

**TK3000 //c**

**TKDOS 3.3**

**MICRODIGITAL**

# TKDOS 3.3

Projeto e execução editorial:  
Departamento de Publicações Técnicas da Microdigital  
Eletrônica Ltda.

É vedada a reprodução total ou parcial deste manual sem  
prévia autorização por escrito da Microdigital Ele-  
trônica Ltda.

A Microdigital reserva-se o direito de alterar futura-  
mente qualquer especificação técnica deste produto.

MICRODIGITAL ELETRÔNICA LTDA.

CAIXA POSTAL 54.121  
São Paulo (SP)

Impresso no Brasil - Printed in Brazil  
1ª edição - 1986

## SUMÁRIO

1. INSTRUÇÕES GERAIS .....	1
1.1 - O Disquete .....	1
1.2 - A Organização do Disquete .....	1
1.3 - O TKDOS 3.3 .....	2
1.4 - Instalação do Disk Interface Card .....	3
1.5 - Cuidados com Unidades de Disco e Disquetes .....	4
2. OPERAÇÃO INICIAL .....	6
2.1 - Introduzindo o TKDOS 3.3 .....	6
2.2 - "BOOT" do TKDOS 3.3 .....	6
2.3 - Inicializando Novos Disquetes .....	7
2.4 - Selecionando o Slot e o Drive .....	9
2.5 - Gravando e Lendo Programas pelo TKDOS ..	9
2.6 - O Comando CATALOG .....	12
2.6.1 - Nomes de Arquivos .....	13
2.7 - Outros Comandos do TKDOS .....	13
2.7.1 - DELETE .....	13
2.7.2 - LOCK .....	14
2.7.3 - UNLOCK .....	14
2.7.4 - RENAME .....	15
2.7.5 - VERIFY .....	15
2.7.6 - O Comando "-" (DASH).....	16
2.8 - Utilização de Comandos do TKDOS Dentro de Programas.....	16
3. ARQUIVOS SEQUENCIAIS .....	18
3.1 - Ativação de Arquivos Seqüenciais .....	18
3.2 - Desativação de Arquivos Seqüenciais ....	19
3.3 - Gravando Arquivos Seqüenciais .....	19
3.4 - Lendo Arquivos Seqüenciais .....	22
3.5 - Adicionando Dados em Arquivos Seqüenciais.....	24
3.5.1 - O Comando APPEND .....	24
3.5.2 - O Comando POSITION .....	25

4. ARQUIVOS DE ACESSO RANDÔMICO .....	27
4.1 - Ativação de Arquivos de Acesso Randômi- co .....	27
4.2 - Lendo e Escrevendo em Arquivos de Acesso Randômico .....	28
5. COMANDOS AUXILIARES DO DOS .....	30
5.1 - O Comando EXEC .....	30
5.2 - Comando MAXFILES .....	32
5.3 - Comandos MON e NOMON .....	32
5.4 - Usando o Comando TRACE com o TKDOS.....	33
6. ARQUIVOS EM LINGUAGEM DE MÁQUINA .....	34
6.1 - BSAVE .....	34
6.2 - BLOAD .....	34
6.3 - BRUN .....	35
APÊNDICE	
A - MENSAGENS DO TKDOS .....	36
B - DESCRIÇÃO DOS PROGRAMAS .....	40
C - MÓDULO DE REFERÊNCIA .....	46

## 1. INSTRUÇÕES GERAIS

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

### 1.1 - O DISQUETE

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

### 1.2 - A ORGANIZAÇÃO DO DISQUETE

Para que as informações possam ser armazenadas no disco, é necessário que este esteja organizado, pois o computador requer pontos de referência para a leitura e a gravação.

Esta organização é alcançada com a formatação do disco, que define o disquete com 35 trilhas concêntricas numeradas de 0 a 34 a partir da borda externa em direção ao centro, cada uma delas dividida em 16 setores numerados de 0 a 15, conforme mostra a figura 1.

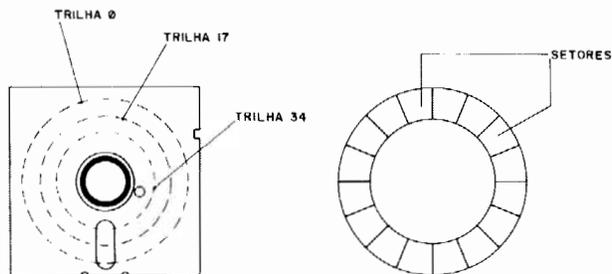


Figura 1

### 1.3 - O TKDOS 3.3

Acompanhando a interface de disco, é fornecido um disquete denominado Sistema Master. Este disquete contém programas especiais que permitem ao computador usar o disk drive, além de alguns programas utilitários.

O programa central, o mais importante do Sistema Master, é denominado TKDOS 3.3. O TKDOS é o programa que formaliza as interações entre o computador e o disk drive. Uma vez introduzido na memória, ele habilitará uma série de novos comandos, que se constituirão em ferramentas importantes no manuseio dos disquetes.

Na borda do disquete Sistema Master existe um selo de proteção. Este selo NÃO deve ser retirado, pois protege o disco contra gravações. Caso seja necessário gravar, é aconselhável adquirir outro disquete e seguir as instruções contidas neste manual.

selo de proteção

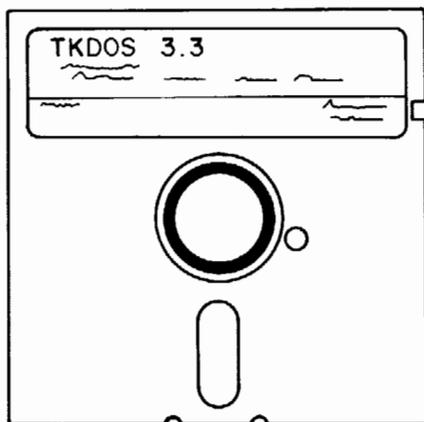


Figura 2

#### 1.4 - INSTALAÇÃO DO DISK INTERFACE CARD

É aconselhável a leitura total deste capítulo antes de iniciar a instalação.

Observe o seguinte procedimento para a instalação:

- Certifique-se de que o TK3000 //e esteja desligado.
- Retire a tampa do gabinete do computador e toque a fonte de alimentação, para descarregar a energia estática de seu corpo e de suas roupas.
- Conecte o(s) cabo(s) da(s) unidade(s) de disco ao Disk Interface Card.

