

IBM PC — Guia do Operador

Comandos Básicos

CIP-Brasil. Catalogação - na - Publicação
Câmara Brasileira do Livro, SP

W719i Wilson, David A., 1943.
 IBM PC : guia do operador : comandos
 básicos / David A. Wilson ; tradução Nelson
 Loureiro Alves ; revisão técnica Carlos de
 Oliveira Gomes . - São Paulo : McGraw-Hill
 do Brasil, 1985.

1. IBM Computador Pessoal I. Título.

85-1350

17. CDD-651.8

18. -001.6404

Índices para catálogo sistemático ;

1. IBM Computador Pessoal : Computadores
eletrônicos Processamento de dados 651.8 (17.)
 001.6404 (18.)

IBM PC — Guia do Operador Comandos Básicos

David A. Wilson

Tradução :

Nelson Loureiro Alves

Engenheiro Eletrônico

Analista de Software

Revisão Técnica :

Carlos de Oliveira Gomes

Analista de Sistemas

Engenheiro Eletricista – PUC-RJ

McGraw-Hill

São Paulo

Rua Tabapuã, 1.105, Itaim-Bibi

CEP 04533

(011) 881-8604 e (011) 881-8528

*Rio de Janeiro • Lisboa • Porto • Bogotá • Buenos Aires •
Guatemala • Madrid • México • New York • Panamá • San
Juan • Santiago*

*Auckland • Hamburg • Johannesburg • Kuala Lumpur •
London • Montreal • New Delhi • Paris • Singapore •
Sydney • Tokyo • Toronto*

Do original

IBM® PC DiskGuide™

Copyright © 1983 by McGraw-Hill, Inc.

Copyright © 1985 da Editora McGraw-Hill do
Brasil, Ltda.

Todos os direitos para a língua portuguesa reservados
pela Editora McGraw-Hill do Brasil, Ltda.

Nenhuma parte desta publicação poderá ser reproduzida, guardada pelo sistema “retrieval” ou transmitida de qualquer modo ou por qualquer outro meio, seja este eletrônico, mecânico, de fotocópia, de gravação, ou outros, sem prévia autorização, por escrito, da Editora.

Editor: Ricardo Reinprecht

Coordenadora de Revisão: Daisy Pereira Daniel

Supervisor de Produção: Edson Sant’Anna

SUMÁRIO

Sistema Operacional em Disco

Arquivos no Disquete do DOS	1
Formatação do Disquete	2
Cópias	3
DOS Teclado	4
Sumário dos comandos do DOS	6
Processamento em Lotes	9
Nomes de Arquivos	10
EDLIN	11

Linguagem BASIC

BASIC Teclado	13
Comandos de Arquivo de Programa	15

Programação

Desenvolvimento de Programas	18
Palavras Reservadas	20
Comandos de Inicialização	21
Constantes e Variáveis	22
Funções Matemáticas	23
Programação Geral	25
Conversão de Dados	27
Operadores	28
Funções de String	28

Entrada e Saída do Monitor	
Modos de Texto e de Gráficos	30
Monitor no Modo Texto	31
Gráficos/Cores	35
Entrada de Teclado	43
Entrada/Saída de Periféricos	
Entrada/Saída de Periféricos em Geral	45
Comunicações.	46
Controle de Impressora.	49
Adaptador para Jogos.	50
Caneta Luminosa.	51
Porta Lógica de Entrada/Saída	53
Motor de Gravador Cassete	53
Miscelânea	
Som	53
Linguagem de Máquina.	55
Arquivos de Dados.	57
Detecção de Erro.	62
Erro: Mensagens e Números.	64
Códigos de Teclado Expandido	65
Conjunto de Caracteres ASCII	66
Índice Geral dos Comandos em BASIC	68

Informação Geral

- Para copiar um disquete, veja página 3.
- Para usar o Sistema Operacional em Disco (DOS), veja página 5.
- Para programar em BASIC, use o Sumário para localizar um assunto específico, ou use o Índice de palavras reservadas do BASIC para achar a palavra-chave específica.
- Palavras-chave em DOS e BASIC, bem como suas principais definições, estão em **negrito**, enquanto *itens opcionais estão em itálico*.
- Características disponíveis somente em Disco ou BASIC Avançado são precedidas por **1** .
Características disponíveis somente em BASIC Avançado são precedidas por **2** .
- Este guia refere-se às versões 1.1 do DOS e do BASIC.

