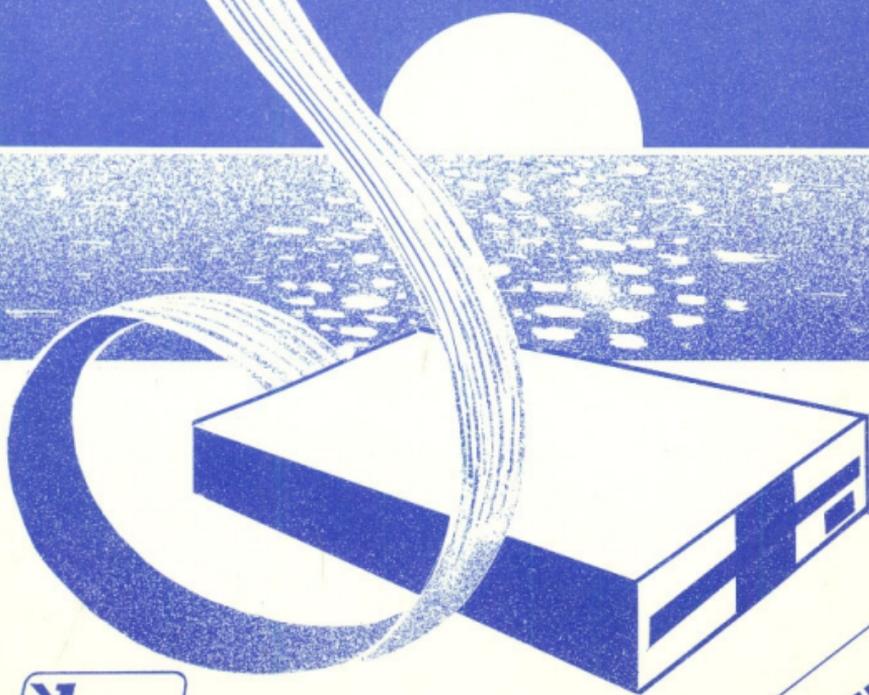


Sistema de Disco para MSX



SOLX-DOS e BASIC de DISCO



**SISTEMA
DE
DISCO
PARA
MSX**



**SOLX-DOS
E
BASIC DE DISCO**



**Microsol
Tecnologia**

Renato da Silva Oliveira
e
Rubens Pereira Silva Jr.

SISTEMA DE DISCO PARA MSX

SOLX-DOS E BASIC DE DISCO

2ª EDIÇÃO

MAIO - 1989



Microsol
Tecnologia

COORDENAÇÃO EDITORIAL: Pierluigi Piazzi
EDITORIAÇÃO: Renato da Silva Oliveira
PRODUÇÃO EDITORIAL: Betty F. Piazzi
REVISÃO TÉCNICA: MICROSOL TECNOLOGIA
ARTE: Ana Lúcia Antico
ILUSTRAÇÕES: D. Odilon Nicoletti
CAPA: Mauro Arghinenti



ALEPH Publicações e
Assessoria Pedagógica Ltda
Av. Dr. Luiz Migliano, 1110
Cj. 301 303 - 05711 - S. Paulo
Cx. Postal: 20 707 - 01498 - SP



(011) 843 3202

Dados de Catalogação na Publicação (CIP) Internacional
(Câmara Brasileira do Livro, SP, Brasil)

051s Oliveira, Renato da Silva, 1960-
Sistema de disco para MSX : SOLX-DOS e Basic de
disco / Renato da Silva Oliveira, Rubens Pereira da
Silva Jr. -- São Paulo : Aleph, 1987.

1. Microcomputadores 2. Microcomputadores (Programação) 3. MSX (Computadores) 4. SOLX-DOS (Sistema operacional de computador) I. Silva Júnior, Rubens Pereira, 1961- II. Título.

CDD-001.6425
-001.64
-001.642

87-0264

Índices para catálogo sistemático:

1. MSX : Computadores : Processamento de dados 001.64
2. Microcomputadores : Processamento de dados 001.64
3. Microcomputadores : Programação : Processamento de dados 001.642
4. SOLX-DOS : Sistema operacional : Computadores : Processamento de dados 001.6425

SUMÁRIO



NOTA DO EDITOR	6
Capítulo 1	
CONCEITOS BÁSICOS	7
Capítulo 2	
COMO INSTALAR E TESTAR O EQUIPAMENTO	17
Capítulo 3	
O BASIC DE DISCO	24
Capítulo 4	
O SOLX-DOS	38
Capítulo 5	
DICIONÁRIO DE COMANDOS E FUNÇÕES	57
APÊNDICES	100
Apêndice A — CONVERSÃO DE ARQUIVOS	101
Apêndice B — CARACTERÍSTICAS TÉCNICAS	104
Apêndice C — CÓPIA DO SOLX-DOS	107
Apêndice D — MENSAGEM DE ERRO	108

NOTA DO EDITOR



Quando a EDITORA ALEPH foi procurada pela MICROSOL para produzir este livro, nos sentimos duplamente orgulhosos.

Em primeiro lugar é sempre gratificante ver um trabalho editorial como o que realizamos ser reconhecido por sua seriedade e qualidade a ponto de gerar uma encomenda de responsabilidade como esta.

O segundo motivo está mais relacionado com nossa filosofia de trabalho: a editora ALEPH nunca publicou traduções, pois sempre acreditou que os autores brasileiros têm, no mínimo, o mesmo grau de inteligência e competência dos "importados".

Foi também muito gratificante, portanto, trabalhar em conjunto com um fabricante genuinamente brasileiro, que acredita na "criatividade nacional", concebendo soluções nossas para nossos problemas específicos e sem medo de trilhar novos caminhos, nunca tentados "lá fora".

CONCEITOS BÁSICOS



O QUE É O DRIVE

Todo computador, para se tornar útil, necessita receber informações de algum periférico para poder processá-las e apresentar algum resultado ao usuário.

Dentre todos os periféricos, o drive é um dos mais confiáveis, versáteis e rápidos que existem atualmente, sendo indispensável para aqueles que queiram utilizar o computador em aplicações mais profissionais.

Podemos fazer uma analogia do drive com um toca-discos: seu objetivo é armazenar, em pequenos discos magnéticos, informações para uso do computador, de forma rápida e eficaz.

Para se conseguir armazenar muitas informações em um disquete através do drive, foram utilizadas as mais modernas técnicas da mecânica e eletrônica, tornando-o assim um equipamento delicado, exigindo alguns cuidados ao manuseá-lo!

- 1 — **NÃO** o exponha a calor ou umidade excessivas;
- 2 — **EVITE** pancadas;
- 3 — **NUNCA** insira nele nada além de disquetes ou cartões protetores;
- 4 — **CUIDADO** ao manuseá-lo!



*Cuide bem do
seu disk drive!
Ele é um
equipamento
s sofisticado
e sensível!*

Nos toca-discos dos equipamentos de som existe uma "agulha" que recebe as informações do disco. Nos drives existe uma "cabeça de leitura/gravação" com a mesma finalidade.

Alguns drives mais modernos possuem duas cabeças de gravação, sendo possível utilizar as duas faces dos disquetes.

Os drives que possuem apenas uma cabeça são os de face simples e os de duas cabeças são os de face dupla.

Desde o tempo em que foi inventado o primeiro drive, várias inovações foram feitas, principalmente para aumentar a capacidade de armazenar dados e a velocidade de acesso a eles.

Quanto mais moderno o drive, mais informações ele consegue armazenar em um menor espaço do disquete. Hoje em dia, o drive mais usado para microcomputador é aquele que utiliza disquetes de 5¼" (cinco polegadas e um quarto), mas existem drives para disquetes de 8" (oito polegadas), 3½" (três polegadas e meia) e até de 3" (três polegadas), face simples ou dupla!

O QUE É DISQUETE

O drive pode gravar ou ler informações nos disquetes.

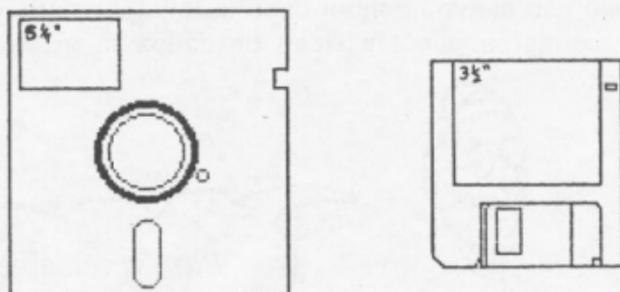
Estes disquetes são feitos do mesmo material utilizado nas

fitas cassete, sendo portanto extremamente sensíveis a campos magnéticos intensos, poeira, gordura dos dedos etc.

Há casos em que uma simples viagem de ônibus elétrico é suficiente para impedir a leitura de um disquete.

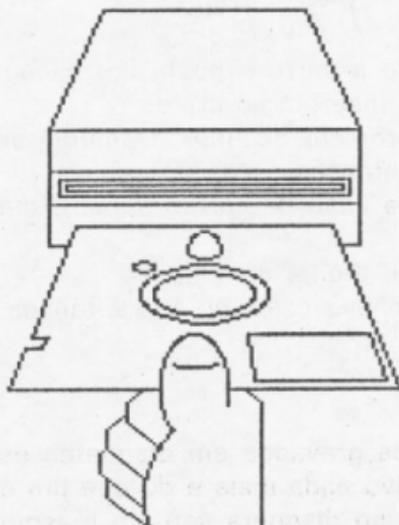
Devido a isso, eles vem envoltos numa capa plástica protetora, ficando expostas apenas algumas regiões para que o drive possa acessá-las (figura 1.1.).

Figura 1.1 — Disquetes



Devido à necessidade de perfeita sincronização entre o drive e o disquete, este só pode ser inserido no drive numa única posição. Caso contrário, corre-se o risco de danificar tanto o disquete como o próprio drive. Não tente, portanto, inserir o disquete invertido no drive (figura 1.2).

Figura 1.2 — Inserindo o disquete no drive



Da mesma maneira que existem drives com uma ou duas cabeças de leitura, existem disquetes com uma ou duas faces preparadas para utilização. Se o seu drive é face dupla, você deverá utilizar disquetes de face dupla também! Os disquetes, ainda nas fábricas, são testados para garantir um perfeito funcionamento. Porém, os disquetes de face simples tem apenas uma face garantida para uso. Se o seu drive tem duas cabeças e tentar utilizar uma face não preparada de um disquete de face simples, poderão ocorrer erros de leitura e gravação.

Embora isso não cause nenhum dano ao equipamento, o uso incorreto dos disquetes põe em risco os dados lá gravados!!!



Para que seus disquetes durem muito tempo, siga as seguintes regras:

- 1 — **NUNCA** toque a parte exposta do disquete;
- 2 — **CUIDADO** ao inserí-lo no drive;
- 3 — **NUNCA** o aproxime de ímãs, transformadores ou campos magnéticos intensos;
- 4 — **GUARDE-O** na capa de cartão sempre que estiver fora de uso;
- 5 — **NÃO** o dobre, molhe ou amasse;
- 6 — **NÃO** o exponha a calor ou frio intensos.

ARQUIVOS

Todos os dados gravados em disquetes estão na forma de arquivos. Um arquivo nada mais é do que um conjunto de informações agrupadas no disquete sob um mesmo nome!

Por exemplo, uma relação de nomes com seus respectivos endereços e telefones pode estar gravada num disquete sob o nome "AGENDA". Um programa em BASIC pode estar gravado como "PROG1.BAS" e um programa em Linguagem de Máquina pode estar gravado como "UTILIT.BIN". Os três são arquivos, porém cada um tem certas peculiaridades.

Costuma-se chamar de programa a um conjunto de dados que tem significado para o computador e não para nós. No exemplo do parágrafo anterior, "PROG1.BAS" e "UTILIT.BIN" são conjuntos de dados que podem ser perfeitamente entendidos pelo micro. Nós, entretanto, se pudéssemos ver o que esses arquivos contém, veríamos apenas uma seqüência aparentemente desordenada de caracteres. Mesmo assim, os programas estão gravados em disquete da mesma maneira que estão os demais arquivos.

Os dados dos arquivos são gravados em partes do disquete. Quando uma parte é completamente preenchida, uma nova parte é utilizada para dar continuidade à gravação dos dados.

Toda e qualquer informação que você quiser gravar em disquete deverá ser feita em forma de arquivos.

Para tanto algumas normas precisam ser obedecidas:

- 1 — É preciso dar um nome ao arquivo;
- 2 — É conveniente especificar o nome do drive, se você estiver utilizando mais de um deles.



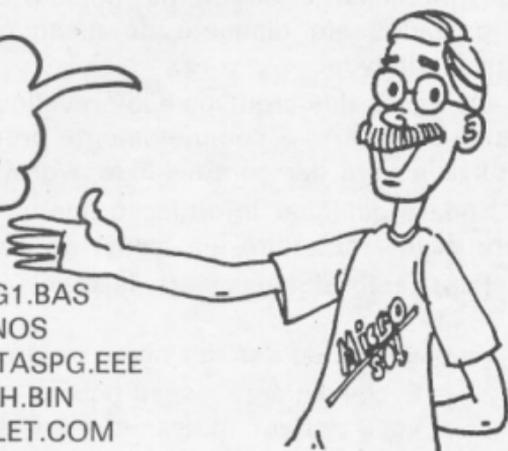
O nome do arquivo pode conter até oito caracteres, um ponto (que não é obrigatório) e mais três outros caracteres (também não obrigatórios). O primeiro caractere do nome de um arquivo deve ser necessariamente uma letra. É indiferente o uso de letras minúsculas ou maiúsculas. Por exemplo, os nomes "CDX2.UTI", "cdx2.ut i" e "CdX2.Ut i" são considerados idênticos pelo drive!

As três letras que podem vir depois do ponto no nome de um arquivo são chamadas "extensão do nome" e devido ao uso freqüente, algumas delas são consideradas "padrões":

EXTENSÃO	SIGNIFICADO
BAS	— programas-fonte em BASIC;
COB	— programas-fonte em COBOL;
ASM	— programas-fonte em ASSEMBLY;
TXT	— textos diversos;
DAT	— dados diversos;
COM	— programa executável sob SOLX-DOS;
BIN	— programas em Linguagem de Máquina;
etc...	

*Estes são alguns
nomes válidos
para arquivos em
disquetes!*

CAD324	PROG1.BAS
TESTE1.BAS	ALUNOS
MEUARQ	CONTASPG.EEE
MICROSOL.TXT	ALEPH.BIN
	TIPOLET.COM



Acostume-se a dar uma extensão ao nome dos seus arquivos para facilitar sua identificação posteriormente!

Após escolher um nome apropriado, pode-se indicar o drive no qual será lido ou gravado seu arquivo. Use "A:" para o drive A e "B:" para o drive B.

Por exemplo, para acessar o arquivo "DADOS.TXT" que está no drive B, deveremos dar ao computador o drive e o nome do arquivo da seguinte maneira:

B:DADOS.TXT

Note que vem primeiro o nome do drive (B) seguido de dois pontos (:), o nome do arquivo (DADOS) e sua extensão se houver (.TXT).

Nos capítulos seguintes saberemos como ler e gravar programas e dados num disquete, mas por enquanto basta saber como dar nome a um arquivo e como indicar para o computador onde ele está.

DIRETÓRIO

Após a formatação, o disquete fica subdividido em trilhas e setores, o que torna muito difícil para nós humanos saber o quanto já utilizamos de seu espaço livre.

Para facilitar um pouco as coisas podemos imaginar o disquete como se fosse uma seqüência de "caixinhas" numeradas de zero até um certo número. Cada "caixinha" consegue armazenar uma certa quantidade de informações.

Quando o computador precisa gravar algo no disquete ele procura a primeira "caixinha" que estiver livre, gravando lá o que precisa e indicando em algum lugar que ela já não está mais livre. Quando o computador precisar apagar os dados de uma "caixinha" ele apenas indica que ela está disponível e que seu conteúdo já não é mais útil.

Cada tipo de drive (e também de disquete) permite uma certa formatação, isto é, uma certa divisão em "caixinhas". Como o computador tem livre acesso a todas essas "caixinhas" fica óbvia a vantagem do drive sobre o gravador cassete ou cartuchos, pois consegue ler e gravar (o cartucho só permite ler) os dados como bem entender (o gravador só permite ler ou gravar dados em seqüência).

Cada disquete, após a formatação, tem um número definido dessas "caixinhas", sendo esse número dependente do tipo de drive utilizado.

Para que o computador possa gerenciar essas "caixinhas" (que de agora em diante vamos chamar de BLOCOS) sem se atrapalhar ou se confundir, ele mantém uma área do disquete com todos os dados necessários para acessar qualquer informação válida gravada. Essa área chama-se "diretório".

No diretório estão os nomes de todos os arquivos gravados no disquete, a data (e em alguns casos a hora também) em que cada arquivo foi atualizado pela última vez, além do seu tamanho. Muitos outros dados estão gravados no diretório, mas esses dados só servem para o computador "se achar" no disquete, não sendo de nenhuma utilidade para nós.

Existem comandos próprios no computador que nos permitem saber o que está gravado no disquete simplesmente nos mostrando seu diretório. Esses comandos serão vistos em outros capítulos.

Podemos imaginar o diretório como sendo um "índice" daquilo que está gravado no disquete.

Nos micros MSX o diretório consegue manter informações de até 112 arquivos gravados no disquete, o que é mais do que suficiente para a maioria das aplicações!

*O diretório é
como um sumário
de todos os arquivos
gravados em um
disquete!*



FORMATAÇÃO DE DISQUETES

Para que o computador possa utilizar o disquete que está dentro do drive ele precisa de algumas informações que devem ser previamente gravadas.

Quando compramos um disquete virgem, não existe nada gravado nele, nem as tais informações. Sendo assim, sempre que quisermos utilizar um disquete virgem, teremos que formatá-lo, ou seja, prepará-lo para uso.

A formatação consiste simplesmente em um preparo do disquete, onde são feitas algumas verificações para garantir que não há partes defeituosas além de serem gravados alguns dados de controle para o computador.

Como existem drives de face simples e de face dupla precisamos informar ao computador que tipo de drive ele está utilizando. Após a formatação o disquete estará pronto para uso, não sendo mais necessárias novas formatações.

Quando o computador vai acessar um disquete já formatado ele fica sabendo (pelas informações de controle lá gravadas) com que tipo de drive ele está trabalhando.

Formatar um disquete nada mais é do que prepará-lo para uso, sendo necessário pelo menos uma vez.

Cuidado apenas para não formatar um disquete que já contenha algumas informações úteis, pois após a formatação os disquetes ficam completamente limpos, com apenas os dados de controle!

Mais adiante, veremos como proceder para formatar um disquete.

*Sem formatar um disquete
ao menos uma vez, o
micro não consegue
utilizá-lo!*

*Após a formatação,
o disquete estará pronto
para ser usado, com
todo o seu espaço
disponível já
preparado!*



BACKUP

Um disquete pode conter muitos arquivos.

Devido ao risco inevitável de acidentes e à fragilidade dos disquetes, é aconselhável que tenhamos ao menos uma cópia ("backup") de cada um deles.

