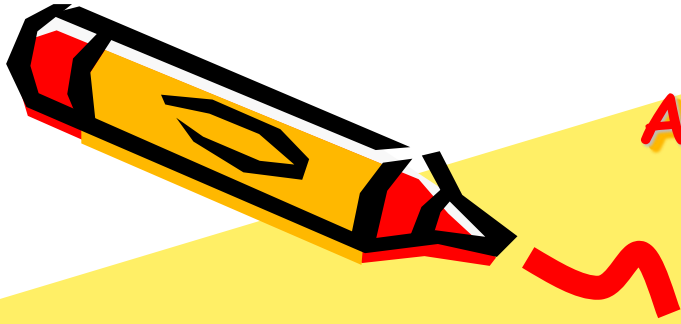


# Arquitetura de Computadores



## UNIDADE 4

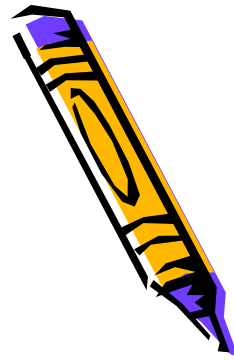
### "Arquitetura de Microprocessadores"

Aula nº 15 e 16



1º Período

Ano letivo 2017/2018



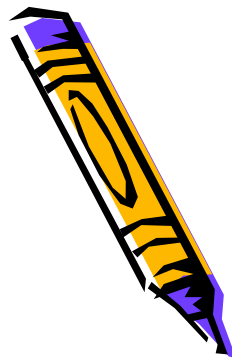
# Sumário

- O CPU (continuação)
- Registos internos;
- BUS interno Vs FSB



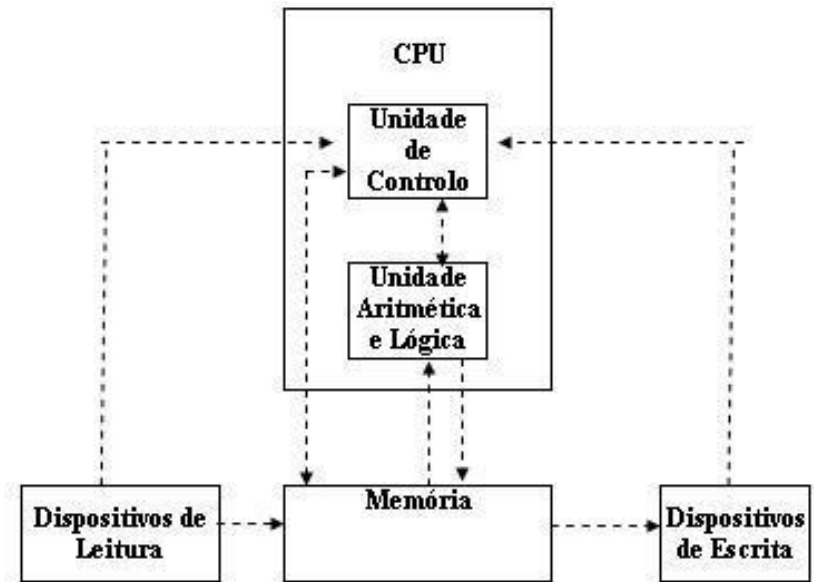
# Microprocessadores

Na última aula...



Blocos que arquitetura von Neuman apresentava?

- **ALU ou ULA, Unidade Lógica e Aritmética**
- **UC, Unidade de Controlo**
- **Memória**
- **Entrada e Saída (Input/Output)**



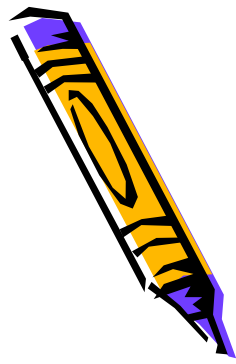
O EDVAC (1952) foi o primeiro a utilizar esta arquitetura

# Microprocessadores

Na última aula...

