

# PROLOGICA

microcomputadores

SISTEMA 700  
DOS - 700

TÍTULO SISTEMA - 700	PAGINA
SUB-TÍTULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

## ÍNDICE

TÍTULO	ITEM	PAG.
I. Introdução.....	1	01
II. Descrição funcional do DOS-700.....	2	03
A. Como referenciar um arquivo.....	2.1	03
B. Selecionando 'drives'.....	2.2	05
C. Comandos do DOS-700.....	2.3	05
Comando DIR.....	2.3.1	05
Comando ERA.....	2.3.2	06
Comando REN.....	2.3.3	07
Comando TYPE.....	2.3.4	07
III. Teclas especiais de controle.....	3	08
A. Funções de controle de console.....	3.1	08
IV. Programas Utilitários.....	4	10
A. STAT.....	4.1	11
STAT.....	4.1.1	11
STAT x:.....	4.1.2	11
STAT AAA.....	4.1.3	12
STAT x:AAA.....	4.1.4	12
STAT x:=R/O		
x:=R/W.....	4.1.5	13
B. PIP.....	4.2	13
C. FORMATA.....	4.3	22
D. GERSIS.....	4.4	22
E. SUBMIT.....	4.5	25
XSUB.....	4.5.1	27
F. EDIT1.....	4.6	29
Funções do editor.....	4.6.1	30
Organização da memória.....	4.6.2	31
Operação da memória.....	4.6.3	31

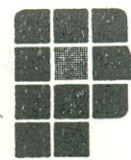
TÍTULO SISTEMA - 700	PAGINA
SUB-TÍTULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

TÍTULO	ITEM	PAG.
Pesquisa e alteração de texto.....	4.6.4	36
Bibliotecas fontes.....	4.6.5	40
Condições de erro do EDIT1.....	4.6.6	40
V. Mensagens de erro do BDOS.....	5	42

TÍTULO SISTEMA - 700	PAGINA 01
SUB-TÍTULO DOS-700	DATA



## INTRODUÇÃO

O DOS-700 é um monitor de controle de operações do Sistema 700.

O DOS-700 permite que o usuário desenvolva seus próprios programas, podendo armazená-los em discos flexíveis, para posterior utilização.

Estes discos possuem 35 trilhas de 40 setores cada uma, com 128 bytes cada setor. Duas dessas trilhas são reservadas para o sistema operacional, totalizando uma capacidade de armazenamento de 175 Kbytes.

O DOS-700 está dividido em 4 partes distintas:

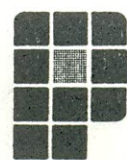
- . BIOS - SISTEMA BÁSICO DE E/S (DEPENDENTES DO HARDWARE)
- . BDOS - SISTEMA BÁSICO OPERACIONAL EM DISCO
- . CCP - PROCESSAMENTO DE COMANDO EM CONSOLE
- . TPA - ÁREA DE PROGRAMA TRANSIENTE

O BIOS é responsável por todo o gerenciamento de arquivos nos discos. Ele pode controlar um ou mais drives contendo discos com diretórios diferentes. O BDOS possui sistema dinâmico de criação de arquivos que diminui consideravelmente o movimento do cabeçote de leitura/gravação durante o acesso. Os arquivos são criados de tal forma que não possuem um tamanho definido, há apenas uma restrição: que o arquivo não exceda a capacidade de armazenamento do disco. O DOS-700 permite que cada disco contenha 64 arquivos diferentes.

O CCP é interface simbólico que permite o diálogo entre o usuário e o DOS-700. O CCP lê comandos no console e os transfere para os devidos módulos do DOS-700 para que sejam executados.

Esses comandos serão discutidos na próxima seção.

TITULO SISTEMA - 700	PAGINA 02
SUB-TITULO DOS-700	DATA

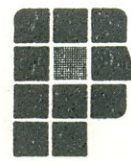


O último segmento do DOS-700 é o chamado de área de programas transientes ( TPA ).

A TPA é uma área de memória onde os programas são carregados, do disco, sob um comando do CCP, e executados.

Por exemplo, durante a edição de um programa a TPA contém o editor de textos que está sendo executado.

TÍTULO	PAGINA
SISTEMA - 700	03
SUB-TÍTULO	DATA
DOS-700	



## 2. DESCRIÇÃO FUNCIONAL DO DOS-700

O usuário se comunica com o DOS-700 através do CCP, que lê e interpreta os comandos digitados no console.

Em geral, o CCP endereça um dos vários discos 'on line' (o sistema permite que até 4 drives sejam ligados). Esses drives são endereçados através das letras A, B, C ou D. Um determinado 'drive' é acessado somente se o CCP estiver com o seu endereço, e se não estiver você deve indicá-lo.

O CCP sempre apresenta no vídeo o endereço do 'drive' seguido do sinal ">" indicando que o CCP está pronto para receber comandos.

Ao ligar o sistema, o disco com o sistema operacional deverá ser introduzido no 'drive' A, para que possa ser carregado para a memória. O sistema operacional após ser carregado, é residente em memória ocupando aproximadamente 11 Kbytes.

Após ser carregado na memória o console apresentará o símbolo "A>" indicando que o DOS-700 está endereçando o 'drive A', e que o sistema está pronto para receber comandos.

### 2.1. COMO REFERENCIAR UM ARQUIVO

O DOS-700, como já pudemos notar, acessa os seus arquivos através do nome desses arquivos. O nome de um arquivo consiste de duas partes, o nome primário e o secundário.

. O Secundário é Opcional.

pppppppp.sss

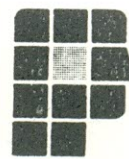
Onde: pppppppp é o nome primário e permite até 8 caracteres.

sss é o nome secundário que serve para especificar o conteúdo do arquivo.

O nome do arquivo não pode conter nenhum dos caracteres abaixo:

<> . , ; : ? \*

TITULO SISTEMA - 700	PAGINA 04
SUB-TITULO DOS-700	DATA



Para se referenciar a um arquivo ou a um grupo de arquivos de um determinado disco, o DOS-700 permite uma seleção, através de caracteres especiais como '?' e '\*'.

EXEMPLOS:

1. X?Z.C?M

Esta especificação será satisfeita por todos os arquivos que possuam o nome primário iniciado por 'X' e terminado por 'Z' que tenham qualquer letra ou número no lugar da '?', e que tenham o nome secundário iniciado por 'C' e terminado por 'M' e que possuam qualquer letra ou número no lugar da '?'.

. XYZ. COM

. X3Z. CAM

Estes nomes de arquivos fariam com que fosse satisfeita a condição acima.

2. \*.\*

Escrever desta forma é equivalente a ??????????.??? ou seja, todos os arquivos do disco.

A utilização do '\*' é a mais frequente, pois, permite as seguintes seleções:

pppppppp.\* = pppppppp.???

\*.sss = ??????????.sss

. XYZ.\* = Todos os arquivos XYZ que tenham qualquer sufixo (nome secundário)

\*.COB = Todos os arquivos que tenham o nome secundário COB.

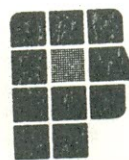
\*.C?M = Todos os arquivos que satisfaçam essa condição.

. X?Z.\* = Todos os arquivos que satisfaçam essa condição com qualquer sufixo.

3. DIR \*.\*

. Todos os arquivos existentes no disco.

TÍTULO SISTEMA - 700	PÁGINA 05
SUB-TÍTULO DOS-700	DATA



#### 4. DIR X.Y

. Especifica somente o nome do arquivo X.Y

#### 5. DIR X?Z.C?M

. Todos os arquivos que satisfaçam essa condição.

