

MATA49

Programação de Software Básico

Introdução a linguagem de montagem com Nasm

Leandro Andrade
leandrojsa@ufba.br

Sintaxe básica

- Não é sensível a letras maiúsculas e minúsculas
 - Não case-sensitive
- Para facilitar a compreensão dos programas sugere-se utilizar o seguinte padrão:
 - Uso de letra maiúscula para código
 - Uso de letra minúscula para comentários

Sintaxe básica

- Estrutura das instruções:

[Nome] [Operador] [Operandos] [;Comentários]

- Exemplo:

EXP1: MOV AX, 25 ;comando para mover

- Os campos devem ser separados por TAB ou espaços

Sintaxe básica

- [Nome]:
 - Pode ser usado como um rótulo, ou identificador de uma instrução
 - Opcional
 - Deve ter tamanho entre 1 e 4095
 - Pode conter letras, número e caracteres como `_`, `$`, `#`, `@`, `~`, `.`, `?`
 - Deve se iniciado por letras ou por `_`, `?`, e `$`

Sintaxe básica

- [Operador]:
 - Contém código de operação simbólico
 - É obrigatório
 - Exemplo: MOV, ADD, CMP, INC

Sintaxe básica

- [Operando]:
 - Instruções podem ter 0, 1, 2, 3 operandos
 - Quando mais de um são separados por vírgula
 - Quando usado dois operandos:
 - O primeiro é o destino, onde o resultado é armazenado; o conteúdo inicial é modificado
 - O segundo é o fonte; não é modificado pela instrução

Sintaxe básica

- [Comentários]:
 - São iniciadas pelo caractere ;
 - São ignoradas pelo montador
 - Opcionais
 - Importantes para descrever o que está sendo feito em cada instrução

Sintaxe básica

- Exemplo:

MOV CX,10 ;movimenta 10 para CX
(CX = 10)

ADD CX, 20 ;Adiciona 20 a CX
(CX= CX +20)

Definição de dados

- Para declaração de números:
 - Binários: 1110101b ou 1110101B
 - Decimais: 64223 ou 64223d ou 64223D
 - 1110101 é considerado decimal (ausência do B)
 - -2184D (número negativo)
 - Hexadecimais: 64223h ou 64223H ou 0x64223
 - 0FFFFh começa com um decimal e termina com h
 - 1B4Dh

Definição de dados

- Exemplos de números ilegais:
 - 1,234
 - caractere estranho (vírgula)
 - FFFFh
 - não começa por número de 0 a 9 difícil distinguir do nome de uma variável
 - 1B4D
 - não termina com h ou H

Definição de dados

- Caracteres ASCII
 - Definidos por aspas simples ou aspas duplas
 - Exemplo: “Bla” ou 'Bla'

Definição de dados

- Definição de variáveis:
 - Possui um tipo e recebe um endereço de memória
 - Usa-se pseudo-instruções para definir o tipo da variável
 - Variam de acordo com o tamanho da memória alocada
 - O montador atribui o endereço de memória

Definição de dados

- Definição de variáveis:

Pseudo-instruções	Descrição
DB	Define um byte (8 bits)
DW	Define um word (16 bits, 2 bytes consecutivos)
DD	Define um doubleword (2 palavras, 4 bytes consecutivos)
DQ	Define um quadword (4 palavras, 8 bytes consecutivos)
DT	define ten bytes (10 bytes consecutivos)

Definição de dados

- Definição de variáveis:
 - Estrutura:
 - <Nome> <Pseudo-instrução> <valor>
 - Exemplos:
 - Teste DB 0 ;equivale a 00h
 - Bli DB 10h
 - Foo DB ? ;não inicializa
 - Ex DB 0150h ;Erro!

Definição de dados

- Definição de variáveis:
 - Exemplos:
 - WORD1 DW 0h ;equivale a 0000h
 - BLI DW 0150h
 - WORD2 DW 1234h
 - byte baixo 34h, endereço WORD2
 - byte alto 12h endereço WORD2+1

Definição de dados

- Definição de variáveis:

- Arrays:

- Sequência de bytes ou words em sequência na memória

- Exemplo:

- BARRAY DB 10h,20h,30h

- WARRAY DW 1000h,123h,0h,0FFFFh

- Pode conter string:

- db 'hello' ; string

- db 'h','e','l','l','o' ; equivalente

Definição de dados

- Constantes

- Para um nome a uma constante utiliza-se da pseudo-instrução EQU
- Sintaxe: <Nome> EQU <valor_constante>
- Exemplos:

