Sistemas de numeração e códigos

- Representação numérica
- Representação binária
- Representação octal
- Representação hexadecimal
- Binário codificado decimal
- Códigos alfanuméricos
- Detecção de erros usando paridade
- Aplicações

Representação numérica

- Usamos 10 símbolos para representar quantidades no sistema decimal
- 0123456789
- Como contamos usando estes símbolos?
  - Podemos raciocinar que a contagem consiste em acrescentar sempre uma quantidade e usar o símbolo correspondente. Quando não houver mais símbolos usamos a combinação de símbolos mais simples possível para a quantidade a ser representada

Representação numérica

- Considere o numero $(347)_{10}$ na base 10 este numero pode ser descrito na forma:
  - $347 = 3x10^2 + 4x10^1 + 7x10^0$
- Pelo mesmo raciocínio dado o numero $(10111)_2$ na base 2 pode-se escrever que:
  - $10111 = 1x2^4 + 0x2^3 + 1x2^2 + 1x2^1 + 1x2^0 = (23)_{10}$

Representação numérica

- De uma maneira geral, a representação no sistema decimal de um numero na base b qualquer é:

$$N_{10} = \sum_{-\infty}^{\infty} a_j \beta^j = \ldots a_j \beta^j + a_{j-1}\beta^{j-1} + \ldots + a_2\beta^2 + a_1\beta^1 + a_0\beta^0 + a_{-1}\beta^{-1} + \ldots$$

onde β é a base

No caso de um numero inteiro:

$$(a_j a_{j-1} \ldots a_2 a_1 a_0)_\beta \rightarrow 0 \le a_k \le (\beta - 1) \cdots k = 1 \ldots j$$

Representação numérica

No caso fracionário temos:

$$(0.a_j a_{j-1} \ldots a_2 a_1 a_0)_\beta \rightarrow 0 \le a_k \le (\beta - 1) \cdots k = 1 \ldots j$$

Representação binária

- Um numero binário representa quantidades com dois símbolos conhecidos como bits. O bit pode ser 0 ou 1.
- Contagem binária. Usa-se o mesmo raciocínio da contagem decimal

Representação binária

- Conversão binário Decimal inteira
- Conversão binário decimal fracionaria
- Conversão decimal-binário inteira
- Conversão decimal-binário fracionário

Binário codificado decimal

- Codigo BCD
  - Cada dígito decimal de 0 a 9 é representado por um código binário de 4 bits
  - Exemplo:converta 35 para o código BCD

Representação Octal

- Conversão Octal- Decimal
- Conversão Decimal-octal
- Converção Binário-Octal e Octal-Binário

Representação Hexadecimal

- Conversão Decimal Hexadecimal
- Conversão binário Hexadecimal
- Conversão Hexadecimal-binário e binário-Hexa
- Conversão Hexadecimal-octal e Octal-Hexa

Códigos alfanumérios

- Mais usado código Ascii
  - A $(100\ 0001)_2$; $(101)_8$; 41h
  - B $(100\ 0010)_2$;$(102)_8$; 42h
  - Ver tabela no Livro.

Detecção de erro usando Paridade

- Paridade: Refere-se à quantidade de numeral 1 presente no numero
  - Paridade Par: Numero par de 1's
  - Paridade impar: numero impar de 1's
- Exemplo paridade par:

Enviar a letra G 1000111 0 → bit de paridade par acrescenta-se o zero pois existem numero par de 1's

Enviar a letra E 1000101 1 → número impar de 1's

- Exemplo paridade impar:

Enviar a letra P 1010000 1 → bit de paridade par acrescenta-se o um pois existem numero par de 1’s

Enviar a letra E 1000101 0 → número impar de 1’s

Sistemas de numeração e códigos

Aritmética binária

Prof. Dr. Ernane Costa

Aritmética binária

- A aritmética binária é essencial em todos os computadores e em muitos sistemas digitais. Nesta aula veremos:
  - Operações binárias básicas
  - Complemento de 1 e de 2 de números binários
  - Números binários sinalizados
  - Operações aritméticas com números sinalizados