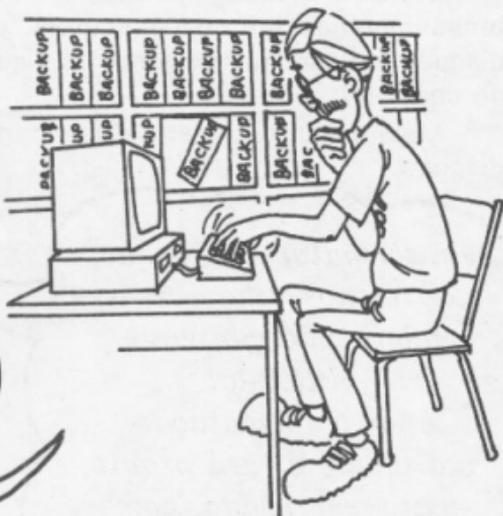
Imag e como seria desagradável perder seu melhor disquete sem ter uma cópia bem guardada e protegida.

Basta cair uma gotícula de café ou uma migalhinha de bolacha numa das regiões descobertas de um disquete para deixá-lo sem condições de uso!

Sendo assim, existem alguns comandos no computador próprios para podermos copiar um arquivo de um disquete para outro, ou mesmo para copiar todos os arquivos de uma só vez.

Mesmo sendo o usuário mais cuidadoso do mundo, você sempre estará sujeito a perder um disquete, afinal as quedas de força (que acontecem sempre quando não podem acontecer), a queima de um fusível (idem!), a poeira, etc., estão sempre ao nosso redor e são excelentes para inutilizar mesmo os melhores disquetes.

*Ter cópias
de todos os
programas e
demais arquivos
em disquetes
diferentes
pode evitar
grandes dores
de cabeça!!!*



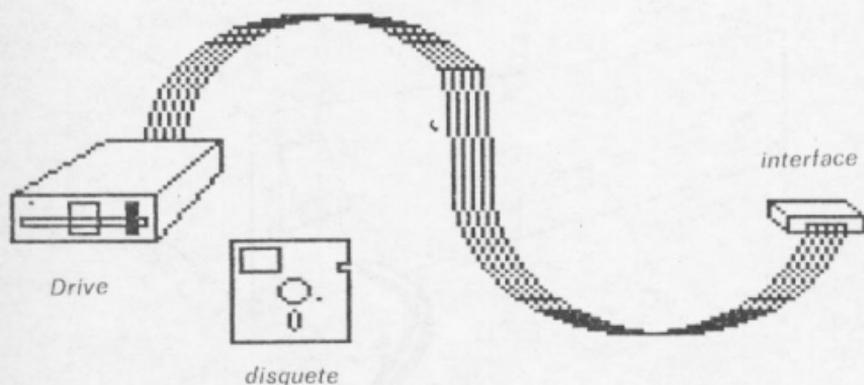
Após instalar o equipamento (no capítulo 2), e aprender a utilizá-lo, não deixe de fazer "BACKUP's" de seus disquetes!

COMO INSTALAR E TESTAR O EQUIPAMENTO



Junto com este livro você deve ter recebido um disquete, um drive e uma interface para controlá-lo (figura 2.1).

Figura 2.1 Drive, interface e disquete



Esse equipamento foi desenvolvido pela Microsol especialmente para a linha de microcomputadores MSX e é totalmente compatível com esse padrão.

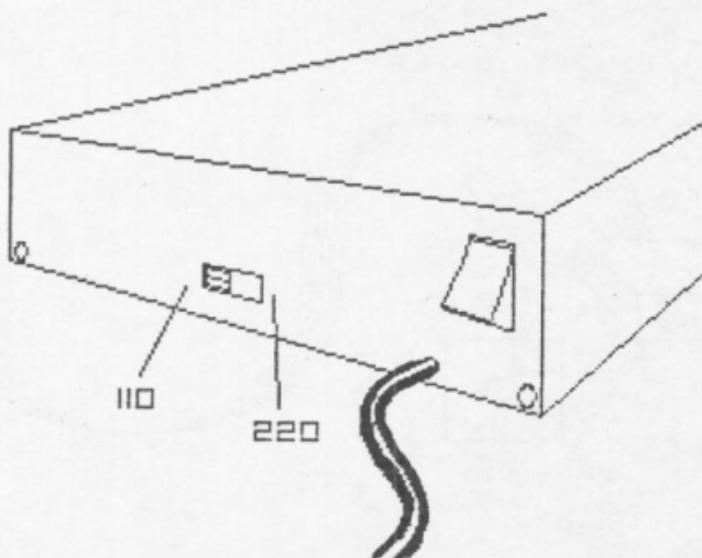
Uma vez conectado a um micro MSX, esse equipamento o tornará muito mais versátil e útil, além de substituir com muitas vantagens o uso de gravadores e fitas cassete.

Como já vimos, este é um equipamento sofisticado e, em alguns aspectos, mais sensível e delicado que o próprio microcomputador. Portanto tenha muito cuidado ao manuseá-lo, evitando choques e outros acidentes. Qualquer pequena desregulagem da parte mecânica ou eletrônica do drive, do cabo de ligação ou da interface pode por a perder toda a precisão do equipamento.

Vamos ver, passo a passo, como proceder a instalação e teste de um sistema microcomputador MSX com o drive Microsol.

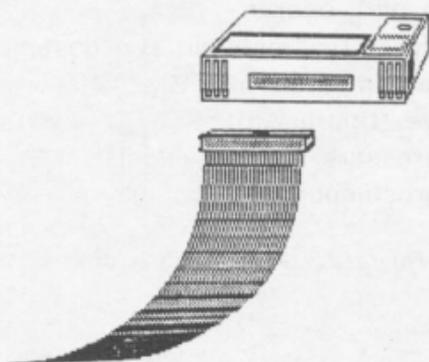
Inicialmente, verifique a voltagem da rede elétrica (110V/220V) e ajuste o seletor de voltagem do drive (figura 2.2). **Não ligue nada ainda!**

Figura 2.2 — Ajuste de voltagem do drive



Agora, conecte a interface CDX-2 ao cabo multi-fio que sai do drive. Observe que existe apenas UMA posição correta para essa conexão (figura 2.3).

Figura 2.3 — Conexão da interface



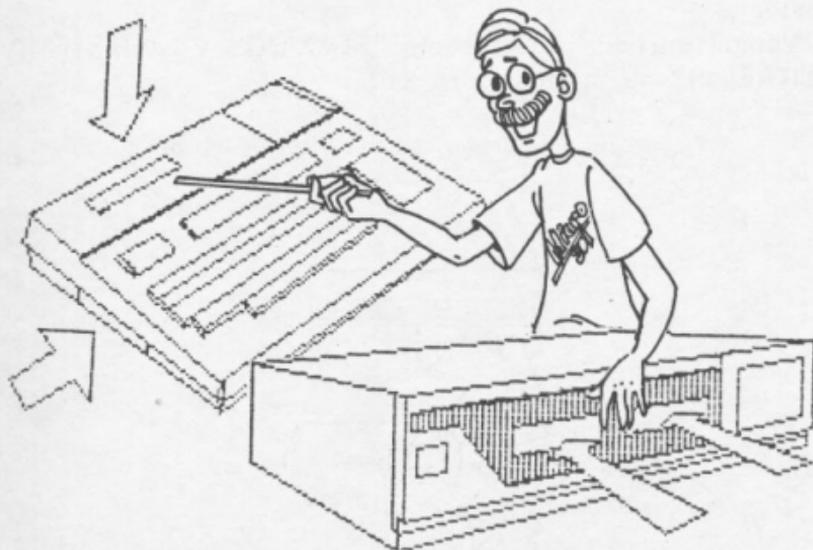
Nosso próximo passo será encaixar a interface no micro (que deve estar desligado).

Se for possível, é aconselhável que a interface seja instalada no "slot" A.

Para saber a posição correta dos cartuchos, consulte o manual do seu micro. Veja a forma correta de instalação observando a posição do furo do cartucho em relação ao micro.

Como exemplo, na figura 2.4, ilustramos a instalação da interface CDX-2 nos MSX Expert e Hotbit.

Figura 2.4 — Instalação da CDX-2 no Expert e no Hotbit

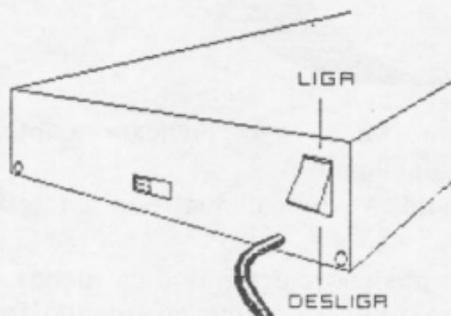


Uma vez conectada a interface ao drive e ao micro, o sistema estará pronto para operar.

Vamos agora fazer um teste de seu desempenho.

- Ligue o vídeo (monitor ou TV);
- Ligue o drive (figura 2.5);
- Ligue a impressora (se instalada!);
- Ligue o microcomputador.

Figura 2.5 — Ligando o drive



Observe que após as mensagens dos fabricantes aparecerem na tela, soará um bip e o drive será acessado (o indicador luminoso em sua parte frontal ficará aceso). Isso significa que a interface está tentando acessar algum disquete no drive.

É necessário inserir algum disquete no drive para liberar o sistema.

Vamos introduzir o disquete "SOLX-DOS V-1.1 SISTEMA E UTILITÁRIOS" no drive (figura 2.6).

Figura 2.6 — Introduzindo um disquete no drive

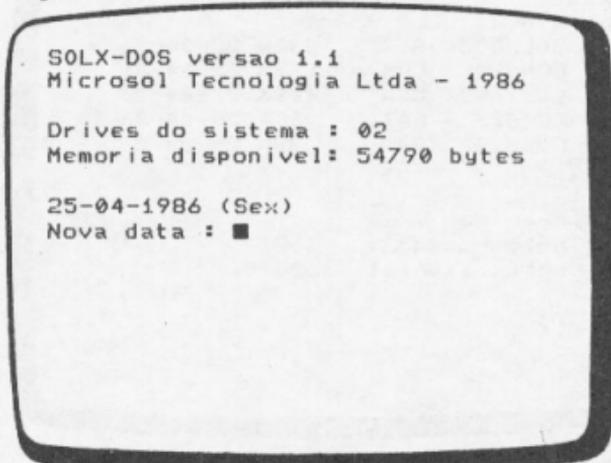


A partir do instante que a trava frontal do drive for acionada, a interface começará a ler o disquete a procura do sistema operacional SOLX-DOS.

Como ele está presente no disquete, será lido e carregado no micro.

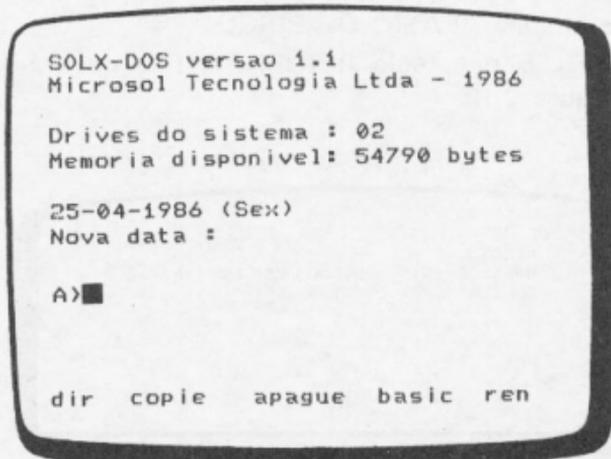
Inicialmente a tela ficará como mostra a figura 2.7.

Figura 2.7 — Tela de inicialização do SOLX-DOS



Não nos interessa agora aprender como usar o sistema, mas apenas testá-lo. Digite a tecla RETURN uma vez! A tela ficará com o aspecto mostrado na figura 2.8.

Figura 2.8 — SOLX-DOS pronto para uso



O sinal "A>" indica que o SOLX-DOS está pronto a espera de algum comando. Vamos usar um de seus comandos para ver quais os arquivos presentes no disquete.

Digite a tecla de função F1.

A tela deverá apresentar-se como mostra a figura 2.9.

Figura 2.9 — Diretório do disquete SOLX-DOS

```
A>dir
SOLXDOS  SIS      8320 25-04-86
CONVSOL  COM      1792 25-04-86
COPIARQ  COM      4096 25-04-86
COPISIS  BAT       384 25-04-86
COPSISD  BAT       384 25-04-86

Arquivos      :          5
bytes usados:  15872
bytes livres: 163840

A>
```

Até aqui estamos operando no sistema SOLX-DOS (carregado do disquete). Vamos sair do SOLX-DOS e entrar no BASIC DE DISCO. Para isso, pressione a tecla de função F4.

Agora, estamos operando em BASIC DE DISCO, isto é, o mesmo BASIC do micro sem a interface, mais alguns comandos e funções residentes na interface.

Vamos testar o BASIC DE DISCO.

Digite FILES e a tecla RETURN. A tela deverá ficar como mostra a figura 2.10.

Figura 2.10 — Comando FILES

```
FILES
SOLXDOS .SIS CONVSOL .COM COPIARQ .COM
COPISIS .BAT COPSISD .BAT
```

Por fim, vamos voltar do BASIC DE DISCO para o SOLX-DOS. Digite CALL SYSTEM e a tecla RETURN. O "A>" deve voltar ao vídeo.

Se até aqui tudo correu bem, como descrito no texto, seu equipamento está funcionando perfeitamente. É conveniente que, antes de continuar a usá-lo, seja feita uma cópia do disquete "SOLX-DOS V1.1 SISTEMA E UTILITÁRIOS". Para isso você vai precisar de um disquete virgem de tamanho 5¼" dupla face. Para fazer a cópia, siga exatamente as instruções do apêndice C.

Caso algo tenha dado errado, tente novamente desde o início. Se o problema persistir, há uma boa chance de seu equipamento estar avariado e é conveniente procurar o mais rapidamente possível uma assistência técnica.

O BASIC DE DISCO



Quando se liga um micro MSX sem que a interface CDX-2 da MICROSOL esteja conectada a ele, o controle de todo o sistema (micro, impressora e cassete) é realizado através de programas gravados na memória permanente (ROM) do micro. Essa memória não se apaga mesmo quando o micro está desligado da tomada e nela estão armazenadas todas as mensagens que aparecem na tela logo que o micro é ligado, todas as mensagens de erro ("Sintaxe error", por exemplo), as mensagens de auxílio (como o "Ok") e também o desenho de todos os 256 caracteres (letras, números e símbolos especiais).

Além disso, a ROM contém programas para ler o teclado, gerar imagens no vídeo, mandar caracteres para a impressora, ler e gravar dados no gravador cassete, ler os joysticks, e muitos outros que fazem o micro operar ativamente mesmo quando nós não o estamos usando.

Esse conjunto de dados e programas gravados pelos fabricantes na ROM dos micros é chamado de SISTEMA OPERACIONAL, pois é ele, em última instância, o verdadeiro responsável pelo funcionamento do sistema. Quase todos os microcomputadores pessoais vem da fábrica com algum sistema operacional residente em ROM.

Nos micros MSX, para poder usar o sistema operacional da ROM o usuário dispõe de uma poderosíssima linguagem de pro-

gramação: o BASIC MSX. Essa linguagem, suas regras sintáticas e seu vocabulário estão também armazenados na ROM. Por exemplo, se digitarmos:

PRINT "CASA"

(e RETURN)

O micro entenderá que deve apresentar a palavra CASA na tela e assim que digitarmos a tecla RETURN ele executará o que entendeu.

*O BASIC MSX
é a linguagem
que permite a
comunicação
entre o usuário
e o sistema.*



Existem muitas outras palavras do BASIC MSX que podem ser entendidas pelo micro mas não nos interessa estudá-las agora. Se você quiser maiores detalhes sobre a ROM, o sistema operacional, os dados e a linguagem BASIC MSX que ela contém, sugerimos a leitura atenta do livro "APROFUNDANDO-SE NO MSX" desta mesma editora.

Tudo o que vimos até aqui tem muito pouco a ver com o drive, porém, a compreensão do que é o sistema operacional e o que é o BASIC MSX é fundamental para começarmos a falar do BASIC DE DISCO.

Quando se liga um micro MSX com uma interface CDX-2 conectada a ele, além dos dados e programas que existem na ROM do micro temos à disposição os programas e dados que existem na ROM da interface (como é óbvio, a interface CDX-2 também tem uma ROM!).

São esses novos dados e programas residentes na interface que permitem o controle do disk drive.

Uma evidência notória da presença de novos dados e programas no sistema com a interface é o aparecimento na tela da mensagem

Microsol Tecnologia 1986

logo após o equipamento ser ligado.

Não só o sistema operacional ficou enriquecido, mas também a linguagem BASIC MSX ganhou algumas novas palavras em seu vocabulário e sintaxes mais versáteis.

Alguns antigos comandos do BASIC MSX passaram a ter novas sintaxes e outros comandos, antes inexistentes, podem agora ser usados. (A ROM do micro não sabe interpretá-los mas a ROM da CDX-2 sabe). Como ambas operam em perfeita harmonia, o usuário passa a dispor de uma linguagem ainda mais rica e poderosa para comandar o sistema (agora, com o drive!). É essa nova linguagem que chamamos de BASIC DE DISCO, um BASIC MSX capaz de controlar até 6 drives!

Com a interface instalada, o micro aumenta seu vocabulário e passa a "falar" o BASIC DE DISCO. Essa linguagem é ainda mais completa que o BASIC MSX e permite o controle de até 6 drives!



No capítulo 5 você poderá encontrar todas as novas palavras disponíveis no BASIC e também as palavras que tiveram suas sintaxes implementadas com novas opções.

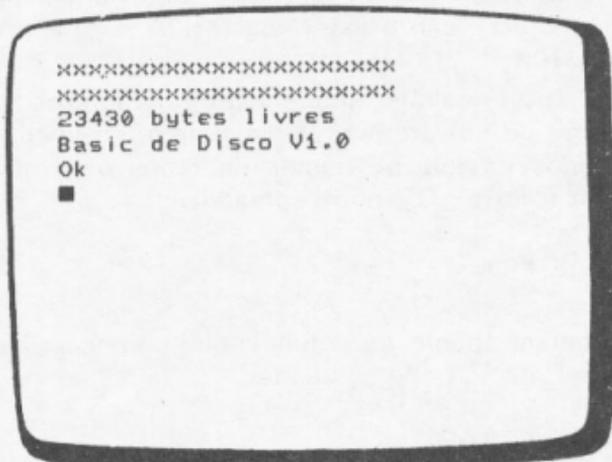
A seguir, vamos descrever apenas as principais delas!

ENTRANDO NO BASIC DE DISCO

Assim que o sistema é ligado (como descrito no capítulo 2), com o "A>" na tela, digite a tecla F4. Isso fará com que o BASIC DE DISCO passe a controlar o sistema.

A tela deverá ficar como mostra a figura 3.1.

Figura 3.1 — Tela de apresentação do BASIC DE DISCO



O "Ok" e o cursor indicam que o BASIC DE DISCO está pronto para receber comandos pelo teclado. Lembre-se: todos os comandos e funções do BASIC MSX continuam válidos.

Vamos experimentar alguns comandos mais usuais do BASIC DE DISCO.

ARQUIVOS EM DISQUETES

Sem tirar o disquete do SOLX-DOS do drive, digite:

FILES (e RETURN)

Na tela aparecerá a relação de todos os arquivos que estão gravados no disquete. Esse é um dos comandos mais úteis do BASIC DE DISCO.

