Sistemas Numéricos

Prof. Daniel Barros Júnior, Prof. Eduardo Augusto Bezerra

16 de agosto de 2004

Página: www.ee.pucrs.br/∼dbarros
E-mail: dbarros@ee.pucrs.br
Versão: 1.15

1 Introdução

1.1 Histórico

O conceito de um computador digital pode ser traçado retornando para Charles Babbage, que desenvolveu, em torno de 1830, um dispositivo computacional mecânico. O primeiro computador digital, eletromecânico, foi construído em 1944 na Universidade de Harvard.

A eletrônica digital moderna iniciou em 1946 com um computador chamado ENIAC.

A tecnologia digital evoluiu da válvula para transistores e posteriormente para circuitos integrados complexos, alguns dos quais contém milhões de transistores.

1.2 Grandezas Digitais e Analógicas

Circuitos eletrônicos podem ser divididos em duas categorias principais: Digital e Analógica.

Eletrônica Analógica envolve valores contínuos.

Figura 1: Valores contínuos

Eletrônica Digital envolve valores discretos.

Figura 2: Valores Discretos

Vantagens da representação digital:

Dados digitais podem ser processados e transmitidos mais eficientemente e confiavelmente do que dados analógicos;

Dados digitais tem uma grande vantagem quando o armazenamento é necessário.

Um sistema que usa tanto métodos analógicos como digitais é o compact disk (CD).

Figura 3: CD player

Levando em conta estas e outras vantagens dos sistemas digitais sirgiu a necessidade de representar as informações de forma numérica.

Um sistema na base B, possui um conjunto de B símbolos, também chamado alfabeto.

Na Tabela 1 na página 2 estão representadas as bases utilizadas freqüentemente.

Base 2:	0	1														
Base 8:	0	1	2	3	4	5	6	7								
Base 10:	0	1	2	3	4	5	6	7	8	9						
Base 16:	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

Tabela 1: Base numéricas

2 Sistema Binário

O sistema binário é o mais elementar pois possui apenas dois símbolos.

Na seqüência binária, cada digito é chamado de BIT (Binary Digit).

Na Figura 4 tem-se um número binário com seu BIT mais significativo (MSB) e o bit menos significativo (LSB) sendo enfatizados.

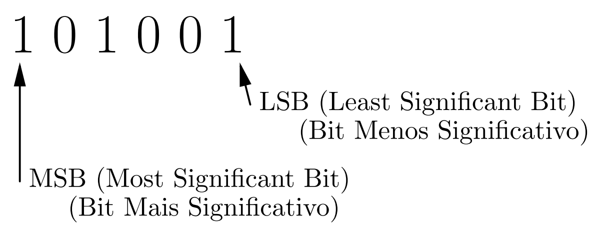


Figura 4: MSB e LSB

Visando facilitar a leitura, os bits são agrupados conforme mostra a Tabela 2, estes grupos recebem nomes específicos. A principal finalidade de agrupar os bits está em facilitar o controle dos digitos.

4 bits	Nibble
8 bits	Byte
16 bits	Word

Tabela 2: Agrupamentos de Dados

3 Tabela dos Números Inteiros

A Tabela 3 mostra os números decimais de 0 até 16 e seus respectivos valores em binário, octal e haxadecimal.

Decimal (10)	Binário (2)	Octal (8)	Hexadecimal (16)
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

Tabela 3: Conversões

Conforme pode ser observado na Tabela 3 para cada digito de um número binário dobram-se as linhas da tabela, sendo o número de possibilidades dado por 2^n, onde n é o número de digitos do número binário.

4 Conversão de um número decimal inteiro para um base B

Exemplo 1: Converter 2468_{10} para a base 16 (hexa).

Solução:

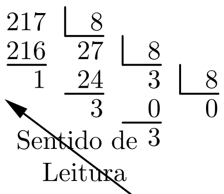
$$
\begin{array}{rrrr}
2468 & \underline{|\,16} & & \\
\underline{2464} & 154 & \underline{|\,16} & \\
4 & \underline{144} & 9 & \underline{|\,16} \\
 & 10 & \underline{0} & 0 \\
 & & 9 &
\end{array}
$$

Sentido de Leitura

$\boxed{2468_{10} = 9A4_{16}}$

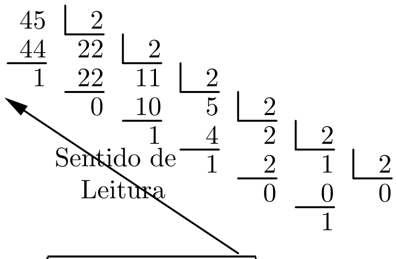
Exemplo 2: Converter 217_{10} para a base 8 (octal).

Solução:



$\boxed{217_{10} = 331_8}$

Exemplo 3: Converter 45_{10} para a base 2 (binário).
Solução:



$\boxed{45_{10} = 101101_2}$

Base B para decimal.
$\boxed{101101_2 = 1.2^5 + 0.2^4 + 1.2^3 + 1.2^2 + 0.2^1 + 1.2^0 = 45_{10}}$

5 Conversão de um número octal ou hexadecimal para a base binária

Exemplo 1: 2357_8 para binário.
Solução:
$2_8 \longrightarrow 010_2$
$3_8 \longrightarrow 011_2$
$5_8 \longrightarrow 101_2$
$7_8 \longrightarrow 111_2$

$2357_8 = 010 \quad 011 \quad 101 \quad 111_2$
$2357_8 = 010011101111_2$

Exemplo 2: $4A05_{16}$ para binário.
Solução:
$4_{16} \longrightarrow 0100_2$
$A_{16} \longrightarrow 1010_2$
$0_{16} \longrightarrow 0000_2$
$5_{16} \longrightarrow 0101_2$

$4A05_{16} = 0100 \quad 1010 \quad 0000 \quad 0101_2$
$4A05_{16} = 0100101000000101_2$

6 Conversão de um número octal em hexa e hexa em octal

Exemplo 1: 127_8 para hexadecimal.
Solução:

$127_8 = 001 \quad 010 \quad 111_2 =$
$127_8 = 0 \quad 0101 \quad 0111_2 = 57_{16}$
$127_8 = 001010111_2 = 57_{16}$

Exemplo 2: 32_8 para hexadecimal.
Solução:

$32_8 = 011 \quad 010_2 =$
$32_8 = 01 \quad 1010_2 = 1A_{16}$
$32_8 = 011010_2 = 1A_{16}$

Exemplo 3: $C3_{16}$ para octal.
Solução:

$C3_{16} = 1100 \quad 0011_2 =$
$C3_{16} = 11 \quad 000 \quad 011_2 = 303_8$
$C3_{16} = 11000011_2 = 303_8$

Exemplo 4: 23_{16} para octal.
Solução:

$23_{16} = 0010 \quad 0011_2 =$
$23_{16} = 00 \quad 100 \quad 011_2 = 43_8$
$23_{16} = 00100011_2 = 43_8$

7 Conversão de frações decimais para base B

Regra da multiplicação refletida:

1. Multiplicar o número decimal pela base B;
2. A parte inteira do resultado é utilizada como digito da base B;
3. A parte inteira é descartada;
4. Retornar ao passo 1 caso a parte fracionária seja diferente de 0 (zero).

