Sistema Operacional em Disco

TK 2000 Color

INTERFACE DO CONTROLADOR DE DISCO

INDICE:

I - INTRODUÇAO 3

I.1 - O Disquete 3
I.2 - A Organização do Disquete 3
I.3 - O TKDOS 3.3 3
I.4 - Instalação da interface de disco 4
I.5 - Cuidados com os Drives e disquetes 6

II - OPERAÇAO INICIAL 7

II.1 - Introduzindo o TKDOS 3.3 7
II.2 - "BOOT" do TKDOS 3.3 7
II.3 - Inicializando novos disquetes 7
II.4 - Gravando e lendo programas pelo TKDOS 8
II.5 - O comando CATALOG 10
II.5.1 - Nomes de arquivos 10
II.6 - Outros comandos do TKDOS 11
II.6.1 - DELETE 11
II.6.2 - LOCK 11
II.6.3 - UNLOCK 12
II.6.4 - RENAME 12
II.6.5 - VERIFY 12
II.7 - Utilização de comandos do TKDOS dentro de programas 12

III - ARQUIVOS SEQUENCIAIS 15

III.1 - Ativação de arquivos sequenciais 15
III.2 - Desativação de arquivos sequenciais 15
III.3 - Gravando arquivos sequenciais 16
III.4 - Lendo arquivos sequenciais 17
III.5 - Adicionando dados em arquivos sequenciais 19
III.5.1 - O comando APPEND 19
III.5.2 - O comando POSITION 19

IV - ARQUIVOS DE ACESSO RANDOMICO 21

IV.1 - Ativação de arquivos de acesso randômico 21
IV.2 - Lendo e escrevendo em arquivos de acesso randômico 21

V - COMANDOS AUXILIARES DO DOS 24

V.1 - O comando EXEC 24
V.2 - Comando MAXFILES 25
V.3 - Comandos MON e NOMON 25
V.4 - Comando TRACE 26

VI - ARQUIVOS EM LINGUAGEM DE MAQUINA 27

VI.1 - BSAVE 27
VI.2 - BLOAD 27
VI.3 - BRUN 27

APENDICE A - MENSAGENS DO TKDOS 29
APENDICE B - SUMARIO DOS PROCEDIMENTOS DO TKDOS 33
APENDICE C - DESCRIÇAO DOS PROGRAMAS UTILITARIOS 39

CAPITULO I - INTRODUÇAO

O Sistema Controlador de Disco possibilita o armazenamento e acesso a informações e programas numa velocidade superior aos mètodos convencionais, ampliando assim a performance do sistema.

I.1 - O DISQUETE

Com a necessidade da criação de um sistema de armazenamento de dados preciso e ràpido e tambèm graças aos sucessos contínuos da tecnologia, chegou-se a criação do disco flexível. Este disco è revestido de material magnètico e requer o uso de unidades especialmente desenvolvidas para acessar informações nele contidas, os disk drives ou unidades de discos.

I.2 - A ORGANIZAÇAO DO DISQUETE

Para se tornar possível o armazenamento de informações no disco è necessàrio que, antes de operà-lo, haja uma organização do disco. Desta forma o computador, ao acessar o disquete, para leitura ou para gravação de informações, pode se referenciar dentro do disco da mesma forma que num armàrio com divisões, prateleiras etc.

Esta organização è alcançada com a formatação do disco, que define o disquete com 35 trilhas concêntricas numeradas de Ø a 34 a partir da borda externa em direção ao centro, cada uma delas dividida em 16 setores numerados de Ø a 15, conforme mostra a figura 1.

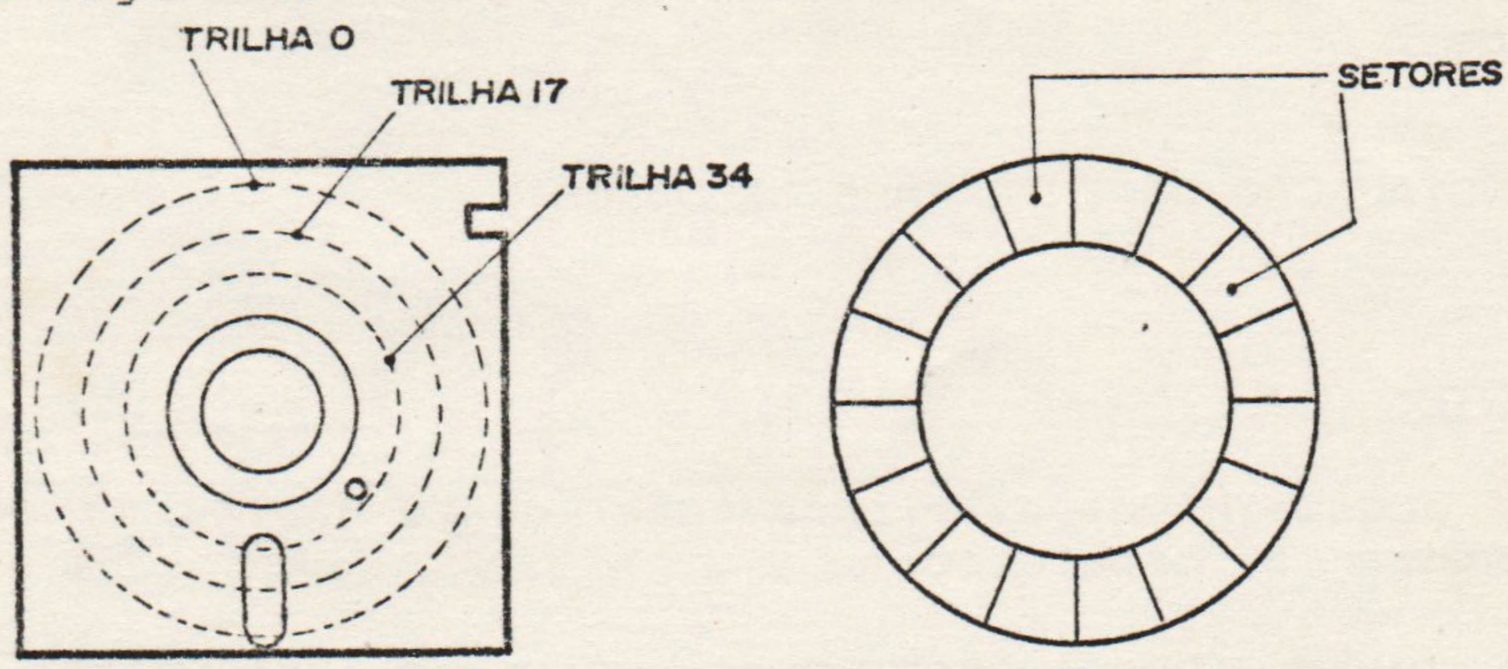


Figura 1

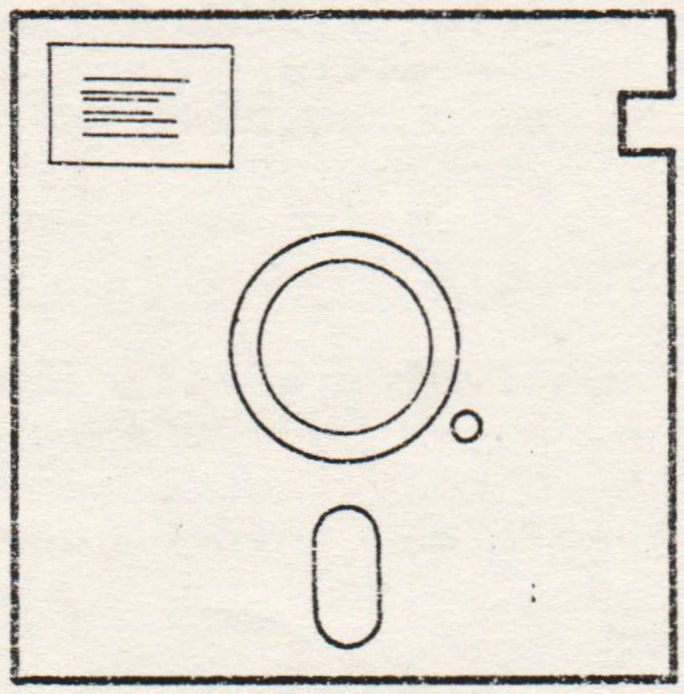
I.3 - O TKDOS 3.3

Acompanhando a interface de disco, è fornecido um disquete denominado MESTRE. Este disquete contèm programas especiais que permitem ao computador usar o disk drive, alèm de alguns programas utilitàrios.

O programa central, o mais importante no disco mestre,

è denominado TKDOS 3.3 (DOS equivale a DISK OPERATING SYSTEM). O TKDOS è o programa que formaliza as interações entre o computador e o disk drive. Uma vez introduzido na memòria, ele habilitarà uma sèrie de novos comandos, que se constituirào em ferramentas importantes no manuseio dos disquetes.

Na borda do disquete MESTRE existe um selo de proteção. ESTE SELO NÃO DEVERA SER RETIRADO, pois cobre uma cavidade especial que, quando tampada, impede que se grave no disquete, protegendo-o assim contra danos. Caso seja necessàrio gravar, è aconselhàvel adquirir outro disquete e seguir as instruções contidas neste manual.



selo de proteção

Figura 2

I.4 - INSTALAÇAO DA INTERFACE DE DISCO

IMPORTANTE:

1) E aconselhàvel a leitura total deste capítulo antes de iniciar a instalação.

2) E muito importante assegurar que o computador esteja desligado, desconectando-o da rede elètrica.

O material necessàrio para a instalação da interface de disco è:

a. O disk drive, com o respectivo cabo conector.
b. A interface de disco do TK-2000
c. O disquete MESTRE
d. Este manual

Inicialmente ligue o cabo do disk drive á interface conforme a figura 3.