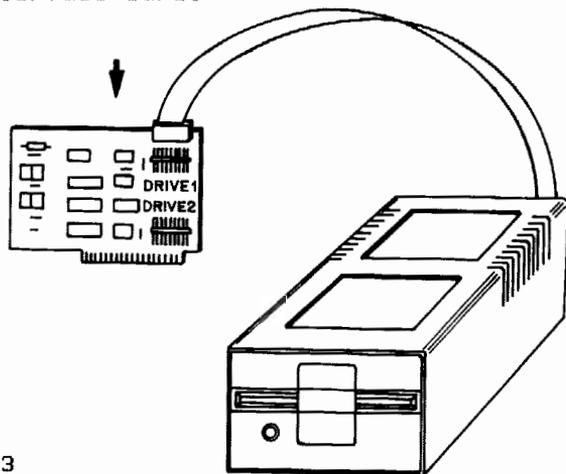


Figura 3

- Introduza o Disk Interface Card no slot adequado. Consulte o capítulo II do Manual do TK3000 //e, para saber quais slots são recomendados.

Nota: Usualmente se instala a unidade de disco no slot 6.

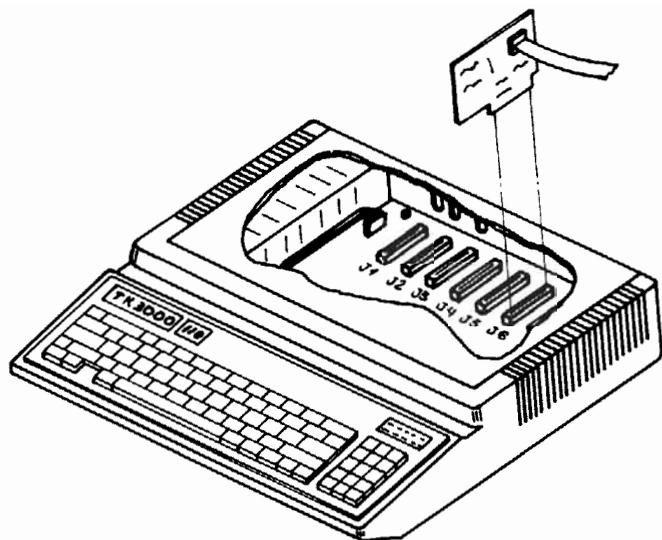


Figura 4

f) Recoloque a tampa do gabinete do TK3000 //e.

#### 1.5 - CUIDADOS COM UNIDADES DE DISCO E DISQUETES

O disk drive é um dispositivo mecânico, contendo motores e partes móveis que o tornam tão suscetível a acidentes quanto o computador. Assim, deve-se protegê-lo de quedas e do choque contra objetos, evitando-se a danificação de componentes ou o mau funcionamento.

O disquete é um pequeno disco plástico revestido com óxido metálico, de maneira a permitir gravação magnética sobre sua superfície. O disquete é permanentemente acondicionado num invólucro quadrado de cor preta, que lhe oferece proteção, ajudando a mantê-lo limpo e facilitando o movimento giratório do disco. Esta embalagem nunca deve ser aberta.

Deve-se evitar tocar na parte exposta do disquete, pois mesmo a impressão digital poderá apagar, ou até destruir a superfície do disquete. Assim, mantenha sempre seus disquetes acondicionados em suas embalagens origi-

nais, prolongando a vida de seus dados e programas.

Muito embora sendo o disquete seja chamado de "floppy-disk" (floppy = flexível), é completamente desaconselhável dobrá-los, pois certamente haverá danos ao disco.

Deve-se evitar expor o disquete a variações bruscas de temperatura, respeitando o limite mínimo de 10 e máximo de 50 graus.

O disquete deve ser utilizado ou guardado em ambientes secos e livres de poeira.

Outra recomendação muito importante é a de se evitar escrever sobre os discos. As etiquetas de identificação devem ser atualizadas com canetas de ponta macia.

## 2. OPERAÇÃO INICIAL

### 2.1 - INTRODUZINDO O TKDOS 3.3

O processo da introdução do TKDOS (ou de qualquer outro sistema operacional), no computador, é denominado "boot". Com este processo carrega-se, numa área predefinida da memória do TK3000 //e, o sistema operacional, que permitirá a utilização de novos comandos de controle para a manipulação de disquetes.

### 2.2 - "BOOT" DO TKDOS

Para efetuar o "boot", desligue o computador e insira o disquete do Sistema Master na unidade de disco, com o lado da etiqueta voltado para cima.

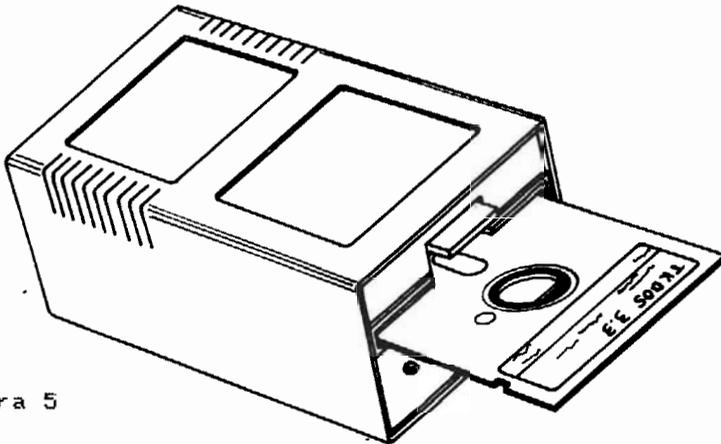


Figura 5

Feche a porta da unidade de disco e ligue o computador.

O led de seu drive imediatamente se acende, e o TKDOS é carregado.

Caso o computador já esteja ligado, para carregar o TKDOS, tecle o comando:

]PR# número do slot (return)

Ao término da carga, a tela apresenta o seguinte display:

TK3000 //e

TKDOS VERSAO 3.3 SISTEMA MASTER

1986 - MICRODIGITAL ELETRONICA LTDA.

]

Nota: Os comandos do TKDOS e os do BASIC podem ser digitados com letras maiúsculas ou minúsculas, indiferentemente. No entanto, os nomes dos arquivos devem ser digitados no mesmo formato em que foram especificados.

