

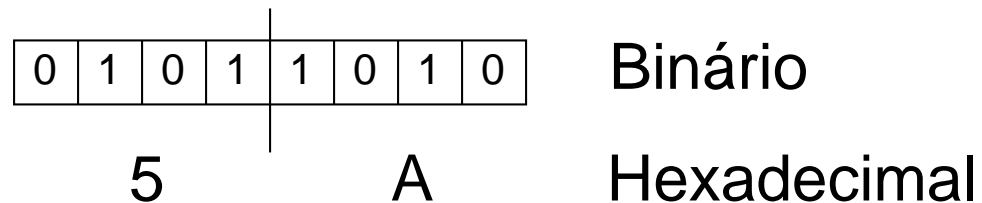
# Revisão de Lógica Digital

Lógica Booleana  
Conversão A/D e D/A  
Circuitos digitais e  
Memória

# Bits e Bytes

Bit: Binary Digit. Unidade elementar de informação: 0 ou 1

Byte = 8 Bits = 256 valores ( $2^8$ )



# Sistema de Numeração:

-Bases

-Casas

-Representação

Notação:

- binária: base 2

- decimal: base 10

- hexadecimal: base 16

Bin	Dec	Hex
0000	00	0
0001	01	1
0010	02	2
0011	03	3
0100	04	4
0101	05	5
0110	06	6
0111	07	7
1000	08	8
1001	09	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

# Sistema Binário

- Bits como número

Bin	Dec
0101	5
<u>0110</u>	<u>6</u> +
1011	11

- Soma binária

- Álgebra Booleana

- George Boole (1815--1864): Matemático e filósofo inglês
- Operadores: E, OU, NÃO (negação)

Operador **E**:  
0 e 0 = 0  
0 e 1 = 0  
1 e 0 = 0  
1 e 1 = 1

Operador **OU**:  
0 ou 0 = 0  
0 ou 1 = 1  
1 ou 0 = 1  
1 ou 1 = 1

## Bits como código:

Codificação dos caracteres em byte.  
Correspondência segundo ASCII.  
(American Standard Code for Information Interchange):

Hex	Name	Meaning	Hex	Name	Meaning
0	NUL	Null	10	DLE	Data Link Escape
1	SOH	Start Of Heading	11	DC1	Device Control 1
2	STX	Start Of Text	12	DC2	Device Control 2
3	ETX	End Of Text	13	DC3	Device Control 3
4	EOT	End Of Transmission	14	DC4	Device Control 4
5	ENQ	Enquiry	15	NAK	Negative Acknowledgement
6	ACK	ACKnowledgement	16	SYN	SYNchronous idle
7	BEL	BELl	17	ETB	End of Transmission Block
8	BS	BackSpace	18	CAN	CANcel
9	HT	Horizontal Tab	19	EM	End of Medium
A	LF	Line Feed	1A	SUB	SUBstitute
B	VT	Vertical Tab	1B	ESC	ESCape
C	FF	Form Feed	1C	FS	File Separator
D	CR	Carriage Return	1D	GS	Group Separator
E	SO	Shift Out	1E	RS	Record Separator
F	SI	Shift In	1F	US	Unit Separator

7 bits + paridade

Hex	Char	Hex	Char	Hex	Char	Hex	Char	Hex	Char
20	(Space)	30	0	40	@	50	P	60	'
21	!	31	1	41	A	51	Q	61	a
22	"	32	2	42	B	52	R	62	b
23	#	33	3	43	C	53	S	63	c
24	\$	34	4	44	D	54	T	64	d
25	%	35	5	45	E	55	U	65	e
26	&	36	6	46	F	56	V	66	f
27	'	37	7	47	G	57	W	67	g
28	(	38	8	48	H	58	X	68	h
29	)	39	9	49	I	59	Y	69	i
2A	*	3A	:	4A	J	5A	Z	6A	j
2B	+	3B	;	4B	K	5B	[	6B	k
2C	,	3C	<	4C	L	5C	\	6C	l
2D	-	3D	=	4D	M	5D	]	6D	m
2E	.	3E	>	4E	N	5E	^	6E	n
2F	/	3F	?	4F	O	5F	_	6F	o
								7F	DEL

Figure 2-41. The ASCII character set.

# Caracteres - ASCII (MS-DOS com código 850 - internacional)

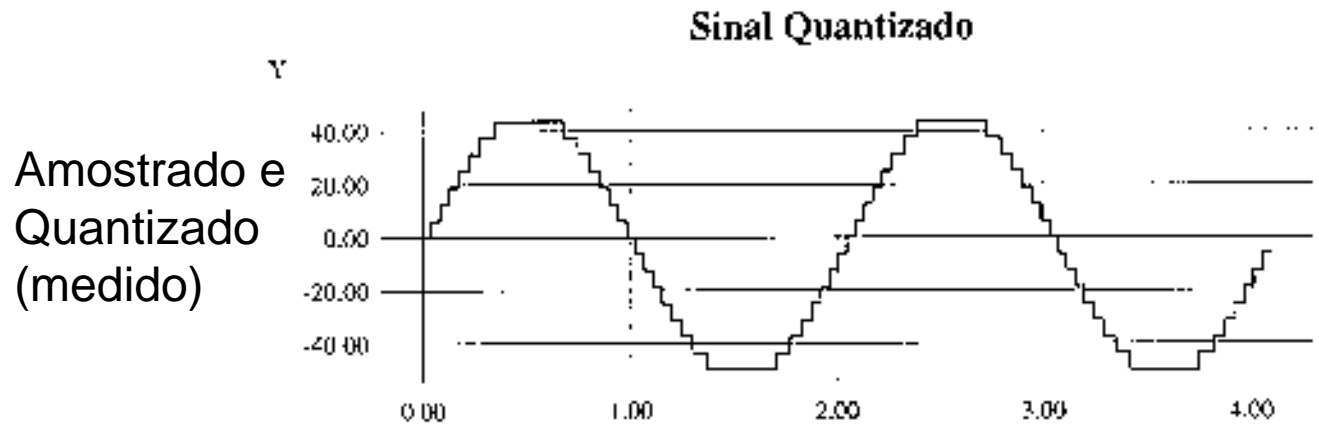
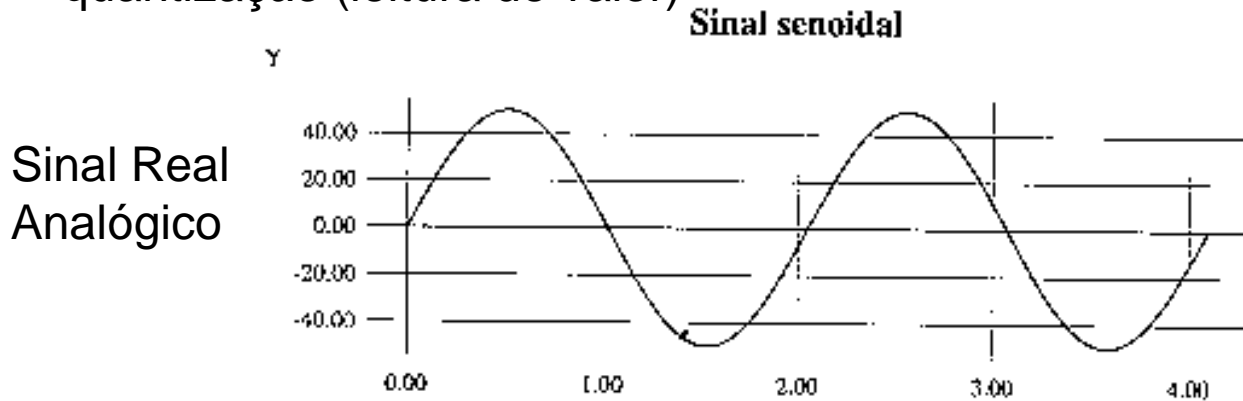
(Primeira coluna código normal em Times Roman e segunda Symbol)

ASCII code page  
Unicode: 2 bytes

000		043	+	+	086	V	V	129	ü	172	¼	<	215	î
001	☺	044	,	,	087	W	W	130	é	173	i	>	216	ï
002	☹	045	-	-	088	X	X	131	â	174	«	fi	217	»
003	♥	046	.	.	089	Y	Y	132	ä	175	»	fl	218	ƒ
004	♦	047	/	/	090	Z	Z	133	å	176			219	█
005	♣	048	0	0	091	[	[	134	ç	177	■	-	220	█
006	♠	049	1	1	092	\	\	135	ê	178	■	-	221	
007	•	050	2	2	093	]	]	136	ë	179		-	222	ì
008	◼	051	3	3	094	^	^	137	è	180		-	223	█
009	○	052	4	4	095	,	,	138	ì	181	À	Æ	224	Ó
010	◐	053	5	5	096	a	a	139	í	182	Á	•	225	Ô
011	◑	054	6	6	097	b	b	140	î	183	Â	•	226	Õ
012	◒	055	7	7	098	c	c	141	ï	184	Ã	•	227	Ö
013	◓	056	8	8	099	d	d	142	Ï	185	Ä	•	228	Ø
014	◔	057	9	9	100	e	e	143	Ï	186	Å	•	229	Ö
015	✳	058	:	:	101	f	f	144	Ê	187	•	•	230	μ
016	◀	059	;	;	102	g	g	145	æ	188	•	•	231	þ
017	▶	060	<	<	103	h	h	146	Æ	189	•	‰	232	þ
018	↕	061	=	=	104	i	i	147	ô	190	•	‰	233	Û
019	!!	062	>	>	105	j	j	148	ö	191	•	‰	234	Ü
020	¶	063	?	?	106	k	k	149	ò	192	•	‰	235	Û
021	§	064	@	@	107	l	l	150	ù	193	•	‰	236	ý
022	—	065	A	A	108	m	m	151	ù	194	•	‰	237	ÿ
023	↕	066	B	B	109	n	n	152	ÿ	195	•	‰	238	-
024	↑	067	C	C	110	o	o	153	Ü	196	•	‰	239	-
025	↓	068	D	D	111	p	p	154	ø	197	•	‰	240	-
026	→	069	E	E	112	q	q	155	ø	198	•	‰	241	±
027	↑	070	F	F	113	r	r	156	£	199	•	‰	242	=
028	↔	071	G	G	114	s	s	157	Ø	200	•	‰	243	¾
029	↔	072	H	H	115	t	t	158	×	201	•	‰	244	¶
030	▲	073	I	I	116	u	u	159	f	202	•	‰	245	§
031	▼	074	J	J	117	v	v	160	á	203	•	‰	246	÷
032		075	K	K	118	w	w	161	í	204	•	‰	247	÷
033	!	076	L	L	119	x	x	162	ó	205	•	‰	248	°
034	"	077	M	M	120	x	x	163	ú	206	•	‰	249	°
035	#	078	N	N	121	y	y	164	ñ	207	•	‰	250	•
036	\$	079	O	O	122	z	z	165	Ñ	208	•	‰	251	•
037	%	080	P	P	123	{	{	166	ª	209	•	‰	252	•
038	&	081	Q	Q	124	}	}	167	º	210	•	‰	253	•
039	'	082	R	R	125	~	~	168	¿	211	•	‰	254	•
040	(	083	S	S	126	◻	◻	169	®	212	•	‰	255	•
041	)	084	T	T	127	◻	◻	170	™	213	•	‰		
042	*	085	U	U	128	◻	◻	171	½	214	•	‰		