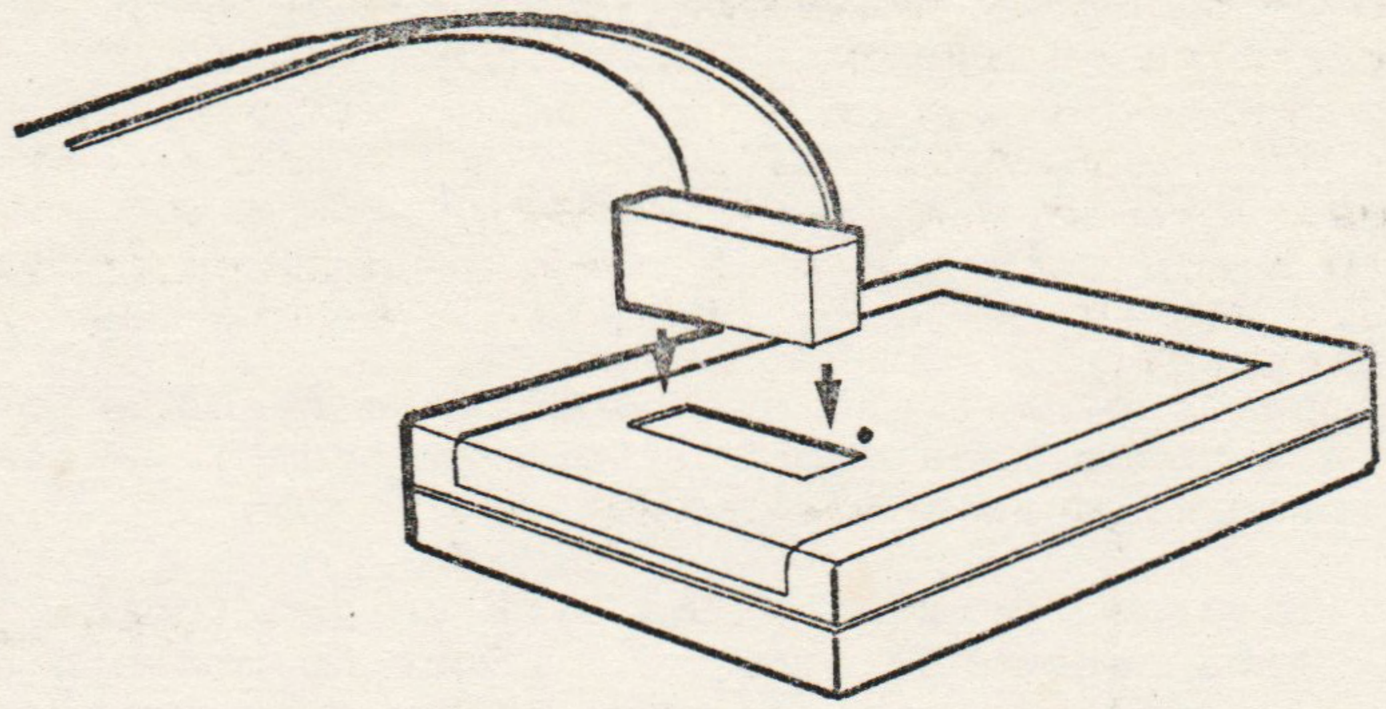


fig.3

Ligue, então a interface de disco ao computador conforme a figura 4.

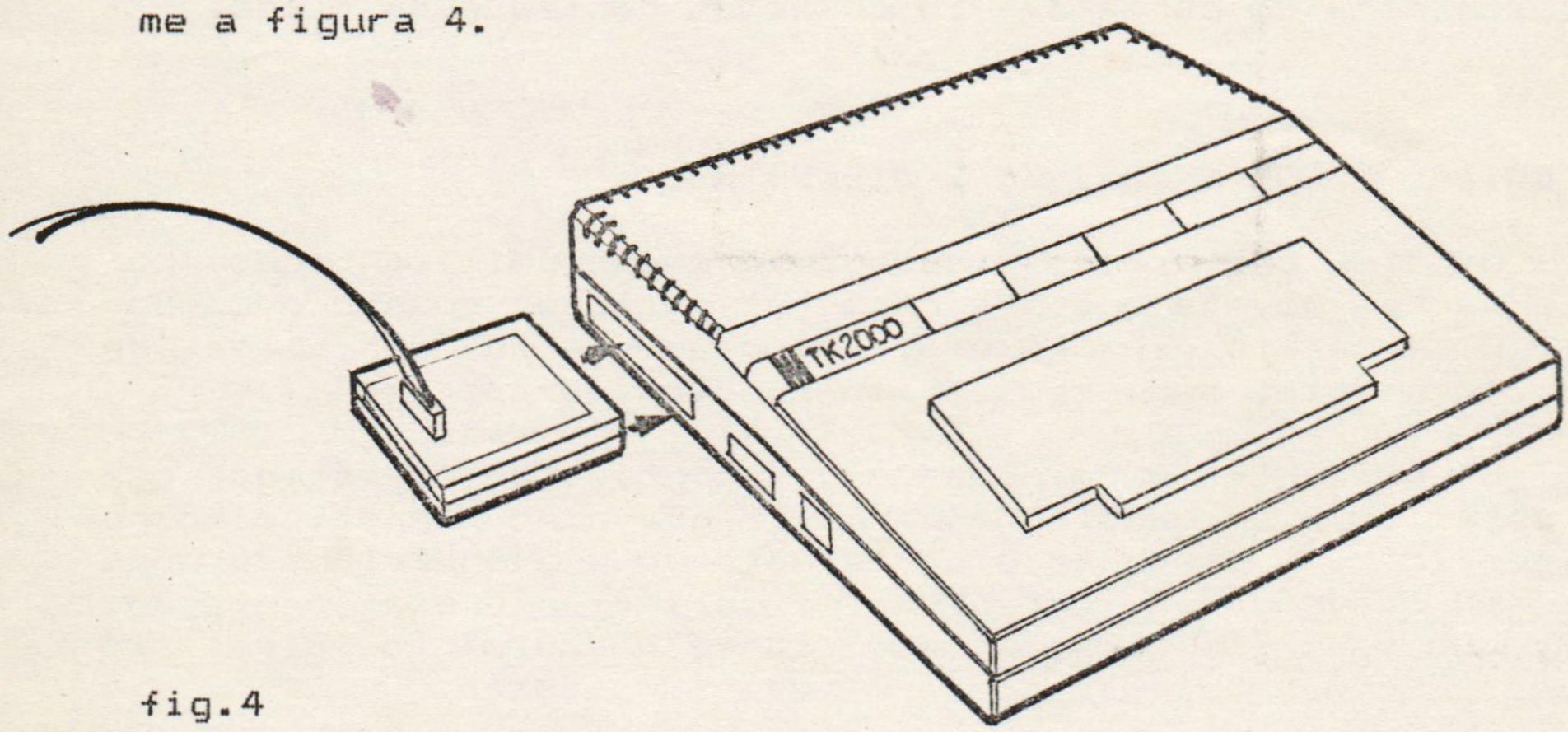


fig.4

A seguir, levante cuidadosamente a tampa do compartimento do disk drive e introduza o disco MESTRE no seu interior (verificando que, o lado da etiqueta deve estar para cima).

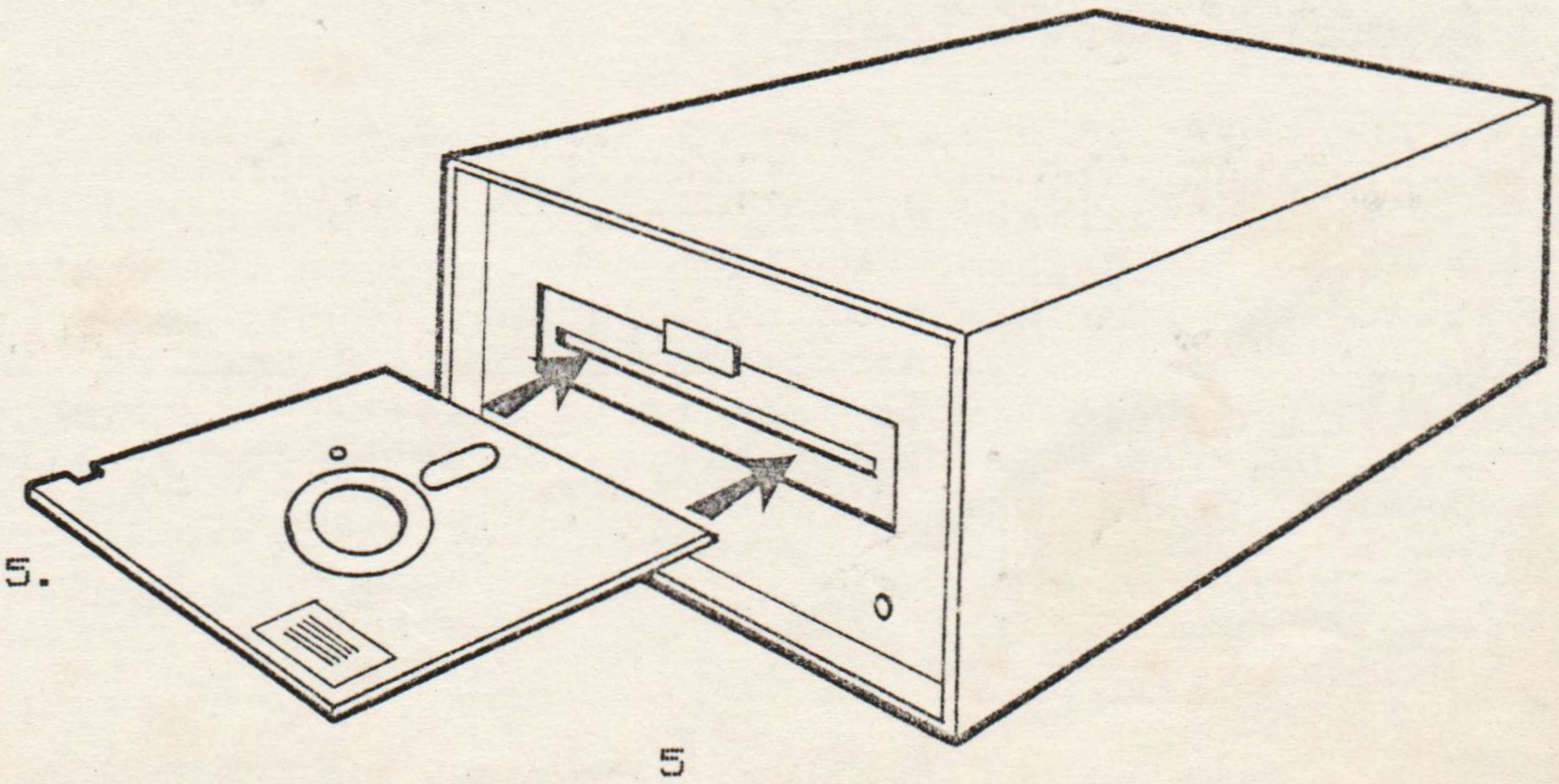


Figura 5.

Após verificar todas as conecções, ligue o computador e note que a luz vermelha existente no disk drive acenderà por uns 15 segundos quando então a tela de seu TK-2000 apresentarà:

TKDOS 3.3 DISCO MESTRE

A operação de ligar o computador e carregar o TKDOS 3.3 na memòria è denominada "boot" (leia-se "but"). Este assunto serà explicado mais adiante.

A lâmpada existente no disk drive indica quando o computador està acessando o disco. Logo, no momento em que ela està acesa, informações estão sendo lidas ou escritas no drive; de maneira semelhante a um gravador cassete. Assim sendo, enquanto esta luz estiver acesa não se deve abrir a porta do drive, sob pena de perda das informações contidas no disco.

I.5 - CUIDADOS COM OS DRIVES E DISQUETES.

O disk drive è um dispositivo mecânico, contendo motores e partes mòveis que o tornam tão delicado quanto o computador. O manuseio inadequado, como quedas ou derrubada de objetos sobre ele, pode causar mau funcionamento.

O disquete è um pequeno disco plàstico revestido com òxido metàlico, de maneira a permitir gravação magnètica sobre sua superfície. O disquete è permanentemente acondicionado numa embalagem quadrada de cor preta, que lhe oferece proteção, ajudando a mantê-lo limpo e facilitando o movimento giratòrio do disco. Esta embalagem nunca deverà ser aberta.

Deve-se evitar tocar na parte exposta do disquete, pois mesmo a impressão digital poderà apagar, ou atè destruir a superfície do disquete. Assim, mantenha sempre seus disquetes acondicionados em suas embalagens originais, prolongando a vida de seus dados e programas.

