

INDICE :

A - Introducao

B - Funcoes

1. Funcao 1 - Status
2. Funcao 2 - Endereco do page frame
3. Funcao 3 - Numero total de paginas existentes e disponiveis
4. Funcao 4 - Aloca paginas
5. Funcao 5 - Mapea paginas
6. Funcao 6 - Desaloca paginas
7. Funcao 7 - Versao da placa
8. Funcao 8 - Salva estado dos registradores
9. Funcao 9 - Restaura estado dos registradores
10. Funcao 10 - Reservada (Endereco dos registradores)
11. Funcao 11 - Reservada (paginas associadas a um handle)
12. Funcao 12 - Numero de handles ativos
13. Funcao 13 - Numero de paginas associadas a um handle
14. Funcao 14 - Handles ativos com numero de paginas associadas
15. Funcao 15 - Mostra ou seta o conteudo dos registradores
16. Variaveis utilizadas nos exemplos das funcoes
17. Codigos de retorno

C - Exemplos de utilizacao da placa EMS

1. Exemplo 1 : Alocacao de paginas
2. Exemplo 2 : Mapeamento de paginas
3. Exemplo 3 : Desalocacao de paginas
4. Exemplo 4 : Salva conteudo dos registradores
5. Exemplo 5 : Restaura conteudo dos registradores

A - Introducao :

- Este manual mostra como utilizar e programar a placa EMS.
- Em cada placa de 2 Mbytes, temos 128 paginas de 16 Kbytes.
- Podem ser mapeadas 4 paginas simultaneamente, que ficarao alocadas em um bloco de 64 Kbytes (page frame). Caso nao existam 64 Kbytes livres, volta mensagem de erro. Este bloco devera estar compreendido entre 0C000H e 0EFFFH da memoria do micro I7000 PCXT. Cada uma dessas 4 paginas estara associada a um registrador, que sera a pagina fisica a ser acessada. O acesso a placa de expansao com ate 8 Mbytes e feito pelo mapeamento das 4 paginas fisicas.
- Para tratamento da placa EMS, sera utilizada a Interrupcao 67 H, e para isto, devera ser instalada como device driver.
- Para reservar um "bloco de memoria", deve-se alocar paginas de 16 Kbytes; este numero deve ser colocado em hexadecimal. Ao serem alocadas, o sistema retorna um handle, que ficara associado a elas, ate serem desalocadas.
- Para a inicializacao do Device, existem alguns parametros que podem ser usados, que sao :
 - a) /I : indica o inicio da busca para o endereco fisico do "page-frame".

Ex: Deseja-se que a busca comece em 0D000H, portanto:

DEVICE=EMM.SYS /I=D000
 - b) /E : indica a regioa que deve ser excluida da busca do endereco fisico do "page-frame".

Ex: Deseja-se que a busca nao seja feita entre 0C000H e 0C800H, portanto:

DEVICE=EMM.SYS /E=C000-C800
- obs: podem ser passados os dois parametros ao mesmo tempo :

Ex: DEVICE=EMM.SYS /I=D000/E=C000-C800

- mapa de memoria p/ PCXT II :

memoria logica (1Mb)		memoria fisica (2Mb)	
0000	640 Kb	n da pag.--> 0	16 Kb
	RAM	1	16 Kb
	-----	2	16 Kb
		3	16 Kb
		.	.
C000	//////////	.	.
	//////////	.	16 Kb
	-----	.	16 Kb
	janela de 64 Kb	.	16 Kb
	//////////	.	.
	//////////	.	.
FFFF	-----	126	16 Kb

B - Funcoes :

Para a utilizacao da placa EMS, existem diversas funcoes que auxiliam o usuario nas alocacoes de paginas, mapeamento, etc.

