

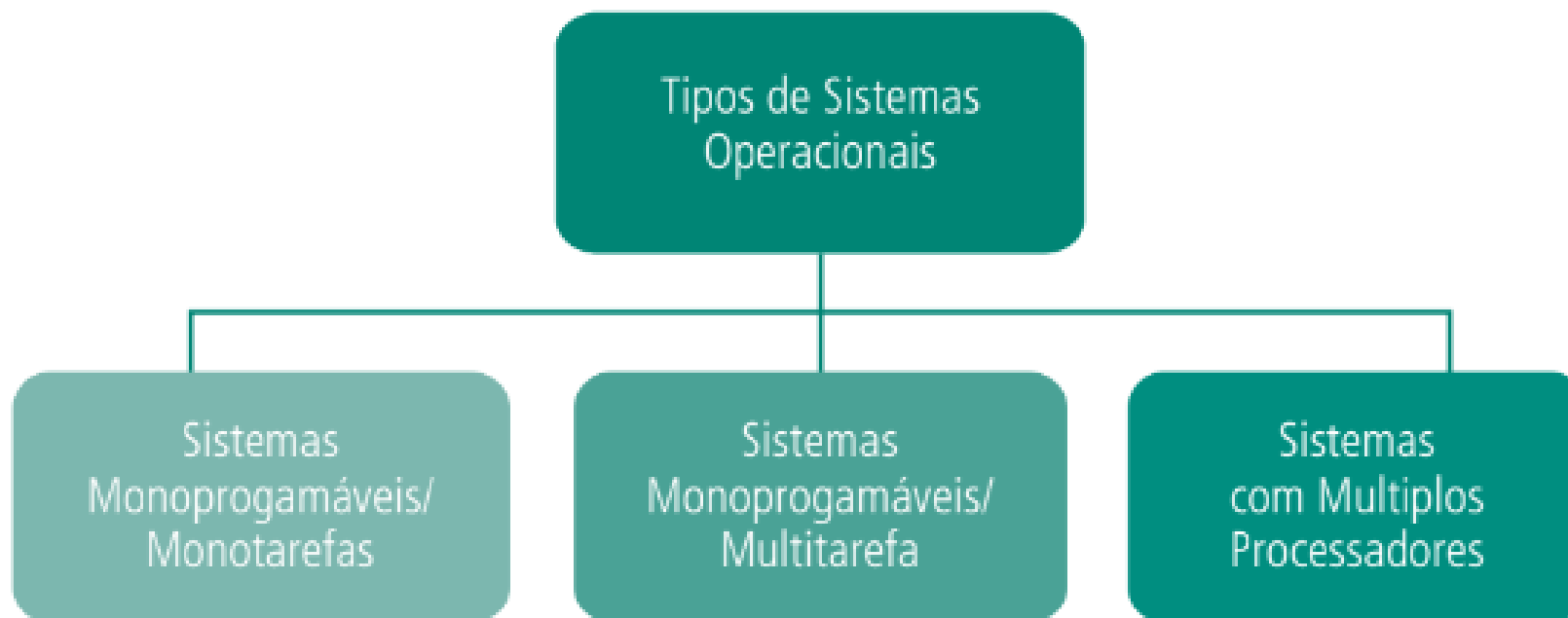
# **ISO-011 – Sistemas Operacionais e Redes de Computadores**

---

Prof. Me. Anderson Vanin

# Tipos de Sistemas Operacionais

Os tipos de sistemas operacionais e sua evolução estão relacionados diretamente com a evolução do hardware e das aplicações por ele suportadas e podem ser classificados conforme a figura abaixo:



