     STD_LOGIC_VECTOR(3 DOWNTO 0) ;
           f    : OUT    STD_LOGIC ) ;
END mux16to1 ;

ARCHITECTURE Structure OF mux16to1 IS
    SIGNAL m : STD_LOGIC_VECTOR(0 TO 3) ;
BEGIN
    Mux1: mux4to1 PORT MAP ( w(0), w(1), w(2), w(3), s(1 DOWNTO 0), m(0) ) ;
    Mux2: mux4to1 PORT MAP ( w(4), w(5), w(6), w(7), s(1 DOWNTO 0), m(1) ) ;
    Mux3: mux4to1 PORT MAP ( w(8), w(9), w(10), w(11), s(1 DOWNTO 0), m(2) ) ;
    Mux4: mux4to1 PORT MAP ( w(12), w(13), w(14), w(15), s(1 DOWNTO 0), m(3) ) ;
    Mux5: mux4to1 PORT MAP
          ( m(0), m(1), m(2), m(3), s(3 DOWNTO 2), f ) ;
END Structure ;
```

Decodificador Binário

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY dec2to4 IS
      PORT (  w    : IN     STD_LOGIC_VECTOR(1 DOWNTO 0) ;
              En   : IN     STD_LOGIC ;
              y    : OUT    STD_LOGIC_VECTOR(0 TO 3) ) ;
END dec2to4 ;

ARCHITECTURE Behavior OF dec2to4 IS
      SIGNAL Enw : STD_LOGIC_VECTOR(2 DOWNTO 0) ;
BEGIN
      Enw <= En & w ;
      WITH Enw SELECT
            y <= "1000" WHEN "100",
                 "0100" WHEN "101",
                 "0010" WHEN "110",
                 "0001" WHEN "111",
                 "0000" WHEN OTHERS ;
END Behavior ;
```

Circuito Combinacional
VHDL

- **Mux 2:1 Comando Condicional**

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
      PORT (  w0, w1, s      : IN      STD_LOGIC ;
              f              : OUT     STD_LOGIC ) ;
END mux2to1 ;


ARCHITECTURE Behavior OF mux2to1 IS
BEGIN
      f <= w0 WHEN s = '0' ELSE w1 ;
END Behavior ;
```

Circuito Combinacional VHDL

- Processo (mux 2:1)
  - Comando IF-Then-Else

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
    PORT (  w0, w1, s      : IN     STD_LOGIC ;
            f              : OUT    STD_LOGIC ) ;
END mux2to1 ;

ARCHITECTURE Behavior OF mux2to1 IS
BEGIN
    PROCESS ( w0, w1, s )
    BEGIN
        IF s = '0' THEN
            f <= w0 ;
        ELSE
            f <= w1 ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Circuito Combinacional
VHDL

- Processo (mux 2:1 – alternativo)

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
      PORT (  w0, w1, s       : IN      STD_LOGIC ;
              f               : OUT     STD_LOGIC ) ;
END mux2to1 ;

ARCHITECTURE Behavior OF mux2to1 IS
BEGIN
      PROCESS ( w0, w1, s )
      BEGIN
            f <= w0 ;
            IF s = '1' THEN
                  f <= w1 ;
            END IF ;
      END PROCESS ;
END Behavior ;
```

Exercício: Qual o Bloco implementado?

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
ENTITY priority IS
      PORT (   w     : IN      STD_LOGIC_VECTOR(3 DOWNTO 0) ;
               y     : OUT     STD_LOGIC_VECTOR(1 DOWNTO 0) ;
               z     : OUT     STD_LOGIC ) ;
END priority ;

ARCHITECTURE Behavior OF xxxxxxx? IS
      BEGIN
      PROCESS ( w )
      BEGIN
            IF w(3) = '1' THEN
                  y <= "11" ;
            ELSIF w(2) = '1' THEN
                  y <= "10" ;
            ELSIF w(1) = '1' THEN
                  y <= "01" ;
            ELSE
                  y <= "00" ;
            END IF ;
      END PROCESS ;
      z <= '0' WHEN w = "0000" ELSE '1' ;
END Behavior ;
```

Circuito Combinacional VHDL

- Processo (Comando Case)

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
    PORT (  w0, w1, s    : IN     STD_LOGIC ;
            f            : OUT STD_LOGIC ) ;
END mux2to1 ;

ARCHITECTURE Behavior OF mux2to1 IS
BEGIN
    PROCESS ( w0, w1, s )
    BEGIN
        CASE s IS
            WHEN '0' =>
                f <= w0 ;
            WHEN OTHERS =>
                f <= w1 ;
        END CASE ;
    END PROCESS ;
END Behavior ;
```