## Características mais importantes

- **Velocidade de relógio** - é a velocidade de processamento no interior do CPU.
- **Largura dos canais de comunicação** - forma como os diversos componentes do interior e exterior do CPU estão interligados.



# Microprocessadores

Na última aula...

## A velocidade de um processador

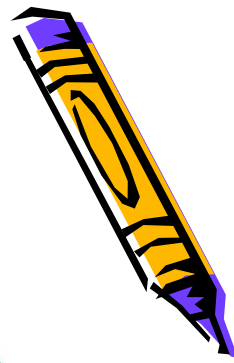
É medida em Hertz (Hz) ou ciclos por segundo ( $S^{-1}$ ) é o inverso do período

$$f = \frac{1}{T} \quad \text{Assim } 1 \text{ Hz} = 1 \text{ ciclos/segundo}$$

Um CPU com **velocidade de 100 Hz executa 100 ciclos por segundo**, enquanto que um processador de 3,2 GHz (GigaHertz) executa 3,2 bilhões de ciclos por segundo.



Heinrich Rudolf Hertz (1857-1894)



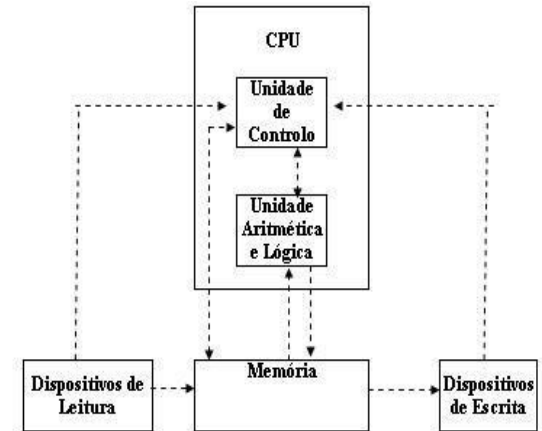
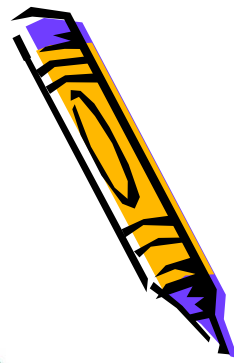
# Microprocessadores

Na última aula...

## A velocidade de um processador

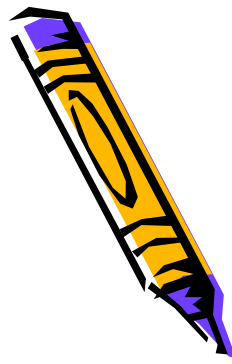
O desempenho de um processador está associado também à largura de canais de comunicação.

- Registos internos
- Barramento de Endereços (Address BUS)



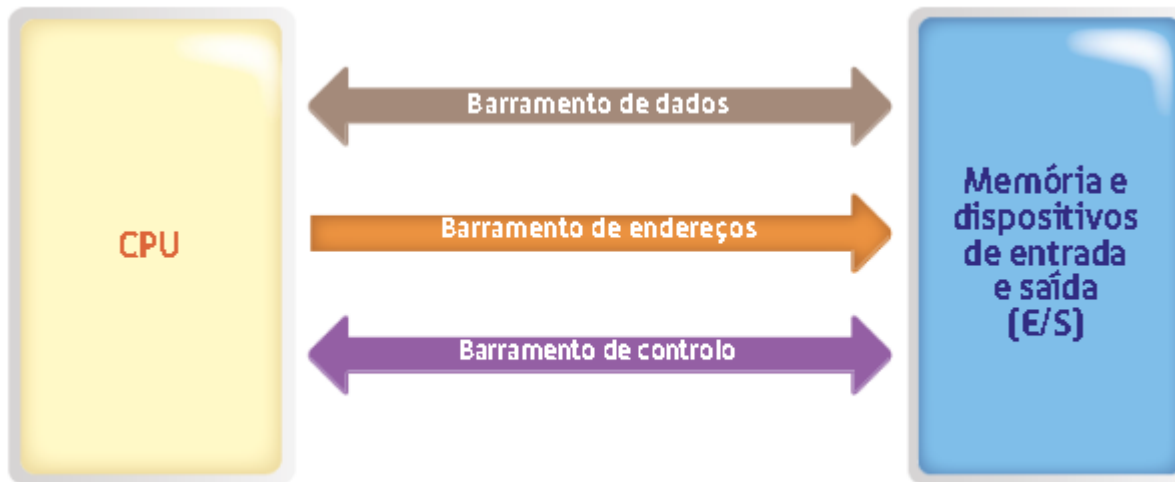
# Microprocessadores

Na última aula...