Arquivos no Disquete do DOS

Somente COMMAND.COM aparece no Diretório

1. Registro de Início Em todos os disquetes formatados; carrega o resto do DOS em RAM

- | | |
|----------------|--|
| 2. IBMBIO.COM | Programa de Entrada/Saída; no disquete, somente se usar FORMAT/S |
| 3. IBMDOS.COM | Arquivo gerenciador de programa; no disquete, somente se usar FORMAT/S |
| 4. COMMAND.COM | Aceita comandos DOS; no disquete, somente se usar FORMAT/S |

Formatação do Disquete

Disquetes novos precisam ser formatados antes de serem usados para programas ou dados. Use disquete de dupla densidade, setorizado, de 5¼”.

Formatação:

- Apaga o conteúdo do disquete
- Checa o disquete inteiro à procura de trilhas ruins – aloca as trilhas ruins
- Inicializa o diretório
- Copia os arquivos do sistema operacional se FORMAT/S for usado; necessário para executar qualquer comando DOS, incluindo Restauração do Sistema

Procedimento:

1. Coloque o disquete do DOS no acionador A;
o prompt **A>** aparecerá na tela
2. Digite **FORMAT B:** ou **FORMAT B:/S** (para um acionador: **FORMAT** ou **FORMAT/S**)
3. Insira o disquete novo no acionador B: (ou A: se tiver somente um acionador)
4. Pressione qualquer tecla

Cópias

Copia todos os novos programas imediatamente, exceto os que estão protegidos. Use **DISKCOPY** para copiar trilha por trilha; use **COPY *.*** para copiar e reorganizar arquivos fragmentados. **DISKCOPY** formatará o disquete de destino, se necessário.

Procedimento (sistema com dois acionadores):

1. Coloque o disquete do DOS no acionador A
2. **A>DISKCOPY B:**
3. Após este procedimento, coloque o disquete-fonte no acionador A e o disquete-destino no acionador B; pressione qualquer tecla
4. Aguarde a mensagem "Copy complete"
5. **A>DISKCOMP B:**

Procedimento (sistema com um acionador):

1. Coloque o disquete do DOS no acionador A
2. **A>DISKCOPY**
3. Coloque o disquete-fonte no acionador A, e pressione qualquer tecla
4. Quando pronto, remova o disquete-fonte; coloque o disquete-destino; pressione qualquer tecla
5. *Repita os passos 3 e 4 até a mensagem "Copy complete" aparecer*
6. **A>DISKCOMP**
7. Repita os passos 3 e 4

DOS Teclado

Usado para edição e controle no DOS e EDLIN

Teclas de Edição

Tecla	Exemplo	Função	Resultado
	(linha informada previamente >Q12345.XY)		
F1 ou ☐	Exibe o próximo caractere	>Q1234 F1	>Q12345
F2	Exibe desde o primeiro caractere até o n.º dado	F2 4	>Q123
F3	Copia os caracteres restantes	>Q123 F3	>Q12345.XY
F4	Salta até o caractere escolhido	F4 5 F3	>5.XY
F5	Salva, mas não executa	F5	>Q12345.XY@
←	Apaga o último caractere	>Q123 ←	>Q12
Del	Salta um caractere	>Q1 Del F3	>Q1345.XY
Ins	Modo de inserção ativo/desativo		
Esc	Cancela a linha atual	>Q1234 Esc	>Q1234
Ctrl Break	Pára a operação atual		

Outras Teclas

Teclas	Função
F6	Coloca a marca de Fim de arquivo (^Z) quando da criação de um arquivo de lote
F6	Delimita séries de caracteres para Recolocação em EDLIN
Ctrl Num Lock Qualquer outra tecla	Pausa na operação atual Resume a operação atual
Ctrl Alt Del	Restaura o sistema (sempre do acionador A)
Ctrl PrtSc	Ecoa ativo/desativo , toda a mensagem do vídeo vai para a impressora (somente no DOS)
↑ PrtSc	Copia a tela atual na impressora (DOS ou BASIC)

