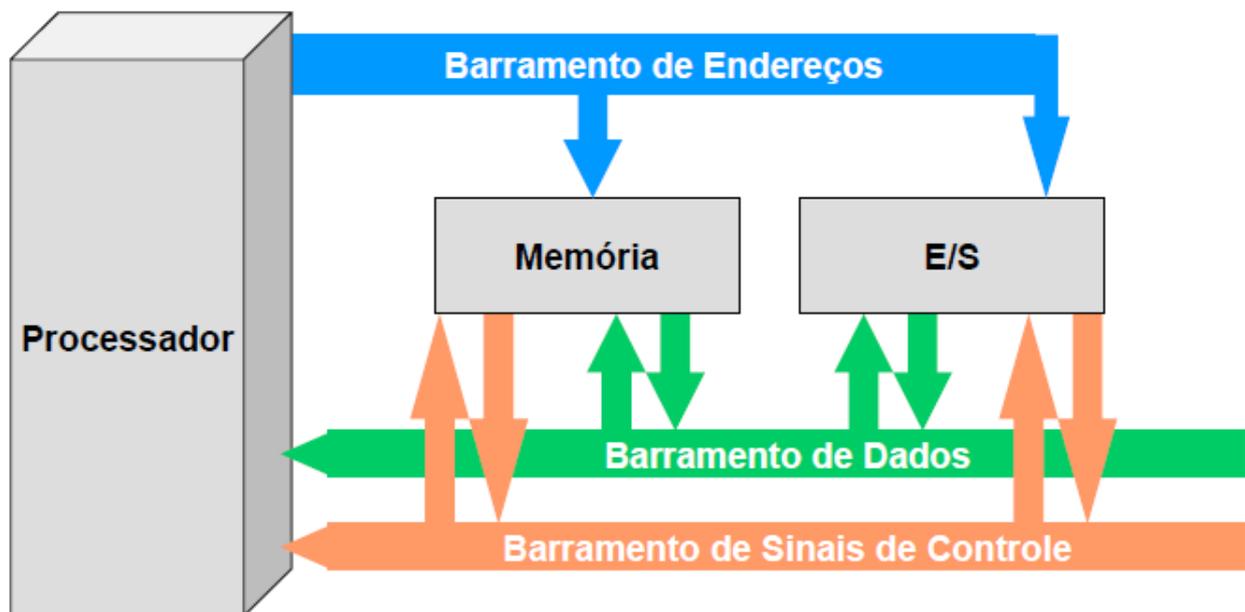


Arquitetura(modos como são construídos) ARM(consome menos energia e gera menor calor), X86(exige cooler porque gera muito calor, 64 bits), já estamos na quinta geração de computadores, hoje as arquiteturas conversam, envolvendo hardware, software básico e software aplicativo.

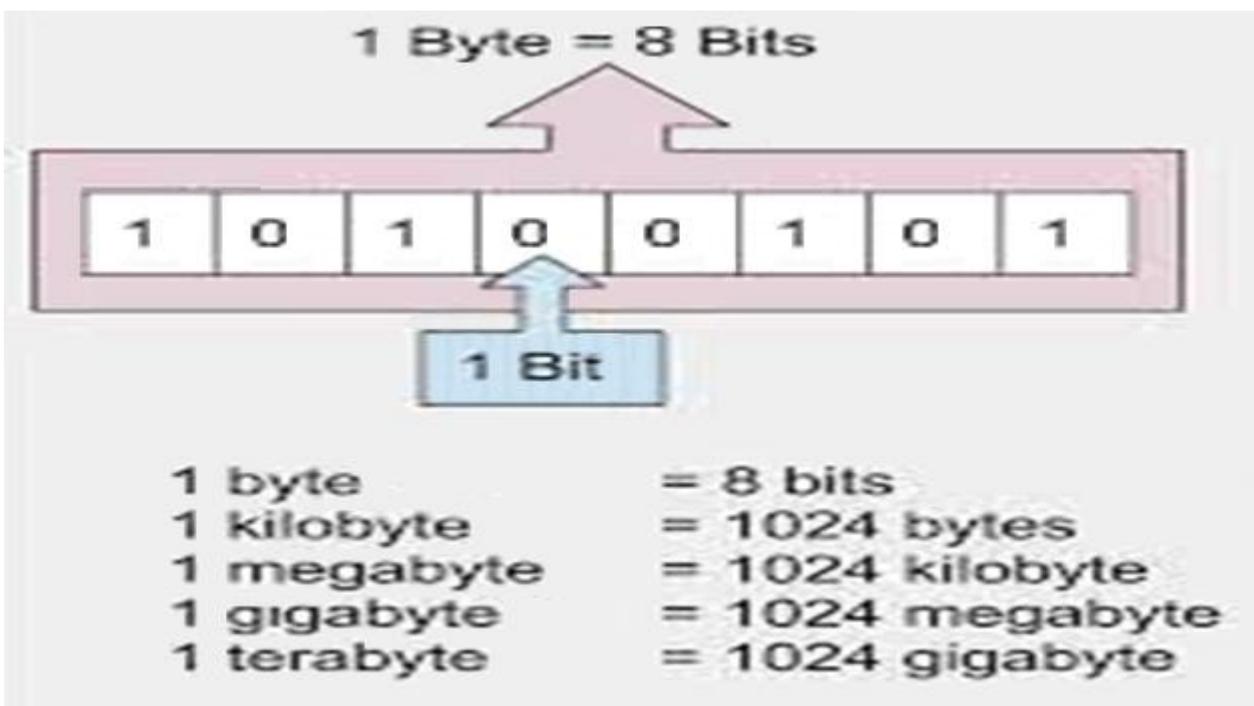
x86	ARM
<ul style="list-style-type: none">• BIOS• OS (Windows X, WES X, Windows Server X, WEC X, Linux, Android,...)• BSP/Driver	<ul style="list-style-type: none">• Bootloader• OS (Linux, WEC7, Android,)• BSP/ Driver

BSP/Driver software que controla a comunicação entre o software e o hardware.