### 2.3. - INICIALIZANDO NOVOS DISQUETES

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

- a) Carregue o TKDOS no computador como já foi explicado.
- b) Retire o disquete Sistema Master e introduza o disquete a ser inicializado.
- c) Digite NEW (return), entrando em seguida o programa introdutório abaixo:

```
J5 REM INICIALIZACAO DO PRIMEIRO DISQUETE
J10 HOME : REM LIMPA A TELA
J20 PRINT "DISQUETE NUMERO 1"
J30 PRINT "OPERADOR: (escreva seu nome)"
J40 END
```

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

d) A seguir escreva:

```
INIT HELLO
```

Após digitar return, o disk drive irá funcionar por cerca de dois minutos, para formatar o disquete e gravar o sistema operacional.

Neste comando também podem ser utilizados os parâmetros indicativos do slot [Sn], do drive [Dn] e do volume [Vn]. A letra n indica um número a ser digitado, que varia de 1 a 7, para S; de 0 a 254, para V; e assume 1 ou 2, para D. Assim, poderemos acrescentar no disco uma identificação que, em caso de necessidade, poderá diferenciá-lo de outros discos com arquivos semelhantes ou instalados em outros periféricos.

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

```
INIT HELLO [,S6][,D1][V1]
```

Caso estes parâmetros não sejam incluídos no comando INIT o volume será assumido como 254, o slot como 6 e o drive como 1.

Nota: Os colchetes [...] não devem ser digitados. Tais sinais são usados neste manual para indicar parâmetros opcionais.

e) Finalmente para testar o novo disquete, existem duas alternativas:

- Desligar e ligar o computador com o disquete dentro do drive; ou,
- Digitar: PR# número do slot (return)

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

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

#### 2.4 - SELECIONANDO O SLOT E O DRIVE

A maioria dos comandos do TKDOS permitem que se escolha o slot e o drive desejados.

Para se definir em que slot está conectado o Disk Interface Card, coloca-se, no final do comando TKDOS, uma vírgula seguida do parâmetro "Sn" (n = número do slot onde o cartão está instalado).

Cada interface de disk drive pode controlar até duas unidades de disco, assim, pode-se especificar o drive desejado. Para tanto, coloca-se, no final do comando do TKDOS, uma vírgula seguida do parâmetro "Dn" (n = número do drive).

Nota: Os comandos do TKDOS podem ser digitados em caracteres minúsculos.

#### 2.5 - GRAVANDO E LENDO PROGRAMAS PELO TKDOS

Chame o TKDOS utilizando:

```
JPR# número do slot (return)
```

Em seguida digite:

```
JNEW (return)
```

Este comando apagará então o programa HELLO que deverá estar na memória.

Escreva o seguinte programa:

```
J5  REM PROGRAMA CONTADOR
J10 FOR I = 0 TO 9
J20 PRINT I
J30 NEXT I
J40 END
```

Ao executar o programa, o resultado será este:

```
JRUN
```

```
0
1
2
3
4
5
6
7
8
9
```

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

```
JSAVE CONTADOR
```

Obs.: Podem ser especificados os parâmetros S, D e V. Após pressionar return a unidade de disco irá funcionar por alguns segundos e, quando o led desta apagar-se e o cursor aparecer novamente, a operação estará terminada.

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

a) JNEW (return)

Para apagar o programa original que ainda consta na memória.

b) JLOAD CONTADOR (return)

Obs.: Podem ser especificados os parâmetros S, D, V.

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:

JRUN (return)

O resultado já apresentado se repetirá.

Obs: Caso, no momento de carregar o programa, fosse digitado um nome que não constasse 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 é, após ter apagado o programa da memória (comando NEW), digitar:

JRUN CONTADOR (RETURN)

Obs.: Podem ser especificados os parâmetros S, D e V. O computador se encarregará de procurar o programa no disquete, carregá-lo e, em seguida, executá-lo. Este modo é mais direto que o anterior. Verifique os dois métodos para sentir a diferença.

## 2.6 - O COMANDO CATALOG

O comando CATALOG lista na tela os nomes dos arquivos existentes no disquete, ou seja, o diretório deste.

Para cada arquivo do disquete, o CATALOG fornece o número de setores ocupados e o tipo de arquivo, e também informa se o arquivo está travado ou não. O CATALOG ainda fornece o número de setores disponíveis do disquete.

Ao digitar:

```
]CATALOG [ ,Dn] [ ,Sn] [ ,Vn] (return)
```

A tela do TK3000 //e apresentará:

```
504 SETORES DISPONIVEIS  
A 002 HELLO  
A 001 CONTADOR
```

]

A primeira linha impressa mostra o número de setores livres no disquete. A letra "A", na coluna à esquerda, indica o tipo de arquivo que foi armazenado. No caso, indica que se trata de um arquivo em BASIC; se aparecesse a letra "B", indicaria um arquivo binário.

Caso você queira parar a listagem para observá-la, pressione qualquer tecla. O CATALOG interrompe-se até que você pressione novamente qualquer tecla.

Opcionalmente, você pode obter somente a listagem dos arquivos de determinado tipo ou daqueles cujo nome começa com determinada letra, ou ainda dos que sejam de determinado tipo e tenham o nome iniciado por determinada letra. Para isso, use o comando CATALOG seguido dos parâmetros desejados:

```
CATALOG [tipo de arquivo] [nome de arquivo]
```

[tipo de arquivo] - caractere que representa o tipo de arquivo desejado (A, B, T ou I).

Se no lugar do tipo deixa-se um espaço em branco, são listados arquivos de todos os tipos.

[nome de arquivo] - primeira letra do nome de arquivo.

Obs.: Não se deve deixar espaços entre o comando CATALOG e estas opções. Por exemplo, para se obter a listagem de todos os arquivos TEXTOS cujo nome comece com a letra R, deve-se digitar:

RCATALOGTR

### 2.6.1 - Nomes de Arquivos

Ao se dar um nome para um programa a fim de armazená-lo em disquete, devem ser respeitados um limite máximo de 30 caracteres para o nome. A primeira posição deverá ser preenchida sempre por uma letra maiúscula e as outras, por qualquer caractere, exceto a vírgula.

## 2.7 - OUTROS COMANDOS DO TKDOS

### 2.7.1 - DELETE

O comando DELETE remove arquivos do disquete. Sua sintaxe é:

DELETE nome do arquivo [,Sn] [,Dn] [,Vn]

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:

JDELETE 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.

2.7.2. - 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.