Aritmética binária

- O que acontece se a soma de dois números for maior que 9 ?
  - Geramos um carry (vai um…)

$$\begin{array}{r} 5 \\ +\ 5 \\ \hline 10 \end{array}$$

- O mesmo acontece quando somamos digitos binários

$$\begin{array}{r} 0 \\ +\ 0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 0 \\ +\ 1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 0 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 1 \\ \hline 10 \end{array}$$

Operações binárias básicas

- Soma:
- 0 + 0 = 0 “carry” (vai um...) de 0
- 0 +1 = 1 vai um 0
- 1 + 0 = 1 vai um 0
- 1 + 1 = 10 da zero e vai um

- Exemplo:

$$\begin{array}{r} 111 \\ +11 \\ \hline 1010 \end{array} \Leftrightarrow \begin{array}{r} 7 \\ 3 \\ \hline 10 \end{array}$$

- Subtração
  - 0 - 0 = 0
  - 1 - 1 = 0
  - 1 - 0 = 1
  - 10 - 1 = 1 ou seja 0 –1 com um “emprestimo” de 1

- Exemplo

$$\begin{array}{r} 101 \\ -\ \underline{011} \\ 010 \end{array} \Leftrightarrow \begin{array}{r} 5 \\ -\ \underline{3} \\ 2 \end{array}$$

- Multiplicação (implica também a soma)
  - 0 x 0 = 0
  - 0 x 1 = 0
  - 1 x 0 = 0
  - 1 x 1 = 1

- Exemplo

$$
\begin{array}{r}
111 \\
101 \\
- \overline{\cdots 111} \\
\cdot 000 \\
111 \\
\text{-----} \\
100011
\end{array}
\Leftrightarrow
\times \begin{array}{r}
7 \\
5 \\
\hline
35
\end{array}
$$

- Divisão Binária (exatamente igual a decimal)
  - Exemplo:
  - $110 \div 11 = 10$

Complemento de 1 e 2 de números binários

- Complemento de 1 e d 2 são importantes porque eles permitem a representação de números binários negativos
- O complemento de 1 de um numero binário é obtido trocando zeros por um.

$$10110010 \xleftrightarrow{\text{complemento.1}} 01001101$$

- O complemento de 2 é o complemento de um mais a soma de 1
- Exemplo:
- Ache o complemento de 2 do numero 10110010
    - C1 = 01001101
    - C2 = 01001101 + 1 = 01001110 = complemento de 2

Números binários sinalizados

- Um numero binário sinalizado possui o bit de sinal e os bits de magnitude.
  - 00011001
    - (0 equivale a +)
    - (1 equivale a -)

Magnitude

Bit de sinal

- Pode ser usado três sistemas diferentes:
  - O sistema magnitude – sinal:
    - Neste sistema o número binário com sinal tem duas partes: Exemplo: considere o seguinte numero binário de oito bits com sinal:
  - 00011001 = +25 (0 equivale a +)
  - 10011001 = -25 (1 equivale a -)

- Sistema complemento de 1
  - Neste sistema o numero negativo é o complemento de 1 de sua magnitude positiva.
    - Exemplo
    - +25 = 00011001
    - -25 = 11100110
- Sistema complemento de 2
  - Neste sistema o numero negativo é o complemento de 2 de sua magnitude positiva
    - Exemplo
    - +25 = 00011001
    - -25 = 11100111