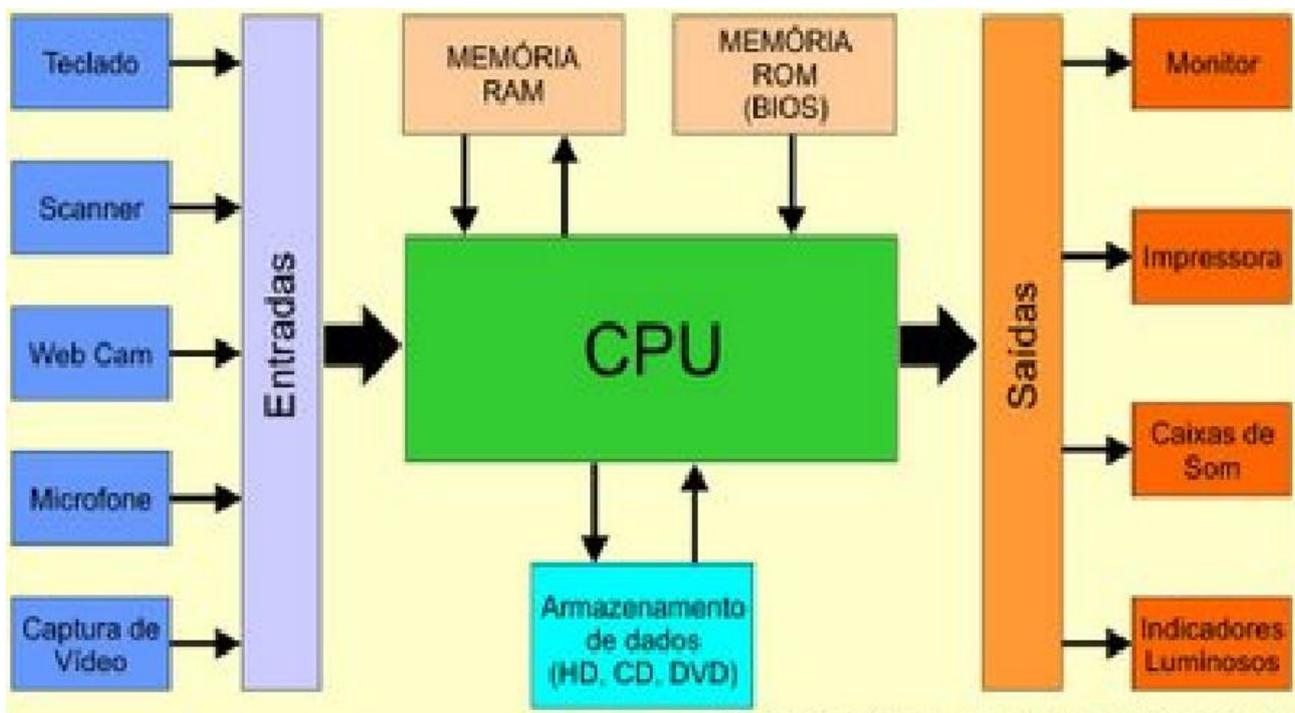
Barramento é a quantidade de bits endereçados de uma vez.

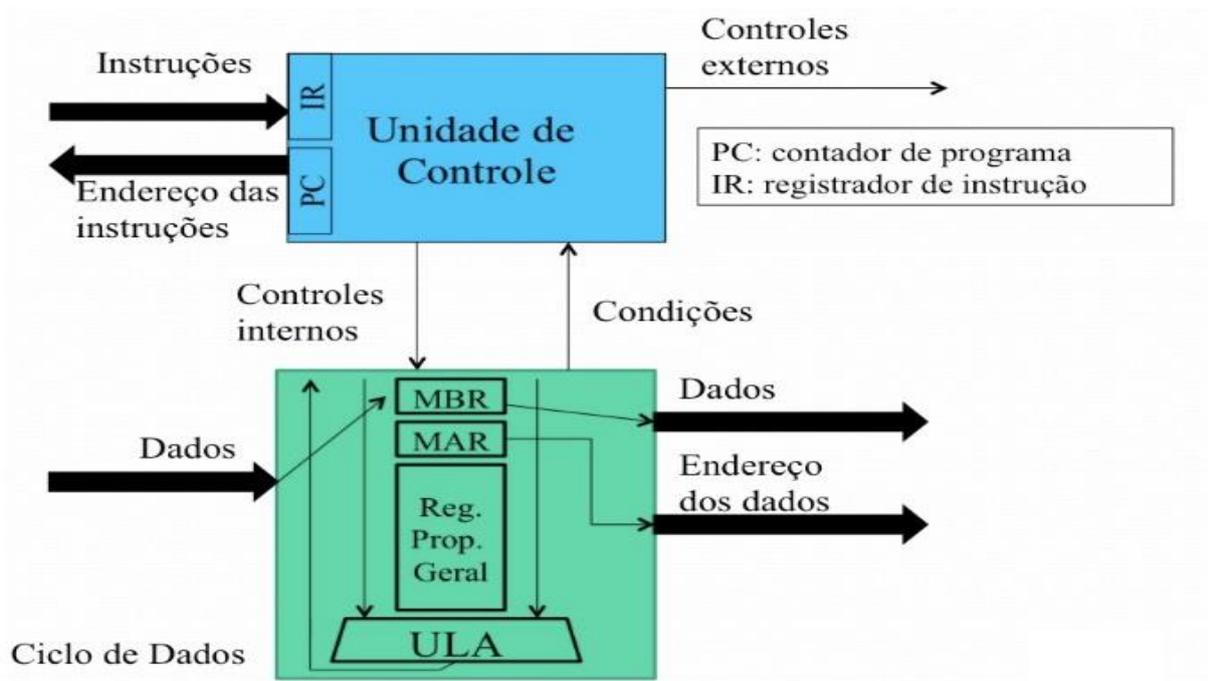
Barramento do processador	Responsável pela comunicação do processador com seu exterior
Barramento de endereços	Indica o local onde os processos devem ser extraídos e para onde devem ser enviados após o processamento.
Barramento de entrada/saída	Iremos abordar mais a fundo a seguir.
Barramento de memória	Barramento responsável pela conexão da memória principal ao processador



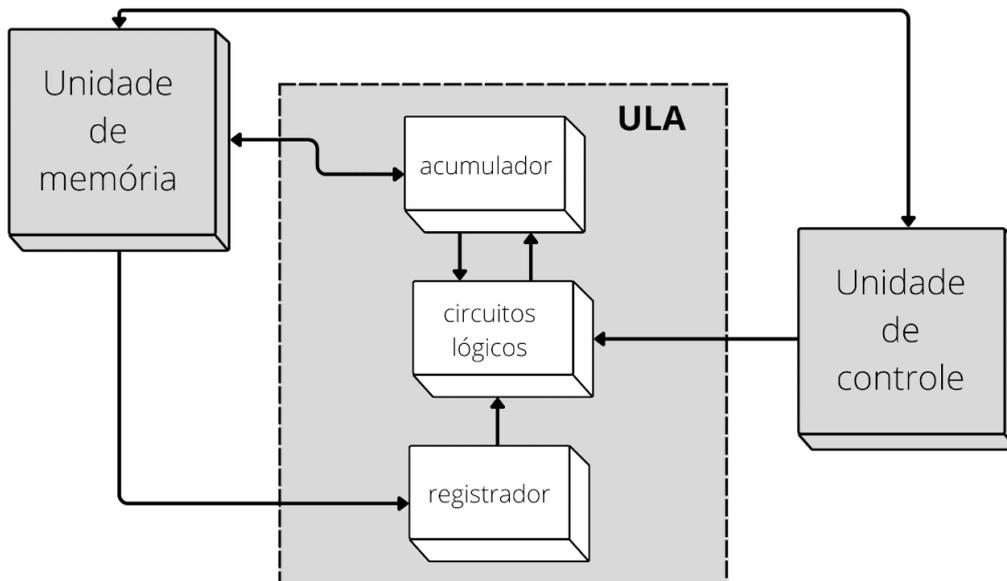
Um registrador de 8 bits quando recebe a letra A recebe 01000001.
 Um registrador de 8 bits quando recebe o número 1 recebe 00110001.