Sua sintaxe é:

```
JLOCK nome do arquivo [,Sn] [,Dn] [,Vn]
```

Se você tentar apagar ou gravar novamente um programa travado, ou ainda alterar o seu nome, o computador dará a seguinte mensagem de erro:

TRAVADO

Ao se dar o comando CATALOG, os arquivos travados são indicados por um asterisco, no lado esquerdo de seu indicativo de tipo. Por exemplo:

```
JLOCK HELLO
JCATALOG
xxx SETORES DISPONIVEIS
*A 002 HELLO
A 001 CONTADOR
```

2.7.3. - UNLOCK

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

```
JUNLOCK HELLO (RETURN)
```

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

Este comando tem a seguinte sintaxe:

```
UNLOCK [,Sn] [,Dn] [,Vn]
```

#### 2.7.4. - RENAME

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

```
RENAME velhonome,novonome [,Sn] [,Dn] [,Vn]
```

O novo nome deverá ser diferente dos que já existem no disco. Um RENAME para um nome já existente provocará duplicidade, e conseqüentemente problemas em posteriores acessos.

O comando RENAME somente poderá ser executado em arquivos destravados.

#### 2.7.5. - VERIFY

O comando VERIFY serve para verificar se um determinado arquivo está íntegro.

Sua sintaxe é:

```
VERIFY nome do arquivo [,Sn] [,Dn] [,Vn]
```

Se quisermos verificar o programa CONTADOR, basta digitar:

```
JVERIFY CONTADOR
```

Caso o programa CONTADOR esteja íntegro, não haverá resposta do TKDOS, caso contrário, ocorrerá uma mensagem de erro:

## I/O ERRO

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

### 2.7.6 - Comando "--" (DASH)

A função deste comando é a de executar arquivos de qualquer tipo (B, I, A ou T).

Este comando somente pode ser usado para executar programas do tipo I, se você estiver em Integer BASIC. E somente pode ser usado para programas do tipo A, se você estiver em BASIC.

Nota: Para entrar ao Integer BASIC, digite: INT. O sinal de prompt muda para >. Para sair, digite FP. O sinal de prompt volta ao normal.

Sua sintaxe é:

- nome do arquivo [ ,Sn] [ ,Dn] [ ,Vn]

Exemplos:

MODO DIRETO

MODO INDIRETO

J - HELLO

J10 PRINT CHR\$(4);"-- HELLO"

Nos dois modos o arquivo será carregado na memória (e executado), independentemente de seu tipo.

## 2.8 - UTILIZAÇÃO DE COMANDOS DO TKDOS DENTRO DE PROGRAMAS

O uso dos comandos TKDOS dentro de programas é bastante simples. Basta dar um PRINT do comando desejado, prece-

dido por um caractere de controle: control-D ou CHR\$(4).

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

```
J10 HOME
J20 D$ = "" : REM ENTRE AS ASPAS FOI DIGITADO
    UM CONTROL-D (CHR$(4))
J30 PRINT D$"CATALOG"
```

É importante observar o seguinte:

- a) Os comandos TKDOS devem sempre estar entre aspas. Os parâmetros opcionais podem ser incluídos, desde que fiquem dentro das aspas.
- b) Sempre que definirmos o control-D, é interessante que se documente esta ocorrência com um REM.
- c) Cada comando PRINT deve conter apenas um comando TKDOS.
- d) A instrução PRINT que contenha comando TKDOS e a instrução PRINT imediatamente anterior nunca deverão terminar com ponto e vírgula (;), nem com vírgula (,).
- e) Os seguintes comandos TKDOS só podem ser usados dentro de programas BASIC, ou seja, não são aceitos em modo direto: OPEN, APPEND, READ, WRITE e POSITION.
- f) 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
UNLOCK	SAVE	-	MAXFILES	PR#	IN#	INT

CHAIN (apenas no Integer BASIC)

### 3. 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 TK3000 //e 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 sequenciais 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.

#### 3.1 - ATIVAÇÃO DE ARQUIVOS SEQUENCIAIS

Para manipular um arquivo sequencial, você deverá utilizar o comando OPEN.

Seu formato é:

```
PRINT CHR$(4);"OPEN nome do arquivo [,Sn][,Dn][,Un]"
```

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

O exemplo abaixo ativa um arquivo com o nome LIVRO.

```
J200 D$ = CHR$(4) : REM CONTROL-D
J300 PRINT D$ "OPEN LIVRO"
J400 END
```

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

### 3.2 - DESATIVAÇÃO DE ARQUIVOS SEQUENCIAIS

O programa do item anterior contém um perigo potencial de destruição do arquivo e mesmo de todo conteúdo do disquete, porque o arquivo não foi "fechado" após seu uso. Para efetuar este fechamento, usa-se o comando CLOSE.

Este comando pode ser dado de duas formas:

a) PRINT CHR\$(4) "CLOSE [,Sn] [,Dn] [,Vn]"

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

b) PRINT CHR\$(4) "CLOSE LIVRO [,Sn] [,Dn] [,Vn]"

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

### 3.3 - GRAVANDO ARQUIVOS SEQUENCIAIS

O comando para gravação de dados em arquivos é WRITE.

O comando WRITE sempre deverá estar incluído num comando PRINT, precedido pelo caracter de controle CHR\$(4) (control-D):

```
PRINT CHR$(4); "WRITE nome do arquivo [,Bn]"
```

Após a execução do comando WRITE, todas as mensagens

que ocorrerem na tela de vídeo (PRINT) serão gravadas no arquivo.

A opção B indica o número do byte do arquivo onde deverá ser carregado o dado. Ela pode variar de 0 a 32767. Se for omitida, o dado é gravado após a última posição do arquivo acessada ou sobre o primeiro registro, caso se tenha usado o comando OPEN.

Como exemplo, inclua as seguintes linhas em nosso programa:

```
1305 PRINT D$"WRITE LIVRO"  
1310 PRINT "PRIMEIRA FRASE DO ARQUIVO"  
1390 PRINT D$"CLOSE LIVRO"
```

Assim ao listarmos (LIST), teremos:

```
1200 D$ = CHR$(4) : REM CONTROL-D  
1300 PRINT D$"OPEN LIVRO"  
1305 PRINT D$"WRITE LIVRO"  
1310 PRINT "PRIMEIRA FRASE DO ARQUIVO"  
1390 PRINT D$"CLOSE LIVRO"  
1400 END
```

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

Quando se grava um arquivo seqüencial com o mesmo nome de outro arquivo do disco, a nova gravação é feita sobre o arquivo anterior, podendo gerar efeitos desagradáveis. Como solução para esta situação deve ser usado o comando DELETE, que removerá o arquivo anterior do disco. Este comando deverá estar incorporado num comando PRINT, precedido de CONTROL-D.