## BUS

- Barramento de controlo

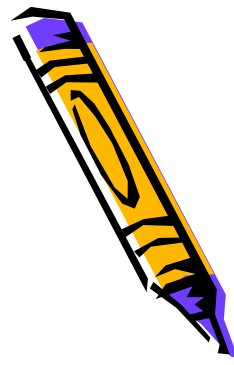


Barramento de dados, endereços e controlo

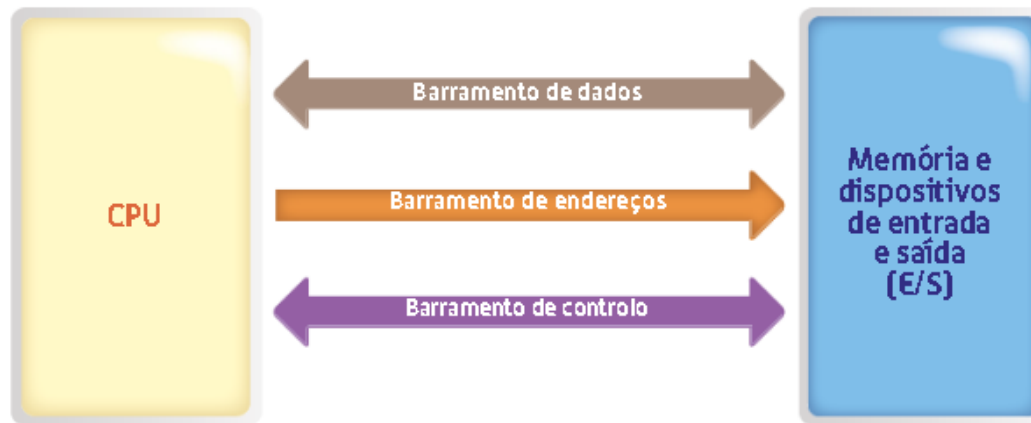


# Microprocessadores

## BUS



Cada BUS tem um número de pistas associadas que mais não é do que a quantidade de bits que pode transportar em simultâneo ( 16 pistas permitem o transporte de 16 bits de cada vez)...



Barramento de dados, endereços e controlo

O barramento de dados e controlo são bidirecionais.

O barramento de endereços tem sempre o mesmo sentido

CPU → Memória/E/S

Apenas o CPU pesquisa por endereços na memória ou dispositivos de E/S.

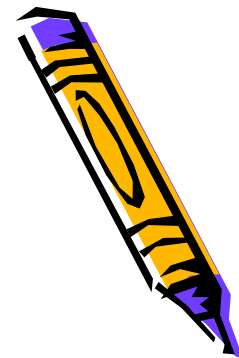
O contrário não faz qualquer sentido por isso é unidirecional.





