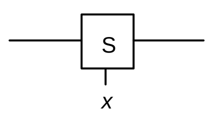
Circuitos Lógicos e Organização de Computadores

Capítulo 2 – Introdução aos Circuitos Lógicos

Ricardo Pannain
pannain@puc-campinas.edu.br
http://docentes.puc-campinas.edu.br/ceatec/pannain/

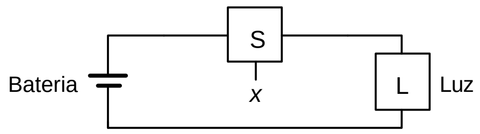
VARIÁVEIS E FUNÇÕES – Chaves de dois estados

(a) Chave binária - dois estados

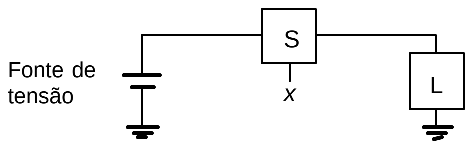


(b) Símbolo de uma chave

VARIÁVEIS E FUNÇÕES – Uma luz controlada por uma chave



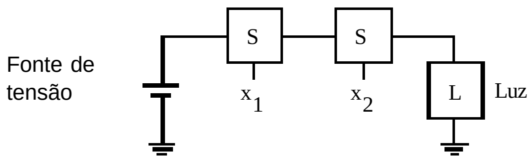
(a) Conexão de uma chave e uma luz a uma bateria



L(x) = x (função lógica onde x = variável de entrada)

(b) Usando a conexão com o terra

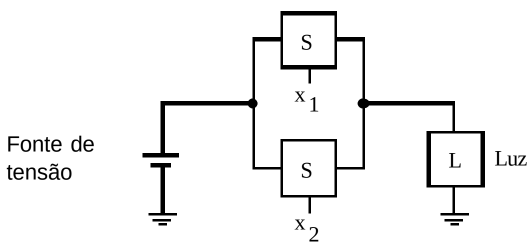
VARIÁVEIS E FUNÇÕES – Funções Básicas



L = x1 . x2

L = 1 se x1 = 1 e x2 = 1; caso contrário L = 0

(a) Função Lógica **AND** (ligação em série)

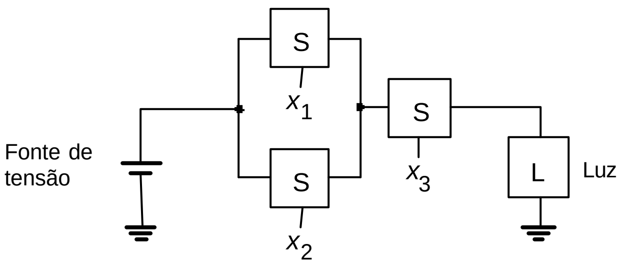


L = x1 + x2

L = 0 se x1 = 0 e x2 = 0; caso contrário L = 1

(b) Função Lógica **OR** (ligação paralela)

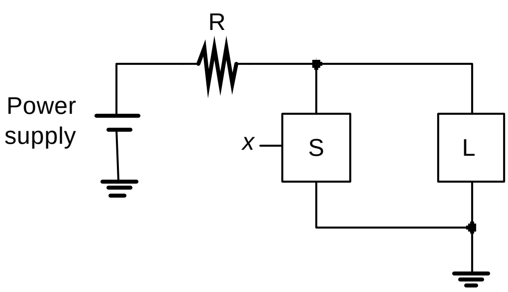
Ligação Série-Paralela



Exercício

Escreva a função lógica de dê os valores de L para todas as combinações possíveis de x1, x2 e x3

Função de inversão - NOT



$L(x) = \bar{x}$; L = 1 se x = 0 e L = 0 se x = 1

Tabela Verdade – Funções AND e OR de duas variáveis

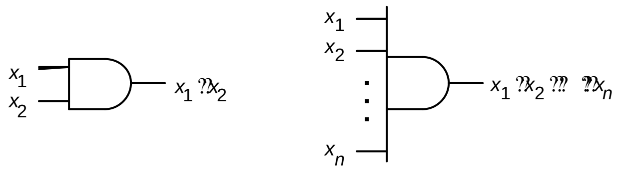
x_1	x_2	$x_1 \cdot x_2$	$x_1 + x_2$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1
		AND	OR

