

Programação de software Básico

Introdução

- Software básico é a denominação usada para descrever software que atende à camada de base do funcionamento de um computador.
- O software básico situa-se entre o hardware e os softwares denominados aplicativos.
- Exemplos de software básico: sistemas operacionais e drivers de dispositivos.

Introdução

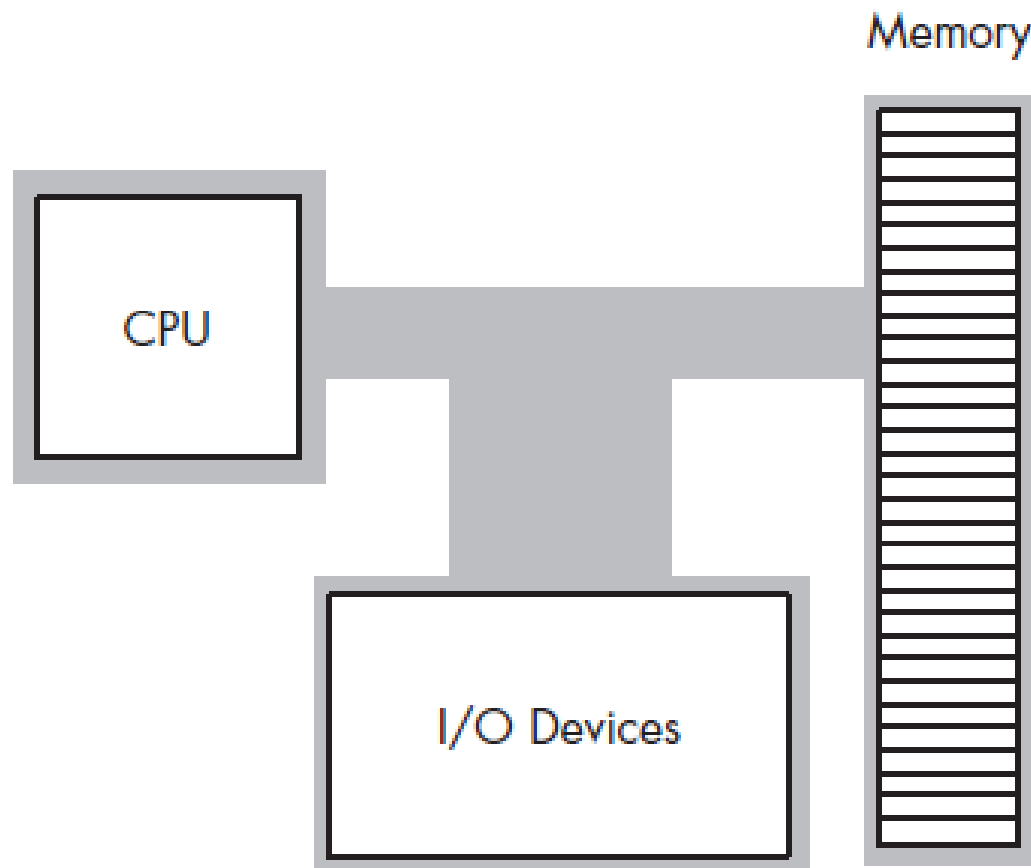
- Software básico controla o hardware para permitir que os softwares aplicativos executem sem se preocupar com a complexidade.
- O software básico viabiliza o funcionamento do sistema computacional.
- O software básico está mais próximo do hardware quando comparado ao software aplicativo.

