

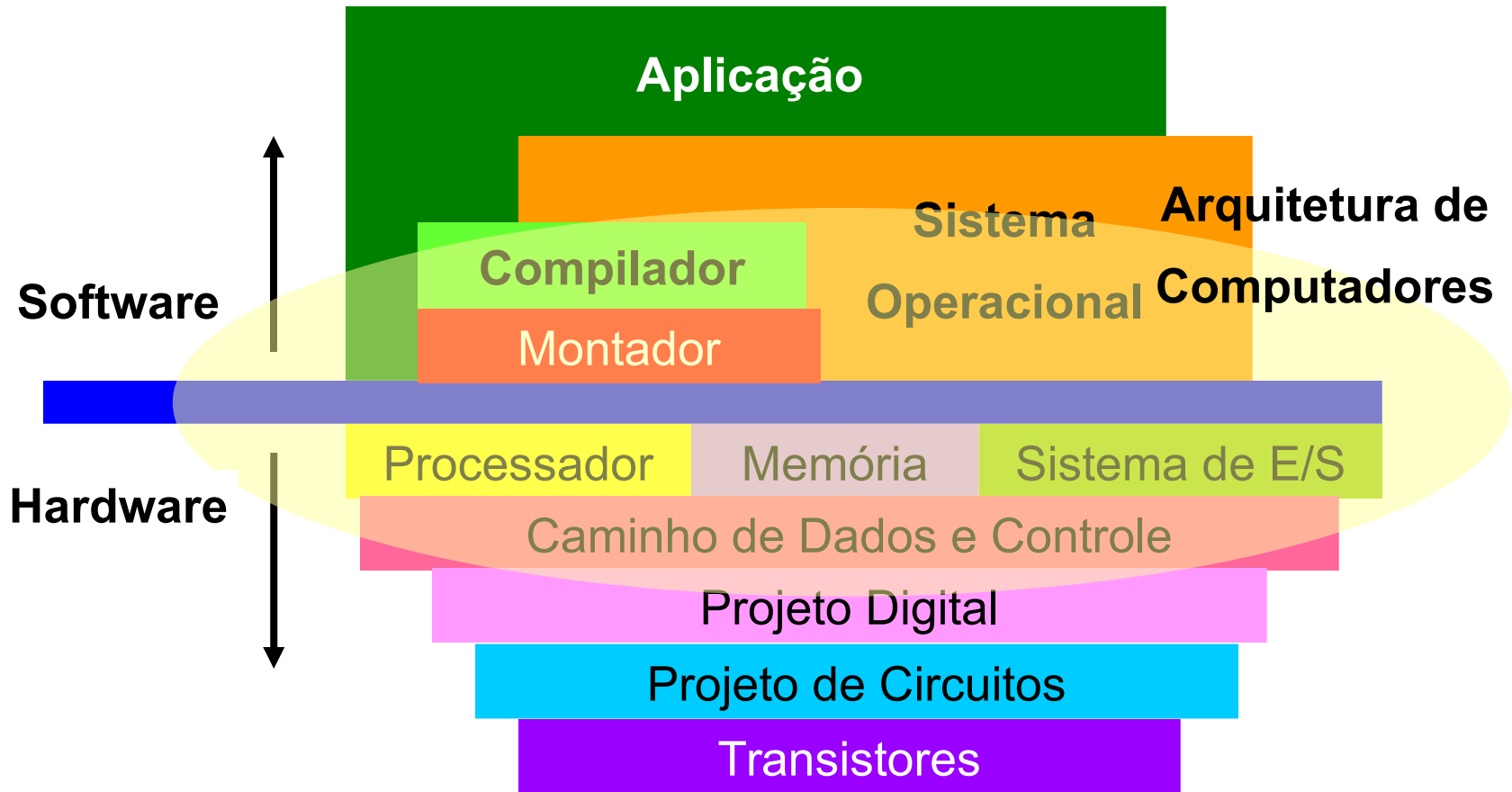


Organização de Computadores

Luciano L. Caimi
lcaimi@uffrs.edu.br



Motivação



Introdução

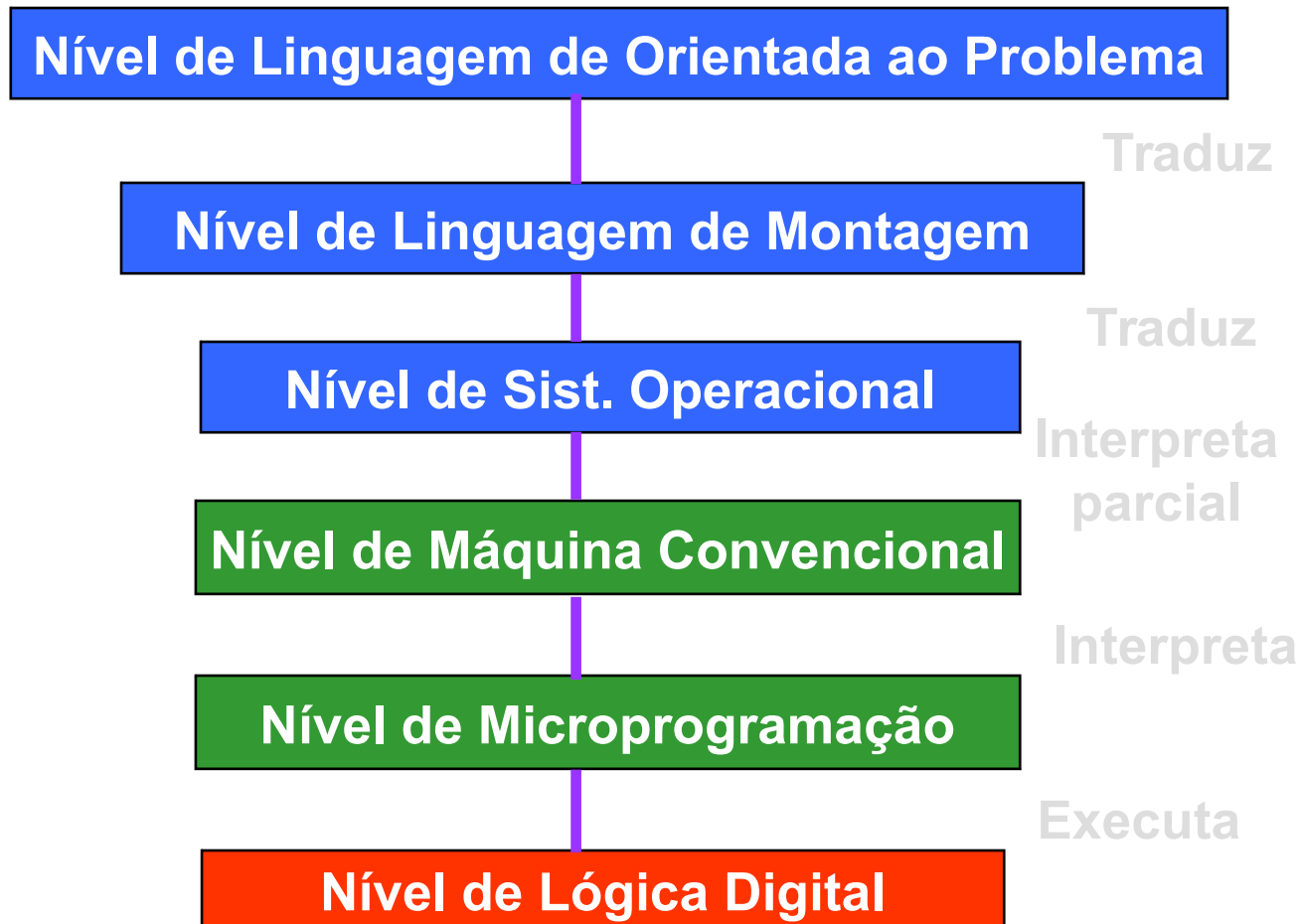


Arquitetura Multinível

- **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





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

Níveis de Programação

