

---

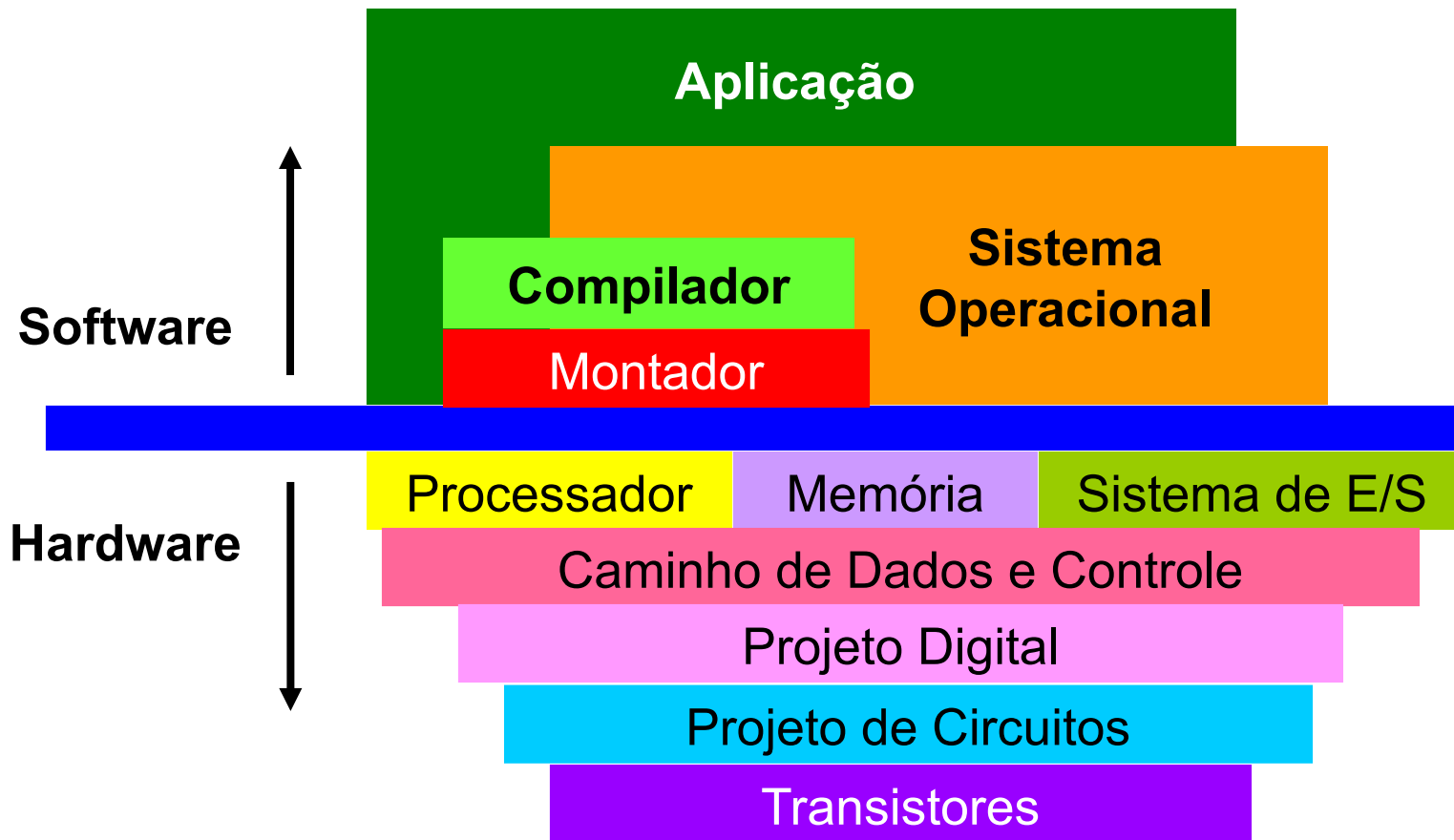
# Organização de Computadores

---

*Luciano L. Caimi*

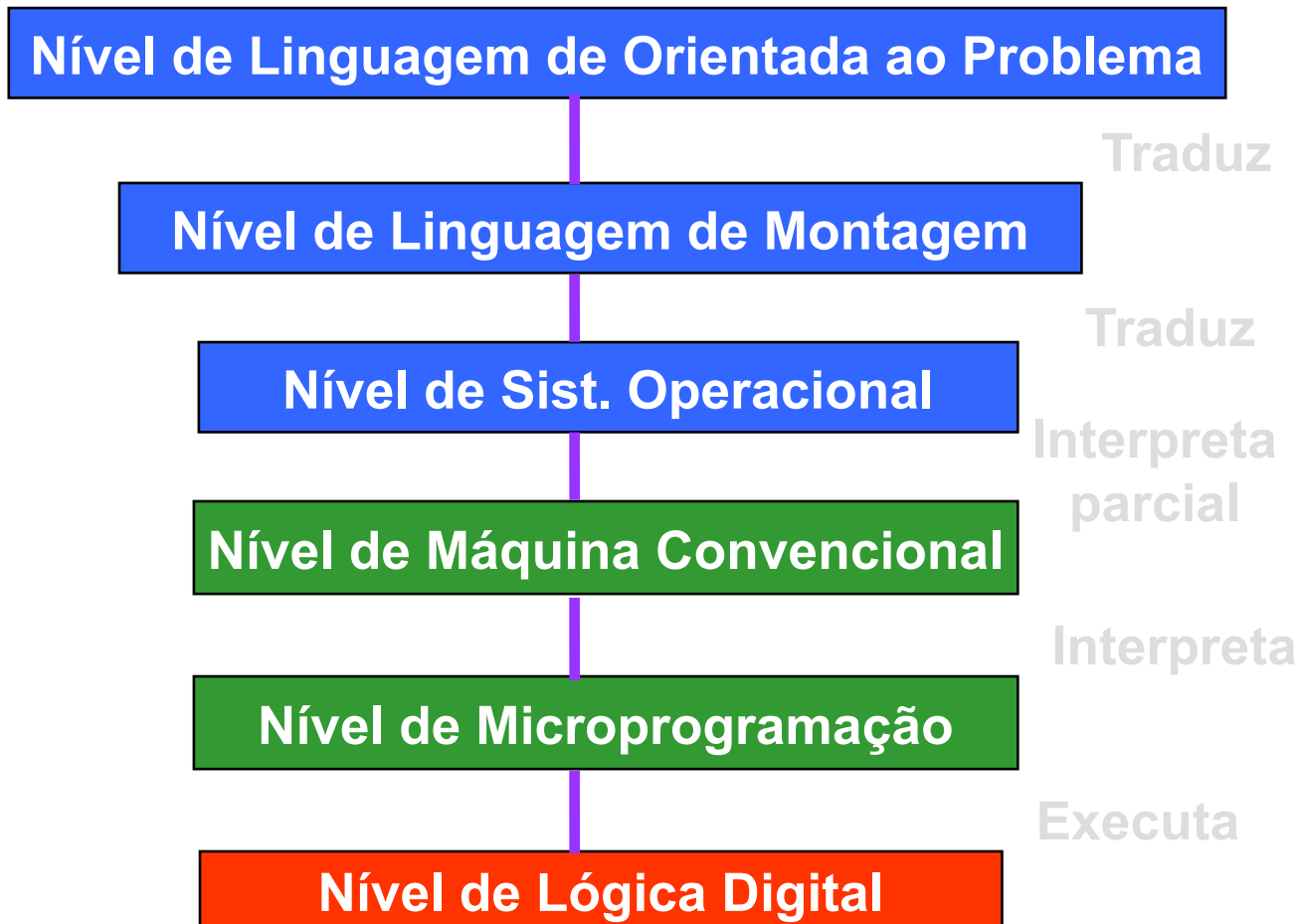
`lcaimi@uffs.edu.br`

# Motivação

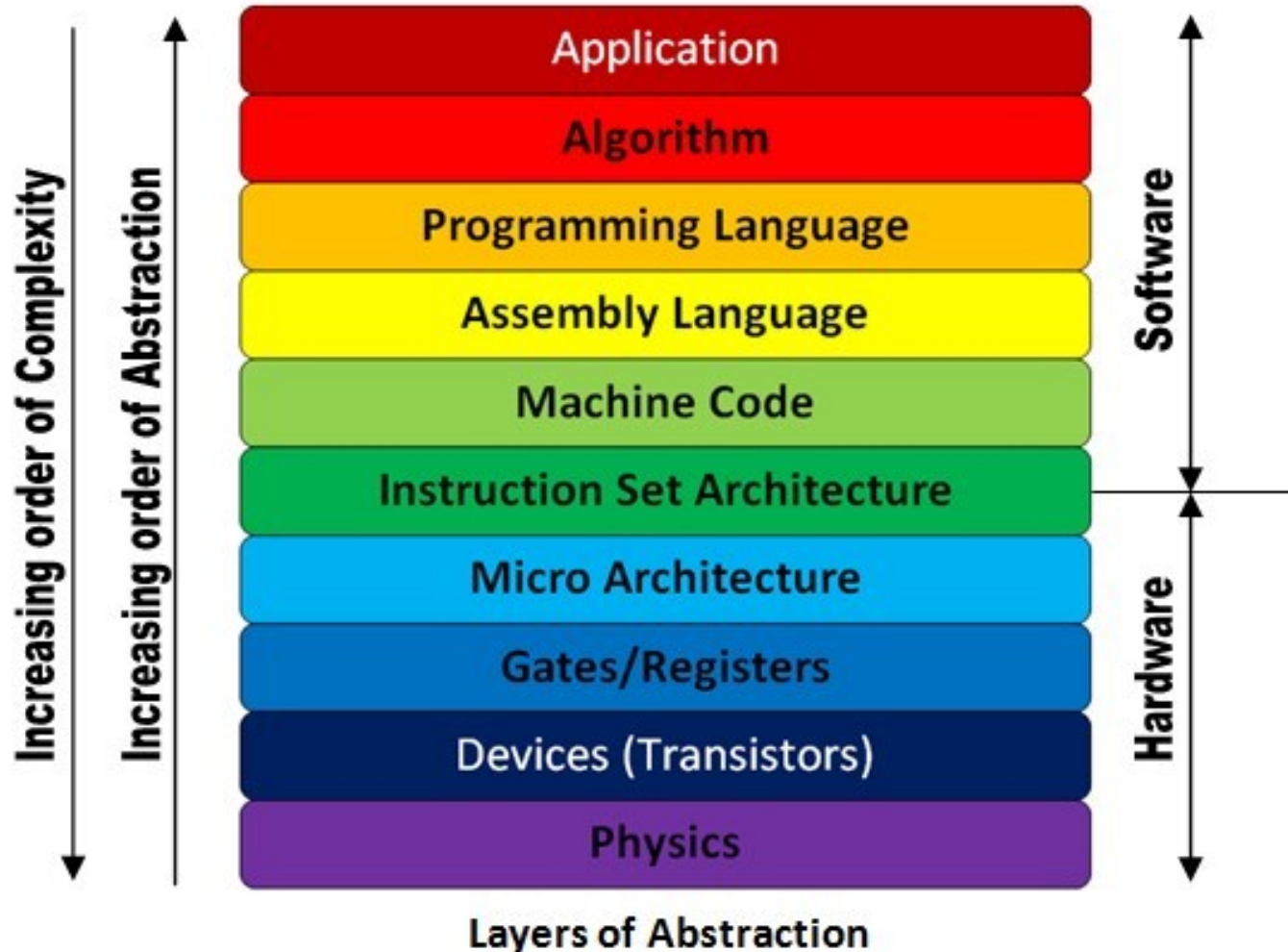


# Introdução

---



# Introdução



# Introdução

---

- Temos uma **máquina real**
- Demais níveis configuram-se **máquinas virtuais:**
  - ~ Através de Interpretação ou Tradução
  - ~ solicitam serviços a níveis inferiores
  - ~ podem ser implementados por software ou hardware

# Introdução

---

## Arquitetura Multinível

- **Sem divisão lógica**
  - ~ Difícil compreensão