Introdução

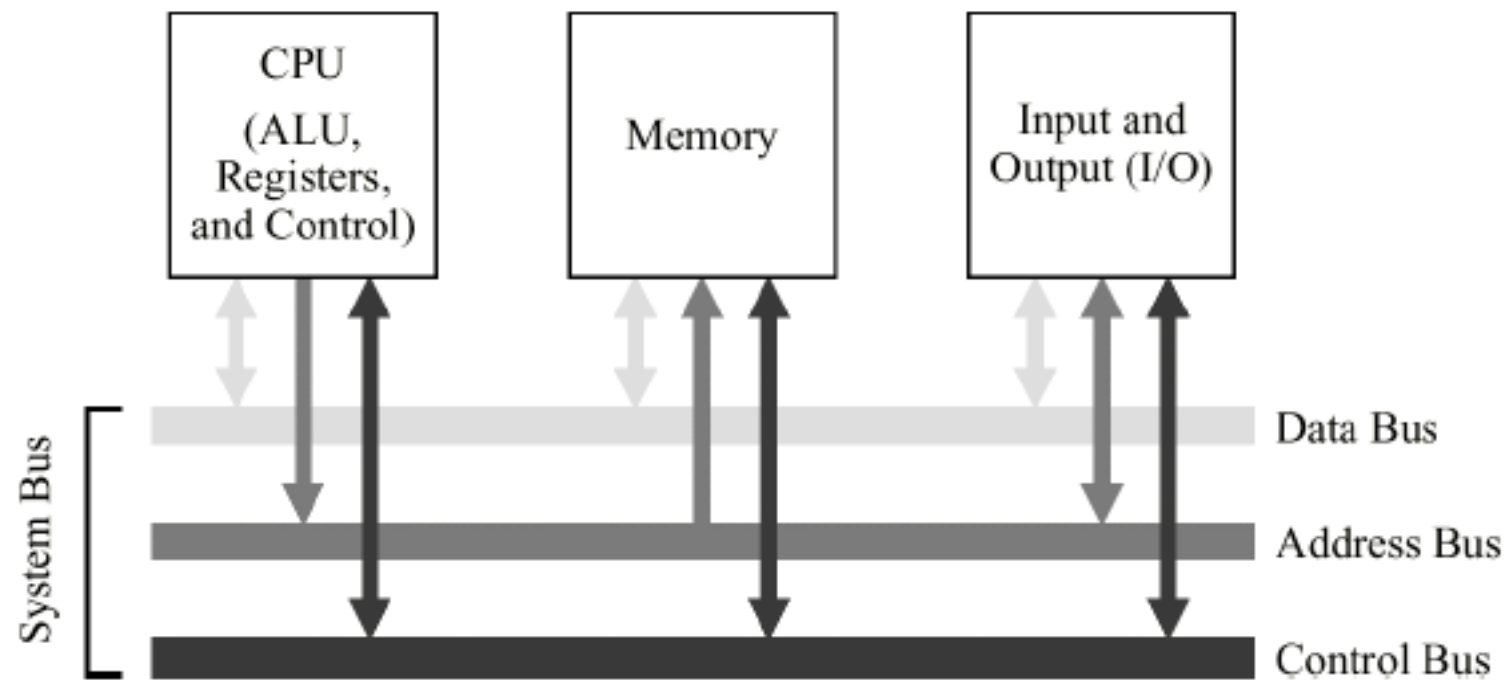
- Para controlar o hardware o software básico interage com a arquitetura do hardware.
- É preciso desvendar os aspectos de hardware.
- Para tanto, é necessário o estudo da arquitetura computacional através da análise de processadores, memória, barramentos e I/O.