```
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

```
00000000101000010000000000011000
00000000100011100001100000100001
10001100011000100000000000000000
10001100111100100000000000000100
10101100111100100000000000000000
10101100011000100000000000000100
00000011111000000000000000001000
```



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

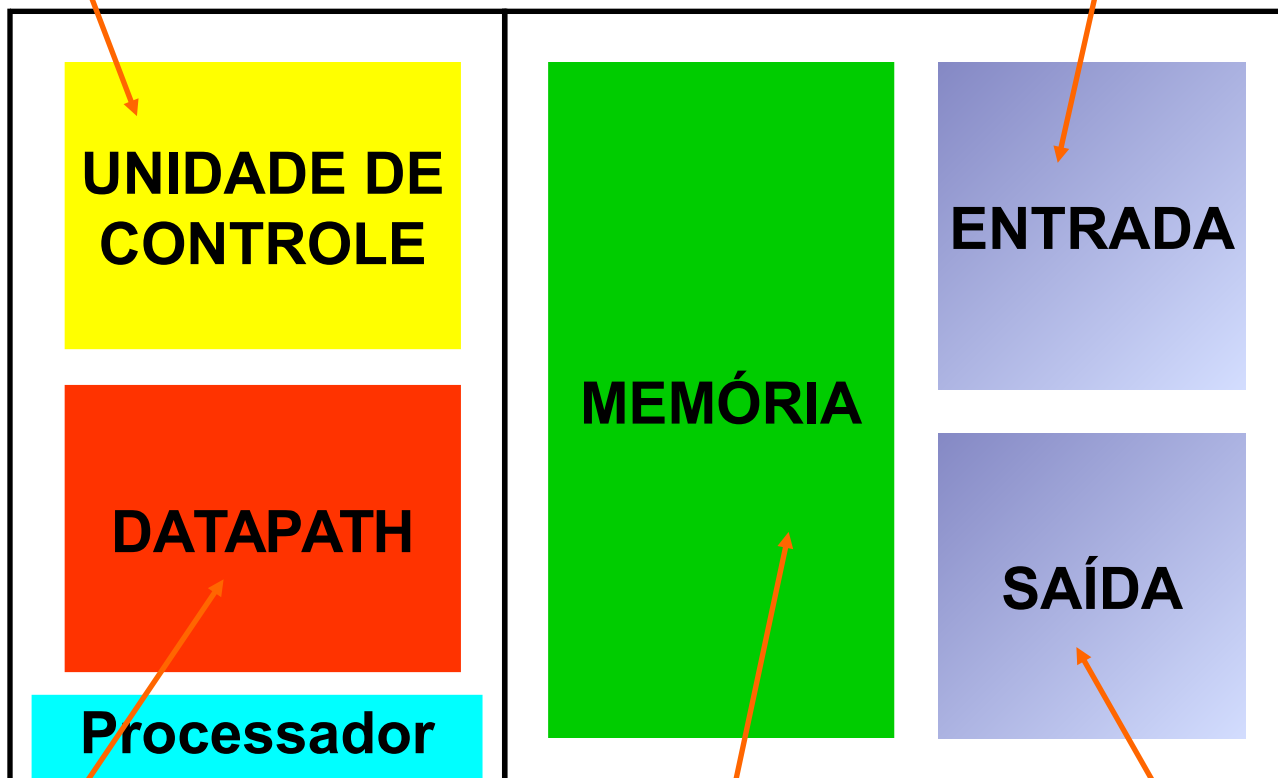
Hardware

Introdução



Informa aos componentes o que fazer

Escreve dados na memória



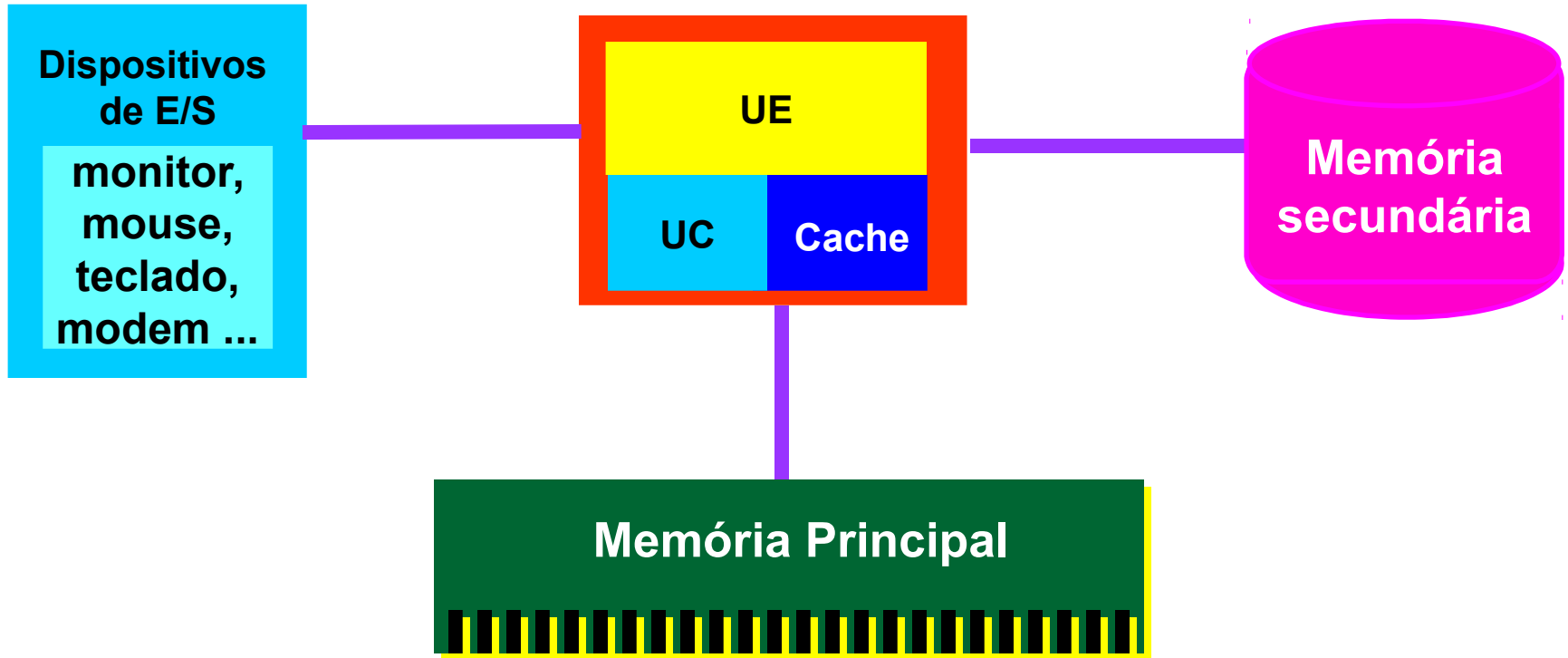
Realiza as operações lógicas e aritméticas

Armazena os programas em execução e seus dados

Lê dados da memória



Introdução



- **4 unidades funcionais**