# Microprocessadores

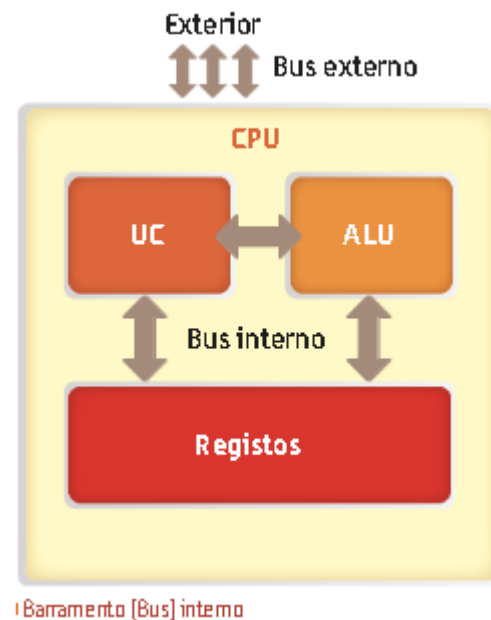
## CPU



### Registos internos

São pequenas memórias de um determinado tamanho (32 bits ou 64 bits, atualmente) que definem o conjunto de bits com que o CPU trabalha de cada vez (ciclo de relógio).

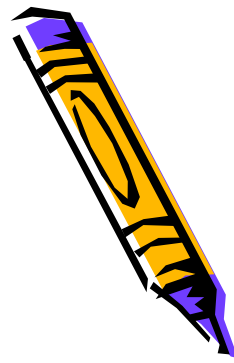
Pelo tamanho dos registos é possível determinar quantidade de informação com que o CPU consegue lidar internamente por ciclo de relógio.



Os processadores a 64 bits existentes hoje em dia, apenas tiram partido máximo da sua arquitetura se correrem num sistema operativo (SO) a 64 bits. Se dispusermos de um SO a 32 bits o processador irá trabalhar a "meio gás".

# Microprocessadores

## CPU



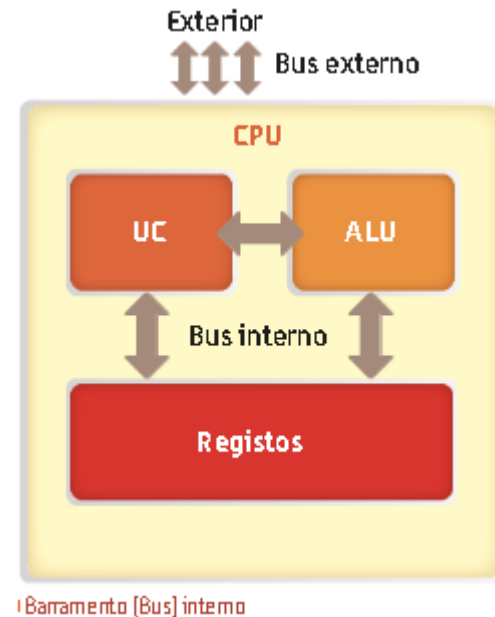
### Registos internos

- ✓ São o tipo de memória mais rápida acessível ao processador;
- ✓ São usados quando determinada informação é necessária novamente no decorrer de uma instrução.

#### Exemplo:

Para obter o resultado de  $4 \times (2+3)$ , o processador realiza a soma de  $2+3$  primeiro e em vez de guardar o resultado na memória RAM, fá-lo nos registos, para que possa **recorrer a esse valor mais rapidamente**.

- ✓ Esta ligação é feita através do **barramento interno (BUS interno)**



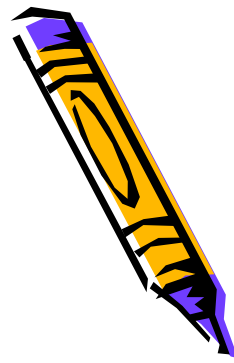
1 Barramento [Bus] interno

### BUS interno

Interliga os diversos componentes no interior do CPU, funciona à velocidade interna.