Sumário dos Comandos do DOS

Comandos internos, disquete do DOS não é necessário

A>B: Escolhe acionador B: como novo acionador default

A>COPY PROG.ABC B:NOVONOME.XY
Copia arquivo de A: para B: com um nome novo

A>COPY PROG2.BAS B:/V
Copia de A: para B: com o mesmo nome e faz a *verificação*

A>COPY A.BAT+*.TXT GRD.BAT
Combina arquivos múltiplos em GRD. BAT; use /A para arquivos em ASCII (cópias até ^Z; adicione ^Z); /B para arquivos binários

A>DATE 07-84-1984 mm-dd-aaaa ou mm/dd/aaaa; 1980-2099

A>DIR*.COM
Lista todos os arquivos *com a extensão .COM*

A>DEL PROGRUIM.TXT
Apaga o arquivo especificado

A>ERASE PROGRUIM.TXT
O mesmo que DEL

A>PAUSE *Are you done?*
Exibe a mensagem dada, seguida por “Strike any key when ready . . .”

A>REM COMENTÁRIO

Exibe um comentário durante a execução de um arquivo de lote

A>RENAME B:PG1.BAS PG2.BAS

Muda o nome do arquivo de PG1.BAS para PG2.BAS

A>TIME 14:35:25.78

Acerta a hora do relógio interno;
hh:mm:ss.xx

A>TYPE B:PG1.TXT

Exibe o conteúdo de um arquivo texto na tela

Utilitários do disquete do DOS; assume-se que o disquete está no acionador A:; COM não é necessário

A>CHKDSK B:

Informa o espaço de disquete e memória disponíveis

A>COMP PG1.BAS B:PG2.BAS

Compara o conteúdo de dois arquivos, *exibe* os erros

A>DISKCOMP A:B:/1

Compara dois disquetes; *face-simples*

A>DISKCOPY B:A:/1

Copia todo o conteúdo do disquete do acionador B: *para A:; face-simples*

A>EXE2BIN W.EXE W1.BIN

Converte W.EXE em W1.BIN em formato de comando

A>FORMAT B:/S/1

**Formata o disquete; copia o DOS;
*face-simples***

A>MODE LPT3:,132,8

**Ajusta a impressora #3 para 132 colunas
(default 80), oito linhas por polegada
(default 6)**

A>MODE 40,R,T

**Ajusta o vídeo de cores/gráficos para 40/80
colunas; move a tela um caractere para a
direita (R) ou esquerda (L); Testa o padrão**

A>MODE COM2:300,E,7,1

**Ajusta a interface de comunicação #2 para
300 bauds; *paridade par (0), ímpar (E) ou
nenhuma ((N); 7 ou 8 bits de dados; 1 ou
2 bits de parada (2 para 110 bauds)***

A>MODE COM1:12,,,P

Usa impressora interface serial 1200 bauds

A>MODE LPT1:=COM1

**Posiciona a saída da impressora paralela
*para COMM #1***

A>SYS B:

**Transfere os arquivos do sistema DOS para
o disquete no acionador B:**

Processamento em Lotes

Para executar:

A>CONTAS Executa o arquivo de lote de nome
CONTAS.BAT

ou liga (ou reinicializa) o computador:

Executa o arquivo AUTOEXEC.BAT
automaticamente se ele estiver no disquete

Para criar:

Use o comando EDLIN

Dê o nome do arquivo da seguinte maneira:
nome.BAT

ou COPY.COM: nome.BAT

Ex., COPY.COM:AUTOEXEC.BAT

PAUSE Pronto para começar?

BASICA SAMPLES

(salva estes comandos no arquivo
AUTOEXECCT.BAT).