- **aritmética binária**



Introdução

- **Entrada**

- Dispositivos através dos quais os dados são gravados na memória do sistema
- Ex.: Mouse, Teclado, Scanners, Câmeras

- **Saída**

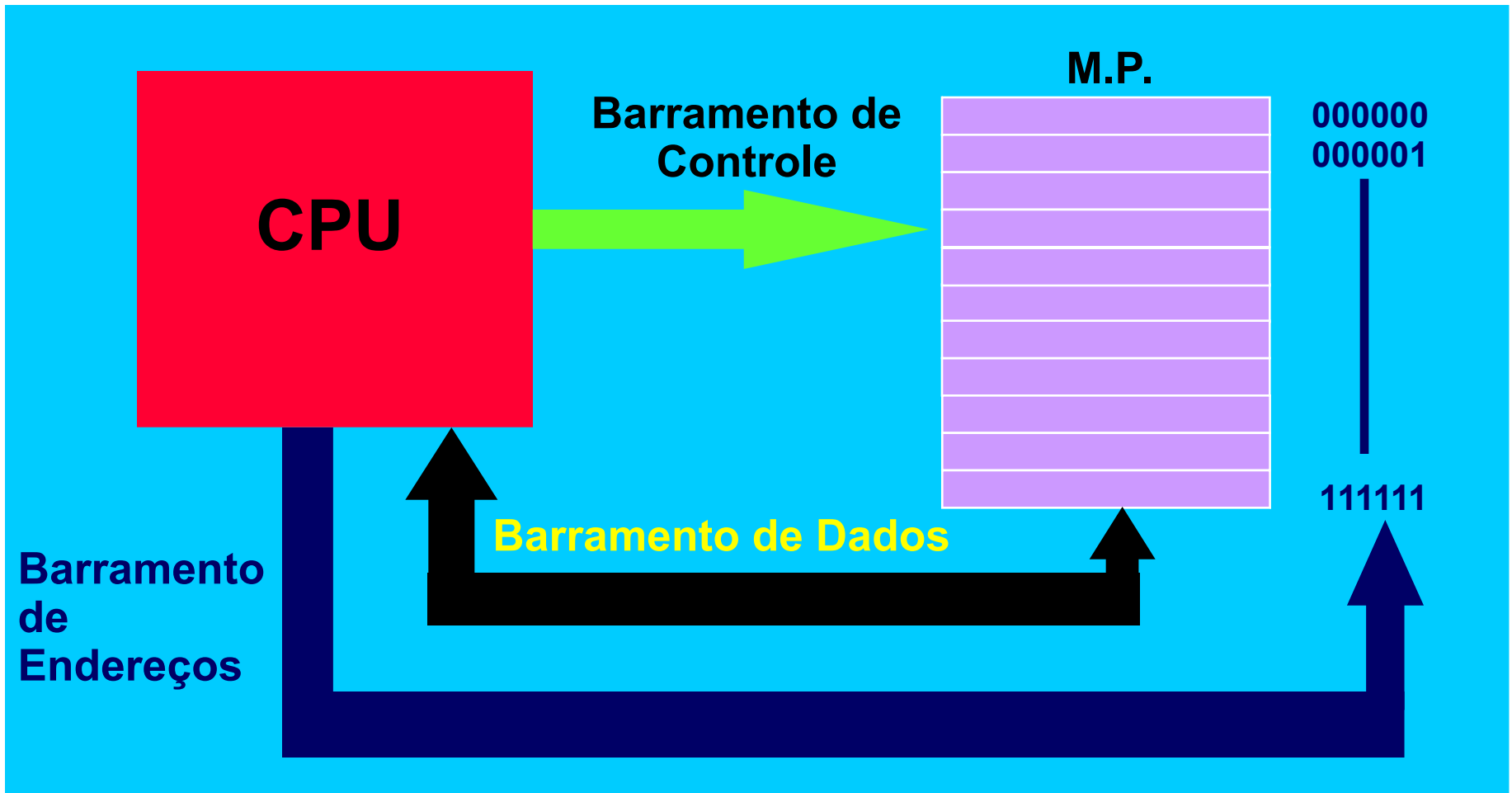
- Estes dispositivos lêem dados da memória apresentando ao usuário
- Ex.: Monitor, Impressora, Placa de Som