A arquitetura de Von Neumann

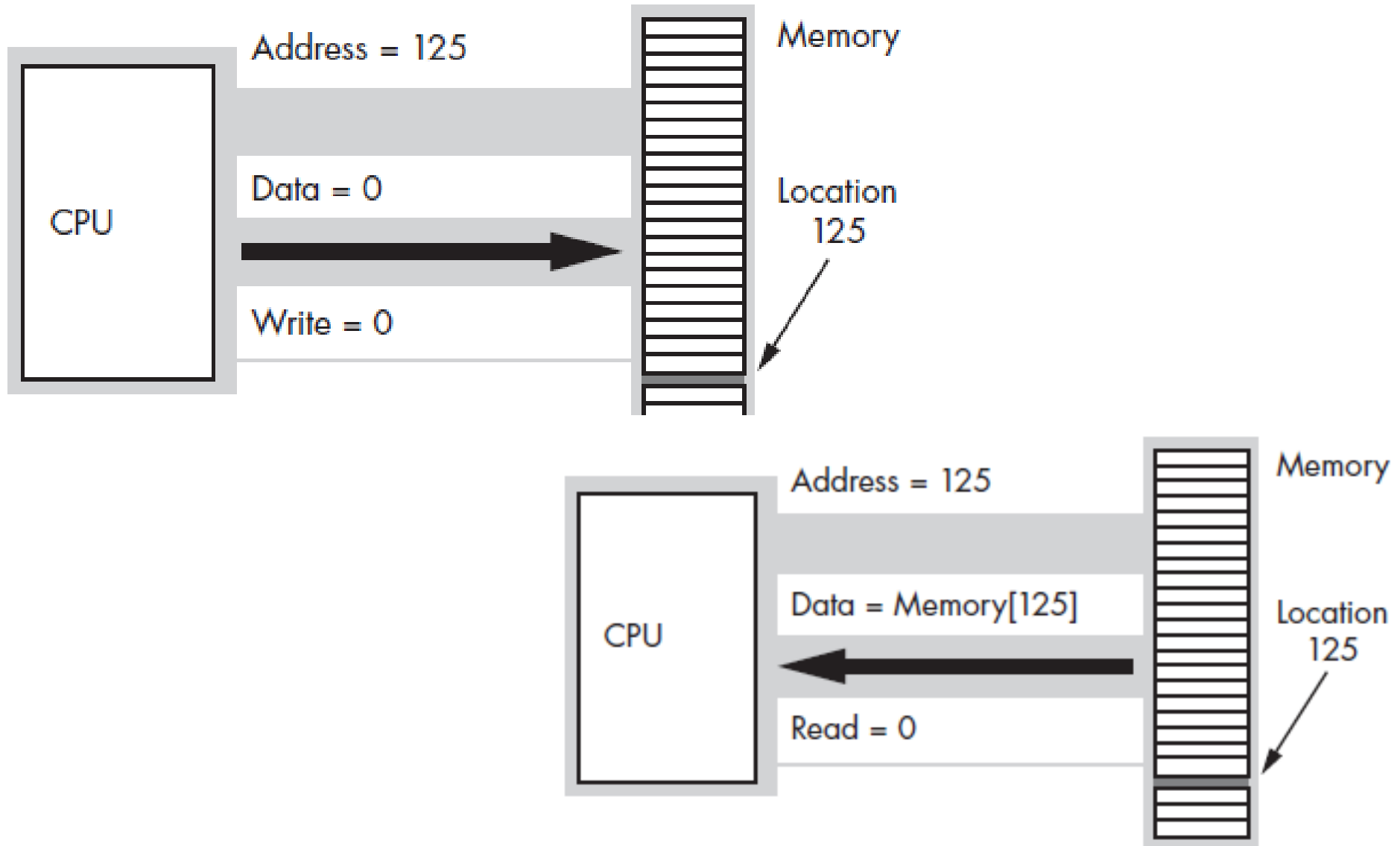
- Para programar próximo da máquina é preciso entender a sua arquitetura