Muito embora sendo o disquete às vezes chamado de "floppy-disk" (floppy = flexível), è completamente desaconselhàvel dobrà-los, pois certamente haverà danos ao disco. Deve-se evitar tambèm, variações bruscas de temperatura, respeitando seu limite mínimo de 10 graus, e màximo de 50 graus. Deve ser mantido em ambientes secos, e livres de impurezas. Outra recomendação muito importante è evitar escrever sobre os mesmos. As etiquetas de identificação devem ser atualizadas com canetas de ponta macia.

CAPITULO II - OPERAÇAO INICIAL

II.1. INTRODUZINDO O TKDOS 3.3

Como já foi explicado no capítulo anterior, o ato de introduzir o TKDOS, ou qualquer sistema operacional, no computador, é denominado "boot". Este processo nada mais é do que carregar numa área pré-determinada da memória do TK-2000, o sistema operacional, que permitirá a utilização de novos comandos de controle para operação com disquetes.

O TKDOS será carregado numa área de memória que inclui a região destinada á segunda página de vídeo (MP). Sendo assim, uma vez com o TKDOS em operação esta área não deverá ser usada.

II.2. "BOOT" DO TKDOS

Para efetuar o "boot", desligue o computador, instale a interface de disco (conforme explicado no capítulo anterior), insira o disquete MESTRE e feche cuidadosamente o drive. Ligue o computador. A luz indicadora "EM USO" de seu drive acenderá e o TKDOS será carregado. Caso o computador já esteja ligado e se queira carregar o TKDOS, tecle o comando:

```
>DSK          (RETURN)
```

II.3. - INICIALIZANDO NOVOS DISQUETES

Para poder-se utilizar um novo disquete é necessário que ele seja inicializado. Apresentaremos abaixo a sequência de inicialização de um disquete, com um programa introdutório, que poderá ser alterado de acordo com a necessidade do usuário:

1. Carregue o TKDOS no computador como já foi exemplificado.

2. Retire o disquete MESTRE, e introduza o disquete a ser inicializado

3. Digite NEW (RETURN), entrando em seguida o programa introdutório abaixo:

```
>5 REM INICIALIZACAO DO PRIMEIRO DISQUETE
>10 HOME : REM LIMPA A TELA
>20 PRINT "DISQUETE NUMERO 1"
>30 PRINT "OPERADOR: (escreva seu nome)"
>40 END
```

O propósito do programa acima é identificar rapidamente o disquete após ser introduzido.

4. A seguir escreva:

```
>INIT HELLO
```

Após digitar RETURN, o disk drive irà funcionar por cerca de dois minutos, para formatar o disquete e gravar o sistema operacional. Neste comando também pode ser utilizado o parâmetro volume. Isto que dizer que poderemos acrescentar no disco uma identificação que, em caso de necessidade, poderà diferencià-lo de outros discos com arquivos semelhantes. Este parâmetro poderà variar de 1 a 254.

Por exemplo, poder-se-ia identificar o seu novo disco desta maneira:

```
>INIT HELLO,VØ1
```

Assim, toda vez que for pedido o diretòrio do disco, o seu nûmero de identificação constarà no topo da lista. Caso este parâmetro não seja incluido no comando INIT o volume serà assumido como "254".

5. Finalmente para testar o novo disquete, existem duas alternativas:

a) Desligar e ligar o computador com o disquete dentro do drive; ou,

b) Digitar DSK e em seguida RETURN.

A seguir aparecerà, no alto da tela, a mensagem:

```
DISQUETE NUMERO 1
OPERADOR:  "seu nome"
```

II.4. GRAVANDO E LENDO PROGRAMAS PELO TKDOS

Chame o TKDOS utilizando:

```
>DSK          (RETURN)
```

em seguida digite

```
>NEW          (RETURN)
```

este comando apagarà então o programa HELLO que deverà estar na memòria. Escreva o seguinte programa:

```
>5  REM PROGRAMA CONTADOR
>1Ø  FOR I  =  Ø  TO  9
>2Ø  PRINT I
>3Ø  NEXT I
>4Ø  END
```

Ao executar o programa, o resultado será este:

```
>RUN
Ø
1
2
3
4
5
6
7
8
9
```

Para guardar este programa no disquete (que será chamado de CONTADOR), basta digitar:

>SAVE CONTADOR

Após pressionar RETURN, o DISK DRIVE irá funcionar por alguns segundos e, quando a luz de "EM USO" apagar e o cursor aparecer novamente, a operação estará terminada. Caso se digite:

SAVEA

ou

SAVET nome

o programa será transmitido à saída de cassete, conforme consta no Manual de Operação do TK-2ØØØ.

Para carregar o programa novamente do disquete para o computador, bastará seguir as instruções abaixo:

1. NEW (RETURN)

Para apagar o programa original que ainda consta na memória.

2. LOAD CONTADOR (RETURN)

O DISK DRIVE irá funcionar por alguns segundos. Assim que ele parar, liste o programa (comando LIST). O seu programa CONTADOR, estará novamente na memória do computador. Para executá-lo, bastará digitar:

>RUN (RETURN)

o resultado já apresentado se repetirá.

OBS: Caso, no momento de carregar o programa, fosse digitado um nome que não consta do arquivo do disquete, o computador

emitiria uma mensagem de erro:

NAO HA ARQUIVO

assim sendo, deve-se conferir o nome do programa, e digitar o comando LOAD, com o nome correto.

Outro modo de executar o programa CONTADOR è, apòs ter apagado o programa da memòria (comando NEW), digitar:

>RUN CONTADOR (RETURN)

o computador se encarregarà de procurar o programa no disquete, carregà-lo e, em seguida, executà-lo. Este modo è mais direto que o anterior. Verifique os dois mètodos para sentir a diferença.

II.5. O comando CATALOG

O comando CATALOG lista na tela os nomes dos arquivos existentes no disquete, tambèm chamado diretòrio.

Para cada arquivo do disquete, o CATALOG lista a classe, a quantidade de setores que ocupa no disco e tambèm se o arquivo està travado ou nào. Ao digitar:

>CATALOG (RETURN)

a tela do TK-2000 apresentarà:

```
  VOLUME   254
A 001  HELLO
A 001  CONTADOR
>
```

o nûmero 254 è o parâmetro volume, que no caso nào foi utilizado. A letra "A" na coluna da esquerda indica o tipo de programa que foi armazenado. No caso, indica que se trata de um programa em BASIC; caso aparecesse a letra "B" indicaria que se trata de um arquivo binàrio.

Caso o disco possua mais de 18 arquivos, a instrução CATALOG apresentarà os primeiros 18 e, para poder prosseguir com a listagem, bastarà pressionar qualquer tecla (menos RESET).

II.5.1. Nomes de arquivos

Ao se dar um nome para um programa a fim de armazenà-lo em disquete, devem ser respeitados um limite màximo de 30 caracteres para o nome. A primeira posição deverà ser preenchida sempre por uma letra e as outras , por qualquer caracter, menos vîrgula.

II.6. OUTROS COMANDOS DO TKDOS

II.6.1 DELETE

O comando DELETE remove arquivos do disquete.

Inicialmente poderíamos fazer uma pequena experiência com o programa CONTADOR. Para não perdê-lo, podemos carregá-lo, e a seguir gravar com outro nome (JOAO, por exemplo), assim, serão dois programas iguais, com nomes diferentes e, ocupando áreas distintas do disquete.

Após realizar esta operação, confira através de CATALOG a existência deste terceiro programa e, a seguir, digite:

>DELETE JOAO (RETURN)

Após o RETURN, o drive funcionará por alguns instantes. Após isso, pode-se conferir pelo CATALOG que o programa JOAO não consta mais na lista de programas do disquete.

II.6.2. LOCK

Alguns programas ou arquivos de dados em disquete devem ser mantidos permanentemente. Para isso, o TKDOS possui uma técnica de proteção chamada "locking"(travamento). Travar um arquivo evita que ele seja acidentalmente eliminado ou que algo seja escrito sobre ele. Para travar um arquivo, basta digitar LOCK, seguido do nome do arquivo, por exemplo:

>LOCK HELLO (RETURN)

se tentar apagar um programa travado, o computador dará a seguinte mensagem de erro

<TRAVADO>

ao se dar o CATALOG aparecerá a seguinte resposta:

```
VOLUME   254

*A  ØØ1  HELLO
 A  ØØ1  CONTADOR
```

sendo que o asterisco existente no lado esquerdo do tipo de arquivo indicará que este arquivo está travado.

II.6.3. UNLOCK

Quando se deseja regravar ou apagar um arquivo travado, a única forma de se remover o LOCK è com o comando UNLOCK, por exemplo:

>UNLOCK HELLO (RETURN)

Pode-se verificar, através do CATALOG, que o asterisco correspondente ao travamento desapareceu.

II.6.4. RENAME

Para se trocar um nome de qualquer arquivo existente num disquete, utiliza-se o comando RENAME através do formato:

RENAME "velhonome","novonome"

O novo nome deverà ser diferente dos que jà existem no disco. Um RENAME para um nome jà existente provocarà duplicidade, e consequentemente problemas em posteriores acessos.

O comando RENAME deverà ser executado somente em arquivos destravados.

II.6.5. VERIFY

O comando VERIFY serve para verificar se um determinado arquivo està integro. Se quisermos verificar o programa CONTADOR, basta digitar:

>VERIFY CONTADOR

caso o programa CONTADOR esteja integro, não haverà resposta do TKDOS, caso contràrio, ocorrerà uma mensagem de erro:

I/O ERRO

E interessante observar que o comando VERIFY não verifica a lògica do arquivo armazenado, e sim, se ele è consistente com o formato usado pelo sistema operacional.

II.7. UTILIZAÇAO DE COMANDOS DO TKDOS DENTRO DE PROGRAMAS

O uso dos comandos TKDOS dentro de programas è bastante simples. Basta dar um PRINT do comando desejado, precedido por um caracter de controle (CTRL-D ou CHR$(4)).

Para exemplificar, se quisermos elaborar um programa que forneça um "CATALOG" dos arquivos em disco, basta digitar:

```
>10 HOME
>20 D$  = "" : REM ENTRE ASPAS FOI DIGITADO
    UM CONTROL-D
>30 PRINT  D$"CATALOG"
```

E importante observar que:

1. Os comandos TKDOS devem sempre estar entre aspas.

2. Sempre que definirmos o CONTROL-D, è interessante que se documente esta ocorrência com um REM.

3. Cada comando PRINT deve conter apenas um comando TKDOS.

4. A instrução PRINT que contenha comando TKDOS, nunca deverà terminar com ponto e vírgula (;), nem com vírgula (,),

5. Os seguintes comandos TKDOS sò podem ser usados dentro de programas BASIC, ou seja, não são aceitos em modo direto:

   OPEN
   APPEND
   READ
   WRITE
   POSITION

6. Os seguintes comandos TKDOS podem ser usados tanto dentro de programas BASIC como em modo direto:

BLOAD	BRUN	BSAVE	CATALOG
CLOSE	DELETE	EXEC	INIT
LOAD	LOCK	MON	NOMON
RENAME	RUN	SAVE	UNLOCK

7. O comando INIT <u>deve</u> ser usado apenas em modo direto.

CAPITULO III - ARQUIVOS SEQUENCIAIS

O TKDOS permite manipular, além de programas, arquivos que contenham informações como: listas de preços, tabelas, etc. Este tipo de arquivo é chamado de arquivo tipo TEXTO. Ao requerer CATALOG, esta classe de arquivo é caracterizada pela letra "T" que precede o nome do arquivo.