Esta solução (o comando DELETE) só deve ser aplicada quando tivermos certeza da existência do arquivo no disquete, pois, caso contrário, o computador emitirá a mensagem de erro: NÃO HÁ ARQUIVO.

Para resolver este problema, é aconselhável garantir a existência do arquivo através de um comando OPEN a mais no início do programa (linhas 210 e 220). Com esta nova opção, nosso programa teria o seguinte aspecto:

```
1200 D$ = CHR$(4) : REM CONTROL-D
1210 PRINT D$"OPEN LIVRO"
1220 PRINT D$"DELETE LIVRO"
1300 PRINT D$"OPEN LIVRO"
1305 PRINT D$"WRITE LIVRO"
1310 PRINT "PRIMEIRA FRASE DO ARQUIVO"
1390 PRINT D$"CLOSE LIVRO"
1400 END
```

Pode-se utilizar variáveis para nomes de arquivo. Neste caso, o próprio programa nos perguntará o nome, como mostra o exemplo abaixo (linha 250):

```
1200 D$ = CHR$(4) : REM CONTROL-D
1205 INPUT "NOME DO ARQUIVO = "; A$
1210 PRINT D$ "OPEN "; A$
1220 PRINT D$ "DELETE "; A$
1300 PRINT D$ "OPEN "; A$
1305 PRINT D$ "WRITE "; A$
1310 PRINT "PRIMEIRA FRASE DO ARQUIVO"
1390 PRINT D$ "CLOSE "; A$
1400 END
```

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

```
1320 PRINT D$
1330 PRINT "VOCE QUER IMPRIMIR OUTRA FRASE NO
ARQUIVO?" : INPUT P$
1340 IF P$ (<) "S" THEN 390
1350 PRINT D$ "WRITE "; A$
1360 PRINT "SEGUNDA FRASE DO ARQUIVO"
```

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

Obs.: Existe a possibilidade de guardarmos números em arquivos sequenciais. Para isto devemos utilizar o comando PRINT, utilizando a seguir uma variável numérica, ou o número diretamente, observando que, apenas um valor deve ser escrito em cada declaração PRINT. Caso contrário, a inclusão de vírgulas, ou ponto e vírgula, gerará resultados diferentes do desejado.

### 3.4 - LENDO ARQUIVOS SEQUENCIAIS

Como já foi explicado anteriormente, a finalidade dos arquivos é a de armazenar dados para uso futuro. No item anterior foi visto como escrever em arquivos. Agora, o comando READ será usado para ler dados de um arquivo.

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

Sua sintaxe é:

```
PRINT CHR$(4) READ [,Bn]
```

B tem a mesma função que no comando WRITE.

O comando PRINT simples, após o comando WRITE, transfere dados ao arquivo, agora o comando INPUT após o comando READ, irá ler dados do arquivo e associá-los a variáveis em memória.

```

JNEW
J500 D$ = CHR$(4) : REM CONTROL-D
J510 INPUT "NOME DO ARQUIVO A SER LIDO: ";E$
J520 PRINT D$"OPEN ";E$
J530 PRINT D$"READ ";E$
J540 INPUT A$
J550 PRINT A$
J560 PRINT D$
J570 INPUT "VOCE QUER LER MAIS DADOS ? "; Q$
J580 IF Q$ = "S" THEN 530
J590 END

```

Através do programa acima note o seguinte:

- a) Para que a variável Q\$ na linha 570 não fosse associada a nenhum dado do arquivo, tivemos de anular o comando READ na linha 530. Se quiséssemos continuar a ler o conteúdo no arquivo, teríamos de retornar à linha 530 com um novo comando READ.
- b) Se quiséssemos ler mais dados do que os existentes no arquivo, ocorreria a mensagem FIM DE ARQUIVO, seguida de PAROU EM 540, interrompendo-se o programa.
- c) No fim do programa não constou a instrução CLOSE E\$. No caso de apenas leitura de um arquivo não é necessário que o desativemos após a leitura, pois neste caso nada foi gravado e nenhuma informação necessita ser atualizada no disquete. No entanto, é recomendável fechar o arquivo para que o buffer utilizado por ele seja liberado.

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

- Usar o comando ONERR GOTO, explicado em detalhes no Manual de Operação do TK3000 //e.
- Introduzir no final do arquivo uma palavra chave

indicando fim. O programa ao ler esta palavra deverá identificá-la, e anular o comando READ, continuando então o processamento sem nenhuma mensagem de erro.

Uma forma de indicar fim de arquivo é inserir um caractere neutro, exemplo: CHR\$(0), usando um comando PRINT antes de fechar o arquivo.

Obs.: Da mesma forma que utilizamos a declaração INPUT, para ler dados do arquivo, poderemos também utilizar a declaração GET. Porém, 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 único inconveniente do uso do comando GET, é que sempre que for usado, o primeiro caractere do próximo comando PRINT será ignorado. Assim, a única solução é que sempre no comando PRINT, seguinte ao GET, comece com um caractere de "espaço", evitando assim perdas de dados.

### 3.5 - ADICIONANDO DADOS EM ARQUIVOS SEQUENCIAIS

#### 3.5.1 - O COMANDO APPEND

Ao se trabalhar com um arquivo sequencial, após este ser fechado, é perdido o controle do bloco em que o último dado foi gravado. Lembre-se também que, como foi visto no item III.3, se abrimos um arquivo, os novos dados serão escritos em cima dos anteriores, estragando o arquivo anterior.

Assim sendo, sempre que quisermos escrever num arquivo já existente, devemos primeiramente localizar o final dele de modo ser possível aumentá-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 não usado.

Outra diferença importante é que o comando APPEND necessita de um arquivo que já existe, caso contrário emitirá a mensagem NAO HA ARQUIVO.

O formato do comando APPEND é o mesmo que o do comando OPEN ou seja, deve ser incluído em um comando PRINT, além de estar precedido do caracter de controle control-D:

```
PRINT CHR$(4);"APPEND nome do arquivo [,Sn][,Dn][,Vn]"
```

### 3.5.2 - 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 só avança o apontador do número 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, será emitida a mensagem de erro FIM DE ARQUIVO.

Sua sintaxe é:

```
PRINT CHR$(4);"POSITION nome do arquivo,Rn"
```

O parâmetro R é obrigatório. O valor de n deve estar entre 0 e 32767.