- As funcoes de 1 a 7, servem para aplicacoes de utilizacao da EMS.
- As funcoes 8 e 9, sao necessarias sempre que se necessitar de alteracoes nos registradores de mapeamento, como device drivers, programas residentes, etc.
- As funcoes 10 e 11 sao reservadas, e atualmente nao sao utilizadas, porem existem alguns antigos aplicativos que as utilizam.
- As funcoes 12, 13 e 14 servem para mostrar o estado da EMS, e como ela esta sendo usada.
- A funcao 15 devera ser utilizada para sistemas de MULTI-TAREFAS, onde se necessita do estado dos registradores de cada placa, e ainda, altera-los.

1. Funcao 1 (40H) : Status

a) Funcao : retorna o codigo de status atual, ou seja, se esta tudo funcionando perfeitamente, ou se ocorreu algum problema de hardware ou software.

b) Entrada : AH = 40H (codigo da funcao)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Exemplo :

```
MOV    AH,40H
INT    67H
CMP    AH,00
JNE    ERRO
```

2. Funcao 2 (41H) : Endereco do page frame

a) Funcao : retorna o segmento do endereco do "page frame", ou inicio do bloco de 64Kbytes.

b) Entrada : AH = 41H (codigo da funcao)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Saída : BX = segmento do endereço

e) Exemplo :

```
MOV    AH,41H
INT     67H
CMP     AH,00
JNE     ERRO
MOV     SEGMENT0,BX
```

3. Função 3 (42H) : Número total de páginas existentes e disponíveis

a) Função : retorna o número de páginas disponíveis e o número total de páginas existentes na placa. Se não existirem páginas disponíveis, retorna '0' em BX.

b) Entrada : AH = 42H (código da função)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Saída : BX = Número de páginas disponíveis
DX = Número total de páginas existentes

e) Exemplo :

```
MOV    AH,42H
INT     67H
CMP     AH,00
JNE     ERRO
MOV     PAG_LIVRE,BX
MOV     PAG_TOTAL,DX
```

4. Função 4 (43H) : Aloca páginas

a) Função : Aloca as páginas requeridas pelo usuário, retornando o handle associado a elas. Não podem ser alocadas '0' (zero) páginas; só podem ser alocadas páginas, se existirem páginas suficientes e disponíveis na placa; e ainda se existir handle livre.

b) Entrada : AH = 43H (código da função)
BX = número de páginas a serem alocadas

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H
AH = 85H
AH = 87H
AH = 88H
AH = 89H

d) Saida : DX = handle associado as paginas alocadas

e) Exemplo :

```
MOV    AH,43H
MOV    BX,NUM_PAG
INT    67H
CMP    AH,00
JNE    ERRO
MOV    HANDLE,DX
```

5. Funcao 5 (44H) : Mapea paginas

a) Funcao : Mapea a pagina logica requerida pelo usuario para a pagina fisica. So podem ser mapeadas paginas que estiverem dentro das especificacoes de extensao.

b) Entrada : AH = 44H (codigo da funcao)
BX = pagina logica (extensao 0)
AL = pagina fisica (extensao : 0,1,2 ou 3)
DX = handle

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 83H
AH = 84H
AH = 8AH
AH = 8BH

d) Exemplo :

```
MOV    AH,44H
MOV    BX,PAG_LOG
MOV    AL,PAG_FIS
MOV    DX,HANDLE
INT    67H
CMP    AH,00
JNE    ERRO
```

6. Funcao 6 (45H) : Desaloca paginas

a) Funcao : Desaloca paginas requeridas pelo usuario, atraves do handle associado a elas. So podem ser desalocadas, as paginas que nao estiverem salvas (funcao 8) ; caso estejam, deve-se restaurar antes (funcao 9).

b) Entrada : AH = 45H (codigo da funcao)

DX = handle

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 83H
AH = 84H
AH = 86H

d) Exemplo :

```
MOV    AH,45H
MOV    DX,HANDLE
INT    67H
CMP    AH,00
JNE    ERRO
```

7. Funcao 7 (46H) : Versao da placa

a) Funcao : retorna a versao da placa, que no caso e 3.2

b) Entrada : AH = 46H (codigo da funcao)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Saida : AL = versao da placa. No nibble mais significativo temos (3), e no nibble menos significativo (2).