Arquitetura detalhada

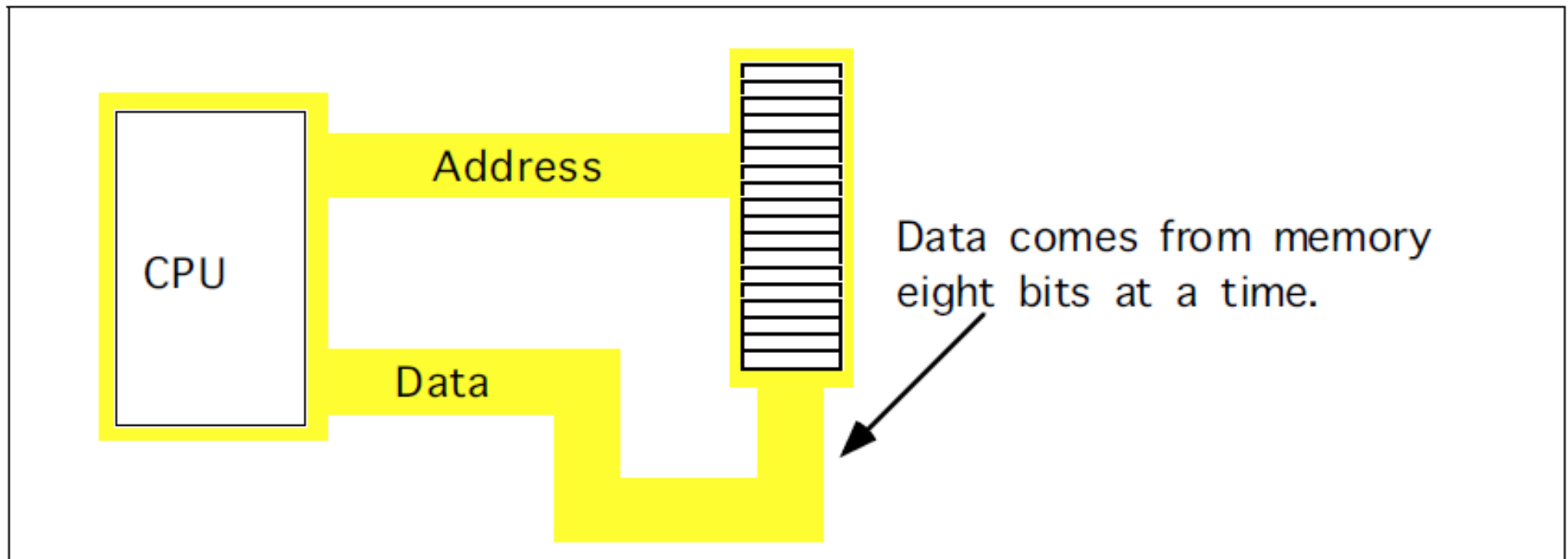


Interação Processador/memória



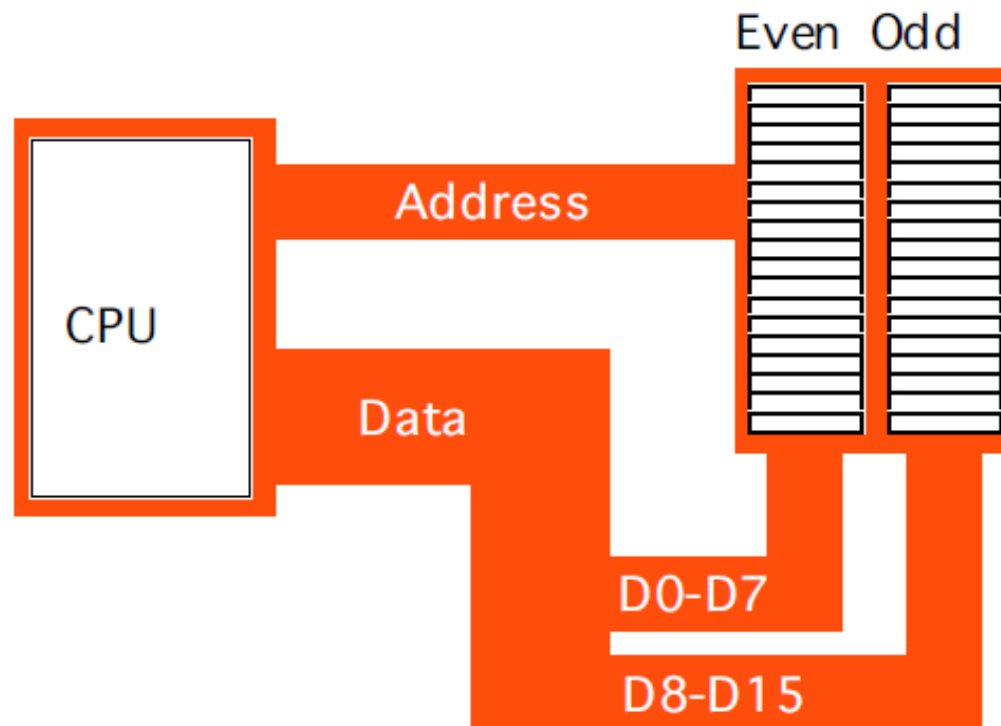
Como os dados são representados