Um exemplo deste comando seria:

```
1500 PRINT D$ "POSITION ARQUIVO, R10"
```

No exemplo acima o apontador seria deslocado 10 regis-

tros a partir da posição atual contando 10 caracteres return. Se usássemos R0, o apontador não se moveria.

Note que:

- a) O comando de localização do apontador deve ser sempre utilizado antes do que qualquer comando de leitura, ou de escrita; caso contrário este comando os cancelaria.
- b) Antes de utilizarmos o comando POSITION devemos ativá-lo através do comando OPEN que localizará o apontador no início, deixando inicialmente a área toda do arquivo à disposição do POSITION. Note também que se o arquivo fosse ativado com um comando APPEND, o comando POSITION perderia a validade em alguns casos já que este não movimenta o apontador para trás.

#### 4. ARQUIVOS DE ACESSO RANDÔMICO (ALEATÓRIO)

Os arquivos de acesso randômico têm um comportamento muito mais flexível que os de acesso sequencial. A principal diferença é que neste caso não há 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 independentemente de sua posição.

O arquivo de acesso randômico, assim como o arquivo sequencial, é composto por vários grupos de dados também chamados registros.

O arquivo de acesso randômico, inicialmente é formado apenas por um registro, mas quando o aumento de informações ultrapassa o comprimento definido do registro automaticamente é formado um outro registro cuja identificação se dará por um valor inteiro maior em uma unidade que o registro anterior. Assim o primeiro registro do arquivo receberá o número 0, o segundo receberá o número um, o terceiro receberá dois, etc.

Deve-se observar que os arquivos do tipo randômico tendem a ocupar mais espaço no disco, do que um arquivo sequencial.

##### 4.1 - ATIVAÇÃO DE ARQUIVOS DE ACESSO RANDÔMICO

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

```
110 LET D$ = CHR$(4) : REM CONTROL-D
120 PRINT D$ "OPEN ARQ1,L20"
```

O parâmetro L deve variar entre 1 e 32767. É importante que este parâmetro seja bem calculado, pois se for inserido algum registro (incluindo vírgulas e o car-

riage return) de tamanho maior que o especificado, ocorrerão consequências imprevisíveis no armazenamento das informações.

Se, ao ler um arquivo, você especificar um parâmetro (L) diferente daquele que foi utilizado para gravá-lo, o TKDOS irá utilizá-lo cegamente para calcular as posições dos registros e os dados lidos não corresponderão aos originalmente gravados. Portanto é comum utilizar o registro 0 para armazenar o valor do parâmetro L e outros dados referentes ao arquivo.

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

#### 4.2 - LENDO E ESCRREVENDO EM ARQUIVOS DE ACESSO RANDÔMICO

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.

```
      J30 PRINT D$ "READ ARQ1, R3"  
ou  
      J30 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:

## JREAD 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:

```
J10 REM CRIA LISTA DE NOMES
J20 D$ = CHR$(4) : REM CONTROL-D
J30 INPUT "NOME: ";N0$
J40 INPUT "END.: ";EN$ : REM COM OS DADOS
J50 INPUT "TEL.: ";TE$
J60 PRINT D$"OPEN LISTA, L150"
J70 PRINT D$"WRITE LISTA, R0"
J80 PRINT N0$ : PRINT EN$ : PRINT TE$
J90 PRINT D$"CLOSE LISTA"
```

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

```
J10 REM LE OS DADOS DA LISTA
J20 D$ = CHR$(4) : REM CONTROL-D
J30 PRINT D$"OPEN LISTA, L150"
J40 PRINT D$"READ LISTA, R0"
J50 INPUT N0$, EN$, TE$
J55 PRINT : REM PULA UMA LINHA
J60 PRINT N0$
J70 PRINT EN$
J80 PRINT TE$
J90 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.

## 5. COMANDOS AUXILIARES TKDOS 3.3

### 5.1 - COMANDO EXEC

O poderoso comando EXEC é muito semelhante ao comando RUN, e é relacionado a arquivos sequenciais que, ao invés de simples informações ou dados, contém 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.

Sua sintaxe é:

```
EXEC nome do arquivo [ ,Rn] [ ,Sn] [ ,Dn] [ ,Vn]
```

R indica a partir de qual campo do arquivo será executado o programa.

A sequência abaixo explicará 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:

```
JNEW  
J10 REM PROGRAMA TESTE  
J30 HOME  
J40 FOR I = 1 TO 100  
J50 K = I ^ 2  
J60 PRINT I,K  
J70 NEXT I  
J80 FOR L = 1 TO 1000 : NEXT L  
J90 END
```

## JSAVE TESTE

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

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

Programa AUTOMATO:

```
JNEW
J10 REM PROGRAMA CRIA AUTOMATO
J20 D$ = CHR$(4) : REM CONTROL-D
J30 PRINT D$ "OPEN AUTOMATO"
J40 PRINT D$ "WRITE AUTOMATO"
J45 PRINT "NEW" <-----
J50 PRINT "LOAD TESTE" !Sequência de
J55 PRINT "HOME" !instruções
J60 PRINT "LIST" !que
J70 PRINT "100 FOR I = 1 TO 10000"!serão
J80 PRINT "120 NEXT I : END" !usadas
J90 PRINT "RUN 100" !pelo
J100 PRINT "CATALOG" !EXEC
J105 PRINT "RUN" <-----
J110 PRINT D$ "CLOSE AUTOMATO"
JSAVE CRIA AUTOMATO
```

Digitando-se então:

```
IRUN CRIA AUTOMATO
```

o arquivo texto AUTOMATO será criado em seu disquete.

Para fazer funcionar o comando EXEC, basta digitar:

```
JEXEC AUTOMATO
```

## 5.2. - MAXFILES

No modo normal, o TK3000 //e ativa até 3 arquivos simultaneamente. Muitas vezes esta quantidade não é 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 é:

```
PRINT D$ "MAXFILES 8"
```

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

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

## 5.3. - MON E NOMON

MON e NOMON são importantes ferramentas do TKDOS, que permitirão depurar programas, isto é, 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 computa-

dor. 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 PR#C,Sn] (ativa o disk drive, carregando novamente o TKDOS) e pelo comando RESET. Por exemplo:

```
IMON C,I,O
```

#### 5.4. - USANDO O COMANDO TRACE COM O TKDOS

O comando BASIC TRACE é uma ferramenta muito útil na depuração de programas BASIC. Quando um programa contém comandos TKDOS e o TRACE é ativado, os comandos TKDOS no interior do programa não funcionam, já que TRACE imprime o número da linha antes do comando TKDOS. Uma solução para este problema é 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 deterá aguardando uma entrada pelo teclado.