# DIGITALIZAÇÃO

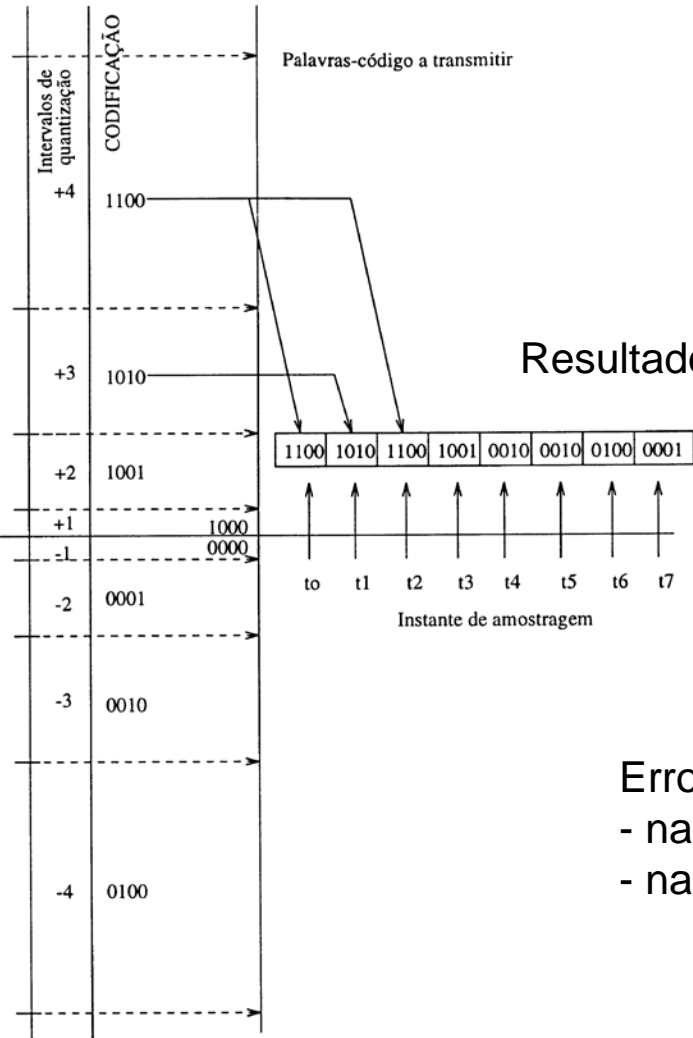
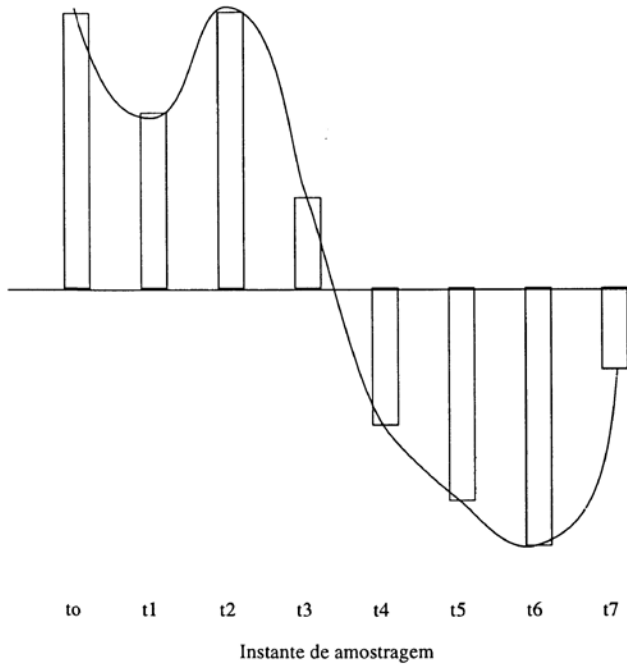
Processo de conversão da informação analógica (real) em valores binários (dígitos). Envolve amostragem(tempo) e quantização (leitura de valor)





# Processo de Digitalização

## Sinal Analógico

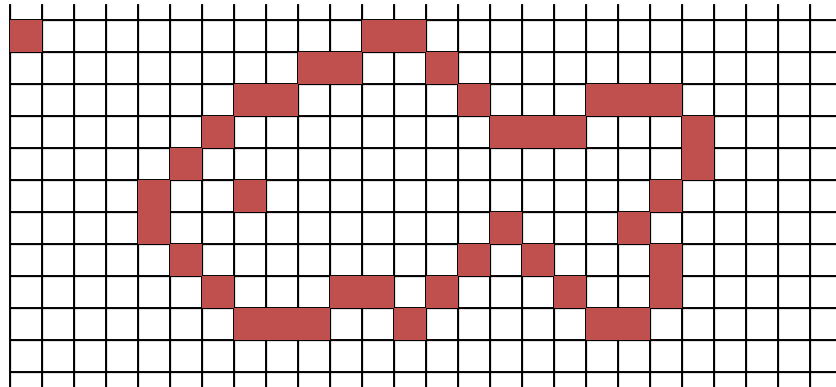


Erros:  
- na amostragem,  
- na quantização

# Digitalização de Imagens por mapeamento em bits (bitmap): Raster

PIXEL: —————→  
um elemento de imagem

Scanners  
Câmeras digitais



Grade ou Matriz

- resolução: número de pixels por unidade de espaço ou área
- impressoras: dpi: dots per inch (pontos por polegada)

VOXEL?

Arquivos de imagem tipo .bmp (bitmap): tratados por programas como Corel, Paintbrush

- cada pixel: um bit (imagem preto e branco)  
um byte (256 tons de cinza)  
dois bytes (64 mil cores, selecionados de uma paleta de cores)  
três bytes (um byte para cada cor), etc

# Bitmap em cores



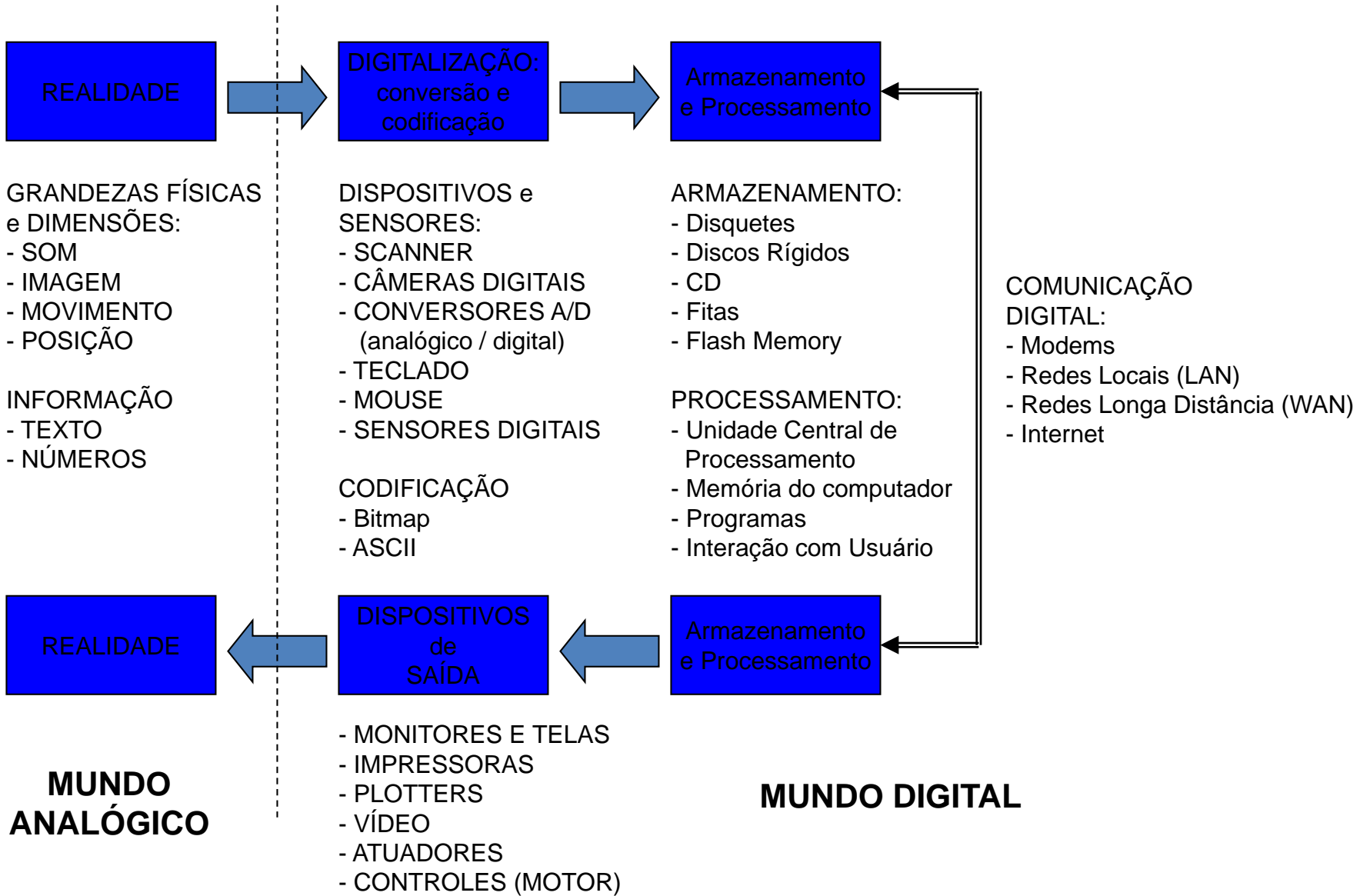
320 x 200 x 16,7 milhões de cores (24 bits)



## BITS & BYTES:

Bits (0 e 1) podem ser:

- números;
- codificados para representar qualquer símbolo (character);
- sinais (após digitalização), som;
- imagens, vídeos;
- podem representar qualquer informação.