CÓDIGO BINARIO				Número	Cód. Binário				
A	01000001	N	01001110	a	01100001	n	01101110	0	00110000
B	01000010	O	01001111	b	01100010	o	01101111	1	00110001
C	01000011	P	01010000	c	01100011	p	01110000	2	00110010
D	01000100	Q	01010001	d	01100100	q	01110001	3	00110011
E	01000101	R	01010010	e	01100101	r	01110010	4	00110100
F	01000110	S	01010011	f	01100110	s	01110011	5	00110101
G	01000111	T	01010100	g	01100111	t	01110100	6	00110110
H	01001000	U	01010101	h	01101000	u	01110101	7	00110111
I	01001001	V	01010110	i	01101001	v	01110110	8	00111000
J	01001010	W	01010111	j	01101010	w	01110111	9	00111001
K	01001011	X	01011000	k	01101011	x	01111000		
L	01001100	Y	01011001	l	01101100	y	01111001		
M	01001101	Z	01011010	m	01101101	z	01111010		

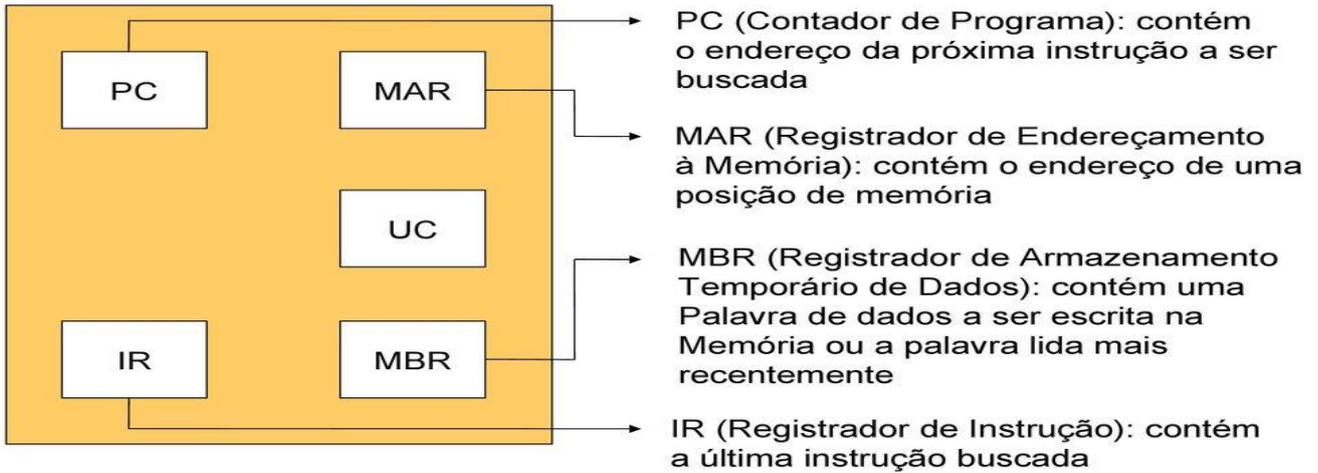




ULA – Temos as seguintes: aritmética, vídeos, músicas e ponto flutuantes.



Alguns registradores de controle e estado

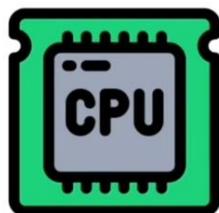
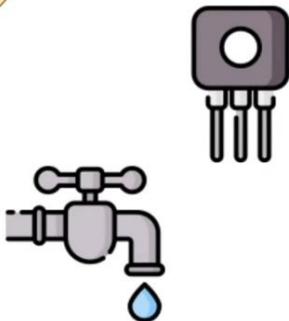


Ao trabalhar com endereço o tamanho do dado é o tamanho do barramento.

Circuitos Lógicos

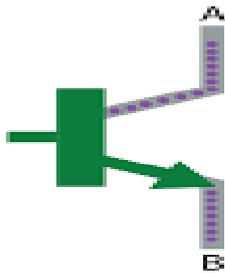
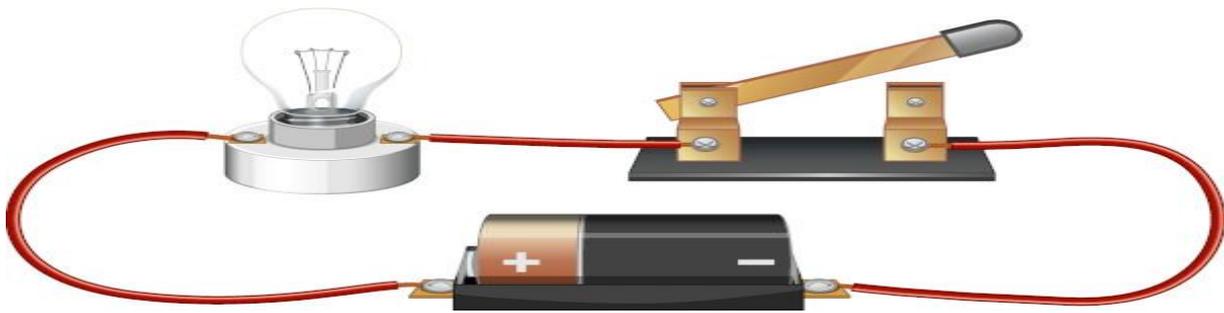
TRANSISTOR

Pequeno dispositivo que transfere sinais eletrônicos através de um resistor

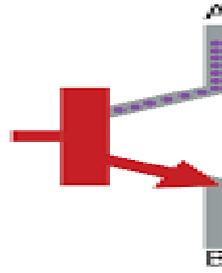


Existem bilhões de transistores ligados entre si no processador, formando circuitos capazes de fazer cálculos simples ou extremamente complexos, como a posição do mouse na tela até o volume de partículas de fumaça em um jogo.

Binário contém apenas 0 ou 1, ligado desligado.



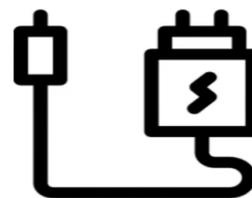
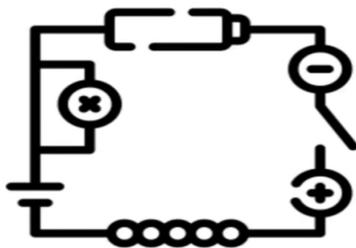
Transistor "ligado":
corrente passa de A para B



Transistor "desligado":
corrente não passa de A para B

Álgebra booleana

Manipulação de objetos que podem assumir apenas dois valores: verdadeiro e falso



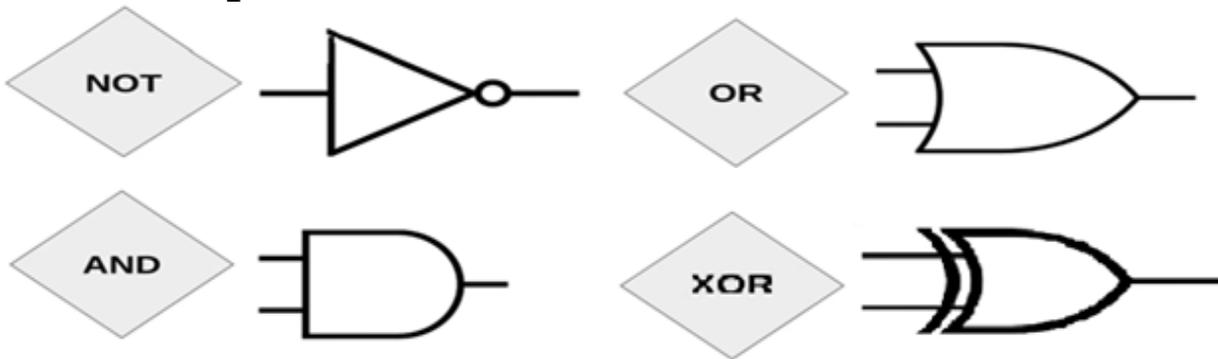
Temos 5 elementos:

Entrada, saída, controle, processamento e memória.