Caso se queira saber se existe um determinado arquivo num disquete, pode-se especificar seu nome no comando FILES. Por exemplo, se quisermos saber se o arquivo SOLXDOS.SIS está gravado no disquete podemos comandar:

FILES "SOLXDOS.SIS"

Assim, se o arquivo procurado existir, seu nome será mostrado na tela. Caso contrário surgirá uma indicação de que o arquivo não foi encontrado.

Um utilíssimo recurso do BASIC DE DISCO é a possibilidade de se usar caracteres como "coringas" nos nomes dos arquivos. Esses caracteres são o asterisco (&H2A = *) e o ponto de interrogação (&H3F = ?).

O "*" substitui qualquer seqüência de caracteres que possa existir no nome de um arquivo. Por exemplo, se você quer que sejam mostrados na tela os nomes de todos os arquivos que começam com a letra "C", pode comandar:

```
FILES "C*.*"
```

Se você quiser apenas os arquivos que começam com "C" e terminam com "BAS" pode comandar:

```
FILES "C*.BAS"
```

O "?" funciona de forma semelhante, porém substitui apenas um caractere. Por exemplo, se você quiser saber quais os arquivos no disquete que terminam com SIS ou com SYS, pode comandar:

```
FILES "*.S?S"
```

O "*" e o "?"
funcionam
como
"coringas" no
comando
FILES!



Além do nome do arquivo, podemos também especificar o nome do drive no comando FILES. Por exemplo, se quisermos saber quais os arquivos no disquete do drive B (se ele estiver instalado!), deveremos comandar:

```
FILES"B:"
```

O uso do "" e do "?" só é permitido no nome do arquivo! O nome do drive não pode ter nenhum dos "coringas"!*



ESPAÇO LIVRE EM DISQUETE

Outro comando importante é o que nos mostra quanto existe de espaço livre no disquete. Experimente digitar:

```
PRINT DSKF(0) (e RETURN)
```

A função DSKF(0) obtém quantos blocos estão livres para serem usados no disquete que estiver no drive corrente. Normalmente, cada bloco contém 1024 bytes (ou 1 Kbyte), e portanto fica fácil saber quantos bytes livres existem no disquete.

Na função DSKF(n), o valor n especificado dentro dos parênteses serve para indicar qual o drive a ser usado. Se n=0, o drive corrente é usado. Para o drive A, deve-se ter n=1; para o drive B, deve-se ter n=2; e assim sucessivamente.

Agora, vamos ver como podemos gravar, ler e apagar dados nos disquetes.

GRAVAÇÃO EM DISQUETE

Daqui para diante, será conveniente que você use um outro disquete e guarde o do SOLX-DOS.

Se esse outro disquete estiver virgem, antes de usá-lo será necessário formatá-lo. Vamos ver como fazer isso.

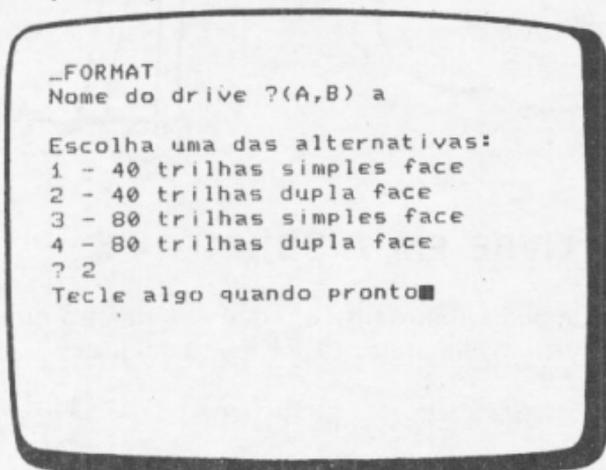
Insira o disquete no drive A e digite:

```
CALL FORMAT ou _FORMAT (e RETURN)
```

O micro perguntará qual o drive a ser usado. Digite a tecla da letra "A" para indicar o drive A.

O micro perguntará qual o tipo de formatação a ser feita. Digite a tecla do número "2" para indicar face dupla e 40 trilhas.

Você ainda terá que digitar uma tecla qualquer para confirmar a formatação e após alguns instantes, o disquete estará formatado e pronto para ser usado.



Agora, digite o programa em BASIC mostrado na figura 3.2.

Figura 3.2 — Programa BASIC

```
10 REM Programa exemplo  
20 PRINT "Arquivos:"  
30 FILES  
40 PRINT  
50 PRINT "Blocos livres:"  
60 PRINT DSKF(0)
```

Vamos gravá-lo no disquete.

Comande:

```
SAVE"A:PROGRAMA.BAS"
```

Ao pressionar RETURN, o programa é enviado para o drive e gravado no disquete.

Comande FILES novamente e surgirá na tela o nome PROGRAMA.BAS, indicando que agora existe um arquivo gravado no disquete.

O comando SAVE, com o drive, funciona de modo análogo ao comando CSAVE com o CASSETE, isto é, os programas são gravados no formato binário condensado.

Se você usar o SAVE para gravar um programa em cassete, ele será gravado no formato ASCII, mas no disco, ele será normalmente gravado em binário condensado para economizar espaço. Para gravar programas em ASCII no disquete, deve-se usar uma nova forma sintática do comando SAVE. Para exemplificar, vamos gravar novamente o programa da figura 3.2, porém em ASCII e com outro nome.

Com o programa na memória do micro, digite:

```
SAVE"A:TESTE.BAS",A (e RETURN)
```

O acréscimo do ".A" ao final do comando SAVE faz com que o programa seja gravado em ASCII.

Isto pode parecer inútil, mas há algumas situações em que é vantajoso ter os programas gravados desse jeito. Por exemplo, quando quisermos fazer um "merge" de um programa, ele terá que estar necessariamente em ASCII!

*O comando SAVE,
com disquetes,
funciona como
o CSAVE para
fitas.*



Comande FILES novamente e os dois arquivos gravados no disquete serão indicados na tela.

Como última observação sobre o SAVE, destacamos o fato de que ao executá-lo o micro dá prioridade ao drive. Isto é, se existir drive no sistema e comandarmos apenas

SAVE"XXX"

sem especificar o dispositivo (CAS:, A:; B:; etc...), o programa XXX será armazenado em formato binário no drive corrente. Se não houvesse drive no sistema, o programa seria enviado em ASCII ao cassete.

Para gravar em cassete com o comando SAVE, agora é essencial especificar "CAS:" como dispositivo!



No comando SAVE, se não especificarmos o dispositivo, será considerado o drive corrente! O micro dá prioridade ao drive sobre o cassete.

A seguir, vamos aprender a carregar programas do disquete para o micro.

TRANSFERÊNCIA DO DISQUETE PARA O MICRO

Comande

NEW

para limpar a memória.

Agora, comande:

LOAD"A:PROGRAMA.BAS"

Com isso, o programa será carregado do disquete para o micro.

Ao terminar a transferência, digite:

LIST (e RETURN)

O programa deverá ser listado na tela.

Vamos renumerá-lo.

Digite:

RENUM 1000 (e RETURN)

Agora o programa na memória do micro começa na linha 1000.

No disquete existe ainda o programa TESTE.BAS, que está gravado em ASCII. Vamos fazer um "merge" com ele!

Digite:

MERGE"A:TESTE.BAS" (e RETURN)

Com isso, o programa TESTE.BAS será transferido para a memória do micro sem apagar o que já estava nela. Quando o "Ok" e o cursor surgirem na tela, comande:

LIST

Agora, os dois programas que carregamos do disquete estão na memória, misturados um ao outro. Como isso não nos serve para nada, comande mais uma vez:

NEW

A seguir, vamos ver como eliminar arquivos do disquete.

APAGANDO ARQUIVOS EM DISQUETE

O programa TESTE.BAS é apenas uma versão em ASCII do programa PROGRAMA.BAS. Vamos apagá-lo do disquete, pois não nos servirá mais para nada.

Para isso, comande:

KILL"A:TESTE.BAS"

O comando KILL apaga arquivos do disquete. Se o nome do drive não for especificado o arquivo será procurado no drive corrente.

O comando KILL, da mesma forma que o FILES, permite o uso do "*" e do "?" no nome do arquivo. Por exemplo, se você quiser apagar todos os arquivos do drive corrente, basta comandar: `KILL"*.*"`

Cuidado! Não faça isso com o disquete que estamos usando, pois ainda vamos precisar do arquivo que ele contém!

ALTERANDO OS NOMES DE ARQUIVOS

Muitas vezes, após gravarmos um arquivo, desejamos mudar seu nome. O BASIC DE DISCO dispõe de uma instrução própria para isso: `NAME "... " AS "... "`

No disquete que estamos usando deve existir ainda um único arquivo de nome `PROGRAMA.BAS`. Vamos alterar seu nome para `TEST_#1.BAS`. Para isso, basta comandarmos:

```
NAME"A:PROGRAMA.BAS" AS "TEST_#1.BAS"
```

Para verificar a alteração do nome, comande `FILES`.

Se o drive não for especificado, será usado o drive corrente.

O comando `NAME` também permite o uso do "*" e do "?" nos nomes dos arquivos.

Agora, continuamos com apenas um programa gravado no disquete: aquele mesmo que gravamos como `PROGRAMA.BAS` e que alteramos para `TEST_#1.BAS`.

Para finalizar nossa breve introdução ao BASIC DE DISCO, vamos aprender como carregar automaticamente programas do disquete para o micro, assim que o sistema for ligado.

PROGRAMAS AUTO-EXECUTÁVEIS

Quando o equipamento (com a CDX-2 conectada!) é ligado, uma das primeiras coisas que ele faz é procurar um sistema operacional em disquete (por exemplo, o SOLX-DOS, o MSX-DOS, etc...). Se existir, o sistema será carregado e passará a controlar todo o equipamento. Foi isso o que ocorreu quando ligamos o equipamento com o disquete do SOLX-DOS dentro (capítulo 2).

O " A>" que surgiu na tela era uma indicação de que o SOLX-DOS estava no controle. Para desligarmos o SOLX-DOS e entrarmos no BASIC DE DISCO tivemos que digitar a tecla F4.

Se no disquete não existir nenhum sistema operacional gravado, a CDX-2 dará automaticamente o controle do equipamento ao BASIC DE DISCO, sem termos que digitar nenhuma tecla para isso.

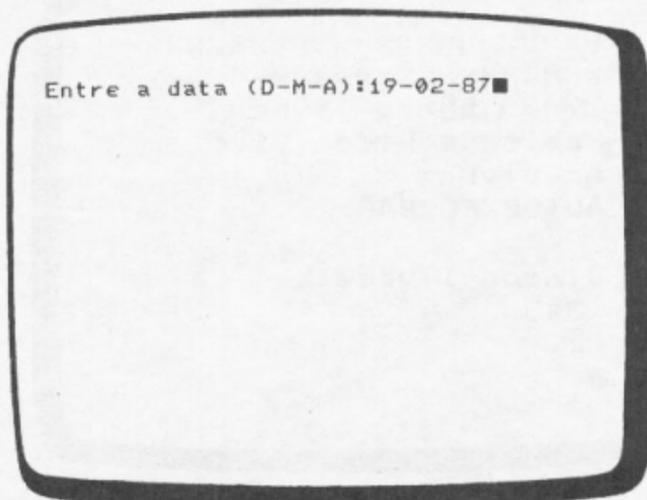
Experimente "ressetar" o equipamento sem tirar o disquete que estamos usando do drive (isso equivale a desligar e tornar a ligar tudo). Para isso, você pode proceder de diversas maneiras:

- 1) Se seu micro é um HOT-BIT, use o botão vermelho na lateral esquerda;
- 2) Se seu micro é um EXPERT V1.1, digite simultaneamente as teclas CONTROL, SHIFT e STOP;
- 3) Se seu micro é um EXPERT V1.0, comande:

```
DEFUSR0=0: ?USR(0)
```

Após o "reset" do equipamento o BASIC DE DISCO será automaticamente acionado, pois não há nenhum sistema operacional gravado no disquete em uso. A tela ficará como mostra a figura 3.3.

Figura 3.3 — Entrada da data no BASIC DE DISCO



O BASIC DE DISCO está pedindo a data atual. Todos os arquivos, quando são gravados, possuem uma indicação da data. Se você não quiser alterar a data, basta digitar a tecla RETURN. Para mudar a data, digite o dia (1 ou 2 dígitos), o mês (1 ou dois dígitos) e o ano (2 ou 4 dígitos), separando-os com "-" ou com "/".

Logo a seguir, surgirão na tela o "Ok" e o cursor.

Assim que o BASIC DE DISCO é acionado, uma das primeiras coisas que ele faz é verificar se no disquete do drive corrente existe algum programa BASIC gravado com o nome AUTOEXEC.BAS. Se existir, ele carregará e executará automaticamente esse programa. Nem mesmo a data atual será requerida na tela.

Para testar o uso de programas auto-executáveis, vamos alterar o nome do programa TEST_#1.BAS para AUTOEXEC.BAS e produzir novamente um "reset" no equipamento.

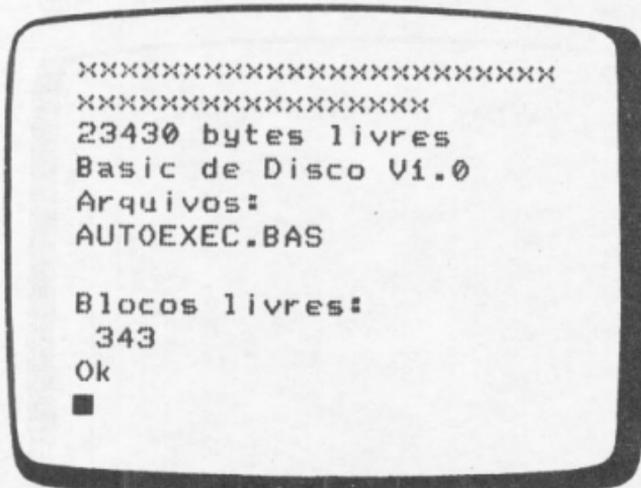
Comande:

```
NAME"TEST_#1.BAS" AS "AUTOEXEC.BAS"
```

Agora, produza um "reset" no micro, como descrito anteriormente.

Se tudo correu bem, a tela deve ter ficado como mostra a figura 3.4.

Figura 3.4 — Tela do AUTOEXEC.BAS.



*Sempre que existir
um programa BASIC
com o nome
AUTOEXEC.BAS num
disquete, o
BASIC DE DISCO
tentará executá-lo
automaticamente!*



Para certificar-se de que o programa foi carregado, comande LIST e veja-o na tela!

Existem muitos outros comandos e funções do BASIC DE DISCO que não vamos comentar agora. Você poderá encontrá-los explicados de forma sistemática no capítulo 5.

No próximo capítulo, vamos estudar o SOLX-DOS, um sistema operacional em disquete. Antes de começar a lê-lo, treine mais um pouco com o BASIC DE DISCO.

O SOLX-DOS



O SOLX-DOS é um sistema operacional residente em disquete desenvolvido pela MICROSOL para os micros da linha MSX.

Um sistema operacional, como já vimos, é apenas um conjunto básico de comandos e instruções que permite ao computador funcionar da maneira desejada, mesmo sem a intervenção do usuário.

Existem algumas diferenças marcantes entre um sistema operacional em disquete (como o SOLX-DOS) e um sistema operacional em ROM (como o BASIC DE DISCO).

Um dos motivos pelos quais normalmente se dá preferência a sistemas em disquete é o fato de que eles dependem menos da parte eletrônica do equipamento que os sistemas em ROM. Na verdade, os primeiros sistemas operacionais em disquete para micros foram criados justamente para permitir que um mesmo programa pudesse rodar sem problemas em máquinas estruturalmente diferentes.

Atualmente existem vários sistemas operacionais em disquete, sendo os mais difundidos o CP/M e o MS-DOS.

Desenvolvidos para rodarem sob o controle do CP/M, existem milhares de programas, para as mais diversas finalidades, e que funcionam perfeitamente em máquinas completamente diferentes como o TRS-80, I-7000, Sistema 700, etc. . .

O MS-DOS é usado, por exemplo, nos micros IBM-PC.

Programas desenvolvidos em BASIC DE DISCO, porém, só funcionam nos micros MSX.

O sistema operacional de disquete do padrão MSX (MSX-DOS) foi criado de modo a poder usar a grande maioria dos programas feitos para CP/M (apesar de usar uma formatação de disquetes totalmente diferente!). Além disso, o MSX-DOS usa uma formatação de disquetes idêntica ao do MS-DOS (usado pelo IBM-PC), o que possibilita o intercâmbio de arquivos entre os dois sistemas.

Como o SOLX-DOS foi criado de modo a ser compatível com o MSX-DOS, ele também está preparado para rodar quase todos os programas feitos para CP/M e para poder trocar arquivos com o MS-DOS. Obviamente, o SOLX-DOS pode rodar diretamente programas feitos para o MSX-DOS.

O CP/M, o MSX-DOS e o SOLX-DOS são quase totalmente compatíveis a nível de comandos. Isso significa que os comandos aceitos por cada um deles são, normalmente, aceitos pelos outros dois.

O MSX-DOS, o SOLX-DOS e o MS-DOS são compatíveis a nível de formatação, organização e acesso a disquetes. Isso significa que podem trocar arquivos entre si

O SOLX-DOS é compatível com o CP/M a nível de comandos, compatível com o MS-DOS, a nível de formatação e acesso a disco, e é compatível com o MSX-DOS sob ambos os aspectos!



Outra vantagem dos sistemas em disquete reside no fato de que usam de forma mais otimizada a memória RAM dos micros.

Você certamente já ouviu e leu que os MSX nacionais tem 64 Kbytes de RAM. Entretanto, quando no BASIC DE DISCO, apenas cerca de 25 Kbytes ficam livres para serem usados com programas.

Usando o SOLX-DOS, cerca de 55 Kbytes de RAM ficam livres para serem usados. Isso permite, por exemplo, o uso de programas extensos como compiladores BASIC, COBOL, ASSEMBLER, FORTH, LISP, C, PASCAL, FORTRAN, etc... Esses programas, além de ocuparem alguma região da RAM, agem sobre outros programas que também ocupam a RAM. Desse modo, quase toda a memória do micro tem que ser usada e no BASIC DE DISCO isso seria quase impossível.

Processadores de textos, planilhas eletrônicas, bancos de dados e muitos outros programas profissionais também podem ser utilizados com o SOLX-DOS.

A seguir, vamos estudar as principais características e os principais comandos do SOLX-DOS. Você irá aos poucos percebendo que sua estrutura é bem diferente da do BASIC DE DISCO.

O QUE É O SOLX-DOS

O SOLX-DOS é constituído por cinco arquivos em disquete:

```
SOLXDOS.SIS  
CONVSOL.COM  
COPIARQ.COM  
COPSIS.BAT  
COPSISD.BAT
```



Sua parte principal está no arquivo SOLXDOS.SIS que contém quase todos os comandos. É ele o primeiro programa a ser carregado do disquete assim que o equipamento é ligado.

Apesar de bastante poderoso o SOLX-DOS possui apenas alguns poucos comandos, o que o torna extremamente fácil de ser utilizado, inclusive com comandos em português! Todos os comandos que estão no SOLXDOS.SIS são chamados de "comandos internos" do sistema.

Os arquivos COPIARQ.COM e CONSOL.COM são dois comandos externos (ou comandos transientes) do sistema.