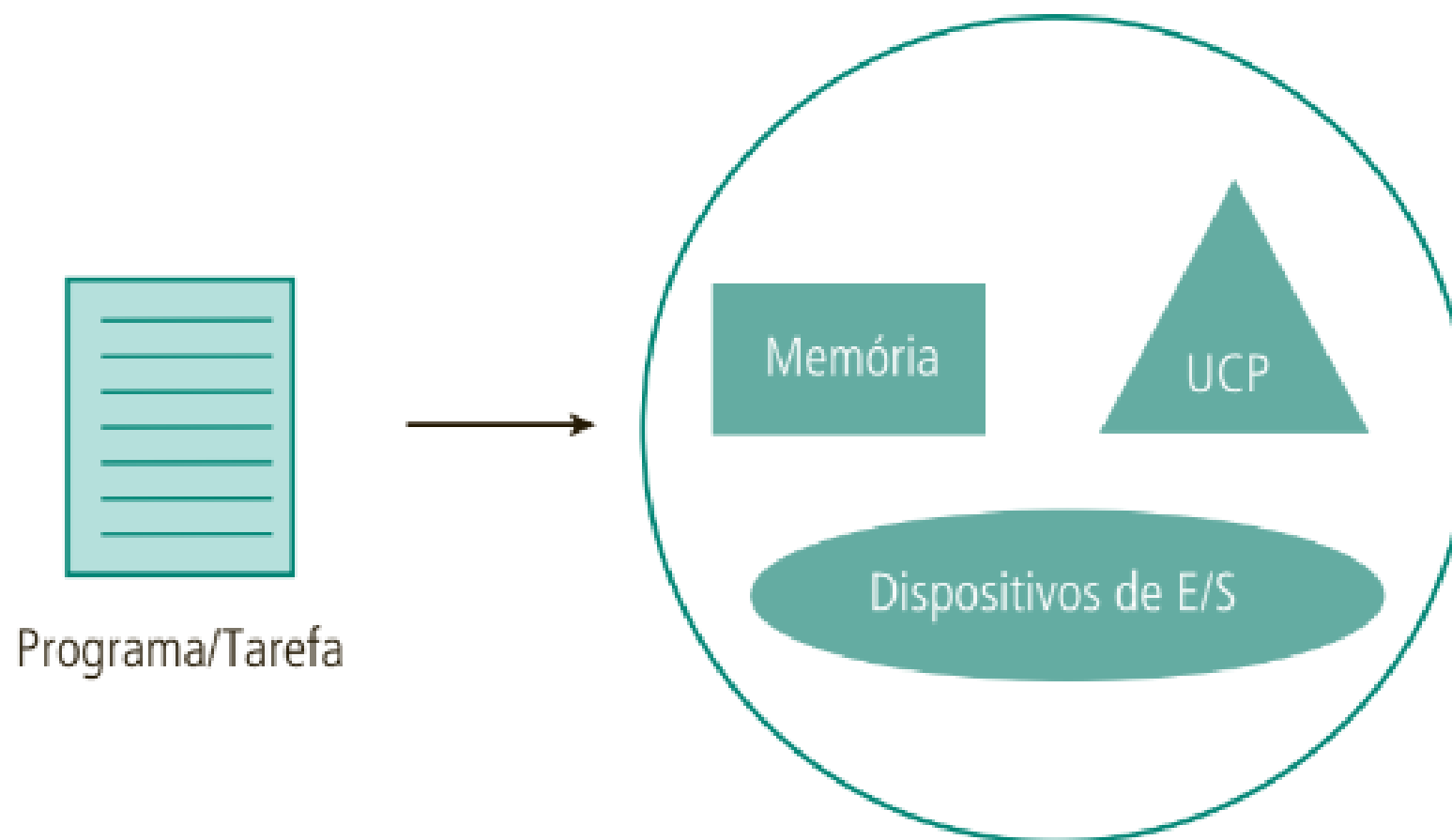
# Monoprogramáveis ou Monotarefa

Podem executar apenas um programa por vez. Para que um usuário possa executar outro programa, deverá aguardar a finalização do programa corrente. Esta era uma característica dos primeiros sistemas operacionais que estavam relacionados ao surgimento dos primeiros computadores na década de 60.

Todos os recursos de hardware ficassem exclusivamente dedicados a um único programa.

Enquanto um programa aguardava por um evento externo, como a digitação de um caractere do teclado, o processador permanecia ocioso.

# Monoprogramáveis ou Monotarefa



# Sistemas Multiprogramáveis ou Multitarefa

Os recursos computacionais são compartilhados entre os diversos usuários e suas aplicações.

Compartilhamento de tempo no processador é distribuído.

Um dos processos ocupa o processador enquanto os outros ficam enfileirados, aguardando a sua vez de entrar em execução. Cabe ao sistema operacional o papel de gerenciar de forma ordenada e protegida o acesso concorrente aos recursos disponíveis.

A vantagem desse tipo de sistema é uma melhor utilização dos recursos disponíveis, o que resulta em menor tempo de resposta das aplicações. Além de um custo reduzido, uma vez que haverá o compartilhamento dos recursos entre as diferentes aplicações e aumento da produção do usuário.

# Sistemas com múltiplos processadores

O sistema operacional distribui as tarefas entre dois ou mais processadores. A vantagem desse tipo de sistema é permitir que mais de um programa possa ser executado simultaneamente ou que um mesmo programa seja dividido em várias partes e executado simultaneamente nos vários processadores, aumentando o desempenho.

Sistemas com múltiplos processadores caracterizam-se por possuir duas ou mais CPU's interligadas e trabalhando de forma conjunta na solução de um problema.

Ideal para aplicações que requeriam um grande poder computacional, como sistemas de previsão do tempo, modelagens, simulações, desenvolvimento aeroespacial, entre outros.