e) Exemplo :

```
MOV    AH,46H
INT    67H
CMP    AH,00
JNE    ERRO
MOV    VERSAO,AL
```

8. Funcao 8 (47H) : salva estado dos registradores

a) Funcao : salva o conteudo dos registradores de mapeamento para todas as placas instaladas. So podem ser salvos, os registradores que ainda nao o tenham sido.

b) Entada : AH = 47H (codigo da funcao)
DX = handle

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 83H
AH = 84H
AH = 8CH
AH = 8DH

d) Exemplo :

```
MOV     AH,47H
MOV     DX,HANDLE
INT     67H
CMP     AH,00
JNE     ERRO
```

9. Funcao 9 (48H) : Restaura conteudo dos registradores

a) Funcao : restaura o conteudo dos registradores de mapeamento, para todas as placas instaladas. so podem ser restaurados, os registradores que ja tenham sido salvos.

b) Entrada : AH = 48H (codigo da funcao)
DX = handle

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 83H
AH = 84H
AH = 8EH

d) Exemplo :

```
MOV     AH,48H
MOV     DX,HANDLE
INT     67H
CMP     AH,0
JNE     ERRO
```

10. Funcao 10 (49H) : Reservada (Endereco dos registradores)

a) Funcao : Retorna em um array, o endereco dos registradores de mapeamento de cada placa instalada.

11. Funcao 11 (4AH) : Reservada (paginas associadas a um certo handle)

a) Funcao : Retorna em um array, as paginas associadas a um certo handle.

12. Funcao 12 (4BH) : Numero de handles ativos

a) Funcao : Retorna o numero de handles ativos

b) Entrada : AH = 4BH (codigo da funcao)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Saida : BX = numero de handle ativos

e) Exemplo :

```
MOV    AH,4BH
INT     67H
CMP     AH,00
JNE     ERRO
MOV     HANDLE_ATIVO,BX
```

13. Funcao 13 (4CH) : Numero de paginas associadas a um handle

a) Funcao : Retorna o numero de paginas alocadas de um dado handle

b) Entrada : AH = 4CH (codigo da funcao)
DX = handle

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 83H
AH = 84H

d) Saida : BX = numero de paginas associadas ao handle .
nca exce- dera 512.

e) Exemplo :

```
MOV     AH,4CH
MOV     DX,HANDLE
INT     67H
CMP     AH,00
JNE     ERRO
MOV     NUM_PAG_ALOC,BX
```

14. Funcao 14 (4DH) : Handles ativos com o numero de paginas associadas

a) Funcao : Retorna em um array, os handles ativos, com os respectivos numeros de paginas associadas. Retorna ainda, o numero de handles ativos.

b) Entrada : AH = 4DH (codigo da funcao)
ES:DI = endereco inicial do array onde sera copiada a tabela. Na primeira word, teremos o handle, e na segunda, o numero de paginas associadas.

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H

d) Saída : numero de handles ativos, e tabela transferida

e) Exemplo :

```
MOV    AX,SEG
MOV    ES,AX
MOV    DI,OFFSET HANDLE_ARRAY
XOR    AX,AX
MOV    AH,4DH
INT    67H
CMP    AH,00
JNE    ERRO
MOV    HANDLE_ATIVO,BX
```

15. Funcao 15 (4EH) : Mostra ou seta o conteudo dos registradores

15.0 Funcao 15/0 : Mostra o conteudo dos registradores

a) Funcao : Retorna em um array o conteudo dos registradores de mapeamento.

b) Entrada : AH = 4EH (codigo da funcao)
AL = 0 (codigo da sub-funcao)
ES:DI = endereco inicial do array

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H
AH = 8FH

d) Saída : tabela transferida

e) Exemplo:

```
MOV    AX,SEGME
MOV    ES,AX
MOV    DI,OFFSET REG_ARRAY
MOV    AH,4EH
MOV    AL,00
INT    67H
CMP    AH,00
JNE    ERRO
```