Este programa pára, pergunta “Pronto para
começar?”; uma vez que uma tecla seja
pressionada, o programa executará
SAMPLE.BAS

Para passar parâmetros:

- Parâmetros de simulação são %0-%9
- %0 é sempre substituído pela especificação do acionador e nome do arquivo, do arquivo de lote
- ex., se NA.BAT é COPY %1.BAS %2.BAS
então A>NA NOME1 B:NOME2 resultará em
COPY NOME1.BAS B:NOME2.BAS

Nomes de Arquivos

Arquivos em disquete:

1-8 caracteres com uma extensão opcional de 1-3 caracteres. Caracteres válidos **A-Z,O-9**, e

() | <> ! @ # \$ % ^ & _ - ~ \ { }

- ex.
1. DGT.BAS
 2. B:12345.TXT
 3. \$\$SALÁRIO

Extensões especiais:

1. .COM
2. .EXE
3. .BAT
4. .BAS

A>NA primeiro procurará por NA.COM,
então por NA.EXE e, a seguir, por
NA.BAT

A>BASICA DGT

procurará por DGT.BAS

Nomes de periféricos reservados:

1. **COM1:/AUX:** interface de comunicação #1
2. **LPT1:/PRN:** impressora #1
3. **CON:** teclado/vídeo
4. **NUL:** periférico simulado para teste

Caracteres-chave de nomes de arquivo:

- ? Substitui qualquer caractere
ex., FILE? = FILE,FILE\$
- * Substitui qualquer série de caracteres
ex., FILE* = FILE1, FILE\$, FILE1234
F.* = F.BAS,F,F.\$ \$\$ \$

EDLIN

Este é um editor de linhas para criar ou editar arquivos de programa ou texto. Cada linha, até 253 caracteres, é referida por um número. Deve haver sempre espaço no disquete para uma cópia extra do arquivo a ser editado (EDLIN cria uma cópia do arquivo), caso contrário a edição será perdida. Exemplos:

A>EDLIN B:ARQ.TXT

**Edita o arquivo ARQ.TXT no acionador B:
com o disquete do DOS no acionador A:**

- 53A** Acrescenta 53 linhas do disquete para o final do arquivo na memória; somente usado se o arquivo for muito grande para caber na memória
- 20 D** Apaga a linha 20
- 20,30 D** Apaga da linha 20 (default=linha atual) até a linha 30; renumera as restantes
- 103** Edita a linha 103; aborta a edição desta linha com **Esc** ou **Ctrl Break** (ou **↵** se o cursor estiver no início da linha)
- ↵** Edita a linha atual (prompt *)
- E** Termina a edição, salva o arquivo original como XXX.BAK, e o arquivo editado com o nome do arquivo original
- 33I** Insere linhas de texto antes da linha 33 (default=linha atual); usado também para criar texto com novos arquivos; **Ctrl Break** termina o modo de Inserção
- L** Lista 23 linhas: linha atual, 11 linhas anteriores e 11 linhas seguintes
- 47, L** Lista 23 linhas, começando com a linha 47
- 10,25 L** Lista da linha 10 (default=11 linhas antes da linha atual) até a linha 25
- Q** Termina a edição; não salva as mudanças; não cria o arquivo .BAK

Rdos **[F6]** DOS

Substitui todas as ocorrências de “dos” por “DOS”; se “DOS” for omitido, então todas as ocorrências de “dos” serão apagadas

7,45 ? Rdos **[F6]** DOS

Substitui todas as ocorrências das linhas 7 a 45; *retorna um prompt com O.K. ? após cada display, Y ou **[Y]** confirmam cada mudança*