  - ~ Sem estruturação
- **Com divisão lógica**
  - ~ Facilidade para compreender
  - ~ Nível inferior da suporte ao nível superior
  - ~ Nível acima oferece recurso não suportado pelo nível abaixo

# Introdução

## Níveis de Programação

Linguagem de alto nível

```
Swap (int v[], int k)
{
    int temp;
    temp = v[k];
    v[k] = v[k+1];
    v[k+1] = temp;
}
```

**Compilador**

Linguagem Assembly

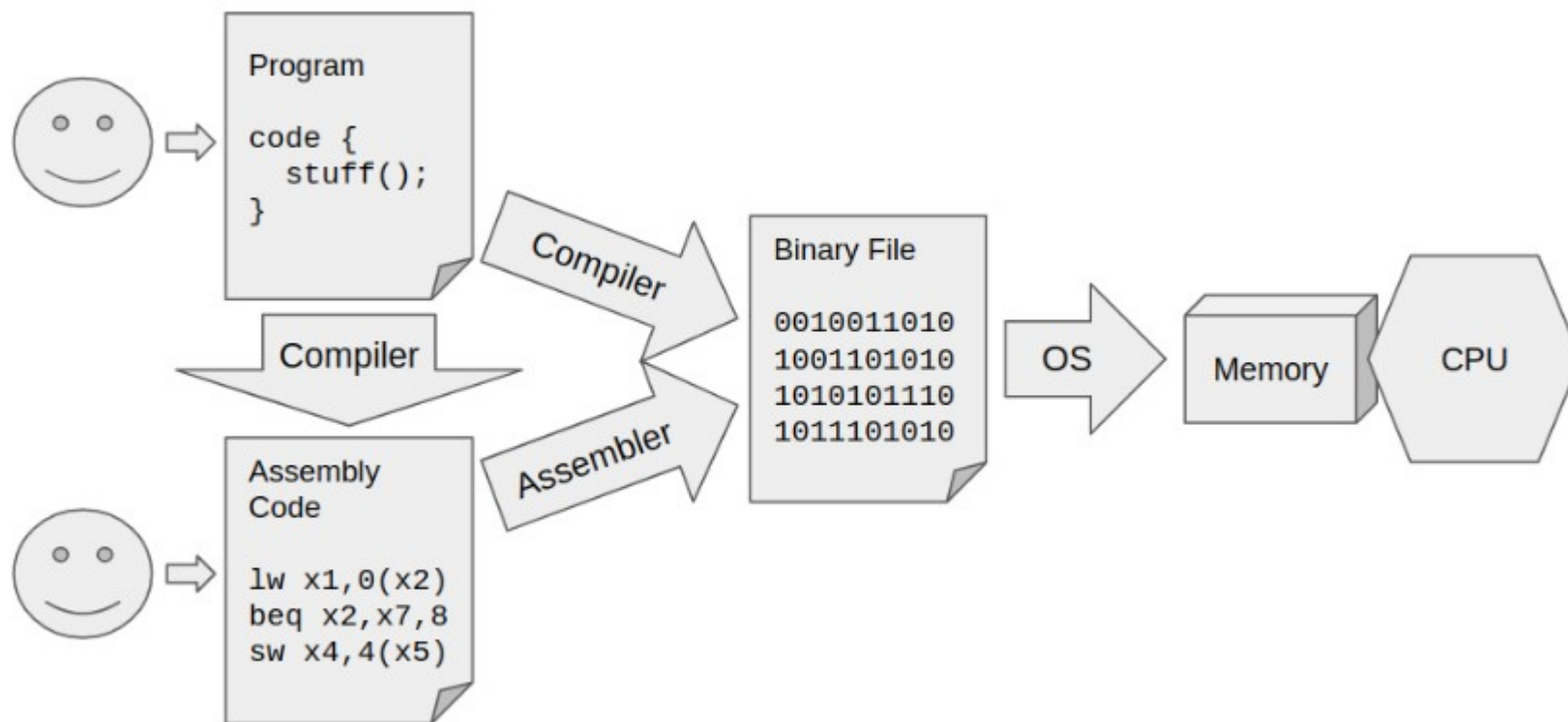
```
Swap: muli $2,$5,4
      add  $2,$4,$2
      lw   $15,0($2)
      lw   $16,4($2)
      sw   $16,0($2)
      sw   $15,4($2)
      jr   $31
```