O TK-2000 pode operar com dois tipos de arquivo em disco: os sequenciais e os de acesso randômico (ou aleatórios). Cada um destes arquivos é composto de grupos de dados, também chamados registros.

Os arquivos sequênciais são caracterizados pela forma de acesso na qual para gravar ou ler um registro, é necessário que se leia todos os anteriores.

Os arquivos randômicos permitem o acesso a um registro diretamente, sem a necessidade de se passar pelos outros registros.

III.1. ATIVAÇÃO DE ARQUIVOS SEQUENCIAIS

Para manipular um arquivo sequencial, você deverá utilizar o comando OPEN. Seu formato é:

```
>OPEN  nome-do-arquivo
```

O comando OPEN só pode ser utilizado dentro de um programa BASIC, devendo estar incluído dentro de uma declaração PRINT, precedido pelo caracter CONTROL-D (CHR$ (4)).

O exemplo abaixo ativa um arquivo com o nome LIVRO.

```
>200  D$ = CHR$(4) : REM CONTROL-D
>300  PRINT  D$ "OPEN LIVRO"
>400  END
```

Ao rodar o programa acima pode-se verificar, através do comando CATALOG, que o arquivo "LIVRO" foi incluído na lista do diretório.

III.2 DESATIVAÇÃO DE ARQUIVOS SEQUENCIAIS

O programa do item anterior contém um perigo potencial de destruição do arquivo já que este não foi "fechado" após seu uso. Para efetuar este fechamento, usa-se o comando CLOSE. Este comando pode ser dado de duas formas:

```
a)   >390  PRINT CHR$(4) "CLOSE"
```

Assim irá desativar todos os arquivos que estejam abertos até este momento (é permitido ativar mais de um arquivo de cada vez) ou,

b) >390 PRINT CHR$(4) "CLOSE LIVRO"

Neste modo irà desativar especificamente o arquivo indicado, sem alterar os outros.

III.3. GRAVANDO ARQUIVOS SEQUENCIAIS

O comando para gravação de dados em arquivos è WRITE.

O comando WRITE sempre deverà estar incluìdo num comando PRINT, precedido pelo caracter de controle CHR$(4) (CONTROL-D).

Apòs a execução do comando WRITE, todas às mensagens que ocorrerem na tela de vìdeo (PRINT) serão gravadas no arquivo.

Como exemplo, inclua as seguintes linhas em nosso programa:

```
>305  PRINT  D$"WRITE LIVRO"
>310  PRINT  "PRIMEIRA FRASE DO ARQUIVO"
```

assim ao listarmos (LIST), teremos:

```
>200  D$ = CHR$(4) : REM CONTROL-D
>300  PRINT D$"OPEN LIVRO"
>305  PRINT D$"WRITE LIVRO"
>310  PRINT "PRIMEIRA FRASE DO ARQUIVO"
>390  PRINT D$"CLOSE LIVRO"
>400  END
```

Note que o prefixo CONTROL-D sò foi usado em comandos TKDOS, e nunca quando se està gravando no arquivo.

Quando se grava um arquivo sequencial, se jà existir um arquivo com o mesmo nome no disco, a nova gravação ocorrerà "por cima" do arquivo anterior, o que pode gerar efeitos desagradàveis.

Como solução para esta situação deve ser usado o comando DELETE, que removerà o arquivo anterior do disco. Este comando deverà estar incorporado num comando PRINT, precedido de CONTROL-D.

Esta solução (o comando DELETE) sò deve ser aplicada quando tivermos certeza da existência do arquivo no disquete, pòis, caso contràrio, o computador emitirà a mensagem de erro: NAO HA ARQUIVO.

Para resolver este problema, è aconselhàvel garantir a existência do arquivo atravès de um comando OPEN a mais no inicio do programa. Com esta nova opção, nosso programa obteria o seguinte aspecto:

```
>200 D$ = CHR$(4) : REM CONTROL-D
>210 PRINT D$"OPEN LIVRO"
>220 PRINT D$"DELETE LIVRO"
>300 PRINT D$"OPEN LIVRO"
>305 PRINT D$"WRITE LIVRO"
>310 PRINT "PRIMEIRA FRASE DO ARQUIVO"
>390 PRINT D$"CLOSE LIVRO"
>400 END
```

Pode-se utilizar variàveis para nomes de arquivo. Neste caso, o pròprio programa nos perguntarà o nome, como mostra o exemplo abaixo:

```
>200  D$  =  CHR$(4) : REM CONTROL-D
>205  INPUT "NOME DO ARQUIVO = ";A$
>210  PRINT  D$ "OPEN "; A$
>220  PRINT  D$ "DELETE "; A$
>300  PRINT  D$ "OPEN "; A$
>305  PRINT  D$ "WRITE "; A$
>310  PRINT  "PRIMEIRA FRASE DO ARQUIVO"
>390  PRINT  D$ "CLOSE "; A$
>400  END
```

Pode-se eliminar o efeito do comando WRITE, sem desativar (CLOSE) o arquivo. No lugar desejado acrescente um comando TKDOS neutro, como o pròprio CONTROL-D. No nosso exemplo, se antes de encerrar o arquivo quisèssemos que aparecesse algo na tela poderìamos incrementar as seguintes linhas:

```
>320  PRINT  D$
>330  PRINT  "VOCE QUER IMPRIMIR OUTRA FRASE NO
                ARQUIVO?"  :  INPUT P$
>340  IF P$  <> "S"  THEN 390
>350  PRINT  D$ "WRITE ";A$
>360  PRINT  "SEGUNDA FRASE DO ARQUIVO"
```

Deste modo, a pergunta da linha 330 não seria incluida no arquivo (apareceria somente na tela); e, para que a segunda frase seja gravada no disco temos que executar um novo comando WRITE.

OBSERVAÇAO: Existe a possibilidade de guardarmos nûmeros em arquivos sequênciais. Para isto devemos utilizar o comando PRINT, utilizando a seguir uma variàvel numèrica, ou o nuhero diretamente, observando que, apenas um valor deve ser escrito em cada declaração PRINT. Caso contràrio, a inclusão de virgulas, ou ponto e virgula, gerarà resultados diferentes do desejado.

III.4. LENDO ARQUIVOS SEQUENCIAIS

Como jà foi explicado anteriormente, a finalidade dos

arquivos è a de armazenar dados para uso futuro. No item anterior foi visto como escrever em arquivos. Agora, o comando READ serà usado para ler dados de um arquivo.

Fazendo uma analogia com o comando do item anterior, o formato do comando READ è o mesmo que do WRITE, ou seja, deve estar incluído num comando PRINT, precedido de CONTROL-D.

O comando PRINT simples, apòs o comando WRITE, transferia dados ao arquivo, agora o comando INPUT apòs o comando READ, irà ler dados do arquivo e associà-los a variàveis em memòria.

```
>500 D$ = CHR$(4) : REM CONTROL-D
>510 INPUT "NOME DO ARQUIVO A SER LIDO: ";E$
>520 PRINT D$"OPEN ";E$
>530 PRINT D$"READ ";E$
>540 INPUT A$
>550 PRINT A$
>560 PRINT D$
>570 INPUT "VOCE QUER LER MAIS DADOS ? "; Q$
>580 IF Q$ = "S"  THEN  530
>590 END
```

Atravès do programa acima note que:

1. Para que a variàvel Q$ na linha 570 não fosse associada a nenhum dado do arquivo, tivemos de anular o comando READ na linha 530 e, se quisèssemos continuar a ler o conteúdo no arquivo, teremos de retornar a linha 530 com um novo comando READ.

2. Caso quisèssemos ler mais dados do que existem no arquivo, apareceria a mensagem FIM ARQUIVO, seguido da mensagem BREAK EM 540, parando o programa.

3. No fim do programa não constou a instrução CLOSE E$. No caso de apenas leitura de um arquivo não è necessàrio que o desativemos apòs a leitura, pois neste caso nada foi gravado e nenhuma informação necessita ser atualizada no disquete.

Para muitos programadores, a ocorrência de uma mensagem de erro seguida de uma interrupção de programa seria uma coisa muito desagradàvel, para isso existem duas maneiras de evitar a mensagem FIM ARQUIVO que aparece na nota 2:

a) Pelo uso do comando ONERR GOTO, explicado em detalhes no capítulo 11, do Manual de Operação do TK-2000 COLOR e no APENDICE B do mesmo manual.

b) Se introduzirmos no final do arquivo uma palavra chave indicando fim. O programa ao ler esta palavra deverà identificà-la, e anular o comando READ, continuando então o processamento sem nenhuma mensagem de erro.

Uma forma de indicar fim de arquivo è inserir um caracter neutro, exemplo: CHR$(Ø), usando um comando PRINT antes de fechar o arquivo.

OBSERVAÇAO: Da mesma forma que utilizamos a declaração INPUT, para ler dados do arquivo, poderemos tambèm utilizar a declaração GET. Porèm, neste caso os dados seriam lidos um a um, caractere por caractere. Assim, com 8 instruções GET, seria possível ler a palavra PRIMEIRA (primeira palavra do arquivo definido no nosso primeiro programa). O ùnico incoveniente do uso do comando GET, è que sempre que for usado, o primeiro caractere do pròximo comando PRINT serà ignorado. Assim, a ùnica solução è que sempre no comando PRINT, seguinte ao GET, comece com um caracter de controle neutro ao BASIC do TK-2ØØØ (CHR$(1), por exemplo), evitando assim perdas de dados.

III.5. ADICIONANDO DADOS EM ARQUIVOS SEQUENCIAIS

III.5.1.O COMANDO APPEND

Ao se trabalhar com um arquivo sequencial, apòs este ser desativado, è perdido o controle do bloco em que o ùltimo dado foi gravado. Lembre-se tambèm que, como foi visto no item III.3, se ativarmos um arquivo, os novos dados serào escritos em cima dos anteriores, estragando o arquivo anterior.

Assim sendo, sempre que quisermos escrever num arquivo jà existente, devemos primeiramente localizar o final dele de modo ser possível aumentà-lo (incrementa-lo) com novos dados.

O comando APPEND, exerce finalidade muito semelhante ao comando OPEN, com a diferença que o segundo ativa o arquivo localizando o apontador de arquivo no início deste, enquanto o primeiro localiza o apontador no primeiro espaço nào usado.

Outra diferença importante è que o comando APPEND necessita de um arquivo que jà existe, caso contràrio emitirà a mensagem NAO HA ARQUIVO.

O formato do comando APPEND è o mesmo que o do comando OPEN ou seja, deve ser incluìdo em um comando PRINT, alèm de estar precedido do caracter de controle CONTROL-D.

III.5.2.O COMANDO POSITION

Em muitas ocasiões, teremos a necessidade de alterar algum dado em qualquer lugar do arquivo; assim, necessitamos de um controle do apontador de arquivo para poder acessar um campo específico, tanto para escrita como para leitura.

O comando POSITION sò avança o apontador do nûmero de campos partindo da sua atual posição. Os campos são delimitados pelo caracter "carriage return" (ou CHR$(13)). Caso não consiga identificar tantos campos quanto solicitado no comando, serà emitida a mensagem de erro FIM ARQUIVO.

Um exemplo deste comando seria:

```
>500  PRINT  D$ "POSITION ARQUIVO, R10"
```

O nûmero que segue a R deve estar entre Ø e 32767.