LF EQU 0Ah ;character Line Feed como LF

CR EQU 0Dh ;character Carriage return como CR

LINHA1 EQU 'Digite seu nome completo'

MENSAGEM DB LINHA1,LF,CR

Instruções de movimentação de dados

- São usadas para mover o conteúdo dos operadores
- Semelhantes ao comando de atribuição das linguagens de alto nível

Instruções de movimentação de dados

- MOV:
 - Transferência de dados entre células de memória, registradores e o acumulador.
 - É feita uma **cópia dos dados**
 - Sintaxe: MOV <destino>, <fonte>
 - Os operadores devem possuir mesmo tamanho
 - Os dois operando não podem ser memória

Instruções de movimentação de dados

- MOV:
 - Possíveis combinações:

Operando fonte	Operando destino		
	Registrador de dados	Registrador de segmento	Posição de memória
Reg. de dados	sim	sim	sim
Reg. de segmento	sim	não	sim
Posição de memória	sim	sim	não
Constante	sim	não	não

Instruções de movimentação de dados

- MOV

- Exemplos:

- MOV AX,[WORD1]

- MOV AH,'A'

- MOV AH,41h

- MOV AH,BL

- MOV AX,CS

Instruções de movimentação de dados

.data

bVal DB 100

bVal2 DB 0

wVal DW 2

dVal DD 5

.code

mov ds,45

Move imediato para DS não permitido

mov esi,[wVal]

Tamanho incompatível

mov eip,[dVal]

EIP não pode ser destino

mov 25,[bVal]

Valor imediato não pode ser destino

mov [bVal2],[bVal]

Move memória-a-memória não permitido

Instruções de movimentação de dados

- MOV

- Não é permitido movimentação direta entre operandos memória

MOV [WORD1],[WORD2] ;instrução inválida

Solução:

MOV AX, [WORD2]

MOV [WORD1], AX

Instruções de movimentação de dados

- MOV:
 - Limitações:
 - Não é possível movimentar dados em operandos de tamanho diferentes

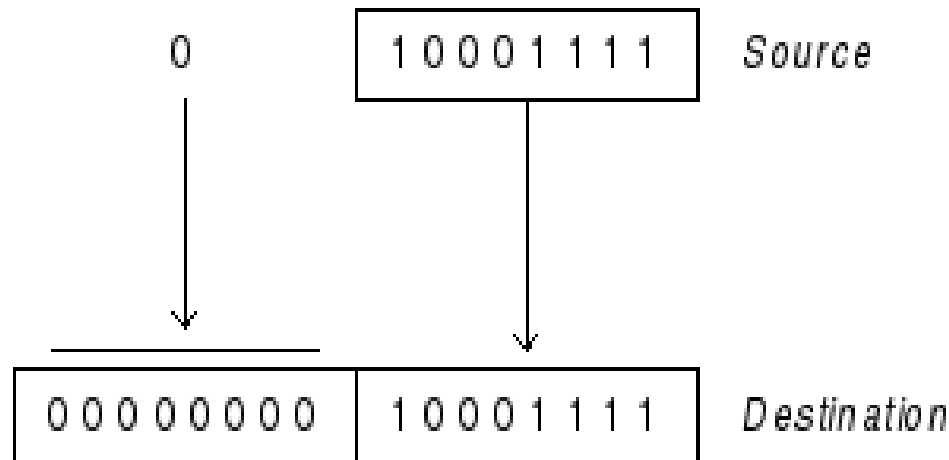
MOV AX, AH

 - Solução: MOVZX e MOVSX

Instruções de movimentação de dados

•MOVZX:

- Quando se copia um valor menor para um destino maior, a instrução MOVZX estende a parte mais significativa do destino com zeros.
- O destino deve ser um registrador