Tabela Verdade – Funções AND e OR de tres variáveis

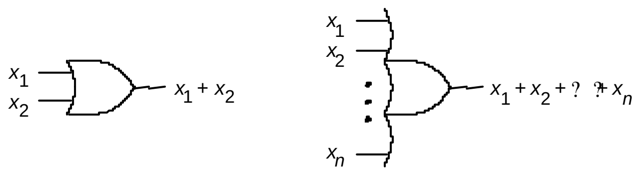
x_1	x_2	x_3	$x_1 \cdot x_2 \cdot x_3$	$x_1 + x_2 + x_3$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

P
O
R
T
A
S

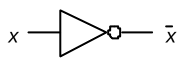
L
Ó
G
I
C
A
S



(a) PORTAS AND

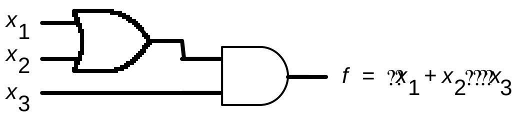


(b) PORTAS OR



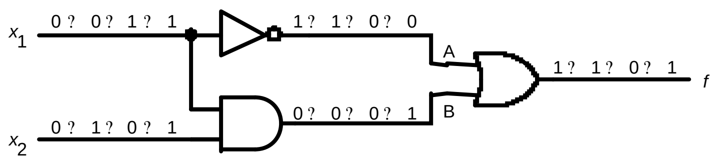
(c) PORTA NOT

Função OR-AND



Exercício – Dê a tabela verdade desta função.

Rede Lógica – Circuito Lógico



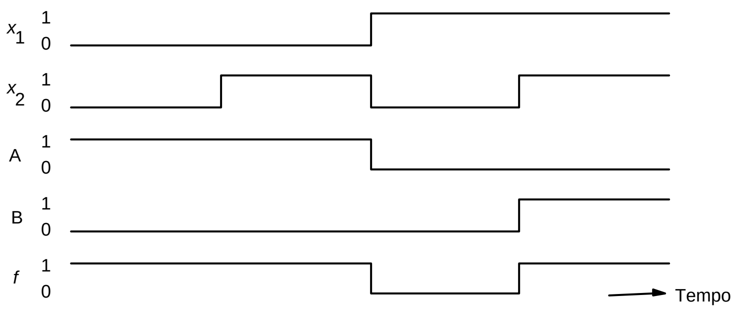
(a) Circuito que implementa $f = \bar{x}_1 + x_1 ? x_2$

x_1	x_2	$f(x_1, x_2)$
0	0	1
0	1	1
1	0	0
1	1	1

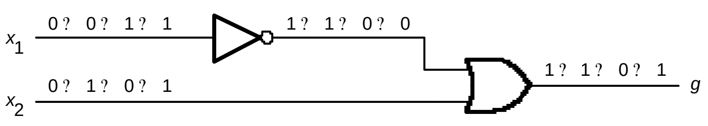
(b) Tabela verdade para f

C
I
R
C
U
I
T
O

L
Ó
G
I
C
O



(c) Diagrama de tempo



(d) Circuito que implementa $g = \bar{x}_1 + x_2$

Exercícios 1 - Qual a diferença dos circuitos da letra (a) e (d) ?
2 - O que são circuitos equivalentes ?