15.1 Funcao 15/1 : Setar conteudo dos registradores

a) Funcao : Grava nos registradores um novo conteudo definido em um array pelo usuario.

b) Entrada : AH = 4EH (codigo da funcao)
AL = 1 (codigo da sub-funcao)
DS:SI = endereco inicial do array

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H
AH = 8FH

d) Saida : Tabela transferida para os registradores

e) Exemplo :

```
MOV    AX,SEGMENTO
MOV    DS,AX
MOV    SI,OFFSET NOVO_ARRAY
MOV    AH,4EH
MOV    AL,01
INT    67H
CMP    AH,00
JNE    ERRO
```

15.2 Funcao 15/2 : Mostra e seta o conteudo dos registradores

a) Funcao : Copia o conteudo dos registradores em uma posicao de memoria, e grava um novo conteudo nos mesmos (Funcoes 0 e 1).

b) Entrada : AH = 4EH (codigo da funcao)
AL = 2 (codigo da sub-funcao)
ES:DI = endereco aonde deve ser copiada a tabela
DS:SI = endereco aonde esta a nova tabela a ser copiada

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H
AH = 8FH

d) Saida : Tabela copiada em uma posicao de memoria e registradores com novo conteudo

e) Exemplo :

```
MOV    AX,SEGMENTO
MOV    ES,AX
MOV    DI,OFFSET REG_ARRAY
MOV    DS,AX
MOV    DI,OFFSET NOVO_ARRAY
MOV    AH,4EH
MOV    AL,02
INT    67H
CMP    AH,00
```

15.3 Funcao 15/3 : Numero de bytes transferidos

a) Funcao : Retorna o numero de bytes transferidos pelas funcoes 15/0, 15/1, 15/2.

b) Entrada : AH = 4EH (codigo da funcao)
AL = 3 (codigo da sub-funcao)

c) Retorno de status : AH = 0
AH = 80H
AH = 81H
AH = 84H
AH = 8FH

d) Saida : AL = Numero de bytes transferidos

e) Exemplo :

```
MOV    AH,4EH
MOV    AL,03
INT    67H
MOV    BYTE_TRANS,AL
```

16. Variaveis utilizadas nos exemplos das funcoes :

- ERRO : rotina de saida, com erro
- SEGMENTO : segmento do endereco do "page-frame"
- PAG_LIVRE : numero de paginas livres
- PAG_TOTAL : numero total de paginas exixtentes
- NUM_PAG : numero de paginas a serem alocadas
- HANDLE : numero associado ao numero de paginas alocadas
- PAG_LOG : pagina logica a ser mapeada
- PAG_FIS : pagina fisica onde sera mapeada a pagina logica
- VERSAO : versao da placa
- HANDLE_ATIVO : numero de handles ativos
- NUM_PAG_ALOC : numero de paginas associadas ao handle
- SEG e HANDLE_ARRAY : segmento e offset do endereco de transferencia de tabela de handles ativos com os respectivos numeros de paginas
- SEGME e REG_ARRAY : segmento e offset do endereco de transferencia da tabela de conteudo dos registradores
- SEGMEN e NOVO_ARRAY : segmento e offset da tabela de trans-

.....
ferencia do novo conteudo dos regis-
tradores

- BYTE_TRANS : numero de bytes transferidos

17. Codigos de Retorno de status (em AH) :

0	-	placa instalada corretamente e Hardware funcionando
80H	-	mal funcionamento de Software
81H	-	mal funcionamento de Hardware
83H	-	nao existe o handle especificado
84H	-	nao existe o codigo da funcao
85H	-	todos os handles estao sendo utilizados
86H	-	os registradores ja foram salvos
87H	-	nao existem paginas suficientes para serem alocadas
88H	-	nao existem paginas livres suficientes p/ alocao
89H	-	nao podem ser alocadas '0' (zero) paginas
8AH	-	pagina logica fora da extensao permitida
8BH	-	pagina fisica fora da extensao permitida
8CH	-	area para salvar registradores cheia
8DH	-	registradores ja foram salvos
8EH	-	nao existe handle correspondente na area para salvar
8FH	-	nao existe a sub_funcao especificada