**Montador**

Linguagem de Máquina

```
000000001010000100000000000011000
00000000100011100001100000100001
10001100011000100000000000000000
100011001111001000000000000000100
10101100111100100000000000000000
101011000110001000000000000000100
0000001111100000000000000000001000
```

# Introdução





# Introdução

Linguagem de 4a. Geração

Linguagem de Alto Nível

Linguagem Assembly (Baixo Nível)

Linguagem de Máquina



Portabilidade  
Facilidade  
Velocidade  
de Projeto



Desempenho  
Flexibilidade  
Custo de  
projeto

# Introdução

---



**Aplicativos**

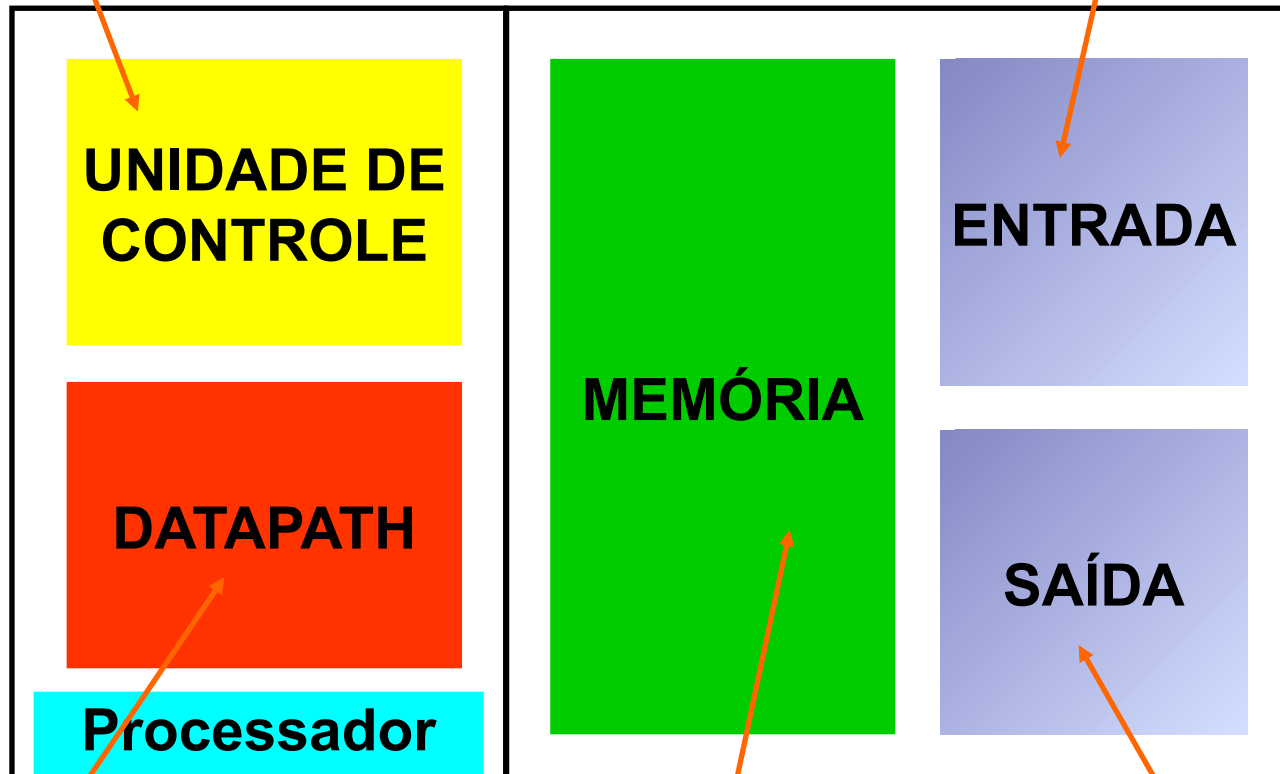
**Software de Sistema**

**Hardware**

# Introdução

Informa aos componentes o que fazer