Álgebra Booleana

George Boole (1849) – teoria algébrica aplicada à lógica

Claude Shannon (1930) – mostrou que esta teoria poderia ser aplicada em circuitos baseados em chaves

Álgebra Booleana

Axiomas

1a. $0 . 0 = 0$

1b $1 + 1 = 1$

2a $1 . 1 = 1$

2b $0 + 0 = 0$

3a. $0 . 1 = 1 . 0 = 0$

3b $1 + 0 = 0 + 1 = 1$

4a Se $x = 0$ então $\overline{x} = 1$

4 b Se $x = 1$ então $\overline{x} = 0$

Teoremas de uma variável

5a. $x . 0 = 0$

5b $x + 1 = 1$

6a $x . 1 = x$

6b $x + 0 = x$

7a. $x . x = x$

7b $x + x = x$

8a $x . \overline{x} = 0$

8b $x + \overline{x} = 1$

9 $\overline{\overline{x}} = x$

Álgebra Booleana

Teoremas de duas e três variáveis

10a. $x \cdot y = y \cdot x$ **Comutativa**

10b $x + y = y + x$

11a $x \cdot (y \cdot z) = (x \cdot y) \cdot z$ **Associativa**

11b $x + (y + z) = (x + y) + z$

12a. $x \cdot (y + z) = x \cdot y + x \cdot z$ **Distributiva**

12b $x + (y \cdot z) = (x + y) \cdot (x + z)$

13a $x + x \cdot y = x$ **Absorção**

13b $x \cdot (x + y) = x$

14a $x \cdot y + x \cdot \overline{y} = x$ **Combinação**

14b $(x + y) \cdot (x + \overline{y}) = x$

15a $\overline{x \cdot y} = \overline{x} + \overline{y}$ **Teorema de DeMorgan**

15b $\overline{x + y} = \overline{x} \cdot \overline{y}$

16a $x + \overline{x} \cdot y = x + y$

16b $x \cdot (\overline{x} + y) = x \cdot y$

Álgebra Booleana

Princípio da Dualidade: Dada uma expressão lógica seu dual é obtido substituindo todos os operadores + pelo operador **.** , e vice versa, e substituindo todos os 0s por 1s, e vice versa

Prova do Teorema DeMorgan

x	y	$x \cdot y$	$\overline{x \cdot y}$	$\overline{x}$	$\overline{y}$	$\overline{x} + \overline{y}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

LHS RHS

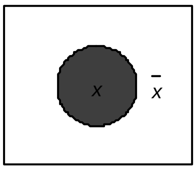
LHS = lefthand side **RHS = righthand side**

Exercícios – exemplos 2.1 e 2.2

Representação - Diagrama de Venn

(a) Constant 1

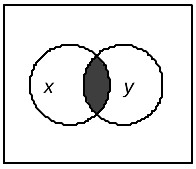
(b) Constant 0



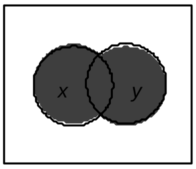
(c) Variable *x*

(d) $\bar{x}$

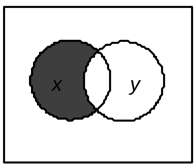
Representação - Diagrama de Venn



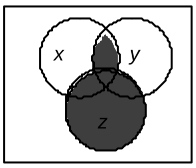
(e) $x \cdot y$



(f) $x + y$

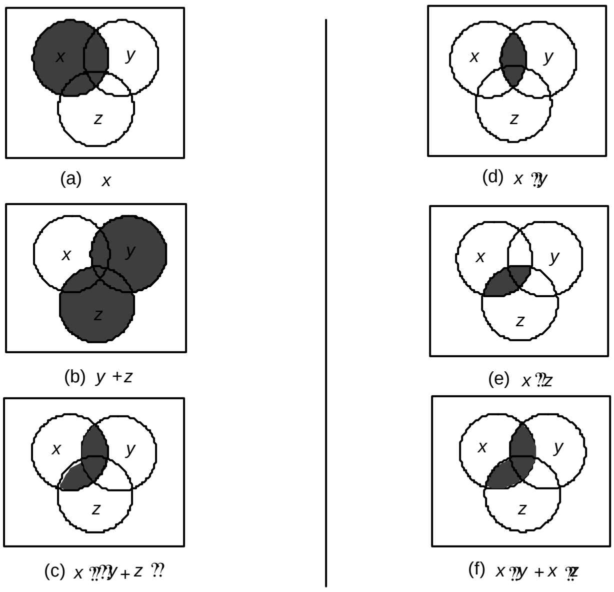


(g) $x \cdot \bar{y}$



(h) $x \cdot y + z$

Representação de Venn – Propriedade Distributiva



Álgebra Booleana

- Notações e Terminologias
  - Soma de produtos (SOP)

