

Arquitetura de Computadores e Sistemas Operacionais

- O que um **bit**?
 - Contração de **binary digit** (dígito binário)
 - **0** ou **1** (em arquitetura) usados para representar informação
 - Menor unidade de informação armazenada eletronicamente
- O que um **byte**?
 - Conjunto de **8 bits**
 - 256 combinações diferentes (2^8 , 0 a 255, 00000000_2 a 11111111_2 , 0x00 a 0xFF)
 - Surgiu por convenção pela necessidade de representar informações como letras do alfabeto (A...Z, a...z), algarismos (0...9), símbolos gráficos (' , @ , - , ! , ? , à , ï , ¬ , \$, ...)

- **Byte**
 - Uma das representações convencionais para estes símbolos é a tabela ASCII
- E uma **palavra**?
 - **word** (em uma arquitetura de um processador)?
 - Quantidade de bits que a CPU processa de uma vez
 - Hoje, a maioria das CPUs processa de 32 a 64 bits

- Obs: notação dos números em diferentes bases

- Em **Python**, imprima os números 127, 1025, 555 nas bases binária, octal e hexadecimal
- Efetue também as somas:
- $362_8 + 75_{16}$
- $2A1_{16} + 1100111_2$
- $48_{10} + 100111_2 + 3C_{16}$

- Por que um **bit**?
 - Representam dois estados nos circuitos lógicos (eletrônicos)
 - Representações frequentes:
 - Posição de chaves (aberta e fechada)
 - Ligado e desligado
 - Valores lógicos (Verdadeiro e Falso) ou '**0**' e '**1**'
 - Tensão elétrica
 - 0 (zero) Volts \rightarrow 0
 - 3,3 Volts ou 5 V \rightarrow 1
 - Também pode ser menor; 1,5V; 1,25V (memória DDR3U)

- Circuitos combinacionais – têm as seguintes propriedades:
 - São circuitos cujos sinais de saída dependem exclusivamente das combinações dos sinais de entrada (ou seja, eles não tem nenhum tipo de memória)
 - Construídos a partir das portas lógicas digitais
 - **Especificação funcional** que determina o valor lógico de cada saída para cada uma das combinações das entradas
 - As portas lógicas implementam as funções dos operadores lógicos (conjunção \wedge , disjunção \vee , negação \neg e seus complementos)
 - Uma ou mais entradas digitais
 - Uma ou mais saídas digitais

- Circuitos combinacionais e a Lógica Matemática

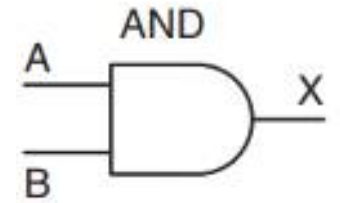
- Os operadores conjunção (\wedge), disjunção (\vee), complemento (\neg) e bicondicional (\leftrightarrow), e as portas lógicas **AND**, **OR**, **NOT** e **XNOR**, respectivamente, executam a mesma operação lógica sobre as variáveis de entrada
- O operador condicional (\rightarrow) pode ser implementado a partir de um circuito combinacional (mais de uma porta lógica)
- As proposições simples correspondem às entradas de um circuito lógico
- Uma proposição composta pode ser entendida como a saída de uma porta ou circuito lógico combinacional

Circuitos Lógicos

- Portas lógicas

- Simulador online de circuitos lógicos: <https://logic.ly/demo/>
- (Exemplos com chaves <http://www.falstad.com/circuit/>)

- Tabela-verdade

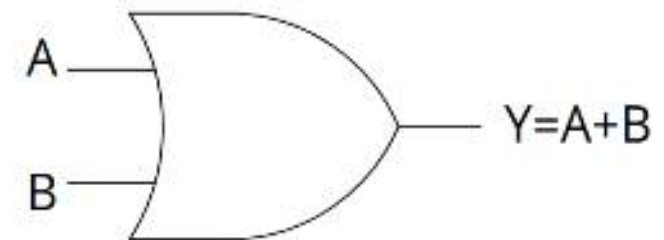


A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Circuitos Lógicos

- OR (OU)
 - Símbolo (+)
 - Simula uma soma binária de um bit
- Tabela-verdade

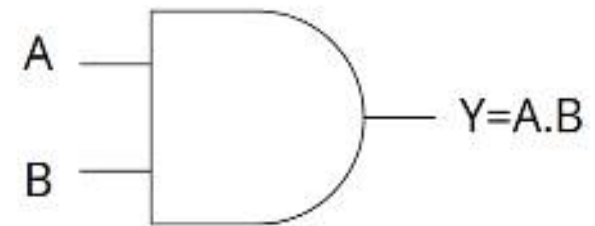
<i>Entradas</i>		<i>Saída</i>
A	B	$Y=A+B$
0	0	0
0	1	1
1	0	1
1	1	1