Exemplo 1: $0,375_{10}$ para binário.
Solução:
$0,375 \quad \times \quad 2 = 0,75 \longrightarrow 0$
$0,75 \quad \times \quad 2 = 1,50 \longrightarrow 1$
Descarta parte inteira
$0,50 \quad \times \quad 2 = 1,00 \longrightarrow 1$
Termina o processo quando a parte fracionária chega até 0 (zero).
$\boxed{0,375_{10} = 0,011_2}$

Obs.:
$\boxed{0,011_2 = 0.2^0 + 0.2^{-1} + 1.2^{-2} + 1.2^{-3} = 0,25 + 0,125 = 0,375_{10}}$

Exemplo 2: $0,2_{10}$ para binário.
Solução:
$0,2 \quad \times \quad 2 = 0,4 \longrightarrow 0$
$0,4 \quad \times \quad 2 = 0,8 \longrightarrow 0$
$0,8 \quad \times \quad 2 = 1,6 \longrightarrow 1$
Descarta parte inteira
$0,6 \quad \times \quad 2 = 1,2 \longrightarrow 1$

Descarta parte inteira
$0,2 \quad \times \quad 2 = 0,4 \longrightarrow 0$
$\boxed{0,2_{10} = 0,00110011\ldots_2}$

Exemplo 3: $3,25_{10}$ para binário.
Solução:
Converter a parte inteira separadamente da parte fracionária.
$0,25 \quad \times \quad 2 = 0,5 \longrightarrow 0$
$0,5 \quad \times \quad 2 = 1,0 \longrightarrow 1$
$\boxed{3,25_{10} = 11,01_2}$

Obs.:
$\boxed{11,01_2 = 1.2^1 + 1.2^0 + 0.2^{-1} + 1.2^{-2} = 2 + 1 + 0,25 = 3,25_{10}}$

8 Operações Aritméticas no Sistema Binário

8.1 Adição

A Tabela 4 apresenta o resultado da soma de dois números de um bit para todas as situações possíveis. Pode-se verificar que na última linha o resultado da soma não pode mais ser armazenado em apenas um bit, gerando um segundo bit o qual é chamado de Transporte (Carry).

$0+0=$	0
$0+1=$	1
$1+0=$	1
$1+1=$	10

Tabela 4: Adição binária

Na operação de soma tem-se $1+1=0$ e vai 1 formando assim o 10 apresentado na Tabela 4.

Ambos os números positivos:
Exemplo 1: $10101_2 + 10111_2 =$
Solução:

$$\boxed{\begin{array}{rr} & 10101_2 \\ + & 10111_2 \\ \hline & 101100_2 \end{array}}$$

Exemplo 2: $111,101_2 + 11,001_2 =$
Solução:

$$\boxed{\begin{array}{rr} & 111,101_2 \\ + & 11,001_2 \\ \hline & 1010,110_2 \end{array}}$$

Exemplo 3: Neste exemplo será feita a operação em octal.
$325,71_8 + 14,5_8 =$
Solução:
$7 + 5 = 12_{10} = 14_8$
$1 + 5 + 4 = 10_{10} = 12_8$

$$\begin{array}{|cr|}\hline & 325,71_8 \\ + & 14,50_8 \\ \cline{2-2} & 342,41_8 \\ \hline\end{array}$$

Número positivo grande e número negativo
Exemplo: $00001111_2 + 11111010_2 =$
Solução:
$00001111_2 = 15_{10}$
$11111010_2 = -6_{10}$

$$\begin{array}{|cr|}\hline & 00001111_2 \\ + & 11111010_2 \\ \cline{2-2} & 100001001_2 \\ \hline\end{array}$$

Retirando o bit mais significativo tem-se $00001001_2 = 9_{10}$.

Número negativo grande e número positivo
Exemplo: $00010000_2 + 11101000_2 =$
Solução:
$00010000_2 = 16_{10}$
$11101000_2 = -24_{10}$

$$\begin{array}{|cr|}\hline & 00010000_2 \\ + & 11101000_2 \\ \cline{2-2} & 11111000_2 \\ \hline\end{array}$$

Tem-se que $11111000_2 = -8_{10}$.

Ambos negativos
Exemplo: $11111011_2 + 11110111_2 =$
Solução:
$11111011_2 = -5_{10}$
$11110111_2 = -9_{10}$

$$\begin{array}{|cr|}\hline & 11111011_2 \\ + & 11110111_2 \\ \cline{2-2} & 111110010_2 \\ \hline\end{array}$$

Retirando o bit mais significativo tem-se $11110010_2 = -14_{10}$.

Obs.: Em algumas referências representa-se a base com números porém em outras são utilizadas letras, a Tabela 5 apresenta um exemplo.

Base	
2	b
8	o
10	d
16	h

Tabela 5: Nomenclatura das bases

Exemplos:
$1111_2 = 1111_b$
$17_8 = 17_o$
$15_{10} = 15_d$
$F_{16} = F_h$

8.2 Subtração

A Tabela 6 apresenta o resultado da subtração de dois números de um bit para todas as situações possíveis. Pode-se verificar que na segunda linha o resultado da subtração não pode mais ser armazenado em apenas um BIT, gerando uma informação de Empresta 1.

$0 - 0 = 0$
$0 - 1 = 1 \; e \; empresta \; 1$
$1 - 0 = 1$
$1 - 1 = 0$

Tabela 6: Subtração binária

A operação de subtração de números binários é equivalente a operação com os números em decimal sendo modificado apenas o número máximo 9_{10} para 1_2.

Exemplo 1: $111_2 - 100_2 =$
Solução:

$$\begin{array}{rr} & 111_2 \\ - & 100_2 \\ \hline & 011_2 \end{array} \qquad \begin{array}{rr} & 7_{10} \\ - & 4_{10} \\ \hline & 3_{10} \end{array}$$

Exemplo 2: $1000_2 - 111_2 =$
Solução:

$$\begin{array}{rr} & 1000_2 \\ - & 111_2 \\ \hline & 0001_2 \end{array}$$

Exemplo 3: $10100_2 - 1011_2 =$
Solução:

$$\begin{array}{rr} & 10100_2 \\ - & 1011_2 \\ \hline & 01001_2 \end{array} \qquad \begin{array}{rr} & 20_{10} \\ - & 11_{10} \\ \hline & 9_{10} \end{array}$$