7,45 ? Sdos

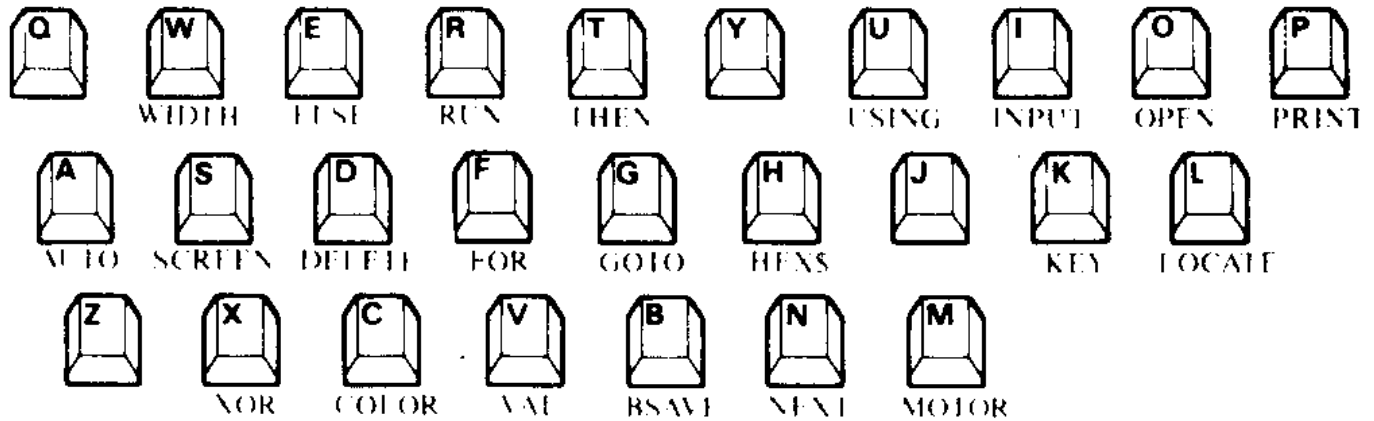
Procura pela próxima ocorrência de “dos” nas linhas 7 a 45; *opção para continuar a pesquisa, responda Y ou **[Y]** para interromper a operação*

540 W Grava as linhas 1 até 540 no disquete; usado se o arquivo for muito grande para a memória; *veja o comando Append*

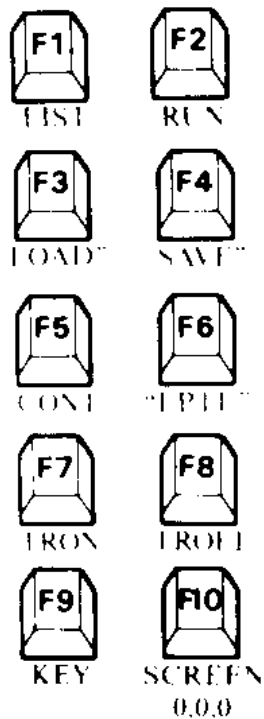
BASIC Teclado

- **[Ctrl]** **[Alt]** **[Del]** = Reinicia o Sistema
- **[Esc]** (Escape) apaga as linhas da tela
- **[Ctrl]** **[Num Lock]** faz uma pausa no programa
- **[Ctrl]** **[Break]** aborta o programa, retorna o prompt do BASIC

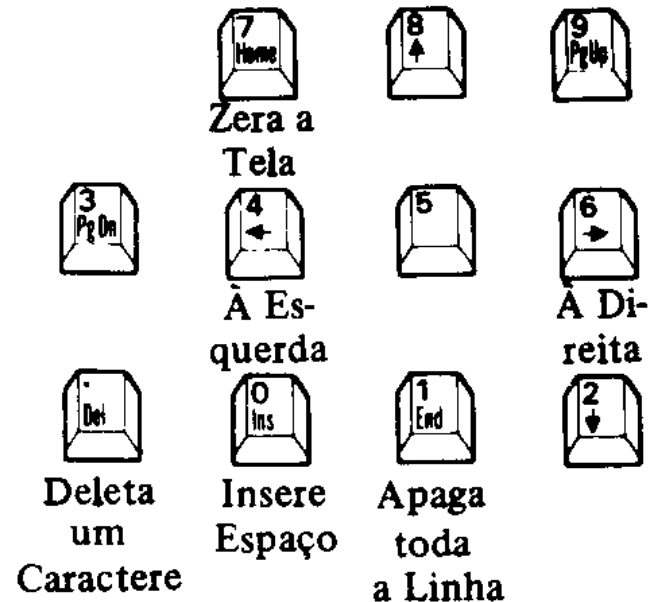
Aciona as palavras-chave pressionando **Alt** + a tecla da letra correspondente:



Teclas de função BASIC pré-definidas:



Aciona os caracteres de função com o teclado numérico:



Teclado numérico com a tecla



Limpa a tela



Move o cursor uma palavra para a esquerda



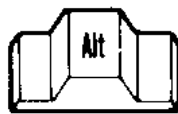
Move o cursor uma palavra para a direita



Apaga até o final da linha

Teclado numérico com a tecla

Aciona os caracteres ASCII correspondentes a
O-255 + 
ex.,



+



+



**Caractere "E"
(ASCII 69)**

Comandos de Arquivos de Programa

Nome de arquivos de fita cassete:

Não mais que oito caracteres; sem vírgulas, hex 00,
ou hex FF

Nome de arquivos de disquete:

O mesmo que o nome de arquivos DOS (*veja p. 10*).