Circuitos Lógicos

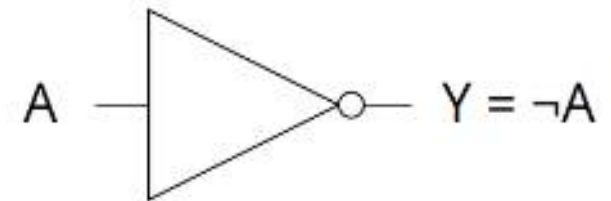
- AND (E)
 - Símbolo (\cdot)
 - Simula uma multiplicação binária de um bit
- Tabela-verdade

A	B	$Y=A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1



- NOT (Inversora/Negação)
 - Símbolo (Barra sobre a variável de entrada \bar{A} , ou $\sim A/\neg A/A'$)
 - Inverte o valor lógico da entrada
- Tabela-verdade

A	$Y = \bar{A} (\sim A/\neg A)$
0	1
1	0

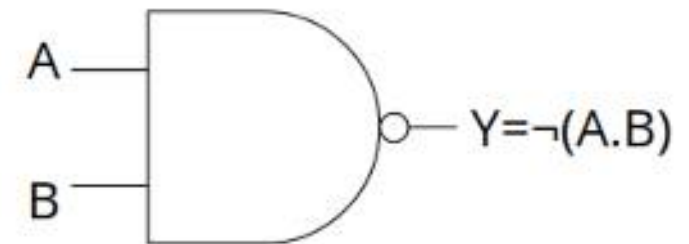


- NAND (E)

- É a negação da operação AND (o resultado da AND é negado, não as entradas)
- Símbolo (barra sobre a operação AND, como exemplo $\overline{A \cdot B}$, ou $\neg(A \cdot B)$)
- ($\overline{A \cdot B}$) é diferente de ($\overline{A} \cdot \overline{B}$), com outra simbologia, $\neg(AB)$ é diferente de ($\neg A \cdot \neg B$)

- Tabela-verdade

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

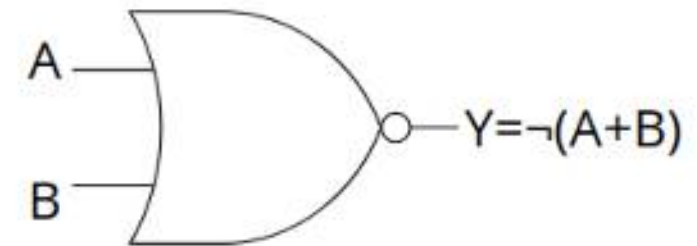


- NOR (Não-OU)

- É a negação da operação OR (o resultado da OR é negado)
- Símbolo (barra sobre a operação OR, como exemplo $\overline{A+B}$, ou $\neg(A+B)$)
- $(\overline{A+B})$ é diferente de $(\overline{A} + \overline{B})$, com outra simbologia, $\neg(A+B)$ é diferente de $(\neg A + \neg B)$

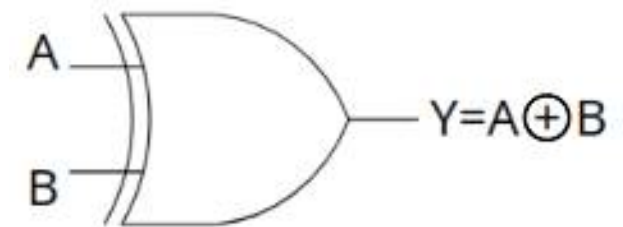
- Tabela-verdade

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



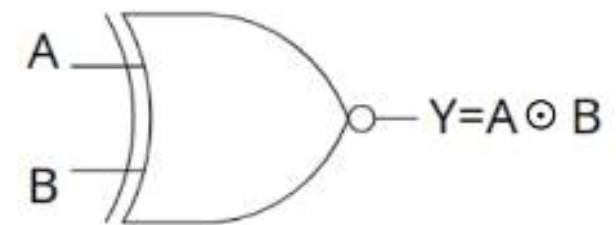
- XOR (OU-eXclusivo)
 - É o mesmo que uma disjunção exclusiva (v)
 - Símbolo (\oplus , como exemplo $A \oplus B$)
- Tabela-verdade

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0






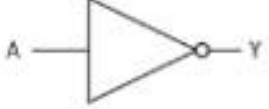

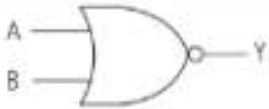
- XNOR (Não-OU-eXclusivo/coincidência)
 - É o mesmo que uma bicondicional
 - Símbolo \odot , como exemplo $A \odot B$
- Tabela-verdade

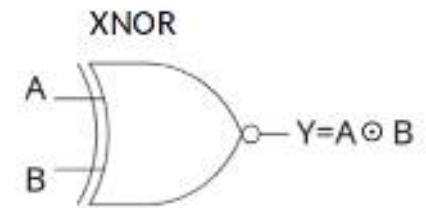
A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0