Escreve dados na memória

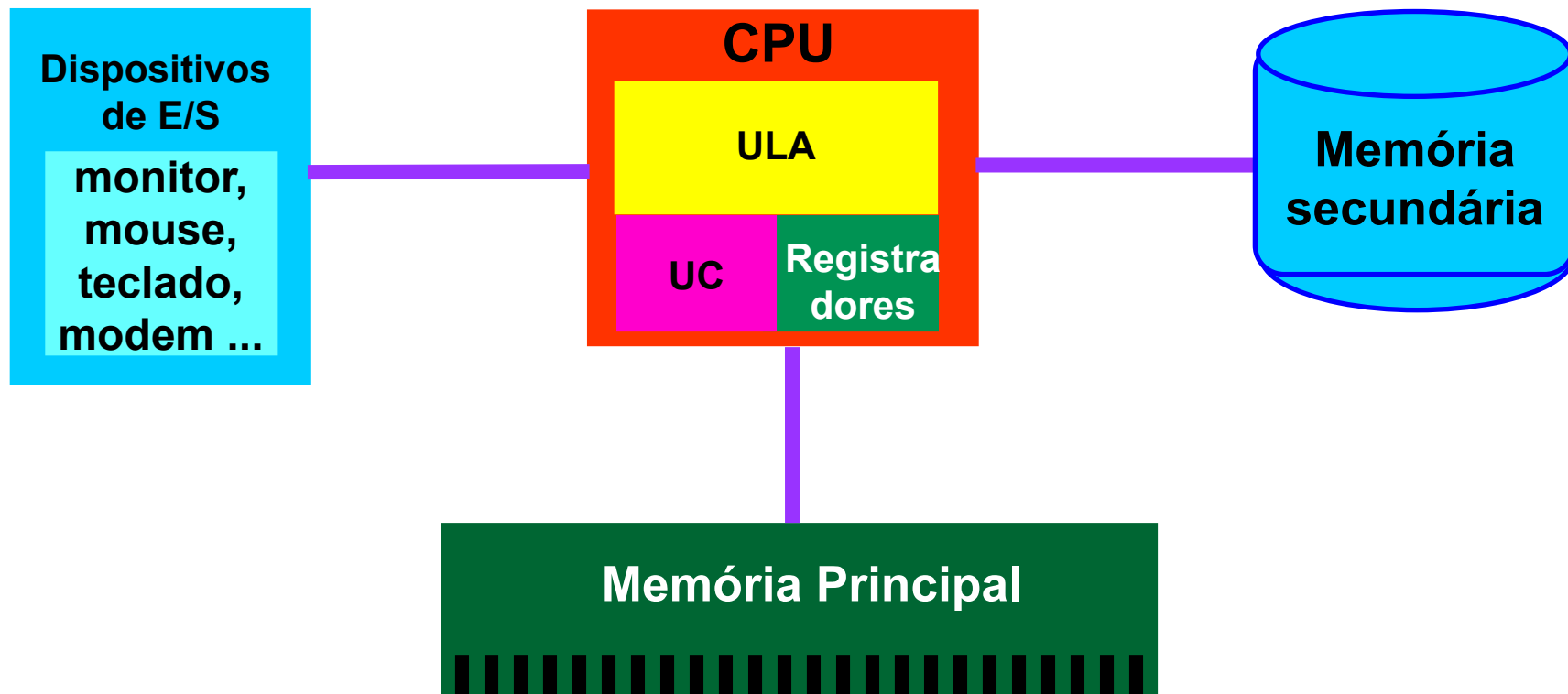


Realiza as operações lógicas e aritméticas

Armazena os programas em execução e seus dados

Lê dados da memória

# Introdução



# Introdução

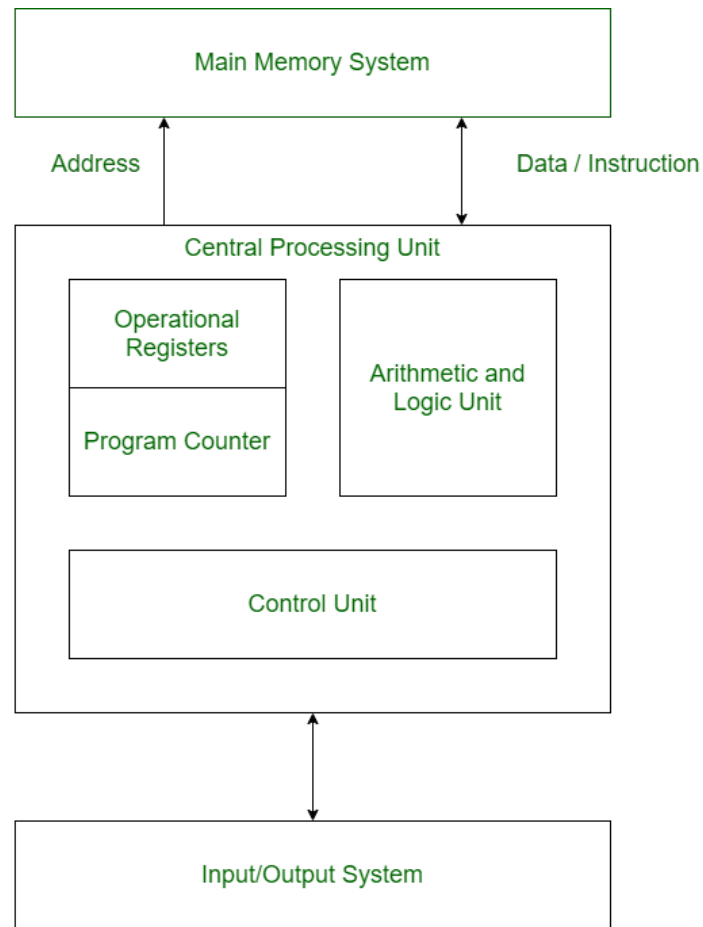
---

## 4 Proposições de Von Neumann:

- **Unidades funcionais**
  - CPU - Unidade de Controle, ULA, Registradores
  - Memória Principal - instruções e dados
  - Entrada e Saída
- **Aritmética binária**
- **Ciclo de instrução repetitivo**
- **Programa armazenado em memória (instruções e dados)**

# Introdução

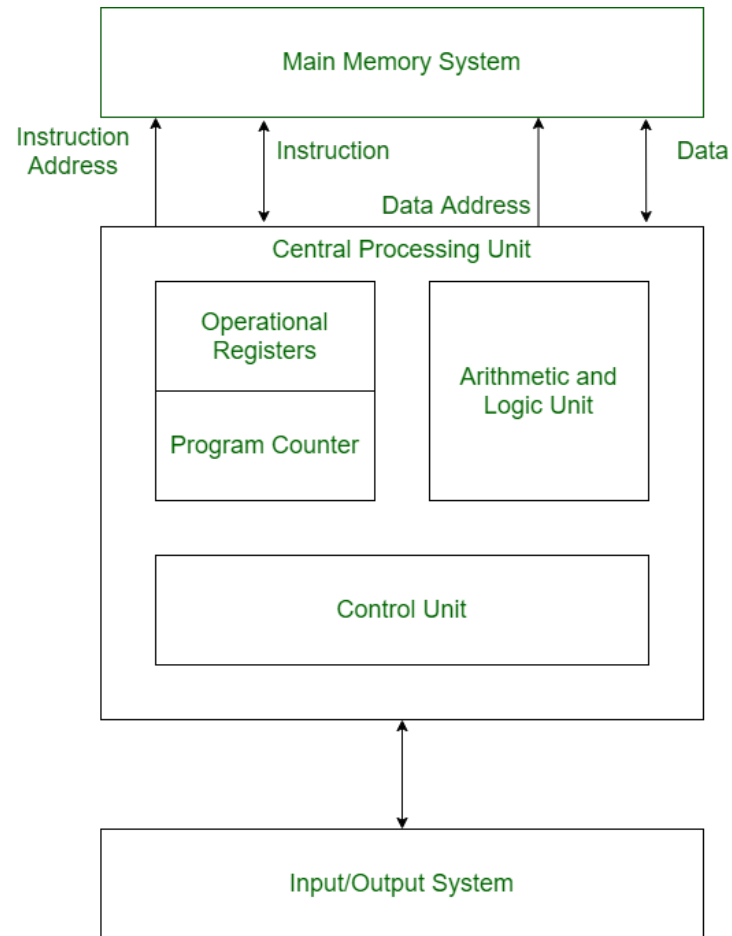
## Arquitetura de Von Neumann



Gargalo!!!  
(bottleneck)