    $x1 . \underline{x2} . x3 + \underline{x1} . x4 + x2 . x3 . \underline{x4}$

  - Produto de somas (POS)

    $(\underline{x1} + x2) . (x1 + \underline{x3}) . (\underline{x2} + x3 + x4)$

OBS - O termo produto da Soma de produtos é chamado de mintermo e o termo soma do Produto de somas é chamado de maxtermo

- Precedência de Operações
  - NOT – AND – OR

OBS - Obedecer os parênteses

Síntese de funções utilizando AND, OR e NOT

Projetar um circuito com duas entradas (x1 e x2), assumindo que x1 e x2 representam o estado de duas chaves, respectivamente. (chave aberta xi = 0 e chave fechada xi = 1).

A saída f(x1,x2) será 1 quando (x1,x2) forem: (0,0) , (0,1) ou (1,1). Será 0 quando (x1,x2) = (1,0)

x_1	x_2	$f(x_1, x_2)$
0	0	1
0	1	1
1	0	0
1	1	1

Tabela Verdade

Síntese – Soma de Produtos (Mintermos)

Para uma função de n variáveis, o termo produto (mintermo) é formado por xi se $\overline{xi}$ = 1 e xi se xi = 0.

m0 = $\overline{x}1\overline{x}2$; m1= $\overline{x}1x2$; m2 = $x1\overline{x}2$; m3 = x1x2

Para a figura anterior:

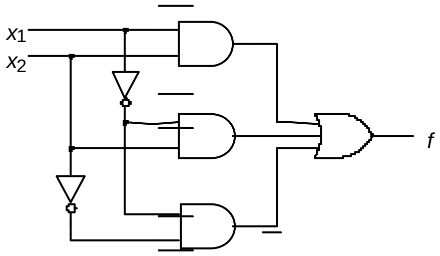
f = m0 . 1 + m1 . 1 + m2 . 0 + m3 . 1 = m0 + m1 + m3 =

= $\overline{x}1\overline{x}2$ + $\overline{x}1x2$ + x1x2

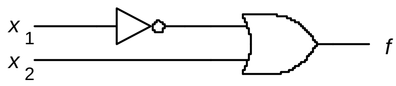
Outra forma de representação:

f(x1,x2) = □(m0, m1, m3) ou f(x1,x2) = □m(0,1,3)

Representação com e portas lógicas



(a) Circuito que representa a Soma de Produtos



(b) Circuito de custo mínimo – mesmo circuito

Custo = n. de gates do circuito + n. de entradas de todos os gates do circuito

Síntese – Produto de Somas (Maxtermos)

Uma função pode ser também representada como uma soma de mintermos onde $\overline{f}$ = 1, ou seja f = 0. No exemplo anterior:

$\overline{f(x1,x2)} = m2 = x1\overline{x2}$

$\overline{\overline{f(x1,x2)}} = f(x1,x2) = \overline{x1\overline{x2}} = \overline{x1} + x2 = \overline{m2} = M2$

Mi = maxtermo

Outras formas de representação do Produto de somas:

F(x1,x2) = □(M2) ou f(x1,x2) = □M(2)

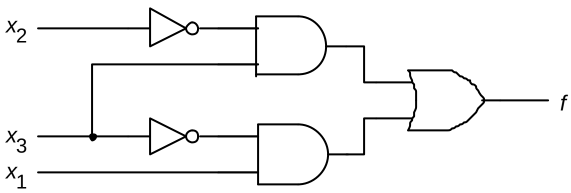
Mintermos and Maxtermos de função de três variáveis