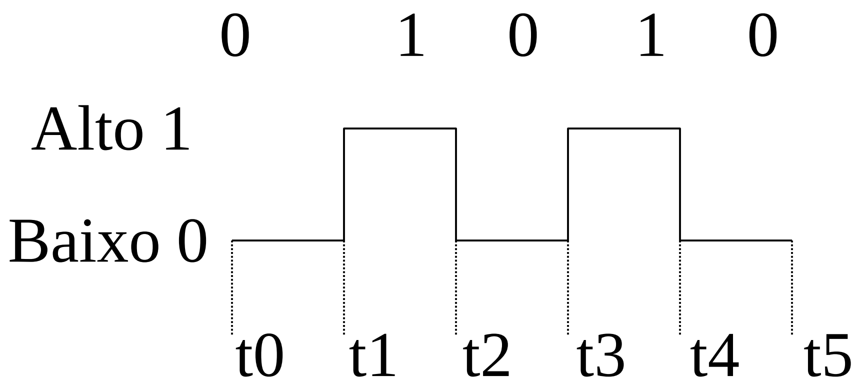
Exemplos

- Conversão de números sinalizados
- Soma
- Subtração
- Multiplicação
- E divisão
- Atividade complementares: Resolver todos os exercícios desafiadores do livro do Tocci

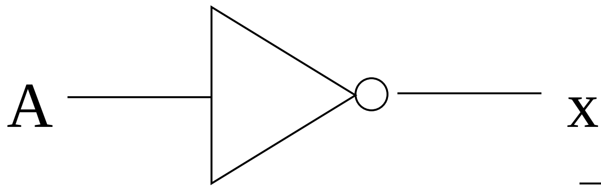
Portas lógicas e álgebra booleana

- Diagrama de tempo
- Porta inversora
- Porta and - e
- Porta or – ou
- Porta nand não-e
- Porta nor não ou
- Porta Xor

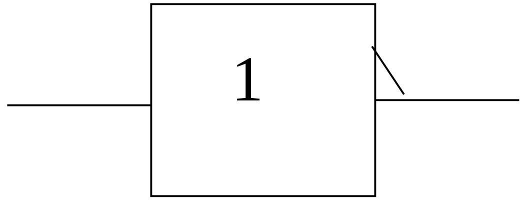
Diagrama de tempo



Porta inversora



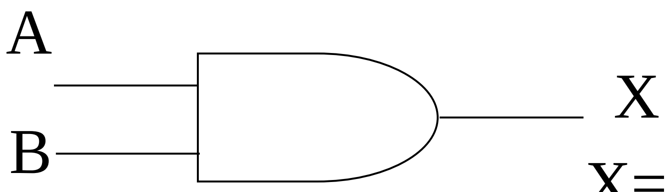
$X = \overline{A}$ = não A



Porta inversora (ANSI/IEEE Std. 91-1984)

Porta END (e)

Porta and de duas entradas(ANSI/IEEE Std. 91-1984)



X=A.B

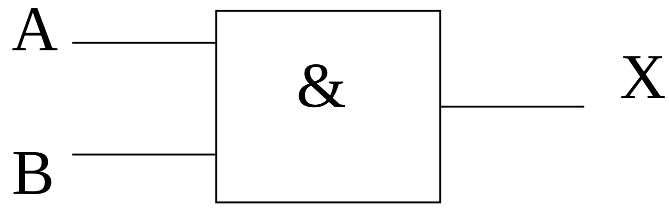
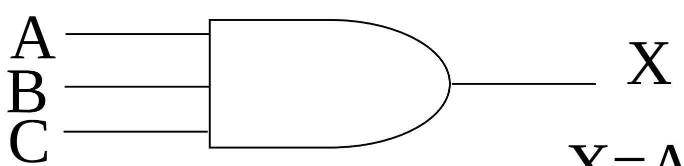


