

## Estrutura do tema ISC

1. Representação de informação num computador
2. Organização e estrutura interna dum computador
3. Execução de programas num computador
4. O processador e a memória num computador
5. Da comunicação de dados às redes

## Níveis de abstracção:

- nível das linguagens HLL (*High Level Languages*): as linguagens convencionais de programação (puro texto)
  - » imperativas e OO (Basic, Fortran, C/C++, Java, ...)
  - » funcionais (Lisp, Haskell, ...)
  - » lógicas (Prolog, ...)
- nível da linguagem *assembly* (de “montagem”): linguagem intermédia (comandos do CPU em formato texto)
- nível da linguagem máquina: a linguagem de comandos, específica p/ cada CPU ou família de CPU's (em binário puro)
  - » arquitecturas CISC (*Complex Instruction Set Computers*)
  - » arquitecturas RISC (*Reduced Instruction Set Computers*)

## Execução de programas num computador (2)

## Execução de programas num computador (3)

```
int x = x+y;
```

- Código C
  - somar 2 inteiros (c/ sinal)

```
addl 8(%ebp), %eax
```

Idêntico à expressão

```
x = x + y
```

- *Assembly* (da GNU p/ IA32)
  - somar 2 inteiros de 4-bytes
    - operandos “long” em GCC
    - a mesma instrução, c/ ou s/ sinal
  - operandos:
    - x: em registo %eax
    - y: na memória M[%ebp+8]

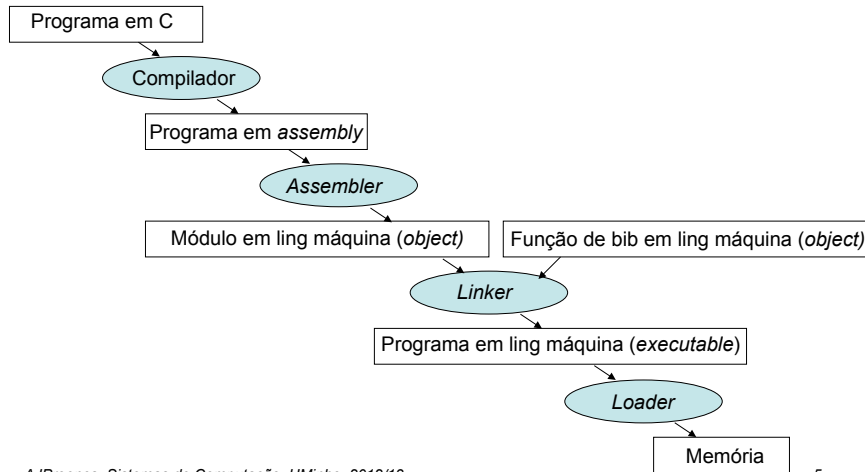
```
0x401046: 03 45 08
```

- Código *object* em IA32
  - instrução com 3-bytes
  - na memória em 0x401046

## Mecanismos de conversão (para comandos do CPU):

- compilador
  - traduz um programa de um nível de abstracção para outro inferior (converte um ficheiro de texto noutro de texto); por ex., de C para *assembly*
  - normalmente inclui mais que um passo de conversão, até chegar à linguagem máquina
- *assembler* (“montador”)
  - “monta” os comandos / instruções em binário (*object*), de acordo com as regras do fabricante do CPU
- interpretador
  - analisa, uma a uma, as instruções de um programa em HLL, e:
    - » gera código em linguagem máquina para essa instrução, e
    - » executa esse código

De um programa em HLL até à sua execução:



Ciclo de execução de instruções:

- Leitura de uma instrução da memória  
... e incremento do IP
- Descodificação da instrução
- Execução da operação
  - cálculo da localização do(s) operando(s), e ir buscá-lo(s), se necessário
  - execução da operação especificada
  - guardar resultado, se necessário

Modelo de computação de von Neumann (1945)

Análise de um exemplo: `movl Loc, %eax`

Modelo de computação de von Neumann, 1945/46 (1)

Modelo de computação de von Neumann, 1945/46 (2)