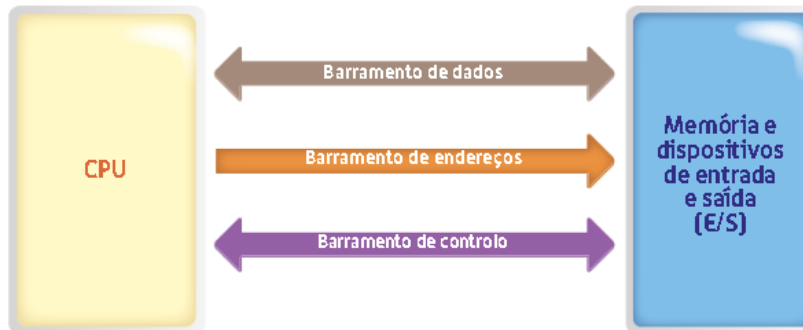
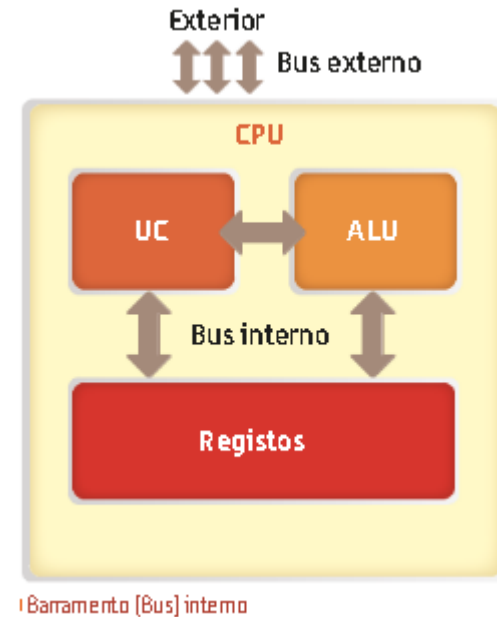
# Microprocessadores

## CPU



### BUS interno Vs FSB

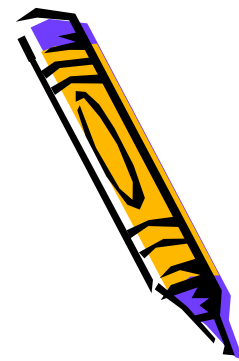
- ✓ Como já referido, os números de transístores contidos no Chip (CPU) conferem-lhe a sua velocidade (velocidade de relógio interno que define com que velocidade os dados são processados no interior do processador).
- ✓ A velocidade com que os dados viajam pela motherboard são assegurados pelo FSB (Front Side Bus), menor velocidade



Barramento de dados, endereços e controlo

# Microprocessadores

## CPU



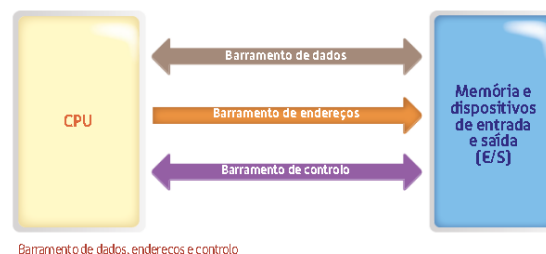
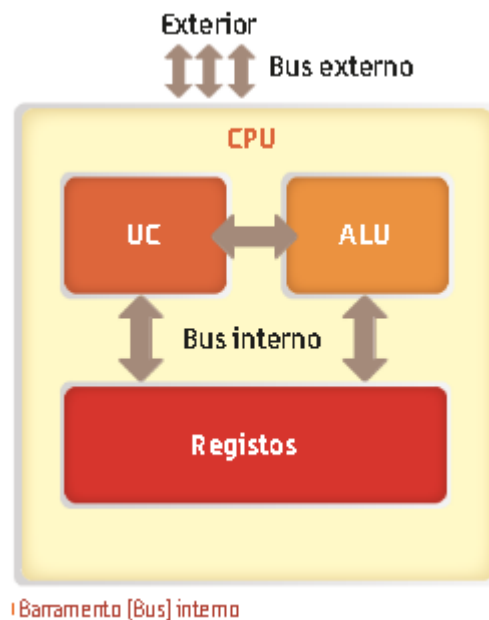
### BUS interno Vs FSB

#### Problema!

O aumento significativo da velocidade do processador, não foram acompanhadas pela velocidade oferecidas pelas Motherboards.