COPIARQ serve para se fazer o intercâmbio de arquivos entre periféricos, isto é, transferir arquivos de um drive para outro ou de um drive para a impressora.

CONVSOL é o comando que permite a leitura de arquivos gravados em disquetes formatados pelo CP/M e os adapta para o formato do SOLX-DOS, ou vice-versa.

Os outros dois arquivos, terminados por ".BAT", são também comandos externos, porém apenas executam uma seqüência de comandos internos (do SOLXDOS.SIS). Mais adiante, no final deste capítulo, nós os estudaremos. Por enquanto, basta saber que todo comando externo tem necessariamente que terminar com ".COM" ou com ".BAT".

Você já viu como utilizar um dos comandos do SOLX-DOS no capítulo 2 mas sem nenhuma explicação detalhada sobre como ele funcionava. Vamos ver agora como utilizar os recursos oferecidos por este sistema operacional.

INSTALANDO O SOLX-DOS

Já vimos no capítulo 2 a maneira correta de entrar no SOLX-DOS, mesmo sem saber bem o que estávamos fazendo.

Se ele não estiver instalado, proceda como descrito a seguir:

- Com o equipamento devidamente conectado e DESLIGADO, insira o disquete do SOLX-DOS no drive SEM fechá-lo (não abaixe a trava);
- Ligue o vídeo;
- Ligue o drive (e outros periféricos, se estiverem conectados);
- Ligue o micro;
- Por fim, feche o drive (abaixe a trava).

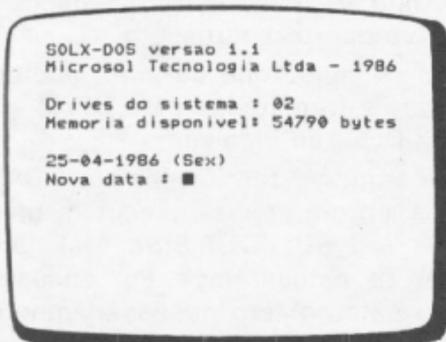
Ao entrar em funcionamento o micro procura no disquete presente no drive o sistema operacional MSX-DOS ou o SOLX-DOS, encontrando então este último. Após ser carregado do disquete, o SOLX-DOS passa a controlar o equipamento, apresentando uma tela com a da figura 4.1.

• Não se assuste com a mensagem:

DRIVES NO SISTEMA : 2

Ela surge porque, mesmo que exista apenas um drive físico conectado ao micro, a interface simula a presença de dois drives

Figura 4.1 — Tela inicial do SOLX-DOS



lógicos: o drive "A" e o drive "B". Ambos funcionarão no drive físico que estiver conectado e para cada um deles será reservada uma área da RAM do micro (buffer de discos).

Você pode forçar a interface a usar apenas o drive físico, se quiser. Para isso, basta manter a tecla CONTROL pressionada desde quando o micro for ligado até que seja mostrada a tela inicial. Isso pode ser útil, pois a área da memória que seria usada pelo drive lógico B ficará livre para outros usos. Existem programas que só conseguem rodar quando essa área está livre, pois são muito grandes.

Se você quiser, pode até fazer com que nenhum drive lógico seja usado, bastando para isso pressionar a tecla SHIFT (ao invés de CONTROL) enquanto liga o equipamento. Nesse caso, a própria interface ficará inativa e nem o BASIC DE DISCO, nem o SOLX-DOS ou qualquer outro sistema será carregado. Apenas o BASIC MSX normal estará operante.

Lembre-se que, enquanto liga o equipamento:

- *Mantendo CONTROL pressionada apenas, os drives físicos serão usados.*
- *Mantendo SHIFT pressionada, a interface ficará inoperante!*

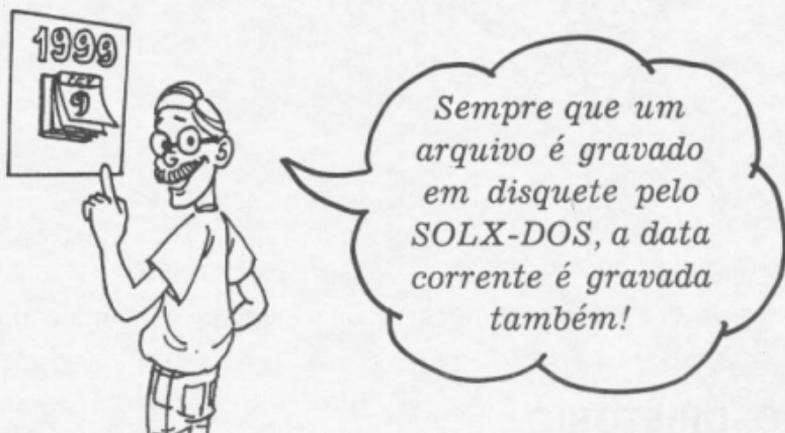


Bem, voltando à tela inicial do SOLX-DOS, vemos que ele está nos pedindo que entremos com uma data.

Você pode digitar alguma data correta ou simplesmente teclar RETURN.

RETURN faz com que a data considerada seja 25/04/86, que é o default do SOLX-DOS.

Para entrar com uma nova data, deve-se digitar o dia (2 dígitos), o mês (dois dígitos) e o ano (2 ou 4 dígitos) separados por "-" ou por "/". Se você digitar uma data inválida o SOLX-DOS a recusará. Caso você erre na data, não se preocupe. No SOLX-DOS existe o comando DATA (ou DATE) que é próprio para alterarmos a data e você pode ver como usá-lo no capítulo 5.



Após a entrada da data (ou de RETURN) o SOLX-DOS apresenta o seu sinal de "em funcionamento": o "A>".

Esse sinal indica que o SOLX-DOS está a espera de um comando.

A letra "A" indica que o drive corrente é o drive A (drive corrente é o que será acessado caso não haja especificação do drive).

A última linha do vídeo apresenta os comandos que estão associados às teclas de função de F1 a F10.

Esses comandos são, normalmente, os mais utilizados no SOLX-DOS e podem estar ou não seguidos do RETURN. Isto é, alguns comando são executados imediatamente após a digitação da tecla de função correspondente e outros apenas aparecem na tela a espera da digitação de parâmetros ou de RETURN.

*Os comandos
mais usados no
SOLX-DOS
estão nas teclas
de F1 a F10.*



F1 → dir + RETURN
F2 → copie
F3 → apague
F4 → basic + RETURN
F5 → ren
F6 → dir
F7 → data + RETURN
F8 → formate
F9 → save
F10 → copied a: b: + RETURN

Agora vamos começar a usar alguns comandos do SOLX-DOS.

O DIRETÓRIO

O comando do SOLX-DOS que apresenta no vídeo informações sobre os arquivos gravados no disquete é o DIR.

Para cada arquivo ele mostra o nome, o tamanho e a data em que ele foi gravado.

Além dessas informações é também apresentado quantos arquivos existem no disquete, quanto do disco já foi utilizado e quanto ainda resta de espaço livre.

Vamos usar o comando DIR. Para isso, podemos digitar:

DIR (RETURN)

ou apenas pressionar a tecla F1.

Pressione F1 e veja o que acontece. Se o disquete usado é o do SOLX-DOS a tela deve ficar como mostra a figura 4.2.

Figura 4.2 — Diretório do SOLX-DOS

```
A>dir
SOLXDOS  SIS      8320 25-04-86
CONVSOL  COM      1792 25-04-86
COPIARQ  COM      4096 25-04-86
COPSISS  BAT       384 25-04-86
COPSISSD BAT       384 25-04-86

Arquivos      :          5
bytes usados:  15872
bytes livres: 163840

A>
```

Para ver o diretório do drive "B", basta especificar "B:" após a palavra DIR, por exemplo:

```
A>DIR B:          (e RETURN)
```

Para facilitar a procura de um ou outro arquivo no disco, você pode digitar seu nome logo após o comando DIR. Experimente comandar:

```
A>DIR CONVSOL.COM
```

Se o arquivo existir, o SOLX-DOS o mostrará junto com o tamanho, data e hora (se houver). Caso contrário, se o arquivo não existir, uma mensagem de erro será mostrada.

Da mesma forma que no comando FILES do BASIC DE DISCO, o "*" e o "?" podem ser usados. Porém, além deles, o comando DIR também aceita o espaço em branco (&H20) como um "coringa". Isso está explicado com mais detalhes no capítulo 5.

Existem ainda duas opções muito úteis no comando DIR: "/W" e "/P".

A opção "/W" faz com que o comando DIR não mostre nem o tamanho, nem a data e nem a hora da última atualização do arquivo. Apenas seus nomes são mostrados em colunas que variam de três (vídeo com 40 colunas) até seis (vídeo com 80 colunas).

A opção "/P" faz com que o comando DIR mostre os arquivos do disco até que seja preenchida toda a tela. Quando isso acontece, o SOLX-DOS pede para ser pressionada qualquer tecla (exceto CONTROL+STOP ou CONTROL+C) para, só então, dar continuidade ao comando DIR. Esta opção é extremamente útil quando se tem muitos arquivos no disco, evitando que alguns desapareçam na parte superior da tela.

As duas opções, "/W" e "/P", podem ser usadas juntas, inclusive com o "?" e o "*". Experimente comandar:

```
DIR *.COM /P /W
```

Caso não se use "/P" e "/W" e no disquete existam muitos arquivos, pode-se interromper a apresentação na tela pressionando-se simultaneamente as teclas CONTROL e S. Para retomar a apresentação interrompida basta pressionar mais uma vez CONTROL e S.

Para interromper definitivamente a execução do DIR (ou de qualquer outro comando do SOLX-DOS!!!) e fazer o "A>" voltar à tela, basta pressionar simultaneamente CONTROL e STOP ou CONTROL e C.



Pressionando simultaneamente as teclas CONTROL e S, a apresentação no vídeo é interrompida ou reiniciada. Pressionando simultaneamente as teclas CONTROL e STOP ou CONTROL e C qualquer comando do SOLX-DOS é interrompido e o "A>" volta à tela!



Muitas vezes, devido a desajustes do vídeo, alguns caracteres nas laterais da tela não são visíveis. Isso pode ser facilmente resolvido com o comando MODO. Ele permite fixar o número de caracteres que se pode ter em cada linha da tela de forma semelhante ao comando WIDTH do BASIC. No capítulo 5 você poderá ver como usá-lo.

CONTEÚDO DE ARQUIVOS

Além de permitir ver na tela os nomes e dados gerais dos arquivos de um disquete, o SOLX-DOS permite ver o conteúdo dos arquivos.

O comando usado para isso é o LISTE (LIST ou TYPE).

O nome do arquivo que queremos examinar deverá estar logo após o comando. Por exemplo, se quisermos saber o que está gravado num arquivo (imaginário) de nome "TEXT0.DAT" deveremos comandar: `A>LISTE TEXT0.DAT`

Com isso o SOLX-DOS irá mostrar o que está gravado no arquivo TEXT0.DAT (se ele estiver presente no disquete!). Caso o arquivo não esteja no disquete, uma mensagem de erro será visualizada.

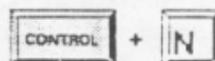
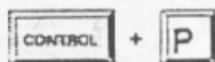
Você poderá fazer uma pausa na listagem, teclando `CONTROL+S`, ou interrompê-la definitivamente com `CONTROL+C`.

O comando LISTE (ou TYPE ou LIST) só tem sentido se utilizado com arquivos de textos ou de dados gravados em ASCII. Os programas não contém informações com sentido para nós humanos, mas apenas para o computador, e além disso não são, normalmente, gravados em ASCII.

Quando se usa o comando LISTE, poderá ser útil a aparição dos dados na impressora, além da tela.

O SOLX-DOS permite que tudo o que for enviado para a tela, seja também enviado à impressora. Para isso, quando o "A>" estiver à espera de um comando, bastará pressionar, simultaneamente as teclas CONTROL e P. Para desativar o "eco" da tela na impressora, bastará pressionar as teclas CONTROL e N.

*CONTROL+P ativa
o "eco" na impressora!
CONTROL+N o
desativa!*



CÓPIAS DE ARQUIVO

O SOLX-DOS permite copiar arquivos através do comando COPIE (ou COPY). Os arquivos podem ser copiados de um disquete para outro ou para o mesmo disquete, mas com um nome diferente. O comando COPIE está na tecla de função F2 e ao digitá-la ele surge na tela.

Por exemplo, se você quiser fazer uma cópia do SOLXDOS.SIS para o mesmo disquete em que ele está, mas dando a ele o nome TESTE.BAK, bastará comandar:

```
A>COPIE A:SOLXDOS.SIS A:TESTE.BAK
```

Note que o nome dos drives pode ser especificado e que o nome do arquivo a ser copiado vem antes do nome do arquivo a ser gerado. Nos nomes dos arquivos podem ser usados o "*" e o "?", quer no fonte, quer no destino.

Quando existir apenas um drive físico no sistema e a cópia for feita de um disquete para outro, o SOLX-DOS solicitará a troca de disquetes. O drive físico funcionará, hora como drive A:, hora como drive B:, simulando assim a presença de dois drives. Tal procedimento NÃO é possível se a tecla CONTROL tiver sido pressionada durante o carregamento do SOLX-DOS.

Se o arquivo a ser gerado tiver o mesmo nome do arquivo a ser copiado, não é necessário especificá-lo. Por exemplo, se comandarmos:

```
A>COPIE A:*.* B:
```

Então todos os arquivos que estiverem gravados no disquete do drive A serão copiados para o disquete no drive B e com os mesmos nomes. Isso foi usado no apêndice C, onde se fala sobre cópia do SOLX-DOS.

Obviamente, o disquete onde serão gerados os arquivos cópia deve estar formatado.

Para formatar o disquete no SOLX-DOS existe o comando FORMATE (ou FORMAT). Nós o usamos também no apêndice C, e agora vamos estudá-lo.

Imagine que temos um disquete para ser formatado.

Com o "A>" à espera de um comando, devemos inserir o disquete no drive A e comandar:

```
A>FORMATE
```

Então será solicitado o drive em que está o disquete a ser formatado. Basta digitar a letra correspondente a ele.

Com isso, o sistema pedirá para escolhermos o tipo de formatação, digitando a tecla do número correspondente à opção escolhida. Se o disquete é de 5¼" face simples, escolha a opção 1; se o disquete for de 5¼" face dupla, escolha a opção 2; se o disquete for de 3½" deve-se escolher a opção 3 ou 4.

Após definido o tipo de formatação a ser feita, deve-se ainda pressionar uma tecla para confirmar a formatação. O SOLX-DOS pede a confirmação para evitar acidentes, pois se um disquete já gravado for re-formatado, todos os dados que ele continha serão apagados.

Existe ainda uma outra forma de uso do FORMATE que é especificando o tipo de formatação e o drive em que ela deve ser feita no próprio comando.

Por exemplo, se quisermos formatar um disquete de 5¼" face dupla no drive B, podemos comandar:

```
A>FORMATE B: 2
```

Ao ser feita a cópia de um arquivo, pode-se fazer uma verificação da gravação. Obviamente o processo se torna mais lento, porém torna-se também mais confiável!

Para ativar a verificação basta acrescentar a opção "/V" ao fim do comando COPIE. Por exemplo:

```
A>COPIE A:*.*S?S B:/V
```

No comando anterior, os arquivos gerados no drive B serão verificados.

Outra forma de ativar a verificação de gravação é usar o comando VERIFICA (ou VERIFY). Veja como usá-lo no capítulo 5.

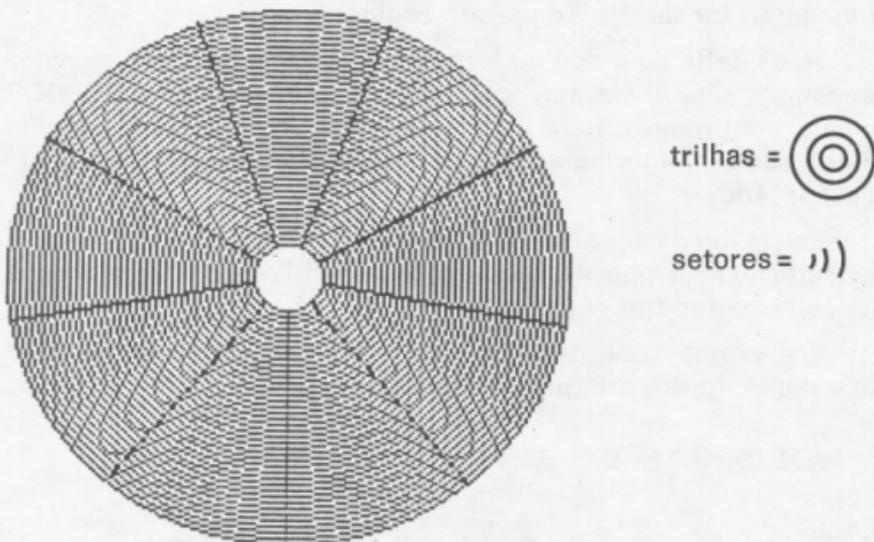
Como última observação sobre o comando COPIE, ressaltamos o fato de que ele copia ARQUIVOS.

A seguir vamos ver um comando muito semelhante, mas que copia DISQUETES. Pode parecer um pouco confuso mas não é! Basta que consigamos compreender como um disquete fica organizado após a formatação para eliminarmos a confusão.

CÓPIA FÍSICA DE DISQUETES

Quando um disquete é formatado ele fica dividido em trilhas e setores como mostra a figura 4.3.

Figura 4.3 — Formatação do SOLX-DOS



Vamos usar como exemplo um disquete de 5 1/4" face dupla. Ele é dividido em 40 trilhas e cada trilha tem 9 setores. O total de setores é, portanto, 360 ($=40 \times 9$).

As primeiras trilhas de um disquete são normalmente usadas para armazenar o diretório e outras informações gerais. Os dados são armazenados nas demais trilhas:

O comando COPIE que, vimos anteriormente, lê no diretório em que setores e trilhas os arquivos a serem copiados estão armazenados, os carrega na memória e os copia.

No SOLX-DOS existe um comando que copia todo o disquete, setor por setor, sem se preocupar se está copiando arquivos ou não: o COPIED, presente na tecla de função F10.

Esse comando espera receber dois parâmetros: o drive onde está o disquete a ser copiado e o drive onde está o disquete em que será produzida a cópia. Por exemplo, se quiséssemos copiar o disquete do drive A para um disquete no drive B bastaria comandar:

A)>COPIED A: B:

O disquete onde será feita a cópia não precisa estar formatado, pois o próprio COPIED se encarregará de formatá-lo de forma idêntica ao disquete a ser copiado.

Agora, apenas observe os dois comandos a seguir:

```
A>COPIE A:*. * B:
```

```
A>COPIED A: B:
```

Se o disquete a ser copiado for o do SOLX-DOS, qual dos dois comandos vai demorar mais tempo para ser executado?

Pense um pouco para responder a pergunta e lembre-se que o COPIED copia o disquete inteiro, setor por setor, mesmo que em todos os lugares existam apenas caracteres "0" gravados!

APAGANDO ARQUIVOS

No SOLX-DOS o comando APAGUE (ou ERASE) serve para apagar um arquivo do disco e funciona de forma semelhante ao KILL do BASIC DE DISCO.