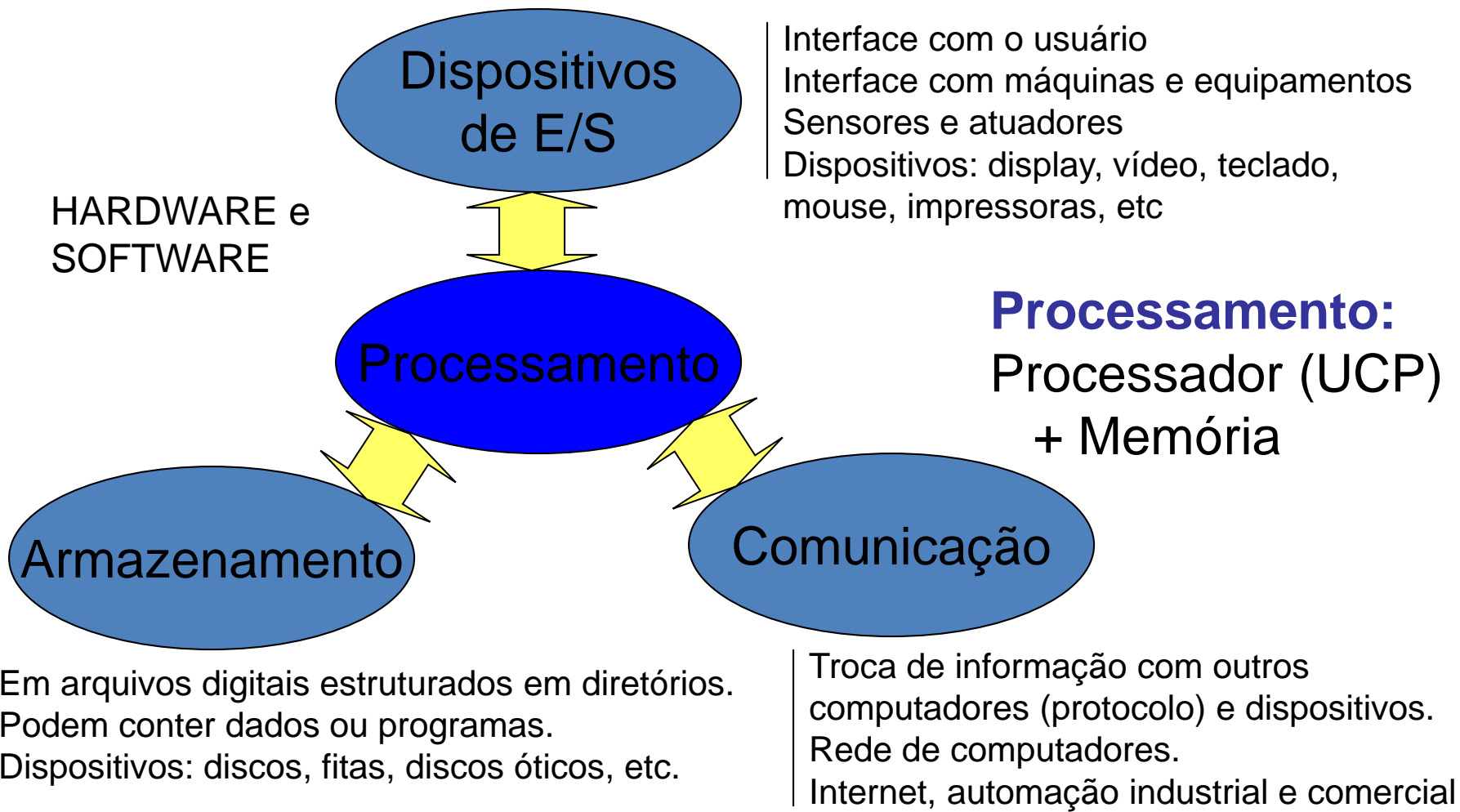
## DIGITALIZAR:

converter informações reais para o formato binário (digital)

## VANTAGENS DA DIGITALIZAÇÃO:

- Uma vez convertidas em informação binária, torna-se mais fácil e robusto o armazenamento e a preservação da informação. Uma cópia de informação binária é uma cópia fiel e idêntica do original.
- As informações no formato digital podem ser manipuladas e tratadas por programas e processadores, possibilitando uma extensa gama de aplicações e uso.

# Estrutura dos Computadores Digitais



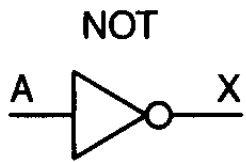
# Revisão Circuitos Digitais



# PORTAS LÓGICAS E SUAS FUNÇÕES

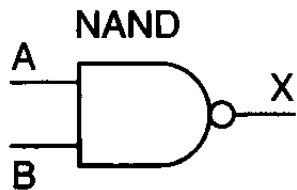
Portas Lógicas:

- elementos básicos na implementação de circuitos digitais
- álgebra booleana: **George Boole** (1815-1864): Matemático e filósofo inglês Criador da álgebra booleana, que possibilitou o vínculo da lógica (filosofia) com a matemática.
- tabela da verdade



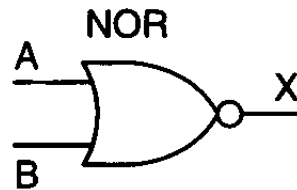
A	X
0	1
1	0

(a)



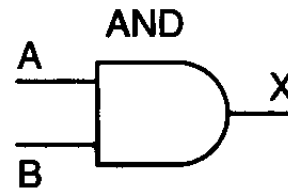
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

(b)



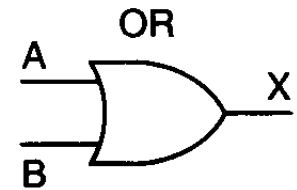
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

(c)



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

(d)



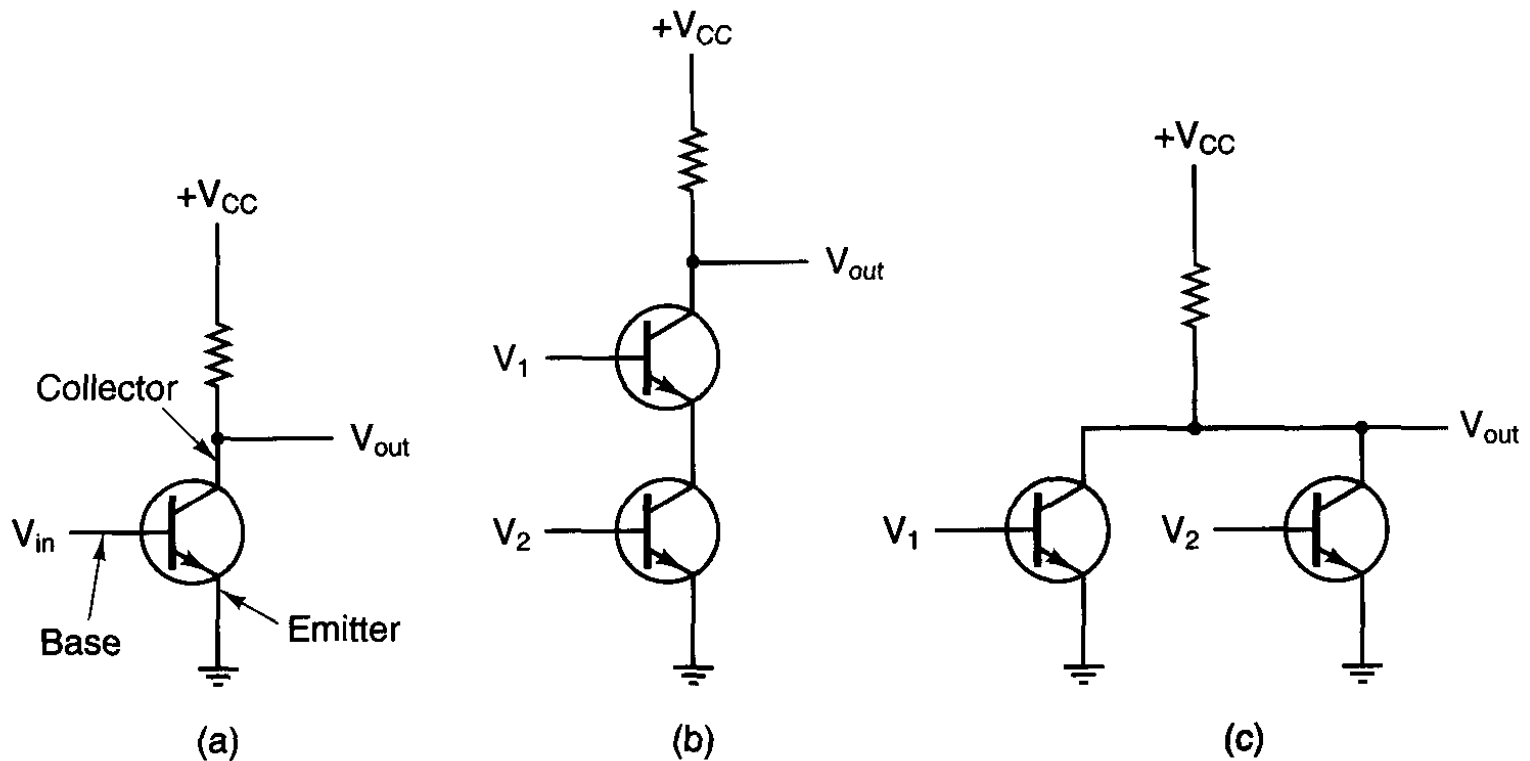
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(e)

The symbols and functional behavior for the five basic gates.

# PORTAS LÓGICAS E O TRANSISTOR

Implementação das portas lógicas por transistores



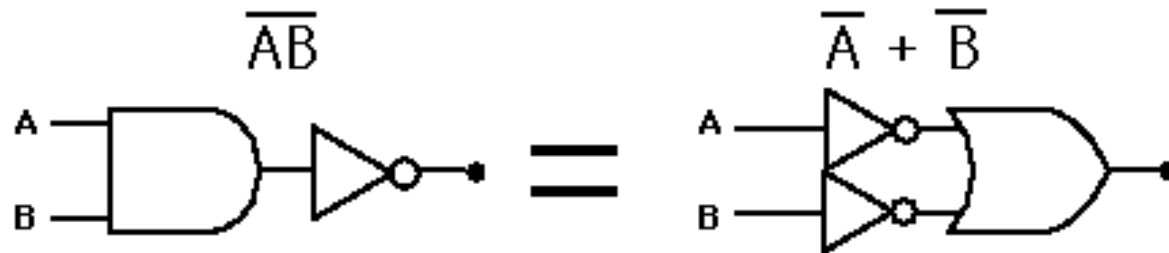
(a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.