Tabela da verdade

A	B	X
1	1	1
0	1	0
1	0	0
0	0	0

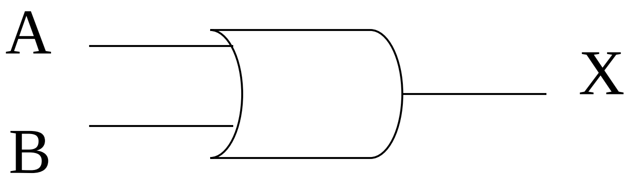


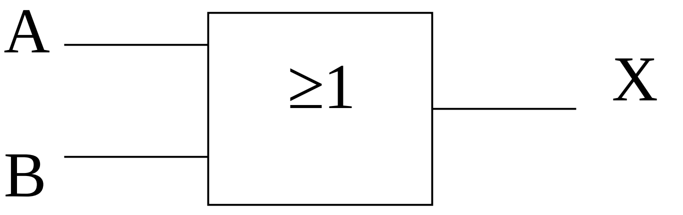
X=A.B.C

Porta and de três entradas(ANSI/IEEE Std. 91-1984)

Porta or (ou)

Porta or de duas entradas(ANSI/IEEE Std. 91-1984)



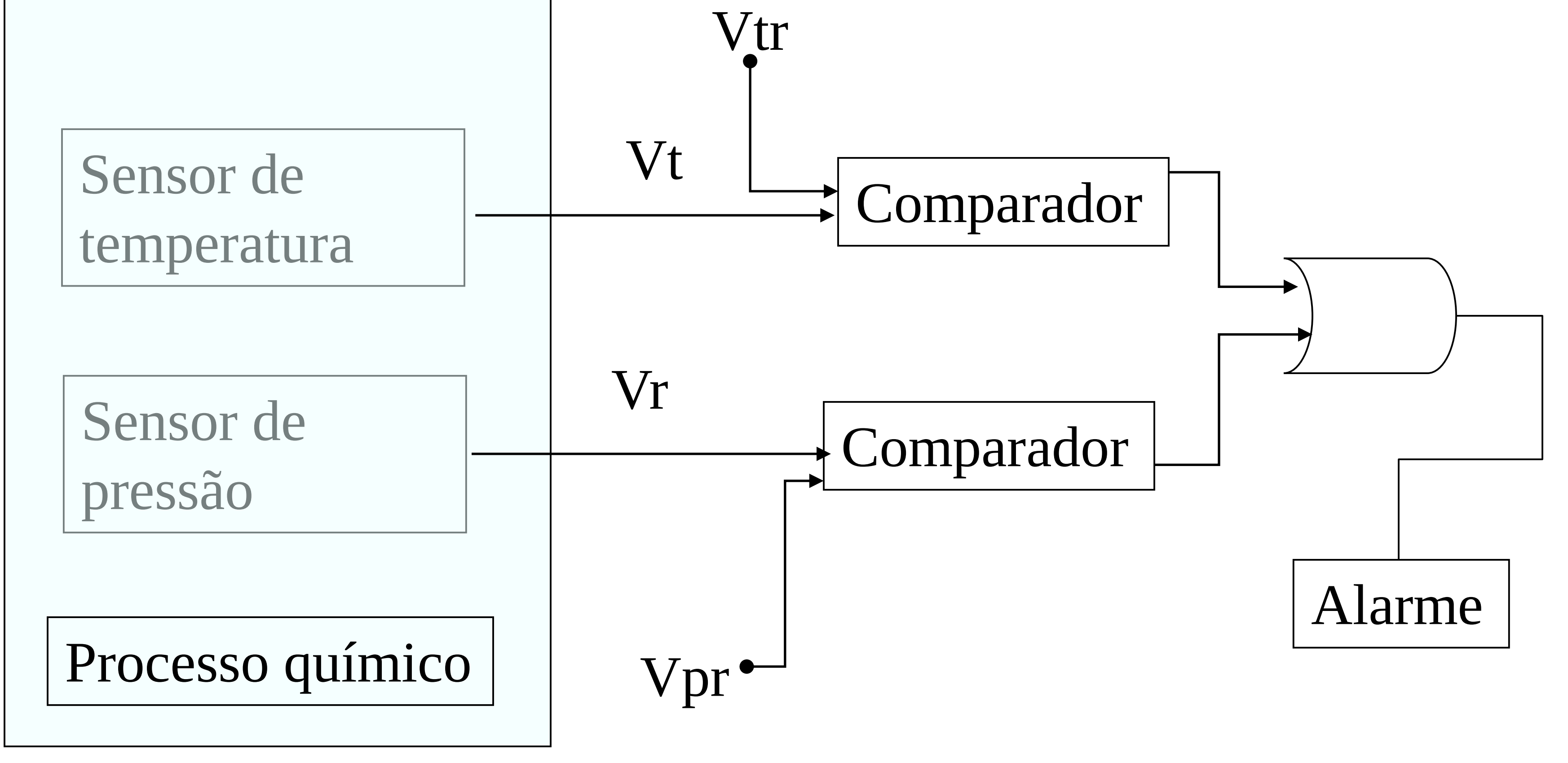


X=A+B

Tabela da verdade

A	B	X
1	1	1
0	1	1
1	0	1
0	0	0

Exemplo Aplicação da porta OR em um sistema de alarme



Portas lógicas representação algébrica

- Representação algébrica das portas:
- inversora
- And
- Or
- Nand
- Nor
- Descrevendo circuitos lógicos

- Obtendo circuitos lógicos a partir de expressões lógicas

- Exemplos:
- Desenhe o circuito que implementa a expressão $X = AB + \overline{B}C$

Teoremas da álgebra booleana

Uma variável

1. $x.0 = 0$
2. $x.1 = x$
3. $x.x = x$
4. $x.\overline{x}=0$
5. $x+0=x$
6. $x +1=1$
7. $x+x=x$
8. $x+\overline{x}=1$

Duas variáveis

9. $x+y = y+x$
10. $x.y =y. x$