Exemplo 4: $1101,1_2 - 110_2 =$
Solução:

$$\begin{array}{rr} & 1101,1_2 \\ - & 110,0_2 \\ \hline & 0111,1_2 \end{array}$$

Exemplo 5: Neste exemplo será feita a operação em octal.
$3521_8 - 1674_8 =$
Solução:
$11_8 - 4_8 = 5_8$
$11_8 - 7_8 = 2_8$
$14_8 - 6_8 = 6_8$

$$\begin{array}{rr} & 3521_8 \\ - & 1674_8 \\ \hline & 1625_8 \end{array}$$

Obs.: Outro método para fazer uma operação de subtração:
Exemplo 1: $1010010_2 - 1001111_2 =$
Solução:

$$\begin{array}{rlcrlcl} & 1010010_2 & \Rightarrow \overline{A} \Rightarrow & & 0101101_2 & & \\ - & 1001111_2 & & + & 1001111_2 & & \\ \hline & 0000011_2 & & & 1111100_2 & \Rightarrow \overline{R} \Rightarrow & 0000011_2 \end{array}$$

Exemplo 2: $11000110_2 - 1011111_2 =$
Solução:

$$\begin{array}{rlcrlcl} & 11000110_2 & \Rightarrow \overline{A} \Rightarrow & & 00111001_2 & & \\ - & 01011111_2 & & + & 01011111_2 & & \\ \hline & 01100111_2 & & & 10011000_2 & \Rightarrow \overline{R} \Rightarrow & 01100111_2 \end{array}$$

8.3 Multiplicação

A Tabela 7 apresenta o resultado da multiplicação de dois números de um bit para todas as situações possíveis.

$0 \times 0 = 0$
$0 \times 1 = 0$
$1 \times 0 = 0$
$1 \times 1 = 1$

Tabela 7: Multiplicação binária

Exemplo 1: $10101_2 \times 111_2 =$
Solução:

$$\begin{array}{cr} & 10101_2 \\ \times & 111_2 \\ \hline & 10101\ _2 \\ + & 10101\ \ _2 \\ + & 10101\ \ \ _2 \\ \hline & 10010011_2 \end{array}$$

$$\begin{array}{cr} & 21_{10} \\ \times & 7_{10} \\ \hline & 147_{10} \end{array}$$

8.4 Divisão

A operação de divisão utiliza de forma conjunta as operações de multiplicação e subtração.
Exemplo 1: $1011_2 \div 1101_2 =$
Solução:
$\boxed{1011_2 \div 1101_2 = 0,11011_2}$

9 Complemento de 1 e complemento de 2

9.1 Complemento de 1

Alteração de todos os 1s por 0s e todos os 0s por 1s.
10110010 $\Rightarrow$ número binário
01001101 $\Rightarrow$ complemento de 1 do número binário acima

9.2 Complemento de 2

$\boxed{\text{Complemento de } 2 = (\text{Complemento de } 1) + 1}$

Exemplo 1: $10110010_2 =$

Solução:

$$\boxed{\begin{array}{cc} 10110010 & \text{Número Binário} \\ \Downarrow & \Downarrow \\ 01001101 & \text{Complemento de 1} \\ + \quad\quad 1 & \Downarrow \\ \hline 01001110 & \text{Complemento de 2} \end{array}}$$

10 Representação de um número com sinal

Existem três maneiras de um número inteiro com sinal ser representado na forma binária: Sinal-Magnitude, Complemento de 1 e Complemento de 2. Os número não inteiros e muito pequenos ou muito grandes podem ser representados na forma de ponto flutuante.

10.1 Bit de Sinal

O bit mais a esquerda é o sinal do número binário, se for 1 o número é negativo e se for 0 o número é positivo.

10.2 Sinal-Magnitude

Quando um número está representado em Sinal-Magnitude o bit mais a esquerda é o Bit de Sinal e os outros bits representam a magnitude do número.

Exemplo:

$+25_{10} = 00011001_2$

$-25_{10} = 10011001_2$

Note que para reprentar os números +25 e −25 apenas o primeiro bit foi alterado.

10.3 Complemento de 1

Na representação de números negativos em complemento de 1 todos os bits do número devem ser negados individualmente.

Exemplo:

$+25_{10} = 00011001_2$

$-25_{10} = 11100110_2$

10.4 Complemento de 2

Na representação de números negativos em complemento de 2 deve-se fazer o complemento de 1 do número e somar 1.

Exemplo:

$+25_{10} = 00011001_2$

$-25_{10} = 11100111_2$

A Tabela 8 apresenta os números binários de 4 bits em complemento de 2.

Para converter o número 2 para −2 realiza-se a operação abaixo:

$0010 \rightarrow \text{Complemento de } 1 \rightarrow 1101 + 1 = 1110$

onde $1110 = -2$.

1000	−8
1001	−7
1010	−6
1011	−5
1100	−4
1101	−3
1110	−2
1111	−1
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Tabela 8: Complemento de 2

11 Valor em decimal de um número com sinal

Sinal-Magnitude:

$10010101_2 = -21_{10}$
$0010101_2 = 2^4 + 2^2 + 2^0 = 16 + 4 + 1 = 21_{10}$

Complemento de 1:
$00010111_2 = +23_{10}$
$2^4 + 2^2 + 2^1 + 2^0 = 16 + 4 + 2 + 1 = +23_{10}$

$11101000_2 = -23_{10}$
$-2^7 + 2^6 + 2^5 + 2^3 = -128 + 64 + 32 + 8 = -24_{10}$
$-24 + 1 = -23_{10}$

Complemento de 2:
$01010110_2 = +86_{10}$
$2^6 + 2^4 + 2^2 + 2^1 = 64 + 16 + 4 + 2 = +86_{10}$

$10101010_2 = -86_{10}$
$-2^7 + 2^5 + 2^3 + 2^1 = -128 + 32 + 8 + 2 = -86_{10}$

12 Exercícios e Bibliografia

12.1 Exercícios

Realize as conversões abaixo:

$0,125_{10}$ para binário
$0,1011_2$ para decimal
$AF2A_h$ para binário
7321_8 para hexadecimal

12.2 Bibliografia

Entre os livros recomendados temos:

Digital Fundamentals, Thomas L. Floyd, Prentice Hall
Elementos de Eletrônica Digital, Idoeta e Capuano, Érica

Funções Lógicas e Portas Lógicas

Prof. Daniel Barros Júnior, Prof. Eduardo Augusto Bezerra

16 de agosto de 2004

Página: www.ee.pucrs.br/∼dbarros
E-mail: dbarros@ee.pucrs.br
Versão: 1.15

1 Introdução

A Tabela 1 apresenta o padrão que será utilizado.

Níveis Lógicos
0 → Falso (Desligado)
1 → Verdadeiro (Ligado)

Tabela 1: Níveis Lógicos

2 NOT

Função complemento ou negação.

Notação: $f(A) = \overline{A}$ ou $f(A) = /A$
A Figura 1 apresenta o esquema elétrico e o símbolo de uma porta lógica NOT.

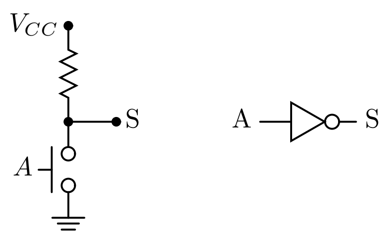


Figura 1: Porta NOT

Tabela Verdade:

A	S
0	1
1	0

CI 7404

3 AND

Função que a saída é verdadeira se todas as entradas são verdadeiras.