# Sistemas com múltiplos processadores

Os sistemas com múltiplos processadores podem ser classificados em **fortemente acoplados e fracamente acoplados**, em função da comunicação entre CPU's e o grau de compartilhamento da memória.

Em sistemas **fortemente acoplados**, há **uma única memória principal compartilhada** por todos os processadores, enquanto em sistemas **fracamente acoplados cada sistema tem sua própria memória**. Com isso, a taxa de transferência entre processadores em sistemas fortemente acoplados é bem maior que em sistemas fracamente acoplados..

# Hardware

- Unidade Central de Processamento (CPU)
- Memória
- Dispositivos de Entrada e Saída



# Unidade Central de Processamento (CPU)

- Unifica todo o sistema.
- Controla as funções realizadas em cada unidade funcional.
- É responsável pela execução de todos os programas, que obrigatoriamente deverão estar armazenados na memória principal.

Dividido em:

- **Unidade de controle (UC);**
- **Unidade lógica e aritmética (ULA);**

# Unidade de controle (UC)

É responsável por controlar as atividades de todos os componentes do computador, mediante a **emissão de pulsos** elétricos (sinais de controle) gerados por um dispositivo chamado **clock**. Esse controle pode ser exercido, por exemplo, sobre a gravação de um dado no disco ou a busca de uma instrução na memória.

# Unidade lógica e aritmética (ULA)

A ULA é responsável pela realização de **operações lógicas** (testes e comparações) e **aritméticas** (somas e subtrações).

# Velocidade de processamento

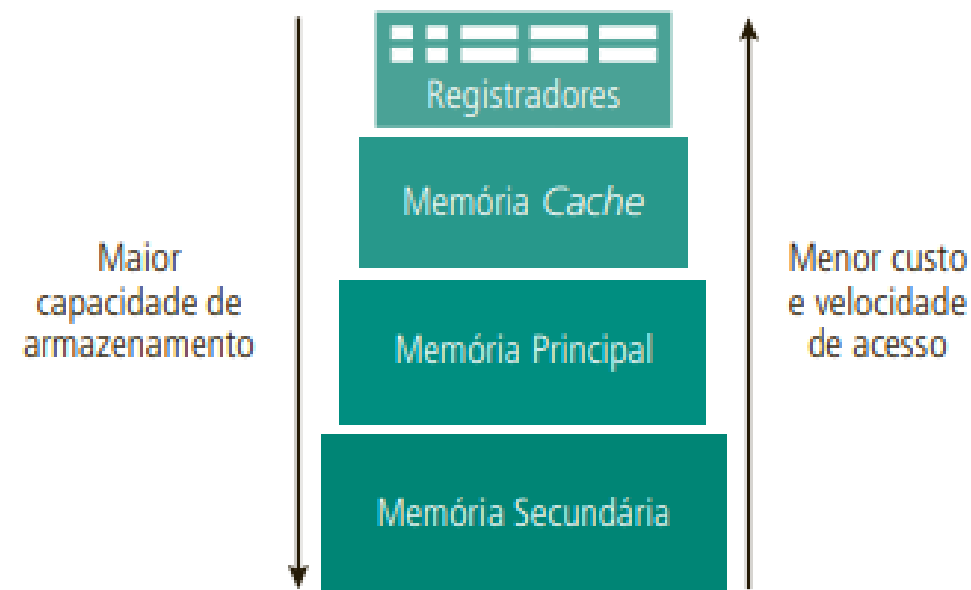
A velocidade de processamento de uma unidade central de processamento (CPU) é determinada pelo número de instruções que o processador executa por unidade de tempo, normalmente em segundos. A unidade de medida é **MIPS** (milhões de instruções por segundo).

# Clock

A transmissão de dados entre o receptor e o transmissor é controlada por um sinal de controle chamado **clock**. Este sinal é usado para sincronizar o transmissor com o receptor, isto é, para informar ao receptor que um dado está sendo transmitido. **É utilizado pela unidade de controle para a execução das instruções.**

# Memória

A memória tem por função **armazenar internamente toda informação que é manipulada pelo computador: os programas e os dados**. A memória pode ser classificada quanto à sua velocidade (ou tempo) de acesso, capacidade de armazenamento, custo e volatilidade. Em função dessas características, pode-se estabelecer uma hierarquia de tipos de memórias.



# Registradores

São dispositivos de **alta velocidade**, localizados fisicamente na unidade central de processamento, para **armazenamento temporário de dados**. O número de registradores varia em função da arquitetura de cada processador. Existem registradores de **uso específico** (com propósitos especiais) e de **uso geral**.