## 6. ARQUIVOS EM LINGUAGEM DE MÁQUINA

O TK3000 //e permite além do que já foi apresentado, que se possa armazenar em disquete regiões de memória definidas. Contida nesta parte da memória podem existir programas em linguagem de máquina, telas de alta resolução ou qualquer outra informação.

### 6.1. - BSAVE

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

BSAVE nome do arquivo, An, Ln [,Sn] [,Dn] [,Vn]

Onde o parâmetro A indica o endereço inicial da porção da memória a ser armazenada no disco; este endereço, se precedido do sinal dólar (\$) significa o endereço em hexadecimal, caso contrário, 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 numérico do comprimento vier precedido do sinal (\$), então indicará que o parâmetro está apresentado em hexadecimal.

Exemplo:

```
BSAVE VIDE01, A$2000,L$2000
```

### 6.2. - BLOAD

O comando BLOAD carrega o conteúdo de um arquivo do disco para a memória de seu TK3000 //e. Os programas BASIC não serão afetados com o carregamento, a menos que os dados sejam carregados na área do BASIC do TK3000 //e.

O formato do comando BLOAD é:

BLOAD nome do arquivo [,An] [,Sn] [,Dn] [,Vn]

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

JBLOAD UTIL, A\$300

### 6.3. - BRUN

O comando BRUN segue exatamente o comportamento de BLOAD, explicado no item anterior, somente que, após ter carregado o programa em linguagem de máquina este faz com que o computador inicie sua execução.

O formato do comando BRUN é:

BRUN [,An] [,Sn] [,Dn] [,Vn]

Note que o formato de BRUN obedecerá 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.

## APÊNDICE 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 NAO DISPONIVEL

Código = 1

O TKDOS sempre emitirá esta mensagem de erro quando houver a tentativa de processar um programa produzido numa linguagem não carregada na memória.

### ERRO DE INDICE

Código = 2 ou 3

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

Por exemplo:

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

Sendo que o maior valor permitido para o parâmetro Volume é 254.

### DISCO PROTEGIDO

Código = 4

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

## FIM DE ARQUIVO

Código = 5

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

## NAO HA ARQUIVO

Código = 6

Ocorre quando é tentado acessar algum arquivo inexistente no disco.

## VOLUME ILEGAL

Código = 7

Ocorre quando é 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 deste não está 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.

## TRAVADO

Código = 10

Ocorre quando é tentado escrever em um arquivo ou alterar um programa que foi protegido através da instrução LOCK. Neste caso, a única saída é 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:

```
J15 PRINT CHR$(4)"OPEM ARQUIVO"
```

## SEM BUFFER DISPONIVEL

Código = 12

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

## TIPO INCOMPATIVEL

Código = 13

Ocorre quando há a tentativa de carregar um programa utilizando um comando incompatível com a classe de arquivo. Por exemplo, se tentarmos carregar o programa em BASIC "TESTE" com o comando destinado à leitura de programa em linguagem de máquina.

Também ocorre quando há a tentativa de ler um arquivo "A" a partir do Integer BASIC ou um arquivo "I" a partir do Applesoft.

```
JBLOAD TESTE (return)
```

## EXCEDE MEMORIA

Código = 14

Acontece ao se carregar um programa que necessite uma memória superior à disponível no seu TK3000 //e. Geralmente ocorre quando a capacidade disponível 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.

## APÊNDICE B - DESCRIÇÃO DOS PROGRAMAS

### TKFID

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

#### JBRUN TKFID

##### 1 - Funções:

O TKFID possui as seguintes funções:

- <1> - COPIA - Copia arquivos
- <2> - CATALOG - Cataloga todos os arquivos do disquete
- <3> - ESPACO - Fornece o número de bytes ocupados e os disponíveis no disquete.
- <4> - UNLOCK - Destrava o arquivo
- <5> - LOCK - Trava o arquivo
- <6> - DELETE - Apaga o arquivo
- <7> - ANULA SLOT & DRIVE - Anula o último slot e drive utilizados
- <8> - VERIFICAR ARQUIVO - Verifica a integridade do arquivo gravado no disco
- <9> - 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 contêm 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:

Quando o sistema pergunta se você deseja ser consultado, responda:

S - Assume controle manual. Após cada nome de arquivo deve-se digitar 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:

```
Jxxx PRINT CHR$(4) "BLOAD CHAIN,A520"  
Jxxx 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 Sistema Master para seu disquete de programas, através do utilitário TKFID.
- 4 - No Integer BASIC já existe este comando DOS.

## CRIA TEXTO

Este utilitário cria arquivos-texto seqüenciais.

Para executá-lo, digite:

```
IRUN 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 ler arquivos-texto gravados pelo programa CRIA-TEXTO.

Para executá-lo, digite:

```
IRUN LE TEXTO (return)
```

Em seguida, introduza o nome do arquivo-texto desejado, e pressione a tecla return.

## RAM COPIA

Utilitário duplicador de disquetes, desenvolvido

especialmente para o TK3000 //e. A diferença básica do RAM COPIA para copiadores de disquetes (COPIA) disponíveis em outros Sistemas Master é a utilização dos bancos de memória do cartão TK Works para armazenar o disco original, aumentando a velocidade de duplicação.

O RAM COPIA pode copiar disquetes em DOS 3.3, CP/M, Pascal 1.1 ou PRODOS.

Obs.: 1) O TKSPEEDOS desativa o RAM COPIA.

2) Se você não dispuser de um TKWorks, a operação será feita normalmente.

Para executá-lo digite:

```
JBRUN RAM COPIA (return)
```

Será apresentado um menu de opções que permite escolher o slot e o drive que contém o original e, a seguir, o slot e o drive que contém a cópia.

O slot 3 não pode ser envolvido em nenhum tipo de operação (ORIGINAL/COPIA). Ao pressionar a tecla esc, o programa será imediatamente interrompido.

Nota: Verifique previamente se o disquete que receberá a cópia não possui arquivos ou informações que não possam ser apagados, pois o disco será novamente inicializado.