Row number	x_1	x_2	x_3	Minterm	Maxterm
0	0	0	0	$m_0 = \overline{x}_1\overline{x}_2\overline{x}_3$	$M_0 = x_1 + x_2 + x_3$
1	0	0	1	$m_1 = \overline{x}_1\overline{x}_2 x_3$	$M_1 = x_1 + x_2 + \overline{x}_3$
2	0	1	0	$m_2 = \overline{x}_1 x_2\overline{x}_3$	$M_2 = x_1 + \overline{x}_2 + x_3$
3	0	1	1	$m_3 = \overline{x}_1 x_2 x_3$	$M_3 = x_1 + \overline{x}_2 + \overline{x}_3$
4	1	0	0	$m_4 = x_1\overline{x}_2\overline{x}_3$	$M_4 = \overline{x}_1 + x_2 + x_3$
5	1	0	1	$m_5 = x_1\overline{x}_2 x_3$	$M_5 = \overline{x}_1 + x_2 + \overline{x}_3$
6	1	1	0	$m_6 = x_1 x_2\overline{x}_3$	$M_6 = \overline{x}_1 + \overline{x}_2 + x_3$
7	1	1	1	$m_7 = x_1 x_2 x_3$	$M_7 = \overline{x}_1 + \overline{x}_2 + \overline{x}_3$

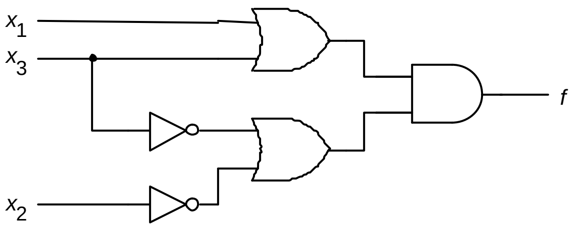
Exemplo – Escrever a função descrita pela tabela verdade abaixo, como Soma de Produtos e Produto de soma

Row number	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Duas representações da função do exemplo anterior



(a) Circuito mínimo referente à SOP



(b) Circuito mínimo referente ao POS

EXEMPLOS

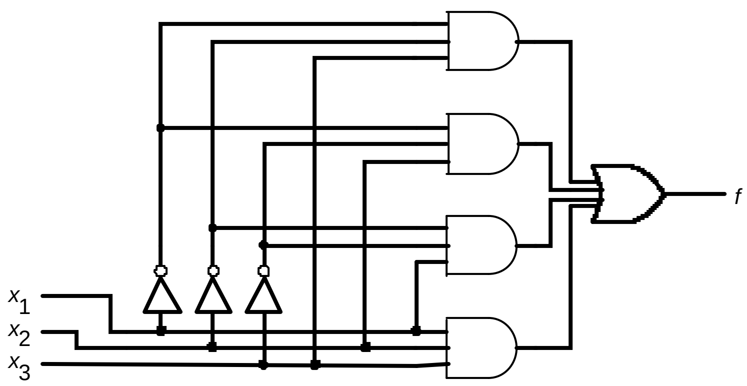
1. Controle de luz por 3 chaves
   - Assumir uma sala com 3 portas e perto de cada porta um interruptor de luz. Queremos projetar um circuito que controle a iluminação da sala de tal maneira que possamos acender ou apagar a lâmpada pela mudança de qualquer chave.

2. Multiplexador
   - È um circuito que permite com que seja escolhida, dentre várias entradas, apenas uma

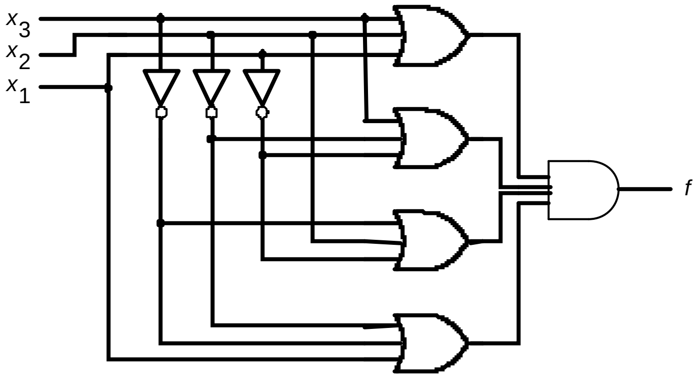
Tabela verdade - Controle de luz por 3 chaves

x_1	x_2	x_3	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Circuito de Controle de luz por 3 chaves - SOP



