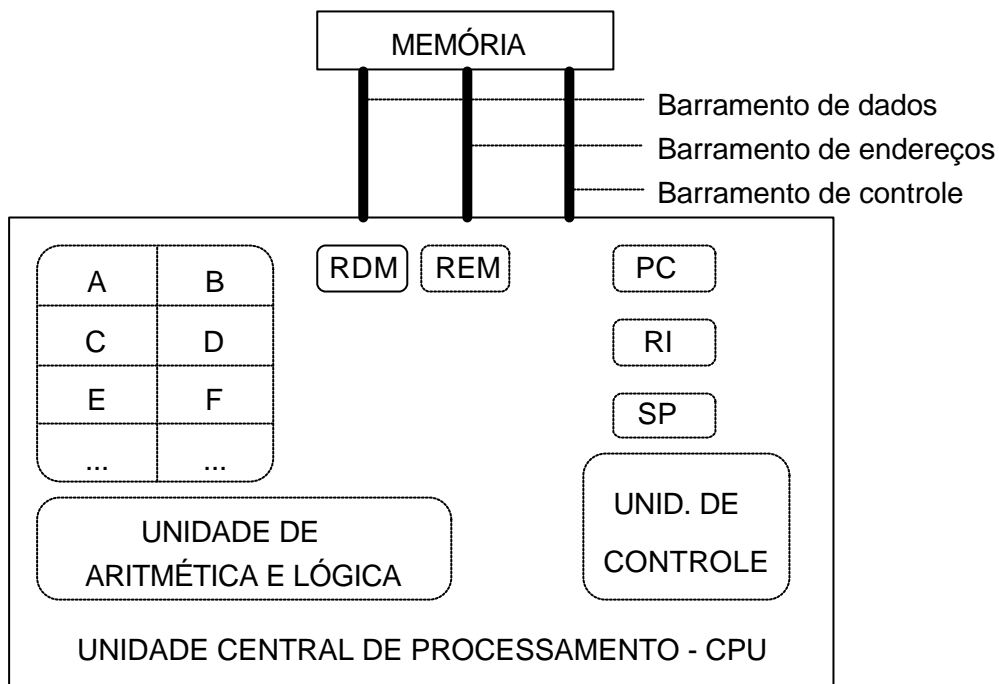


## A CPU e a execução de um programa



### Ciclo de busca e execução de uma instrução

1a. etapa: Busca da instrução na memória (“*FETCH*”)

REM  $\leftarrow$  PC

read

RDM  $\leftarrow$  (REM)

PC  $\leftarrow$  PC + n

2a. etapa: Decodificação da instrução

RI  $\leftarrow$  RDM

ocorre a decodificação do conteúdo de RI na UC

3a. etapa: Busca dos operandos da instrução (se houver)

REM  $\leftarrow$  endereço do operando

read

RDM  $\leftarrow$  (REM)

PC  $\leftarrow$  PC + n

4a. etapa: Execução da instrução – depende da instrução

### Exemplo:

O que acontece com os registradores da CPU na execução da seguinte instrução: **MOV AL,5h** ?

Seqüência de acontecimentos:

...	

PC  $\leftarrow$  01h

REM  $\leftarrow$  PC                      fetch  
RDM  $\leftarrow$  (REM)  
PC  $\leftarrow$  PC + 2

RI  $\leftarrow$  RDM                      decod.  
Decodificação de MOV

REM  $\leftarrow$  PC                      busca  
RDM  $\leftarrow$  (REM)                  operando  
PC  $\leftarrow$  PC+2