1. BAS é assumido quando utiliza-se BASIC e BASIC Avançado
**A>BASIC /F:arquivos/S:rand.buff/C:comm.buffer/
M:basic.tamanho**

Carrega o BASIC do disquete do DOS com as opções apresentadas abaixo; use A>BASICA/, etc., para utilizar as características do BASIC Avançado

Opção	Descrição	Default	Max.	Comentários
/F:	Número máximo de arquivos abertos	3	15	188 bytes/arquivo mais buffer
/S:	Tamanho do buffer do arquivo randômico	128	32767	512 é ótimo
/C:	Tamanho do buffer de recepção	256	32767	1024 para linhas de alta velocidade
/M:	Tamanho da área de trabalho do BASIC	Toda (<=64K)	64 K	Deixa espaço para a linguagem de máquina

BLOAD "arquivo1", 100

Carrega um arquivo binário com offset 100
(default onde tiver sido aplicado BSAVE)

BSAVE "arquivo", 5, &H4000

Salve o arquivo binário de comprimento hex 4000 com offset=5

1 CHAIN "prog2", 300

Carrega e executa PROG2.BAS, começando a execução na linha 300, passando variáveis comuns (COMMON)

1 CHAIN "PROG2" , , ALL

Carrega e executa PROG2.BAS começando a execução no início do programa, passando todas as variáveis (todos os DEF . . . devem ser recomeçados)

1 CHAIN MERGE "PROG2" , 2000 , DELETE 3000 , 3999

Junta PROG2 com o programa na memória, deixando os arquivos abertos e mantendo os valores variáveis, *apaga as linhas 3000 -3999 do programa na memória; começa a execução na linha 2000*

1 FILES "*.bas"

Lista todos os arquivos com *a extensão .BAS*; como o comando DIR no DOS

1 KILL "jogo1.bas"

Apaga o arquivo especificado (inclui a extensão); como o comando ERASE no DOS

LOAD "b:jogo1" , R

Carrega o programa na memória, e *executa-o* (RUN), (.BAS não é necessário)

MERGE "JOGO2"

Adiciona linhas do programa JOGO2. BAS (em ASCII) ao programa contido na memória

1 **NAME** "jogo2" AS "jogo3"

Dá um novo nome ao arquivo em disquete;
como o comando **RENAME** no DOS

RUN "jogo3", R

Carrega e executa o programa **JOGO3**.
BAS do disquete default; *abre os
arquivos de dados*

SAVE "jogo3"

Salva o programa especificado na forma
compacta

SAVE "jogo3", A

Salva o programa especificado no formato
ASCII

SAVE "jogo3", |P

Salva o programa especificado no formato
protegido; não é permitida listagem ou
edição.

Desenvolvimento de Programas

Insira a linha atual para **AUTO**, **DELETE**, **EDIT**,
LÍST, **LLIST** usando "." (ex., **LIST**.-50).

AUTO 70,5

Numeração automática de linha começando
com a linha número 70 (default 10), e
incrementando de 5 (default 10)

CLEAR, TOP.OF.MEMORY, STACK.SPACE

Limpa todas as variáveis; deixa espaço para o programa em linguagem de máquina, e espaço extra para a pilha se necessário

CONT Continua o programa após uma instrução STOP,END, Ctrl Breek ou um erro

DELETE 75-110

Apaga as linhas 75-110, inclusive

EDIT 110

Mostra a linha especificada para edição

PRINT FRE (O)

Retorna o número de bytes ainda disponíveis para o BASIC

A=FRE (" ")