- Organização da memória em Processadores de 8 bits:



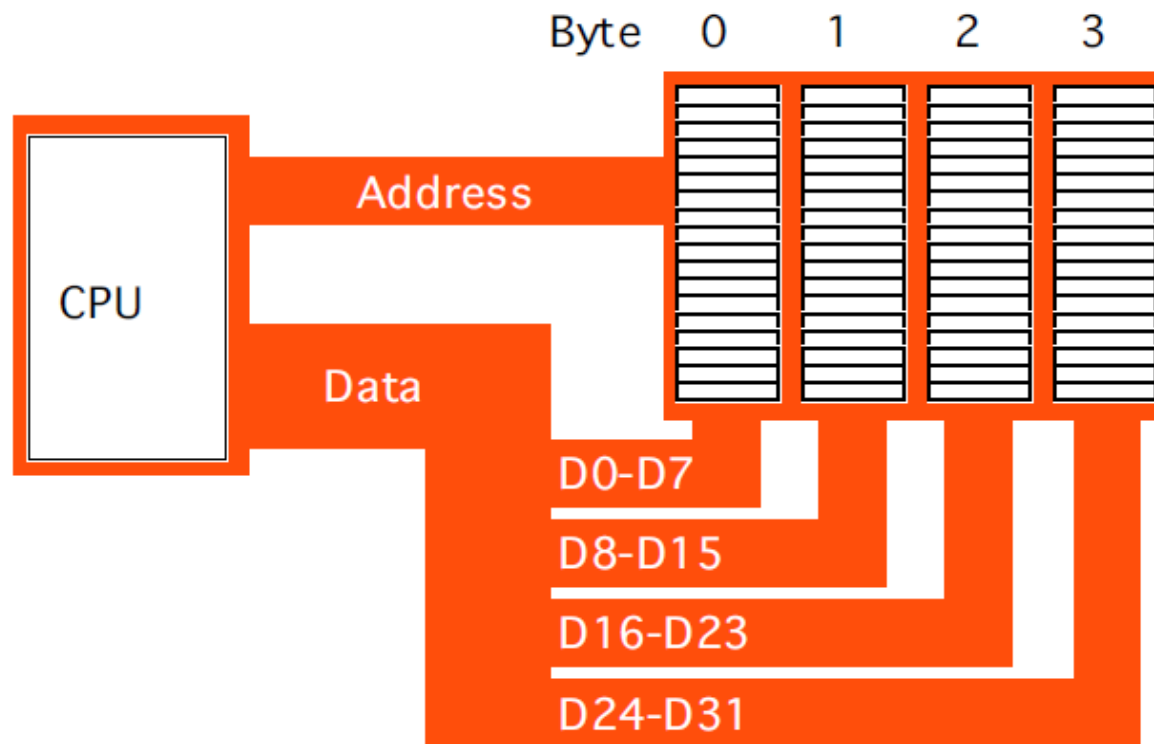
Como os dados são representados

- Organização da memória em Processadores de 16 bits:



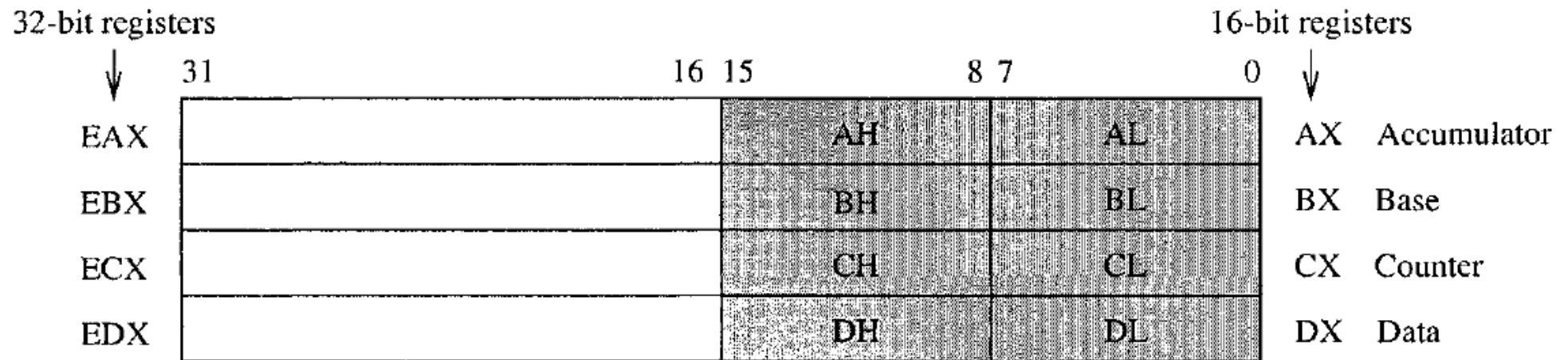
Como os dados são representados

- Organização da memória em Processadores de 32 bits:



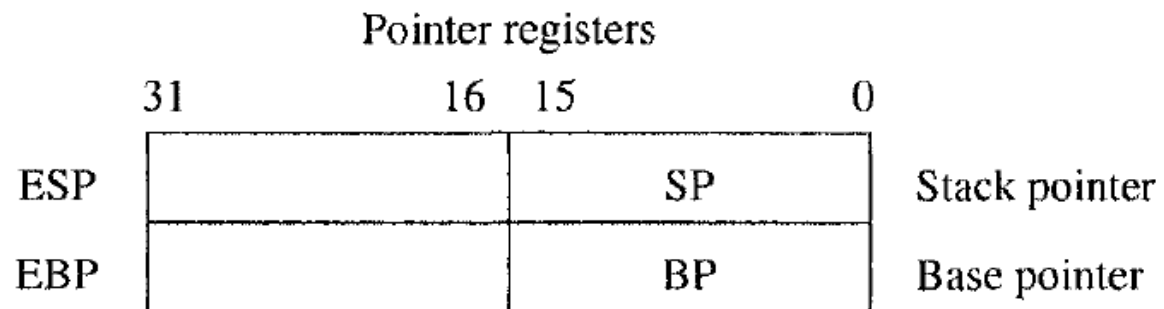
Registadores

- Registadores de dados



Registradores

- Registradores de ponteiro



Registadores

- Registradores de segmento

15		0
	CS	Code segment
	DS	Data segment
	SS	Stack segment
	ES	Extra segment
	FS	Extra segment
	GS	Extra segment

Exercício

- Qual é a base da interação entre a arquitetura computacional dos processadores ?
- Como a evolução do tipo de arquitetura de processadores influencia na sua organização ?
- Cite alguns grupos de registradores e comente a sua funcionalidade

O Desenvolvimento de Software básico

- O software básico utiliza de linguagens de programação próximas do hardware
- Essas linguagens são denominadas linguagens de montagem.
- As linguagens de montagens estão diretamente ligadas a arquitetura do processador.
- Determina os registradores e tipos de instruções

```
MOV AX,A  
MOV B,AX
```