No exemplo acima o apontador seria deslocado 1Ø registros a partir da posição atual contando 1Ø carriage return. Se usàssemos RØ, o apontador não se moveria.

Note que:

1. O comando de localização do apontador deve ser sempre utilizado antes do que qualquer comando de leitura, ou de escrita; caso contràrio este comando os cancelaria por se tratar de um comando TKDOS (vide os ìtens anteriores)

2. Antes de utilizarmos o comando POSITION devemos ativà-lo atravès do comando OPEN que localizarà o apontador no inicio, deixando inicialmente a àrea toda do arquivo á disposição do POSITION. Note tambèm que se o arquivo fosse ativado com um comando APPEND, o comando POSITION perderia a validade em alguns casos jà que este não movimenta o apontador para tràs.

CAPITULO IV - ARQUIVOS DE ACESSO RANDOMICO (ALEATORIO)

Os arquivos de acesso randômico tem um comportamento muito mais flexivel que os de acesso sequencial. A principal diferença è que neste caso não hà necessidade de acessar o arquivo inteiro para alcançar qualquer ponto em seu interior como no sequencial. Neste tipo de arquivo podemos acessar qualquer campo do arquivo independente de sua posição.

O arquivo de acesso randômico, assim como o arquivo sequencial, è composto por vàrios grupos de dados tambèm chamados registros.

O arquivo de acesso randômico, inicialmente è formado apenas por um registro, mas quando o aumento de informações ultrapassa o comprimento definido do registro automaticamente è formado um outro registro cuja identificação se darà por um valor inteiro maior em uma unidade que o registro anterior. Assim o primeiro registro do arquivo receberà o nûmero Ø, o segundo receberà o nûmero um, o terceiro receberà dois, etc...

Deverà ser observado que, os arquivos do tipo randômico tendem a ocupar mais espaço no disco, do que um arquivo sequencial.

IV.1. ATIVAÇAO DE ARQUIVOS DE ACESSO RANDOMICO

Para se ativar um arquivo de acesso randômico, o procedimento è semelhante ao do arquivo sequencial, com a diferença que neste caso, temos que obrigatoriamente definir o comprimento de cada registro atravès do parâmetro L.

```
>1Ø  LET D$  =  CHR$(4)  :  REM CONTROL-D
>2Ø  PRINT D$  "OPEN ARQ1,L2Ø"
```

O parâmetro L deve variar entre 1 e 32767. E importante que este parâmetro seja bem calculado, pois se for inserido algum registro (incluindo vîrgulas e o carriage return) de tamanho maior que o especificado, ocorrerão consequências imprevisîveis no armazenamento das informações.

Se, a ler um arquivo, você especificar um parâmetro (L) diferente daquele que foi utilizado para gravà-lo, o TKDOS irà utilizà-lo cegamente para calcular as posições dos registros e os dados lidos não corresponderão aos originalmente gravados.

Para fechar um arquivo de acesso randômico usa-se o comando CLOSE do mesmo modo que no sistema de arquivos sequenciais.

IV.2 LENDO E ESCREVENDO EM ARQUIVOS DE ACESSO RANDOMICO

Os arquivos de acesso randômico operam de modo semelhante aos de acesso sequencial, somente que neste caso o parâmetro registro deve, obrigatoriamente, acompanhar os comandos

READ e WRITE. Este parâmetro (R), como já foi dito, representa o registro que desejamos acessar.

```
>30  PRINT  D$ "READ ARQ1, R3"
```

ou

```
>30  PRINT  D$ "WRITE ARQ1, R3"
```

Nos exemplos acima, tanto o comando de leitura como o de escrita estavam dirigidos ao registro número 3 do arquivo ARQ1.

O último parâmetro a ser descrito é o BYTE (B). Ele pode acompanhar os comandos READ, WRITE e POSITION (logo, é também utilizável em arquivos sequenciais).

O parâmetro BYTE move o apontador para um byte específico em um registro já selecionado. No exemplo abaixo será fácil determinar sua utilidade:

```
READ ARQ1, R13, B7
```

Neste comando, a leitura iniciará no registro 13 a partir do byte 7.

Este comando, dentro de um mesmo registro, pode movimentar o apontador para frente como também para trás.

A seguir, é apresentado um pequeno exemplo de utilização de arquivos de acesso randômicos:

```
>10  REM CRIA LISTA DE NOMES
>20  D$  =  CHR$(4)  :  REM CONTROL-D
>30  INPUT "NOME: ",NO$
>40  INPUT "END.: ";EN$ :  REM COM OS DADOS
>50  INPUT "TEL.: ";TE$
>60  PRINT D$"OPEN LISTA, L150"
>70  PRINT D$"WRITE LISTA, R0"
>80  PRINT NO$ : PRINT EN$ : PRINT TE$
>90  PRINT D$ "CLOSE LISTA"
```

Para ler os dados guardados no arquivo - lista, basta seguir o programa abaixo:

```
>10  REM LE OS DADOS DA LISTA
>20  D$ = CHR$(4) : REM CONTROL-D
>30  PRINT D$ "OPEN LISTA, L150"
>40  PRINT D$ "READ LISTA, R0"
>50  INPUT NO$, EN$, TE$
>55  PRINT : REM PULA UMA LINHA
>60  PRINT NO$
>70  PRINT EN$
>80  PRINT TE$
>90  PRINT D$ "CLOSE LISTA"
```

Se quisermos montar uma lista com vários nomes, endereços e telefones associados entre si (como um caderno de telefones), basta associar-se a cada registro uma pessoa com seu endereço e telefone.

CAPITULO V - COMANDOS AUXILIARES TKDOS 3.3

V.1. COMANDO EXEC

O poderoso comando EXEC é muito semelhante ao comando RUN, e é relacionado a arquivos sequenciais que, ao invès de simples informações ou dados, contèm comandos ou linhas de programa BASIC, inseridas no arquivo. O EXEC permite que um pseudo-programa contido no arquivo texto seja executado, em um nível de controle hierarquicamente superior ao de um programa executado por RUN.

A sequência abaixo explicarà melhor este comando:

Supondo que você queira fazer um programa que execute as seguintes funções:

a) Listar o programa TESTE.
b) Paralizar a execução por alguns segundos.
c) Fornecer o CATALOG do disco.
d) Rodar o programa.

Inicialmente devemos criar e salvar o programa TESTE. Este programa poderia ser assim:

```
>NEW
>10  REM PROGRAMA TESTE
>30  HOME
>40  FOR I = 1 TO 100
>50  K = I ^ 2
>60  PRINT I,K
>70  NEXT I
>80  FOR L = 1 TO 1000 : NEXT L
>90  END
>SAVE TESTE
```

Quando a luz do disk drive se apagar, o programa TESTE jà estarà gravado no disco.

O seguinte programa, ao qual chamamos de CRIA AUTOMATO, criarà o arquivo EXEC.

Programa AUTOMATO:

```
>NEW
>10 REM PROGRAMA CRIA AUTOMATO
>20 D$  =  CHR$(4)  :  REM CONTROL-D
>30 PRINT  D$  "OPEN AUTOMATO"
>40 PRINT  D$  "WRITE AUTOMATO"
>45 PRINT "NEW"                          <-!--------------
>50 PRINT  "LOAD TESTE"                    !Sequência de
>55 PRINT  "HOME"                          !instruções
>60 PRINT  "LIST"                          !que
>70 PRINT  "100 FOR I = 1 TO 10000"!serão
>80 PRINT  "120 NEXT I : END"              !usadas
>90 PRINT  "RUN 100"                       !pelo
>100 PRINT "CATALOG"                       !EXEC
>105 PRINT "RUN                          <-!--------------
>110 PRINT  D$ "CLOSE AUTOMATO"
>SAVE CRIA AUTOMATO
```

Digitando-se então:

>RUN CRIA AUTOMATO

o arquivo texto AUTOMATO será criado em seu disquete. Para fazer funcionar o comando EXEC, basta digitar:

>EXEC AUTOMATO

e veja a sequência.

V.2. MAXFILES

No modo normal, o TK-2000 ativa até 3 arquivos simultaneamente. Muitas vezes esta quantidade não e suficiente para as nossas necessidades. Para contornar este problema existe o comando MAXFILES, que permite alterar este número para até 16 arquivos simultâneos. O formato deste comando è:

PRINT D$ "MAXFILES 8"

deste modo, agora serão permitidos que até 8 arquivos sejam ativados de uma só vez.

Convêm avisar que este comando quando necessário, deve sempre estar logo no ínicio do programa, caso contrário, muitas funções lógicas do BASIC poderão não funcionar adequadamente.

V.3. MON E NOMON

MON e NOMON são importantes ferramentas do TKDOS, que permitirão depurar programas, isto è, acompanhar seu comportamento enquanto estes estão "rodando"; maneira muito utilizada para se encontrar erros de lógica em programas.

O comando MON permite que sejam monitoradas uma série de informações. Ele pode ser desativado pelo comando NOMON, restabelecendo-se a operação normal do computador. Para este comando existem três parâmetros diferentes e independentes que podem ser aplicados:

C = monitora todos os comandos de disco (OPEN, READ, etc..)

I = monitora os comandos de entrada do disco (ler um arquivo- READ, GET)

O = monitora os comandos de saída do disco (escrever no arquivo - WRITE)

Estes parâmetros são exclusivos dos comandos MON e NOMON.

O comando MON pode também ser desativado pela instrução DSK (ativa o disk drive, carregando novamente o TKDOS) e pelo comando RESET. Por exemplo:

>MON C,I,O

V.4. TRACE

O comando TRACE, è uma ferramenta muito útil na depuração de programas BASIC. Quando um programa contêm comandos TKDOS e o TRACE è ativado; os comandos TKDOS no interior do programa não funcionarão, jà que TRACE imprime o número da linha antes do comando TKDOS. Uma solução para este problema è inserir um carriage return (CHR$(13)) antes do CONTROL-D, na definição de D$. Por exemplo:

```
10 D$ = CHR$(13)  +  CHR$(4)
```

Assim, a maioria dos comandos TKDOS poderão operar normalmente com o TRACE ativado. A exceção corresponde a leitura de disco através de INPUT sendo que neste caso o programa se deterà aguardando uma entrada pelo teclado.

CAPITULO VI - ARQUIVOS EM LINGUAGEM DE MAQUINA

O TK-2000 permite alèm do que jà foi apresentado, que se possa armazenar em disquete regiões de memòria definidas. Contida nesta parte da memòria podem existir programas em linguagem de màquina, telas de alta resolução ou qualquer outra informação.

VI.1. BSAVE

Este comando armazena no disco um arquivo de nome "k", situado em uma faixa de memòria e seu formato è:

BSAVE k, Aa, Lj

onde o parâmetro A indica o endereço inicial da porção da memòria a ser armazenada no disco; este endereço, se precedido do sinal dòlar ($) siginifica o endereço em hexadecimal, caso contràrio, em decimal.