- **Armazenamento**

- Possibilitam a persistência dos dados e instruções (programas) utilizados pelo sistema de computação
- Ex.: HD, Discos flexíveis, CD, Blue-Ray, SSDs

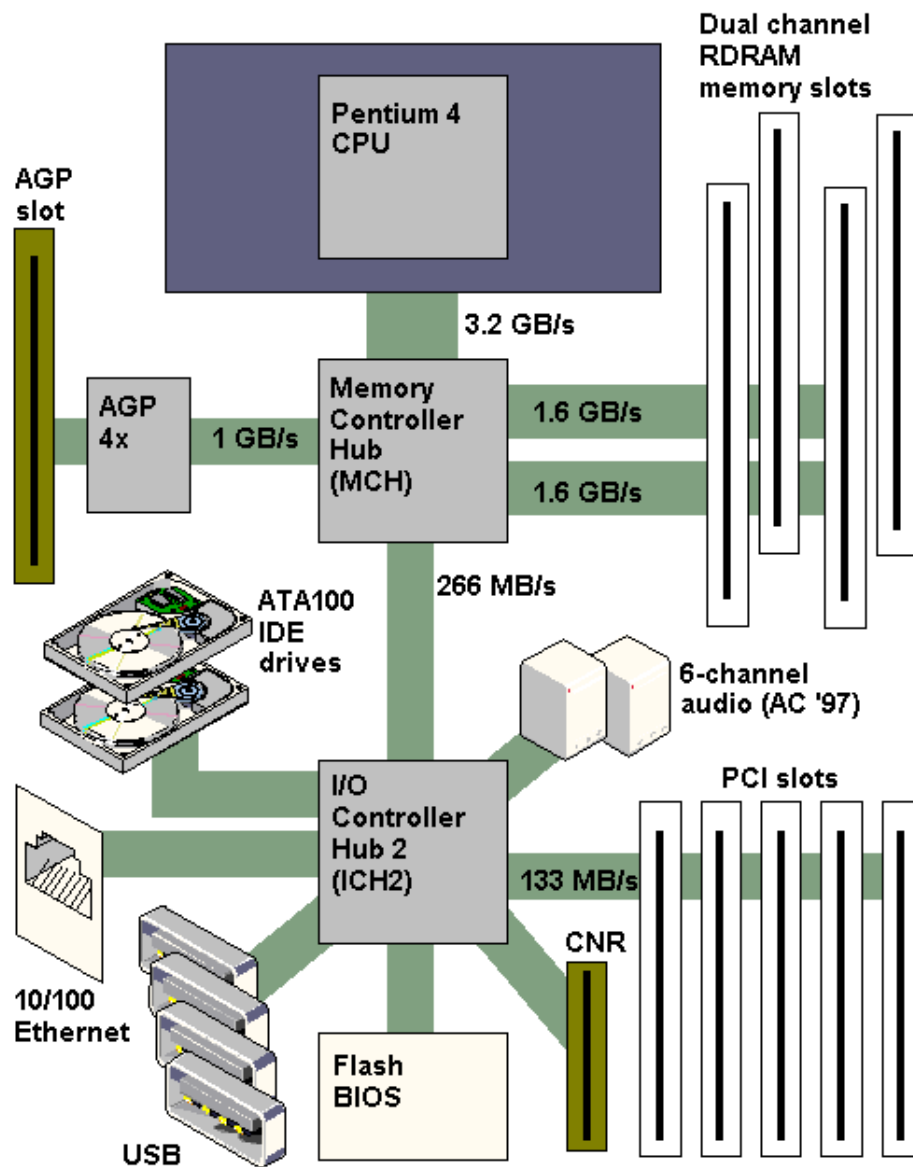


Introdução



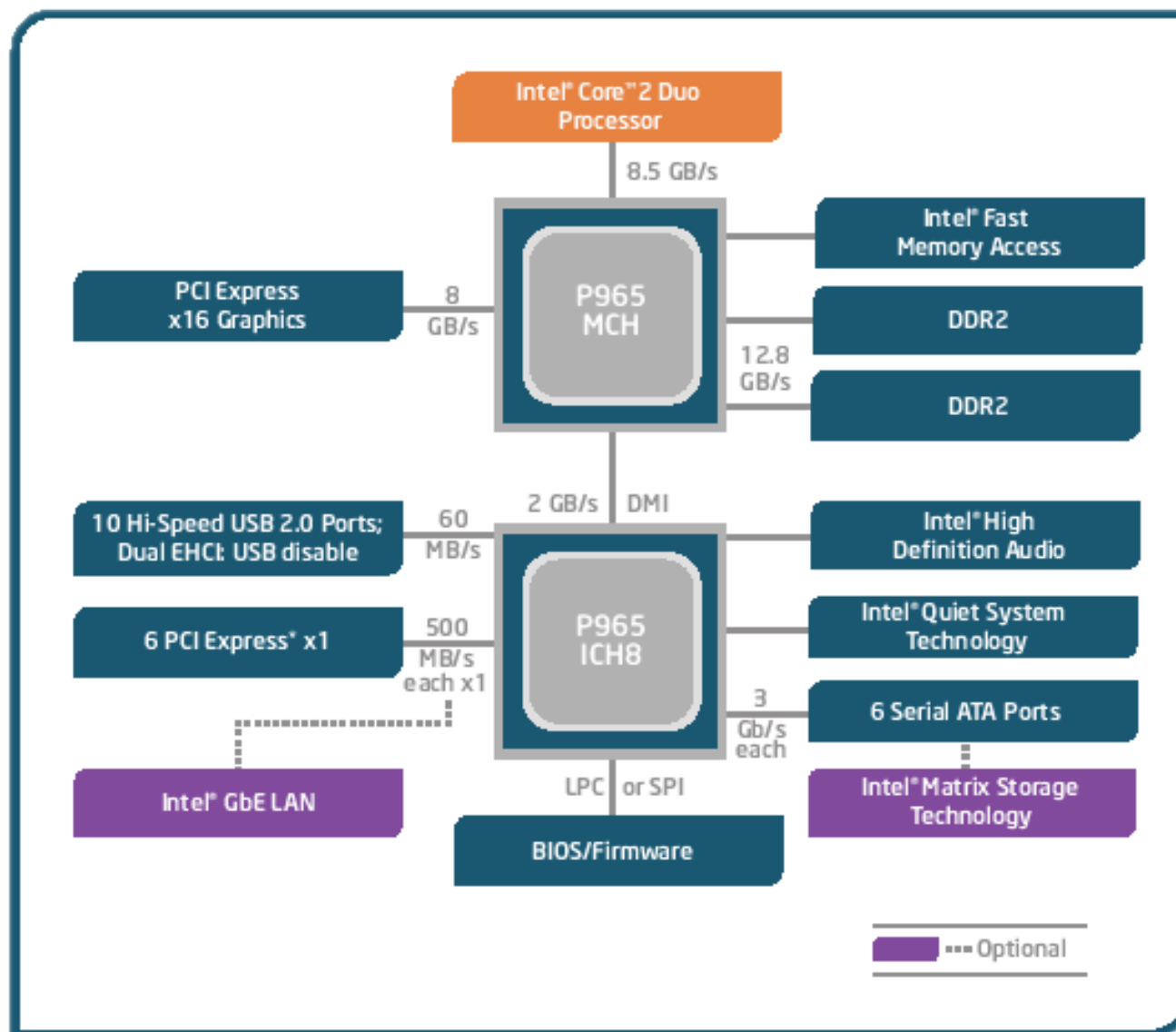


Introdução



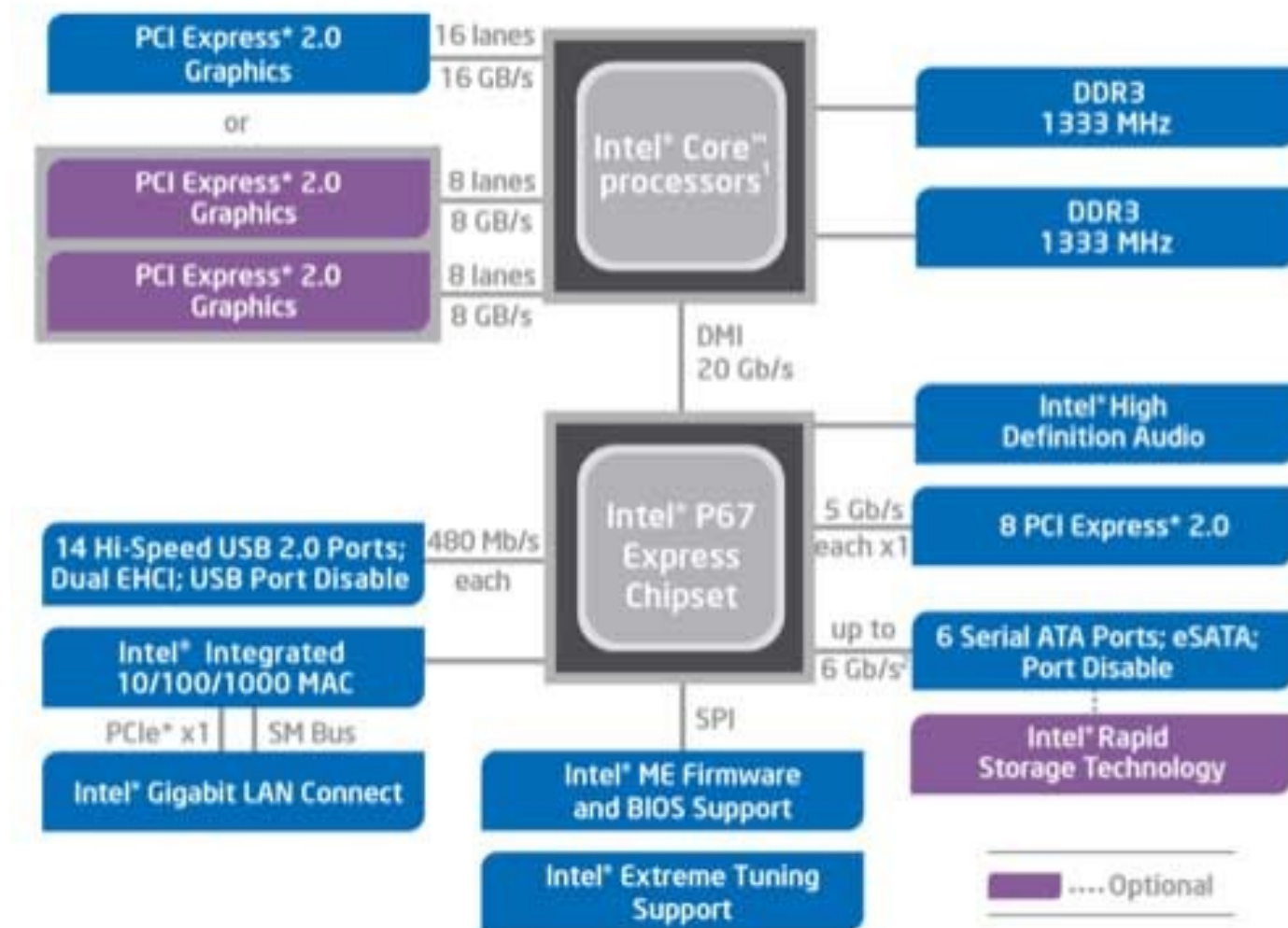