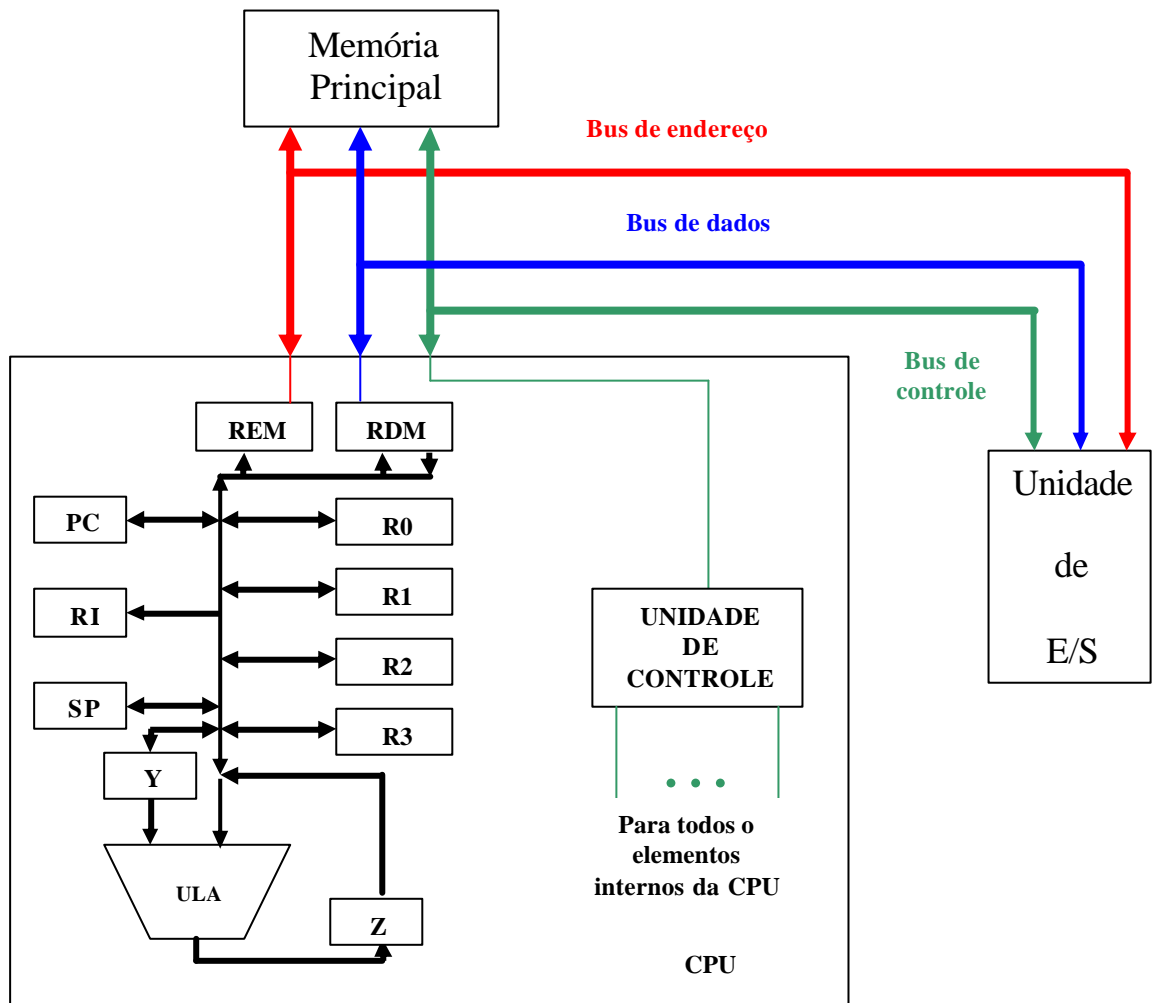
AL  $\leftarrow$  RDM                      execução

Memória  
Endereço   Conteúdo

01h	MOV AL
02h	5h
03h	....
04h	....

## Uma CPU Hipotética

Seja um sistema (visto na figura abaixo) que trabalhe com um palavra de dados e de endereços de 8 bits, Memória Principal organizada em palavras de 8 bits, os barramentos de dados e endereços de largura de 8 bits e a CPU com registradores de 8 bits.