### 2.2. SELECIONANDO 'DRIVES'

O operador pode selecionar o 'drive' em que deseja trabalhar, digitando somente o nome do 'drive' (A,B,C ou D) seguido de dois pontos (:).

O comando DIR, abaixo exemplifica o que foi dito.

### 2.3. COMANDOS DO DOS-700

Os comandos abaixo fazem parte do sistema operacional e encontram-se residentes na memória.

- . DIR - Lista nomes de arquivos contidos em disco
- . ERA - Apaga arquivos especificados
- . REN - Troca nome de arquivo
- . TYPE- Lista conteúdo do arquivo especificado

Estes comandos serão descritos com detalhes mais adiante.

#### 2.3.1. COMANDO DIR

O comando DIR permite verificar o conteúdo de um disco em um 'drive' especificado.

Quando o nome do 'drive' for omitido, é assumido o 'drive' corrente.

Seu formato genérico é:

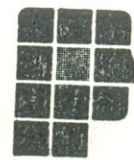
. DIR d :nnn

onde:- d é o nome do 'drive'

nnn é qualquer especificação de nome arquivo



TÍTULO SISTEMA - 700	PAGINA 06
SUB-TÍTULO DOS-700	DATA



EXEMPLOS:

```

1  A> DIR
      A: EX1      COB  A:  EX2    A:  EX1      BAK
2  A> B:
3  B> DIR *.COM
      B: COBOL  COM
4  B> A:
      A>

```

1. Lista todos os arquivos contidos no 'drive' corrente. No caso acima é o 'drive A'.
2. Passa o comando para o 'Drive B'.
3. Lista todos os arquivos que tenham como sufixo o 'string' COB, no 'drive' corrente, no caso o 'B'.
4. Retorna o comando para o 'drive A'.

2.3.2. COMANDO ERA

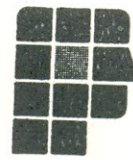
Este comando apaga o arquivo ou arquivos especificados.

. ERA nnn

Ele permite as seguintes sintaxes:

- . A>ERA X.Y            O arquivo chamado X.Y é apagado do disco 'A'.
- . A>ERA X.\*            Todos os arquivos com nome primário X são apagados do disco 'A'
- . A>ERA \*.COB        Todos os arquivos com o nome secundário COB são apagados do disco 'A'.
- . A>ERA X?Y.C?M      Todos os arquivos que satisfaçam a esta condição são apagados do disco 'A'.
- . A>ERA B:\*.\*        Apaga todos os arquivos do disco 'B' mas antes de apagar o CCP solicita confirmação: ALL ( Y/N )?

TÍTULO	PAGINA
SISTEMA - 700	07
SUB-TÍTULO	DATA
DOS-700	



que requer somente Y como resposta para apagar o disco.

. A>ERA B:\*.PRN

Todos os arquivos no disco 'B' que contêm PRN como parte secundária são apagados.

OBS.: Todas as referências ao 'drive' B, C, D ou A são independentes do 'drive' corrente.

### 2.3.3. COMANDO REN

Este comando troca o nome do arquivo pelo nome especificado.

A sintaxe é a seguinte:

. REN <drive> :B= <drive> :A

onde:

- . A = nome do arquivo que se deseja alterar
- . B = novo nome do arquivo

EXEMPLO:

. A REN X.Y=Q.R

O arquivo cujo o nome é Q.R é mudado para X.Y no disco 'A'.

### 2.3.4. COMANDO TYPE

O comando 'TYPE' mostra o conteúdo, em ASCII, de arquivo fonte.

SINTAXE:

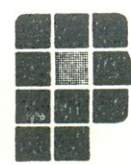
. TYPE <drive>:<nome do arquivo>

Exemplo:

. A TYPE B: X.Y

O conteúdo do arquivo X.Y do disco 'B' é listado, no vídeo.

TÍTULO SISTEMA - 700	PAGINA 08
SUB-TÍTULO DOS-700	DATA



### 3. TECLAS ESPECIAIS DE CONTROLE

O CCP permite que certas teclas especiais sejam digitadas durante a edição de um programa ou mesmo quando se está no sistema operacional.

Estas teclas são descritas abaixo:

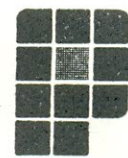
- . CTRL-U - Cancela linha digitada no vídeo e habilita para digitação de uma nova linha, na linha seguinte.
- . CTRL-X - Cancela linha digitada no vídeo e habilita para digitação de uma nova linha, na mesma linha.
- . CTRL-R - Copia linha digitada.
- . CTRL-E - Fim físico de uma linha de comando, o cursor retorna ao início de uma nova linha, porém o comando não é interpretado até que se digite a tecla 'ENTER' ou 'RETURN'.
- . CTRL-C - Recarrega o DOS-700 do disco para memória.

#### 3.1. FUNÇÕES DE CONTROLE DE CONSOLE

As funções descritas abaixo não estão implementadas na versão atual do COBOL. Elas somente funcionarão quando o sistema estiver aguardando comandos através do CCP, ou operando com FORTRAN ou BASIC.

- . CTRL-P - Após digitar essas duas teclas o sistema passa a listar na impressora tudo que é digitado no vídeo. O CTRL-P pode ser digitado em qualquer ponto da linha de comando.  
  
Sua execução termina quando for digitado um segundo CTRL-P.
- . CTRL-S - Parada temporária de processamento. A execução do programa continua normalmente quando o próximo caractere for digitado no console (um outro CTRL-S). O CTRL-S é usado para parar a saída de dados em consoles de alta velocidade, como é o caso do vídeo do Sistema 700.

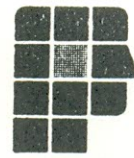
TÍTULO	PAGINA
SISTEMA - 700	09
SUB-TÍTULO	DATA
DOS-700	



**PROLOGICA**  
microcomputadores

Observe que estas funções (CTRL-TECLA), somente funcionarão quando pressionadas simultaneamente, ou seja, primeiro CTRL e depois a tecla desejada.

TITULO SISTEMA - 700	PAGINA 10
SUB-TITULO DOS-700	DATA



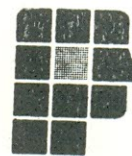
**PROLOGICA**  
microcomputadores

#### 4. PROGRAMAS UTILITÁRIOS

Os programas utilitários estão armazenados no disco de sistema operacional e são carregados para memória através de comandos via CCP.

Observe que a sintaxe de chamada desses programas é semelhante à dos comandos descritos a partir do item 2.3 deste manual.

TITULO	PAGINA
SISTEMA - 700	11
SUB-TITULO	DATA
DOS-700	



**PROLOGICA**  
microcomputadores

#### 4.1 STAT

O STAT pode nos fornecer informações a respeito do disco:

- . Quanto a ocupação
- . Quanto ao número de 'bytes' ainda disponíveis
- . Características do disco

O sintaxe genérica do STAT é a seguinte:

A>STAT Linha de Comando

Os possíveis comandos serão descritos abaixo.

##### 4.1.1. STAT

Se o operador digitar uma linha de comando vazia o STAT, calcula o número de 'bytes' restantes em todos os 'drives' ativos até o 'drive' corrente e imprime a seguinte mensagem:

x:R/W, espaço: nnnk

ou

x:R/O, espaço: nnnk

onde:

x = drive ativo

R/W = Read/Write o disco está pronto para ser lido ou gravado.

R/O = Read Only o disco só pode ser lido

nnn = número de bytes restantes no drive x

$0 \leq nnn \leq 163$

##### 4.1.2. STAT X:

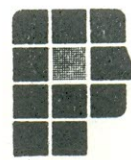
Se o nome de 'drive' é dado, o STAT calcula o número de 'bytes' disponíveis naquele 'drive' e imprime uma mensagem.