Notação: $f(A, B, C, \ldots) = A.B.C. \ldots$ ou $f(A, B, C, \ldots) = A\,B\,C \ldots$
A Figura 2 apresenta o esquema elétrico e o símbolo de uma porta lógica AND.

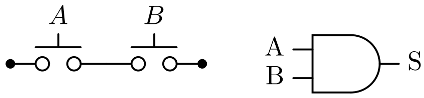


Figura 2: Porta AND

Tabela Verdade:

A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

CI 7408

4 OR

Função que a saída é verdadeira se qualquer entrada for verdadeira.

Notação: $f(A, B, C, \ldots) = A + B + C + \ldots$
A Figura 3 apresenta o esquema elétrico e o símbolo de uma porta lógica OR.

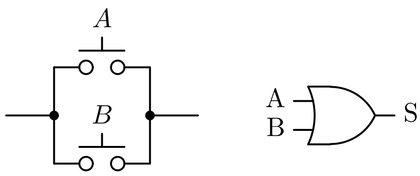


Figura 3: Porta OR

Tabela Verdade:

A	B	S
0	0	0
0	1	1
1	0	1
1	1	1

CI 7432

5 XOR

Notação: $f(A, B) = A \oplus B$
A Figura 4 apresenta o símbolo de uma porta lógica XOR.

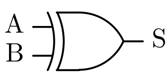


Figura 4: Porta XOR

Tabela Verdade:

A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

CI 7486

6 NAND

Notação: $f(A,B,C,\ldots) = \overline{A.B.C.\ldots}$

A Figura 5 apresenta o símbolo de uma porta lógica NAND.

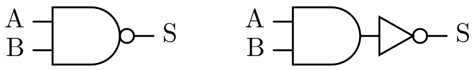


Figura 5: Porta NAND

Tabela Verdade:

A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

CI 7400

7 NOR

Notação: $f(A,B,C,\ldots) = \overline{A+B+C+\ldots}$

A Figura 6 apresenta o símbolo de uma porta lógica NOR.

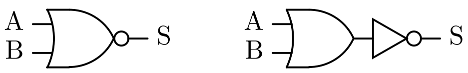


Figura 6: Porta NOR

Tabela Verdade:

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

CI 7402

8 XNOR

Notação: $f(A,B) = \overline{A \oplus B}$

A Figura 7 apresenta o símbolo de uma porta lógica NXOR.

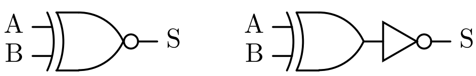


Figura 7: Porta NXOR

Tabela Verdade:

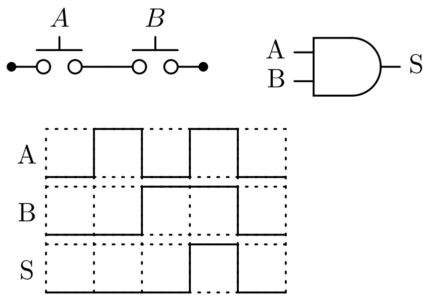
A	B	S
0	0	1
0	1	0
1	0	0
1	1	1

CI 74266

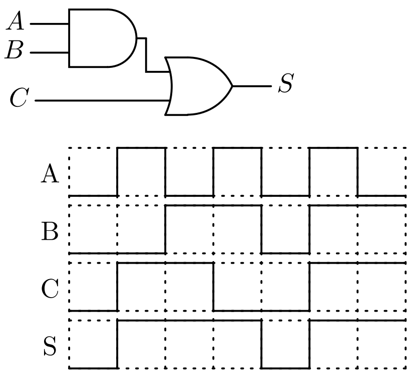
9 Diagrama de Tempo

Exemplos com diagrama de tempo:

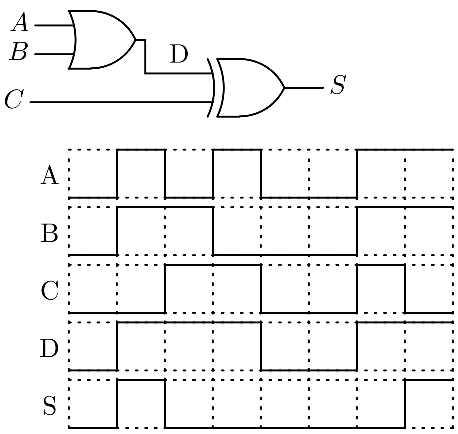
Exemplo 1:



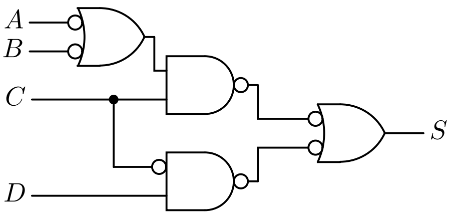
Exemplo 2:

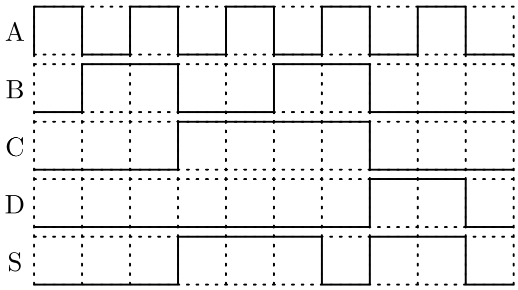


Exemplo 3:



Exemplo 4:





$S = (\overline{A} + \overline{B}).C + \overline{C}.D$

10 Equivalência das portas lógicas

Consiste em usar uma porta lógica para executar a função de outra.

Exemplo 1: Usar uma porta OR e portas NOT, para executar a função AND.

Solução:

A	B	A+B	$\overline{A}+\overline{B}$	$\overline{\overline{A}+\overline{B}}$
0	0	0	1	0
0	1	1	1	0
1	0	1	1	0
1	1	1	0	1

$A.B = \overline{\overline{A}+\overline{B}}$

Exemplo 2: Implementar um inversor (NOT) com portas NAND.

Tabela Verdade da Porta NAND:

A	B	S
0	0	1
0	1	1
1	0	1
1	1	0

Solução 1:

Ligando-se uma das entradas em 1 tem-se:

A	B	S
1	0	1
1	1	0

Solução 2:

Ligando-se as entradas uma na outro tem-se:

A	B	S
0	0	1
1	1	0

Exemplo 3: implementar um inversor com portas NOR.