CPU hipotética e sua ligação com a Memória Principal e a Unidade de E/S

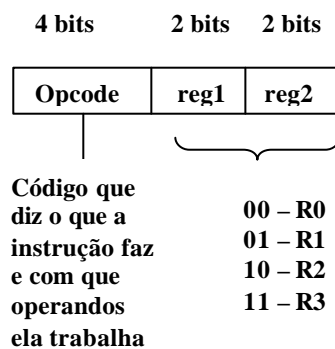
## Conjunto de Instruções

Cada processador tem o seu conjunto de instruções (ISA – Instruction Set Architecture). Este conjunto contém todas as instruções, em linguagem de máquina, que o processador pode executar. Segue abaixo a especificação do conjunto de instruções da CPU hipotética.

### ? Formato das Instruções

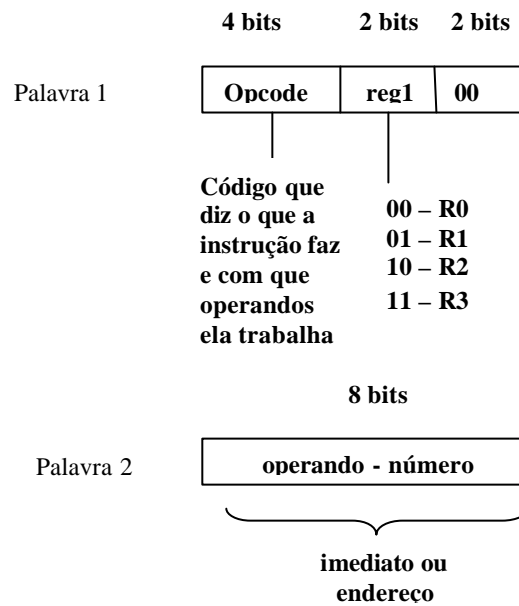
Para a CPU hipotética, iremos trabalhar com dois formatos de instruções diferentes:

? Formato tipo I – Uma palavra de 8 bits, com os seguintes campos:



Exemplo: MOV R0,R1 ; R0  $\leftarrow$  R1

? Formato tipo II – Duas palavras de 8 bits, com os seguintes campos:



Exemplos:

MOV R0, 5 ; R0 ~~←~~ 5

MOV R0, [5] ; R0 ~~←~~ [5]

Obs – Quando usamos [ ], significa que estamos acessando um conteúdo de memória, cujo endereço está dentro dos colchetes.

## Conjunto de Instruções

Mnemônico	Operandos	Opcode	Significado
<b>Instruções de Movimentação de Dados</b>			
MOV	Reg1,Reg2	0000	Reg1 $\leftarrow$ Reg2
MOV	Reg,imed	1000	Reg $\leftarrow$ imed
MOV	Reg,[end]	1001	Reg $\leftarrow$ [end]
MOV	[end],Reg	1010	[end] $\leftarrow$ Reg
<b>Instruções Aritméticas e Lógicas</b>			
ADD	Reg1,Reg2	0001	Reg1 $\leftarrow$ Reg1 + Reg2
ADD	Reg,imed	1011	Reg $\leftarrow$ Reg + imed
SUB	Reg1,Reg2	0010	Reg1 $\leftarrow$ Reg1 - Reg2
SUB	Reg,imed	1100	Reg $\leftarrow$ Reg - imed
AND	Reg1,Reg2	0011	Reg1 $\leftarrow$ Reg1 <u>e</u> Reg2
AND	Reg,imed	1101	Reg $\leftarrow$ Reg <u>e</u> imed
OR	Reg1,Reg2	0100	Reg1 $\leftarrow$ Reg1 <u>ou</u> Reg2
<b>Instruções de Manipulação de Pilha</b>			
PUSH	Reg	0101	SP--, [SP] $\leftarrow$ Reg
POP	Reg	0110	Reg $\leftarrow$ [SP], SP++
<b>Instruções de Controle de Fluxo de Execução</b>			
JMP	end	1110	PC $\leftarrow$ end
CALL	end	1111	SP--, [SP] $\leftarrow$ PC, PC $\leftarrow$ end
RET	-	0111	PC $\leftarrow$ [SP], SP++