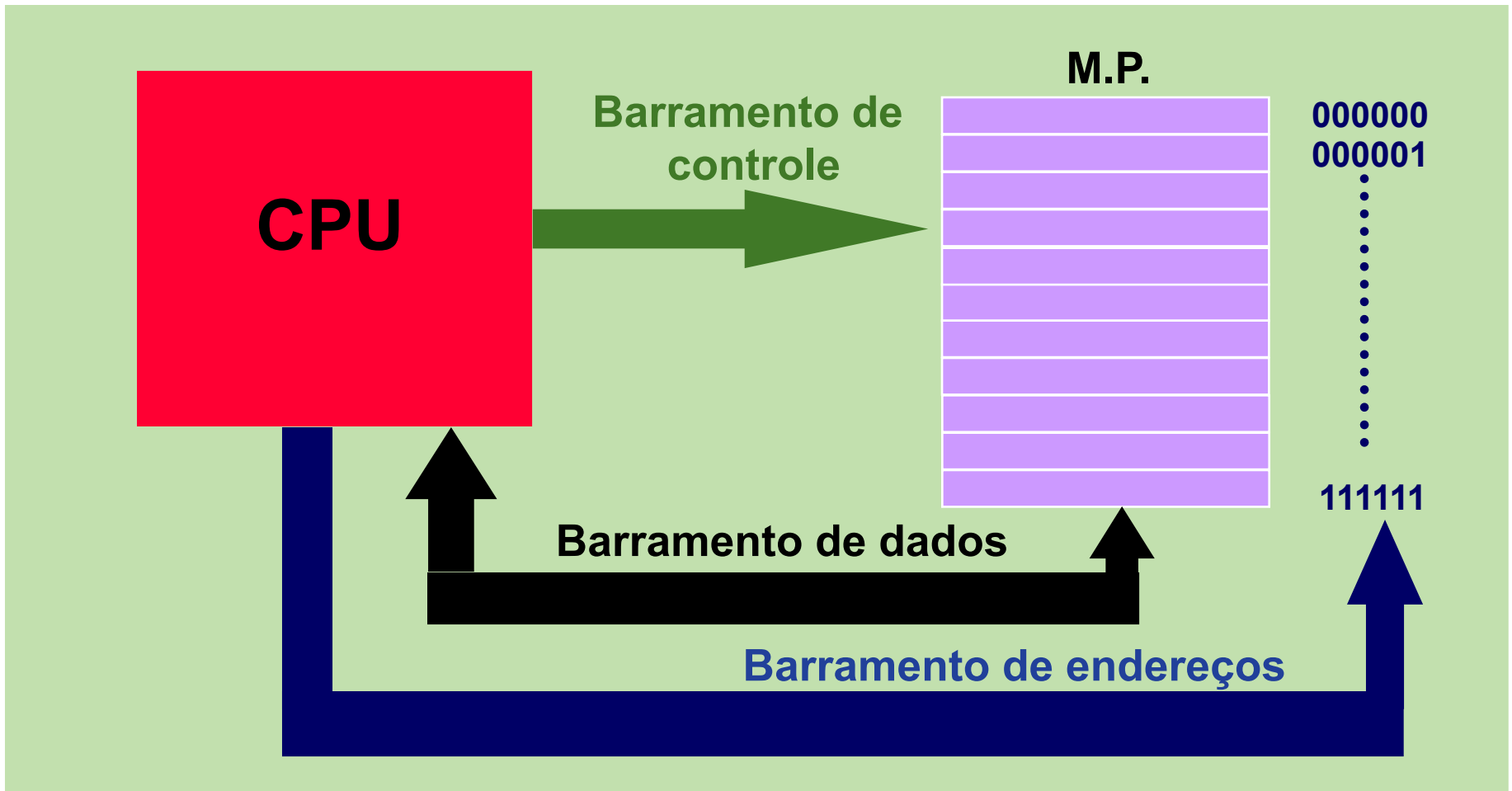
# Introdução

## Arquitetura Harvard



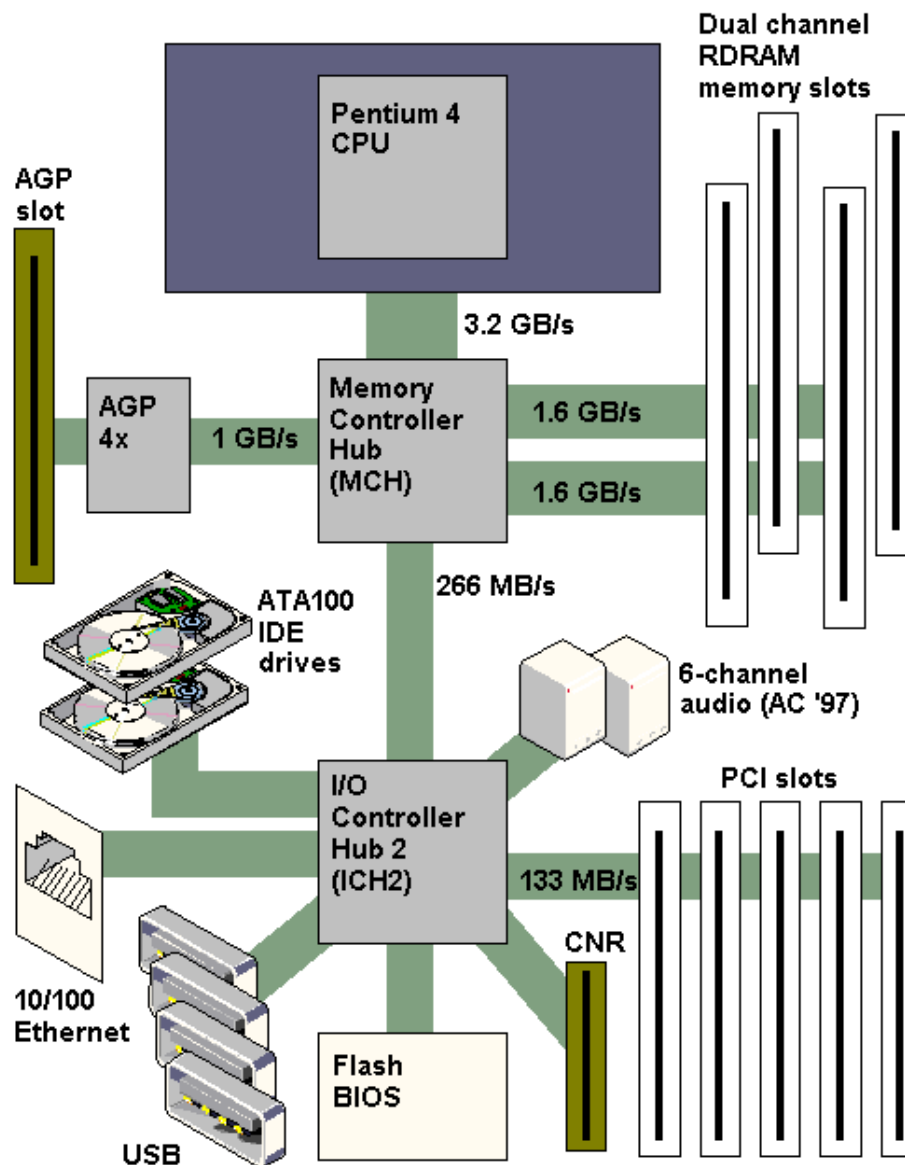
# Introdução

## Barramento local (Front-Side Bus)

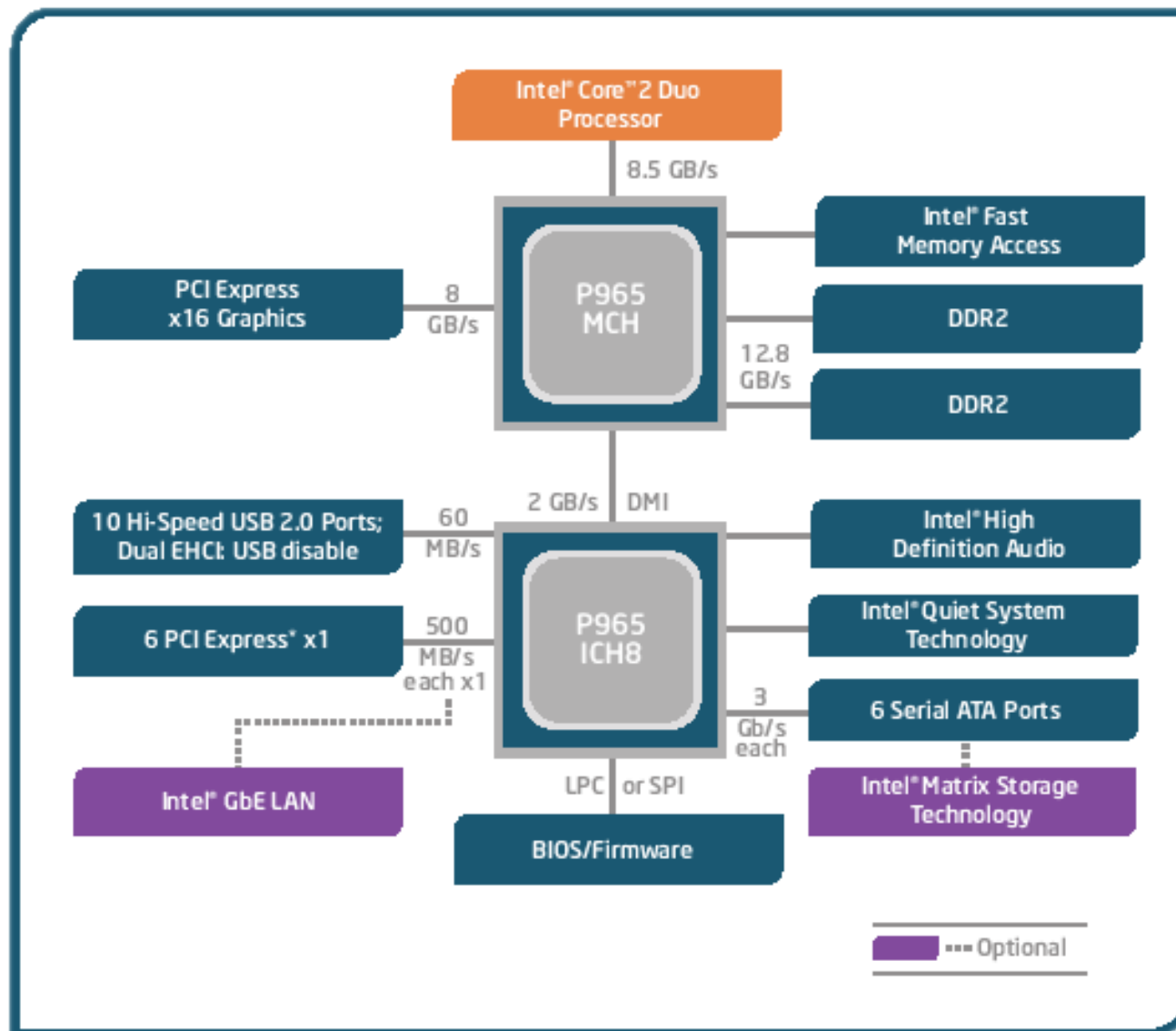




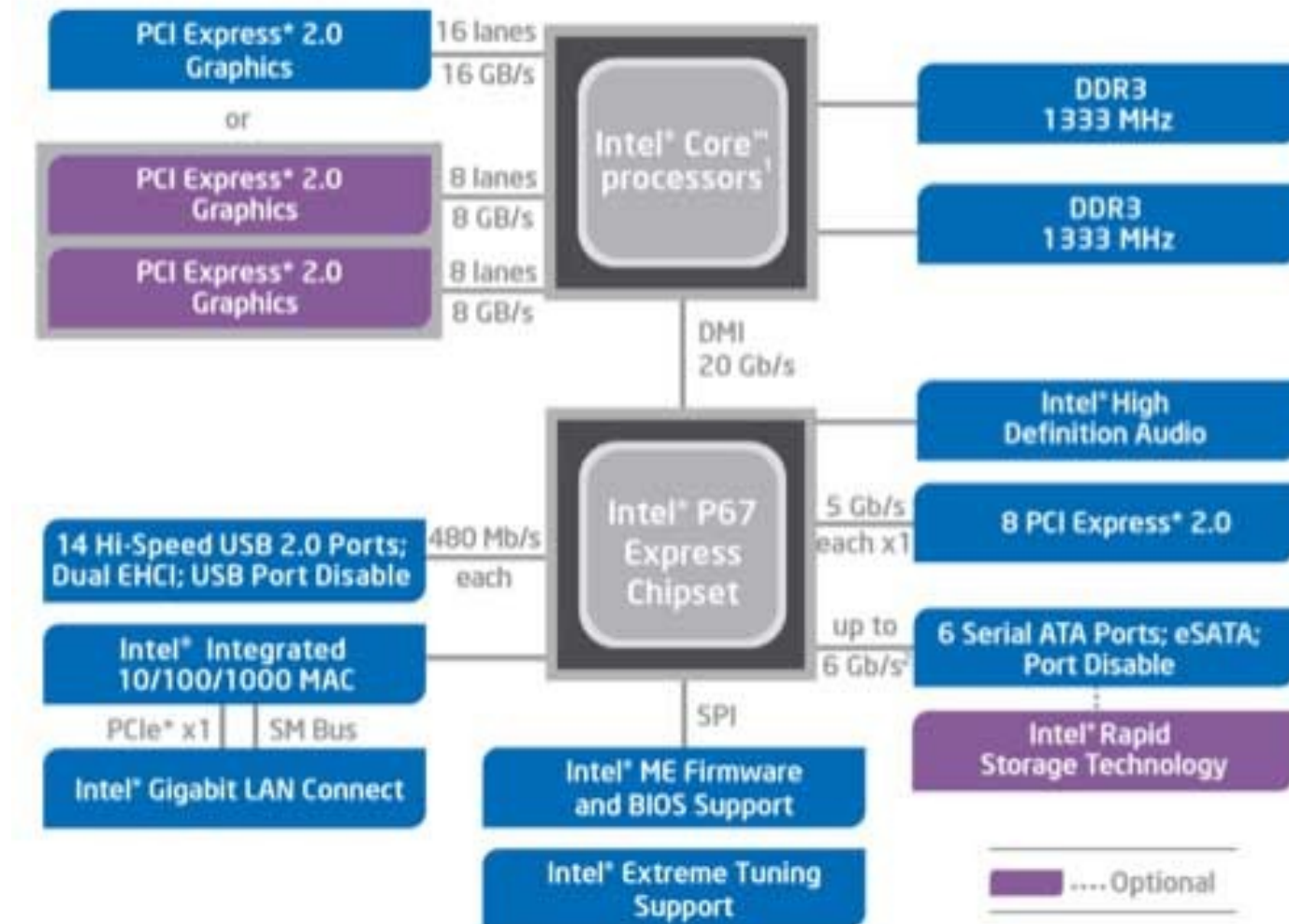
# Introdução



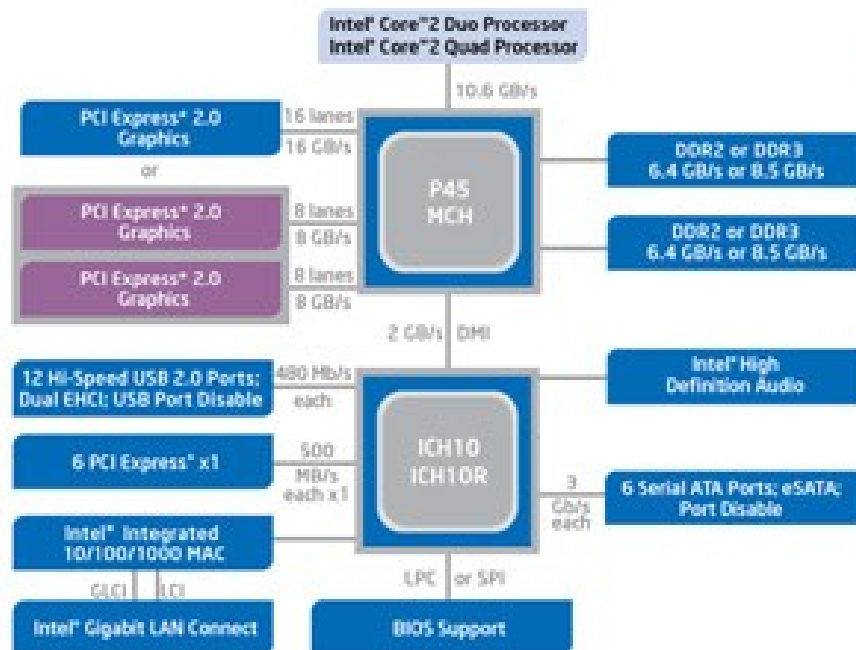