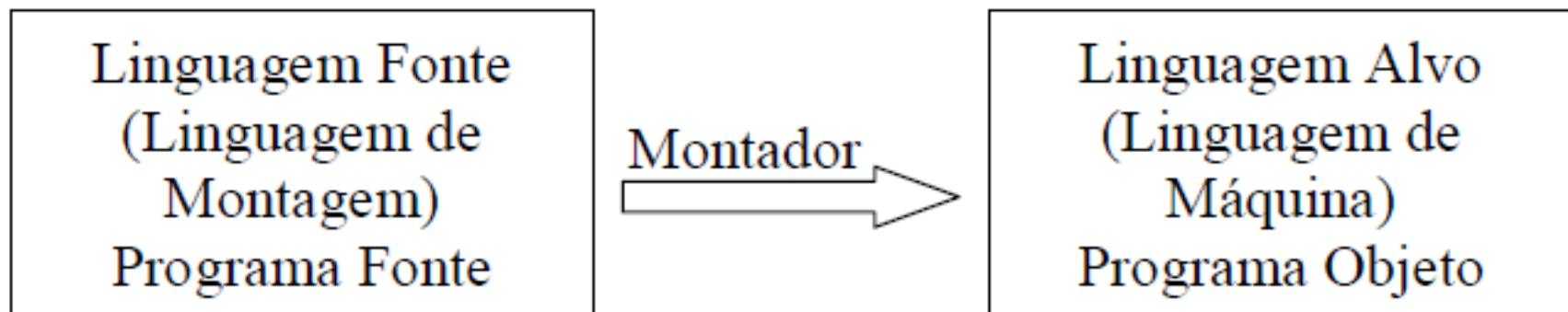
O nível da linguagem de
montagem

Linguagem de montagem

- Os programas que convertem um programa de usuário escrito em uma linguagem para outra linguagem são chamados de tradutores.
- Ocorre uma tradução entre a Linguagem fonte e linguagem alvo
- Conversão para um programa-objeto ou programa binário executável

Linguagem de montagem

- Existem 2 tipos de tradutores:
 - Compilador: linguagem fonte = linguagem de alto nível; linguagem objeto = linguagem de máquina
 - Montador: linguagem fonte = linguagem de montagem (*Assembly*) representação simbólica da linguagem de máquina; linguagem objeto = linguagem de máquina.



O que é uma linguagem de montagem

<u>Label</u>	<u>Operação</u>	<u>Operando(s)</u>	<u>Comentários</u>
K:	DD	2	; Define Dado K reservando 4 bytes iniciando com 2.
DESTINO	ADD	EAX, K	; Adicione K a EAX
	JUMP	DESTINO	; Desvie para DESTINO

Campo de Rótulo (Label): nome simbólico atribuído a um endereço. Permite que destinos de desvios sejam identificados facilmente pelo programador.

Campo de Operação: abreviatura simbólica do OP-CODE. Deve lembrar o tipo de operação realizada pela instrução. Exemplo: ADD = soma.

Campo de operando(s): especifica simbolicamente o(s) endereço(s), registrador(es) ou constante(s) utilizados como operandos.

Campo de comentários: documento o programa (o qual seria ilegível sem este campo). Ignorado pelo montador.

Por que usar uma linguagem de montagem

- Melhora no desempenho.
- Uso de assembly em pontos criticos de desempenho
 - É comum 10% do codigo de um programa ser responsavel por 90% do tempo de execucao
- Ao contrário da programação em alto nível, o programador de linguagem de montagem dispõe de todo o hardware.
Acesso direto ao hardware
 - Certos componentes so podem ser acessados por linguagem de montagem:
 - Desenvolvimento de drivers
 - Ex: Controladores de dispositivos embutidos de tempo real

Exercícios

- Descreva o formato de uma instrução em linguagem de montagem
- Cite os motivos para a utilização de uma linguagem de montagem
- Diferencie montador e compilador

O que é uma linguagem de montagem

- É aquela na qual cada comando produz exatamente uma instrução de máquina.
- Relação 1 para 1
- Tipos de instruções:
 - MOV - realizam cópia de valores entre registros
 - ADD, SUB, MULT,... - realizam operações aritméticas
 - EQ, NEQ, LEQ,... - realizam operações de comparação