Por exemplo:

- . Se X = B, a mensagem seria a seguinte:

BYTES RESTANTES NO B: nnnk

TÍTULO SISTEMA - 700	PAGINA 12
SUB-TÍTULO DOS-700	DATA



#### 4.1.3. STAT AAA

A linha de comando pode especificar um conjunto de arquivos a ser listado, no comando acima, AAA é um conjunto de arquivos. Como se referenciar a este conjunto de arquivos está descrito na seção 2.1 deste manual.

Após o comando a seguinte mensagem é impressa:

```
. REGS BYTS EX D:<nome arq.> tipo
. RRRR bbbk EE D: pppppppp.sss
```

onde:

- . RRRR = Número de registros de 128 bytes, cada um, que foram gravados no disco.
- . bbb = Número de Kbytes alocados no arquivo  
bbb = RRRR \* 128/1024
- . EE = Número de extensões de 16 Kbytes, cada uma.  
EE = bbb/16
- . D = Nome do 'drive' que contém o arquivo
- . ppp.. = Nome primário do arquivo
- . sss = Nome secundário do arquivo

Ao término da listagem, é informado o número de 'bytes' restantes do disco.

#### 4.1.4. STAT X:AAA

Para facilitar na operação, o nome do 'drive' pode preceder o conjunto de arquivos a listar. Neste caso antes de iniciar a listagem o 'drive' é selecionado automaticamente.

#### EXEMPLO:

```
A>STAT B:*.COB
```

Neste exemplo será dada uma posição de todos os arquivos do 'drive B' cujo o nome secundário é COB.

TÍTULO SISTEMA - 700	PAGINA 13
SUB-TÍTULO DOS-700	DATA



#### 4.1.5. STAT X:=R/O X:=R/W

Originalmente um disco é colocado para trabalhar pelo sistema R/W (Read-Write). Este sistema pode ser alterado através do comando acima, seria um método de proteção do disco.

Se for selecionado o primeiro comando o disco X será habilitado somente para leitura (R/O = Read Only).

Se o segundo comando for selecionado o disco volta a seu estado original (R/W = Read Write )

Se por acaso for feita alguma tentativa de gravação num disco que está no modo R/O o BDOS imprimirá a seguinte mensagem:

```
. BDOS ERR ON x: READ ONLY
```

Onde:

```
. X é o nome do 'drive'
```

Neste momento o DOS-700 aguarda que seja digitada qualquer tecla.

A digitação de qualquer tecla faz com que a execução do comando pare, e retorna ao DOS-700.

Neste momento o disco se torna R/W.

#### 4.2 PIP

O PIP é um programa que permite copiar, listar e combinar arquivos em disco.

O PIP é iniciado digitando-se um dos seguintes comandos:

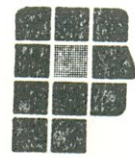
(1) PIP cr

(2) PIP linha de comando cr

Em ambos os casos, PIP é carregado para dentro do TPA e executado.

No caso 1 o PIP lê diretamente no console as linhas de comando avisando com um '\*' que está pronto para receber comandos.





Para encerrar o operação basta digitar 'cr'. As regras da sintaxe para as linhas de comando serão descritas abaixo.

No caso 2 do PIP os comandos são equivalentes aos do caso 1 exceto que somente uma linha de comando é lida e automaticamente executada, ao termino da cópia, o PIP não permite que outras linhas de comando sejam digitadas como acontece no caso 1.

O formato de cada linha de comando é:

Destino = Origem 1, origem 2, ..., origem n <cr>

onde:

Destino = Arquivo ou periférico que receberá os dados.

Origem 1...n = 0 nome de sequência de nomes de arquivos serão copiados da esquerda para direita, para o destino.

Quando vários arquivos são referenciados na linha de comando (n>1) é assumido que cada arquivo contenha caracteres em ASCII e um EOF (CTRL-Z) no fim de cada arquivo.

Os caracteres minúsculos são traduzidos, internamente para maiúsculos, para que fiquem compatíveis com as convenções do DOS-700.

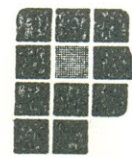
Finalmente, a linha de comando não pode exceder o limite de 255 caracteres.

Os nomes dos arquivos de origem e destino devem ser formados de acordo com as regras descritas no ítem 2.1 deste manual e podem ou não ser precedidos do nome do 'drive' (A,B,C, ou D) que definem o disco onde o arquivo será lido e/ou gravado.

Quando o nome do 'drive' não é incluído na linha de comando é assumido o 'drive' corrente.

Se mais de um arquivo de origem for referenciado eles serão copiados para o arquivo de destino, um a um, da esquerda para direita.

TÍTULO SISTEMA - 700	PAGINA 15
SUB-TÍTULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

Se o arquivo de destino já existir ele é destruído, e seu conteúdo atual será igual a linha de comando (o arquivo não é destruído se houver erro na linha de comando).

As seguintes linhas de comando são válidas para o PIP:

X=Y <cr>

copia o arquivo Y para o X.

o conteúdo de Y não é alterado.

X=Y,Z <cr>

copia os arquivos Y e Z para o X, sendo que Y é copiado e logo em seguida o Z é copiado.

O conteúdo de Y e Z não são alterados, X conterá Y e Z.

X.COB=Y.COB.Z.COB,FIM.COB <cr>

Cria o arquivo X.COB que conterá os arquivos Y, Z e FIM do tipo COB.

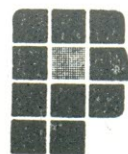
NOVO.ARQ =B:VELHO.ARQ <cr>

Faz a cópia do VELHO.ARQ que se encontra no 'drive B', para o NOVO.ARQ que se encontrará no 'drive' corrente.

B:A.U=B:B.V,A:C.W,D.X. (cr)

Concatena o arquivo B.V do 'drive B' com o C.W do 'drive' 'A' e o D.X. do 'drive' corrente cria o arquivo A.U. no 'drive' 'B'.

Para uma utilização mais simples e rápida para o usuário, o PIP permite comandos abreviados para transferir dados entre discos.



Estes comandos são os seguintes.

PIP A:=X.Y (cr)

Esta forma copia o arquivo X.Y do 'drive' corrente para o arquivo X.Y no 'drive' 'A'.

PIP A:=B.X.Y (cr)

Copia o arquivo X.Y do 'drive' 'B' para o arquivo X.Y no 'drive' 'A'.

Observe que os discos de origem e destino podem ser diferentes em todos os casos já vistos e que o nome do 'drive' pode ser A, B, C ou D.

Os comandos seguintes ilustram como se deve proceder para fazer cópias disco a disco total ou por um processo seletivo.

É importante ressaltar que toda a parte referente a este processo seletivo ou seja referência a arquivos está escrita no item 2.1 deste manual.

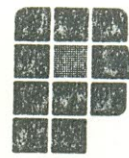
B:=\*.COM (cr)

Copia todos os arquivos, que possuem o nome secundário COM, e que se encontram no drive corrente, para o 'drive' 'B'; esses arquivos são copiados com o mesmo nome dos arquivos de origem.

A:=B:ZAP.\* (cr)

Copia todos os arquivos que possuam o nome primário ZAP do 'drive' 'B' para o 'drive' 'A'.

TÍTULO SISTEMA - 700	PÁGINA 17
SUB-TÍTULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

ZAP.COB=B (cr)

Este comando é equivalente a se escrever ZAP.COB=B ZAP COB ou seja, irá copiar o arquivo ZAP COB do 'drive' 'B' para o ZAP.COB no 'drive' corrente.

B:ZAP.COM=A: (cr)

É equivalente a se escrever B:ZAP.COM=A:ZAP.COM

B:=GAMA.BAS (cr)

É o mesmo que B:GAMA.BAS=GAMA.BAS, ou seja, copia o arquivo GAMA.BAS do 'drive' corrente para o GAMA.BAS que se encontra no 'drive' 'B'.

