



## LINGUAGEM DE MONTAGEM (8086) – PARTE I

### Objetivos

1. Estudar e executar programas em linguagem “Assembler”;
2. Familiarizar-se com os comandos e recursos do montador;
3. Estudar instruções do 8086 e serviços da BIOS/DOS via INT (interrupção de *software*).

### Roteiro

1. Implemente, execute e entenda o primeiro programa listados na tabela I.
2. Faça um programa para escrever a operação matemática ( $5+2=7$ ) na tela de saída.
3. Faça um programa para escrever a frase “Oi mundo !” na tela de saída.
4. Implemente, execute e entenda os demais programas listados na tabela I.

Tabela I – Listagens de programas de exemplo.

	Leitura com eco do teclado do PC	Leitura sem eco do teclado do PC
org 100h  mov ax, 5 mov cx, 2 add cx, ax  mov dl, '=' mov ah, 2 int 21h  add cx, 48 mov dl, cl mov ah, 2 int 21h  mov ah, 0 int 16h  ret	ORG 100h CALL KEY .EXIT  KEY: MOV AH, 1 INT 21h XOR AL, 1110011b CLC JNZ KEY1 STC KEY1: RET END	ORG 100h CALL KEYS .EXIT  KEYS: MOV CX, 10 XX: MOV AH, 6h MOV DL, 0FFh INT 21h JZ XX STOSB LOOP XX RET END
<b>Programa para lê duas linhas de dados do teclado usando a INT 21h</b>		
ORG 100h JMP INICIO BUF1 DB 16 DUP (?) ; define buffer 1 (VARIÁVEL) BUF2 DB 16 DUP (?) LIN DB 0Dh, 24h INICIO: MOV BUF1, 14 ; no. de caracteres a serem lidos MOV DX, OFFSET BUF1 ; endereço do BUF1 CALL LINE ; leitura da linha MOV DX, OFFSET LIN MOV AH, 09h INT 21h MOV BUF2, 14 MOV DX, OFFSET BUF2 CALL LINE .EXIT LINE: MOV AH, 0AH ; seleciona função 0AH do DOS/BIOS INT 21h ; acessa o interrupção do DOS/BIOS RET ; retorna do procedimento  END ; fim de listagem		

### Questões adicionais

- a) Comente PROG\_1 linha por linha, e justifique para quais entradas no teclado teremos CF = 1, e para quais entradas teremos CF = 0 (comprove o valor de CF a partir do botão “flags” no EMU8086).
- b) Comente PROG\_2 linha por linha. A instrução STOSB altera um trecho da memória RAM. Usando a opção “view” → “memory” no menu do EMU8086 documente o trecho da memória RAM alterada, e justifique os endereços das posições de memória alteradas (assim como os valores armazenados nestas posições de memória) a partir dos registradores utilizados pela instrução STOSB – documente os valores destes registradores.
- c) Comente PROG\_3 linha por linha. Usando a opção “view” → “memory” no menu do EMU8086 documente os bytes do trecho da memória RAM que armazena BUF1 e BUF2 – justifique como identificou os endereços que armazenam BUF1 e BUF2.
- d) Escreva um programa em linguagem “Assembly” que pergunte na tela do PC os seguintes dados pessoais: Nome, Telefone e Endereço. Logo após a digitação dos dados, o programa deverá repetir logo abaixo todos os dados digitados, conforme mostra a figura abaixo. Escreva um programa em linguagem “Assembler” que pergunte na tela do PC os seguintes dados pessoais: Nome, Telefone e Endereço. Logo após a digitação dos dados, o programa deverá repetir logo abaixo todos os dados digitados, conforme mostra a figura abaixo.

Nome: (digitar nome)
Telefone: (digitar telefone)
Endereço: (digitar endereço)
(nome digitado)
(telefone digitado)
(endereço digitado)

- e) Escreva um programa em linguagem “Assembly” que pergunte na tela do PC a senha do usuário. A leitura do teclado deve ser sem ECHO e a após cada dígito inserido o programa deve mostrar o caractere “\*”. Logo após a digitação da senha, o programa deverá repetir logo abaixo a senha digitada.

Comente detalhadamente todos os programas estudados, descrevendo cada comando do montador e instruções do processador.