# Registradores

Os registradores de uso específico são:

**Contador de instruções** - responsável por armazenar o endereço da próxima instrução que a unidade central de processamento deverá executar. Toda vez que uma instrução já está sendo processada o endereço da próxima instrução a ser processada é armazenado no contador de instruções;

**Apontador de pilha:** responsável por armazenar o endereço de memória do topo da pilha. Pilha é uma estrutura de dados onde o sistema mantém informações sobre tarefas que estavam sendo processadas, mas que por algum motivo tiveram que ser interrompidas;

**Registrador de estado:** responsável por armazenar informações sobre a execução do programa (status do programa). A cada instrução executada, o registrador de estado é alterado conforme o resultado gerado pela instrução



# Memória cache

É uma **memória volátil de alta velocidade**. Quando o **processador** faz referência a um dado armazenado na memória principal, **verifica antes se este dado não está armazenado na memória cache**. Ao encontrar o dado armazenado na memória cache, o processador não acessa a memória principal, diminuindo o tempo de processamento.

# Memória cache - Níveis

O nível da cache mais alto é chamado de **L1** (Level 1), com baixa capacidade de armazenamento e com altíssima velocidade de acesso. O segundo nível, **L2** (Level 2), possui maior capacidade de armazenamento, porém com velocidade de acesso inferior a L1, e assim sucessivamente. Quando a CPU necessita de uma informação da memória principal, primeiramente verifica a cache L1, caso não ache, segue para a cache L2, assim por diante. Se não encontrar em nenhum dos níveis, busca o dado na memória principal.

# Memória principal

É a memória **responsável pelo armazenamento dos programas que estão sendo executados pela CPU em um certo instante**, bem como dos dados utilizados pelos programas em execução. Para que um programa possa ser executado pela CPU é necessário que ele seja **previamente armazenado na memória principal**. Existem ainda dois tipos de memória: **ROM** (read Only memory – memória somente leitura) e **RAM** (random access memory – memória de acesso randômico).

A principal característica da ROM é que o seu conteúdo não é perdido quando cortamos a sua fonte de alimentação e por isso é utilizada para manter os programas iniciais do computador.

# Memória principal

Um programa armazenado em ROM recebe o nome de **firmware**. Na memória ROM do micro há basicamente três programas (firmware) principais:

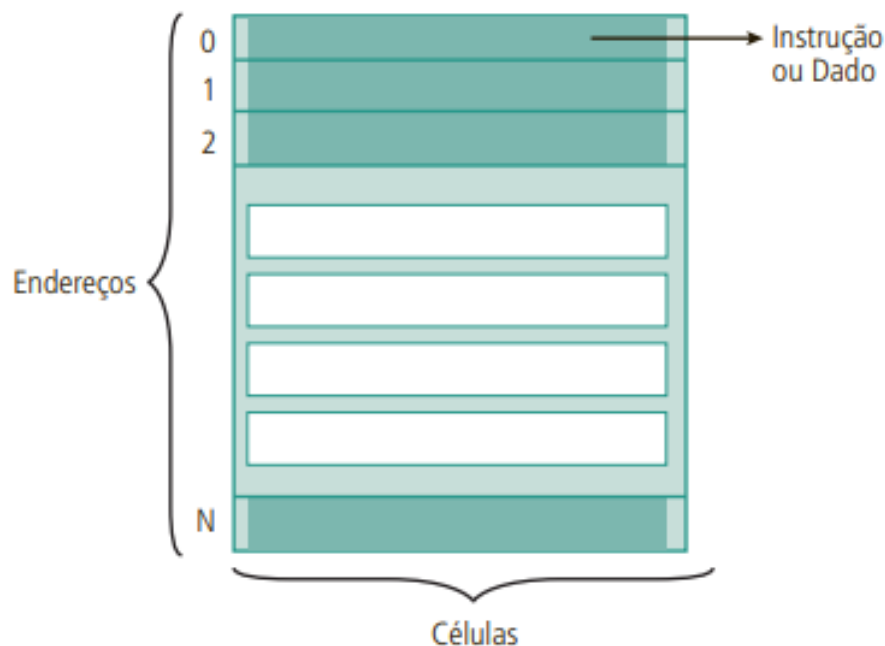
- **BIOS** (Basic Input/Output System, Sistema básico de Entrada/Saída);
- **POST** (Power On Self Test, Autoteste ao Ligar);
- **Setup** (programa que permite alterar vários itens da configuração do computador).

# Memória principal

- **Composta por unidades de acesso chamadas células**, cada uma capaz de armazenar um determinado número de bits.
- Cada célula tem um **endereço** (posição da célula dentro da memória
- Quando um programa deseja ler ou escrever um dado em uma célula, deve primeiro especificar qual o endereço de memória desejado.
- O **endereço da célula a ser acessada fica armazenado em um registrador** denominado **registrador de endereço de memória**.