11. $x+(y+z) = (x+y)+z=x+y+z$
12. $x.(yz)=(xy)z=xyz$
13. $x(y+z)=xy+xz$

13b. $(w+x)(y+z)=wy+xy+wz+xz$

14. $x+xy=x$
15. $x+\overline{x}y=x+y$

exemplos

- Simplifique a expressão $A\overline{B}D+A\overline{B}\,\overline{D}$
- Fatore as variáveis comum $A\overline{B}$ usando o teorema 13:
  - $y = A\overline{B}(D+\overline{D})$
  - Pelo teorema 8 o termo entre parênteses é um
  - Logo:
    - $Y = A\overline{B}.1$ usando o teorema 2 temos $y = A\overline{B}$

Teoremas de DeMorgan

- 17 . $\overline{(x + y)} = \bar{x} . \bar{y}$
- 18 . $\overline{(x . y)} = \bar{x} + \bar{y}$

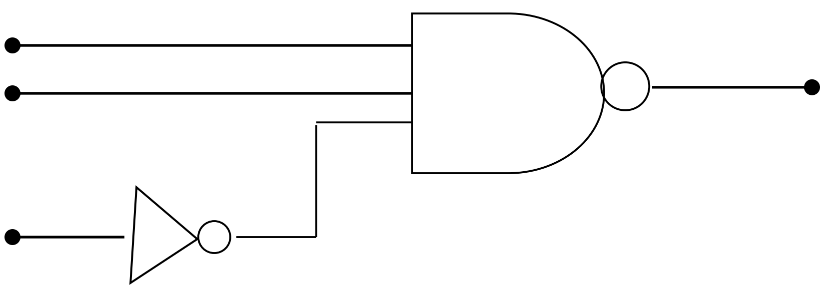
exemplos

- Simplifique a expressão $z = \overline{(\bar{A}+C).(B+\bar{D})}$

- Usando o teorema 18 podemos escrever
- $Z = \overline{(\bar{A}+C)}+\overline{(B+\bar{D})}$ agora usando o 17 temos
- $Z = (\bar{\bar{A}}.\bar{C})+(\bar{B}.\bar{\bar{D}})$
- Logo:
- $Z = A\bar{C} + \bar{B}D$