O nome do arquivo a ser apagado deve ser especificado logo após o comando APAGUE. Por exemplo, se quisermos apagar de um disquete no drive A todos os arquivos terminados por .BAS temos que comandar:

```
A>APAGUE A:*.BAS
```

Note que podemos usar o "*" e o "?" no nome do arquivo. O nome do drive, entretanto, pode ser omitido, e nesse caso será considerado o drive corrente.

Se o arquivo especificado estiver presente no disquete, será apagado. Caso contrário uma mensagem de erro será mostrada.

O comando APAGUE está na tecla F3 e em vez de digitá-lo letra por letra você pode simplesmente digitar a tecla F3.



*Cuidado ao apagar
um arquivo, pois
ele nunca mais
poderá ser recuperado.*

Quando utilizamos *.* como nome do arquivo estamos dizendo que queremos apagar tudo que está no disco, o que é um tanto perigoso. Quando fazemos isso, o SOLX-DOS pede uma confirmação da operação antes de executá-la.

Não apague nenhum dos arquivos que estão no disquete "SISTEMA OPERACIONAL SOLX-DOS" pois eles são de extrema importância e muita utilidade!

ALTERANDO O NOME DE UM ARQUIVO

Agora, imagine que você tem um disquete em que todos os arquivos terminam por ".BAS". Se você quiser apagar todos os arquivos menos um, por exemplo, de nome "ESZ.BAS", não poderá comandar "APAGUE *.*" e nem "APAGUE *.BAS".

A solução seria mudar o nome do arquivo "ESZ.BAS", a ser preservado, para algo do tipo "ESZ.XXX" e comandar "APAGUE *.BAS". Dessa forma, apenas o arquivo "ESZ.XXX" permaneceria no disquete. Depois, poderíamos alterar seu nome para "ESZ.BAS" novamente.

No SOLX-DOS o comando que permite alterar o nome de um arquivo é o RENAME (ou REN).

Para usá-lo, deve-se especificar primeiro o nome atual do arquivo e, logo em seguida, o novo nome a ser dado à ele. No exemplo dado acima, deveríamos comandar:

```
A>RENAME ESZ.BAS ESZ.XXX
A>APAGUE *.BAS
A>RENAME ESZ.XXX ESZ.BAS
```

Pode-se também modificar o nome de diversos arquivos de uma só vez, utilizando o "?" e o "*". Quando os utilizamos estamos dizendo ao SOLX-DOS que naquela posição os caracteres não serão modificados, e que devem ser modificados apenas aqueles que especificamos. Veja um exemplo:

```
A>REN *.COM *.SEM
```

Com este comando, todos os programas terminados por ".COM" terão seus nomes modificados de modo a terminarem por ".SEM", não sendo mais reconhecidos como programas pelo SOLX-DOS.

A tecla F5 já mostra a palavra REN para lhe poupar o trabalho de digitá-la letra por letra.

PROCESSAMENTO BATCH

Com o uso constante do SOLX-DOS, você acabará tendo muita intimidade com os comandos e saberá qual deles utilizar em cada situação específica.

Acontece, porém, que em certas situações precisamos digitar vários comandos seguidos e mesmo com muita prática este serviço acaba sendo maçante.

Existem dois tipos de arquivos que podem ser executados no SOLX-DOS: os terminados em ".COM" e os terminados em ".BAT".

Todo arquivo terminado por ".COM" é reconhecido como um programa. Por exemplo, no próprio disquete do SOLX-DOS os arquivos COPIARQ e CONVSOL são programas. Para executar um programa no SOLX-DOS basta que, quando o "A>" estiver a espera de um comando, seja digitado o nome do programa sem a terminação ".COM". Por exemplo, para executar o CONVSOL.COM deve-se comandar:

```
A>CONVSOL
```

Agora, imagine que você queira executar os comandos DIR, APAGUE TESTE.BAS e APAGUE TESTE.BIN, um após o outro.

Você pode comandar:

```
A>DIR  
A>APAGUE TESTE.BAS  
A>APAGUE TESTE.BIN
```

Ao invés de receber as linhas de comandos pelo teclado, o SOLX-DOS pode receber uma ou várias linhas de comandos de um arquivo em disquete.

Para isso, devemos inicialmente escrever em um arquivo todos os comandos que digitaríamos no teclado. Depois indicamos ao SOLX-DOS para passar a executar os comandos lá presentes. Então, é só ficar vendo o resultado: o computador trabalhando sozinho!!!

A única particularidade do arquivo com os comandos a serem executados está no nome: ele obrigatoriamente deve ter a extensão ".BAT" para que o SOLX-DOS o reconheça como arquivo de comandos.

INDO E VOLTANDO AO BASIC

Você já deve ter observado que a tecla de função F4 no SOLX-DOS é programada com o comando BASIC.

Este comando permite a você sair do SOLX-DOS e entrar no BASIC DE DISCO.

Ao utilizá-lo, o SOLX-DOS sai automaticamente de execução dando lugar para o BASIC DE DISCO.

Você poderá especificar um nome de programa para o BASIC DE DISCO executar assim que receber o controle. Para isso basta digitar seu nome logo após o comando BASIC. Por exemplo:

```
A>BASIC JOG01
```

No exemplo anterior, JOG01 tem necessariamente que ser um programa escrito na linguagem BASIC. Se não for, o BASIC DE DISCO emitirá uma mensagem de erro.

Uma vez estando no BASIC DE DISCO, você poderá voltar ao SOLX-DOS usando o comando CALL SYSTEM.

Note porém que este comando (CALL SYSTEM) só funciona no BASIC DE DISCO se este tiver sido executado pelo comando BASIC do SOLX-DOS. Se você digitar CALL SYSTEM e surgir uma mensagem de erro é porque o SOLX-DOS não foi carregado antes.

Pode acontecer também de se trocar o disquete e ao se comandar CALL SYSTEM o BASIC DE DISCO não encontrar o SOLX-DOS. Nesse caso, ele lhe pedirá para inserir algum disco que contenha o SOLX-DOS gravado e só então conseguirá devolver o controle do equipamento a ele.

Experimente comandar:

```
A>BASIC
```

E a seguir, experimente:

```
CALL SYSTEM
```

O comando
CALL SYSTEM
só será executado
se o **SOLX-DOS**
tiver sido usado
antes e se estiver
presente no disquete
do drive corrente.



ARQUIVOS AUTO-EXECUTÁVEIS

No disco "SISTEMA OPERACIONAL SOLX-DOS" existem dois arquivos terminados com ".BAT": o "COPSIS.BAT" e o "COPISD.BAT". Ambos são típicos exemplos da utilidade dos arquivos tipo "BATCH".

Vamos analisar o "COPSIS.BAT".

Nesse arquivo existem alguns comandos para formatar e tirar uma cópia do disquete "SISTEMA OPERACIONAL SOLX-DOS". Os comandos estão dispostos da mesma maneira na qual você os digitaria.

Para utilizá-los, basta digitar seu nome (sem a extensão ".BAT") e pronto: o SOLX-DOS passará a executar os comandos.

Para os arquivos "BATCH" existem dois comandos muito úteis: o PAUSE e o REM (ou /).

O comando PAUSE faz com que o SOLX-DOS gere uma pausa na execução dos comandos contidos no arquivo, mostrando uma mensagem na tela e esperando você teclar algo para continuar o processo.

Para cancelar a execução do processamento tipo "BATCH" basta teclar CONTROL-STOP ou CONTROL-C. Neste caso o SOLX-DOS perguntará se realmente você deseja cancelar a execução.

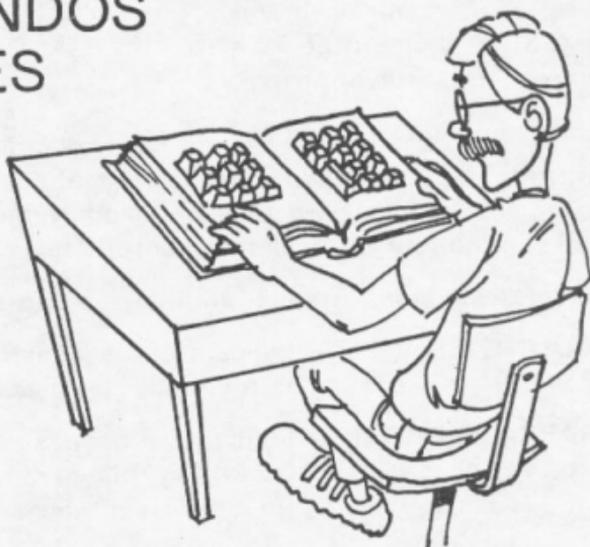
O comando REM (ou /) permite que sejam inseridos comentários no arquivo. Quando o SOLX-DOS recebe um comando REM ele simplesmente ignora toda aquela linha.

Existe um nome especial de arquivo tipo "BATCH" que funciona no SOLX-DOS de forma semelhante ao "AUTOEXEC.BAS" no BASIC DE DISCO: o AUTOEXEC.BAT.

Sempre que o SOLX-DOS entra em execução ele procura este arquivo. Se ele existir o SOLX-DOS passa a executar seus comandos automaticamente, possibilitando assim que certos programas sejam executados assim que o SOLX-DOS entre em ação.

Para produzir arquivos "BATCH" pode-se usar editores de texto (como o SCED, o MSXWRITE, etc.) ou o comando COPIE com o dispositivo CON: (console) como entrada. Veja no capítulo 5 como digitar textos e gravá-los em arquivos com o comando COPIE.

DICIONÁRIO DE COMANDOS E FUNÇÕES



Nas páginas a seguir estão relacionados todos os comandos do BASIC DE DISCO e do SOLX-DOS. Para cada comando ou função é dada a forma sintática para sua correta utilização, comentários e observações importantes e exemplos elucidativos.

Na figura abaixo apresentamos um "mapa" das informações para cada comando ou função.

Nome do comando
ou função

Especificação
de tipo

XXXXX



Sintaxe: Forma correta de se usar.
Função: Função a que se presta o comando ou função.
Observações: Observações importantes sobre o comando ou função.
Exemplo: Exemplo(s) de uso.

Em todos os textos do dicionário, algumas abreviações foram utilizadas:

dsp = dispositivo (CRT:, GRP:, CAS:, drv, etc)
drv = drive (A:, B:, C:, D:, E:, F:)

prg	=	nome de programa
var	=	variável numérica
arq	=	nome de arquivo
nXXX	=	número do XXX
vars	=	variável string

Além dessas abreviações, na especificação da sintaxe correta de cada comando ou função, adotou-se uma convenção de símbolos lógicos para indicar quando uma construção é opcional (obrigatória ou exclusivamente).

Os símbolos e seus significados são os seguintes:

- COLCHETES:** — Indicam que a construção xxx é opcional, isto é, pode ser usada ou não;
[xxx]
- BARRA VERTICAL:**— Indica que apenas uma das construções (xxx ou yyy) pode ser usada, isto é, se xxx for usada yyy não será e vice-versa;
xxx | yyy
- CHAVES:** — Indicam a obrigatoriedade de uso de uma das construções xxx, yyy, etc, isto é, faz-se absolutamente necessário o uso de ao menos uma das construções entre as chaves.
{ xxx yyy etc }

A seguir apresentamos um sub-índice do dicionário. Para achar alguma palavra a ser procurada, veja aqui em que página ela se encontra. Note que algumas palavras, como FORMAT e COPY, existem tanto no BASIC DE DISCO quanto no SOLX-DOS e desempenham funções semelhantes em ambos os sistemas. Cuidado para não ler as explicações do BASIC DE DISCO quando queria ler as do SOLX-DOS ou vice-versa!

PALAVRAS DO DICIONÁRIO DO BASIC DE DISCO

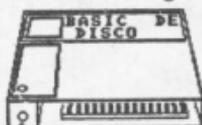
BLOAD	60	BSAVE	61
CLOSE	62	COPY	62
CVD	63	CVI	63
CVS	63	DSKF	63
DSKIS	64	DSKOS	65
EOF	66	FIELD	67

FILES	68	FORMAT	68
GET	69	INPUT #	70
INPUTS	71	KILL	71
LINE INPUT #	72	LOAD	73
LOC	74	LOF	75
LSET	75	MERGE	76
MKD\$	77	MKIS\$	77
MKS\$	77	NAME	78
OPEN	79	PRINT #	80
PUT	81	RSET	82
RUN	83	SAVE	83
SYSTEM	84	VARPTR	84

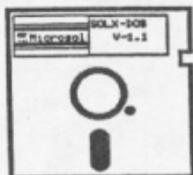
PALAVRAS DO DICIONÁRIO DO SOLX-DOS

APAGUE	85	BASIC	85
CONVSOL	86	COPIARQ	87
COPIE	91	COPIED	92
COPY	91	DATA	92
DATE	92	DEL	85
DIR	93	ERASE	85
FORMAT	94	FORMATE	94
LIST	95	LISTE	95
MODE	95	MODO	95
PAUSA	96	PAUSE	96
REM	96	REN	97
RENAME	97	SALVE	98
SAVE	98	TIME	99
TYPE	95	VERIFICA	99
VERIFY	99		

Para cada comando e função, especifica-se o sistema operacional através das figuras abaixo:

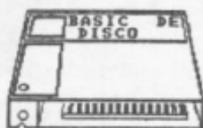


= BASIC DE DISCO

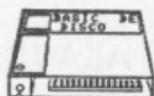


= SOLX-DOS

DICIONÁRIO DO BASIC DE DISCO



O *BASIC DE DISCO*
é o *Sistema Residente*
em *ROM!!!*



BLOAD

Sintaxe: BLOAD "{[dsp] [arq]}" [,R][,S] [,deslocamento]

Função: Transferir um programa armazenado em fita cassete ou em disquete no formato binário para a memória do micro-computador.

Observações: **dsp** deve ser CAS: para cassete, A: , B: , C: , D: , E: ou F: para os drives. Quando não especificado, será usado o drive corrente (se instalado) ou o gravador cassete. O arquivo deve ter sido gravado no formato binário com o comando BSAVE e não precisa ser especificado se a transferência for feita do gravador cassete.

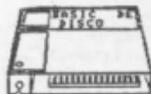
O programa será carregado no endereço de memória especificado por BSAVE somado ao **deslocamento**. Se o deslocamento não for especificado, assume-se que ele é nulo.

A opção **,R** faz com que o programa seja automaticamente executado assim que terminar a transferência.

A opção **,S** faz com que seja carregada na VRAM uma imagem de vídeo salva por BSAVE.

Exemplo:

```
10 BLOAD"B=TELA.BIN",S
20 BLOAD"COP3C000.BIN",R
30 GOTO 30
```



BSAVE

Sintaxe: BSAVE "{[dsp] [arq]}" , início, fim [,execução ,S]

Função: Transferir um programa em binário ou dados da VRAM para fita cassete ou disquete.

Observações: **dsp** deve ser CAS: para o gravador cassete, A: , B: , C: , D: , E: ou F: para os drives. Quando não especificado a transferência será feita para o drive corrente (quando instalado) ou para o gravador cassete.

início é o endereço do primeiro byte da memória a ser transferido.

fim é o endereço do último byte da memória a ser transferido.

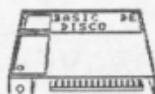
execução é o endereço do byte onde começará a execução do programa. Se execução não for especificado, o endereço de início será também o de execução.

A opção **,S** faz com que seja gravada uma região da VRAM.

Exemplo:

```
10 SCREEN 2
20 FOR F=10 TO 80 STEP 10
30   CIRCLE (128,86),F
40 NEXT F
50 BSAVE"TELA.BIN",0,&H4000,S
60 GOTO 60
```

CLOSE



Sintaxe: CLOSE [#arq1] [,#arq2] [...]

Função: Encerrar o uso de arquivos.

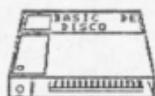
Observações: **arqn** indica o arquivo a ser fechado. Quando omitido, serão fechados todos os arquivos abertos.

END, CLEAR, RUN, NEW e MAXFILES executam um CLOSE automaticamente.

Exemplo:

```
10 OPEN "A:FONTE" AS #1 LEN=1
20 FIELD #1, 1 AS IN$
30 FOR F=1 TO LOF(1)
40   PRINTUSING"#####";F;
50   GET#1,F : C$=IN$ : PRINT "...";
60   IF ASC(C$)>31 THEN PRINT C$
70   IF ASC(C$)<32 THEN PRINT "<32"
80 NEXT F
90 CLOSE
```

COPY



Sintaxe: COPY "[drv1]arq1" TO "[drv2]arq2"

Função: Copiar arquivo(s) de um disquete para outro.

Observações: **drv1** é o drive lógico onde está o arquivo a ser copiado (**arq1**).

drv2 é o drive lógico em que será feita a cópia (**arq2**). Se **arq2** não for especificado, será considerado igual a **arq1**.

Pode-se usar "*" e "?" em **arq1** e **arq2**.

Quando apenas um drive físico estiver conectado ao micro, pode-se proceder à cópia como se existissem dois drives. O sistema solicitará a troca dos disquetes durante a transferência sempre que necessário. Quando **arq2** não for especificado, será assumido o mesmo nome do **arq1**.

Exemplo:

```
COPY "A:TESTE*.*)" TO "B:"
```

CVI, CVS, CVD



Sintaxe: var = CVI (vars)
var = CVS (vars)
var = CVD (vars)

Função: Converter uma variável string de 2 bytes numa variável numérica inteira.

Converter uma variável string de 4 bytes numa variável numérica de precisão simples.

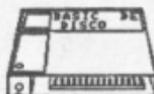
Converter uma variável string de 8 bytes numa variável numérica de precisão dupla.

Observações: Valores numéricos inteiros, de precisão simples ou de precisão dupla são armazenados em arquivos na forma de strings (veja MKI\$, MKS\$, MKD\$). Ao ser lida de um arquivo a string correspondente a um valor numérico, é necessário convertê-la com CVI, CVS ou CVD, conforme o valor seja respectivamente INTEIRO, de PRECISÃO SIMPLES ou de PRECISÃO DUPLA.

Exemplo:

```
10 OPEN "A:HORRAD" AS #1 LEN=8
20 FIELD #1,8 AS O$
30 FOR F=1 TO LOF(1)/8
40     GET #1,F : D=CVD(O$) : PRINT D
50 NEXT F
60 CLOSE
```

DSKF



Sintaxe: DSKF(ndrv)

Função: Fornecer a quantidade de blocos lógicos não usados num disquete.

Observações: ndrv deve ser 0 para o drive corrente, 1 para o drive A, 2 para o drive B, 3 para o drive C, 4 para o drive D, 5 para o drive E e 6 para o drive F.

Exemplo:

```
PRINT DSKF(0)
```

DSKI\$



Sintaxe: DSKI\$ (ndrv,nsetor)

Função: Ler um setor de um disquete.

Observações: **ndrv** deve ser 0 para o drive corrente, 1 para o drive A, 2 para o drive B, 3 para o drive C, 4 para o drive D, 5 para o drive E e 6 para o drive F.

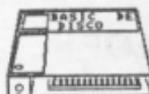
nsetor começa em 0 (para o primeiro setor do disco) e vai até 359.

Os bytes lidos no disquete são armazenados numa região da RAM cujo primeiro endereço é indicado por &HF351 e &HF352.