- ✓ Existem duas velocidades distintas num PC que funcionam a ritmos diferentes.
- ✓ **Exemplo:** Um processador Pentium 2,4 GHz tem uma velocidade interna de 2,4 GHz mas a motherboard funciona apenas a 266 MHz, pelo que a velocidade de BUS é de 266 MHz. Por cada ciclo da motherboard passam 9 ciclos de processador

$$(9 \times 266 \text{ MHz} \approx 2,4 \text{ GHz})$$



Barramento de dados, endereços e controlo

# Microprocessadores

## CPU

### BUS interno Vs FSB

**Cálculo do número de bits que passam no BUS por segundo.**

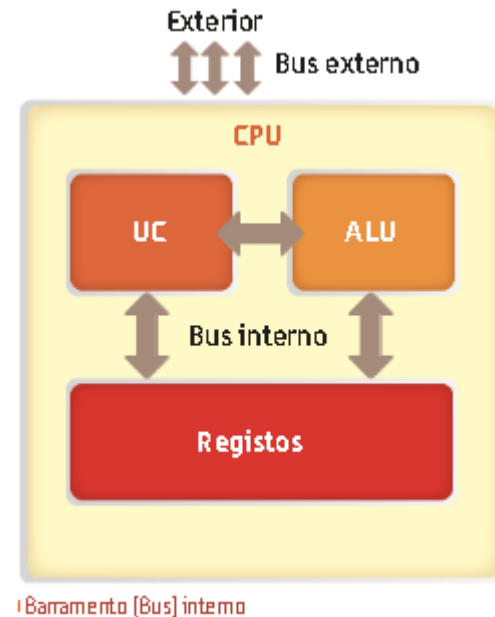
- ✓ **Exemplo:** Para uma motherboard que opere a 266 MHz com um interface de 16 bits (nr de pistas)

Nr bits = velocidade que opera  $\times$  nr bits

$$266 \times 10^6 \times 16 = 4256 \text{ Mb/s}$$

Sabendo que 16 bits = 2 Bytes  
temos:

$$266 \times 10^6 \times 2 = 532 \text{ MB/s}$$



A barreira de 64 bits para este tipo de barramento trouxe problemas na sincronização na transferência de dados entre o interior do CPU e o exterior

# Microprocessadores

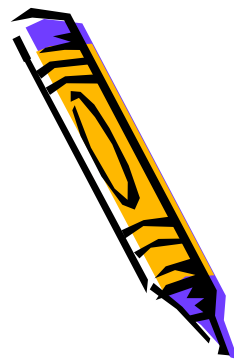
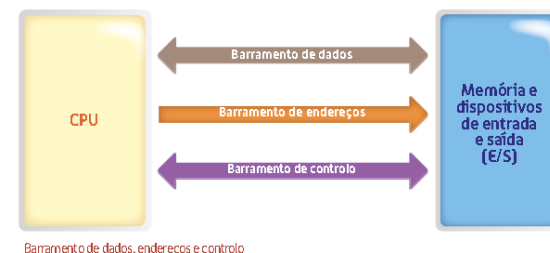
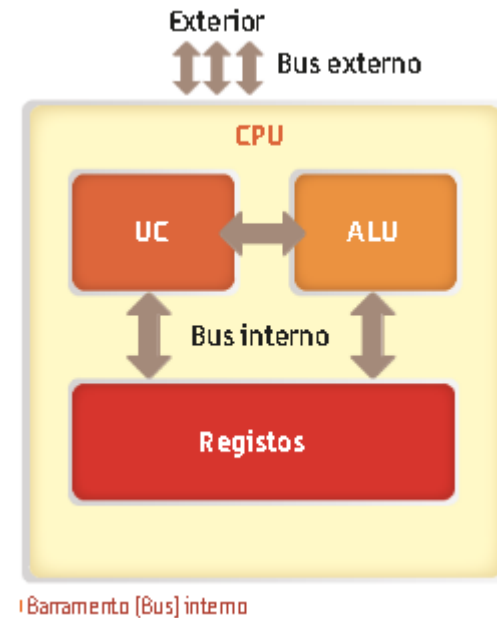
## CPU

### BUS interno Vs FSB

#### Problema - 64 bits

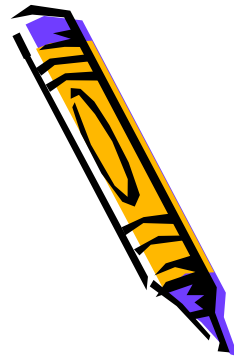
A barreira de 64 bits para este tipo de barramento trouxe problemas na sincronização com a transferência de dados entre o interior do CPU e o exterior.