Tabela Verdade da Porta NOR:

A	B	S
0	0	1
0	1	0
1	0	0
1	1	0

Solução 1:

Ligando-se uma das entradas em 0 tem-se:

A	B	S
0	0	1
0	1	0

Solução 2:

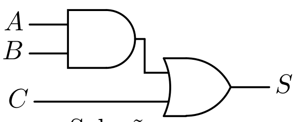
Ligando-se as entradas uma na outra tem-se:

A	B	S
0	0	1
1	1	0

11 Expressões Booleanas

11.1 Expressão Booleana a partir de diagrama de portas lógicas

Exemplo 1:



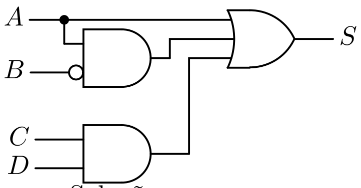
Solução:

$M = A\,B$

$S = M + C$

$\boxed{S = A\,B + C}$

Exemplo 2:



Solução:

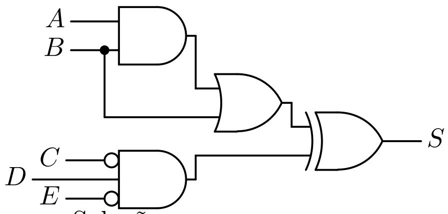
$M = A\,\overline{B}$

$N = C\,D$

$S = A + M + N$

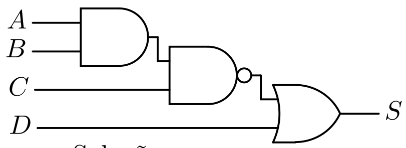
$\boxed{S = A + A\,\overline{B} + C\,D}$

Exemplo 3:



Solução:
$M = A\,B$
$N = B + M$
$P = \overline{C}\,D\,\overline{E}$
$S = N \oplus P$
$\boxed{S = (B + A\,B) \oplus (\overline{C}\,D\,\overline{E})}$

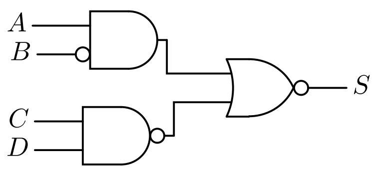
Exemplo 4:



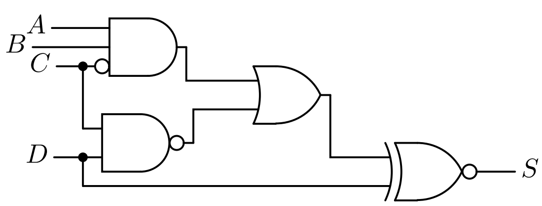
Solução:
$M = A\,B$
$N = \overline{M\,C}$
$S = N + D$
$\boxed{S = \overline{A\,B\,C} + D}$

11.2 Obtenção do circuito a partir das equações booleanas

Exemplo 1: $f(A,B,C,D) = \overline{A\,\overline{B} + \overline{C\,D}}$
Solução:



Exemplo 2: $f(A,B,C,D) = \overline{\left(A\,B\,\overline{C} + \overline{C\,D}\right) \oplus D}$
Solução:



11.3 Obtenção da expressão booleana a partir da tabela verdade

Procedimento:

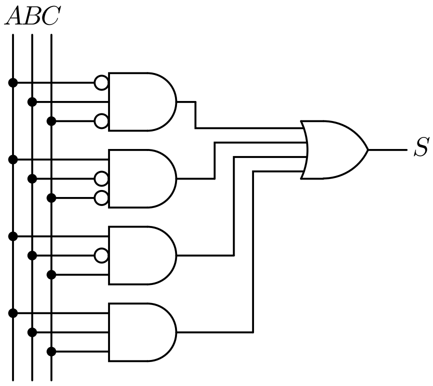
1. Pesquisa-se todas as posições de saída 1;
2. Faz-se uma Porta AND de todas as entradas associadas a esta saída;

3. Agrupa-se todas as saídas utilizando a Porta OR.

Exemplo 1:
Monte o circuito e a expressão booleana da tabela abaixo:

A	B	C	S	
0	0	0	0	
0	0	1	0	
0	1	0	1	$\overline{A}\,B\,\overline{C}$
0	1	1	0	
1	0	0	1	$A\,\overline{B}\,\overline{C}$
1	0	1	1	$A\,\overline{B}\,C$
1	1	0	0	
1	1	1	1	$A\,B\,C$

Solução:
$S = \overline{A}\,B\,\overline{C} + A\,\overline{B}\,\overline{C} + A\,\overline{B}\,C + A\,B\,C$



Exemplo 2: Monte o circuito e a expressão booleana da tabela abaixo:

A	B	X	Y	Z
0	0	0	1	0
0	1	1	0	1
1	0	0	1	1
1	1	1	1	0

Solução:
$X = \overline{A}\,B + A\,B$
$Y = \overline{A}\,\overline{B} + A\,\overline{B} + A\,B$
$Z = \overline{A}\,B + A\,\overline{B}$

11.4 Obtenção da tabela verdade a partir da equação booleana

Procedimento:

1. Verifique o número de variáveis da expressão booleana;
2. Fazer uma tabela verdade com todas as possibilidades;
3. Verificar o comportamento de cada entrada.

Exemplo 1: $Z = A\,B + \overline{A}\,C$
Solução:

A	B	C	Z	
0	0	0	0	
0	0	1	1	$\overline{A}\,\overline{B}\,C$
0	1	0	0	
0	1	1	1	$\overline{A}\,B\,C$
1	0	0	0	
1	0	1	0	
1	1	0	1	$A\,B\,\overline{C}$
1	1	1	1	$A\,B\,C$

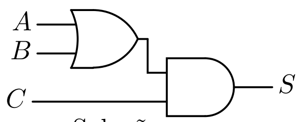
$Z = \overline{A}\,\overline{B}\,C + \overline{A}\,B\,C + A\,B\,\overline{C} + A\,B\,C \quad \equiv \quad A\,B + \overline{A}\,C$

11.5 Obtenção da tabela verdade através do circuito elétrico

Procedimento:

1. Determinar o número de entradas do circuito;
2. Fazer uma tabela verdade com todas as possibilidades;
3. Verificar o comportamento do circuito.

Exemplo 1:



Solução:

A	B	C	A+B	C	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	0	0
1	1	1	1	1	1

11.6 Obtenção da tabela verdade a partir de uma descrição textual

Exemplo 1:

Determinar a tabela verdade que implemente o problema abaixo.

Em uma sala há três pessoas (A,B,C) que podem votar Sim (1) ou Não (0), sobre um determinado assunto. Implemente a lógica que seja capaz de identificar as seguintes situações.

X - Indicar que a maioria votou Sim;
Y - Indicar que a maioria votou Não;
W - Indicar que houve unanimidade de Sim;
Z - Indicar que houve unanimidade de Não.