Transistores com portas lógicas:

NOT tem uma entrada e uma saída.

As demais podem ter duas ou mais entradas.



AND para multiplicação.

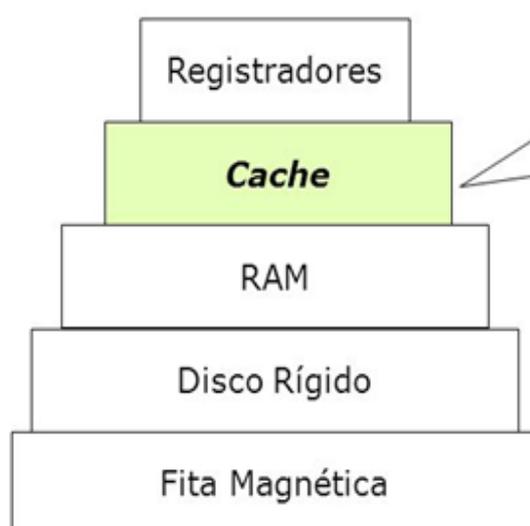
OR para adição.

Resumo dos Blocos Lógicos

Nome	Símbolo Gráfico	Função Algébrica	Tabela Verdade															
E (AND)		$S=A \cdot B$ $S=AB$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>S=A·B</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	S=A·B	0	0	0	0	1	0	1	0	0	1	1	1
A	B	S=A·B																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
OU (OR)		$S=A+B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>S=A+B</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	S=A+B	0	0	0	0	1	1	1	0	1	1	1	1
A	B	S=A+B																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
NÃO (NOT) Inversor		$S=\bar{A}$ $S=A'$ $S= \neg A$	<table border="1"> <thead> <tr> <th>A</th> <th>S=\bar{A}</th> </tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>	A	S= \bar{A}	0	1	1	0									
A	S= \bar{A}																	
0	1																	
1	0																	
NE (NAND)		$S=\overline{A \cdot B}$ $S=(A \cdot B)'$ $S= \neg(A \cdot B)$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>S=$\overline{A \cdot B}$</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	S= $\overline{A \cdot B}$	0	0	1	0	1	1	1	0	1	1	1	0
A	B	S= $\overline{A \cdot B}$																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOU (NOR)		$S=\overline{A+B}$ $S=(A+B)'$ $S= \neg(A+B)$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>S=$\overline{A+B}$</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	S= $\overline{A+B}$	0	0	1	0	1	0	1	0	0	1	1	0
A	B	S= $\overline{A+B}$																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
XOR		$S=A \oplus B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>S=A⊕B</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	S=A⊕B	0	0	0	0	1	1	1	0	1	1	1	0
A	B	S=A⊕B																
0	0	0																
0	1	1																
1	0	1																
1	1	0																



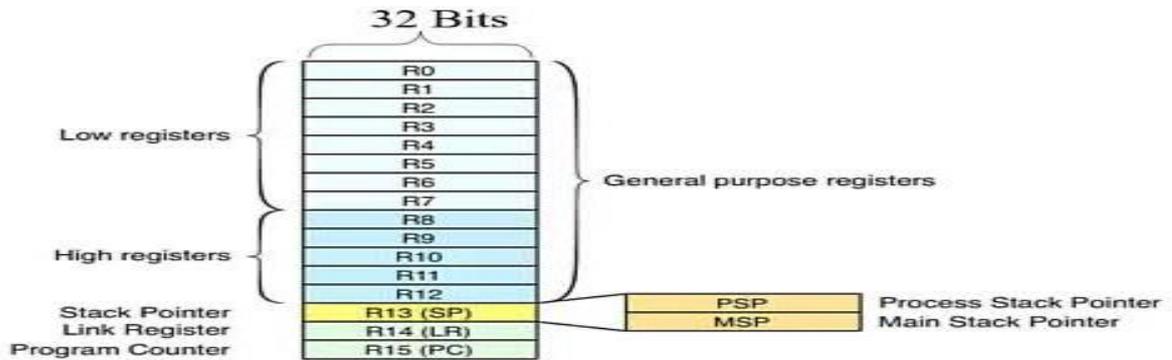
Memória



Rápida: nseg
Capacidade: K/M bytes
Controlada por hardware
Alto custo por byte
Volátil
Vários sub-níveis

Registadores

Registadores de Propósito Geral

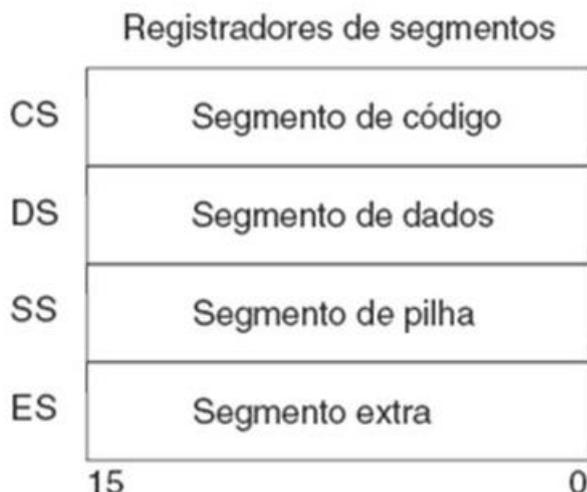


Registadores trabalham com bits, tem um barramento específico.
Um dado no registrador á tratado com endereço e tamanho.

Tipos de registadores:

De segmento, de estado, gerais e de ponteiro e índice.

Registadores de segmento



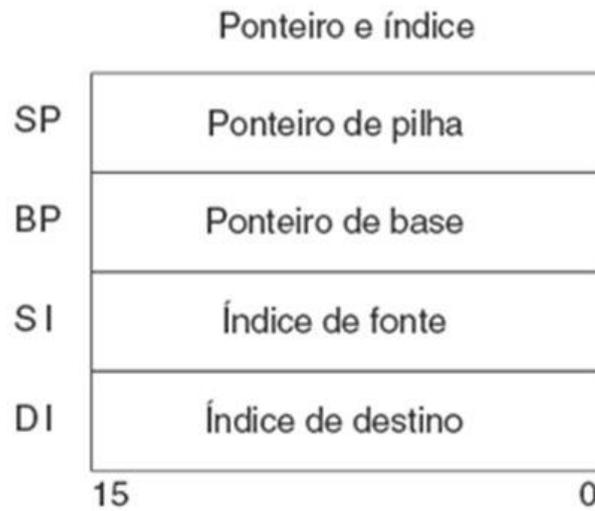
Registrador de Estado

- Flags utilizadas para indicar os resultados obtidos em operações aritméticas ou lógicas:
 - CF = Carry Flag
 - PF = Parity Flag
 - AF = Auxiliary Flag
 - ZF = Zero Flag
 - SF = Sign Flag
 - OF = Overflow Flag

Registradores gerais

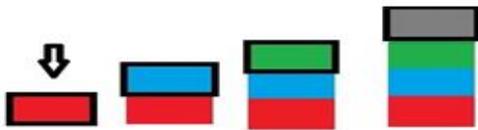
- **AX**
 - Acumulador.
 - Comumente usado em operações matemáticas e de E/S.
- **BX**
 - Base.
 - Comumente usado como uma base ou registrador apontador.
- **CX**
 - Contador.
 - Usado frequentemente em loops.
- **DX**
 - Deslocamento.
 - Similar ao registrador de base.