Decodificador de 7 segmentos

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
ENTITY seg7 IS
      PORT (   bcd     : IN      STD_LOGIC_VECTOR(3 DOWNTO 0) ;
               leds    : OUT     STD_LOGIC_VECTOR(1 TO 7) ) ;
END seg7 ;
ARCHITECTURE Behavior OF seg7 IS
BEGIN
      PROCESS ( bcd )
      BEGIN
            CASE bcd IS            --              abcdefg
                  WHEN "0000"    => leds <=   "1111110" ;
                  WHEN "0001"    => leds <=   "0110000" ;
                  WHEN "0010"    => leds <=   "1101101" ;
                  WHEN "0011"    => leds <=   "1111001" ;
                  WHEN "0100"    => leds <=   "0110011" ;
                  WHEN "0101"    => leds <=   "1011011" ;
                  WHEN "0110"    => leds <=   "1011111" ;
                  WHEN "0111"    => leds <=   "1110000" ;
                  WHEN "1000"    => leds <=    "1111111" ;
                  WHEN "1001"    => leds <=    "1110011" ;
                  WHEN OTHERS    => leds <=    "-------" ;
            END CASE ;
      END PROCESS ;
END Behavior ;
```

74381 ALU

Operation	Inputs s_2 s_1 s_0	Outputs F
Clear	0 0 0	0 0 0 0
B−A	0 0 1	$B - A$
A−B	0 1 0	$A - B$
ADD	0 1 1	$A + B$
XOR	1 0 0	A XOR B
OR	1 0 1	A OR B
AND	1 1 0	A AND B
Preset	1 1 1	1 1 1 1

74381 ALU

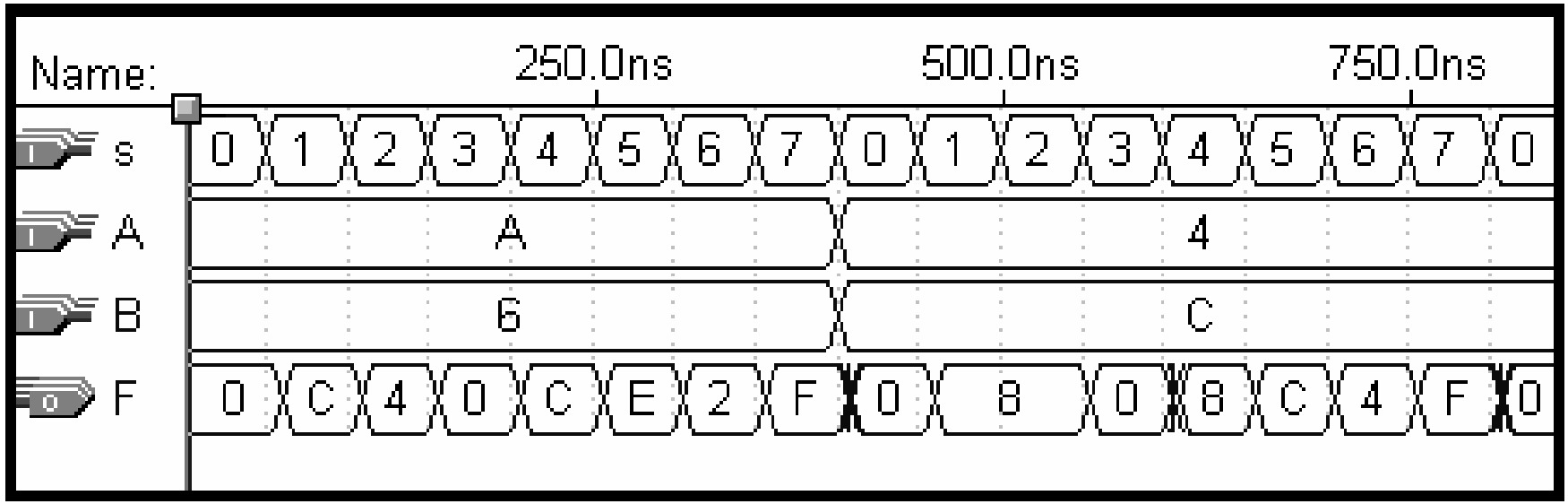
```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
USE ieee.std_logic_unsigned.all ;


ENTITY alu IS
        PORT (  s       : IN    STD_LOGIC_VECTOR(2 DOWNTO 0) ;
                A, B    : IN    STD_LOGIC_VECTOR(3 DOWNTO 0) ;
                F       : OUT   STD_LOGIC_VECTOR(3 DOWNTO 0) ) ;
END alu ;
```

74381 ALU

```
ARCHITECTURE Behavior OF alu IS
BEGIN
        PROCESS ( s, A, B )
        BEGIN
                CASE s IS
                        WHEN "000" =>
                                F <= "0000" ;
                        WHEN "001" =>
                                F <= B - A ;
                        WHEN "010" =>
                                F <= A - B ;
                        WHEN "011" =>
                                F <= A + B ;
                        WHEN "100" =>
                                F <= A XOR B ;
                        WHEN "101" =>
                                F <= A OR B ;
                        WHEN "110" =>
                                F <= A AND B ;
                        WHEN OTHERS =>
                                F <= "1111" ;
                END CASE ;
        END PROCESS ;
END Behavior ;
```

74381 ALU


Name:
250.0ns
500.0ns
750.0ns
s
A
B
F
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0
A
4
6
C
0 C 4 0 C E 2 F 0 8 0 8 C 4 F 0