Solução:

A	B	C	X	Y	W	Z
0	0	0	0	0	0	1
0	0	1	0	1	0	0
0	1	0	0	1	0	0
0	1	1	1	0	0	0
1	0	0	0	1	0	0
1	0	1	1	0	0	0
1	1	0	1	0	0	0
1	1	1	0	0	1	0

$X = \overline{A}\,B\,C + A\,\overline{B}\,C + A\,B\,\overline{C}$
$Y = \overline{A}\,\overline{B}\,C + \overline{A}\,B\,\overline{C} + A\,\overline{B}\,\overline{C}$
$W = A\,B\,C$
$Z = \overline{A}\,\overline{B}\,\overline{C}$

12 Exercícios e Bibliografia

12.1 Exercícios

12.2 Bibliografia

Entre os livros recomendados temos:

Digital Fundamentals, Thomas L. Floyd, Prentice Hall
Elementos de Eletrônica Digital, Idoeta e Capuano, Érica

Álgebra Booleana

Prof. Daniel Barros Júnior, Prof. Eduardo Augusto Bezerra

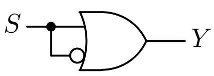
16 de agosto de 2004

Página: www.ee.pucrs.br/∼dbarros
E-mail: dbarros@ee.pucrs.br
Versão: 1.15

1 Introdução

Desenvolvida por George Boole em 1854. O principal objetivo é a simplificação dos circuitos lógicos.

Exemplo 1:
$f(B, S, F) = B\,\overline{S}\,\overline{F} + B\,S\,\overline{F}$
Solução:
$f(B, S, F) = B\,\overline{S}\,\overline{F} + B\,S\,\overline{F}$
$f(B, S, F) = B\,\overline{F}\,\left(\overline{S} + S\right)$
$f(B, S, F) = B\,\overline{F}$



1.1 Leis da Álgebra Booleana

Prioridade	
1	Parênteses
2	NOT
3	AND
4	OR

Comutatividade:
$A + B = B + A$
$A\,B = B\,A$

Associatividade:
$A + (B + C) = (A + B) + C$
$A\,(B\,C) = (A\,B)\,C$

Distributividade:
$A\,(B + C) = A\,B + A\,C$

1.2 Regras da Álgebra Booleana

$A+0=A$		$A\,.0=0$		$\overline{\overline{A}}=A$
$A+1=1$		$A\,.1=A$		$(A+B)\ (A+C)=A+B\,C$
$A+A=A$		$A\,.A=A$		
$A+\overline{A}=1$		$A\,.\overline{A}=0$		
$A+A\,B=A$				
$A+\overline{A}\,B=A+B$				

1.3 Teorema de Demorgan

$$\boxed{\overline{A\,B\,C\ldots}=\overline{A}+\overline{B}+\overline{C}+\ldots}$$

$$\boxed{\overline{A+B+C+\ldots}=\overline{A}\,\overline{B}\,\overline{C}\ \ldots}$$

Prove as expressões abaixo:
Exemplo 1:
$A+A\,B=A$
Solução:

$$\begin{aligned} A+A\,B &= A\ (1+B) \\ &= A.1 \\ &= A \end{aligned}$$

Exemplo 2: $A+\overline{A}\,B=A+B$
Solução:

$$\begin{aligned} A+\overline{A}\,B &= (A+AB)+\overline{A}\,B \\ &= A\,A+A\,B+\overline{A}\,B \\ &= A\,A+A\,\overline{A}+A\,B+\overline{A}\,B \\ &= \left(A+\overline{A}\right)\ (A+B) \\ &= 1.\,(A+B) \\ &= A+B \end{aligned}$$

Resolver os exercício utilizando o Teorema de Demorgan:
Exemplo 1:
$\overline{(A+B+C)\ D}\ =$
Solução:

$$\begin{aligned} \overline{(A+B+C)\ D} &= \overline{A+B+C}+\overline{D} \\ &= \overline{A}\,\overline{B}\,\overline{C}+\overline{D} \end{aligned}$$

Exemplo 2:
$\overline{A\,B\,C+D\,E\,F}\ =$
Solução:

$$\begin{aligned} \overline{A\,B\,C+D\,E\,F} &= \overline{A\,B\,C}\ \overline{D\,E\,F} \\ &= \left(\overline{A}+\overline{B}+\overline{C}\right)\ \left(\overline{D}+\overline{E}+\overline{F}\right) \end{aligned}$$

Exemplo 3:
$\overline{A\,\overline{B}+\overline{C}\,D+E\,F}\ =$
Solução:

$$\begin{aligned} \overline{A\,\overline{B}+\overline{C}\,D+E\,F} &= \overline{A\,\overline{B}}\,\overline{\overline{C}\,D}\,\overline{E\,F} \\ &= \left(\overline{A}+\overline{\overline{B}}\right)\ \left(\overline{\overline{C}}+\overline{D}\right)\ \left(\overline{E}+\overline{F}\right) \\ &= \left(\overline{A}+B\right)\ \left(C+\overline{D}\right)\ \left(\overline{E}+\overline{F}\right) \end{aligned}$$

1.4 Simplificações usando Álgebra Booleana

Exemplo 1: $A\,B + A\ (B + C) + B\ (B + C) =$

Solução:

Simplificações		Operação utilizada
$A\,B + A\ (B + C) + B\ (B + C) =$		$A\ (B + C) = A\,B + A\,C$
$A\,B + A\,B + A\,C + B\,B + B\,C =$		$A\,A = A$
$A\,B + A\,B + A\,C + B + B\,C =$		$A + A = A$
$A\,B + A\,C + B + B\,C =$		$A + A\,B = A$
$A\,B + A\,C + B =$		$A + A\,B = A$
$A\,C + B$		Máxima simplificação

Exemplo 2:

$\left[A\,\overline{B}\ (C + B\,D) + \overline{A}\,\overline{B}\right]\ C =$

Solução:

Simplificações		Operação utilizada
$\left[A\,\overline{B}\ (C + B\,D) + \overline{A}\,\overline{B}\right]\ C =$		$A\ (B + C) = A\,B + A\,C$
$\left(A\,\overline{B}\,C + A\,\overline{B}\,B\,D + \overline{A}\,\overline{B}\right)\ C =$		$A\,\overline{A} = 0$
$\left(A\,\overline{B}\,C + 0 + \overline{A}\,\overline{B}\right)\ C =$		$A + 0 = A$
$\left(A\,\overline{B}\,C + \overline{A}\,\overline{B}\right)\ C =$		$A\ (B + C) = A\,B + A\,C$
$A\,\overline{B}\,C + \overline{A}\,\overline{B}\,C =$		$A\ (B + C) = A\,B + A\,C$
$\overline{B}\,C\ \left(A + \overline{A}\right) =$		$A + \overline{A} = 1$
$\overline{B}\,C$		Máxima simplificação

Exemplo 3: $\overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C} + A\,\overline{B}\,C + A\,B\,C =$

Solução:

Simplificações		Operação utilizada
$\overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,\overline{C} + A\,\overline{B}\,C + A\,B\,C =$		$A\ (B + C) = A\,B + A\,C$
$B\,C\ \left(\overline{A} + A\right) + A\,\overline{B}\ \left(\overline{C} + C\right) + \overline{B}\,\overline{C}\ \left(\overline{A} + A\right) =$		$A + \overline{A} = 1$
$B\,C + A\,\overline{B} + \overline{B}\,\overline{C}$		Máxima simplificação

Exemplo 4:

$\overline{A\,B + A\,C} + \overline{A}\,\overline{B}\,C =$

Solução:

Simplificações		Operação utilizada
$\overline{A\,B + A\,C} + \overline{A}\,\overline{B}\,C =$		$\overline{A + B + C + \ldots} = \overline{A}\,\overline{B}\,\overline{C}\ \ldots$
$\overline{A\,B}\ \overline{A\,C} + \overline{A}\,\overline{B}\,C =$		$\overline{A\,B\,C\ldots} = \overline{A} + \overline{B} + \overline{C} + \ldots$
$\left(\overline{A} + \overline{B}\right) + \left(\overline{A} + \overline{C}\right) + \overline{A}\,\overline{B}\,C =$		
$\overline{A} + \overline{A}\ \overline{C} + \overline{A}\ \overline{B} + \overline{B}\ \overline{C} + \overline{A}\,\overline{B}\,C =$		
$\overline{A} + \overline{A}\ \overline{C} + \overline{A}\ \overline{B}\ (1 + C) + \overline{B}\ \overline{C} =$		
$\overline{A}\ \left(1 + \overline{C} + \overline{B}\right) + \overline{B}\ \overline{C} =$		
$\overline{A} + \overline{B}\ \overline{C}$		Máxima simplificação

2 Mapa de Karnaugh

2.1 Formato

O mapa de Karnaugh é similar a tabela verdade pois apresenta todos os possíveis valores das variáveis de entrada e a saída resultante para cada entrada.

A \ B	0	1
0		
1		

A \ B	0	1
0	00	01
1	10	11

Figura 1: Mapa de Karnaugh de 2 variáveis

A \ BC	00	01	11	10
0				
1				

Figura 2: Mapa de Karnaugh de 3 variáveis

AB \ CD	00	01	11	10
00				
01				
11				
10				

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	C	D	F	E
10	8	9	B	A

Figura 3: Mapa de Karnaugh de 4 variáveis

2.2 Utilização

Para uma soma de produtos na forma padrão deve ser atribuído 1 para o local de cada um dos termos da soma e 0 (zero) para os pontos restantes.

Exemplo 1:

$$S = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,C + A\,B\,\overline{C} + A\,\overline{B}\,\overline{C}$$

Solução:

A \ BC	00	01	11	10
0	1	1	0	0
1	1	0	0	1

Exemplo 2:

$$\begin{aligned} S = {} & \overline{A}\,\overline{B}\,C\,D + \overline{A}\,B\,\overline{C}\,\overline{D} + A\,B\,\overline{C}\,D + A\,B\,C\,D + \\ & + A\,B\,\overline{C}\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}\,D + A\,\overline{B}\,C\,\overline{D} \end{aligned}$$

Solução:

AB \ CD	00	01	11	10
00	0	1	1	0
01	1	0	0	0
11	1	1	1	0
10	0	0	0	1

Para uma soma de produtos fora do padrão deve ser atribuído 1 para o local de cada uma das variações possíveis de cada um dos termos da soma e 0 (zero) para os demais pontos.

Exemplo 1:
$S = \overline{A} + A\,\overline{B} + A\,B\,\overline{C}$
Solução:

$\overline{A}$	$A\,\overline{B}$	$A\,B\,\overline{C}$
000	100	110
001	101	
010		
011		

A \ BC	00	01	11	10
0	1	1	1	1
1	1	1	0	1

Exemplo 2:
$S = \overline{B}\,\overline{C} + A\,\overline{B} + A\,B\,\overline{C} + A\,\overline{B}\,C\,\overline{D} + \overline{A}\,\overline{B}\,\overline{C}\,D + A\,\overline{B}\,C\,D$
Solução:

$\overline{B}\,\overline{C}$	$A\,\overline{B}$	$A\,B\,\overline{C}$	$A\,\overline{B}\,C\,\overline{D}$	$\overline{A}\,\overline{B}\,\overline{C}\,D$	$A\,\overline{B}\,C\,D$
0000	1000	1100	1010	0001	1011
0001	1001	1101			
1000	1010				
1001	1011				

AB \ CD	00	01	11	10
00	1	1	0	0
01	0	0	0	0
11	1	1	0	0
10	1	1	1	1

2.3 Simplificações

O objetivo do mapa de Karnaugh é possibilitar a obtenção da expressão booleana mais simplificada possível.

Agrupando os 1s:
Pode-se agrupar os 1s do mapa unindo as células adjacentes que contém 1s. O objetivo é maximizar a dimensão dos grupos e minimizar o número de grupos.

Regras:

1. Um grupo deve conter 2^n células (1,2,4,8,16, 32, ...);
2. As células do grupo devem ser adjacentes;
3. Incluir o maior número de células no grupo;
4. Cada 1 do mapa deve ser incluído em pelo menos um grupo. Os 1's já agrupados podem fazer parte de outros grupos.

Obs.: Manter apenas as variáveis que não mudam de valor.

Exemplo 1:
$S = \overline{A}\,\overline{B} + A\,\overline{B} + A\,B$
Solução:

A \ B	0	1
0	1	0
1	1	1

$S = A + \overline{B}$

Exemplo 2:

A \ BC	00	01	11	10
0				1
1	1	1		1

Solução:
$S = A\,\overline{B} + B\,\overline{C}$

Exemplo 3:

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Solução:
$S = \overline{B} + \overline{A}\,\overline{C} + A\,C$

Exemplo 4:

AB \ CD	00	01	11	10
00	1	1		
01	1	1	1	1
11				
10		1	1	

Solução:
$S = \overline{A}\,\overline{C} + \overline{A}\,B + A\,\overline{B}\,D$

Exemplo 5:

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Solução:
$S = \overline{D} + B\,\overline{C} + A\,\overline{B}\,C$

Exemplo 6:
$S = A\,\overline{B}\,C + \overline{A}\,B\,C + \overline{A}\,\overline{B}\,C + \overline{A}\,\overline{B}\,\overline{C} + A\,\overline{B}\,\overline{C}$

Solução:

A \ BC	00	01	11	10
0	1	1	1	
1	1	1		

$S = \overline{B} + \overline{A}\,C$

Exemplo 7:

AB \ CD	00	01	11	10
00	1			1
01		1		
11				
10	1			1

Solução:

$S = \overline{B}\,\overline{D} + \overline{A}\,B\,\overline{C}\,D$

2.4 Funções definidas parcialmente

O valor de saída do circuito pode não importar por dois motivos:

1. A saída não será utilizada;
2. A condição de entrada nunca irá acontecer.

As condições de entrada que estão nesta situação são chamadas de "don't care"e são representadas com um "X".

Estas saídas "X"podem assumir o estado 0 ou 1, usando-se o valor mais conveniente.

A	B	C	D	S
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0

A	B	C	D	S
1	0	0	0	1
1	0	0	1	1
1	0	1	0	X
1	0	1	1	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

Sem don't care

AB \ CD	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	0	0	0	0
10	1	1	0	0

$S = \overline{A}\,\overline{C} + \overline{B}\,\overline{C}$

Com don't care

AB \ CD	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	X	X	X	X
10	1	1	X	X

$S = \overline{C}$

2.5 Mapa de Karnaugh de 5 e 6 variáveis

A Figura 4 da página 8 mostra o mapa de Karnaugh de 5 variáveis.