Circuitos Lógicos

- Resumo

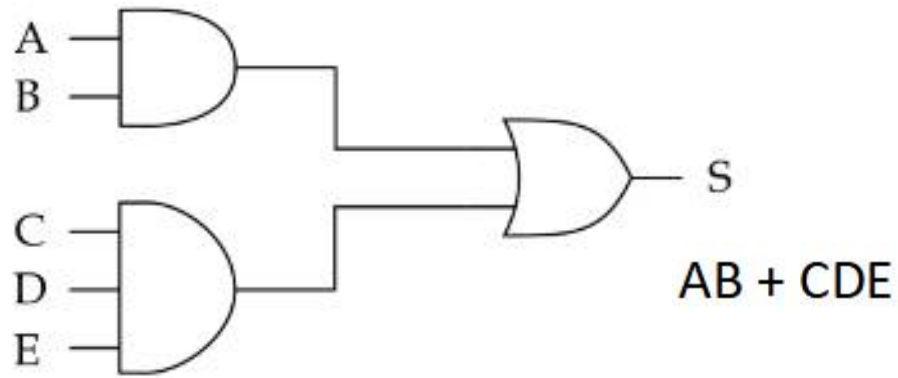
Função Lógica Básica	Símbolo Gráfico da Porta	Equação Booleana
AND		$Y = A.B$
OR		$Y = A+B$
XOR		$Y = A \oplus B$
NOT		$Y = \bar{A}$
NAND		$Y = \overline{A.B}$
NOR		$Y = \overline{A+B}$



Circuitos Lógicos

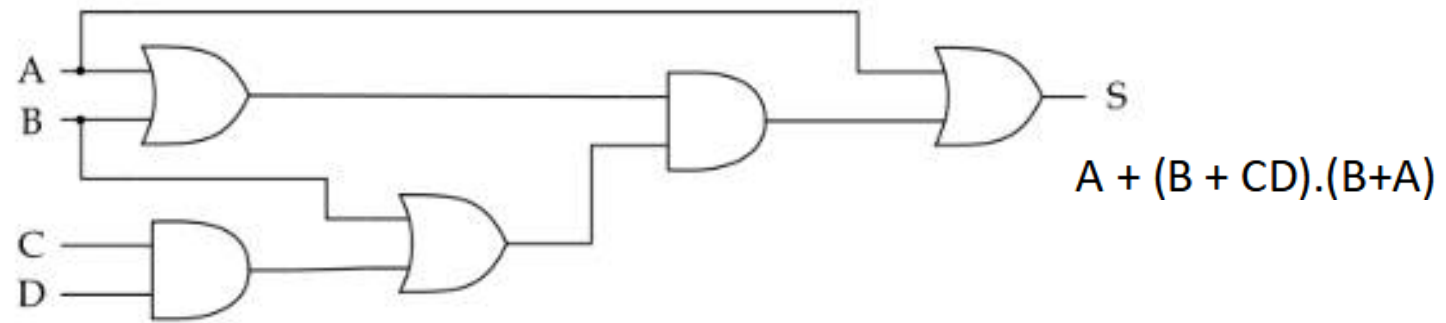
- Exercícios

- Obter as expressões lógicas dos seguintes circuitos:
 - a)



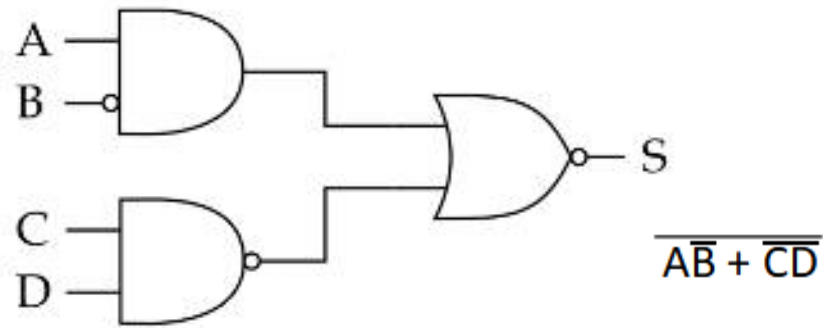
Circuitos Lógicos

- b)



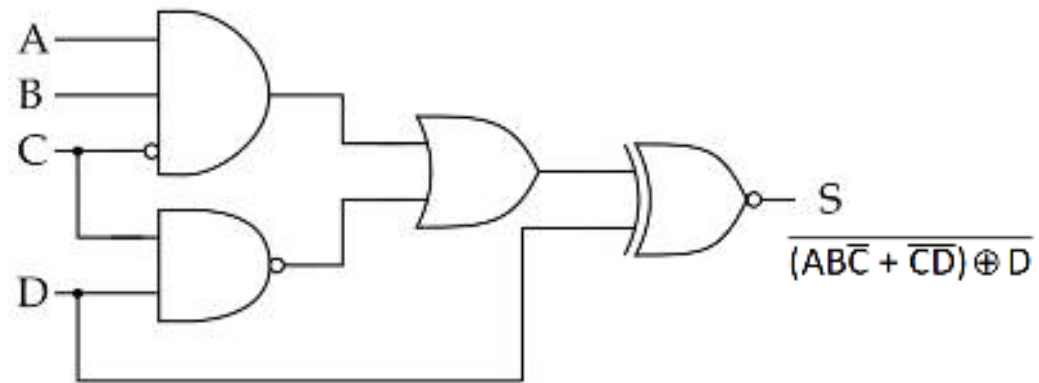
Circuitos Lógicos

- c)



Circuitos Lógicos

- d)





uniesp

Centro Universitário