B:=A:GAMA.COM (cr)

É o mesmo que se escrever B:GAMA.COM=A:GAMA.COM.

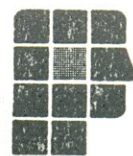
Existe ainda um comando especial do utilitário PIP que permite que um ou mais arquivos sejam copiados para a impressora, ou seja, listado ou listados; esses arquivos são impressos com as linhas numeradas

PIP PRN:=A:ZAP.COB

Listará na impressora principal do sistema o arquivo ZAP.COB que se encontra no 'drive' 'A'.

No início, a impressora dará um salto de página e iniciará a impressão no início de uma nova folha e daí em diante de 60 em 60 linhas a impressora saltará de folha.

TÍTULO SISTEMA - 700	PÁGINA 18
SUB-TÍTULO DOS-700	DATA



PIP PRN:=A:ZAP.COB,ZAP1.COB

Listará o arquivo ZAP.COB que se encontra no 'drive' 'A'. Ao encontrar fim de arquivo (CTRL-Z) iniciará a listagem do arquivo ZAP1.COB que se encontra no 'drive' corrente. A numeração das linhas será sequencial só terminado quando encontrar fim do arquivo ZAP1.COB.

Observe que se o arquivo de destino for em disco, um arquivo temporário, com o nome secundário \$\$\$, é criado. Esse nome é trocado após a operação ter sido completada normalmente.

Se a operação for abortada, ou se ocorreu um erro durante o processamento, o PIP cancela também a execução de uma rotina catalogada, que veremos mais adiante.

A operação de cópia pode ser cancelada a qualquer momento, se o operador desejar, basta pressionar qualquer tecla de console, um 'return (cr)' é suficiente.

O PIP mostrará no vídeo a mensagem 'ABORTED' que indica que a operação não foi completada normalmente.

Até agora vimos algumas das possíveis linhas de comando do utilitário PIP, que podem ser usadas no caso 2.

No caso 1, ou seja, comandos lidos diretamente do console estas linhas também são válidas, veja os comandos abaixo:

PIP (cr)

Carrega o PIP para a TPA para que possa receber comandos, o PIP imprime '\*' para avisar que está pronto.

TÍTULO	PÁGINA
SISTEMA - 700	19
SUB-TÍTULO	DATA
DOS-700	



\*PRN:=X.COB.Y.COB

Concatena os dois arquivos X e Y, que são fontes em COBOL e copiam para a unidade lógica PRN. No término da cópia outro asterisco aparecerá no vídeo, para encerrar a execução do PIP, neste caso, um simples <cr> é suficiente.

O usuário pode também especificar um ou mais parâmetros para o PIP, estes parâmetros têm que estar entre parênteses quadrados [ ] separados ou não por brancos.

Cada parâmetro causa um efeito sobre a operação de cópia e a lista de parâmetros deve ser incluída logo após a especificação do último nome de arquivo de origem. Geralmente cada parâmetro pode ser seguido por um valor decimal inteiro, que é opcional.

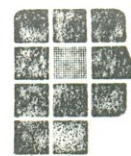
Os parâmetros válidos para o PIP são listados abaixo.

**Dn** Apaga os caracteres que ultrapassam a coluna n na transferência de dados do arquivo de origem para o de destino.  
Este parâmetro é normalmente usado para eliminar os caracteres que ultrapassam a coluna 72 de um programa COBOL.

**E** Exibe no vídeo toda a operação de transferência de dados, como elas estão sendo feitas.

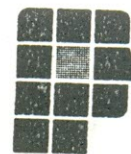
**F** Elimina todos os 'form feeds' encontrados no arquivo. O parâmetro P pode ser usado simultaneamente, para inserir novos 'form feeds'..

TITULO SISTEMA - 700	PAGINA 20
SUB-TITULO DOS-700	DATA



- L Transforma todas as letras maiúsculas em minúsculas.
- N Numera cada linha transferida do arquivo de destino, começando de 1 e incrementado de 1. Os zeros à esquerda são suprimidos e o número é seguido de dois pontos (:). Se N2 for especificado então os zeros à esquerda não serão suprimidos, e o caracter TAB é inserido após o número da linha, o 'TAB' é expandido se o parâmetro T for especificado.
- P<sub>n</sub> Inclui o caracter 'form feed' a cada n linhas (iniciado por 'form feed').  
Se n= 1 ou omitido um 'form feed' será incluído a cada 60 linhas.  
Se o parâmetro F for especificado o caracter 'form feed' é apagado em seguida é executada a função P.
- T<sub>n</sub> Coloca o caracter TAB em cada n-ésima coluna durante a transferência de caracteres do arquivo de origem, para o destino.
- U Transforma os caracteres minúsculos em maiúsculos, durante a operação de cópia.
- V Verifica se o dado foi copiado corretamente, executando uma leitura após a operação de gravação (o arquivo de destino deve estar em disco).

TÍTULO SISTEMA - 700	PAGINA 21
SUB-TÍTULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

Os seguintes comandos, com especificação de parâmetros, são validos para o PIP.

PIP X.COB=B:[V]<cr>

Copia o arquivo X.COB, do 'drive' 'B', para o drive corrente e verifica se os dados foram copiados corretamente.

PIP PRN:=X.COB[NT8V]

Copia o arquivo X.COB para impressora, número cada linha expande caracteres TAB a cada 8 colunas.

PIP PRN:=X.COB [P 50]

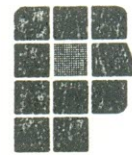
Lista o arquivo X.COB saltando de folha a cada 50 linhas.

O parâmetro NT8P60 é assumido caso não houver especificação de parâmetros.

P50 troca esse 'default'.



TÍTULO	PÁGINA
SISTEMA - 700	22
SUB-TÍTULO	DATA
DOS-700	



#### 4.3. FORMATA

O utilitário formata inicializa um disco virgem para que possa ser utilizado.

A formatação do disco é realizada por 'software' e é a seguinte:

. 35 trilhas com 40 setores de 128 'bytes' cada um.

Para carregar o programa basta digitar FORMATA e ele é carregado para o TPA para ser executado.

O programa manterá diálogo constante com o operador, para que não haja erros.

#### EXEMPLO:

A> FORMATA

PROLÓGICA PROGRAMA FORMATA VER 3.0-512 BYTES/SETOR  
SELECCIONE UNICA/DUPLA FACE (U/D) (OU RETUTN P/SAIR)

DIGITE U.

COLOQUE DISCO NÃO FORMATADO NO DRIVE B TECLE 'F' PARA  
COMEÇAR

FORMATANDO E VERIFICANDO LADO 1 TRILHA XX

Onde XX é número de trilha que está sendo formatada.

#### 4.4. GERSIS

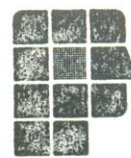
O GERSIS permite a geração do sistema operacional DOS-700 em um disco.

O utilitário GERSIS também mantém diálogo com o operador.

GERSIS (cr)

Carrega o GERSIS para o TPA

TÍTULO SISTEMA - 700	PÁGINA 23
SUB-TÍTULO DOS-700	DATA



GERSIS VERSÃO 2.2

Mensagens do GERSIS

NOME DO DRIVE FONTE  
(OU cr PARA TERMINAR)

Responda com o nome do 'drive' que contém o disco com o sistema operacional DOS-700; normalmente A.  
Causará a seguinte resposta do microcomputador.

FONTE NO X TECLE cr

FUNÇÃO COMPLETA

NOME DO DRIVE DESTINO  
(OU cr PARA TERMINAR)

Coloque um disco formatado no 'drive' digitado.