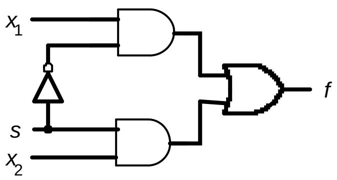
Circuito de Controle de luz por 3 chaves - POS



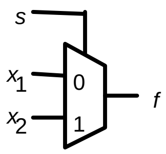
MULTIPLEXADOR

$s\ x_1\ x_2$	$f(s, x_1, x_2)$
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	0
1 1 1	1

(a)Tabela Verdade



(b) Circuito

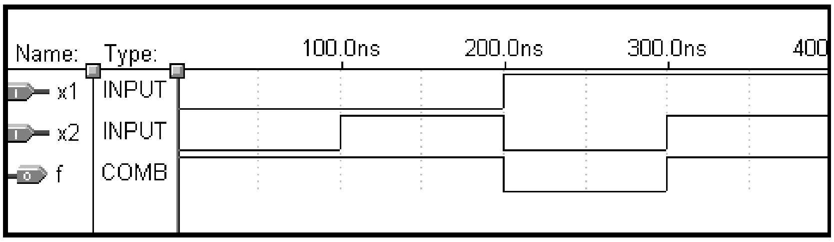


(c) Símbolo Gráfico

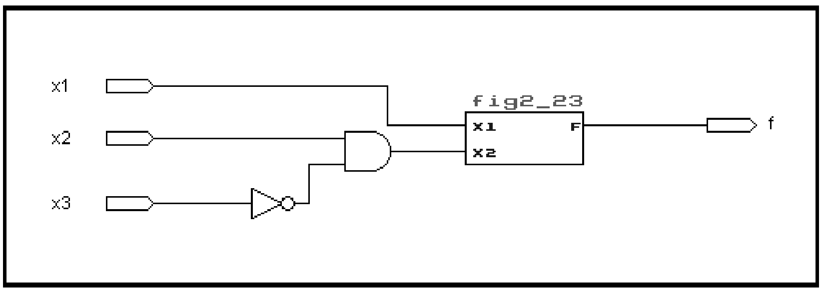
s	$f(s, x_1, x_2)$
0	x_1
1	x_2

(d) Representação compacta da tabela verdade

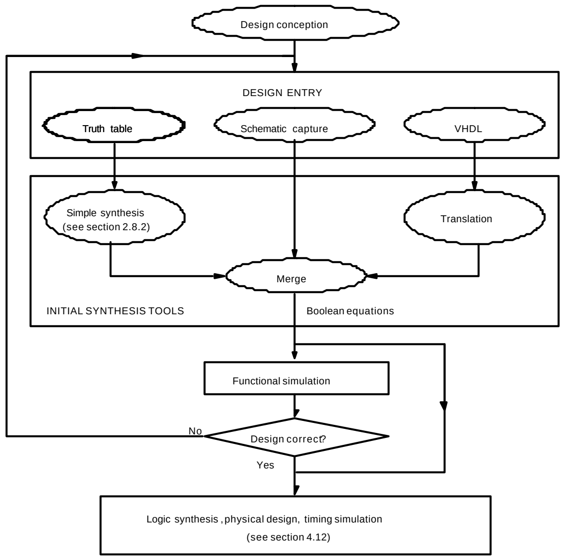
Ferramentas de Auxílio ao Projeto de Circutos –
Editor de Forma de Ondas



Ferramentas de Auxílio ao Projeto de Circutos –
Captura Esquemática



Sistema de CAD –Primeiros Estágios



VHDL - Very High Speed Integrated Circuit Hardware Description Language - Introdução

- Linguagem para descrição de hardware
  - Representação dos sinais digitais ✍ BIT (0 ou 1)
  - Código VHDL
    - Declaração dos sinais de entrada e saída do circuito ✍ ENTITY / PORT /IN / OUT
    - Exemplo

```
entity nome is
    port (entrada1:in;
     saida1:out);
end nome;
```