# Memória principal

O número de células endereçadas na memória principal é limitado pelo tamanho do registrador de endereço. No caso de o registrador possuir  $n$  bits, a memória poderá endereçar  $2^n$  células.

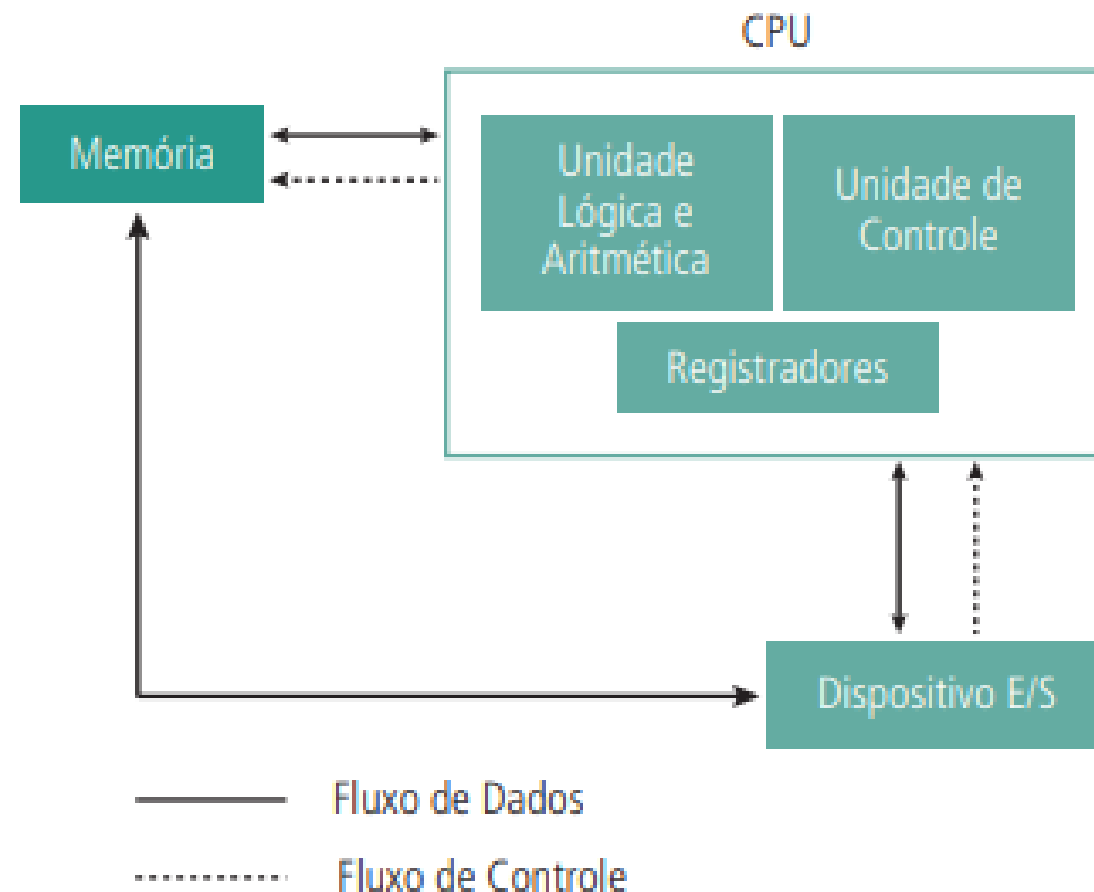


# Memória secundária

Memória secundária é um termo genérico para designar diversos componentes que **permitem gravar e ler dados permanentes**. O seu **acesso é lento**, se comparada com as memórias cache ou principal, porém relativamente apresentam **custo mais baixo e capacidade de armazenamento superior**. Exemplos de memória secundária são as fitas magnéticas, discos rígidos (HDs), CDs, DVDs, etc.

# Barramento

A CPU, a memória principal e os dispositivos de E/S são interligados através de **linhas de comunicação denominadas barramentos**, barras ou vias. Um barramento é um conjunto de fios paralelos (linhas de comunicação), por onde trafegam informações, como dados, endereços ou sinais de controle.





# Barramento

Existem três tipos de barramentos:

- **Barramento de dados:** transmite informações entre a memória principal e a unidade central de processamento;
- **Barramento de endereços:** utilizado pela unidade central de processamento para especificar o endereço da célula de memória que será acessada;
- **Barramento de controle:** é por onde a unidade central de processamento envia os pulsos de controle relativos às operações de leitura e gravação.