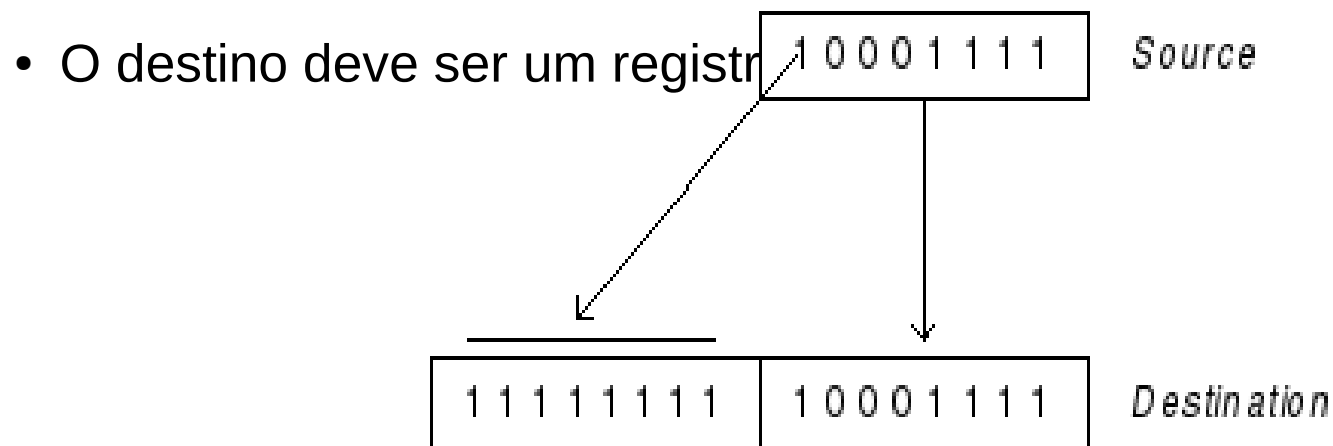


```
mov bl,10001111b  
movzx ax,bl ; zero-extension
```

Instruções de movimentação de dados

•MOVSX:

- A instrução MOVSX preenche a parte mais significativa do destino com uma cópia do bit de sinal do operando fonte.



```
mov bl,10001111b
```

```
movsx ax,bl
```

; sign extension

Instruções de movimentação de dados

- XCHG:
 - XCHG troca os valores de dois operandos
 - Pelo menos um operando deve ser um registrador.
 - Não é permitido operandos numéricos (operando imediatos)

Operando fonte	Operando destino	
	Registrador de dados	Posição de memória
Reg. de dados	sim	sim
Reg. de segmento	não	não
Posição de memória	sim	não

Instruções de movimentação de dados

- XCHG

- Exemplo:

XCHG AX,[WORD1] ;troca o conteúdo da posição de memória WORD1 com o do registrador AX

XCHG [VAR1], [VAR2] ;Não permitido

Instruções para operações aritméticas simples

Operações Aritméticas

- ADD e SUB
 - Soma e subtração
 - Sintaxe:
 - ADD destino, fonte
 - SUB destino, fonte
 - Combinações de operandos:

Operando fonte	Operando destino	
	Registrador de dados	Posição de memória
Reg. de dados	sim	sim
Posição de memória	sim	não
Constante	sim	sim

Operações Aritméticas

- ADD e SUB
 - Exemplos:
 - ADD AX,BX
 - ADD AX,[WORD1]
 - SUB [WORD2],AX
 - SUB BL,5

Operações Aritméticas

- INC e DEC
 - Incrementa/Decrementa 1 ao conteúdo do operando
 - Somente 1 operando
 - Registrador ou memória
 - Exemplo:
INC CX
DEC WORD1

Operações Aritméticas

- NEG

- Substitui o operando pelo seu complemento de 2
- 1 operando

- Registrador ou memória

- Exemplo:

(BX = 00110001b)

NEG BX (BX = 11001111b)

Operações Aritméticas

- Resolvendo operações aritméticas:

$$Rval = Xval - (-Yval + Zval)$$

```
mov ebx,[Yval]  
neg ebx  
add ebx,[Zval]  
mov eax,[Xval]  
sub eax,ebx  
mov [Rval],eax
```

Operações Aritméticas

- Exercício

BLA DW 25

BLI DW 18

BLE DW 9

RESULT DW 0

Resolva em assembly:

a) $RESULT = BLA - (-(BLI + BLE))$

b) $RESULT = BLI + BLA + (BLA - BLE)$

Operações Aritméticas

• Exercício

a) $RESULT = BLA - (-(BLI + BLE))$

```
MOV  AX, [BLI]
ADD  AX, [BLE]
NEG  AX
MOV  BX, [BLA]
SUB  BX, [AX]
MOV  [RESULT], BX
```

Operações Aritméticas

- Exercício

b) $RESULT = BLI + BLA + (BLA - BLE)$

```
MOV AX, [BLE]
```

```
NEG [BLE]
```

```
MOV BX, [BLA]
```

```
SUB BX, [BLA]
```

```
ADD BX, [BLA]
```

```
ADD BX, [BLI]
```

```
MOV [RESULT], BX
```

Operações Aritméticas

- Como resolveríamos o item a se:
 - RESULT DD 0
 - BLA DD 9
- a) $\text{RESULT} = \text{BLA} - (-(\text{BLI} + \text{BLE}))$