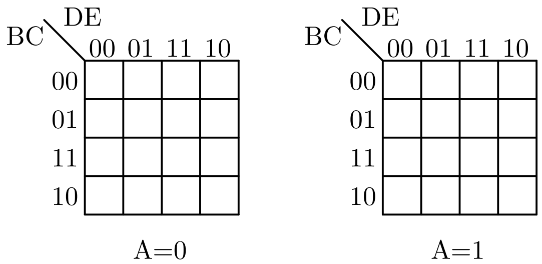


Figura 4: Mapa de Karnaugh de 5 variáveis

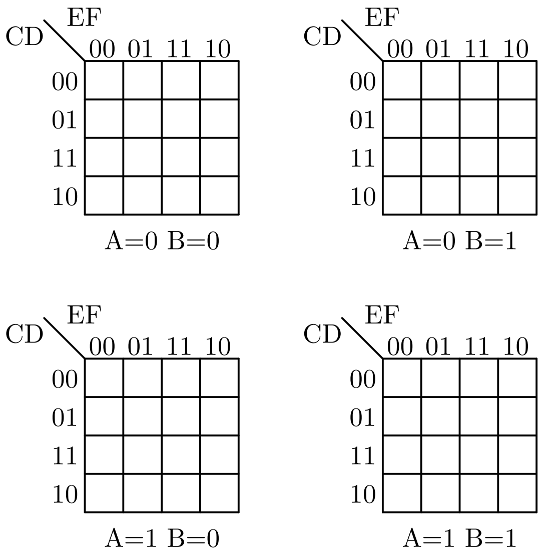


Figura 5: Mapa de Karnaugh de 6 variáveis

3 Exercícios e Bibliografia

3.1 Exercícios

3.2 Bibliografia

Entre os livros recomendados temos:

Digital Fundamentals, Thomas L. Floyd, Prentice Hall
Elementos de Eletrônica Digital, Idoeta e Capuano, Érica

Sumário

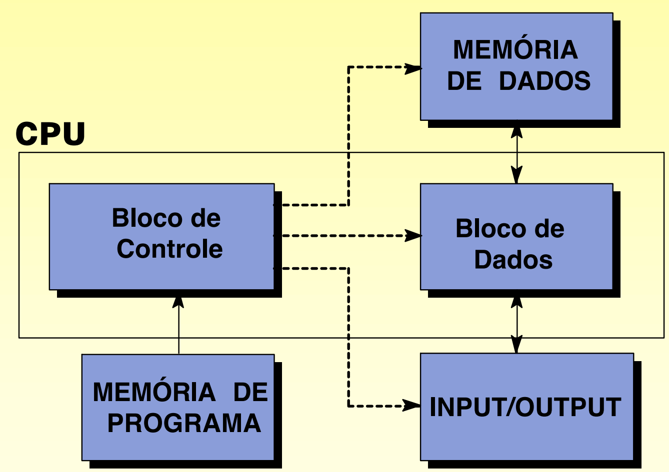
7 - Organização x Arquitetura

- Afinal, o que é um computador? Uma definição:
  - Máquina com capacidade de acesso a meios de armazenamento onde estão estocadas informações a serem processadas e as informações que dizem como processar as primeiras. CPU (Central Processing Unit)
    - Informações a serem processadas – dados
    - Informações de como processar – programas
      - Programas – seqüência de instruções retiradas de um conjunto fixo de instruções reconhecidas como tal pela máquina
  - Funcionamento: repetir, infinitamente, seqüência de 3 ações: buscar instrução, identificar instrução buscada, executar instrução buscada.
  - Execução de instruções pode incorrer em acesso a dispositivos de entrada e saída.

7 - Organização x Arquitetura

- Organização de Computadores – A visão abstrata do engenheiro (elétrico, de computação) de um computador:
  - Transistores, portas lógicas, registradores, unidades lógico-aritméticas, fios, multiplexadores, etc.
- Arquitetura de Computadores – A visão abstrata do programador de baixo nível (linguagem de montagem, em inglês, *assembly language*):
  - Instruções, registradores para armazenar dados, a linguagem de programação de montagem, modos de endereçamento, formato das instruções, etc.

Modelo Geral de um computador

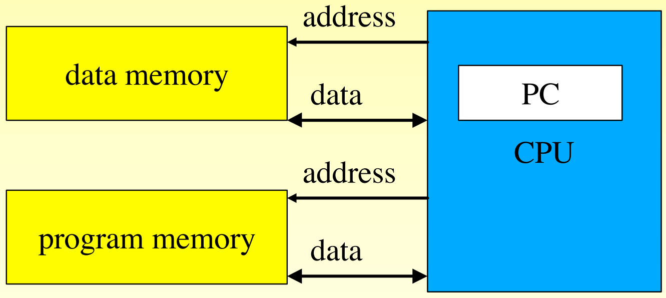


Comentar: FPU, MMU, CACHES, predição de salto (CPU hoje é chamada de *core*)

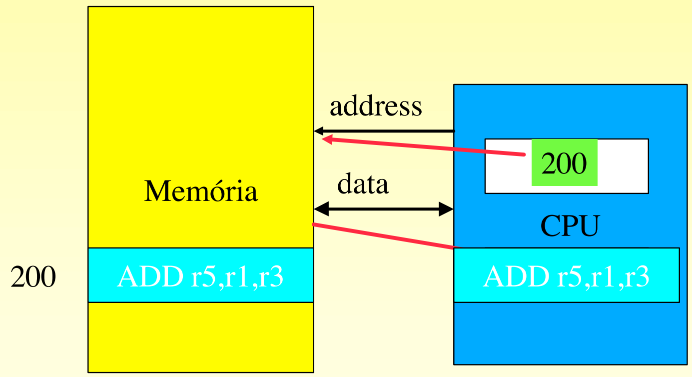
7 - Organização x Arquitetura

- Existem modelos gerais que estabelecem as formas de implementação da máquina computador
- Classificação de organizações de computadores:
  - **Modelo von Neumann** – dados e programas compartilham um meio de armazenamento único
    - Mais simples, menos restritivo, menos eficiente – dados e programas misturados permitem ao programador intercambiar a semântica de dados e programas ao longo do tempo
  - **Modelo Harvard** – dados e programas estocados em meios de armazenamento distintos
    - Mais propenso a fomentar paralelismo, mais caro, mais complexo – dados e programas separados permitem que ambos sejam facilmente tratados em paralelo

Interface CPU-memória no modelo Harvard



Interface CPU-memória no modelo von Neuman



von Neumann vs. Harvard

- Harvard permite duas leituras de memória simultâneas (dado e instrução).
- A maioria dos processadores DSP (celulares, telecom, câmeras digitais,…) usam organização Harvard:
  - Maior largura de band de memória;
  - Tempo de acesso a dados mais previsível.