Registradores de ponteiro e índice

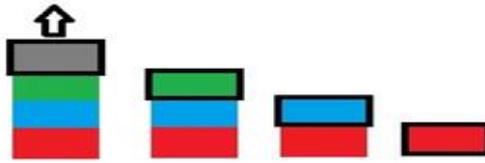


Representação de Pilha (LIFO)

O Último a Entrar...
(Last In...)

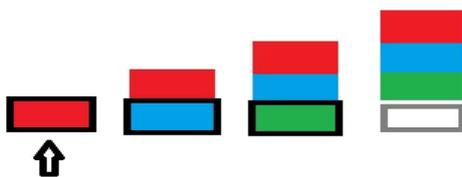


É o Primeiro a Sair.
(First Out)

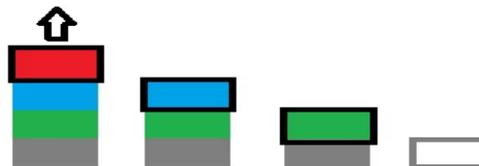


Representação de Fila (FIFO)

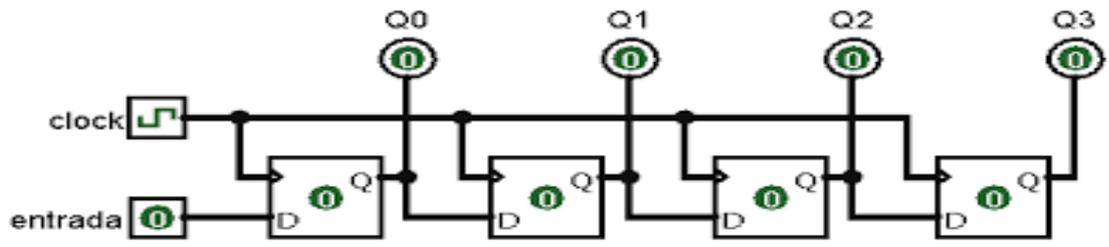
O Primeiro a Entrar...
(First In...)



É o Primeiro a Sair.
(First Out)



Registrador serial síncrono(deslocamento) **Shift Register**



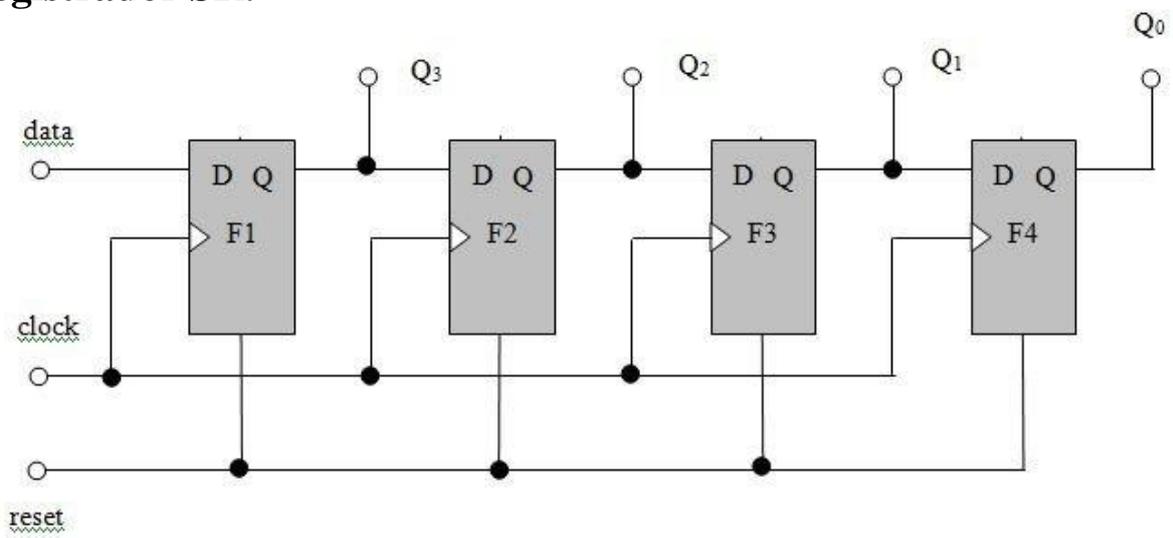
DADO DE ENTRADA = 0101

SITUAÇÃO INICIAL	Q1 = 0	Q2 = 0	Q3 = 0	Q4 = 0
PRIMEIRA SUBIDA DO CLOCK	Q1 = 1	Q2 = 0	Q3 = 0	Q4 = 0
SEGUNDA SUBIDA DO CLOCK	Q1 = 0	Q2 = 1	Q3 = 0	Q4 = 0
TERCEIRA SUBIDA DO CLOCK	Q1 = 1	Q2 = 0	Q3 = 1	Q4 = 0
QUARTA SUBIDA DO CLOCK	Q1 = 0	Q2 = 1	Q3 = 0	Q4 = 1



Do menos significativo para o mais significativo.

Registrador SR.



Flip-Flop com Clock

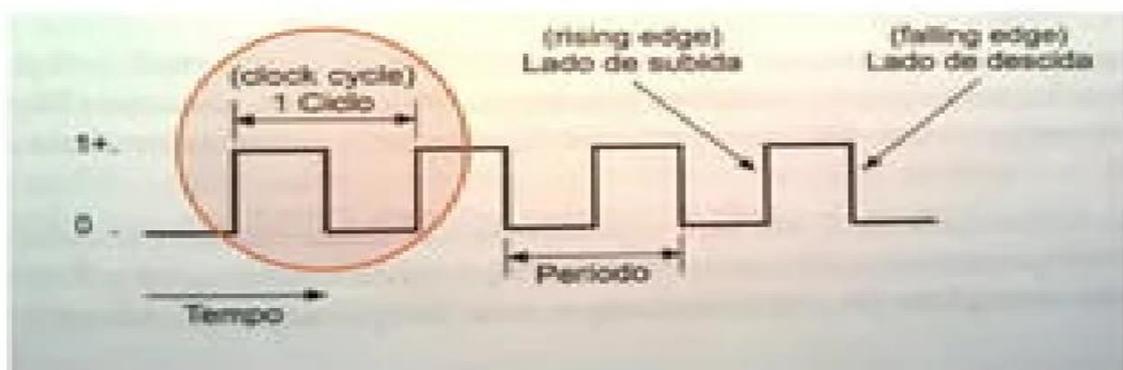
- ❑ Sistemas Digitais podem operar de 2 maneiras:
 - ✓ **Assíncrona**: As saídas do FF podem mudar a qualquer momento em que uma ou mais entradas mudam de estado;
 - ✓ **Síncrona**: Os momentos exatos que uma saída qualquer pode mudar são determinados por um sinal de Clock.
- ❑ **Clock** = É um trem de pulsos retangulares ou uma onda quadrada;
 - ✓ Geralmente é distribuído por todo o sistema, fazendo que a maioria das saídas só mudem quando ocorre uma **transição no sinal** do Clock.