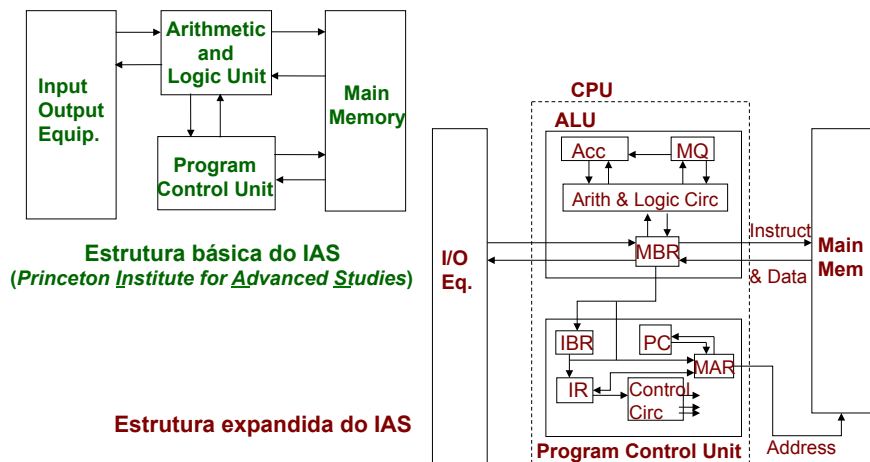
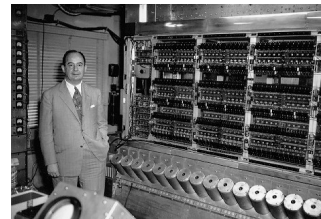
ENIAC (1ª geração, 1945)

- objectivo: cálculo tabelas de artilharia (mas 1º teste foi p/ bomba H)
- máquina **decimal**
- 17.468 válvulas, 27 toneladas
- programação: manual, alterando as conexões (cablagem)



Von Neumann introduz conceito de **stored-program** :

- dados e instruções em **binário**, e armazenados numa memória
- memória acessada pelo endereço da informação
- execução de instruções de modo sequencial (dai o *Program Counter*, PC), interpretadas pela *unid. controlo*
- constrói novo computador, IAS



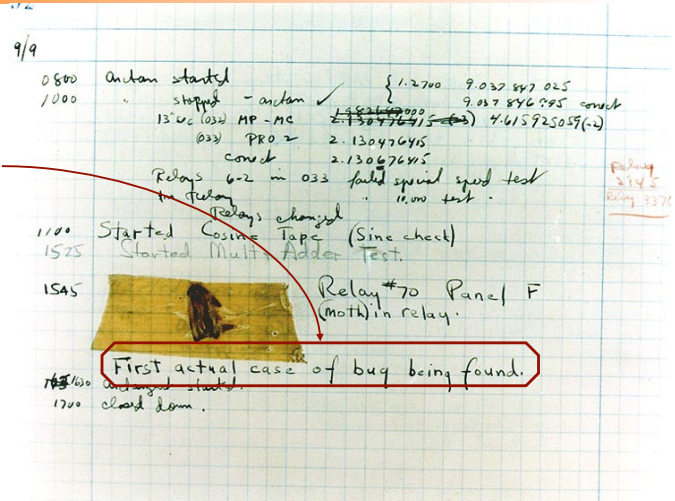
Estrutura básica do IAS (Princeton Institute for Advanced Studies)

Estrutura expandida do IAS

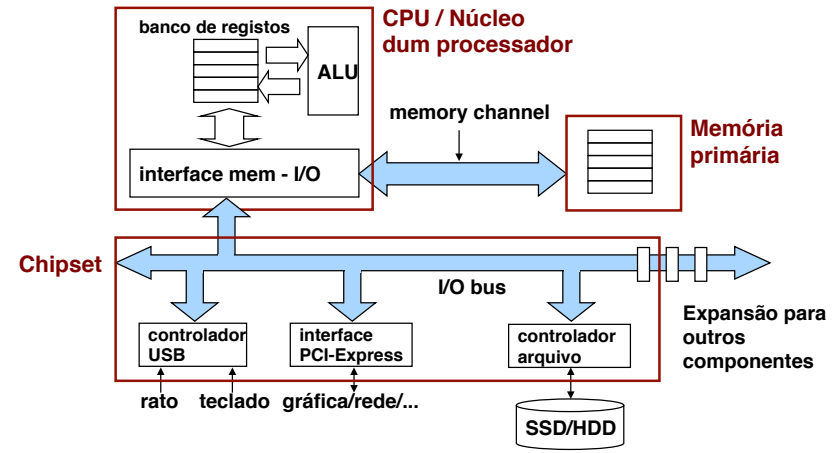
## Relato do primeiro bug num computador

## Modelo de arquitetura de um computador elementar

**IBM, Mark II**  
(09-Set-1947):  
- entrada que os operadores de serviço fizeram no logbook...



### Exemplo de execução de uma instrução em linguagem máquina (1)



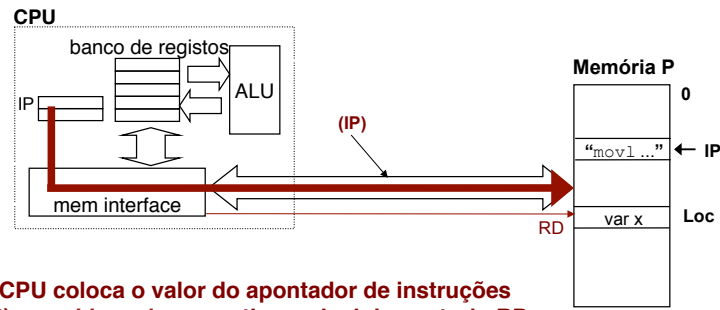
### Exemplo de execução de uma instrução em linguagem máquina (2)