# Introdução



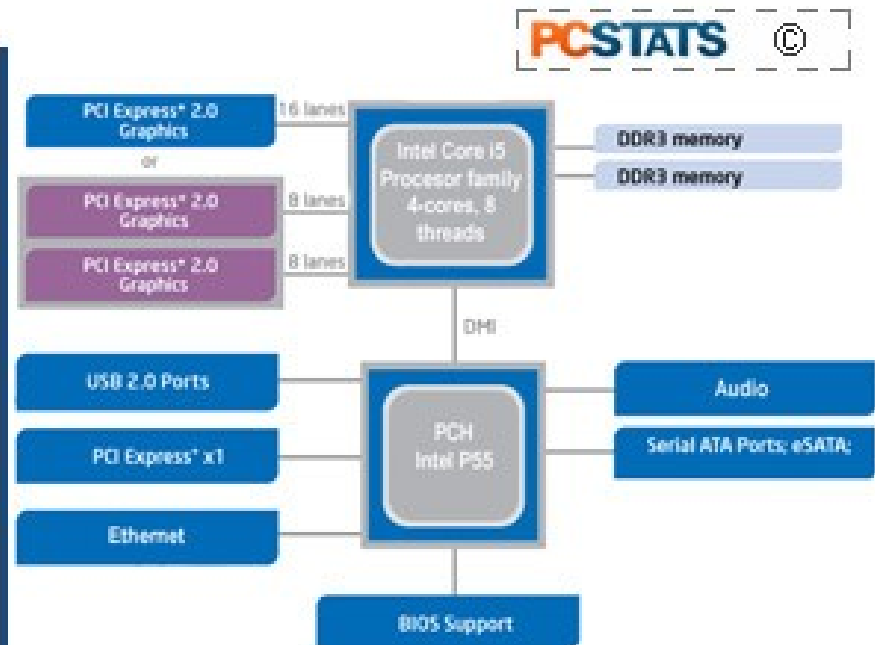
# Introdução



# Introdução

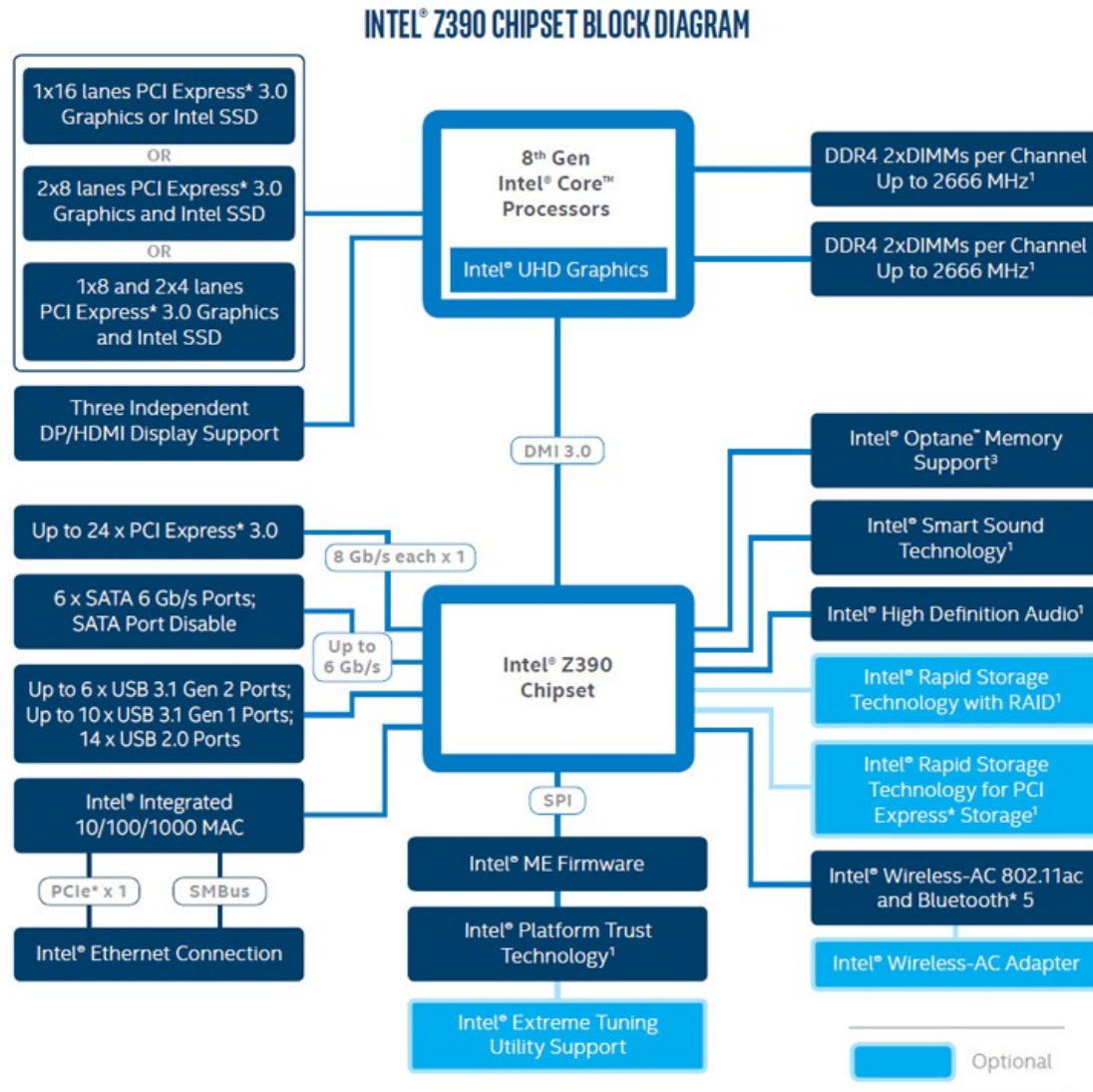


3-Chip Architecture  
(old)