Introdução



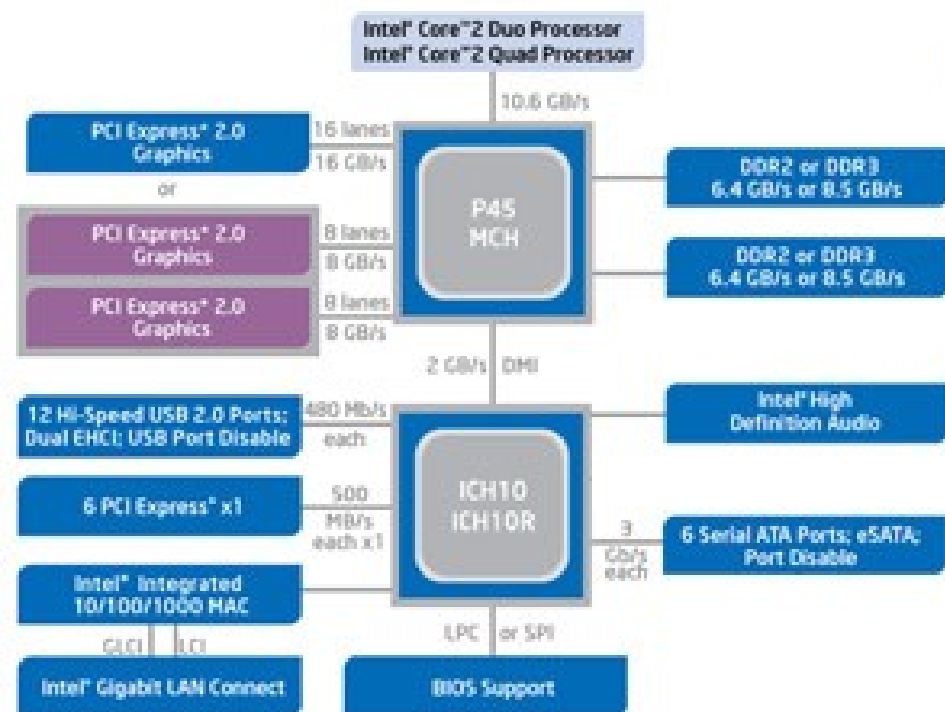


Introdução

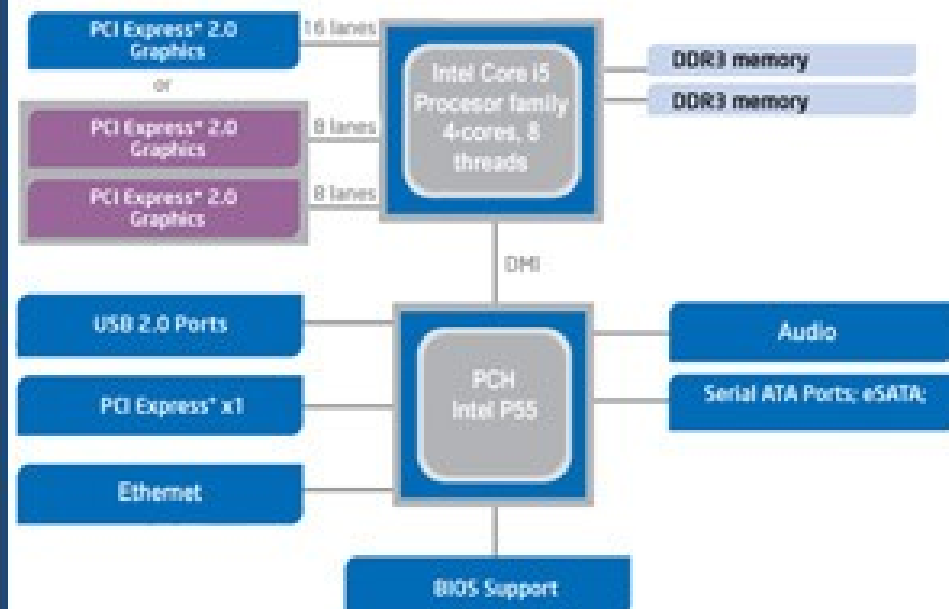




Introdução



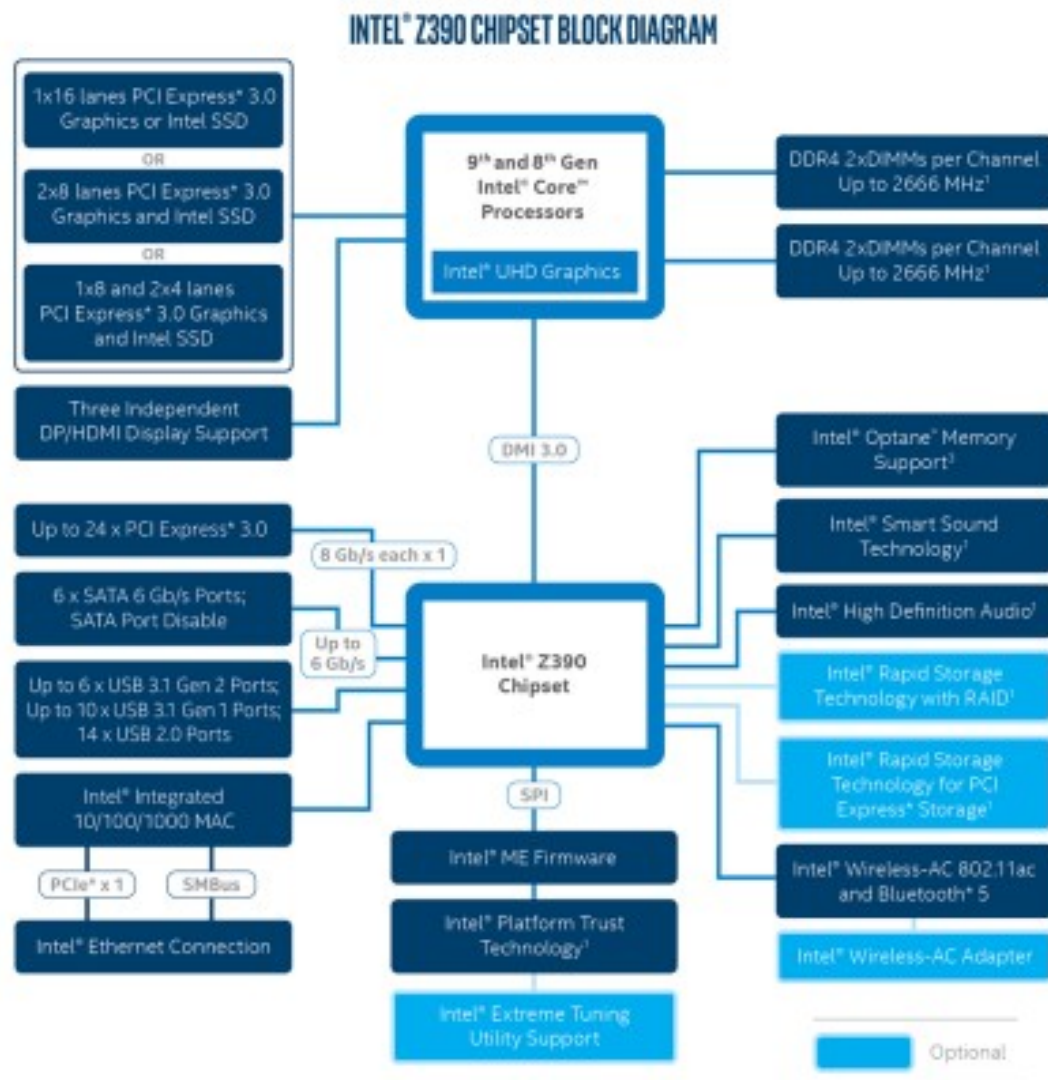
3-Chip Architecture
(old)



2-Chip Architecture
(new)



Introdução





Introdução

Executando instruções:

- Pedro **some** para mim **5** mais **17**



- **representação binária:**