## TEOREMA DE DEMORGAN

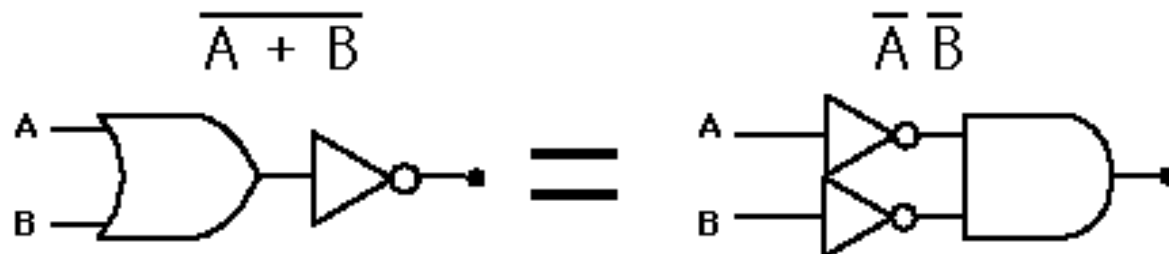
$$\overline{A + B} = \bar{A}\bar{B} \quad \text{E} \quad \overline{AB} = \bar{A} + \bar{B}$$

## TEOREMA DE DEMORGAN

$$\overline{A + B} = \bar{A}\bar{B} \quad \text{E} \quad \overline{AB} = \bar{A} + \bar{B}$$

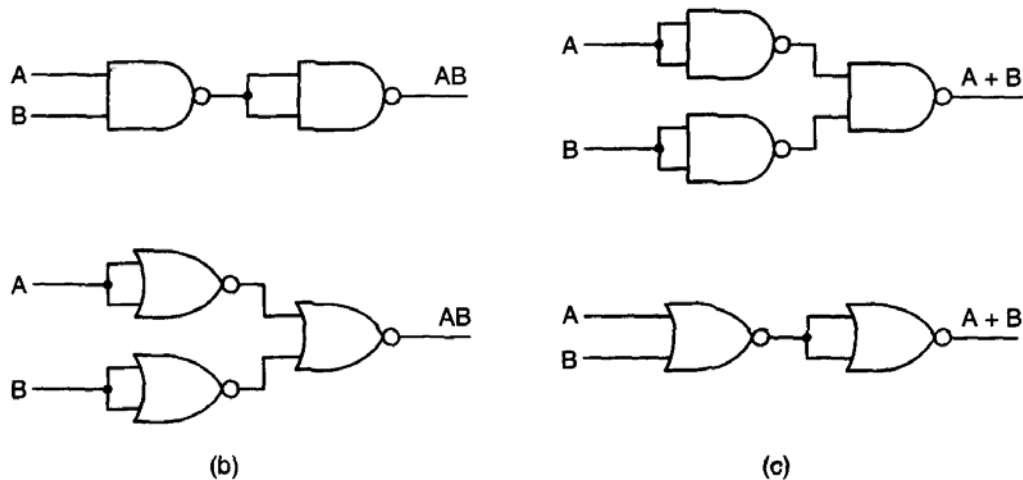
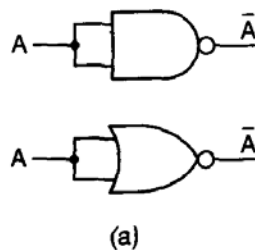


A NAND gate is equivalent to an inversion followed by an OR



A NOR gate is equivalent to an inversion followed by an AND

# CONSTRUÇÕES UTILIZANDO SOMENTE NAND OU NOR

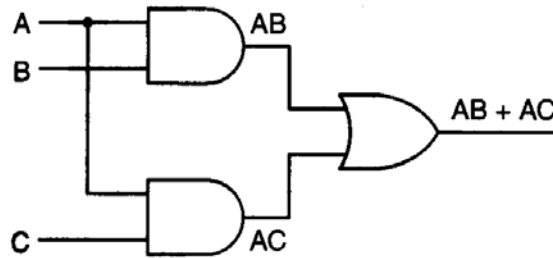


Construction of (a) NOT, (b) AND, and (c) OR gates using only NAND gates or only NOR gates.

Vantagem: utiliza um único tipo de circuito transistorizado

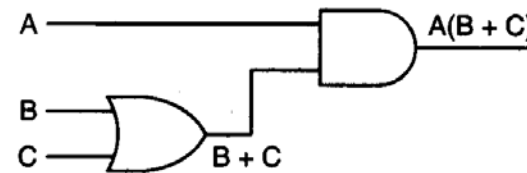
# ÁLGEBRA BOOLEANA: EQUIVALÊNCIA DE EQUAÇÕES

Álgebra Booleana: exemplo de equivalência



A	B	C	AB	AC	AB + AC
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

(a)

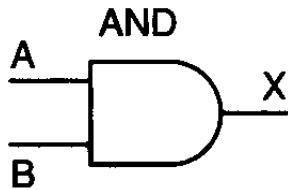


A	B	C	A	B + C	A(B + C)
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	1
1	1	1	1	1	1

(b)

Two equivalent functions. (a)  $AB + AC$ . (b)  $A(B + C)$ .

# EQUIVALÊNCIA ELETRÔNICA COM A LÓGICA



- (a): Implementação AND eletrônica
- (b): Equivalente lógico positivo
- (c): Equivalente lógico negativo

A	B	F
0 <sup>V</sup>	0 <sup>V</sup>	0 <sup>V</sup>
0 <sup>V</sup>	5 <sup>V</sup>	0 <sup>V</sup>
5 <sup>V</sup>	0 <sup>V</sup>	0 <sup>V</sup>
5 <sup>V</sup>	5 <sup>V</sup>	5 <sup>V</sup>

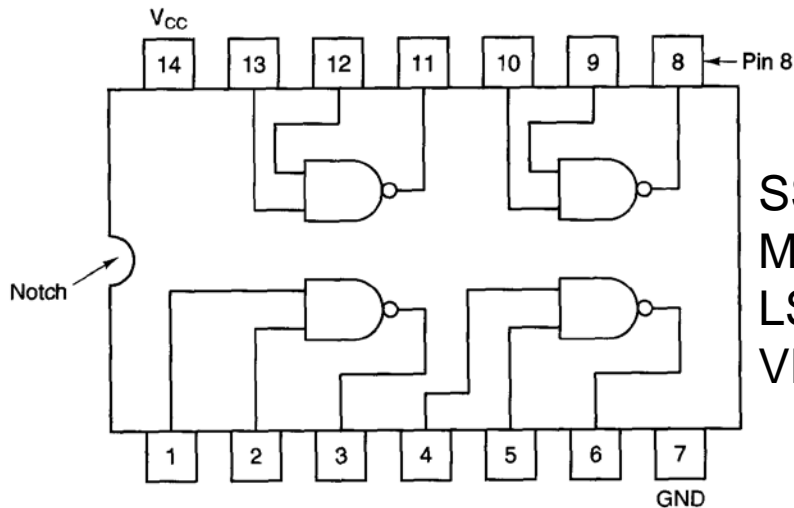
(a)

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

(b)

A	B	F
1	1	1
1	0	1
0	1	1
0	0	0

(c)



An SSI chip containing four gates.

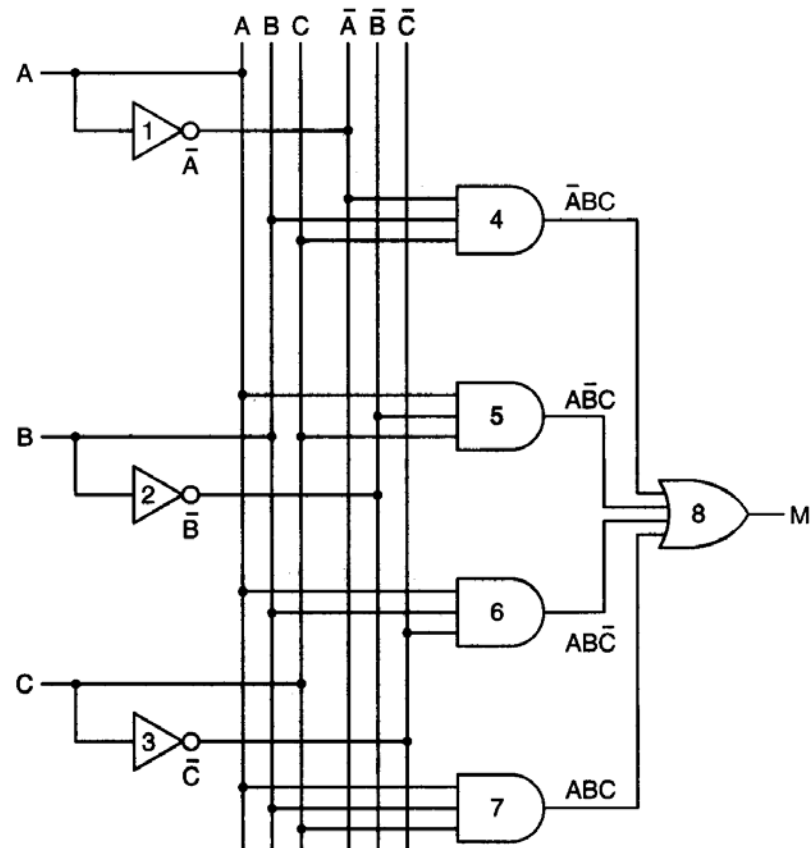
- SSI: Small Scale Integrated: 1 a 10 portas
- MSI: Medium Scale Integrated: 10 a 100 portas
- LSI: Large Scale Integrated: 100 a 100 mil portas
- VLSI: Very Large Scale Integrated: >100 mil portas

# EXERCÍCIO DE ANÁLISE DE UM CIRCUITO LÓGICO

Identificar a função realizada pelo circuito lógico

A	B	C	M
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

(a)



(b)

(a) The truth table for the majority function of three variables.

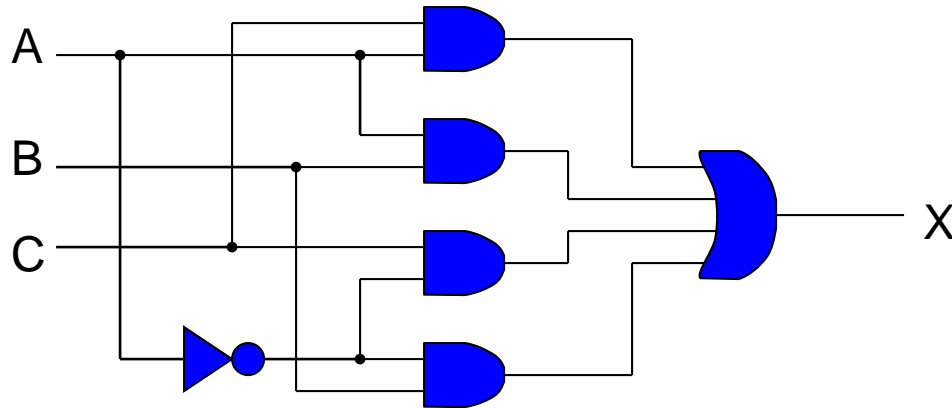
(b) A circuit for (a).