Ex.: `movl Loc, %eax`

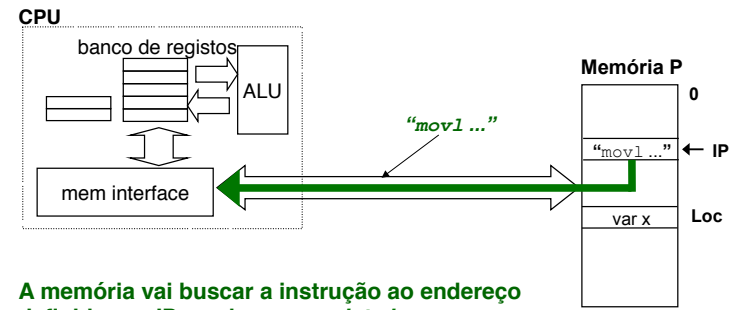
Ex.: `movl Loc, %eax`

#### 1. Leitura da instrução na memória (1)

#### 1. Leitura da instrução na memória (2)



O CPU coloca o valor do apontador de instruções (IP) no *address bus*, e activa o sinal de controlo RD



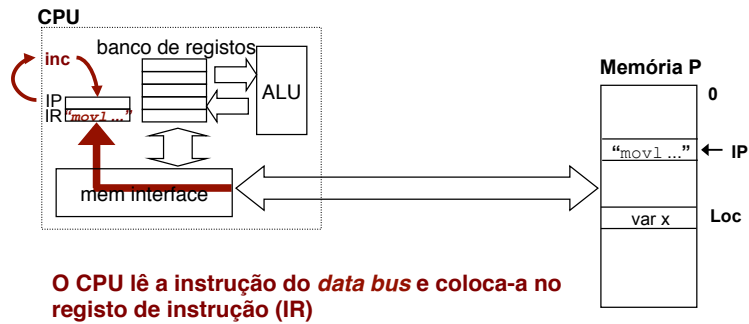
A memória vai buscar a instrução ao endereço definido por IP e coloca-a no *data bus*

Exemplo de execução de uma instrução em linguagem máquina (3)

Ex.: `movl Loc, %eax`

1. Leitura da instrução na memória (3)

... e incremento do IP



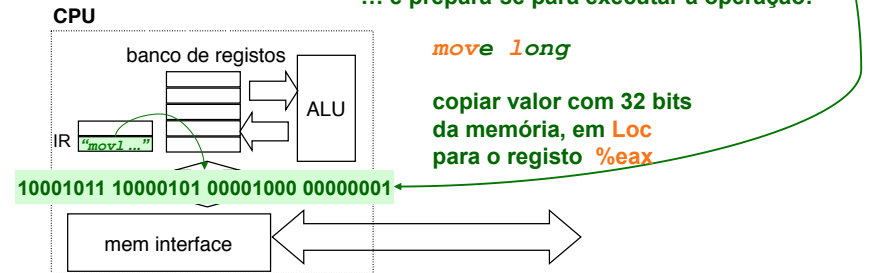
Exemplo de execução de uma instrução em linguagem máquina (4)

Ex.: `movl Loc, %eax`

2. Descodificação da instrução

A unidade de controlo do CPU descodifica a instrução...

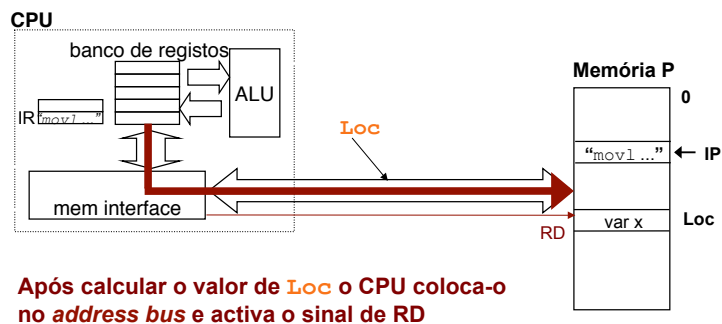
... e prepara-se para executar a operação:



Exemplo de execução de uma instrução em linguagem máquina (5)