2 = 10 **5** = 101 **17** = 10001

2 = 00000010 **5** = 00000101 **17** = 00010001

- **Codificação:**

Somar --- 100

Subtrair --- 101

Dividir --- 111

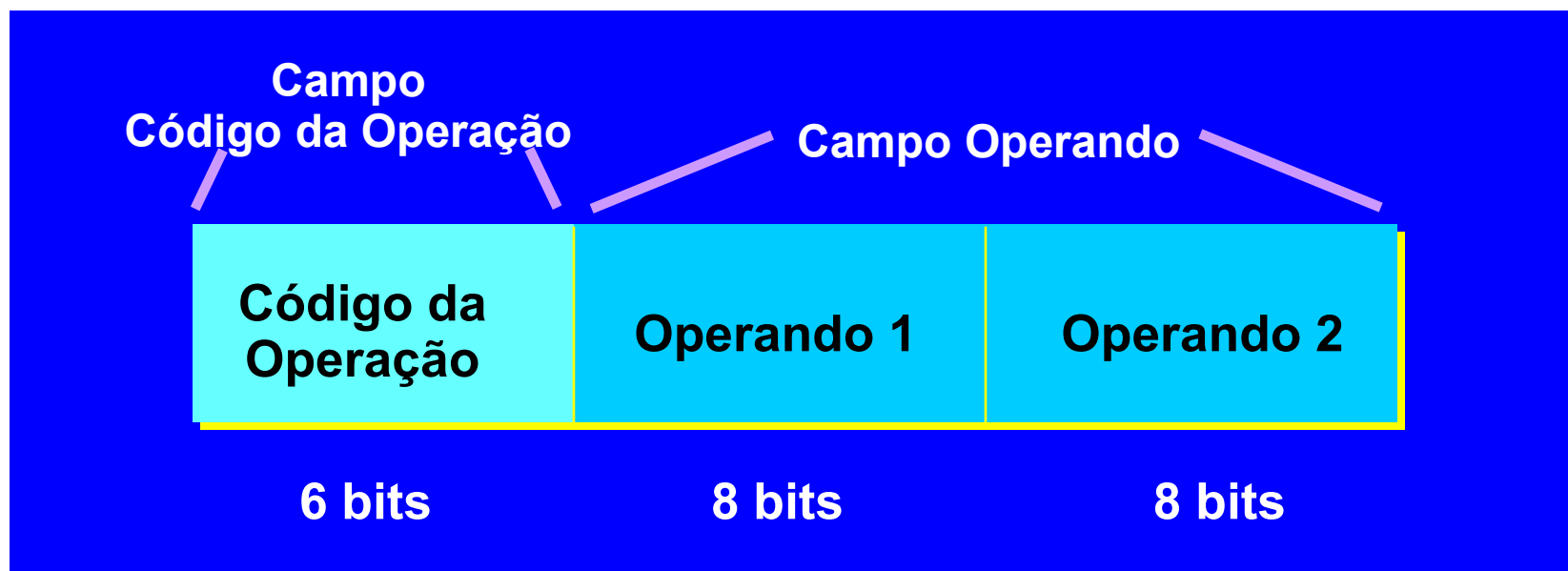
Desvie se zero --- 110



Introdução

- Instrução de máquina:

$$2^{N_Bits} = X$$



00010000000010100010001



Introdução

- 1. Processador busca instrução na memória
- 2. A instrução é decodificada (determina qual é a instrução e o que precisa para executar)
- 3. Os dados da instrução são carregados (podem vir da memória, de um registrador ou mesmo da instrução)
- 4. A instrução é executada
- 5. O resultado da instrução é armazenado (na memória ou num registrador)