## ÁLGEBRA BOOLEANA: NOTAÇÃO E PROPRIEDADES

Name	AND form	OR form
Identity law	$1A = A$	$0 + A = A$
Null law	$0A = 0$	$1 + A = 1$
Idempotent law	$AA = A$	$A + A = A$
Inverse law	$A\bar{A} = 0$	$A + \bar{A} = 1$
Commutative law	$AB = BA$	$A + B = B + A$
Associative law	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Distributive law	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$

## Exercício de Lógica Booleana



Dado e esquema lógico acima:

- elaborar a Tabela Verdade;
- escrever a equação booleana;
- fatorar / simplificar a equação booleana;
- elaborar novo esquema lógico a partir da equação simplificada
- conferir a Tabela Verdade

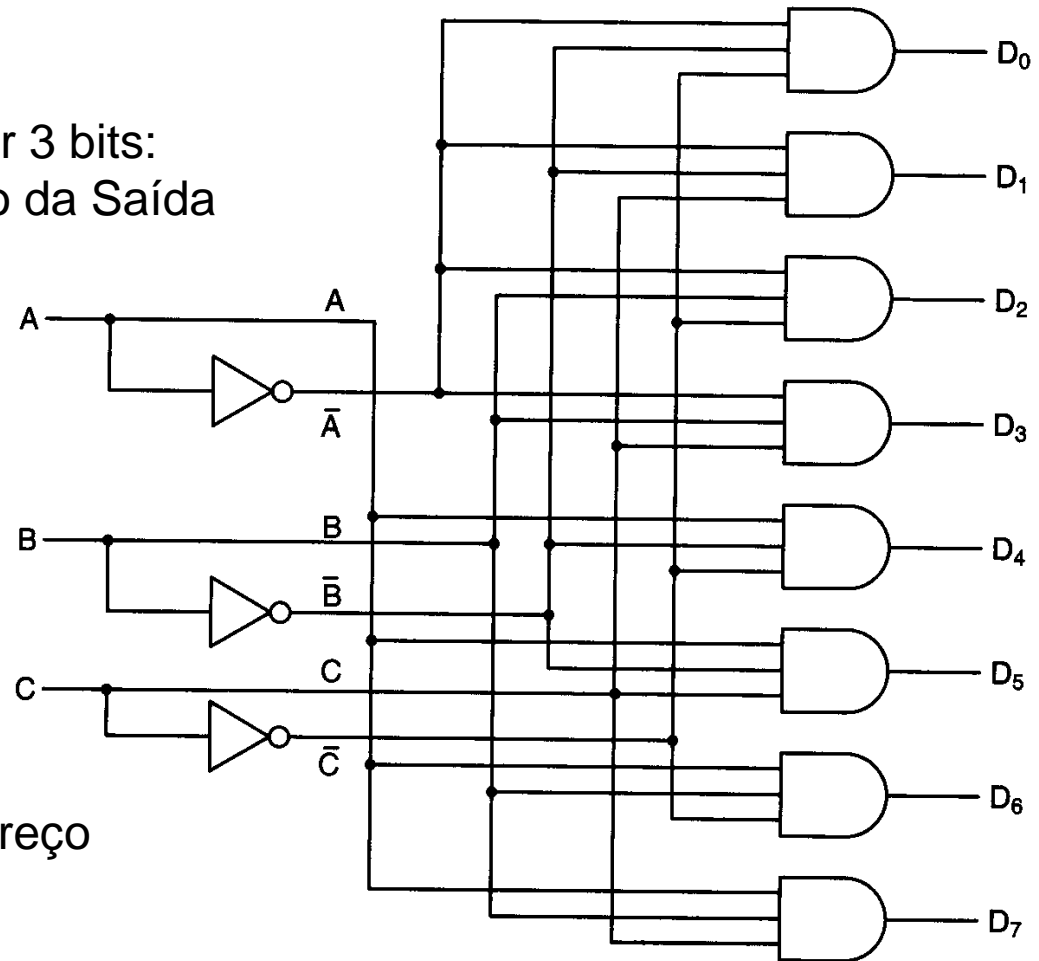
# ULA: Unidade Lógica e Aritmética

# DECODIFICAÇÃO OU DEMUTIPLEXADOR

Exemplo: Decodificador 3 bits:  
Capacidade de seleção da Saída

A	B	C	Saídas
0	0	0	D0 = 1
0	0	1	D1 = 1
...	...	...	
1	1	1	D7 = 1

Decodificação de endereço  
de memória



A 3-to-8 decoder circuit.

# IMPLEMENTAÇÕES DO XOR (OU EXCLUSIVO)

Importante na implementação do Somador Binário

Símbolo XOR

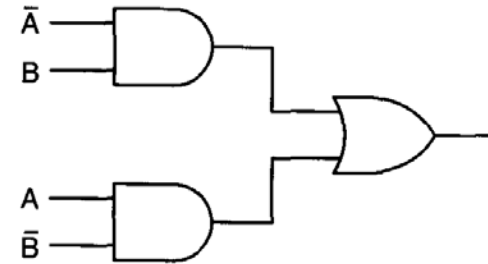


Bin	Dec	
0101	5	A
0110	6 +	B
1011	11	

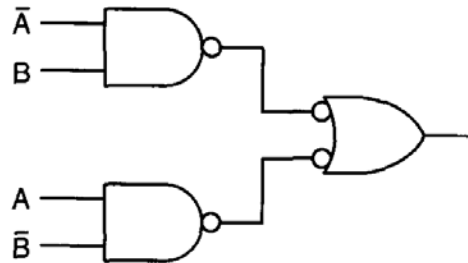
OU  
exclusivo  
(Observem a  
operação!)

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

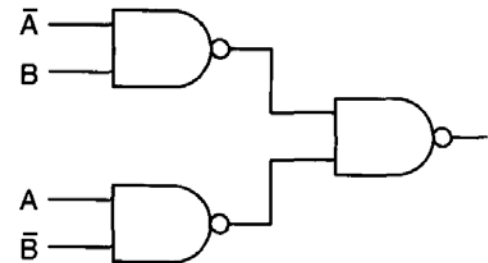
(a)



(b)



(c)



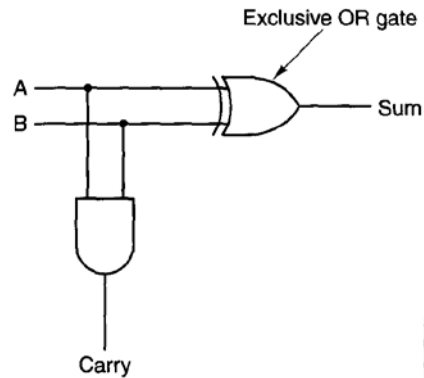
(d)

Possíveis implementações de XOR (verificar)

# IMPLEMENTAÇÃO DE ALGUMAS OPERAÇÕES DA ULA

## Operação Somador Binário

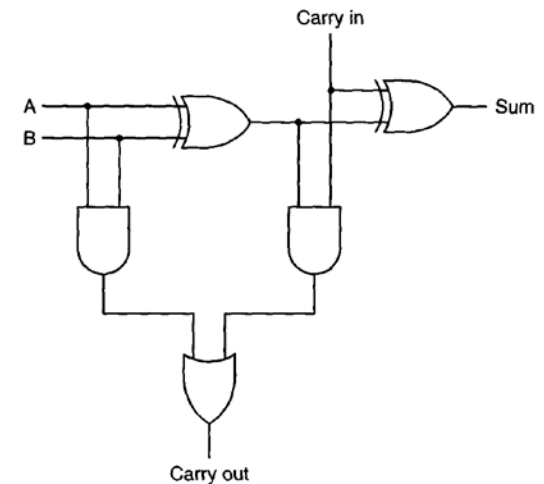
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Sem Carry (vai-um)

Completo:  
Com Carry (vai-um)

A	B	Carry in	Sum	Carry out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

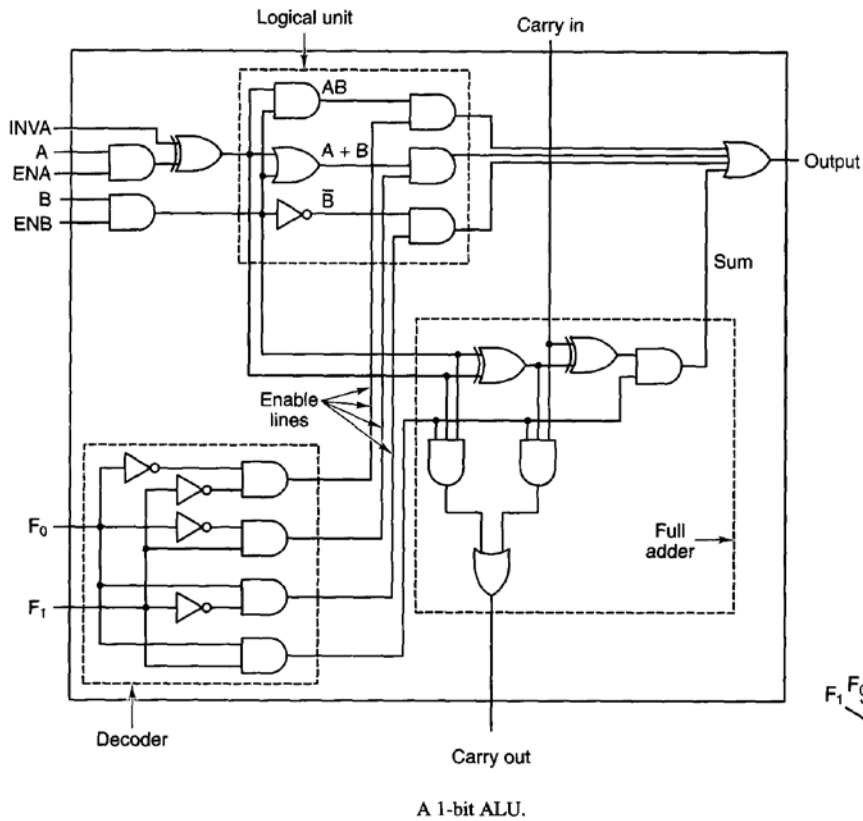


(a)

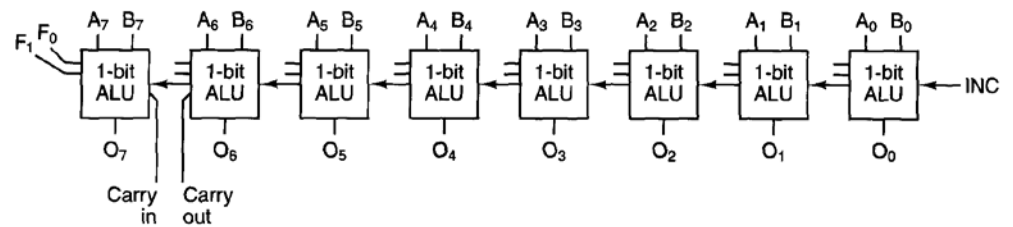
(b)

(a) Truth table for full adder. (b) Circuit for a full adder.