O parâmetro L indica o comprimento em bytes, do arquivo a ser armazenado. Da mesma forma que o parâmetro A, se o valor numèrico do comprimento vier precedido do sinal ($), então indicarà que o parâmetro està apresentado em hexadecimal.
Exemplo:

>BSAVE VIDEO1, A$2000,L$2000

VI.2. BLOAD

O comando BLOAD carrega o conteùdo de um arquivo do disco para a memòria de seu TK-2000. Os programas BASIC não serão afetados com o carregamento, a menos que os dados sejam carregados na àrea do BASIC do TK-2000.
O formato do comando BLOAD è:

BLOAD k,[Aa]

Note que no caso de BLOAD, não è necessàrio acrescentar nenhum parâmetro como ocorreu em BSAVE. Isto ocorre porque o computador irà reconhecer o arquivo e, então, carregà-lo exatamente no mesmo endereço em que està quando foi armazenado no disco. Caso se utilize o parâmetro A (endereço) opcionalmente no comando BLOAD, o arquivo então serà carregado a partir do endereço especificado por A. Exemplo:

>BLOAD UTIL, A$300

VI.3. BRUN

O comando BRUN segue exatamente o comportamento de BLOAD, explicado no item anterior, somente que, apòs ter carregado o programa em linguagem de màquina este faz com que o

806

814

computador inicie sua execução.

O formato do comando BRUN è:

BRUN k

Note que o formato de BRUN obedecerà as mesmas regras de BLOAD, inclusive quanto ao parâmetro opcional A, o qual, se utilizado, carregaria o programa no endereço especificado provocando um salto (JMP), para o endereço inicial do bloco.

APENDICE A - MENSAGENS DO TKDOS

Ao ocorrer um erro reconhecido pelo TKDOS, se o processamento for desviado por um comando ONERR GOTO, o código de erro poderá ser obtido através de PEEK(222).

LINGUAGEM INEXISTENTE

Código = 1

O TKDOS sempre emitirá esta mensagem de erro quando for tentado processar um programa originalmente produzido em outra linguagem que não o APPLESOFT BASIC.

INDICE ERRO

Código = 2 ou 3

Esta mensagem de erro è emitida quando um dos parâmetros do TKDOS ou um pròprio comando TKDOS apresentar valores fora da faixa permitida.

Por exemplo:

```
>10 PRINT CHR$(4) "OPEN ARQUIVO, V344"
```

sendo que o maior valor permitido para o parâmetro Volume è 254.

DISCO PROTEGIDO

Código = 4

Esta mensagem ocorrerà sempre que for tentado gravar em um disco, com o selo de proteção no lugar. A ùnica saída, neste caso, è de remover o selo e repetir a operação.

FIM ARQUIVO

Código = 5

Geralmente esta situação ocorre quando, na leitura de um arquivo, chega-se ao final deste, e o processador tenta obter mais informação.

NAO HA ARQUIVO

Codigo = 6

Ocorre quando è tentado acessar algum arquivo inexistente no disco.

VOLUME ILEGAL

Código = 7

Ocorre quando è acessado um arquivo-texto utilizando-se o parâmetro V (Volume) e este parâmetro não confere com o valor registrado no momento da gravação deste arquivo.

I/O ERRO

Código = 8

Ocorre nas seguintes situações:

1 - O disquete não está colocado no disk drive ou a porta do deste não esta fechada.

2 - O disquete não está devidamente formatado.

3 - Existe alguma irregularidade na gravação do programa no disquete decorrida ou por dano existente no disco ou por irregularidade ocorrida no processo de gravação/leitura.

<LOTADO>

Código = 9

Ocorre quando for tentado gravar algum programa ou arquivo em um disco que não contenha espaço disponível para receber esta informação adicional. Neste caso a melhor saída seria procurar outro disco com maior área disponível ou deletar algum arquivo sem importância contido no disco aumentando assim a área livre.

O programa utilitário TKFID que já vem contido no disco MESTRE permite avaliar a área livre em um disco evitando que erros como este possam ocorrer.

<TRAVADO>

Código = 10

Ocorre quando è tentado escrever em um arquivo ou alterar um programa que foi protegido através da instrução LOCK. Neste caso, a única saída è de gravar esta informação com outro nome ou senão utilizar a instrução UNLOCK sobre o arquivo ou programa permitindo que haja esta nova alteração.

SINTAXE #ERRO

Código = 11

Ocorre ao existir um erro na sintaxe de algum comando TKDOS, por exemplo:

>15 PRINT CHR$(4)"OPEM ARQUIVO"

SEM BUFF. DISPONIVEL

Código = 12

Ocorre quando è tentado abrir um número maior de arquivos simultaneamente do que o permitido. A única saída para este caso è utilizar o comando MAXIFILES para expandir a capatidade do sistema.

ARQ. INCOMPATIVEL

Código = 13

Ocorre quando è tentado carregar um programa utilizando-se um comando incompativel com a classe de arquivo. Por exemplo, se tentarmos carregar o programa em BASIC "TESTE" com o comando destinado a leitura de programa em linguagem de máquina.

>BLOAD TESTE <RETURN>

EXCEDE MEMORIA

Código = 14

Acontece ao se carregar um programa que necessite uma memòria superior à disponivel no seu TK-2000. Geralmente ocorre quando a capacidade disponivel foi alterada através do comando HIMEM:

COMANDO ILEGAL

Código = 15

Este erro será indicado ao se tentar utilizar um comando TKDOS APPEND, OPEN, POSITION, READ ou WRITE através do modo direto. Estes comandos TKDOS devem ser usados somente através do comando PRINT nas linhas do programa.

APENDICE B - SUMARIO DOS PROCEDIMENTOS DO DOS

CARREGANDO O DOS

Quando no modo BASIC, digite DSK, seguido por RETURN.

INICIALIZANDO O DISQUETE

Para inicializar um disquete virgem basta:

a) Carregar o TKDOS 3.3

b) Colocar o disquete virgem no disk drive

c) Digitar o programa no qual se deseja apresentar o disquete, por exemplo:

```
 5  REM  APRESENTA O DISCO E SEU CATALOGO
10  D$ =  CHR$(4)  :  REM CONTROL - D
20  HOME
30  PRINT  " DISQUETE NUMERO 1"
40  PRINT  D$  "CATALOG"
```

d) Considerando que você tenha denominado o programa acima pelo nome ALO, digite então o comando:

```
> INIT  ALO
```

e) Após a luz do disk drive se apagar, o seu disquete já estará inicializado, agora o melhor passo seria retirar o disquete do drive e identificá-lo com uma etiqueta.

UTILIZANDO COMANDOS DO TKDOS DENTRO DE UM PROGRAMA

Os comandos TKDOS podem ser executados dentro de programas em BASIC, através do comando PRINT, acompanhados de um CONTROL-D. Pode-se tambem, definir o caracter CONTROL-D, já no inicio de programa, através da declaração:

```
D$  =  CHR$(4)
```

Os comandos abaixo, obrigatoriamente, deverão ser usados no modo de execução indireta (dentro de um programa), ou seja , dentro de um comando PRINT e precedido do caracter CONTROL-D:

OPEN
READ
APPEND
WRITE
POSITION

MUDANDO O PROGRAMA APRESENTADOR DO DISQUETE

Caso se queira mudar o programa apresentador de um determinado disquete, basta escrever ou carregar o novo programa apresentador e a seguir, armazenà-lo com o mesmo nome do programa que foi utilizado anteriormente.

CRIANDO E ACESSANDO ARQUIVOS SEQUENCIAIS

Na criação ou ativação de um arquivo sequencial, o comando OPEN deve sempre ocorrer antes do comando WRITE; após o WRITE, toda instrução PRINT enviarà todos os caracteres ao disquete. O comando WRITE sò serà cancelado, através de uma instrução INPUT, ou através de uma outra instrução TKDOS como, por exemplo, uma simples instrução neutra.

```
PRINT  CHR$(4)
```

Após um arquivo ser escrito, este deverà ser destivado com o comando CLOSE, sob o risco de perder todos os dados contidos no arquivo. A seguir, è apresentado um exemplo de ativação e escrita de um arquivo sequencial:

```
>10 REM PEOGRAMA PRIMEIRO
>20 D$  =  CHR$(4)  :  REM DEFININDO O CONTROL-D
>30 REM CERTIFICANDO QUE O ARQUIVO ESTEJA APAGADO
>40 PRINT  D$  "OPEN PRIMEIRO"
>50 PRINT  D$  "DELETE PRIMEIRO"
>60 REM  INAUGURANDO UM NOVO ARQUIVO
>70 PRINT  D$  "OPEN PRIMEIRO"
>80 PRINT  D$  "WRITE PRIMEIRO"
>90 PRINT "O TRANSPORTE DE DADOS"
>100 PRINT  "VAI COMECAR"
>110 FOR B = 0  TO  100  STEP  2
>120 PRINT B
>130 NEXT
>140 REM FECHANDO O ARQUIVO
>150 PRINT D$  "CLOSE PRIMEIRO"
```

Ao ativar-se um arquivo que jà existe, os dados após o comando WRITE, irào cobrir os dados jà existentes. Para que os dados comecem a entrar diretamente ao final do arquivo, basta abrir o programa diretamente com o comando APPEND no lugar de OPEN.

A seguir, serà acessado o programa exemplo "PRIMEIRO", definido acima. Caso se queira ver o que està sendo lido no disquete, pode-se utilizar o comando:

```
>MON  I

>10 REM  ACESSANDO O PROGRAMA PRIMEIRO
>15 DIM  P(60)
>20 D$  =  CHR$(4)
>30 REM ABRINDO O PROGRAMA PRIMEIRO
>40 PRINT  D$"OPEN PRIMEIRO"
>50 PRINT  D$"READ PRIMEIRO"
>60 REM LENDO INICIALMENTE OS REGISTROS DE CADEIAS
>70 INPUT  R1$,  R2$,  R3$
>80 FOR I  =  1  TO  51
>90 INPUT  P (I)
>100 NEXT
>110 REM FECHANDO O ARQUIVO PRIMEIRO
>120 PRINT D$"CLOSE PRIMEIRO"
```

O comando READ deverà sempre estar precedido pelo comando de abertura do arquivo, e sò è cancelado apòs encontrar o caracter CONTROL-D dentro de uma declaraçàò PRINT, existindo ou não, comando DOS no comando.

ACRESCENTANDO DADOS A UM ARQUIVO SEQUENCIAL

Para acrescentar dados em um arquivo sequencial, basta ao invès do comando OPEN, ativar o arquivo com o comando APPEND que, deste modo transportarà o apontador de arquivo ao final deste.

Com o apontador de arquivo no final os dados acrescentados atravès de comando WRITE irào se acumulando ao arquivo. A seguir è apresentado um exemplo desta operaçàò.

```
>10 REM  INCREMENTANDO DADOS NO ARQUIVO PRIMEIRO
>20 D$  =  CHR$(4)  :  REM CONTROL-D
>30 REM ABRINDO O ARQUIVO,  E LEVANDO O  APONTADOR
    PARA O FINAL
>40 PRINT  D$  " APPEND PRIMEIRO "
>50 PRINT  D$  " WRITE PRIMEIRO "
>60 PRINT  "FIM DA PARTE 1"
>70 REM  FECHANDO O ARQUIVO
>80 PRINT  D$  "CLOSE PRIMEIRO"
```

UTILIZANDO UM ARQUIVO SEQUENCIAL COMO UM PROGRAMA BASIC

Ao ser executado, o programa seguinte cria um arquivo-texto de nome AUTOMATICO, que conterà os comandos

```
LIST
RUN ALO
CATALOG
```

```
>10 REM  PROGRAMA CRIADOR DO AUTOMATICO
>20 D$  =  CHR$(4)
>30 PRINT  D$ "OPEN AUTOMATICO"
>40 PRINT  D$ "WRITE AUTOMATICO
>50 PRINT  "LIST"          <-----------------------------
>60 PRINT  "RUN ALO"       <-!Programa destinado ao EXEC
>70 PRINT  "CATALOG"       <-----------------------------
>80 REM  FECHANDO O ARQUIVO AUTOMATICO
>90 PRINT  D$  "CLOSE AUTOMATICO"
>RUN
```

Ao funcionar o programa CRIADOR DO AUTOMATICO, o arquivo-texto AUTOMATICO è criado no disco. Para que o conteùdo deste arquivo seja executado, basta digitar:

```
> EXEC AUTOMATICO
```

e veja os resultados.