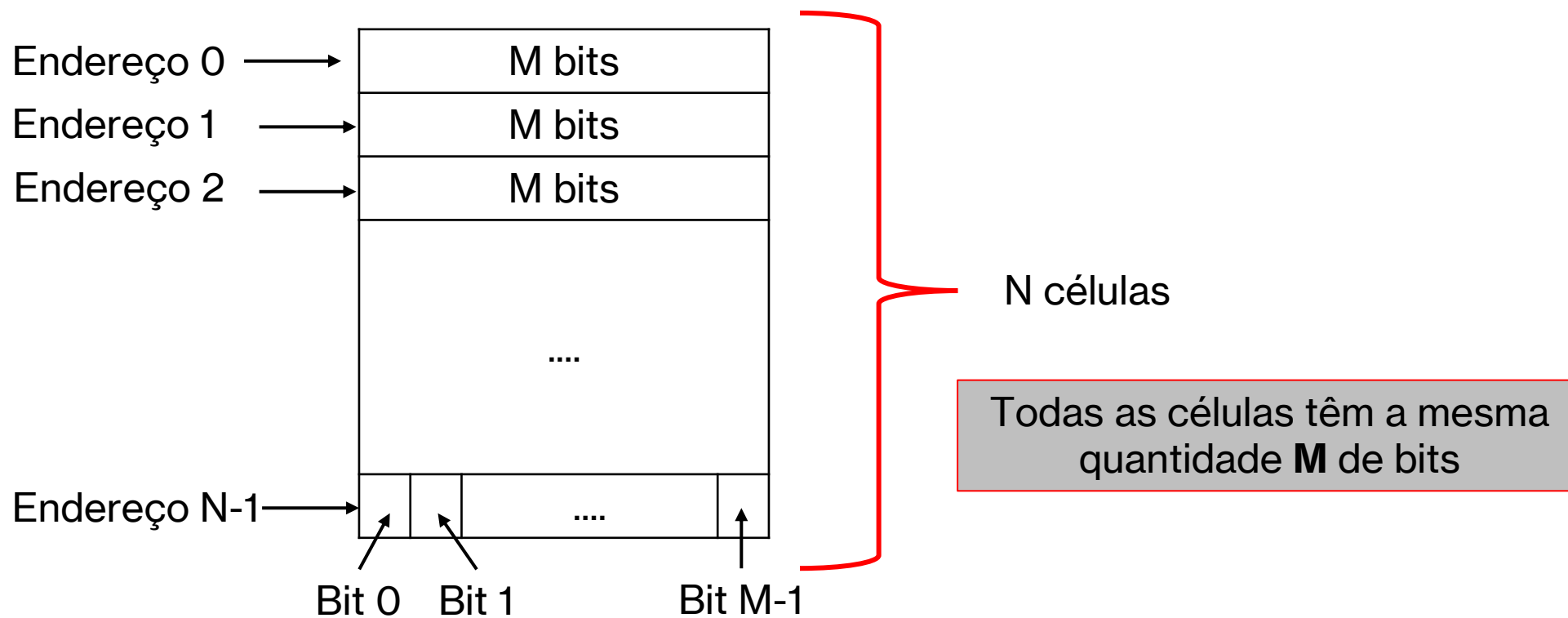
# Pipelining

O conceito de processamento pipeline é a **divisão de uma tarefa em uma sequência de subtarefas**. O processador, por meio de suas várias unidades funcionais pipeline, funciona de forma a permitir que, **enquanto uma instrução se encontra na fase de execução, uma outra instrução possa estar na fase de busca**.

# Pesquisa

**Realizar uma pesquisa sobre as características e funcionamento de softwares: Tradutores, Compiladores, Montadores, Linker, Loader e Depurador.**

# Memória Principal - Organização



# Memória Principal - Organização

0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

End 0							
End 1							
End 2							
End 3							
End 4							
End 5							
End 6							
End 7							
End 8							
End 9							
End 10							
End 11							
End 12							
End 13							
End 14							
End 15							

8 bits  
Célula  
16 células X 8 bits = 128 bits

End 0															
End 1															
End 2															
End 3															
End 4															
End 5															
End 6															
End 7															

16 bits  
Célula  
8 células X 16 bits = 128 bits

End 0				
End 1				
End 2				
End 3				
End 4				
End 5				
End 6				
End 7				
End 8				
End 9				
End 10				
End 11				

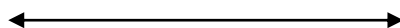
4 bits  
Célula  
12 células X 4 bits = 48 bits

Precisamos  
de 4 bits  
para 16  
endereços

Sendo N o número de endereços da RAM o tamanho do barramento de endereços será  $\log_2 N$

# Memória Principal - Organização

0	0	0	0	End 0
0	0	0	1	End 1
0	0	1	0	End 2
0	0	1	1	End 3
0	1	0	0	End 4
0	1	0	1	End 5
0	1	1	0	End 6
0	1	1	1	End 7
1	0	0	0	End 8
1	0	0	1	End 9
1	0	1	0	End 10
1	0	1	1	End 11
1	1	0	0	End 12
1	1	0	1	End 13
1	1	1	0	End 14
1	1	1	1	End 15