# IMPLEMENTAÇÃO DE UM BIT DA ULA, COM ALGUMAS OPERAÇÕES



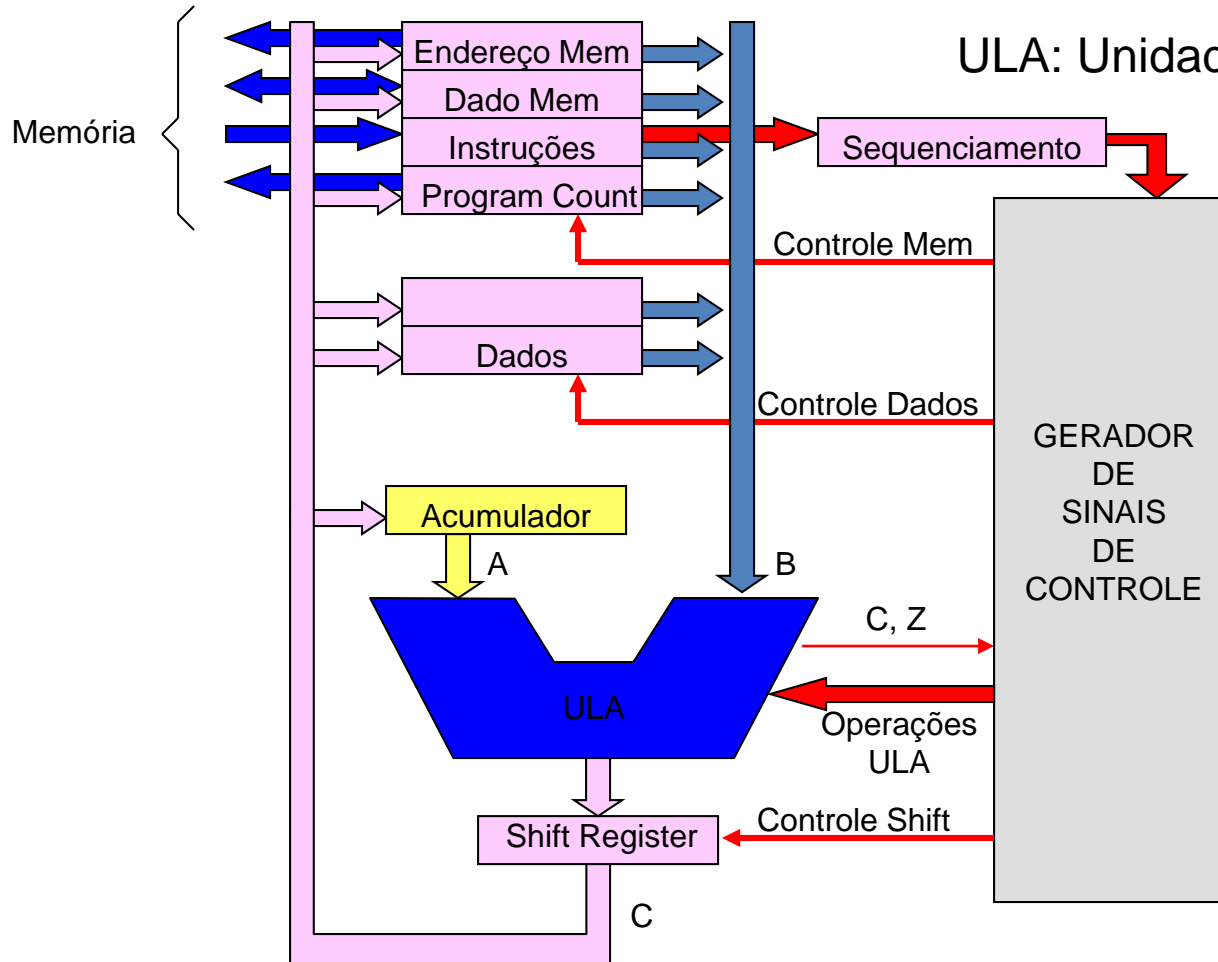
# IMPLEMENTAÇÃO DE ULA DE 8 BITS



Eight 1-bit ALU slices connected to make an 8-bit ALU. The enables and invert signals are not shown for simplicity.

# PROCESSAMENTO E A ULA

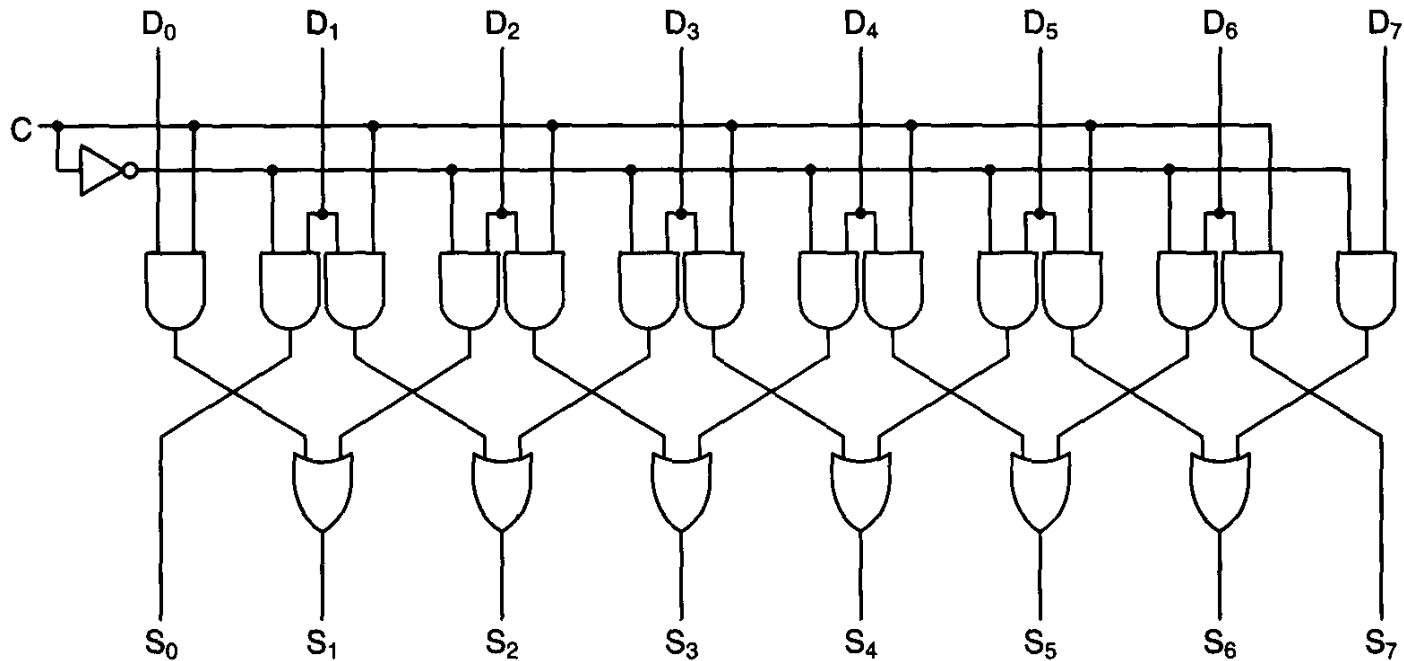
Estrutura Interna do Processador:  
Típica von Neumann  
ULA: Unidade Lógica Aritmética





# IMPLEMENTAÇÃO DE ALGUMAS OPERAÇÕES DA ULA

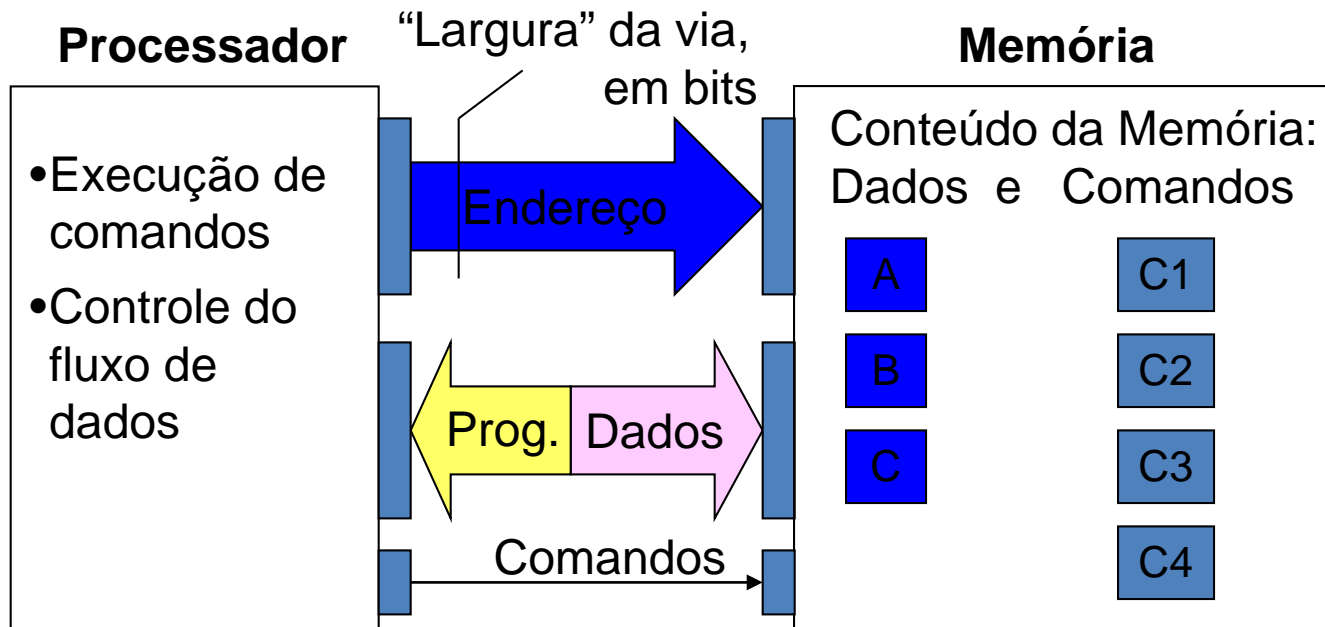
Deslocamento e Rotação (Shift) à esquerda e à direita



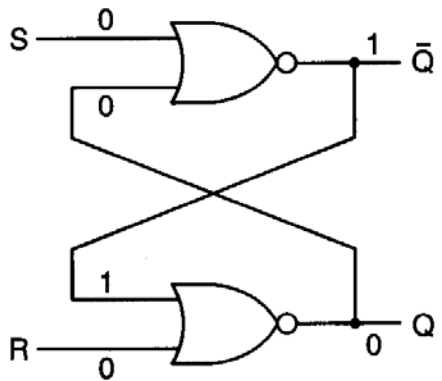
A 1-bit left/right shifter.

**MEMÓRIA**

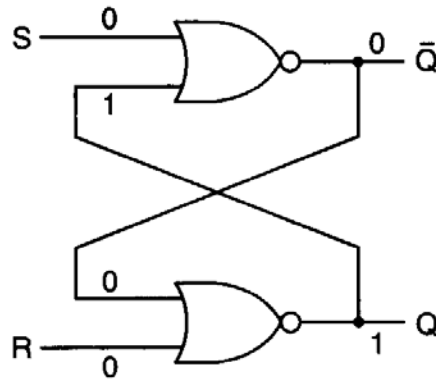
# PROCESSAMENTO E A MEMÓRIA



# IMPLEMENTAÇÃO DA MEMÓRIA: LATCHES



(a)

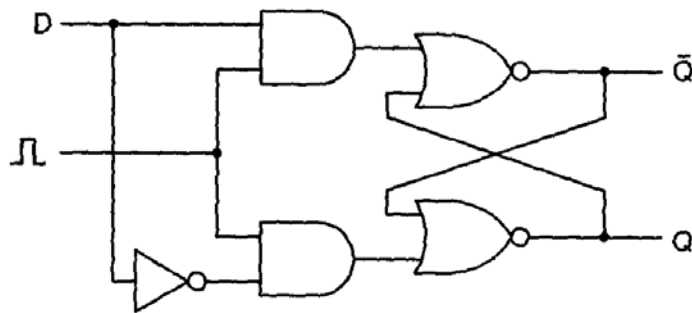


(b)

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

(c)