DESTINO NO X TECLE cr

Confirme digitando cr.  
Neste momento o Sistema operacional é copiado para as duas primeiras trilhas do disco no 'drive' especificado.

FUNÇÃO COMPLETA

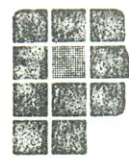
O disco está pronto para ser usado.

O nome do 'drive' destino será sempre solicitado até que um cr seja digitado no console, senão, mais que um disco poderá ser inicializado.

Após a geração do sistema ter sido completada com sucesso o novo disco contém o sistema operacional, o usuário pode copiar todos os arquivos de um disco que já existe, digitando o seguinte comando:

. PIP B:=A\*.\* [V] <cr>

TÍTULO SISTEMA - 700	PÁGINA 24
SUB-TÍTULO DOS-700	DATA



PROLOGICA  
microcomputadores

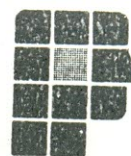
Este comando copiará todos os arquivos do disco 'A' para o disco 'B' e verifica se cada arquivo foi copiado corretamente.

O nome do arquivo que está sendo copiado aparece automaticamente no vídeo.

O GERSIS não destrói arquivos que já existem no disco, pois, os diretórios dos arquivos são colocados em setores especiais previamente selecionados pelo sistema operacional.

Essa operação não altera a capacidade de armazenamento do disco que atualmente é de 175 Kbytes.

TÍTULO SISTEMA - 700	PAGINA 25
SUB-TÍTULO DOS-700	DATA



#### 4.5. SUBMIT

O utilitário SUBMIT permite que o DOS-700 opere com processamento automático, tipo 'BATCH'.

Sintaxe:

SUBMIT nome de arquivo param 1 param 2 param 3 param n

Onde:

Nome arquivo = Nome do arquivo que contém a sequência de comandos do DOS-700 que devem ser executados.

Param n = Parâmetros externos que são passados para os comandos dentro do arquivo.

O arquivo é processado sequencialmente, comando a comando, até que terminem.

Este arquivo deve ser criado com o utilitário EDIT1, que é o editor de textos e que será descrito mais adiante.

Os parâmetros dentro do arquivo são representados pelo símbolo '\$' seguido de um número decimal inteiro.

\$1 \$2 \$3.....\$n

Onde:

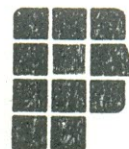
\$1 é substituído pelo primeiro parâmetro, e assim sucessivamente.

Se o número de parâmetros passados for menor ou maior declarados, a execução do SUBMIT é abortada e uma mensagem de erro será emitida no vídeo.

O utilitário SUBMIT cria um arquivo temporário de trabalho com o nome \$\$\$SUB no disco corrente.

Quando o SUBMIT termina, esse arquivo é lido pelo CCP como se fosse comandos do console.

TÍTULO	PÁGINA 26
SUB-TÍTULO	DATA



**PROLOGICA**  
microcomputadores

Se o utilitário SUBMIT for executado sobre um 'drive' diferente de 'A' os comandos não são processados, até que o disco que contém o \$\$\$SUB seja inserido no 'drive A'.

A execução dos comandos pode ser abortada a qualquer instante que o operador desejar, teclando `cr`. Neste caso o arquivo \$\$\$SUB é apagado do disco e os próximos comandos serão lidos do console.

O processamento dos comandos é também abortado se o CCP detectar um erro de sintaxe em algum comando.

O último comando dentro do arquivo SUB pode iniciar outro arquivo SUB.

Suponha que o arquivo ASMBL.SUB exista no disco e contém os seguintes comandos:

```
ASM $1
DIR $1
ERA *.BAK
PIP $2:= 1.PRN
ERA $1.PRN
```

e o comando `SUBMIT ASMBL X PRN <cr>` é digitado pelo operador.

O programa SUBMIT lê o arquivo ASMBL.SUB substituindo todos \$1 por X e todos \$2 por PRN resultando um arquivo \$\$\$SUB contendo os comandos:

```
ASM X
DIR X.*
ERA *.BAK
PIP PRN:=X.PRN
ERA X.PRN
```

que são executados, na sequência em que estão escritos, pelo CCP.

O utilitário SUBMIT pode acessar um arquivo SUB que se encontra em um 'drive' diferente de 'A', para isto basta preceder o nome do arquivo com o nome do 'drive' seguido de dois pontos (:).

TÍTULO	PAGINA
SISTEMA - 700	27
SUB-TÍTULO	DATA
DOS-700	



É possível também criar um arquivo \$\$\$SUB num 'drive' diferente de 'A', que será executado mais tarde, quando colocado no 'drive A'.

#### 4.5.1. XSUB

O XSUB permite incluir linhas em programas, ou seja, se você de  
seja executar uma série de programas em uma determinada ordem  
o SUBMIT resolveria o problema, porém, se um desses programas  
solicitar dados do console esses dados não serão aceitos sem  
a utilização do XSUB.

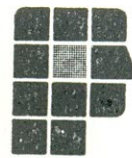
O XSUB deve ser a primeira instrução de seu procedimento catalo  
gado, para avisar ao DOS-700 que haverá entrada de dados atra  
vés do console.

Se dentro do seu procedimento catalogado houver uma chamada pa  
ra um segundo e este também solicitar entrada de dados via con  
sole, este não deve conter a instrução XSUB, pois uma vez ativo  
o XSUB permanece na memória enquanto não for dado um 'reset' ou  
CTRL-C.

#### EXEMPLO:

. XSUB	Executa função XSUB
. DIR\$1.*	Exibe arquivos especificado
. EDITI \$1.\$2	Usa o editor no arquivo especifi cado.
. #A	Carrega todo o programa para memó ria.
. B	Move o CP para o início da primei ra.
. i	Insere linhas
. UTILIZAÇÃO DO XSUB <cr>	Linha lida do console
. CTRL-Z	
. E	Fim do editor

TITULO SISTEMA - 700	PÁGINA 28
SUB-TITULO DOS-700	DATA



**PROLOGICA**  
microcomputadores

A execução deste procedimento seria feita pelo seguinte comando:

```
A> SUBMIT EXEC EX1 COB
```

Todos os \$1 dentro de EXEC.SUB seriam substituídos por EX1 e todos os \$2 por COB, então teríamos:

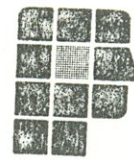
- . 1 - Execução da função XSUB
- . 2 - DIR EX1.\*
- . 3 - EDIT1 EX1.COB
- . 4 - #A
- . 5 - B
- . 6 - I
- . 7 - Entrada de dados via console
- . 8 - E

Ao término dos comandos o XSUB permanecerá ativo, exibindo no vídeo a mensagem:

(XSUB ACTIVE)

que só será desativada se for dado um 'reset' ou CTRL-C.

TITULO	PAGINA
SISTEMA - 700	29
SUB-TITULO	DATA
DOS-700	



**PROLOGICA**  
microcomputadores

#### 4.6 EDIT1

O EDIT1 é um editor de textos para o DOS-700, e é usado para criar ou alterar arquivos fontes.

O EDIT1 é carregado para memória digitando

$$\text{EDIT1} \left\{ \begin{array}{l} \langle \text{nome arquivo} \rangle \\ \langle \text{nome arquivo.tipo} \rangle \end{array} \right\}$$

Normalmente o EDIT1 lê segmentos de um arquivo fonte para memória central. O arquivo é manipulado pelo operador, posteriormente é gravado no disco. Se o arquivo fonte ainda não existir, o EDIT1 o criará com o nome dado.

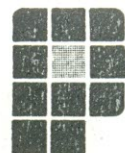
O EDIT1 opera sobre um arquivo fonte, como mostra a figura 1, e passa todo o texto para memória, onde ele pode ser listado ou alterado.