CRIANDO E ACESSANDO ARQUIVOS DE ACESSO RANDOMICO

No exemplo abaixo, è criado um arquivo de acesso aleatòrio de nome SEGUNDO, sendo que seus registros serão limitados a 50 caracteres cada. Na sequência, o programa grava nos registros 3 a 20, o string "REGISTRO NUMERO", seguido pelo nùmero do registro. Finalmente, o programa alterarà os registros 15 e 16 para conter somente o string "MICRODIGITAL".

```
>10 REM  ARQUIVO SEGUNDO
>20 D$  =  CHR$(4)  :  REM CONTROL-D
>30 REM  APAGA O ARQUIVO SEGUNDO,SE PORVENTURA EXISTIR
>40 PRINT  D$"OPEN SEGUNDO "
>50 PRINT  D$"DELETE SEGUNDO "
>60 PRINT  D$"OPEN SEGUNDO,L50 "
>70 FOR J = 3 TO 20
>80 PRINT  D$"WRITE SEGUNDO, R" ;  J
>90 PRINT  " REGISTRO NUMERO ";J
>100 NEXT
>110 FOR J  =  15 TO 16
>120 PRINT  D$" WRITE SEGUNDO,R" ;J
>130 PRINT  " MICRODIGITAL "
>140 NEXT
>150 PRINT  D$"CLOSE SEGUNDO "
```

Note que, nas linhas 80 e 120 do programa acima, foi

utilizado o ponto e vírgula para unir o indicador do parâmetro R com o índice representado por J. Esta è uma pràtica muito usual para esta espècie de arquivo.

O programa a seguir irà verificar o arquivo formado pelo programa acima do terceiro ao vigèsimo registro. Quando este verificar que os cinco primeiros caracteres do arquivo forem MICRO, surgirà uma mensagem "HOUVE UMA ALTERACAO NO REGISTRO NUMERO s DO ARQUIVO", sendo "s" o registro alterado, apòs a mensagem a procura continuarà.

```
>10 REM  ACESSANDO O ARQUIVO SEGUNDO
>20 D$  =  CHR$(4)  :  REM CONTROL-D
>30 PRINT  D$  "OPEN SEGUNDO, L50"
>40 FOR K  =  3 TO 20
>50 PRINT D$  "READ SEGUNDO, R"  ;K
>60 INPUT  T$
>70 IF LEFT$(T$,5) = "MICRO"  THEN PRINT  "HOUVE
    UMA  ALTERACAO  NO REGISTRO  NUMERO  ";K;"  DO
    ARQUIVO"
>80 NEXT  K
>90 PRINT D$  "CLOSE SEGUNDO"
```

APENDICE C - DESCRIÇAO DOS PROGRAMAS UTILITARIOS

TKFID

O TKFID é um utilitário para uso geral em disco.
Para executá-lo digite: "BRUN TKFID"

1- Funções:

O TKFID possui as seguintes funções:

<1> - Cópia - Copia arquivos

<2> - Catalog - Cataloga todos os arquivos do disquete

<3> - Espaço - Fornece o número de bytes ocupados e os disponíveis no disquete.

<4> - Destrava - Destrava o arquivo

<5> - Trava - Trava o arquivo

<6> - Verifica - Verifica a integridade dos arquivos.

<7> - Fim - Retorna ao BASIC

2 - Nomes de arquivos:

Para fornecer nomes de arquivos você pode entrar o nome completo dos mesmos ou ainda utilizar a eficiente rotina de busca disponível neste programa.

O símbolo "=" serve para informar ao programa, durante a seleção, que qualquer caracter naquela posição é aceitável. Veja no exemplo abaixo algumas configurações possíveis.

EXEMPLO:

=XX= - Serão relacionados todos os arquivos que contenham o string "XX" em qualquer posição de seu nome.

XX= - serão relacionados todos os arquivos que tenham o seu nome iniciado por XX.

=XX - serão relacionados todos os arquivos que tenham o seu nome finalizado por XX.

3 - Controle Manual: Ao ser perguntado sobre o controle manual, responda:

- "S" - assume controle manual em que após cada nome de arquivo deve ser digitado "S" para processar ou "N" para cancelar.

- "N" - assume que todos os arquivos selecionados serão processados.

CHAIN

Este utilitário permite o encadeamento de programas sem a perda das variáveis em memória, possibilitando a um novo programa carregado utilizar estas variáveis.

Para encadear, insira as seguintes linhas no programa que efetuará o encadeamento:

```
>xxx PRINT CHR$(4) "BLOAD CHAIN,A520"
>xxx CALL 520"nome do programa a ser carregado"
```

Observações:

1 - Insira as linhas uma após a outra.
2 - Não deixe "brancos" entre o "CALL 520" e as aspas.
3 - Copie o programa "CHAIN" do disco MESTRE para seu disquete de programas, através do utilitário TKFID.

CRIA TEXTO

Este utilitário cria arquivos-texto sequenciais.
Para executá-lo, digite:

```
>RUN CRIA TEXTO      (RETURN)
```

Os registros serão inseridos através de "linhas", sendo que cada uma representa uma registro. Cada "linha" pode conter um máximo de 239 caracteres, sendo que após o seu preenchimento deve-se pressionar a tecla RETURN. Após pressionar RETURN duas vezes, será pedido o nome do arquivo a ser gravado.

LE TEXTO

Este utilitário serve para recuperar arquivos-texto gravados pelo programa CRIA-TEXTO.
Para executá-lo, digite:

```
>RUN LE TEXTO      (RETURN)
```

Em seguida, introduza o nome do arquivo-texto desejado, pressionando-se em seguida a tecla RETURN.

MINIASM

Este utilitário permite montar programas em linguagem de máquina através dos mneumônicos 6502. Digite "BRUN MINIASM" para carregar o programa.

Para iniciar sua operação digite:

```
>TK2000      RETURN      em BASIC ou,

@3F8G        RETURN      no modo monitor.
```

O modo de operação e' idêntico ao descrito no manual técnico para o mini-assembler da ROM.

Observe que não deve ser usado o comando ASS quando esteja instalada a interface de disco.

COPIA

Utilitário copiador de disquetes. Digite "RUN COPIA" para executar o programa. Verifique previamente se o disquete que receberá a cópia não possue arquivos ou informações que não possam ser apagados, pois o disco será novamente inicializado.

RENUMERADOR

Utilitário que realiza renumeração de linhas e/ou MERGE de programas BASIC.

Digite "RUN RENUMERADOR INSTRUCOES" para instruções, ou "RUN RENUMERADOR" para executar o programa.

DEMO CORES

Este utilitário gera efeitos visuais coloridos, em alta resolução para demonstração dos recursos gráficos do TK-2000 COLOR.

Para executar o programa, use o comando:

```
>RUN DEMO CORES
```

Os gráficos são gerados aleatoriamente, oferecendo um padrão de cores bastante diversificado.

O espaçamento entre as linhas impressas no vídeo (passo), também é randômico, sendo especificado abaixo da figura gerada.

A tecla espaço interrompe a execução do programa.

OTHELLO

Instruções:

Este è um jogo de estratègia que se desenvolve num tabuleiro de 8x8 posições cujo objetivo è possuir o maior numero de peças que o computador.

Para executar o programa digite "RUN OTHELLO"

Para ganhar as peças adversàrias, você deve aprisionà-las em uma das três posições (horizontal, vertical ou diagonal) entre duas peças suas.

A tecla R, ao ser pressionada, reinicia a partida e a tecla P interrompe o jogo informando o vencedor.

LISTA TELEFONICA

Instruções:

Este programa demonstra o uso do computador em arquivo pessoal de telefones.

Para executà-lo digite:

>RUN LISTA TELEFONICA <RETURN>

O programa apresenta 7 opções:

1 - Consulta de registros
2 - Listagem dos registros
3 - Entrada de novos registros
4 - Exclusão de registros
5 - Alteração de registros
6 - Impressora ligada/desligada
7 - Final do programa

EDITOR

O Editor è um programa escrito em linguagem de màquina, que auxilia nos trabalhos de edição, normalmente em programação BASIC.

Para carregà-lo, com o TKDOS ativo, digite "BRUN EDITOR", e RETURN. Aguarde atè aparecer novamente o cursor.

Agora você poderà editar programas BASIC ou outros comandos digitados diretamente pelo teclado.

O processo adotado para a edição, neste programa, è do tipo TELA, o qual exige que o texto a ser editado esteja constando na tela do televisor (ou monitor de vìdeo) no momento da edição.

A edição è efetuada em duas etapas. A primeira delas terà como objetivo alcançar o ponto, na tela, onde se encontra

o texto a ser editado. Para isso o cursor deverà ser conduzido usando as teclas de movimentação neutra, explicadas mais adiante.

A segunda etapa, com o cursor posicionado, o texto deverà ser "digitado" utilizando-se das teclas de movimentação horizontal (-->) e (<--), ao mesmo tempo em que se corrige o texto existente.

As teclas de movimentação neutra são:

(CONTROL-Z) - habilita as teclas seguintes a trabalhar no modo de movimentação neutra;

A - leva o cursor para direita, sò 1 posição;
B - leva o cursor para esquerda, sò 1 posição;
C - leva o cursor para baixo, sò 1 posição;
D - leva o cursor para cima, sò 1 posição;

I - leva o cursor para cima, com repetição;
J - leva o cursor para a esquerda, com repetição;
K - leva o cursor para direita, com repetição;
M - leva o cursor para baixo, com repetição.

Apòs o CONTROL-Z, a digitação de qualquer tecla que não as indicadas acima, anularà o modo de movimentação neutra, portanto se for necessàrio outra movimentação, deverà ser novamente aplicado um CONTROL-Z.

O processo de movimentação do cursor è efetuado com a tecla -->, ao "passar por cima" de um caracter na tela, equivalerà a "redigitar" o dito caracter, enquanto que a tecla <-- "apaga" o caracter.

Usaremos daqui por diante as expressòes "mova o cursor", querendo indicar o uso das teclas de movimentação neutra, "redigite" para usar a tecla -->, e "apague" para usar a tecla <--. Todo processo de edição è finalizado pressionando-se RETURN.

Exemplo: Digite textualmente o seguinte programa:

```
>10 PRIMT "MEU NOME E ";
>20 PRINT "JOAP"
>
```

Percebendo dois erros de sintaxe, um em cada instrução, você irà corrigi-los.

Usando inicialmente CONTROL-Z, "mova" o cursor atè o 1 da linha 10. "Redigite" a linha atè o cursor ficar em cima da letra M, digite N, e continue "redigitando" atè o final da linha, ultrapassando o ponto e vìrgula. Pressione RETURN.