Clock

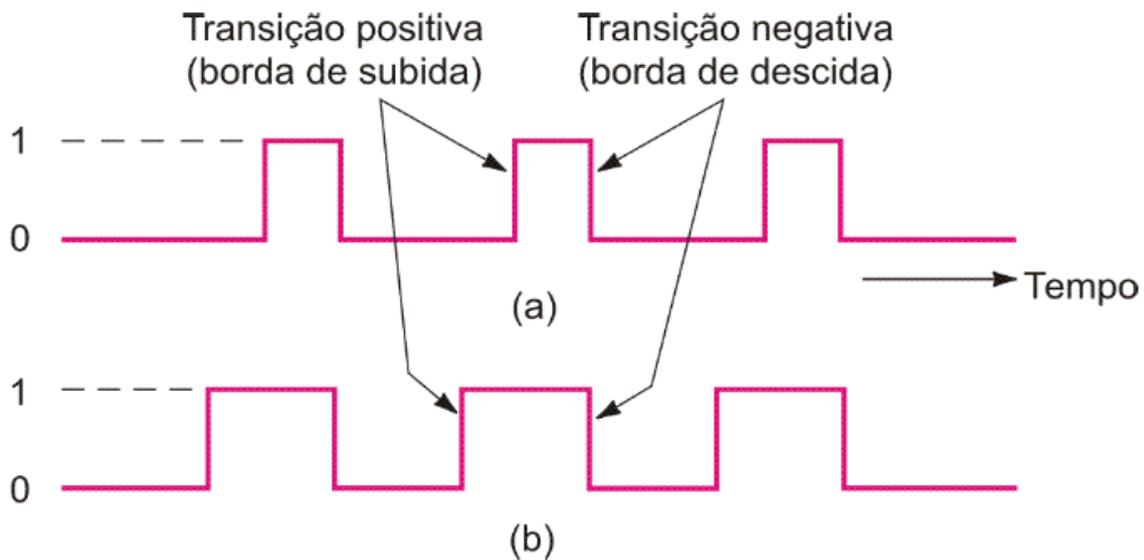
- Elementos do CLOCK:

- **Ciclo de relógio ou apenas ciclo:**

- É o intervalo de tempo entre o início da subida, ou da descida, de um pulso até o início da subida, ou da descida, do outro pulso; (a figura abaixo tem 4 ciclos)

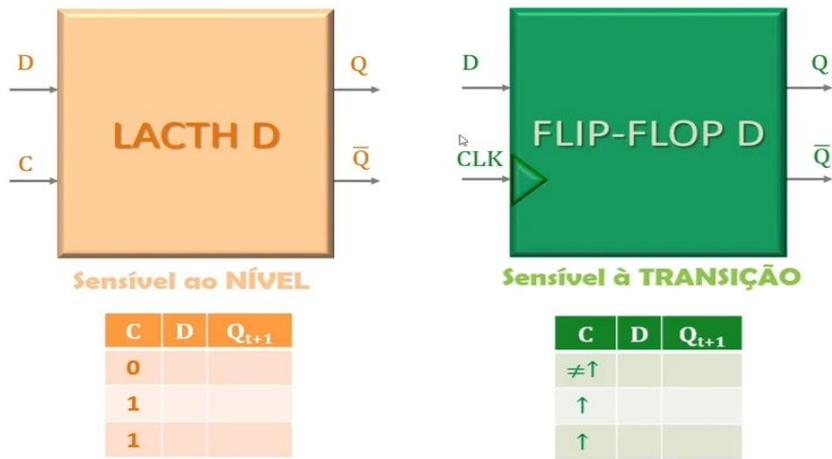


Conceitos: Sinais de “clock”.

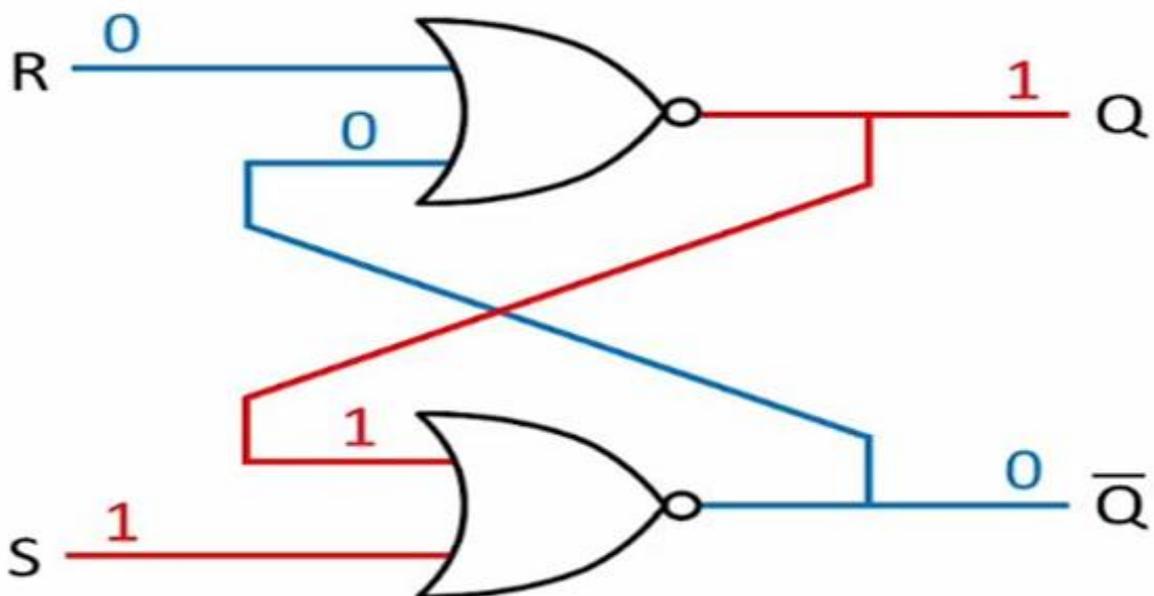


Núcleos processam os dados simultaneamente.