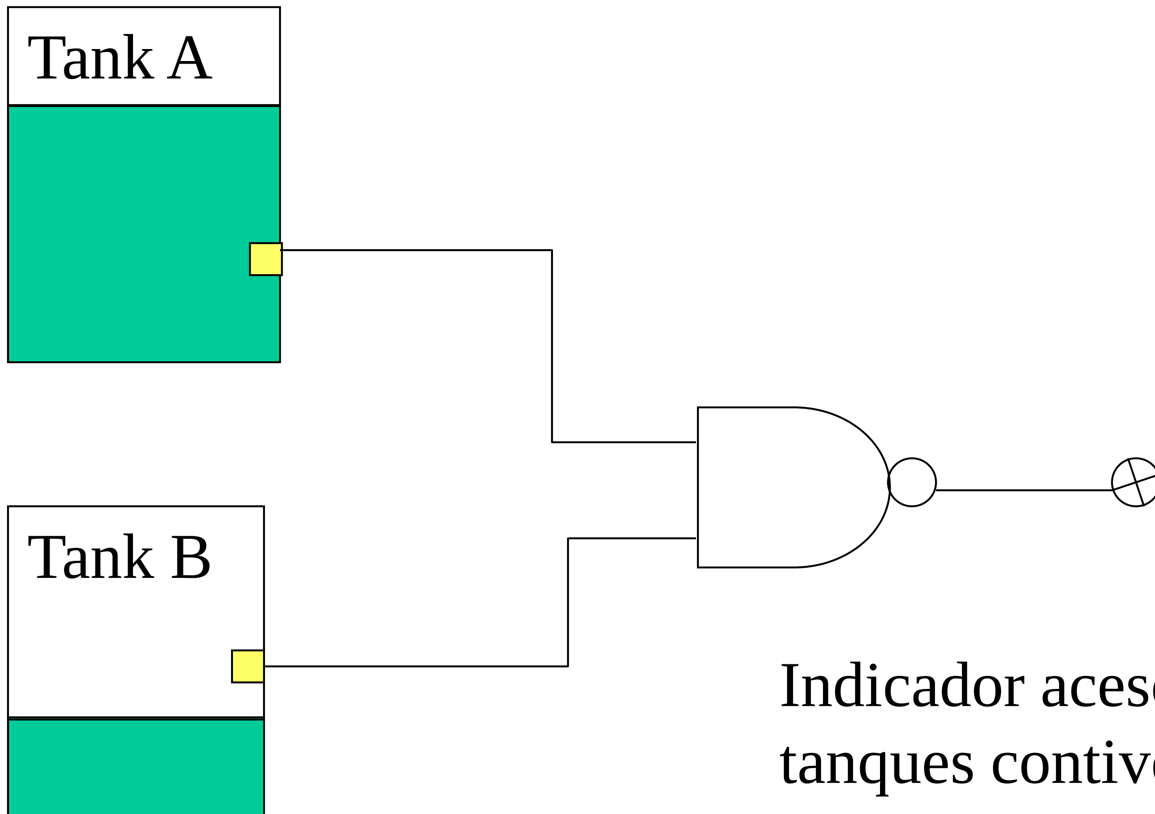
exemplo

- Determine a expressão lógica para a saída do circuito da figura abaixo e simplifique-a usando os teoremas de DeMorgan



Aplicação

- Um engenheiro de alimentos precisa implementar um sistema de supervisão automática conforme a figura. O sistema deve acender um led vermelho quando um dos tanques estiver com menos de ¼ do volume ou os dois estiverem com menos de ¼ do volume. Mostre como o engenheiro de alimentos pode implementar este sistema.



Indicador aceso quando tanques contiverem menos que ¼ da capacidade total