Efetue um LIST e você verà que a linha 10 foi corrigida. Agora, mova o cursor atè ficar em cima do 2 da linha 20, e redigite atè o caracter "P", digite no lugar o caracter "O", e redigite atè o fim da linha. Pressione RETURN e efetue um novo LIST para conferir.

Observe que a tecla de redigitação permite copiar textos que aparecem na tela. Suponhamos que você queira incrementar a seguinte linha:

30 PRINT "MEU NOME E JOAO"

Em vez de digitar esta linha diretamente, usaremos as funções de edição.

Efetue um LIST, digite 30, acione CONTROL-Z e mova o cursor até a linha 10 no início da palavra PRINT, redigite até as aspas finais. Acione CONTROL-Z e mova o cursor até a letra J da linha 20, redigite até o fim da linha e pressione RETURN. Liste agora o programa.

Para inserir um texto dentro de uma linha de programa deve-se preceder como no exemplo adiante. Nele iremos inserir TAB(10) na linha 30.

Liste o programa emova o cursor até o número 3 da linha 30, redigite até após a palavra PRINT, "mova" o cursor para cima 1 posição, digite "TAB(10)"; mova o cursor novamente para linha 30 para a mesma posição de onde o cursor foi deslocado. Dai, redigite até o fim da linha. Pressione RETURN e confira o programa.

Existem ainda, outras funções auxiliares habilitadas também por CONTROL-Z usando as seguintes teclas:

E - Apaga a linha, desde onde se encontra o cursor até o final da mesma

F - Apaga a tela desde onde a linha em que o cursor se encontra até o final.

Use CONTROL-T <RETURN> ao invés de HOME para limpar a tela sempre que o programa EDITOR estiver ativado.

IMP01

Software para impressora. Para carregá-lo do disco MESTRE, digite "BRUN IMP01".

O modo de utilização é descrito no manual que acompanha o kit de impressora.

MODULO DE REFERENCIA - TKDOS

Neste cartão, os comandos TKDOS estão agrupados em 5 categorias:

-Comandos próprios do TKDOS:

INIT	LOAD	DELETE	VERIFY	MAXFILES
CATALOG	RUN	LOCK	MON	SAVE
RENAME	UNLOCK	NOMON		

-Comandos de acesso:

DSK

-Comandos de arquivos-texto sequenciais:

OPEN	READ	APPEND	EXEC
CLOSE	WRITE	POSITION	

-Comandos de arquivos-texto aleatórios:

OPEN	CLOSE	READ	WRITE

-Comandos de arquivos em linguagem de máquina.

BLOAD	BRUN	BSAVE

NOTAÇAO E SINTAXE

Os parâmetros serão representados por letras maiúsculas, normalmente acompanhadas por um número (representado por letras minúsculas), que oferece informações adicionais para executar um comando. Os parâmetros múltiplos podem aparecer em qualquer ordem, mas devem estar separados um do outro por vírgulas. Os parâmetros apresentados entre colchetes [como estes] são opcionais.

Um nome-de-arquivo (apresentado aqui por uma letra X) devem estar imediatamente após uma palavra de controle. Estes nomes devem obrigatoriamente começar com uma letra; somente os 30 primeiros caracteres serão reconhecidos como nome-de-arquivo. Uma vírgula deverá separar o nome dos seus parâmetros.

CHR$(4) ou CONTROL-D (digitando <D> enquanto a tecla CONTROL já está pressionada é usada num comando PRINT para indicar o

início de um comando TKDOS. Por exemplo:

```
>10 D$ = CHR$(4):REM CHR$(4) = CONTROL-D
>20 PRINT D$; "CATALOG"
```

PARAMETROS DOS COMANDOS TKDOS

Uma mensagem de erro irá ocorrer se a quantidade apresentada no parâmetro TKDOS for muito grande ou muito pequena.

Parâmetros gerais de arquivo:

Parâmetro	Apresentação	Min	Max
Volume	,Vv	V0*	V254

* Utilizando V0 è equivale a omitir o parâmetro Volume: o menor número de volume que o comando INIT irá reconhecer será o 1.

Arquivos Texto Sequenciais:

Parâmetro	Apresentação	Min	Max
Byte	,Bb	B0	B32767
Registro*	,Rp	R0	R32767

* Com EXEC, sempre está relativo para o registro 0.

Arquivos Texto de Acesso Randômico:

Parâmetro	Apresentação	Min	Max
Comprimento de linha	,Lj	L1	L32767
Número de linha	,Rr	R0	R32767

Arquivos Binários

Parâmetro	Apresentação	Min	Max
Endereço inicial	,Aa	A0	A65535
Número de bytes	,Lj	L1	L32767

Comando TKDOS

Parâmetro	Apresentação	Min	Max
MAXFILES	MAXFILES n	n=1	n=16

Se um comando omitir o parâmetro Volume ou utilizar V0, o número do volume do disquete è ignorado. Um comando que utilizar o parâmetro Volume Vv não irá

executar a menos que o número de volume seja v.

Comandos próprios do TKDOS

INIT X [,Vv]
Inicializa um disquete virgem para o formato de disco auxiliar. Assume como programa apresentador o programa de nome X e número de volume v (se especificado).

CATALOG
Apresenta o número de volume e todos os arquivos e programas contidos no disquete com seus respectivos tipos e comprimentos (número de setores utilizados). O símbolo * indica um arquivo travado.

TIPO	DESCRIÇAO
I	INTEGER BASIC (não roda no TK-2000)
A	Programa BASIC tipo APPLESOFT
T	Arquivo-texto
B	Programa em linguagem de máquina

SAVE X [,Vv]
Grava o programa BASIC contido na memória no disquete com o nome X. Escreve sobre qualquer arquivo anterior que seja do mesmo tipo e com o mesmo nome sem fornecer qualquer aviso.

LOAD X [,Vv]
Carrega para a memória um programa BASIC contido no disco, depois de limpá-la do programa anterior.

RUN X [,Vv]
Carrega o programa BASIC com o nome X executando o mesmo a seguir.

RENAME X,Y [,Vv]
Troca o nome do arquivo X para Y.

DELETE X [,Vv]
Apaga o arquivo X do disquete.

LOCK X [,Vv]
Trava o arquivo X contra uma mudança ou

apagamento acidental. Este travamento pode ser observado no CATALOG através do *.

UNLOCK X [,Vv]
Destrava o arquivo X (caso este já estivesse travado) permitindo mudança ou deletação do mesmo.

VERIFY X [,Vv]
Checa o arquivo X em sua consistência interna. Se X não possuir nenhum erro, não será fornecido nenhuma mensagem.

MON [,C] [,I] [,O]
Causa a apresentação dos comandos de disco (C), instruções de INPUT no disco (I), e instruções de OUTPUT para o disco (O). Se nenhum destes parâmetros acompanhar o comando, este será ignorado.

NOMON [,C] [,I] [,O]
Cancela a apresentação dos comandos de disco (C), instruções de INPUT no disco (I), e instruções de OUTPUT para o disco (O). Se nenhum destes parâmetros acompanhar o comando, este será ignorado.

MAXFILES n
Reserva n buffers de arquivo para a entrada e saída de disco (normalmente são reservados apenas 3 buffers de arquivo). Este comando deve ser usado sempre antes de carregar ou rodar um programa.

COMANDO DE ACESSO

DSK
Acessa o disk drive introduzindo o TKDOS no sistema. Caso o TKDOS já estava contido na memória, os programa BASIC contido nela será apagado entrando em seu lugar o programa apresentador do novo disco.

COMANDOS DE ARQUIVOS-TEXTO SEQUENCIAIS

OPEN X [,Vv]
Abre ou cria um arquivo sequencial com o nome X, atribuindo a ele um buffer, e preparando para receber os comandos WRITE ou

READ para o início do arquivo.

CLOSE [X]
Completa o comando WRITE X, se necessário, e desatribui o buffer reservado para o arquivo-texto X. Sem o nome do arquivo, este comando irà fechar todos os arquivos que eventualmente estejam abertos nesta situação (exceto ou arquivo EXEC).

WRITE X [,Bb]
Permite que a instrução PRINT envie caracteres para o arquivo-texto sequencial X. A escrita inicia-se no início do arquivo ou (se especificado) pelo byte b. Este comando è cancelado por qualquer comando TKDOS posterior a ele.

READ X [,Bb]
Permite que a instrução INPUT leia caracteres do arquivo-texto X. A leitura inicia-se no início do arquivo ou (se especificado) pelo byte b. A leitura inclui um registro por vez, sendo que o registro è delimitado pelo caracter RETURN. Este comando è cancelado por qualquer comando TKDOS posterior a ele.

APPEND X [,Vv]
Abre um arquivo-texto X jà existente, este comando è similar ao OPEN, mas difere que este prepara o arquivo para receber o comando WRITE diretamente no final deste de modo a acrescentar mais informação.

POSITION X, Rp
Em um arquivo-texto aberto atravès do comando OPEN, permite deslocar o indicador de arquivo atè uma região indicada pelo parâmetro p.

EXEC X [,Rp] [,Vv]
Executa sucessivamente os campos de um arquivo-texto sequencial X interpretando cada campo deste arquivo como uma linha de um programa BASIC. O parâmetro p serve apenas para indicar de qual campo (linha) se deseja iniciar a execução.

COMANDOS DE ARQUIVO-TEXTO ALEATORIOS

OPEN X, Lj [,Vv]

Abre ou cria um arquivo de acesso randômico de nome X, atribuindo um buffer para este arquivo, e definindo um comprimento de campo de j bytes. Prepara para receber um comado WRITE ou READ do início do campo Ø. O mesmo comprimento de campo deve ser atribuído a cada vez que se abre o mesmo arquivo.

CLOSE [X] [,Vv]

Finaliza o comando WRITE X, se necessário e desatribui o buffer reservado ara o arquivo texto X. Se o nome do arquivo, o comando CLOSE fecharà todos os arquivos abertos neste instante.

WRITE X [,Rr] [,Bb]

Permite que a instrução PRINT escreva informação no arquivo X. Quando não è apresentado nenhum parâmetro, o comando inicia diretamente na posição do apontador de arquivo. Com apenas o parâmetro Rr, o comando inicia no byte Ø do campo r. Com o parâmetro Bb, o comando inicia no byte b da posição do apontador de arquivo (campo r, caso este esteja especificado). Este comando è cancelado com qualquer comando TKDOS posterior a ele.

READ X [,Rr] [,Bb]

Permite que a instrução INPUT leia os caracteres do arquivo-texto de acesso aleatòrio X. A identificação da posição do inicio de leitura tem o mesmo comportamento do comando WRITE, descrito acima.

COMANDOS DE ARQUIVOS BINARIOS

BSAVE X, Aa, Lj [,Vv]

Guarda em disquete, sob o nome X, o conteùdo dos j bytes de memòria começando pelo endereço a.

BLOAD X [,Aa] [,Vv]

Carrega o arquivo binàrio X no mesmo endereço de memòria no qual ele foi origi-

nalmente salvado ou (se especificado) no endereço a.

BRUN X [,Aa] [,Vv]

Carrega o arquivo binário X, e executa a instrução JMP para o endereço inicial deste arquivo.

MICRODIGITAL
ELETRÔNICA LTDA.
Caixa Postal 54.088 — São Paulo — Brasil