As técnicas para resolução deste problema chama-se *Hypertransport* e *Quickpath Interconnect* e serão aprofundadas mais à frente.



# Microprocessadores

## EXECUÇÃO DE UM PROGRAMA RESIDENTE EM MEMÓRIA



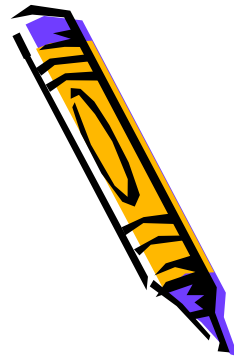
Na arquitetura de von Neumann, as instruções e dados **partilham a mesma memória e canais de comunicação (barramentos)**.

Para que o CPU identificasse quais as posições de memória que continham instruções ou dados, era necessário que estas estivessem distribuídas **de forma consecutiva**.



# Microprocessadores

## EXECUÇÃO DE UM PROGRAMA RESIDENTE EM MEMÓRIA



**Exemplo:** somar dois valores

O CPU sabe que a primeira leitura que fará da memória irá conter o código da instrução e que a posição seguinte irá conter os operandos que foram indicados na instrução.

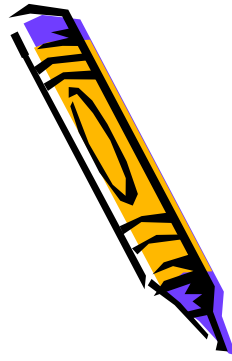
Terminada a execução desta instrução sabe automaticamente que na próxima posição de memória encontrar-se-á a próxima instrução a executar.





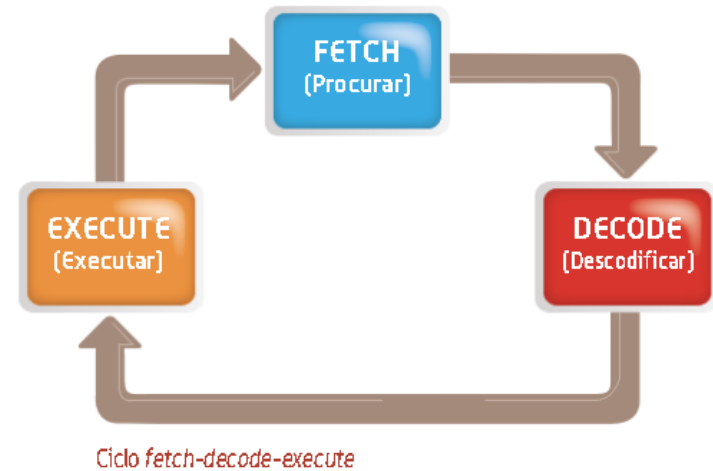
# Microprocessadores

## EXECUÇÃO DE UM PROGRAMA RESIDENTE EM MEMÓRIA



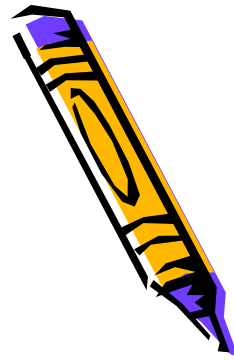
**Independente das modificações introduzidas a execução implica um conjunto de etapas:**

- **Busca (fetch)** à unidade de memória da próxima instrução a ser executada;
- **Descodificação** da instrução a executar. Identificação dos operandos (caso existam) e qual o tipo de operação a realizar.