## LISTA TELEFONICA

Instruções:

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

Para executá-lo digite:

```
JRUN LISTA TELEFONICA (return)
```

O programa apresenta 7 opções:

- 1 - CONSULTA DE REGISTROS
- 2 - LISTAGEM DOS REGISTROS
- 3 - ENTRADA DE NOVOS REGISTROS
- 4 - EXCLUSAO DE REGISTROS
- 5 - ALTERAÇÃO DE REGISTROS
- 6 - IMPRESSORA LIGADA/DESLIGADA
- 7 - FINAL DO PROGRAMA

#### TKSPEEDOS

Este programa aumenta a velocidade do DOS até 40 vezes, agilizando a leitura e a gravação de dados.

Para obter este efeito, digite:

```
JBRUN TKSPEEDOS (return)
```

Quando o cursor reaparece na tela, você poderá digitar qualquer comando DOS e rodar qualquer programa que não realize um boot do disquete, obtendo as vantagens da velocidade maior.

Nota: O comando INIT é desativado quando se usa o TKSPEEDOS. Este programa também desativa o programa RAM COPIA.

#### TKPROG

Este programa permite a você gravar ou carregar a programação de teclado do TK3000 //e.

Digite:

```
JRUN TKPROG (return)
```

Você verá um menu com três opções:

- 1 - CARREGA PROGRAMACAO DE TECLADO
- 2 - SALVA PROGRAMACAO RESIDENTE
- 3 - SAIDA PARA O BASIC

A primeira opção carrega alguma programação do teclado que você já tenha feito. Ao ser selecionada, mostra o diretório e pergunta qual o nome do arquivo a ser carregado. A rigor, qualquer arquivo poderia ser carregado, porém, se você carregar um arquivo que não seja uma programação de teclado, a programação do teclado ficará sem significado, e o teclado pode travar, perdendo sua função.

A segunda opção permite que você grave a programação do teclado. Ao ser selecionada, apresenta o catálogo e pede que você digite o nome do arquivo a ser gravado. Sugerimos que você use um nome que lembre programação de teclado. Por exemplo: TECLADO, PROGTEC1.

A opção 3 permite voltar ao BASIC.

#### XADREZ e INVASORES

Estes são dois jogos.

Digite:

```
JRUN XADREZ      (return)
  ou
JRUN INVASORES  (return)
```

Cada um dos jogos contém uma tela com a correspondente explicação.

## APÊNDICE C - MÓDULO DE REFERÊNCIA

Neste apêndice 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		

- Comando de acesso:

IN# PR#

- Comandos de arquivos-texto seqüenciais:

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
-------	------	-------

### NOTAÇÃO 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:

```
J10 D$ = CHR$(4):REM CHR$(4) = CONTROL-D
J20 PRINT D$; "CATALOG"
```

#### PARÂMETROS 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	,Vn	V0*	V254
Slot	,Sn	S1	S7
Drive	,Dn	D1	D2

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

Arquivos Texto Seqüenciais:

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 é 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 [ ,Sn] [ ,Dn] [ ,Vn]

Inicializa um disquete virgem para o formato de disco auxiliar. Assume como programa apresentador o programa de nome X e número de volume n (se especificado).

Fica desativado ao se acionar o programa TKSPEEDOS.

CATALOG [T] [L] [ ,Sn] [ ,Dn] [ ,Vn]

Apresenta o número de setores disponíveis 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.

As opções T e L são, respectivamente, a letra indicativa do tipo do arquivo e a primeira letra do nome do arquivo.

TIPO      DESCRIÇÃO

I	INTEGER BASIC
A	Programa BASIC tipo APPLESOFT
T	Arquivo-texto
B	Programa em linguagem de máquina

SAVE X [ ,Sn] [ ,Dn] [ ,Vn]

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 [ ,Sn] [ ,Dn] [ ,Vn]

Carrega para a memória um programa BASIC contido no disco, depois de limpá-la do programa anterior.

RUN X [,Sn] [,Dn] [,Vn]

Carrega o programa BASIC com o nome X executando o mesmo a seguir.

RENAME X,Y [,Sn] [,Dn] [,Vn]

Troca o nome do arquivo X para Y.

DELETE X [,Sn] [,Dn] [,Vn]

Apaga o arquivo X do disquete.

LOCK X [,Sn] [,Dn] [,Vn]

Trava o arquivo X contra uma mudança ou apagamento acidental. Este travamento pode ser observado no CATALOG através do \*.

UNLOCK X [,Sn] [,Dn] [,Vn]

Destrava o arquivo X (caso este já estivesse travado) permitindo mudança ou deleção do mesmo.

VERIFY X [,Sn] [,Dn] [,Vn]

Checa o arquivo X em sua consistência interna. Se X não possuir nenhum erro, não será fornecida 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

PR#n  
IN#n

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

COMANDOS DE ARQUIVOS-TEXTO SEQUENCIAIS

OPEN X [S] [D] [V]

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] [S] [D]

Completa o comando WRITE X, se necessário, e desatribui o buffer reservado para o arquivo-texto X. Sem o nome do arquivo, este comando irá 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 é 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 é delimitado pelo caracter RETURN. Este comando é cancelado por qualquer comando TKDOS posterior a ele.

#### APPEND X [,Sn] [,Dn] [,Vn]

Abre um arquivo-texto X já existente, este comando é 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 através do comando OPEN, permite deslocar o indicador de arquivo até uma região indicada pelo parâmetro p.

#### EXEC X [,Rp] [,Sn] [,Dn] [,Vn]

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 ALEATÓRIOS

OPEN X, Lj [,Sn] [,Dn] [,Vn]

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 comando WRITE ou READ do início do campo 0. O mesmo comprimento de campo deve ser atribuído a cada vez que se abre o mesmo arquivo.

CLOSE [X] [,Sn] [,Dn] [,Vn]

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

WRITE X [,Rr] [,Bb]

Permite que a instrução PRINT escreva informação no arquivo X. Quando não é apresentado nenhum parâmetro, o comando inicia diretamente na posição do apontador de arquivo. Com apenas o parâmetro Rr, o comando inicia no byte 0 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 é 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 aleatório X. A identificação da

posição do início de leitura tem o mesmo comportamento do comando WRITE, descrito acima.

#### COMANDOS DE ARQUIVOS BINÁRIOS

BSAVE X, Aa, Lj [,Sn] [,Dn] [,Vn]

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

BLOAD X [,Aa] [,Sn] [,Dn] [,Vn]

Carrega o arquivo binário X no mesmo endereço de memória no qual ele foi originalmente salvo ou (se especificado) no endereço a.

BRUN X [,Aa] [,Sn] [,Dn] [,Vn]

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

- X [,Aa] [,Sn] [,Dn] [,Vn]

Executa um arquivo independentemente de seu tipo.