C - Exemplos de utilizacao da placa EMS :

1. Exemplo 1 : Alocao de paginas

- salva o segmento do endereco do page frame em "ENDERECO"
- verifica o numero de paginas disponiveis para ver se e possivel, alocar mais paginas.
- compara a volta do item anterior, com "PAGINA", para ver se o numero de paginas disponiveis e superior ao desejado. Se for segue o programa; se nao for, retorna erro.
- aloca paginas desejadas.
- salva o valor do handle retornado pelo item anterior em "HANDLE".
- salva o status da placa em STATUS.

- Programa :

```

ALOCA      PROC      NEAR

              MOV      AH,41H          ; RETORNA SEGMENTO EM BX
              INT      67H

              CMP      AH,00
              JNZ      FIM

              MOV      ENDRECO,BX      ; SALVA SEGMENTO

              MOV      AH,42H          ; RETORNA N PAGS.LIVRES
              INT      67H

              CMP      BX,PAGINA      ; VERIFICA SE EXISTEM
              JNAE     FIM

              MOV      AH,43H          ; ALOCA PAGINAS
              MOV      BX,PAGINA
              INT      67H

FIM :        MOV      STATUS,AH        ; SALVA STATUS

              RET

ALOCA      ENDP

```

2. Exemplo 2 : Mapeamento de paginas

- Mapea paginas associadas ao handle retornado pelo exemplo 1.
- Este acesso a EMS, so pode ser feito se os parametros estiverem dentro das especificacoes, que sao:
 - handle colocado em 'DX', com paginas associadas a ele.
 - pagina logica colocada em 'BX', dentro da extensao que podera ser de 0 a 127 se existir uma placa, e assim por diante.
 - pagina fisica colocada em 'AL', tambem dentro da extensao, que sera 0, 1, 2 ou 3.

- Programa :

```

MAPEA      PROC      NEAR

              MOV      AH,44H          ; MAPEA PAGINA FISICA
              MOV      DX,HANDLE
              MOV      BX,PAGINA_LOGICA
              MOV      AL,PAGINA_FISICA
              INT      67H
              MOV      STATUS,AH
              RET

MAPEA      ENDP

```

3. Exemplo 3 : Desalocacao de paginas

- Desaloca paginas associadas o handle retornado pelo exemplo 1.
- So podem ser desalocadas paginas que tiverem um handle associado, e que nao estiverem salvas (exemplo 4). Neste caso, deve-se primeiro restaurar (exemplo 3), para depois desalocar.
- Deve-se salvar o status da placa, para verificar posteriormente, se as paginas foram desalocadas.
- Programa :

```
DESALOCA PROC NEAR
          MOV AH,45H ; DESALOCA PAGINAS
          MOV DX,HANDLE
          INT 67H
          MOV STATUS,AH
          RET
DESALOCA ENDP
```

4. Exemplo 4 : Salva conteudo dos registradores

- Salva o conteudo dos registradores de mapeamento, associados a um certo handle retornado pelo exemplo 1.
- Os registradores so podem ser salvos, se nao foram salvos antes, e se o handle designado tiver paginas associadas, mapeadas.
- Programa :

```
SALVA PROC NEAR
        MOV AH,47H ; SALVA MAPEADORES
        MOV DX,HANDLE
        INT 67H
        MOV STATUS,AH
        RET
SALVA ENDP
```

- Exemplo 5 : Restaura conteudo dos registradores

- Restaura o estado dos registradores associados a um certo handle.
- Este estado so pode ser restaurado se o handle ja foi salvo antes.
- Programa :

RESTAURA PROC

NEAR

MOV
MOV
INT
MOV
RET

AH,4BH
DX,HANDLE
67H
STATUS,AH

; RESTAURA MAPEADORES

RESTAURA ENDP

A>