Ex.: `movl Loc, %eax`

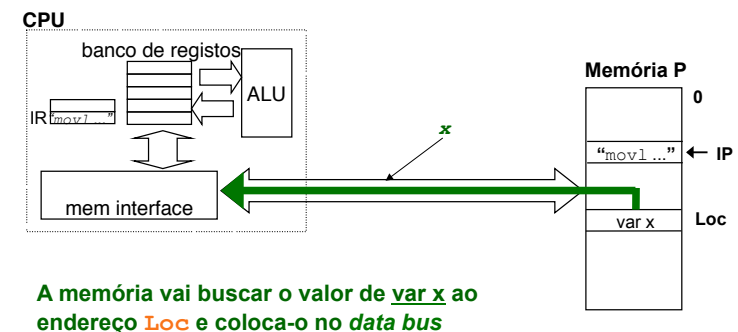
3. Execução da operação (1)



Exemplo de execução de uma instrução em linguagem máquina (6)

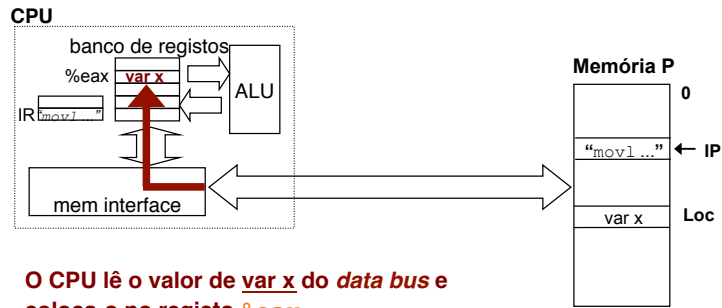
Ex.: `movl Loc, %eax`

3. Execução da operação (2)



Ex.: `movl Loc, %eax`

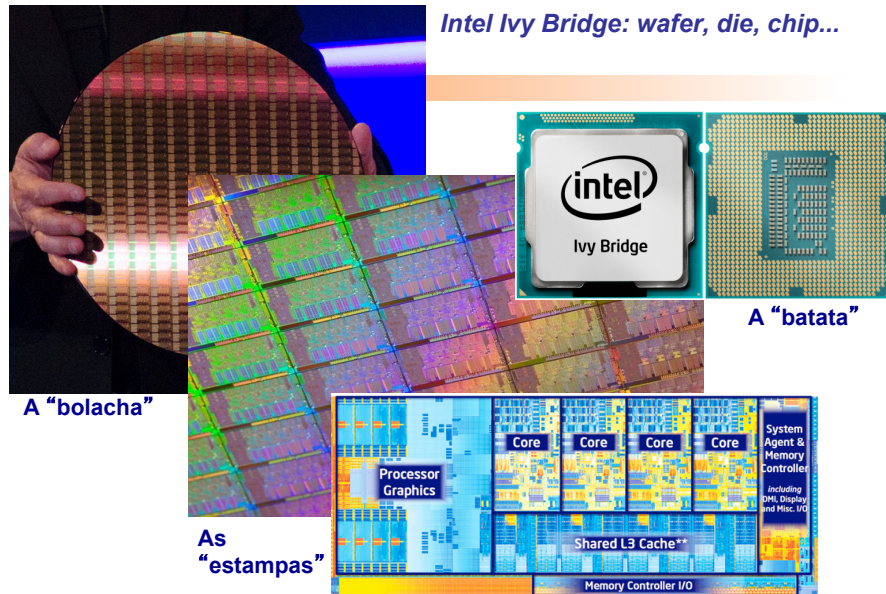
3. Execução da operação (3)



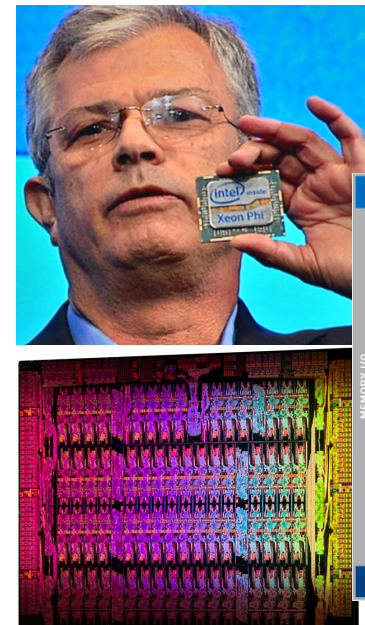
O CPU lê o valor de `var x` do data bus e coloca-o no registro `%eax`

Algumas palavras chave na terminologia tecnológica:

- instruction set
- CPU data width
- 32-bit processor
- clock cycle/ frequency
- overclocking
- throughput
- pipeline, superscalar
- dual/ quad/ ... core
- wafer, die, chip, ...
- RAM / ROM
- SRAM / DRAM
- cache memory
- flash memory
- access time
- magnetic drive, HDD
- SSD (Solid State Drive)
- NUMA structure
- USB/ SATA/ PCIe ...



Intel Ivy Bridge: wafer, die, chip...



Intel Many Integrated Core (MIC): the Xeon Phi with 60 cores