8 bits  
Célula

16 células X 8 bits = 128 bits

Precisamos  
de 4 bits  
para 16  
endereços

Para sabermos a  
quantidade de bits para  
endereçarmos 16  
endereços podemos fazer  
 **$\log_2 16 = \log_2 2^4 = 4 \text{ bits}$**

**Portanto o barramento de  
endereçamento para essa  
memória precisa de 4 bits**

Sendo N o número de endereços da RAM o tamanho do  
barramento de endereços será  **$\log_2 N$**

# Relembrando Registradores

- Registradores de propósito específico:

PC (*program counter*): o contador de programa, que indica o endereço da próxima instrução a ser executada pela CPU

IR (*instruction register*): o registrador de instrução, que contém a instrução a ser executada

MAR (*memory address register*): registrador de endereço de memória, que contém um endereço da RAM

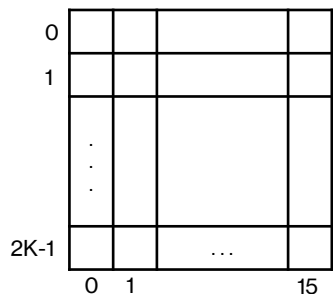
MBR (*memory buffer register*): registrador de dados da memória, que contém uma palavra da memória

# Exemplo 1

Uma memória RAM tem um espaço máximo de endereçamento de 2K. Cada célula pode armazenar 16 bits. Qual é o valor total de bits que pode ser armazenado nesta memória? Qual é o tamanho de cada endereço? Qual é o valor do maior endereço da RAM?



# Exemplo 1 - Resolução



Qual é o valor total de bits que pode ser armazenado nesta memória?

$$C = 2K * 16 = 32Kb \text{ (K bits)}$$

Qual é o tamanho de cada endereço?

Aqui queremos saber a qtde de bits para cada endereço (**MAR**)

$$MAR = \log_2 2K = \log_2 2^1 * 2^{10} = \log_2 2^{11} = 11 \text{ bits}$$

$$1K = 2^{10} = 1024$$

Qual é o valor do maior endereço da RAM?

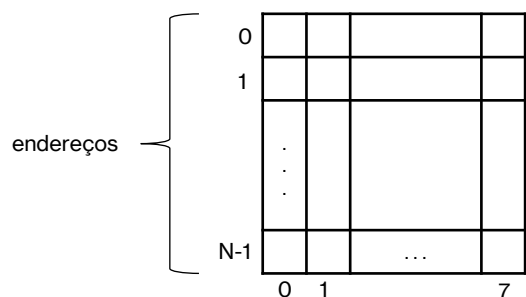
$$\text{Maior\_end} = 2K - 1 = 2048 - 1 = 2047$$

Uma memória RAM tem um espaço máximo de endereçamento de 2K. Cada célula pode armazenar 16 bits. Qual é o valor total de bits que pode ser armazenado nesta memória? Qual é o tamanho de cada endereço? Qual é o valor do maior endereço da RAM?

## Exemplo 2

Uma memória RAM é fabricada com a possibilidade de armazenar um máximo de 256Kb. Cada célula pode armazenar 8 bits. Qual é o tamanho de cada endereço e qual é o total de células que podem ser utilizadas na RAM?

# Exemplo 2 - Resolução



Capacidade da Memória

**C = 256Kb**

Lembrando que

C = num de bits por célula \* num de endereços

Então:

$256kb = 8 * (N)$

**N = 32K endereços**

$$1K = 2^{10} = 1024$$

Qual é o tamanho de cada endereço?

**MAR =  $\log_2 32K = \log_2 2^5 * 2^{10} = \log_2 2^{15} = 15$  bits**

qual é o total de células que podem ser utilizadas na RAM?

**N = 32K células**

Uma memória RAM é fabricada com a possibilidade de armazenar um máximo de **256Kb**. Cada célula pode armazenar 8 bits. Qual é o tamanho de cada endereço e qual é o total de células que podem ser utilizadas na RAM?

## Exemplo 3

Uma memória RAM é fabricada com a possibilidade de armazenar um máximo de 512M palavras. Cada célula pode armazenar 1 palavra, que possui 32 bits. Qual é o tamanho de cada endereço e qual é a capacidade total da RAM, em GB (Giga Bytes)?