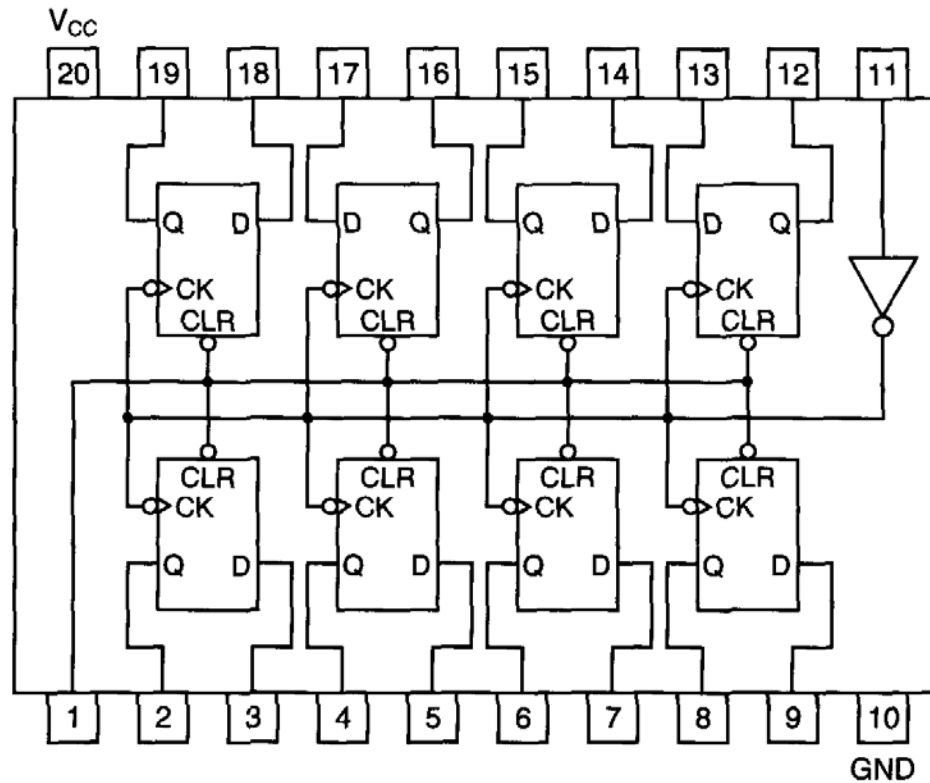
(a) NOR latch in state 0. (b) NOR latch in state 1. (c) Truth table for NOR.



A clocked D latch.

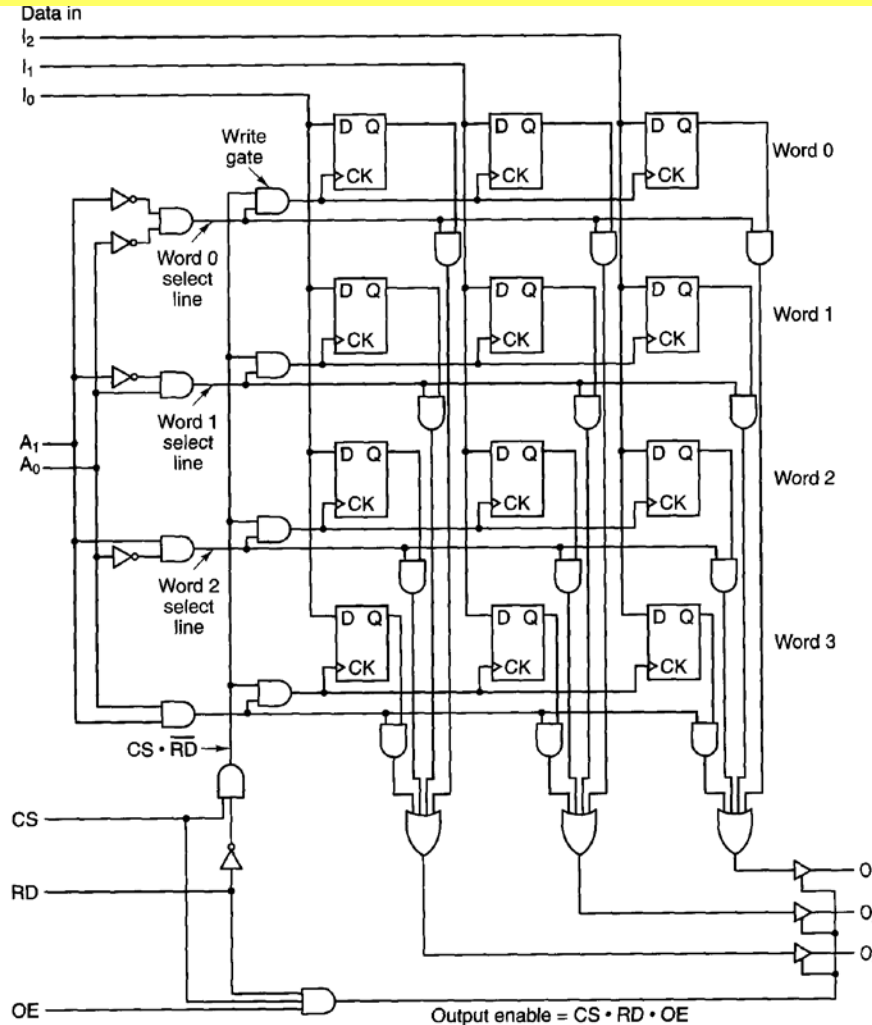
Flip-flop, com pulso de relógio (clock)

# IMPLEMENTAÇÃO DA MEMÓRIA: EXEMPLO DE CI ELEMENTAR



Octal flip-flop

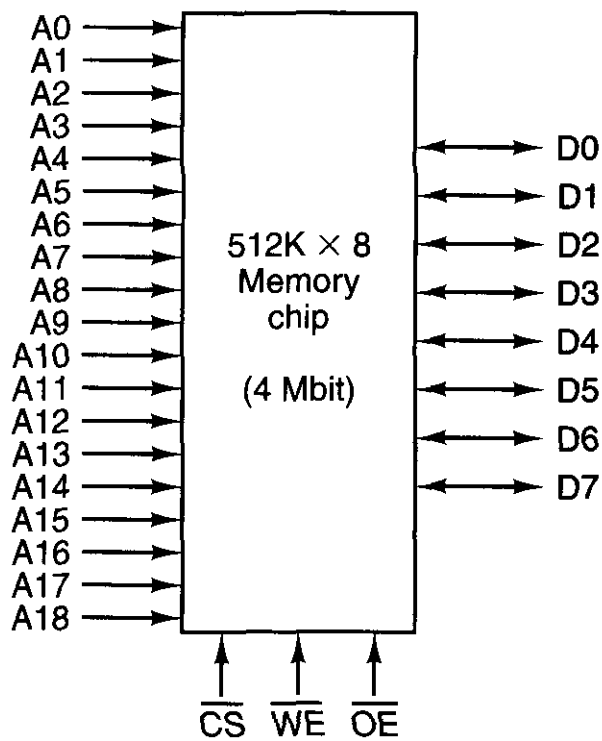
# IMPLEMENTAÇÃO DA MEMÓRIA: EXEMPLO 4 X 3



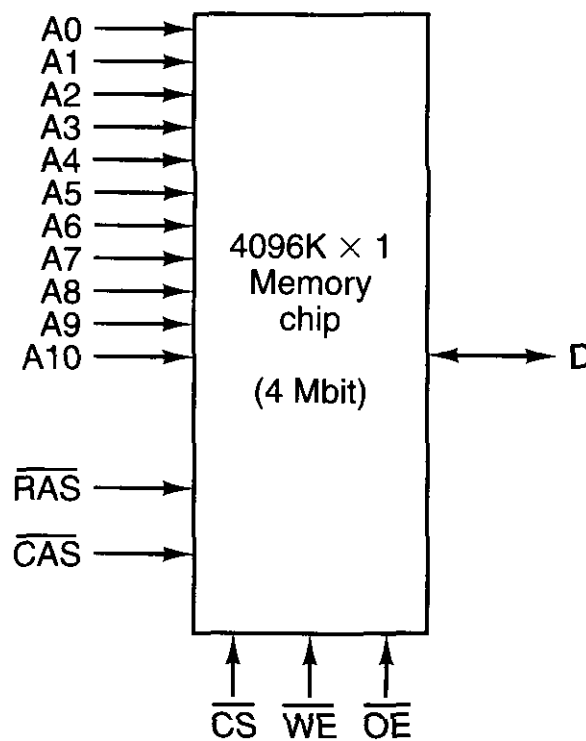
- Memória contendo:
- 4 palavras (words)
  - 3 bits por palavra
  - endereços:  $A_0$  e  $A_1$
  - sinais de controle

Logic diagram for a  $4 \times 3$  memory. Each row is one of the four 3-bit words. A read or write operation always reads or writes a complete word.

# ORGANIZAÇÃO DE UM CI DE MEMÓRIA REAL DE 4 MEGA



(a)



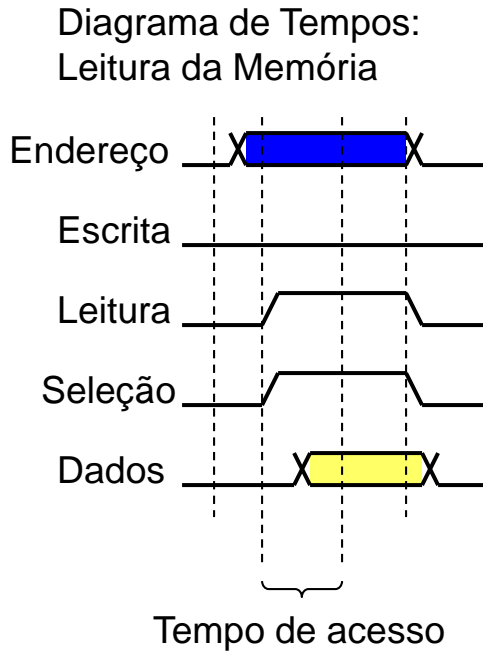
(b)

Two ways of organizing a 4-Mbit memory chip.