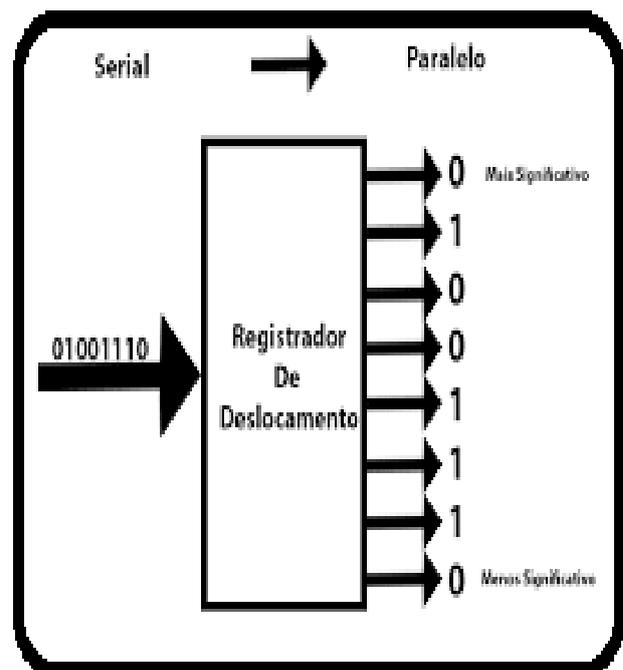
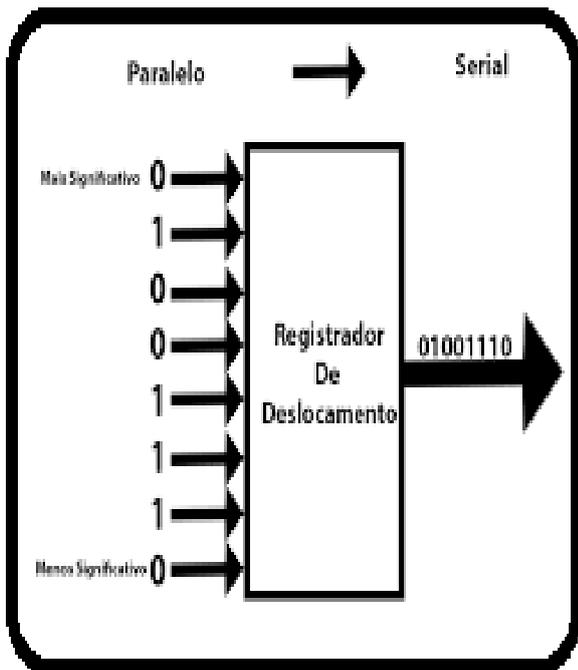
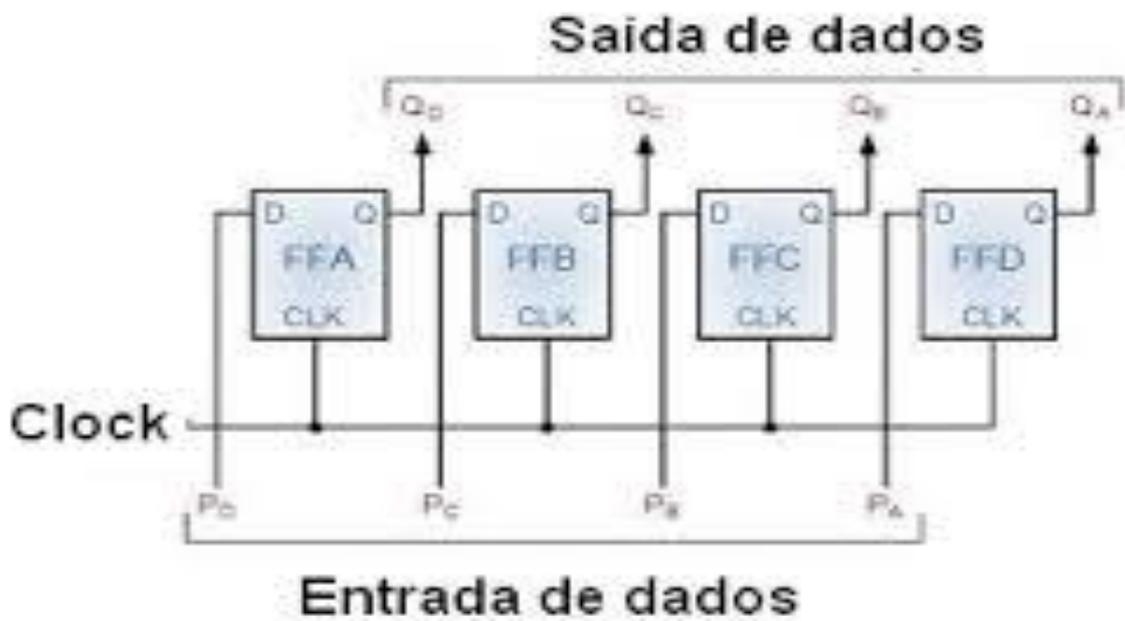
ARMAZENAMENTO DE BIT: LATCH e FLIP-FLOP



Latch tipo SR assíncrono(existe outro o JK)

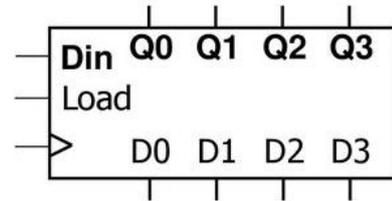


Entrada paralela de dados(uma subida de clock apenas).

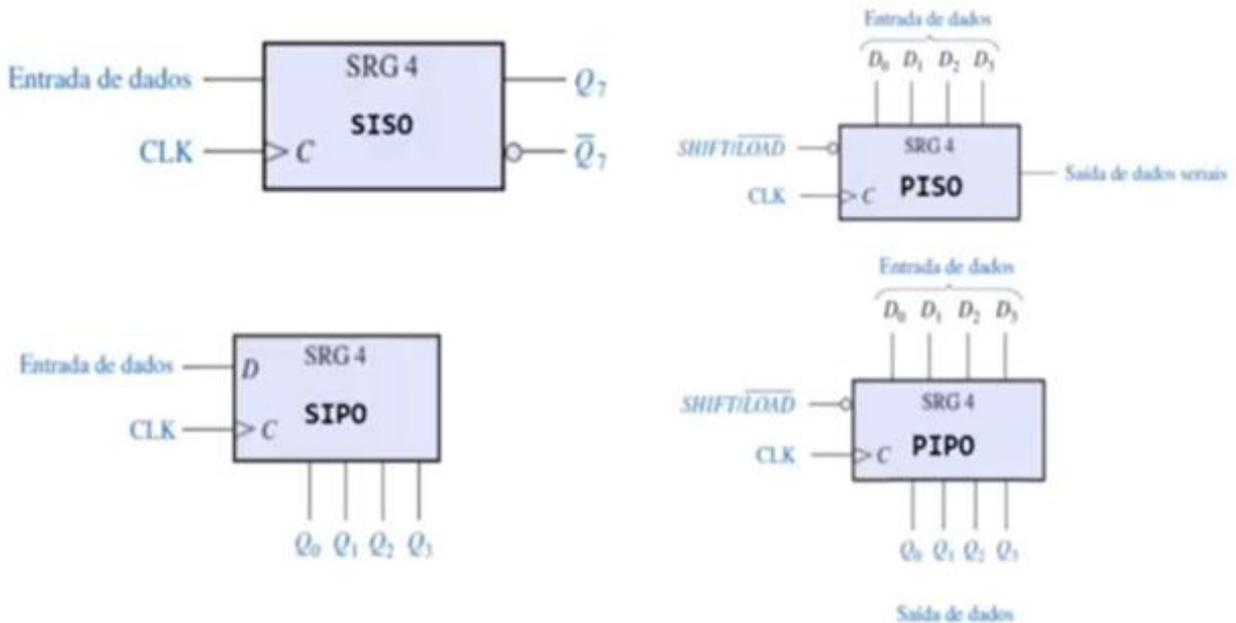


Registadores universais

- Comercialmente existe um tipo de registrador de deslocamento chamado **universal**. Este tipo de registrador permite trabalhar de todos os modos possíveis:
 - Entrada-serial / saída-paralelo
 - Entrada-paralelo / saída-serial
 - Entrada-paralelo / saída-paralelo
 - Entrada-serial / saída-serial