# Exemplo 3 - Resolução

Capacidade da Memória

**C = 512 M palavras**

$$C = 512M * 32 = 2^9 * 2^5 = 2^{14} Mb$$

$$C = \frac{2^{14}}{2^{10}} = 2^4 \text{ Gb}$$

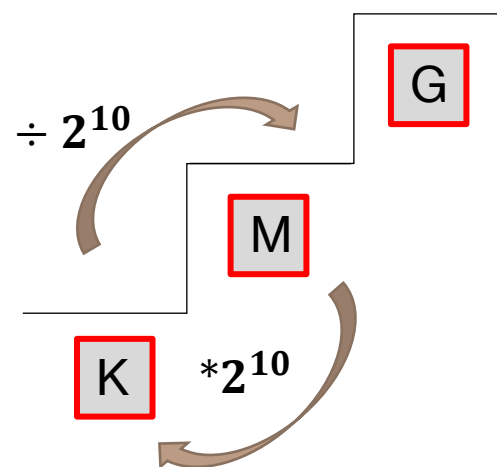
1 byte = 8 bits (de bits para bytes divide por 8)

$$C = \frac{2^4}{2^3} = 2 \text{ GB (Giga Bytes)}$$

Qual é o tamanho de cada endereço?

$$\text{MAR} = \log_2 512M = \log_2 2^9 * 2^{20} = \log_2 2^{29} = 29b \text{ bits}$$

$$1K = 2^{10} = 1024$$



Uma memória RAM é fabricada com a possibilidade de armazenar um máximo de 512M palavras. Cada célula pode armazenar 1 palavra, que possui 32 bits. Qual é o tamanho de cada endereço e qual é a capacidade total da RAM, em GB (Giga Bytes)?

## Exemplo 4

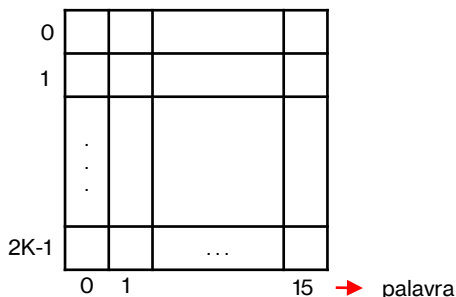
Um computador, cuja memória RAM tem uma capacidade máxima de armazenamento de 2K palavras de 16 bits cada, possui um MAR e um MBR. Assumindo que cada célula armazena uma palavra, qual é o tamanho destes registradores? Qual é o valor do maior endereço dessa RAM em binário e qual a quantidade de bytes que nela pode ser armazenada?

# Exemplo 4 - Resolução

Capacidade da Memória

**C = 2 K palavras => 32Kb**

1 palavra = 16b



O **MAR** armazena o endereço da memória, então precisamos saber o num de bits para representar os endereços. Então precisamos saber o num de endereços disponíveis. Se a capacidade é de 2K palavras e cada célula armazena 1 palavra, **teremos 2K células!**

**MBR armazena um palavra de dados**

Qual é o tamanho de cada endereço?

**MAR =  $\log_2 2K = \log_2 2^{11} = 11b$  bits**

Qual é o valor do maior endereço em binário ?

**$2K - 1 = 2048 - 1 = 2047_{(10)} = 1111111111_{(2)}$**

Qual a quantidade de bytes ?

**$32Kb = 32/8 = 4KB$  (Kilo Bytes)**

Qual o tamanho do MBR?

Se 1 palavra = 16 bits

**MBR = 16b (bits)**

Um computador, cuja memória RAM tem uma capacidade máxima de armazenamento de 2K palavras de 16 bits cada, possui um MAR e um MBR. Assumindo que cada célula armazena uma palavra, qual é o tamanho destes registradores? Qual é o valor do maior endereço dessa RAM em binário e qual a quantidade de bytes que nela pode ser armazenada?

# Exercícios

1. Um computador possui uma Memória Principal com capacidade para armazenar palavras de 16 bits em cada uma de suas  $N$  células. O Barramento de Endereços tem 12 bits de tamanho. Quantos bytes poderão ser armazenados nessa memória?



# Exercícios

2. Um microcomputador possui uma capacidade máxima de Memória Principal de 32K células, cada uma capaz de armazenar uma palavra de 8 bits. Pergunta-se:
- a) Qual é o maior endereço, em decimal, desta memória?
  - b) Qual o tamanho do Barramento de Endereços deste sistema?
  - c) Qual é o total de bits que podem existir nesta memória?

# Exercícios

3. Uma memória principal tem espaço máximo de endereçamento de 2K. Cada célula pode armazenar 16 bits. Qual o valor total de bits que pode ser armazenado nesta memória e qual o tamanho de cada endereço?
4. Uma memória principal é fabricada com a possibilidade de armazenar um máximo de 256K bits. Cada célula pode armazenar 8 bits. Qual é o tamanho de cada endereço e qual é o total de células que podem ser utilizadas naquela memória?