Similar ao acima, mas o argumento da série de caracteres provoca uma "arrumação"

LIST 140-190

Lista na tela as linhas do programa 140-190 (default=primeira linha-última linha)

LIST 140-190, LPT2:

Lista as linhas 140-190 (default=tudo o programa) no periférico ou arquivo especificado

LLIST 140-310

Lista as linhas 140-310 (default=tudo o programa) na impressora #1

NEW Apaga o programa que está na memória, fecha os arquivos, limpa as variáveis

RENUM 500,100,20

Muda o número da linha de 100 (default= primeira linha do programa) para 500 (default=linha 10) e renumera as linhas seguintes em incrementos de 20 (default=10)

RUN 245 Inicia a execução do programa na linha 245 (default=início do programa)

STOP Termina a execução do programa

1 **SYSTEM**

Retorna para o DOS, fecha todos os arquivos

TRON Ativa o modo Trace; mostra o número das linhas que estão sendo executadas (ex., [10] [20] [55])

TROFF Desativa o modo Trace

Palavras Reservadas

Todas as palavras-chave do ÍNDICE BASIC exceto COM. Não as use para nomes de variáveis, strings (séries de caracteres) ou numéricas, a menos que estejam embutidas.

Nomes de variáveis: até 40 caracteres significativos A-Z, O-9, ponto decimal

Comando de inicialização

COMMON B\$,C ()

Passa B\$, matriz C para programas encadeados ou , use CHAIN "PROG",,ALL

1 DATE\$="02-24-1984"

Define a data como *mm-dd-aaaa* ou *mm/dd/aaaa* (1980-2099 somente)

DEF FNXXXX

Função numérica ou de série definida pelo usuário; veja MATH ou STRINGS

DEFINT I-K

Todas as variáveis começando com I,J,K serão inteiras

DEFSNG Q,S-U,X

Todas as variáveis começando com Q,S,T,U,X serão de precisão simples

DEFDBL A,D

Todas as variáveis começando com A ou D são de precisão dupla

DEFSTR B B,BF,B4, etc., são séries

DEFUSR 9=32768

Offset de início da sub-rotina em linguagem de máquina **USR9** (*default=USRO*)

DIM A(15),B\$(12,8),C%(103)

Necessário se o máximo número de elementos da matriz for maior que 10

ERASE SÉRIE1,SÉRIE2

Elimina as matrizes para permitir
reDIMensionamento

OPTION BASE 1

Define o índice da matriz em 1 para o
primeiro elemento

RANDOMIZE –1931

Reativa o gerador de números aleatórios
(*default=número inicial solicitado*)

1 TIMES\$="15:35:15"

Define o relógio de 24 horas em hh:mm:ss;
pode ser definido no DOS; funciona como
uma variável (ex., A\$ = TIME\$)

Constantes e Variáveis

Tipo	Dígitos	Bytes	Valor	Exemplo	Nome da Variável
Inteiro	5	2	-32768 a 32767	1099%	MÃE.PAI%
Precisão Simples	7+2exp	4	overflow= +1.7E38	1.234567E-12	V2 ou SSS!
Precisão Dupla	16+2exp	8	underflow =2.95E-39	1.23456789D32	A12#
Hexadecimal	4	2	0000 a FFFF	&H92EF	
Octal	6	2	000000 a 177777	&O307 ou	
String		1-255		"Hora=1:21"	HORA\$

Funções Matemáticas

Ângulos em radianos

ABS(X)	Retorna com o valor absoluto de x
ATN(X)	Arctangente de x
COS(X)	Co-seno de x
EXP(X)	e elevado a x; $x < 89$
FIX(X)	Trunca o valor de x (ex., $\text{FIX}(-5.7) = -5$, $\text{FIX}(5.7) = 5$)
INT(X)	Maior inteiro menor que ou igual a x (ex., $\text{INT}(-5.7) = -6$, $\text{INT}(5.7) = 5$)
CINT(X)	<i>Arredonda</i> para o valor absoluto mais próximo do inteiro (ex., $\text{CINT}(-5.7) = -6$; $\text{CINT}(5.7) = 6$)
LOG(X)	Log natural (base e) de x
RND(X)	Número aleatório (0 a 1); $x < 0$ para início; $x = 0$ para repetir; $x > 0$ para o próximo
SGN(X)	Sinal de x ($-1 = \text{negativo}$; $0 = \text{zero}$; $1 = \text{positivo}$)
SIN	Seno de x
SQR(X)	Raiz quadrada de x
TAN(X)	Tangente de x
SWAP H, V(5)	Troca os valores de duas variáveis (as variáveis devem ser do mesmo tipo)