O que é uma linguagem de montagem

- A linguagem de montagem e a linguagem de máquina possuem uma relação de uma para um, ou seja, cada instrução de montagem possui uma instrução de máquina equivalente;
- A diferença entre as linguagens de montagem e de máquina é que a primeira é uma representação simbólica da segunda, que é puramente numérica;

Linguagem de máquina	Linguagem de montagem
BA0B01	mov dx,msg
B409	mov ah,9
B44C	mov ah,4Ch
CD21	int 21h
48656C6C6F2C20576F	msg db 'Hello, World!',0Dh,0Ah,'\$'

O que é uma linguagem de montagem

- Quando uma linguagem de programação é uma representação simbólica de uma linguagem de máquina ela é chamada de **linguagem de montagem** ou **assembly**
 - A instrução de máquina:
 - 10110000 01100001
 - Pode ser representada por:
 - MOV AL, 61h
 - Que significa mover o número 97 para o registrador AL

Linguagem de montagem X linguagem alto nível

- Por exemplo:
 - O comando na linguagem C abaixo não representa uma instrução na linguagem de Maquina `int x = y + 5`
 - Já a instrução **ADD AX, 5** representa um único comando de maquina para o processador

Linguagem de montagem X linguagem alto nível

As linguagens de alto nível possuem uma relação 1-para-muitos com as linguagem de montagem.

– Por exemplo:

Programa em C:

```
int    Y;  
int    X = (Y + 4) * 3;
```

O mesmo programa em assembly:

```
mov    eax, Y  
add    eax, 4  
mov    ebx, 3  
imul   ebx  
mov    X, eax
```

Linguagem de montagem X linguagem alto nível

- A linguagem de alto nível utiliza instruções que operam sobre estruturas de dados complexas.

A compilação de uma sequência de instruções em linguagem de alto nível geralmente gera uma sequência maior de instruções que operam sobre estruturas de dados mais simples e diversas daquelas presentes no programa fonte.

Assembly

- Tipos de linguagens

Linguagem de alto nível:

A=5;

Linguagem de montagem:

MOVE A, #5;

Linguagem de máquina:

0011001100000101

- Comparando com a linguagem de alto nível:

B = A

```
MOV AX, A
MOV B, AX
```

A Linguagem Assembly

- Comparando com a linguagem de alto nível:

```
A = 5 - A
```

```
MOV AX, 5  
SUB AX, A  
MOV A, AX
```

```
A = B - 2 x A
```

```
MOV AX, B  
SUB AX, A  
SUB AX, A  
MOV A, AX
```

Exercício

Usando apenas MOV, ADD, SUB, INC e DEC, traduza as expressões abaixo em linguagem de alto-nível para linguagem Assembly.

a) $A = B - A$

b) $C = A + B$

c) $B = 3 \times B + 7$

Utilizando MOV, MUL, DIV, traduza as expressões abaixo em linguagem de alto-nível para linguagem assembly.

a) $C = 3 \times 4 \times A - 5$

b) $D = 5 / 5 - 1$

c) $E = 6 \times 4 - 3 + 1 + C$

A Linguagem Assembly

- Podemos utilizar o programa debug no prompt do DOS.
- Visualizando os registradores:

```
-R
AX=0000  BX=0000  CX=0011  DX=0000  SP=0100  BP=0000  SI=0000  DI=0000
DS=152B  ES=152B  SS=153D  CS=153B  IP=0000  NV UP EI PL NZ NA PO NC
153B:0000 B80B00          MOV     AX,000B
-
```

A Linguagem Assembly

Podemos utilizar montadores (ASSEMBLER) para o desenvolvimento de código assembly:

- ✓ - MASM (Microsoft Macro Assembler)
- ✓ - TASM (Turbo Assembler)
- ✓ - NASM (Netwide Assembler)
- ✓ - GAS (GNUassembler)