Exemplo:

```
100 X=40*9-1
110 FOR F=0 TO X
120 PRINT:PRINT "Setor";F;" : "
130 A%=DSKI$(0,F)
140 PT=PEEK(&HF351)+256*PEEK(&HF352)
150 FOR G=0 TO 255
160 A=PT+G : A%=CHR$(PEEK(A))
170 PRINT HEX$(A);" =>";
180 PRINT HEX$(PEEK(A));" " ;
190 IF PEEK(A)>31 THEN PRINT A$
200 PRINT
210 NEXT G
220 NEXT F
```

DSKOS



Sintaxe: DSKOS ndr_v,nsetor

Função: Inserir diretamente dados num setor do disquete.

Observações: Os dados serão gravados no setor lógico de número dado por **nsetor**.

ndrv deve ser 0 para o drive corrente, 1 para o drive A, 2 para o drive B, 3 para o drive C, 4 para o drive D, 5 para o drive E e 6 para o drive F.

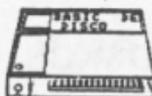
Os bytes da RAM a serem gravados tem seu primeiro endereço indicado por &HF351 e &HF352.

Exemplo:

```
10 REM Este disquete e meu, meu e de mais  
s ninguem! Nao quero que ninguem consiga  
ler esta mensagem, mas se voce a ler, q  
uero que a destrua e a esqueca! Isto e t  
udo, por enquanto, mas nao se esqueca: e  
squeca a mensagem, ok ? Caso nao consiga  
esquece-la ..
```

```
20 X=32775!  
30 MA%=INT(X/256)  
40 MI%=X-256*MA%  
50 POKE &HF351,MI%  
60 POKE &HF352,MA%  
70 DSKOS 0,359
```

EOF



Sintaxe: EOF(narq)

Função: Checar se o final de um arquivo sequencial aberto como entrada foi encontrado.

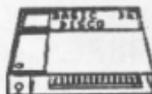
Observações: narq deve ser o número de um arquivo sequencial aberto como entrada.

O valor fornecido por EOF é -1 se o final do arquivo foi achado. Caso contrário será 0.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" FOR INPUT AS #1
120 OPEN "A:DESTINO" FOR OUTPUT AS #2
130 IF EOF(1) THEN CLOSE : END
140 INPUT#1,A$
150 PRINT#2,A$ : F=F+1
160 PRINTUSING"#####";F+1;
170 PRINT,A$
180 GOTO 130
```

FIELD



Sintaxe: FIELD[#]narq,nbytes AS var1
[,nbytes AS var2] [,nbytes AS var3] [...]

Função: Reservar uma área do buffer dos arquivos randômicos para variáveis strings.

Observações: narq deve ser o mesmo usado para abertura do arquivo.

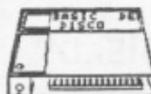
nbytes é o tamanho que terá no buffer cada variável string (var1, var2, etc...) e não pode ser uma variável numérica. As variáveis strings usadas por FIELD não podem ser usadas por INPUT ou LET.

O uso do FIELD é essencial antes do uso do PUT e do GET.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" AS #1 LEN=1
120 FIELD #1, 1 AS IN$
130 OPEN "A:DESTINO" AS #2 LEN=1
140 FIELD #2, 1 AS OU$
150 FOR F=1 TO LOF(1)
160   PRINT USING "#####";F;
170   PRINT ,
180   GET#1,F
190   C$=IN$
200   IF ASC(C$)>31 THEN PRINT C$
210   IF ASC(C$)<32 THEN PRINT "<32"
220   LSET OU$=C$
230   PUT#2,F
240 NEXT F
250 CLOSE
```

FILES



Sintaxe: FILES ["[drv] [arq]"]

Função: Apresentar os arquivos presentes num disquete.

Observações: Podem ser usados "*" e "?" em arq.

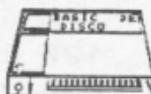
Se **arq** não for especificado, serão mostrados todos os arquivos do disquete.

Se **drv** for omitido, será assumido o drive corrente.

Exemplo:

```
10 REM AUTOEXEC.BAS
20 SCREEN 0
30 WIDTH 39
40 PRINT"OS ARQUIVOS DESTE ";
50 PRINT"DISQUETE SAO:"
60 PRINT
70 FILES
```

FORMAT



Sintaxe: CALL FORMAT
_FORMAT

Função: Formatar um disquete para uso do BASIC DE DISCO, do SOLX-DOS, ou do MSX-DOS.

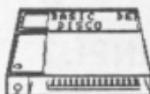
Observações: Todos os disquetes virgens devem ser formatados antes de serem usados.

Se um disquete já gravado for formatado novamente, todas as informações que ele continha serão apagadas.

Exemplo:

```
...FORMAT
```

GET



Sintaxe: GET[#]narq[,nregistro]

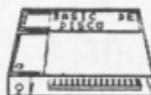
Função: Ler um registro de um arquivo aleatório.

Observações: narq deve ser o número do registro a ser lido.
Quando omitido, será lido o seguinte ao último já lido.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" AS #1 LEN=1
120 FIELD #1, 1 AS IN$
130 OPEN "A:DESTINO" AS #2 LEN=1
140 FIELD #2, 1 AS OU$
150 FOR F=1 TO LOF(1)
160   PRINTUSING"#####";F;
170   PRINT ,
180   GET#1,F
190   C$=IN$
200   IF ASC(C$)>31 THEN PRINT C$
210   IF ASC(C$)<32 THEN PRINT "<32"
220   LSET OU$=C$
230   PUT#2,F
240 NEXT F
250 CLOSE
```

INPUT



Sintaxe: INPUT # narq,var1[,var2] [,...]

Função: Ler dados de um arquivo sequencial.

Observações: narq deve ser o número de um arquivo aberto.

Os dados serão lidos sequencialmente e armazenados nas variáveis **var1**, **var2**, etc... que podem ser numéricas ou strings.

Os caracteres Carriage Return (&H0D) e Line Feed (&H0A) são ignorados. O caractere espaço (&H20) é ignorado em dados numéricos e no início de dados alfanuméricos.

Para indicar o final dos dados podemos usar o caractere vírgula (&H2C) ou um dos três citados acima. Se forem lidos 255 caracteres ou se o caractere ^Z (&H1A) for encontrado, o comando INPUT # será encerrado.

Para dados alfanuméricos, toda sequência de caracteres contida entre aspas (&H24) é considerada como um único dado.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" FOR INPUT AS #1
120 OPEN "A:DESTINO" FOR OUTPUT AS #2
130 IF EOF(1) THEN CLOSE : END
140 INPUT#1,A$
150 PRINT#2,A$ : F=F+1
160 PRINTUSING"#####";F+1;
170 PRINT,A$
180 GOTO 180
```

INPUT\$



Sintaxe: INPUT\$ (ncharacteres[.[#]narq])

Função: Receber uma string com um certo número de caracteres de um arquivo sequencial aberto para entrada.

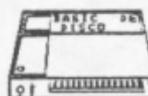
Observações: narq deve ser o número de um arquivo sequencial aberto para entrada.

Quando apenas ncharacteres for especificado, a leitura será feita do teclado.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" FOR INPUT AS #1
120 OPEN "A:DESTINO" FOR OUTPUT AS #2
130 IF EOF(1) THEN CLOSE : END
140 X$=INPUT$(1,#1)
150 PRINT #2,X$ : A=A+1
160 PRINT USING"#####";A+1;
170 IF ASC(X$)>31 THEN PRINT X$
180 IF ASC(X$)<32 THEN PRINT "< 32"
190 GOTO 130
```

KILL



Sintaxe: KILL "[drv]arq"

Função: Eliminar arquivos de um disquete.

Observações: arq é o arquivo a ser apagado e pode usar o "*" e o "?".

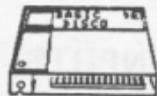
Se drv não for especificado, será assumido o drive corrente.

Não se pode apagar arquivo aberto por OPEN.

Exemplo:

```
KILL "A:TESTE*.*"
```

LINE INPUT



Sintaxe: LINE INPUT # narq,vars.

Função: Receber de um arquivo sequencial uma sequência com até 254 caracteres e passá-la para uma variável string.

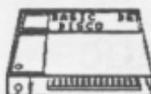
Observações: **narq** deve ser o número de um arquivo aberto.
vars é a variável string para a qual será transferida a sequência de caracteres lida.

Os caracteres CR (&H0D) e LF(&H0A) encerram a execução do comando.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" FOR INPUT AS #1
120 OPEN "A:DESTINO" FOR OUTPUT AS #2
130 IF EOF(1) THEN CLOSE : END
140 LINE INPUT#1,A$
150 PRINT#2,A$ : F=F+1
160 PRINTUSING"#####";F+1;
170 PRINT,A$
180 GOTO 130
```

LOAD



Sintaxe: LOAD "[dsp] [arq]"[.R]

Função: Transferir um programa BASIC de um disquete ou fita cassete para a memória do micro.

Observações: **arq** só será carregado se for programa BASIC ou se estiver gravado em formato ASCII.

Arquivos que não sejam programas BASIC mas que tenham sido gravados em ASCII poderão ser lidos, porém não serão executados.

Se **dsp** for CAS:, é essencial que o programa esteja no formato ASCII.

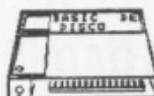
A opção **,R** faz com que o programa seja executado automaticamente assim que terminar a transferência.

Exemplo:

```
LOAD "PROGEXE.BAS
```

```
LOAD "PROG1.BAS
```

LOC



Sintaxe: LOC(narq)

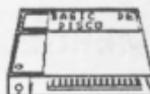
Função: Indicar o número do último registro acessado em um arquivo aleatório.

Observações: narq deve ser o número de um arquivo aleatório aberto.

Exemplo:

```
100 CLEAR 2000
110 MAXFILES=2
120 OPEN "A:FONTE" FOR INPUT AS #1
130 OPEN "A:DESTINO" AS #2 LEN=1
140 FIELD #2,1 AS R$
150 IF EOF(1) THEN 250
160 LINE INPUT#1,A$
170 FOR F=1 TO LEN(A$)
180     B$=MID$(A$,F)
190     LSET R$=B$ : PRINT R$;
200     PUT #2,LOC(2)+1
210 NEXT F
220 LSET R$=CHR$(13) : PRINT R$
230 PUT #2,LOC(2)+1
240 GOTO 150
250 LSET R$=CHR$(0) : PRINT R$
260 PUT #2,LOC(2)+1
270 CLOSE
```

LOF



Sintaxe: LOF(narq)

Função: Indicar quantos bytes existem num arquivo.

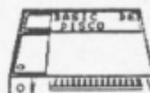
Observações: narq deve ser o número de um arquivo aberto.

Para saber quantos registros existem num arquivo, divida o valor obtido por LOF pelo tamanho de cada registro, especificado em LEN=.

Exemplo:

```
10 OPEN "A:TESTE" AS #1 LEN=128
20 T=LOF(1)/128
30 PRINT "Existem";T;"registros no arquivo."
40 CLOSE
```

LSET



Sintaxe: LSET vars = expressão string

Função: Alinhar à esquerda os dados nas variáveis do comando FIELD.

Observações: vars receberá os dados e normalmente é definida no comando FIELD.

LSET alinha os dados a esquerda. O caractere espaço em branco é usado para completar as posições vagas. Dados excedentes são simplesmente truncados.

Variáveis numéricas não podem ser alinhadas. Devem ser transformadas em variáveis strings (veja MKI\$, MKS\$ e MKD\$).

Exemplo:

```
10 OPEN "A:ROMMSX1" AS #1 LEN=1
20 FIELD #1,1 AS A$
30 FOR F=0 TO 16383
40 X=PEEK(F) : X$=MID$(STR$(X),2)
50 LSET A$=X$ : PUT #1,F+1
60 PRINT F,RIGHT$("00"+HEX$(X),2)
70 NEXT F
80 CLOSE
```

MERGE



Sintaxe: MERGE "{[dsp] [arq]}"

Função: Juntar programas gravados no formato ASCII com outro na memória do micro.

Observações: **dsp** pode ser CAS: ou qualquer drive.

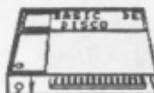
arq é o nome do programa gravado que será sobreposto ao programa que está na memória. As linhas do programa em disquete que tiverem o número igual ao de uma linha do programa no micro terão prioridade.

Exemplo:

```
MERGE "TESTE.ASC
```

```
MERGE "PRÓG1.ASC
```

MKIS, MKS\$, MKD\$



Sintaxe: MKIS (valor)
MKS\$(valor)
MKD\$(valor)

Função: Converter um valor numérico inteiro numa string de 2 bytes.

Converter um valor numérico de precisão simples numa string de 4 bytes.

Converter um valor numérico de precisão dupla numa string de 8 bytes.

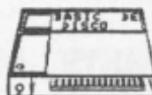
Observações: valor deve ser um número inteiro ou uma variável numérica inteira em MKIS; um número ou uma variável numérica de precisão simples em MKS\$ e um número ou uma variável numérica de precisão dupla em MKD\$.

Todos os valores numéricos devem ser convertidos para strings antes de serem gravados em arquivos (veja CVI, CVS, CVD).

Exemplo:

```
100 PI=4*ATN(1)
110 OPEN"A:HORRAD" AS #1 LEN=8
120 FIELD #1,8 AS O$
130 READ A$ : IF A$="FIM" THEN END
150 READ B$,C$
160 A=VAL(A$) : B=VAL(B$) : C=VAL(C$)
190 X#=(A+(B+C/60)/60)*PI/6
200 RSET O$=MKD$(X#)
210 PUT #1,LOC(1)+1
220 GOTO 130
230 REM Dados em h, min, s
240 DATA 23,45,16
250 DATA 12,36,55
260 DATA 01,27,11
270 DATA 00,51,04
280 DATA 03,45,46
290 DATA 33,21,26
300 DATA "FIM"
```

NAME



Sintaxe: NAME "[drv]arq1" AS "[drv]arq2"

Função: Alterar o nome de um arquivo em disquete.

Observações: **arq1** é o nome do arquivo cujo nome deve ser alterado.

arq2 é o nome com o qual ele deve ficar.

Ambos os arquivos devem estar no mesmo drive.

Se **drv** não for especificado será assumido o drive corrente.

Exemplo:

```
NAME "*.BAS" AS "*.ASC"
```

```
NAME "*.COM" AS "*.BIN"
```

```
NAME "*.OVL" AS "*.OVR"
```



OPEN

Sintaxe: OPEN "{[dsp] [arq]}" [FOR modo] AS [#] narq
[LEN=n]

Função: Inicializar um arquivo para o uso.

Observações: Se **dsp** for omitido, será assumido o drive corrente.

modo deve ser OUTPUT para indicar arquivo sequencial como saída, INPUT para especificar arquivo sequencial como entrada ou APPEND para especificar arquivo sequencial já existente como saída. Se **modo** for APPEND, os dados serão gravados após o último dado presente no arquivo do disquete, preservando os dados já gravados.

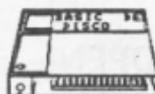
Se **modo** for omitido, o arquivo aberto será de acesso aleatório e o tamanho de seus registros deverá ser especificado pelo número **n**.

narq é o número com que o arquivo será aberto e não deve exceder o valor especificado por MAXFILES.

Exemplo:

```
100 CLEAR 2000
110 MAXFILES=2
120 OPEN "A:FONTE" FOR INPUT AS #1
130 OPEN "A:DESTINO" AS #2 LEN=1
140 FIELD #2,1 AS R$
150 IF EOF(1) THEN 250
160 LINE INPUT#1,A$
170 FOR F=1 TO LEN(A$)
180     B$=MID$(A$,F)
190     LSET R$=B$ : PRINT R$;
200     PUT #2,LOF(2)+1
210 NEXT F
220 LSET R$=CHR$(13) : PRINT R$
230 PUT #2,LOF(2)+1
240 GOTO 150
250 LSET R$=CHR$(0) : PRINT R$
260 PUT #2,LOF(2)+1
270 CLOSE
```

PRINT



Sintaxe: PRINT #narq,d1[,d2] [,d3] [...]

Função: Gravar dados num arquivo sequencial aberto como saída.

Observações: narq deve ser o número do arquivo sequencial aberto como saída em que se deseja gravar dados.

d1, d2, ..., etc são expressões numéricas ou strings.

O uso do USING é permitido com a mesma sintaxe com que é usado no comando PRINT.

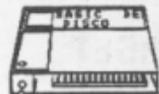
Dados numéricos devem ser separados pelo caractere ponto e vírgula (&H3B).

Dados strings devem estar separados pelo caractere vírgula (&H2C).

Se o caractere vírgula for um dado string, deve ser colocado entre aspas (&H22).

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" FOR INPUT AS #1
120 OPEN "A:DESTINO" FOR OUTPUT AS #2
130 IF EOF(1) THEN CLOSE : END
140 INPUT#1,A$
150 PRINT#2,A$ : F=F+1
160 PRINTUSING"#####";F+1;
170 PRINT,A$
180 GOTO 130
```



PUT

Sintaxe: PUT #narq,nregistro

Função: Gravar um registro no arquivo randômico.

Observações: narq deve ser o número de um arquivo de acesso aleatório aberto.

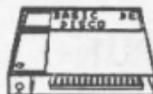
nregistro é o número do registro no qual os dados serão escritos.

É essencial que os dados a serem gravados estejam alinhados por LSET ou RSET.

Exemplo:

```
100 MAXFILES=2
110 OPEN "A:FONTE" AS #1 LEN=1
120 FIELD #1, 1 AS IN$
130 OPEN "A:DESTINO" AS #2 LEN=1
140 FIELD #2, 1 AS OU$
150 FOR F=1 TO LOF(1)
160   PRINT USING "#####";F;
170   PRINT ,
180   GET#1,F
190   C$=IN$
200   IF ASC(C$)>31 THEN PRINT C$
210   IF ASC(C$)<32 THEN PRINT "<32"
220   LSET OU$=C$
230   PUT#2,F
240 NEXT F
250 CLOSE
```

RSET



Sintaxe: RSET vars = expressão string

Função: Alinhar à direita os dados nas variáveis do comando FIELD.

Observações: vars receberá os dados e deve ter sido especificada pelo comando FIELD.

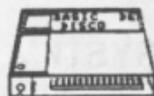
RSET alinha os dados a direita. O caractere espaço em branco é usado para completar as posições vagas. Dados excedentes são simplesmente truncados.

Variáveis numéricas não podem ser alinhadas. Devem ser transformadas em variáveis strings (veja MKIS, MKSS e MKDS).

Exemplo:

```
100 MAXFILES=3
110 OPEN "A:FONTE" AS #1 LEN=1
120 FIELD #1,1 AS A$
130 OPEN "A:DESTIN01" AS #2 LEN=3
140 FIELD #2,3 AS B$
150 OPEN "A:DESTIN02" AS #3 LEN=3
160 FIELD #3,3 AS C$
170 FOR F=1 TO LOF(1)
180     GET #1,F
190     X$=A$ : PRINT X$;"!";
200     LSET B$=X$ : PRINT B$;"!";
210     PUT #2,F
220     RSET C$=X$ : PRINT C$;"!"
230     PUT #3,F
240 NEXT F
250 CLOSE
```

RUN



Sintaxe: RUN ["[dsp]arq" [,R]] | [nlinha]

Função: Executar um programa BASIC.

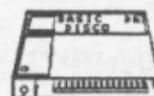
Observações: Se nenhuma opção for especificada, será executado o programa que estiver na memória do micro.
Se **dsp** for omitido será assumido o drive corrente.
arq deve ser obrigatoriamente um programa BASIC.
RUN fecha todos os arquivos abertos, a não ser que a opção **,R** seja especificada.

Exemplo:

```
RUN" AUTOEXEC.BAS
```

```
RUN" TESTE.BAS
```

SAVE



Sintaxe: SAVE {"[dsp] [arq]"} [,A]

Objetivo: Transferir um programa da memória do micro para um disquete ou fita cassete.

Observações: Se **dsp** não for especificado, será assumido o drive corrente.

arq deve ser necessariamente um programa BASIC.

Se **dsp** for CAS:, **arq** será gravado automaticamente no formato ASCII.

Se **ds;** for um drive, **arq** será normalmente gravado em binário condensado. Para que **arq** seja gravado em disquete no formato ASCII é necessário especificar a opção **,A**.

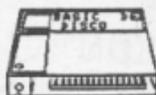
Exemplo:

```
SAVE" TESTE.BAS
```

```
SAVE" AUTOEXEC.BAS"
```

```
SAVE" PROG1.ASC",A
```

SYSTEM



Sintaxe: CALL SYSTEM
 _SYSTEMEM

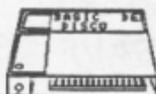
Função: Voltar do BASIC DE DISCO para o SOLX-DOS.

Observação: Para que se possa usar este comando é essencial que o SOLX-DOS tenha sido carregado e que o BASIC DE DISCO tenha sido chamado por ele.

Exemplo:

```
_SYSTEM
```

VARPTR



Sintaxe: VARPTR ({[[#] narq] | [variável]}¹)

Função: Fornecer o endereço de memória do início do buffer do arquivo especificado ou o endereço do início do armazenamento da variável especificada.

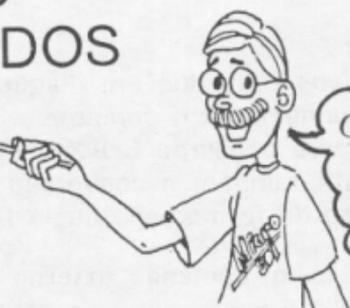
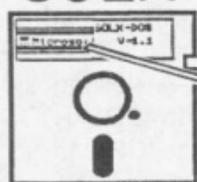
Observações: narq deve ser o número de um arquivo aberto.
variável pode ser numérica ou string.

Quando for especificada uma variável string, VARPTR retorna o endereço de seus apontadores.

Exemplo:

```
10 OPEN "A:ROMMSX1" AS #1 LEN=1
20 FIELD #1,1 AS A$
30 FOR F=0 TO 16383
40 GET #1,F+1 : X=VAL(A$)
50 IF PEEK(F)<>X THEN PRINT"ERRO"
60 NEXT F
70 PRINT "Buffer=&H";HEX$(VARPTR(#1))
80 CLOSE
```

DICIONÁRIO DO SOLX-DOS



O SOLX-DOS é
o Sistema
residente em
disquete!!!



APAGUE ou ERASE

Sintaxe: APAGUE arq
ERASE arq
DEL arq (válido apenas na versão 1.2)

Função: Eliminar arquivos do diretório de um disquete.

Observações: arq pode usar o "*" e o "?".

Exemplo:

```
A>APAGUE B:*.*BAS
```

BASIC

Sintaxe: BASIC [arq]

Função: Mudar do SOLX-DOS para o BASIC DE DISCO.

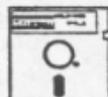
Observações: arq deve ser necessariamente um programa BASIC.

Quando especificado, o programa arq será automaticamente executado.

Veja também SYSTEM.

Exemplo:

```
A>BASIC
```



CONVSOL



Sintaxe: CONVSOL

Função: Converter arquivos gravados em disquetes no padrão SOL/M, S-700 ou qualquer outro compatível com CP/M de formatação similar para o padrão SOLX-DOS. A versão 2.0 do CONVSOL permite também a conversão do SOLX-DOS para o CP/M e o uso de formatação dupla face CP/M.

Observações: CONVSOL é um comando externo do SOLX-DOS.

Ao ser executado, ele apresenta alguns menus de opções na tela. O primeiro deles pede o tipo de CP/M a ser usado, que pode ser o SOL/M ou o S-700, face simples ou dupla. Nesse menu, a opção 4 permite retornar ao sistema.

Disquetes gravados em padrão APPLE não são convertidos, mesmo que estejam em formato CP/M!

O segundo menu pede a especificação do tipo de conversão: SOLX-DOS para CP/M ou vice-versa. Teclando 0 pode-se voltar ao primeiro menu.

Após a escolha do tipo de CP/M e do tipo de conversão, o CONVSOL pede a introdução do nome do arquivo a ser convertido. Se o nome for conhecido, digite-o com o nome do drive em que ele está (exemplo: B:PIP.COM). Caso contrário, para saber quais os arquivos presentes no disquete fonte, digite o nome do drive em que ele está (A: ou B:) seguido da palavra DIR.

Após fornecer o nome do arquivo a ser convertido, o CONVSOL pedirá o nome do arquivo destino a ser gerado. Forneça-o também com o nome do drive. Se o nome do arquivo a ser gerado não for especificado será assumido o nome do arquivo fonte. O nome do arquivo a ser convertido pode usar o "*" e o "?".

Se houver apenas um drive físico no sistema, será necessário trocar os disquetes. O CONVSOL pedirá a troca sempre que necessário. Veja em detalhes, passo a passo, como usar o CONVSOL no apêndice A.

NOTA: O SOL/M é a versão CP/M desenvolvida pela Microsol para o CP-500 e o S-700 é a versão CP/M para o Sistema 700 da Prológica.

Exemplo:

```
A>CONVSOL
```

COPIARQ



Sintaxe: COPIARQ [origem [PARA] | [,] destino [(opções)]]

Função: Transferir arquivos de um disquete para outro e/ou concatenar arquivos e/ou transferir arquivos para impressora.

Observações: COPIARQ é um comando externo e pode ser executado com parâmetros ou sem eles.

origem, destino e opções são especificações para escolha dos dispositivos e arquivos a serem transferidos ou concatenados e são absolutamente necessários para que o COPIARQ possa funcionar. Vamos comentá-los mais adiante. Caso **origem, destino e opções** não sejam especificados, o COPIARQ apresentará as seguintes mensagens na tela:

```
A>COPIARQ
COPIARQ - Microsol Ltda 1985 -
Versao 1.2 Solx-Dos 10-10-86
```



O ponto e o cursor indicam que o COPIARQ está a espera de **origem, destino e opções**. Como eles não foram declarados antes, precisam ser declarados agora. Caso se queira voltar ao SOLX-DOS basta teclar RETURN e o "A>" retornará à tela.

origem deve ser do tipo:

drv1arq1 [+drv2arq2] [+...] [+drvnarqn].

drv1, drv2, ..., drvn são os drives onde estão os arquivos **arq1, arq2, ..., arqn** a serem concatenados e/ou copiados. Pode-se usar o "*" e o "?" nos nomes dos arquivos.

destino deve ser do tipo:

dsp[arqd]

dsp é o dispositivo em que serão gerados os arquivos copiados ou concatenados e pode ser A:, B:, C:, D: ou F: (para o drive), ou ainda IMP: (para a impressora). **arqd** pode usar o "*" e o "?".

opções devem ser do tipo:

([o1] [,o2] [,...] [,on])

o1, o2, ..., on são opções e parâmetros que definem o tipo de cópia e devem estar sempre entre parênteses. Se for usado mais de uma, é necessário separá-las por vírgulas (.). A seguir estão especificadas as opções e seu funcionamento:

- E — Usada apenas quando **dsp** é algum drive. Normalmente o COPIARQ cria um arquivo provisório de nome RASCUNHO.SOL com os dados dos arquivos de **origem**. Ao terminar a cópia, se tudo correu bem, o arquivo RASCUNHO.SOL tem seu nome trocado pelo do arquivo **destino**. A opção E impede a criação do RASCUNHO.SOL e pode ser usada quando há pouco espaço para a cópia no disquete.
- V — Usada apenas quando **dsp** é algum drive. Esta opção faz com que apenas os arquivos que não existem no disquete **destino** sejam copiados.
- P — Êste parâmetro faz com que antes de copiar ou concatenar um arquivo, o COPIARQ peça confirmação ao usuário.
- M — Faz com que todas as letras maiúsculas dos arquivos de **origem** sejam substituídas por letras minúsculas.
- N — Faz com que todas as letras minúsculas dos arquivos de **origem** sejam substituídas por letras maiúsculas.
- Fn — Usada apenas para cópias na impressora, isto é, quando **dsp=IMP**. Define o tamanho de cada página de impressão através do número **n** ($0 \leq n \leq 255$). Se **n=0** todos os parâmetros de impressão serão ignorados. O valor default de **n** é 55.
- Tn — Usada apenas para cópias na impressora, isto é, quando **dsp=IMP**. Define o tamanho da tabulação (através do caractere &H09) em função do número **n** ($0 \leq n \leq 255$). Se **n=0** ou **n=255** a tabulação terá seu tamanho máximo que é de 256. O valor default de **n** é 8.
- L"texto" — Usada apenas para cópias na impressora, isto é, quando **dsp=IMP**. Define uma mensagem a ser impressa como cabeçalho em todas as páginas.
- R"texto" — Usada apenas para cópias na impressora, isto é, quando **dsp=IMP**. Define uma mensagem a ser impressa como rodapé em todas as páginas. **texto** é o texto do rodapé. No rodapé pode-se colocar

numeração automática de páginas. Para isso o texto deve ser uma seqüência de caracteres # sendo que a quantidade de # usada definirá a quantidade máxima de dígitos que poderá ter o número de uma página. Cada # corresponde a um dígito.

Se não for necessária a formatação, pode-se usar o comando COPIE para transferências entre o drive e a impressora.

Exemplo:

A:TESTE.COM PARA B:TESTE1.COM

Copia o arquivo TESTE.COM do drive A: para o drive B: com o nome TESTE1.COM.

A:TESTE.COM B:

Copia o arquivo TESTE.COM do drive A: para o drive B: sem alterar o nome.

A:*.BAS,B:

Copia todos os arquivos do tipo .BAS do drive A: para o drive B:

TEXT0.TXT PARA IMP:

Copia o arquivo TEXT0.TXT para a impressora.

***.TXT IMP:**

Copia todos os arquivos do tipo .TXT para impressora.

A:ARQ1.TXT+B:ARQ2.TXT+A:TEXT0.TXT B:

Nesse caso, o arquivo destino terá o nome do primeiro arquivo.

A:ARQ1.TXT+B:ARQ.TXT+A:TEXT0.TXT B:C.TXT

O arquivo C.TXT é a concatenação dos arquivos: ARO1.TXT, ARO.TXT e TEXT0.TXT.

A:*.* B:(P)

Resultado:

```
CONFIG.COM COPIA? (s/n/f): S
MBASIC.COM COPIA? (s/n/f): N
DDT.COM COPIA? (s/n/f): F
```

Respondendo "S", o programa copia o arquivo. Respondendo "N" o programa não copia e salta para o próximo arquivo (caso exista). Respondendo "F" o programa abandona a cópia. Se, por acaso, o arquivo fonte coincide com algum arquivo já existente no disco destino e se houver erro na gravação, o arquivo no disco destino será destruído com a utilização deste parâmetro.

*.TXT IMP: (R"Pagina: ### 18/07/85" T9)

Ao final de cada página, será impresso:

Página: 001 18/07/85; Página: 002 18/07/85; etc.

```
A>COPIARQ
COPIARQ - MICROSOL LTDA -
.B:DDT.COM A:
DDT.COM
.B:TEXT0.TXT IMP:
```

```
SAIDA PARA IMPRESSORA
TEXT0.TXT
```

A>

Para executar o programa repetidamente, isto é, com mais de uma linha de comando, se aconselha usar o "prompt". O "prompt" se consegue chamando o programa sem a linha de comando. Nesse caso, para se retornar ao sistema tecla RETURN no início do "prompt".

COPIE ou COPY



Sintaxe: COPY [dsp1] [arq1] [dsp2] [arq2] [[/V] | [/L]]

Função: Copiar arquivos.

Observações: **dsp1** é o dispositivo em que está o arquivo a ser copiado e pode ser qualquer drive ou o console (CON:). Usa-se **dsp1** como console para produzir arquivos "BATCH" e, nesse caso, após digitar o texto, é necessário digitar ^Z e RETURN para que o texto seja gravado.

dsp2 é o dispositivo em que está o arquivo onde será feita a cópia e pode ser qualquer drive, ou a impressora (PRT:).

arq1 é o nome do arquivo a ser copiado e pode usar o "*" e o "?".

arq2 é o arquivo que receberá a cópia e, se omitido, será considerado igual a **arq1**.

/V ou **/L** ativa a verificação. Mesmo que tenha sido dado um comando VERIFICA/D, estas opções ativam a verificação.

Pode-se copiar arquivos de um disquete para outro mesmo que haja apenas um drive físico no sistema. Nesse caso, o SOLX-DOS solicitará a troca de disquetes sempre que necessário e o drive conectado funcionará como drive lógico A: ou B: alternadamente.

Exemplo:

```
A>COPY CON A:TESTE.BAT
CON
REM ESTE PROGRAMA GERA O DIRETORIO
REM DO DISQUETE NO DRIVE CORRENTE E
REM EXEMPLIFICA O USO DO 'COPY' COM
REM O CONSOLE COMO DISPOSITIVO FONTE
PAUSA
DIR
^Z
```

1 Arquivo(s) copiado(s)

A>■

COPIED



Sintaxe: COPIED drv1 drv2

Função: Fazer uma cópia completa de um disquete.

Observações: **drv1** é o drive em que está o disquete a ser copiado.

drv2 é o drive em que está o disquete em que será feita a cópia.

Se existir apenas um drive no sistema, o COPIED pedirá a troca de disquetes sempre que necessária.

Para cópias num mesmo disquete, basta comandar COPIED.

Exemplo:

```
A>COPIED A: B:
```

DATA ou DATE



Sintaxe: DATA [dia-mês-ano] ou DATE [dia-mês-ano]
DATA [dia/mês/ano] ou DATE [dia/mês/ano]

Função: Apresentar ou mudar a data corrente do sistema.

Observações: O sinal separador entre o **dia**, o **mês** e o **ano** pode ser o "-" ou o "/".

Exemplo:

```
A>DATA 19-02-87
```

```
A>DATA 19/02/1987
```

DIR



Sintaxe: DIR [drv] [arq] [/P] [/W]

Função: Apresentar os arquivos existentes num disquete.

Observações: Se **drv** for omitido será considerado o drive corrente.

Se **arq** for omitido, serão mostrados todos os arquivos presentes no disquete. O uso do "*" e do "?" é permitido em **arq**.

A a opção **/W** faz com que somente os nomes dos arquivos sejam mostrados.

A opção **/P** faz com que a listagem dos arquivos seja interrompida a espera do pressionamento de uma tecla sempre que a tela ficar cheia.

No SOLX-DOS, o caractere &H20 (espaço em branco) pode funcionar de forma semelhante ao "*". Por exemplo, o comando "DIR" é equivalente a "DIR *.*" e o comando "DIR TESTE" é equivalente a "DIR TESTE.*".

Exemplo:

```
A>DIR A:*.*COM/W
```

```
A>DIR A:*.*BAT/W
```

```
A>DIR A:C*.*
```

```
A>DIR B:/P
```

FORMATE ou FORMAT



Sintaxe: FORMATE [[drv] [n]]
 FORMAT [[drv] [n]]

Função: Formatar um disquete para uso do SOLX-DOS ou do BASIC DE DISCO.

Observações: Todos os disquetes virgens devem ser formatados antes de serem usados. Após a execução do comando FORMATE o disquete ficará sem nenhum arquivo.

drv é o drive em que está o disquete a ser formatado e deve ser A: ou B: se houver apenas uma interface no sistema.

Caso haja mais de uma interface, pode ser C: ou D:

Atenção: Não é possível conectar duas interfaces CDX-2 no mesmo micro simultaneamente!

n indica o tipo de formatação a ser feita. A interface da Microsol pode fazer os seguintes tipos de formatação:

- 1 — 40 trilhas simples face
- 2 — 40 trilhas dupla face
- 3 — 80 trilhas simples face
- 4 — 80 trilhas dupla face

Disquetes de 5 1/4" devem ser formatados com 40 trilhas!
Disquetes de 3 1/2" devem ser formatados com 80 trilhas!

Para poder gerenciar drives de 3 1/2" a interface CDX-2 reserva cerca de 1 Kbyte a mais da memória RAM do micro que outras interfaces específicas para drives só de 5 1/4" ou só de 3 1/2".

Caso **drv** e **n** não sejam especificados, o SOLX-DOS pedirá a escolha de ambos através de menus.

Exemplo:

```
A>FORMATE B: 2
```

LISTE, LIST ou TYPE



Sintaxe: LISTE [drv] arq
LIST [drv] arq
TYPE [drv] arq

Função: Apresentar o conteúdo de um arquivo na tela e na impressora.

Observações: Se **drv** não for especificado será assumido o drive corrente.

arq deve estar gravado no formato ASCII.

Para fazer com que a apresentação seja feita também na impressora deve-se digitar ^P. Para interromper a apresentação no vídeo ou retomá-la deve-se usar ^S.

Exemplo:

```
A>LISTE COPSIS.BAT
```

```
A>LISTE COPSISD.BAT
```

MODO ou MODE



Sintaxe: MODO n ou
MODE n

Função: Definir a quantidade máxima de caracteres que poderão existir numa linha do vídeo.

Observações: n pode variar de 32 a 40 no MSX modelo 1 sem placa de 80 colunas. No MSX modelo 2 ou no modelo 1 com placa de 80 colunas n pode ir de 32 a 80.

Exemplo:

```
A>MODO 32
```

```
A>MODO 39
```

PAUSA ou PAUSE



Sintaxe: PAUSA [comentários]
PAUSE [comentários]

Função: Interromper a execução de um "BATCH" para mostrar uma mensagem na tela.

Observações: Quando PAUSE é executado dentro de um "BATCH", a execução é interrompida e a mensagem (se especificada) é mostrada na tela. Para continuar a execução dos comandos do BATCH basta pressionar uma tecla qualquer.

Exemplo:

```
A>PAUSA Ligue a impressora !
```

REM



Sintaxe: REM [comentários]

Função: Apresentar mensagens na tela sem interromper a execução do "BATCH".

Observações: Os caracteres vírgula (&H2C), espaço (&H20) e tabulador (&H09) podem ser usados como separadores.

Exemplo:

```
REM Lembre-se de gravar  
REM este programa com a  
REM extensao '.BAT'  
PAUSE
```

REN ou RENAME



Sintaxe: REN [drv] arq1 arq2
RENAME [drv] arq1 arq2

Função: Alterar o nome de arquivos num disquete.

Observações: Se **drv** não for especificado, será assumido o drive corrente.

arq1 é o arquivo a ter o nome trocado e pode usar o "*" e o "?".

arq2 é o novo nome que será atribuído a **arq1** e pode usar o "*" e o "?".

Exemplo:

```
A>RENAME A:DUMPSOL.COM DUMP.COM
```

```
A>RENAME A:*. * *.COM
```

```
A>RENAME A:*.BAS *.ASC
```

```
A>RENAME A:*.BIN *.ASM
```

```
A>RENAME A:*.OVL *.OVR
```

SALVE ou SAVE



Sintaxe: SALVE n arq ou
 SAVE n arq

Função: Gravar uma região da memória num disquete.

Observações: n é o número de blocos de 256 bytes a serem gravados.

Para determinar o valor de n proceda da seguinte forma: Subtraia do, endereço final do programa, o valor &H100 (ou 256 em decimal) correspondente ao início do programa. Isso lhe dará o comprimento, em bytes, do programa.

Então, divida o resultado por &H100 (ou por 256 se quiser fazer a conta em decimal). O resultado obtido já será a quantidade de blocos a serem gravados.

O valor n, entretanto, deve necessariamente estar em decimal. Se o cálculo foi feito em hexadecimal, converta o resultado para decimal.

arq é o nome do arquivo que será gerado e onde os dados serão gravados.

Não se deve usar o (*) e o "?" em **arq** ! Isso poderá causar sérios problemas.

SALVE deve ser executado antes que qualquer outro comando transiente, caso contrário a área de programas (TPA) poderá ser alterada. Veja o apêndice B.

Exemplo:

```
A>SALVE 10 teste.com  
A>SALVE 3 MODIFICA.COM
```

TIME



Sintaxe: TIME [h:min:s]

Função: Apresentar e alterar o instante indicado pelo relógio interno do micro.

Observações: TIME só funciona na versão 1.2 do SOLX-DOS e em micros nos quais existam relógios internos.

Nos micros MSX que não possuam relógio interno, os valores fornecidos por TIME serão sempre 00:00:00.

h pode ser qualquer número entre 0 e 23 ou qualquer número entre 1 e 12 seguido de A ou P.

min pode ser qualquer número entre 0 e 59.

s pode ser qualquer número entre 0 e 59.

Para não alterar o instante, basta pressionar RETURN.

Exemplo:

VERIFICA ou VERIFY



Sintaxe: VERIFICA ([/L | ON] | [/D | OFF])
VERIFY ([/L | ON] | [/D | OFF])

Função: Ativar ou desativar a verificação de gravação.

Observações: VERIFY ON ou VERIFY /L ativa a verificação, fazendo com que os dados, após serem gravados no disquete, sejam lidos e verificados. A leitura de verificação é efetuada após a gravação de cada setor.

VERIFY OFF ou VERIFY /D desativa a verificação.

Exemplo:

```
A>VERIFICA/L
A>COPY A:*.*BAS B:
A>VERIFICA/D
```

APÊNDICES



Nas páginas a seguir estão comentados assuntos específicos ou cuja importância merece destaque.

O primeiro apêndice é sobre o CONVSOL e explica passo-a-passo como usá-lo para converter arquivos gravados sob CP/M para o formato do SOLX-DOS.

O segundo é um apêndice sobre características técnicas do equipamento ou do sistema SOLX-DOS.

O terceiro apêndice mostra detalhadamente o procedimento para se fazer um "backup" do disquete do SOLX-DOS mesmo sem saber usá-lo.

Finalmente, o quarto apêndice contém um resumo das mensagens de erro do BASIC DE DISCO e do SOLX-DOS.

APÊNDICE A — CONVERSÃO DE ARQUIVOS	101
APÊNDICE B — CARACTERÍSTICAS TÉCNICAS ..	104
APÊNDICE C — CÓPIA DO SOLX-DOS	107
APÊNDICE D — MENSAGENS DE ERRO	108

APÊNDICE A - CONVERSÃO DE ARQUIVOS

A seguir, descreve-se passo-a-passo o procedimento necessário para se converter arquivos do SOL/M (compatível com o CP/M) para o SOLX-DOS (compatível com o MSX-DOS).

Para exemplificação, considera-se que o equipamento esteja devidamente instalado com apenas uma interface conectada e operando com o SOLX-DOS no drive A.

CONVERSÃO CP/M PARA SOLX-DOS

1) Com o disquete do SOLX-DOS no drive A, comande:
A>CONVSOL

2) Vamos supor que se queira converter um disquete gravado no formato SOL/M com 40 trilhas em face dupla. Nesse caso, deve-se digitar a tecla do número um. Na parte inferior da tela deve aparecer um segundo menu,

3) Agora, deve-se fazer a opção pela conversão do CP/M para o SOLX-DOS, pressionando para isso a tecla do número 1.

O CONVSOL está pedindo o nome do arquivo a ser convertido (juntamente com o drive em que ele está, que em nosso caso é necessariamente o drive A). Vamos supor que não sabemos ou não lembramos do nome.

4) Agora, retire o disquete do SOLX-DOS do drive e insira o disquete gravado em CP/M.

5) Para ver quais os arquivos presentes no disquete gravado em CP/M, basta digitarmos:

A:DIR

Na tela deverão aparecer os nomes dos arquivos gravados em CP/M e, na parte inferior, o CONVSOL estará pedindo novamente o nome do arquivo a ser convertido. Vamos supor que no disquete exista um arquivo de nome DUMP.COM e que seja ele o que queremos converter.

6) Devemos simplesmente digitar seu nome logo após o nome do drive:

A:DUMP.COM

O CONVSOL pedirá, então, o nome do arquivo a ser gerado (junto com o drive!).

7) Digite, por exemplo:

A:DUMPSOL.COM

O CONVSOL mostrará uma mensagem na tela pedindo para que seja inserido o disco do SOLX-DOS no drive.

- 8) Retire o disquete gravado em CP/M, insira o do SOLX-DOS no drive A e pressione qualquer tecla. Na tela, deverá surgir o nome do arquivo gravado em CP/M e outra mensagem pedindo que seja feita uma nova troca dos disquetes.
- 9) Retire o disquete do SOLX-DOS do drive e insira o disquete gravado em CP/M. A seguir pressione alguma tecla. O CONVSOL pedirá mais uma vez a troca de disquetes.
- 10) Faça a troca de disquetes quantas vezes forem necessárias.

Agora, para certificar-se que a conversão foi executada, comande DIR.

Para converter arquivos do SOLX-DOS para o CP/M, deve-se proceder de forma análoga a descrita.

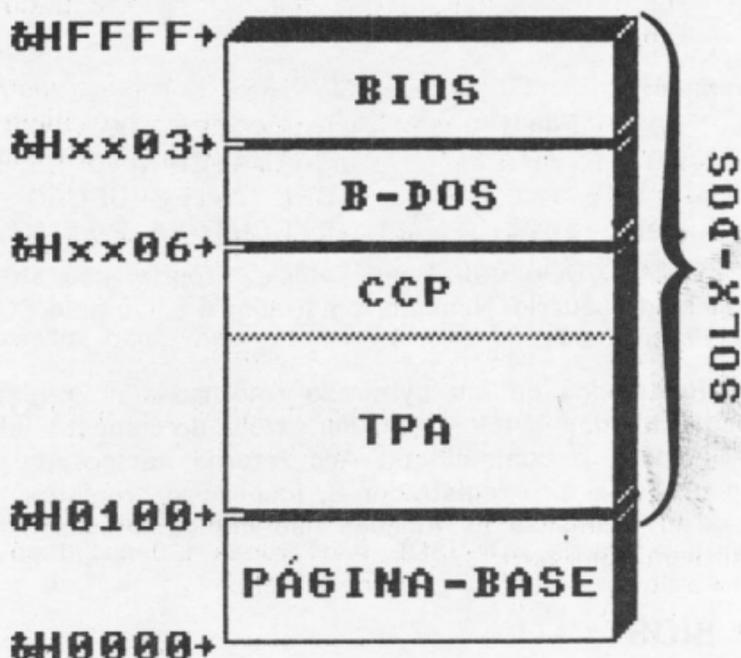
Os procedimentos descritos neste apêndice são válidos para a versão 2.0 do CONVSOL. As versões 1.0 e 1.1 só aceitam leitura de disquetes CP/M face simples.

APÊNDICE B - CARACTERÍSTICAS TÉCNICAS

Este apêndice se destina a programadores mais avançados que tenham experiência em sistemas. Se esse não é o seu caso, as informações aqui descritas podem ser ignoradas.

ESTRUTURA DO SOLX-DOS

O SOLX-DOS possui o seguinte mapeamento de memória:



As informações contidas na página-base, são:

Endereço	Função
0000-0002	Vetor para o "warm-boot" do BIOS.
0005-0007	Vetor para o B-DOS.
0030-0032	Vetor para chamada a rotinas em outros "slots".
005C-0068	Primeiro FCB do SOLX-DOS.
006C-007F	Segundo FCB do SOLX-DOS.
0080-00FF	Linha de comando e endereço "default" do buffer de transferência.

CHAMADAS A FUNÇÕES DO SOLX-DOS

A fim de possibilitar a execução de programas desenvolvidos para CP/M-80, as funções executadas pelo SOLX-DOS são, em sua maioria, compatíveis com as do CP/M versão 2.2.

O procedimento básico para a chamada de qualquer função é o seguinte:

- 1 — Carregar o registrador C com o número da função.
- 2 — Carregar, quando requisitado, o registrador E com um valor de 8 bits ou o par DE com um valor de 16 bits.
- 3 — Executar a instrução "CALL 5".

Exemplo:

```
LD E,41H ; CHARACTER 'A'  
LD C,6H ; SAIDA PARA O CONSOLE  
; SEM INTERRUPTAO  
CALL 0005H ; EXECUTA A ROTINA
```

O SOLX-DOS utiliza um "stack" próprio, não alterando o "stack" do usuário. Nenhum registrador é salvo pelo SOLX-DOS, incluindo IX e IY.

Resultados de um byte são retornados no registrador A (Acc) e, de dois bytes em HL. Por razões de compatibilidade com o CP/M-80, o conteúdo de Acc retorna sempre igual ao do registrador L e, o registrador B, igual ao do registrador H. Em caso de chamadas as funções não implementadas, Acc e HL retornam com zero.

O BIOS

O SOLX-DOS possui um PSEUDO-BIOS que se organiza da seguinte maneira:

```
XX03 JP WBOOT ;PARTIDA A QUENTE (WARM-BOOT)  
XX05 JP CONST ;ESTADO DO CONSOLE (CONSOLE STATUS)  
XX09 JP CONIN ;ENTRADA DE CONSOLE (CONSOLE INPUT)  
XX0C JP CONOUT ;SAÍDA PARA O CONSOLE (CONSOLE OUT  
PUT)  
XX0F JP LIST ;SAÍDA PARA A IMPRESSORA (LIST) — V.1.2
```

Nota: O último vetor (LIST) foi implementado na versão 1.2 para permitir o funcionamento total de alguns programas CP/M que a acessam (como o MBASIC).

O endereço do BIOS varia de acordo com o número de drives do sistema. Para conseguir a base para o BIOS, pegue o endereço da posição de memória 0001H. Este endereço aponta para a primeira rotina (WBOOT). As demais rotinas têm os LSBs como indicado acima.

Ao contrário do CP/M, para ler e gravar setores não se pode usar o BIOS.

WBOOT — Todos os "buffers" de arquivos são descarregados. Arquivos que foram gravados e não fechados figurarão incorretamente no diretório. Um "JMP 0" desvia para esta rotina. O SOLX-DOS assume o controle.

CONST — Verifica o estado do teclado. Se alguma tecla estiver pronta, Acc retorna com FFH, caso contrário com zero.

CONIN — Espera que um caractere seja teclado, retornando com o mesmo em Acc.

CONOUT — Na entrada, o registrador C deve vir com a caractere a ser enviado para a tela.

LIST — Envia para a impressora o caractere em C.

FORMATO DO FCB

Byte	Função
0	Número do drive: 0="default", 1=A, 2=B, etc.
1-8	Nome do arquivo alinhado à esquerda.
9-B	Extensão do arquivo.
C	Campo de extensão (arquivos seqüenciais).
D	S1. Reservado
E	S2. LSB do tamanho do bloco, também usado como registro seqüencial corrente.
F	MSB do tamanho do bloco, também usado como registro seqüencial corrente.
10-13	Tamanho do arquivo em bytes.
14-15	Data da criação ou modificação do arquivo: Bits F-9 (ano) = 0-119 (1980-2099) Bits 8-5 (mês) = 1-12 Bits 4-0 (dia) = 1-31.

- 16-17 Horário da criação ou modificação do arquivo:
 Bits F-B (horas) = 0-23
 Bits A-5 (minutos) = 0-59
 Bits 4-0 (segundos) = 0-29 (seg/2).
- 18-1F Reservados.
- 20 Registro relativo ~~dentro da~~ extensão (0-127). Usado por arquivos seqüenciais (este byte não é iniciado na abertura).
- 21-24 Número do registro randômico.

CÓDIGOS DE CONTROLE DO VÍDEO

Os códigos de controle do vídeo (Terminal VT52) são os seguintes:

ASCII	HEXA	RESULTADO
ESC,j	1B,6A	Limpa a tela (CLS)
ESC,E	1B,45	Limpa a tela (CLS)
LF	0C	Limpa a tela (CLS)
ESC,K	1B,4B	Limpa do cursor ao fim da linha
ESC,J	1B,4A	Limpa do cursor ao fim da tela
ESC,I	1B,6C	Limpa a linha inteira
ESC,L	1B,4C	Insera uma linha
ESC,M	1B,4D	Elimina uma linha
ESC,Y,SP+L,SP+C	1B,59,20+L,20+C	Posiciona cursor
ESC,A	1B,41	Cursor linha acima
ESC,B	1B,42	Cursor linha abaixo
ESC,C	1B,43	Cursor à direita.
ESC,D	1B,44	Cursor à esquerda
ESC,H	1B,48	Topo da tela (HOME)
ESC,x,4	1B,78,34	Cursor cheio
ESC,y,4	1B,79,34	Cursor sublinhado
ESC,x,5	1B,78,35	Cursor ativo
ESC,y,5	1B,79,35	Cursor desativo

APÊNDICE C - CÓPIA DO SOLX-DOS

A seguir descreve-se passo a passo o procedimento para fazer uma cópia do SOLX-DOS para outro disquete.

- 1) Instale o equipamento como descrito no capítulo 2, com o disquete do SOLX-DOS no drive;
- 2) Entre no SOLX-DOS e a seguir retire o disquete do drive;
- 3) Arrume um disquete de 5¼" dupla face que esteja virgem ou cujo conteúdo possa ser apagado e insira-o no drive "A:";
- 4) Formate o disquete com o comando FORMAT. Com o "A>" no vídeo, digite FORMAT e a tecla RETURN. Digite a tecla da letra "A" e a tecla do número "2". Digite uma vez a tecla RETURN. Após alguns instantes o disquete estará formatado e pronto para ser usado;
- 6) Digite:

```
COPY *.* B:
```

- 7) Quando o sistema pedir para inserir um disquete no drive "B:", retire o disquete do SOLX-DOS do drive e insira nele o disquete recém formatado;
- 8) Digite a tecla RETURN;
- 9) Quando o "A>" surgir, para verificar se a cópia foi feita, digite DIR e duas vezes a tecla RETURN.

Se tudo correu bem, devem ter sido copiados 5 arquivos e seus nomes devem estar na tela do vídeo.

Agora a cópia já deve estar pronta.

APÊNDICE D - MENSAGENS DE ERRO

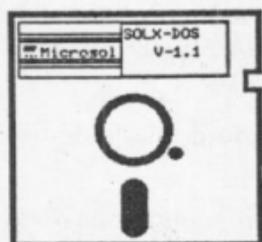
MENSAGENS DE ERRO DO BASIC-DISCO

CÓDIGO	MENSAGEM	SIGNIFICADO
60	Erro na FAT	O disquete em uso não está corretamente formatado.
61	Acesso incorreto	Houve uma tentativa de utilizar PUT, GET ou LOF com um arquivo seqüencial, LOAD com um arquivo aleatório ou OPEN de modo inválido.
62	Drive inválido	O drive foi especificado incorretamente.
63	Setor inválido	Foi tentado o acesso a um setor inexistente.
64	Arquivo já aberto	O arquivo foi aberto mais de uma vez.
65	Arquivo já existente	O novo nome do arquivo já existe (NAME).
66	Disco cheio	O disquete não suporta mais informações.
67	Diretório cheio	O diretório não suporta a criação do arquivo.
68	Disco protegido	O disquete está protegido contra gravação.
69	Erro de E/S	Ocorreu um erro irrecuperável durante a gravação.
70	Disco desconectado	O drive está desalinhado ou desconectado.
71	Erro ao renomear	Houve uma especificação de drives diferentes na renomeação (NAME).

MENSAGENS DE ERRO DO SOLX-DOS

MENSAGEM	SIGNIFICADO
Arquivo não encontrado	— Ocorre quando se tenta acessar um arquivo que não existe.
Erro ao renomear arquivo	— Surge quando o novo nome já existe ou o nome antigo não existe no uso do RENAME.
Erro ao apagar arquivo	— Indica que o arquivo não pôde ser apagado (ou não existe).
Drive inválido	— O drive selecionado está fora do sistema.
Parâmetro ilegal	— Acusa erro na sintaxe de um comando que necessita de parâmetro (VERIFICA, SALVE, FORMATE, MODO).
Erro na formatação	— Indica que o disquete não pôde ser formatado.
Erro interno (FAT)	— Significa que o disquete não está formatado perfeitamente.
Erro de leitura	— Informa um erro durante a leitura do disquete fonte (COPIED).
Data inválida	— Houve um erro na especificação da data.
Erro de gravação	— Um registro não pôde ser gravado.
Diretório cheio	— O diretório não suporta a criação de um novo arquivo.
Arquivo grande	— O arquivo não pôde ser carregado.
Comando ou arquivo inexistente	— O arquivo chamado não existe nas formas ".COM" ou "BAT".

*Lembre-se!
Para proteger um
disquete, impedindo
gravações posteriores,
feche este entalhe
com uma etiqueta!*



Se você quiser mais informações técnicas sobre o equipamento ou sobre o SOLX-DOS, escreva para

MICROSOL TECNOLOGIA LTDA.

Serviço de Apoio ao Usuário

Cx. Postal 350 - CEP 60000

Fortaleza - CE

Procure ser o mais claro possível, especificando cada uma de suas dúvidas separadamente!

Caso tenha problemas com o disquete do SOLX-DOS, remeta-o para nós. Isso tornará a resposta mais rápida!



Para receber gratuitamente o boletim informativo da ALEPH, contendo dicas de programação, artigos técnicos e informações sobre os últimos lançamentos para seu micro, envie seu nome e endereço completos (incluindo o CEP) para:

EDITORA ALEPH
Caixa Postal: 20.707
01498 São Paulo SP



Este livro foi:

composto na
Linotipadora Relâmpago Ltda.
Rua Vitória, 162/166
Sta. Efigênia - São Paulo
Fone: 223-1755



impresso na
press grafic
editora e gráfica ltda.

Rua Barra do Tibagi, 444 - Bom Retiro
Cep 01128 - Telefone: 221-8317

Para permitir um total controle dos usuários sobre os disk drives da linha MSX, a Microsol desenvolveu o SOLX-DOS, um poderosíssimo sistema operacional compatível com o MSX-DOS e semi-compatível com o CP/M e com o MS-DOS.

Juntamente com o BASIC DE DISCO, esse sistema é explicado detalhadamente neste livro, onde os autores comentam e exemplificam cada comando e cada função de forma clara e didática.

Além disso, noções gerais sobre o equipamento, suas aplicações e características técnicas são fornecidas ao longo do texto ou em apêndices.



ALEPH Publicações e
Assessoria Pedagógica Ltda.