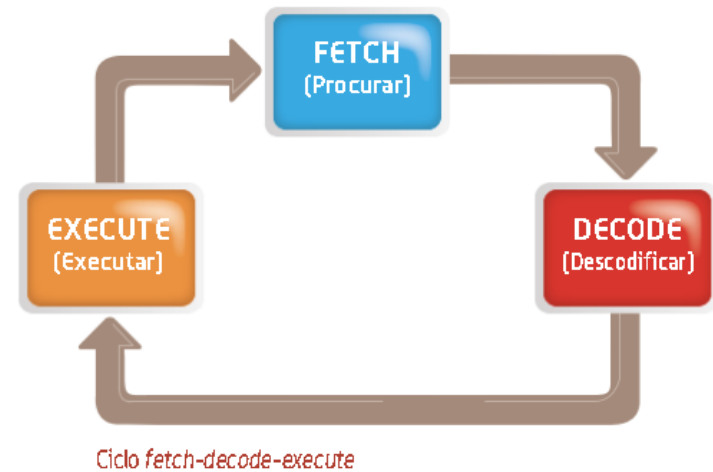
# Microprocessadores

## EXECUÇÃO DE UM PROGRAMA RESIDENTE EM MEMÓRIA



Independente das modificações introduzidas a execução implica um conjunto de etapas:

- **Obtenção dos Operandos** (caso tenham sido especificados na instrução anterior) e carregamento destes nos registos internos do CPU;
- **Execução da tarefa;**
- **Armazenamento do resultado em memória interna do CPU;**

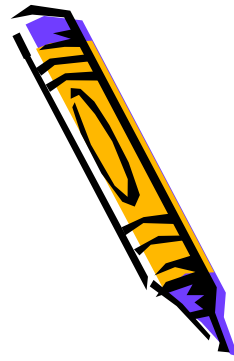


Todo o processo irá repetir-se para a próxima instrução. Todavia nem todos os programas necessitam de percorrer todos estes passos.

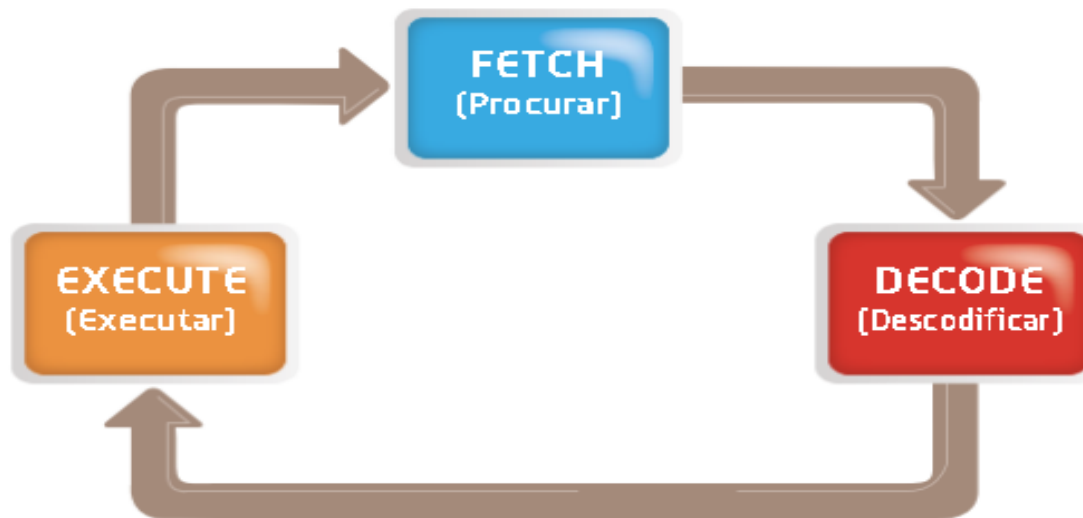
Este ciclo denomina-se por **fetch-decode-execute cycle**

# Microprocessadores

EXECUÇÃO DE UM PROGRAMA RESIDENTE EM MEMÓRIA



*fetch-decode-execute cycle*



*Ciclo fetch-decode-execute*

Nas próximas aulas apresentaremos em pormenor como todas estas etapas são sequenciadas no interior do microprocessador