O texto editado pode ser gravado num arquivo temporário através de um comando se o operador assim desejar.

Após o término da edição do programa, as linhas digitadas são gravadas, do arquivo temporário, seguidas pelo restante do texto, no arquivo fonte.

O nome do arquivo original X.Y é trocado para X.BAK, dessa forma o arquivo original não é modificado e pode ser chamado se necessário. O nome arquivo temporário é então trocado de X.\$\$\$ para X.Y que resulta no arquivo editado.

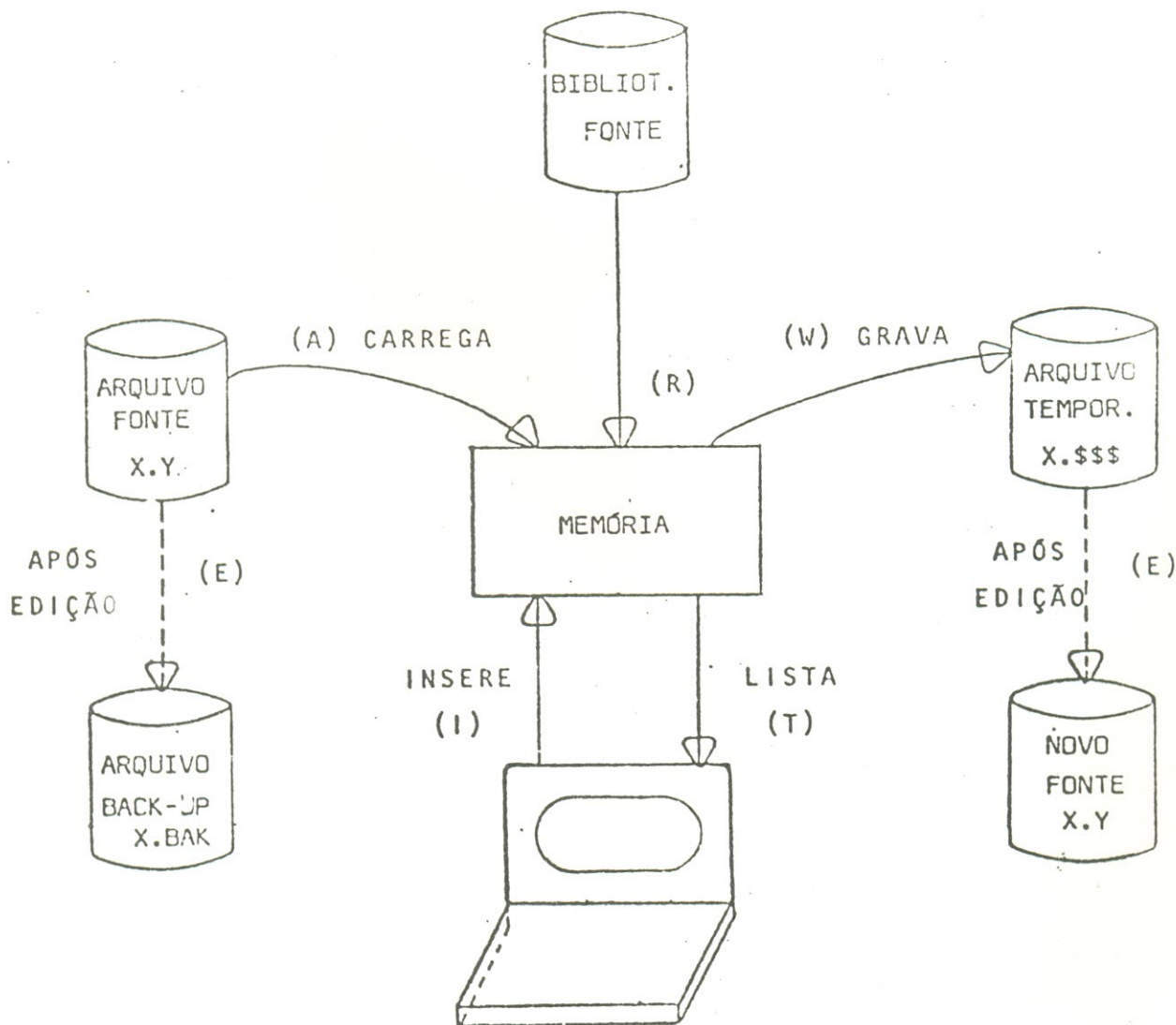




4.6.1 FUNÇÕES DO EDITOR

Seja  $n$  um número inteiro que varia de 0 a 65535, os seguintes comandos transferem linhas do texto do arquivo fonte para o temporário através da memória.

FIGURA 1 - OPERAÇÃO DO EDITOR



TÍTULO SISTEMA - 700	PÁGINA 31
SUB-TÍTULO DOS-700	DATA



#### 4.6.2 ORGANIZAÇÃO DA MEMÓRIA

A memória pode ser considerada como uma sequência de linhas fontes que são carregadas do arquivo fonte, com o comando A.

Na memória existe um 'character pointer' (imaginário),  $\square_{CP}$ , que se move de acordo com os comandos digitados.

A memória é representada, como mostra a figura 3, onde os hífens representam os caracteres da linha fonte, de tamanho indefinido, terminadas por <cr> (carriage return) e lf (line feed) e o  $\square_{CP}$  representa o imaginário 'character pointer'. A linha corrente é a linha fonte que contém o  $\square_{CP}$ .

#### 4.6.3 OPERAÇÃO DA MEMÓRIA

Após o EDIT1 ser carregado para TPA a memória encontra-se limpa para receber as linhas fontes através do comando A ou através do comando de inserção de linhas.

I <cr>

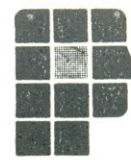


FIGURA 2. ORGANIZAÇÃO DA MEMÓRIA

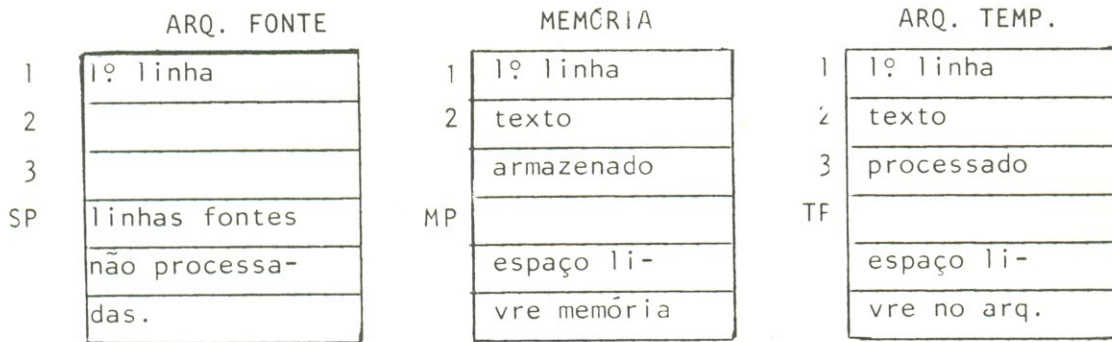
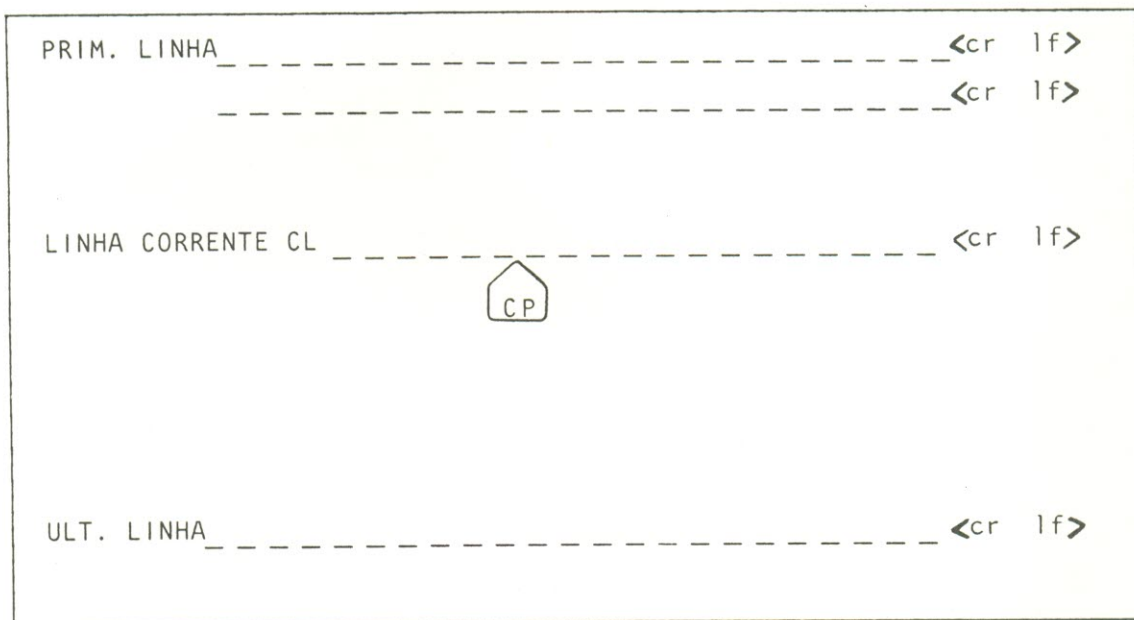
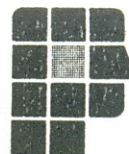