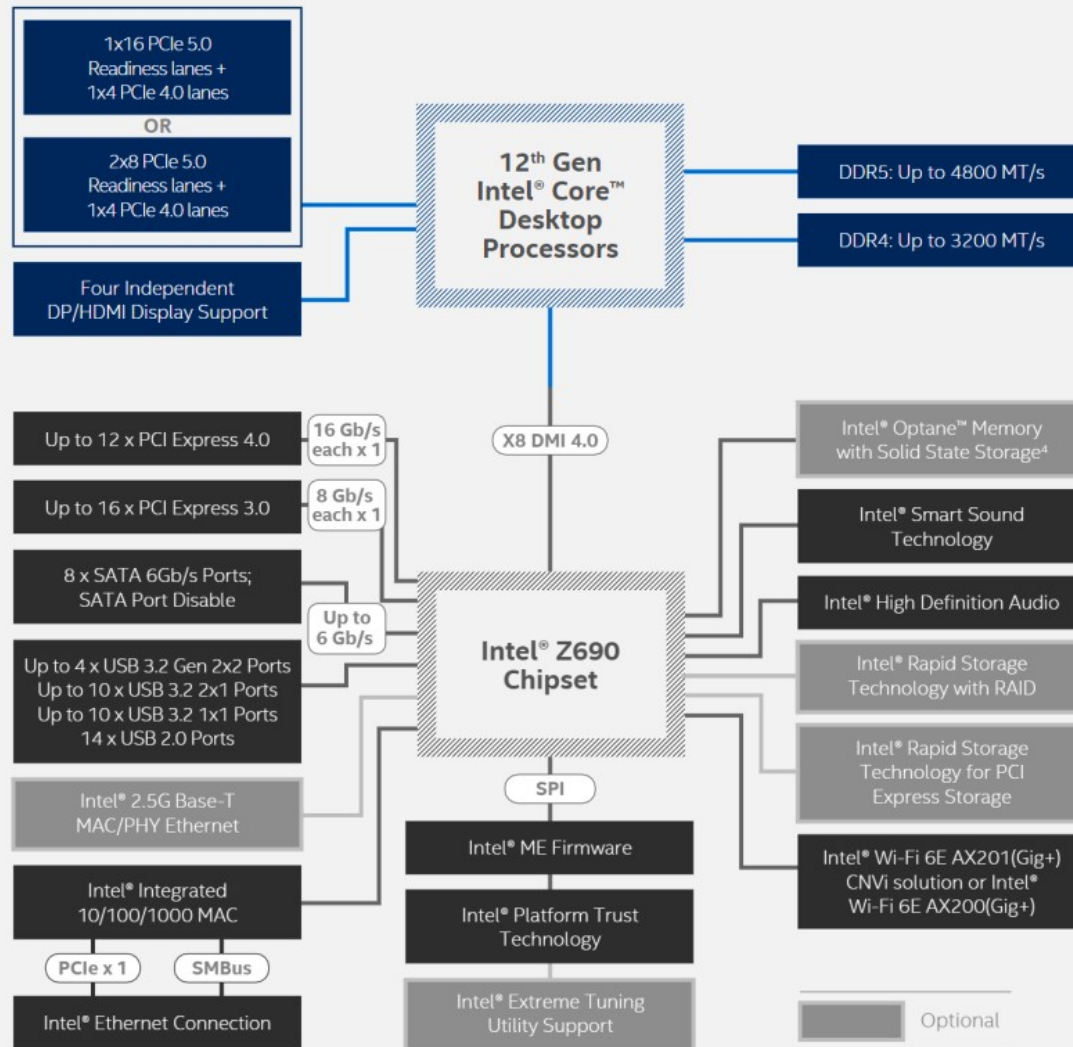


2-Chip Architecture  
(new)

# Introdução

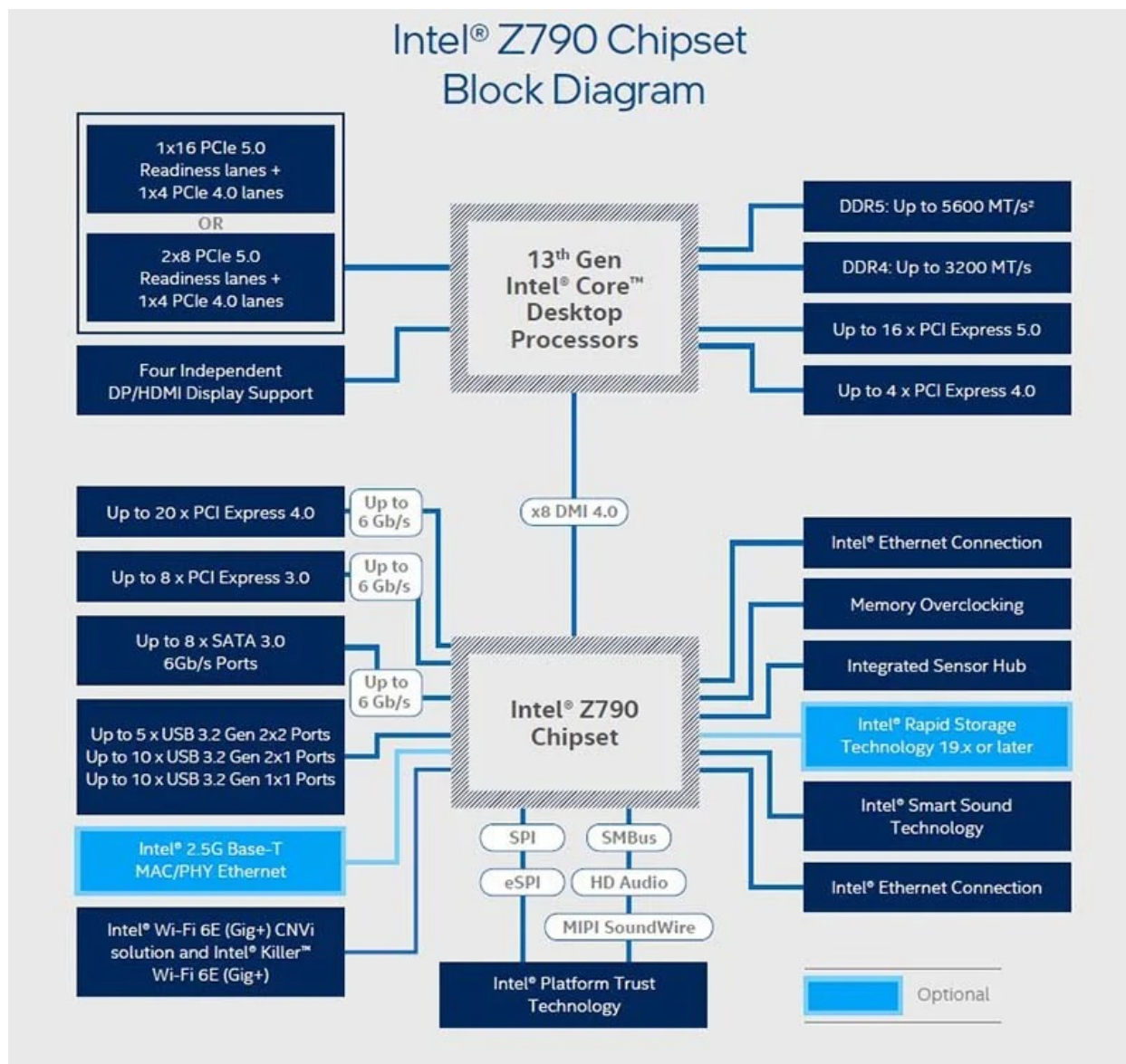


# Introdução



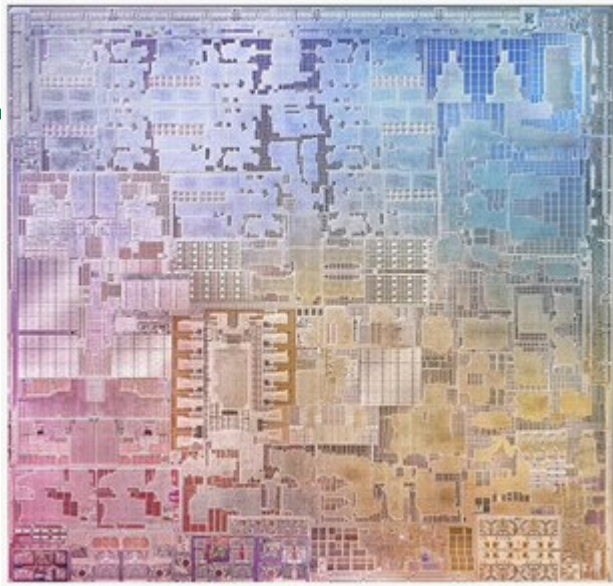
Intel® Z690 Chipset  
Block Diagram

# Introdução





# Intro



Apple M1