VHDL - Very High Speed Integrated Circuit Hardware Description Language - Introdução

- Código VHDL - continuação
  - Descrição da funcionabilidade ✍ ARCHITECTURE
  - Funções ✍ AND, OR, NOT, NAND, NOR, XOR, XNOR
  - Exemplo

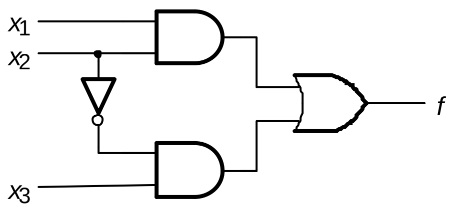
architecture *nome_da_entidade* **is**

begin

    f <= entrada1 **AND** entrada2 **;**

end *nome_da_entidade;*

Código VHDL de uma função simples




```
ENTITY example1 IS
    PORT ( x1, x2, x3   : IN    BIT ;
           f            : OUT   BIT ) ;
END example1 ;

ARCHITECTURE LogicFunc OF example1 IS
BEGIN
    f <= (x1 AND x2) OR (NOT x2 AND x3) ;
END LogicFunc ;
```

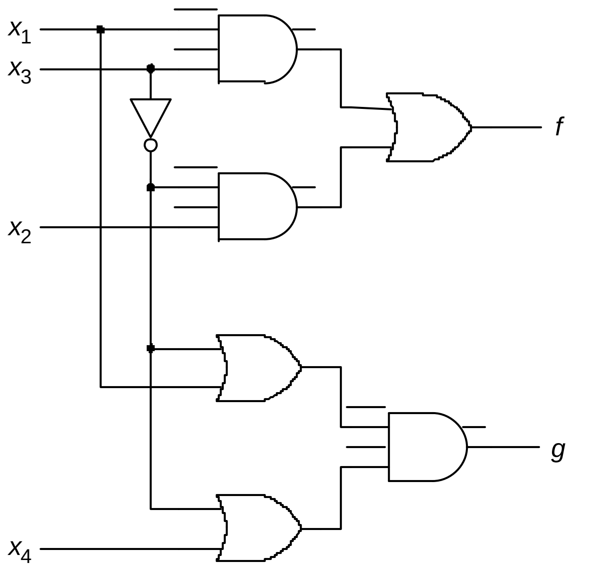
Código VHDL de uma função de quatro entradas

```
ENTITY example2 IS
      PORT ( x1, x2, x3, x4   : IN    BIT ;
                  f, g                : OUT  BIT ) ;
END example2 ;

ARCHITECTURE LogicFunc OF example2 IS
BEGIN
      f <=  (x1 AND x3) OR (NOT x3 AND x2) ;
      g <= (NOT x3 OR x1) AND (NOT x3 OR x4) ;
END LogicFunc ;
```

Capítulo 2 - Introdução aos Circuitos Lógicos 39

Circuito de uma função de quatro entradas



Capítulo 2 - Introdução aos Circuitos Lógicos 40

Diagrama de tempo que representa uma função lógica

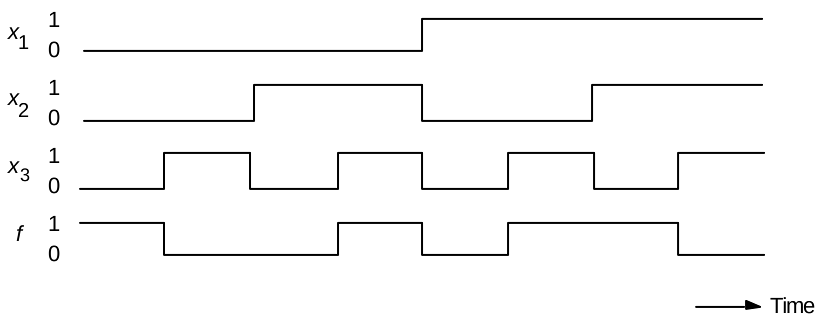


Diagrama de tempo que representa uma função lógica