FIGURA 3. ORGANIZAÇÃO LÓGICA DA MEMÓRIA



nA <cr> Tráz para memória as próximas n linhas fontes não processadas do arquivo fonte até SP ou até o fim da memória MP.  
Incrementa SP e MP de n.

TÍTULO SISTEMA - 700	PAGINA 33
SUB-TÍTULO DOS-700	DATA

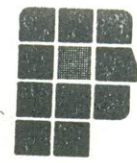


- nW <cr> Grava as primeiras n linhas da memória no espaço livre do arquivo temporário.  
Joga para o topo da memória o restante das linhas a partir linha n+1  
Incrementa TP de n
- E <cr> Fim do editor. Copia todo o texto armazenado na memória para o arquivo temporário.  
Atualiza os nomes dos arquivos, como foi descrito previamente.
- H <cr> O arquivo temporário fica sendo o novo arquivo fonte, a memória é limpa e um novo arquivo temporário é criado.  
Este comando é equivalente a dar o comando E e logo em seguida chamar o EDIT1, usando o arquivo X.Y.
- O <cr> Retorna ao arquivo original. A memória é limpa e o arquivo temporário é apagado.  
Não houve alterações no seu arquivo fonte.
- Q <cr> Cancela a execução do EDIT1 não fazendo nenhuma alteração no arquivo fonte.

Alguns casos especiais têm que ser considerados.

Se o inteiro n for omitido em um comando onde é permitido, o editor assume 1, logo o comando A e o comando W trazem uma linha para a memória ou gravam apenas uma linha, respectivamente.

TÍTULO SISTEMA - 700	PÁGINA 34
SUB-TÍTULO DOS-700	DATA



Cada linha digitada tem que ser terminada por um <cr lf> (o lf é colocado automaticamente), até que um CTRL-Z seja digitado. O que significa fim de inserção.

O  fica posicionado após o último caracter digitado.

A sequência

```

I <cr>
VAMOS <cr>
PROGRAMAR <cr>

```

fica na memória


```



VAMOS <cr lf>
PROGRAMAR <cr lf>

```



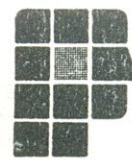
Os comandos mostrados abaixo, precedidos de mais (+) ou menos (-) ou n indicam que um valor inteiro, opcional, pode ser digitado.

+B<cr> Move o  para o início da memória ou para o fim se o sinal '-' for especificado. Se não especificar sinal, é assumido '+'.  
Se não especificar sinal, é assumido '+'.

+ nD<cr> Apaga n caracteres a partir da posição do  se '+' for digitado, e para trás do  se '-' for digitado.

+nK<cr> Cancela + n linhas a partir da posição do .

TÍTULO SISTEMA - 700	PÁGINA 35
SUB-TÍTULO DOS-700	DATA



+ nL <cr> Move o CP para início da linha corrente, depois soma n com o número da linha corrente, e move o CP para o início da linha resultado.  
Se o valor especificado for maior ou menor que o número de linhas fontes na memória, o CP fica posicionado no início da memória ou no início da última linha armazenada na memória.

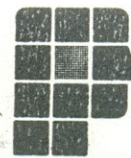
+nT <cr> Se  $n=0$  exibe o conteúdo da linha anterior.  
Se  $n=1$  exibe o conteúdo da linha corrente.  
Se  $n>1$  exibe o conteúdo da linha corrente até a linha  $n-1$ , se '+' for especificado.  
Se  $n=1$  e '-' for especificado, exibe o conteúdo das n linhas previamente digitadas.

+ C <cr> Move o CP n caracteres para direita, ou esquerda, se '-' especificado.

Suponha que o conteúdo da memória seja:

SISTEMA <cr lf>  
700 <cr lf>  
PROLÓGICA <cr lf>

COMANDO	EFEITO	RESULTADO DA MEMÓRIA
B2T <cr>	Move o <u>CP</u> para o início do 'buffer' e exibe 2 linhas 'SISTEMA 700'	<u>CP</u> SISTEMA <cr lf> 700 <cr lf> PROLÓGICA <cr lf>



COMANDO	EFEITO	RESULTADO DA MEMÓRIA
5CØT <cr>	Move o <input type="text" value="CP"/> 5 caracteres e imprime o início da linha 'SISTE'	SISTE <input type="text" value="CP"/> MA <cr lf> 700 <cr lf> PROLÓGICA <cr lf>
2L-T <cr>	Move o <input type="text" value="CP"/> 2 linhas para baixo e lista a anterior. '700'	SISTEMA <cr lf> 700 <cr lf> <input type="text" value="CP"/> PROLÓGICA <cr lf>
k<cr> DA<cr> CRTL-Z	Inserir linhas no texto, a linha corrente assim como restante do texto é deslocado para baixo.	SISTEMA <cr lf> 700 <cr lf> DA <cr lf> <input type="text" value="CP"/> PROLÓGICA <cr lf>
<cr>	Move <input type="text" value="CP"/> para o início da próxima linha. 'PROLÓGICA'	SISTEMA <cr lf> 700 <cr lf> DA <cr lf> PROLÓGICA <cr lf>

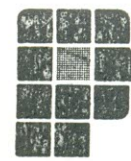
#### 4.6.4 PESQUISA E ALTERAÇÃO DE TEXTO

O EDITI possui comandos que permitem ao operador localizar uma cadeia de caracteres no texto que se encontra na memória.

nFc1, c2, c3.....ck <cr>

Onde c1 a ck representam a cadeia de caracteres a ser, pesquisada, seguido de <cr>

O EDITI inicia a pesquisa a partir da posição atual do



e verifica todos os k caracteres.

O caracter ou cadeia de caracteres é pesquisada n vezes, ou seja, se  $n=2$  irá para segunda cadeia de caracteres, e se encontrar, o CP é movido para o final dessa cadeia ou após o caracter ck.

Se a cadeia de caracteres não for encontrada o CP permanece na sua posição atual.

Os comandos seguintes ilustram a utilização do comando F.