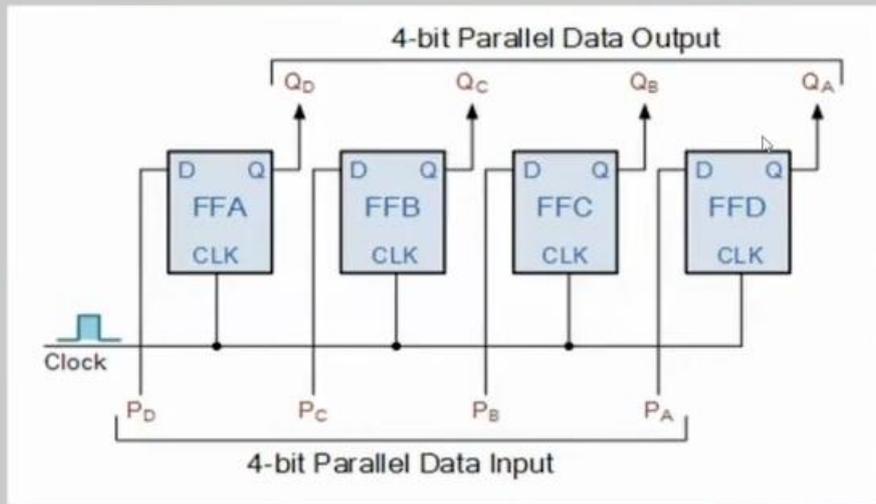
Registadores com entradas e saídas seriais e paralelas.



Registadores

Entrada Paralela e Saída Paralela

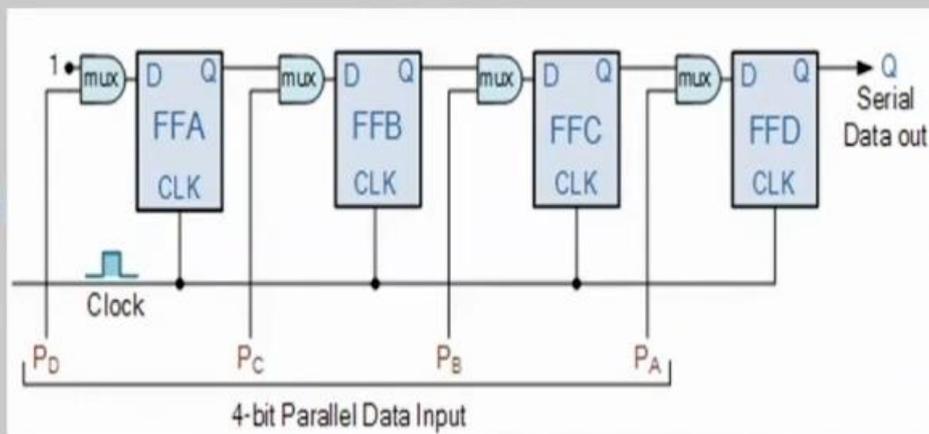
PIPO



Registadores

Entrada Paralela e Saída Serial

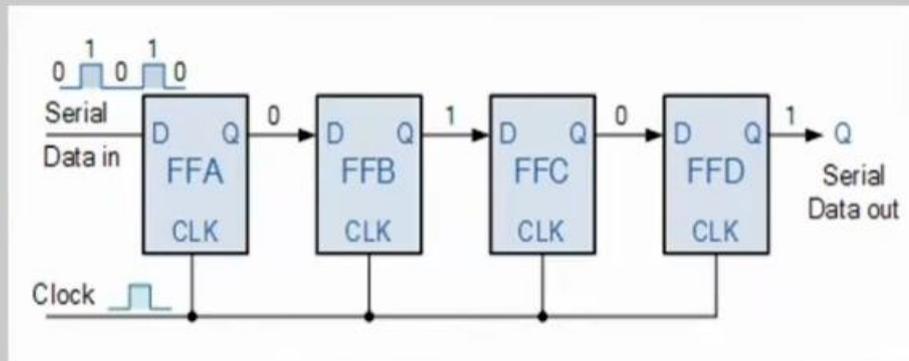
PISO



Registadores

Entrada Serial e Saída Serial

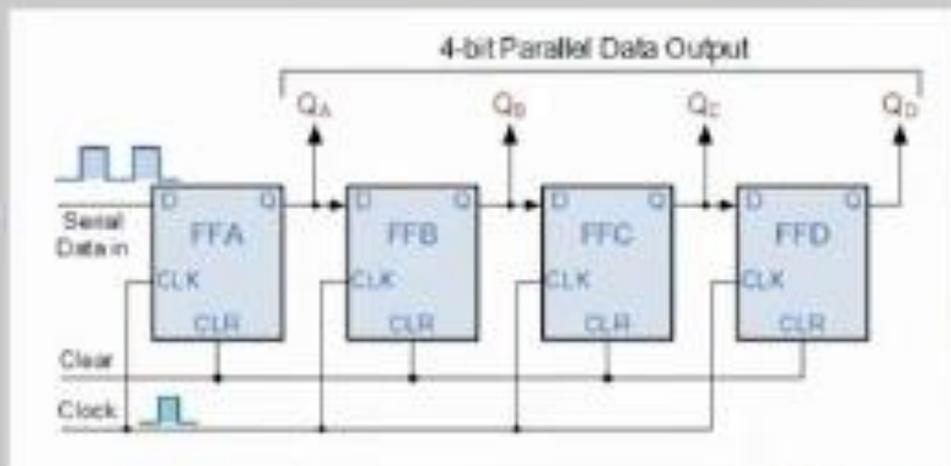
SISO



Registadores

Entrada Serial e Saída Paralela

SIPO



Comunicação

Paralela

LPT (IEEE1284)

IDE

PCI

Serial

Assíncrona

RS232

RS485

OneWire

Síncrona

SPI

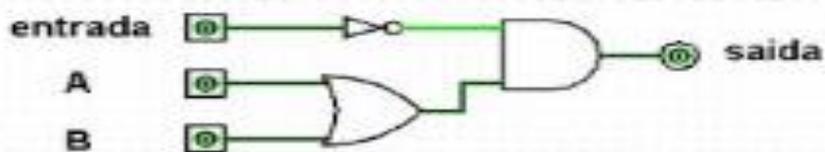
I2C

I2S

Circuito combinacional, só depende das suas entradas.

Circuito sequencial, depende de entradas e saídas.

Circuito Combinacional



Circuito Sequencial