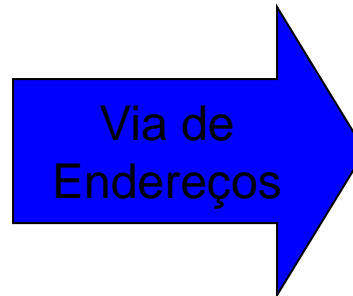
# ACESSOS AO CI DE MEMÓRIA

## EXEMPLO DE MEMÓRIA DE 64K x 8 bits

## Memória RAM



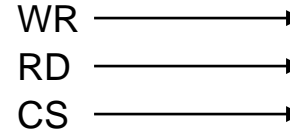
Barramento ou "Bus"  
de Endereço de 16 bits



Barramento ou "Bus"  
de Dados de 8 bits



Sinais de  
Comandos



64K x 8 bits  
512K bits

WR: Escrita (write)

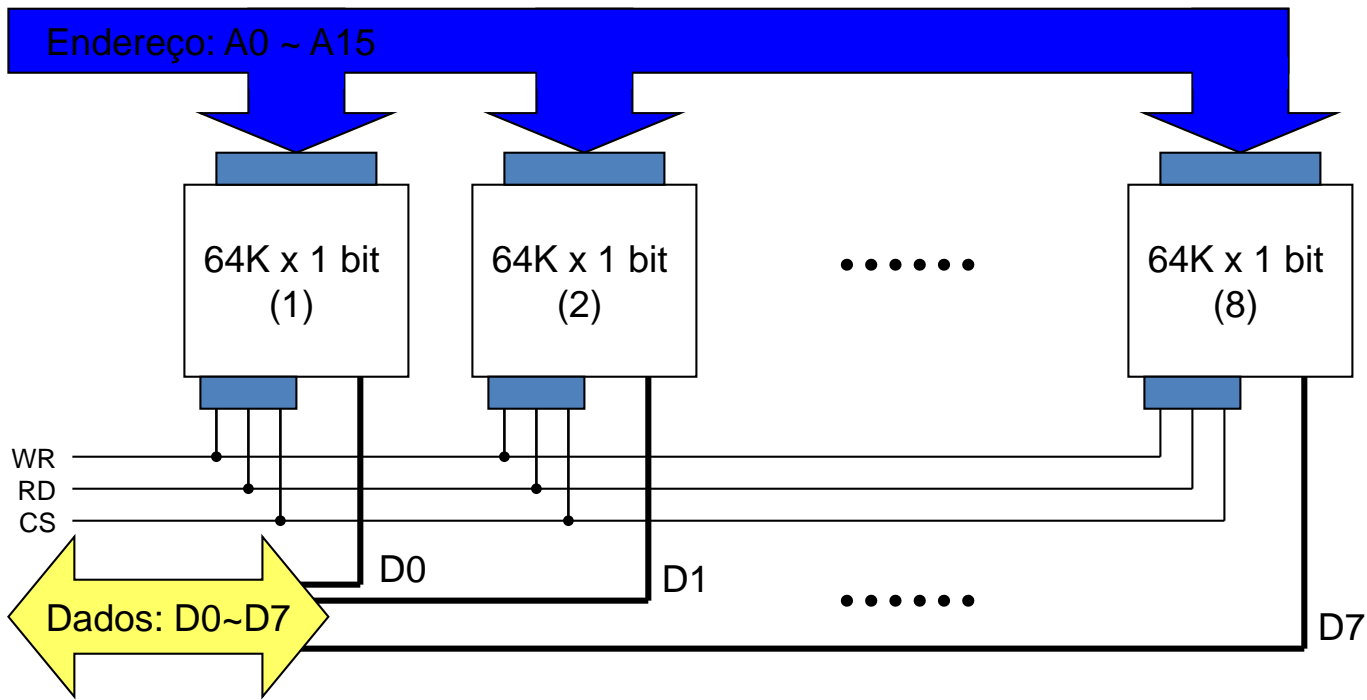
RD: Leitura (read)

CS: Seleção (chip select)



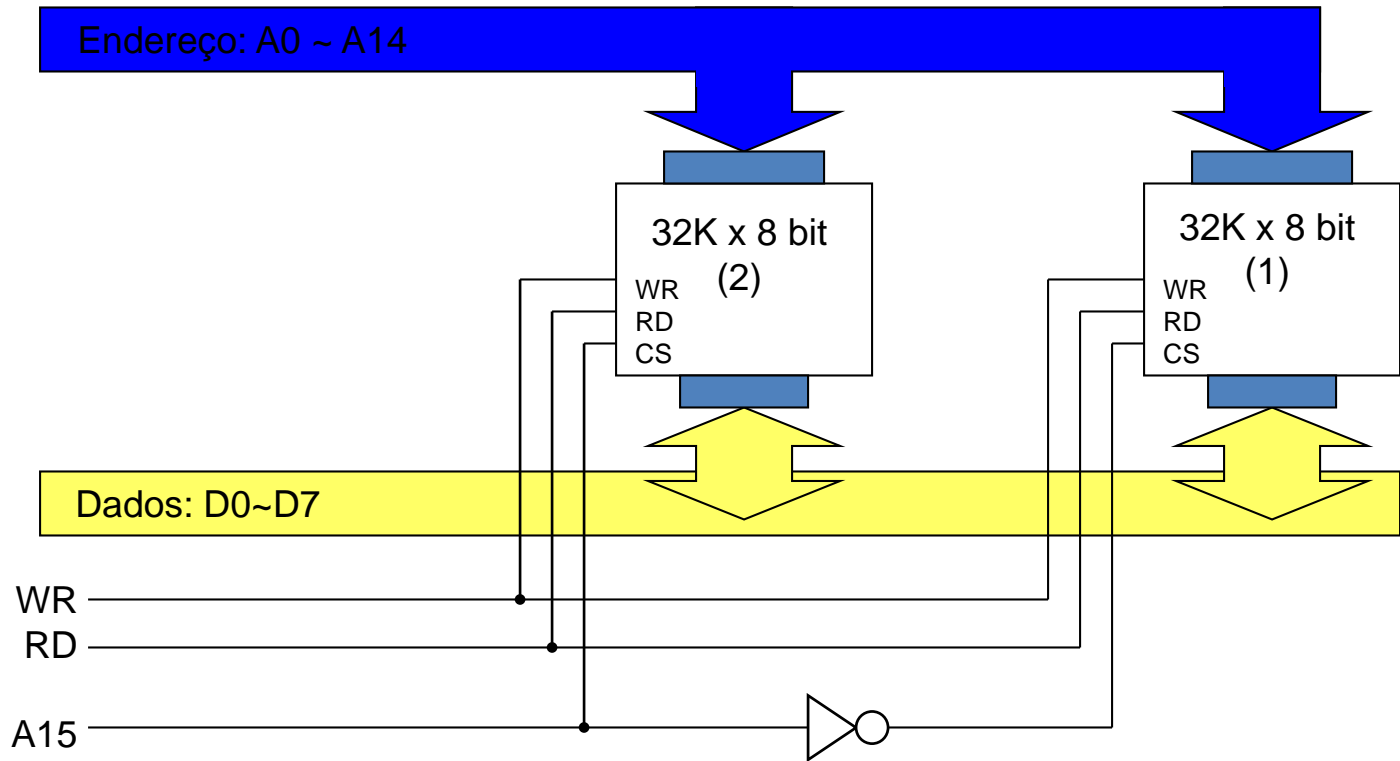
# CONFIGURAÇÃO PARA MÚLTIPLOS CHIPS (1)

EXEMPLO PARA 8 CHIPS DE 64K x 1 bit: TOTAL 64K Bytes



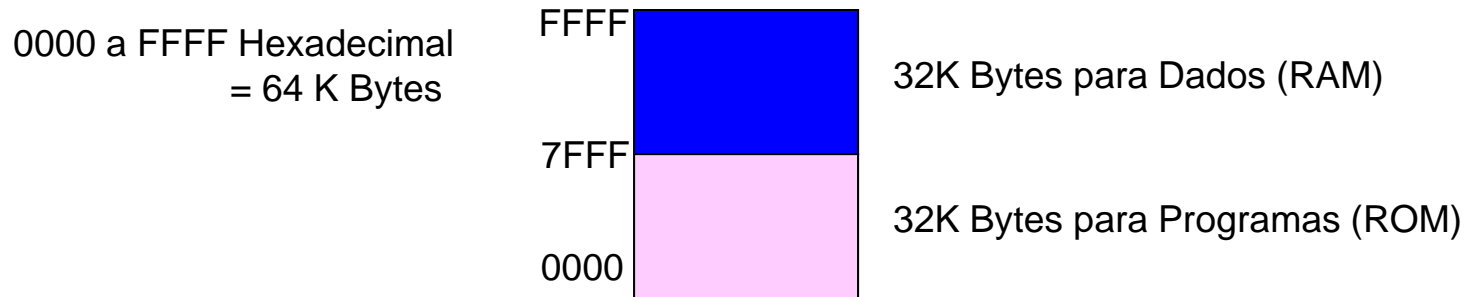
# CONFIGURAÇÃO PARA MÚLTIPLOS CHIPS (2)

EXEMPLO PARA 2 CHs DE 32K x 8 bit: TOTAL 64K Bytes

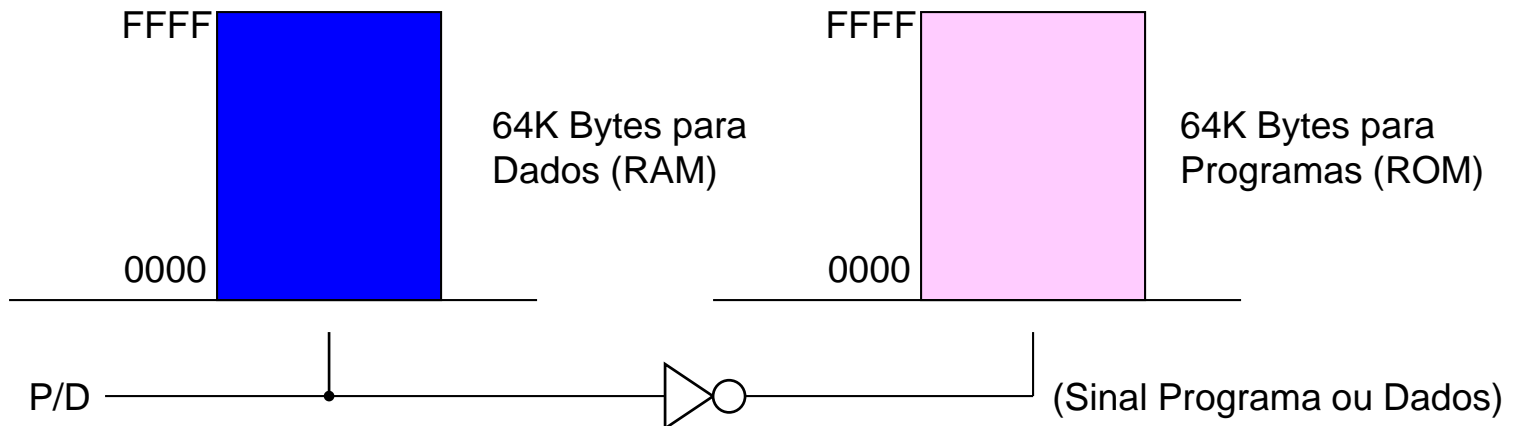


# MAPA DE MEMÓRIA

## EXEMPLO PARA ÁREA DE MEMÓRIA DE 64K Bytes: DIVISÃO EM PROGRAMAS E DADOS



## EXEMPLO PARA ÁREA DE MEMÓRIA DE 64K Bytes: ÁREAS DE PROGRAMAS E DADOS



## TIPOS DE MEMÓRIA

Type	Category	Erasure	Byte alterable	Volatile	Typical use
SRAM	Read/write	Electrical	Yes	Yes	Level 2 cache
DRAM	Read/write	Electrical	Yes	Yes	Main memory
ROM	Read-only	Not possible	No	No	Large volume appliances
PROM	Read-only	Not possible	No	No	Small volume equipment
EPROM	Read-mostly	UV light	No	No	Device prototyping
EEPROM	Read-mostly	Electrical	Yes	No	Device prototyping
Flash	Read/write	Electrical	No	No	Film for digital camera

*A comparison of various memory types.*

## **Bibliografia:**

Tanenbaum, A. S. – **Organização Estruturada de Computadores**. Quinta Edição, Prentice Hall Brasil, 2007