COMANDO	EFEITO	RESULTADO
BT <cr>	Move o CP para o início do texto e lista o conteúdo da memória.	<span style="border: 1px solid black; padding: 2px;">CP</span> SISTEMA <cr lf> 700 <cr lf> DA <cr lf> PROLÓGICA <cr lf>
FA <cr>	Pesquisa o 'A' e posiciona o <span style="border: 1px solid black; padding: 2px;">CP</span> logo após a primeira ocorrência.	SISTEMA <span style="border: 1px solid black; padding: 2px;">CP</span> <cr lf> 700 <cr lf> DA <cr lf> PROLÓGICA <cr lf>
FP↑ZOTT <cr>	Pesquisa o próximo P e lista o conteúdo restante da linha corrente.	SISTEMA <cr lf> 700 <cr lf> DA <cr lf> PROLÓGICA <cr lf>
	'PROLÓGICA'	<span style="border: 1px solid black; padding: 2px;">CP</span>

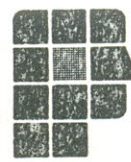
O formato abreviado do comando de inserção também é permitido e pode ser usado em conjunto com o comando F para fazer simples troca de texto.

lclc2c3 .....cn<cr>ou↑Z.

Onde cl a cn são os caracteres a inserir.



TÍTULO SISTEMA - 700	PÁGINA 38
SUB-TÍTULO DOS-700	DATA



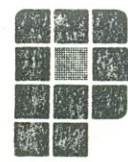
Se o 'string' a inserir for terminado por um ↑Z, os caracteres cl a cn são inseridos após a posição do CP e o CP é movido para a posição imediatamente após o caracter cn.

A ação é a mesma se o comando for seguido por um cr exceto que um <cr lf> é automaticamente inserido no texto após o caracter cn.

Considere os comandos abaixo como exemplo de utilização do comando F e l

COMANDO	EFEITO	RESULTADO
BIESSE E 0 ↑Z <cr>	Inseres os caracteres 'ESSE E 0' no início do texto na memória.	ESSE E 0 CP SISTEMA cr lf> 700 <cr lf> DA <cr lf> PROLÓGICA <cr lf>
2FØ ↑ZI DA <cr>	Pesquisa a segunda ocorrência do 'Ø' e insere 'DA'	ESSE E 0 SISTEMA <cr lf> 700 DA <cr lf> DA <cr lf> PROLÓGICA.
1LK <cr>	Avança uma linha e a cancela.	ESSE E 0 SISTEMA <cr lf> 700 DA <cr lf> PROLÓGICA <cr lf>
BFO ↑ZI <cr>	Mov <sup>e</sup> CP para o início do texto, pesquisa o caracter 0, insere <cr lf>	ESSE E 0 <cr lf> SISTEMA <cr lf> 700 DA <cr lf> PROLÓGICA <cr lf>

TÍTULO SISTEMA - 700	PÁGINA 39
SUB-TÍTULO DOS-700	DATA



O próximo comando é uma combinação dos comandos F e I para executar simples substituições de caracteres.

$$nSc1, \dots ck \ z-KD1d1d2 \dots dm \begin{cases} cr \\ \uparrow z \end{cases}$$

O EDITI pesquisa a memória a partir da posição do  $\square$ CP e substitui a primeira cadeia de caracteres pela segunda e se  $\underline{n}$  for especificado executa a operação  $\underline{n}$  vezes.

Para maior conveniência do usuário um comando similar ao F foi implementado no EDITI.

$$nNc1c2 \dots ck \begin{cases} cr \\ \uparrow z \end{cases}$$

Que pesquisa todas as linhas na memória para encontrar a  $n$ -ésima ocorrência da cadeia de caracteres  $c1 \dots ck$ . Caso a cadeia não for encontrada um comando  $\#W$  automático é executado.  $\#W$  grava as primeiras 65535 linhas da memória.

O último comando de alteração de texto do editor é o seguinte:

$$nJc1c2 \dots ck \ zd1d2 \dots dm \ zele2e3 \dots eq \begin{cases} cr \\ \uparrow z \end{cases}$$

Este comando pesquisa a cadeia  $c1 \dots ck$ , em seguida insere os caracteres  $d1 \dots dm$ , e apaga a linha exceto os caracteres  $e1 \dots eq$ .

TÍTULO	PÁGINA
SISTEMA - 700	40
SUB-TÍTULO	DATA
DOS-700	



**PROLOGICA**  
microcomputadores

#### 4.6.5. BIBLIOTECAS FONTES

O EDIT1 permite a inclusão de bibliotecas fontes durante o processo de edição através do comando R.

$$Rf1f2\dots f8 \left\{ \begin{array}{l} \uparrow z \\ cr \end{array} \right.$$

Onde f1...f8 é o nome da biblioteca fonte do disco, é assumido o nome de um arquivo com sufixo 'LIB'

O EDIT1 lê essa biblioteca e coloca os caracteres a partir da posição do  $\hat{CP}$ , esse comando é similar ao I.

Exemplo:

RMACRO <cr>

O EDIT1 procura no disco corrente o arquivo MACRO.LIB e o insere a partir da linha corrente.

#### 4.6.6. CONDIÇÕES DE ERRO DO EDIT1

Quando ocorrer alguma condição de erro no EDIT1, ele imprime uma mensagem de erro, seguida de um dos caracteres abaixo, e logo em seguida o último caracter lido antes do erro ocorrer.

- ? Comando Inválido
- > Memória cheia, use um dos comandos D,K,N,S ou W para remover ou gravar linhas para liberar espaço na memória.
- # Não é possível a execução do comando o número de vezes especificado.
- 0 Não pode abrir o arquivo 'LIB' durante a execução do comando R.

TÍTULO SISTEMA - 700	PÁGINA 41
SUB-TÍTULO DOS-700	DATA



A cada operação de E/S em disco é realizada uma verificação para que não ocorra erros de E/S. Caso ocorrer algum erro, o DOS-700 exibe no vídeo a mensagem:

PERM ERR DISK d

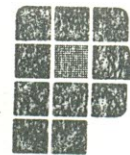
onde o d é o drive corrente ou selecionado..

Se ocorrer esse erro o usuário pode ignorá-lo digitando qualquer tecla e logo em seguida verificar se os dados, gravados estão ok ou simplesmente resetar o sistema e chamar o arquivo 'back up',

X.BAK

que conterà as informações do arquivo original antes das alterações.

TÍTULO SISTEMA - 700	PÁGINA 42
SUB-TÍTULO DOS-700	DATA



## 5. MENSAGENS DE ERRO DO BDOS

Há três situações de erro que o BDOS interrompe o processamento.

Quando uma dessas situações ocorre o BDOS imprime uma mensagem:

BDOS ERR ON x: erro

Onde x é o nome do 'drive' e erro é o nome das mensagens abaixo:

BAD SECTOR

SELECT

READ ONLY

O erro 'BAD SECTOR' indica que o controlador do disco detectou uma condição de erro de leitura ou gravação do disco. Essa condição ocorre normalmente por um eventual mau funcionamento do controlador de disco ou muito dificilmente, erro físico do disco.

O erro 'SELECT' ocorre quando o nome do 'drive' selecionado não se encontra no intervalo A a D, ou quando o 'drive' selecionado não está presente.

O erro 'READ ONLY' ocorre quando se tenta fazer gravações em disco que for designado somente para leitura, ou quando houver troca de disco sem que o sistema operacional tenha sido avisado para efetuar essa troca.

Os DOS-700 permite que um ou mais discos sejam trocados, criando dessa maneira, arquivos multi-volumes. Há apenas uma condição para efetuar a troca; que o usuário chame, através do seu programa, o utilitário ASØ1, que avisa ao DOS-700 que será efetuada uma troca de discos.