DEF FN ABC (X,Y,Z)=X*Y+Z

**Função definida pelo usuário; uma
linha; pode ser recursiva**

Funções trigonométricas derivadas

DEF FNSEC(X)=1/COS(X)	Secante
DEF FNCSC(X)=1/SIN(X)	Co-secante
DEF FNCOT(X)=1/TAN(X)	Co-tangente
DEF FNARCSIN(X)=ATN(X/SQR(1-X*X))	Arco seno
DEF FNARCCOS(X)=1.570796-ATN(X/SQR(1-X*X))	Arco co-seno
DEF FNARCSEC(X)=ATN(SQR(X*X-1))+(X<0)*3.141593	Arco secante
DEF FNARCCSC(X)=ATN(1/SQR(X*X-1))+(X<0)*3.141593	Arco co-secante
DEF FNARCCOT(X)=1.57096-ATN(X)	Arco co-tangente
DEF FNSINH(X)=(EXP(X)-EXP(-X))/2	Seno hiperbólico
DEF FNCOSH(X)=(EXP(X)+EXP(-X))/2	Co-seno hiperbólico
DEF FNTANH(X)=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))	Tangente hiperbólica
DEF FNARCSINH(X)=LOG(X+SQR(X*X+1))	Arco seno hiperbólico
DEF FNARCCOSH(X)=LOG(X+SQR(X*X-1))	Arco co-seno hiperbólico
DEF FNARCTANH(X)=LOG((1+X)/(1-X))/2	Arco tangente hiperbólico
DEF FNARCSECH(X)=LOG((1+SQR(1-X*X))/X)	Arco secante hiperbólico
DEF FNARCCSH(X)=LOG((1+SGN(X)*SQR(1+X*X))/X)	Arco co-secante hiperbólico
DEF FNARCCOTH(X)=LOG((X+1)/(X-1))/2	Arco co-tangente hiperbólico

Para converter graus em radianos, use as fórmulas:

Graus = $x*180/3.141593$ (x em radianos)

Radianos = $x*3.141593/180$ (x em radianos)

Programação Geral

DATA 1.3,&H7A,&177,JUNHO''JULHO 23, 1981''	Fornece a data para a instrução READ
END	Fecha todos os arquivos; termina o programa
FOR J=J1 TO J2 STEP J3	Salta-se o laço se $J2 > J1$; <i>default STEP=1</i>
NEXT J	Final do laço; mais rápido com variáveis inteiras e não com nome de variáveis
GOSUB 1120	Desvia para a sub-rotina na linha 1120
GOTO 75	Desvia direto para a linha 75; continua o programa a partir daí
IF A=5, THEN A=A+1 ELSE A=0	Se a expressão após o IF for verdadeira, executa a instrução entre THEN e ELSE; <i>se não, executa a instrução após ELSE</i> (default=próxima linha do programa)
IF A=5 GOTO 270 ELSE 290	Se $A=5$, vai para linha 270; se $A \neq 5$ vai para a linha 290 (default=próxima linha do programa)
LET A=1.73	Define o valor da variável; <i>LET é opcional</i>

ON M GOTO 170,180,190 Semelhante a ON GOSUB
ON N GOSUB 25,55,73,104

Se N=1, vai para a linha 25; se N=2,
vai para a linha 55, etc., pula se
N<1 ou o número da linha não tiver
correspondência

READ A,BHEX,C.OCTAL,D\$,E\$

Lê os dados das instruções **DATA** e os
coloca em A, BHEX, etc.

REM QUALQUER COMENTÁRIO!\$?

Insera qualquer caractere como
comentário; **REM** pode ser
substituído por'

RESTORE 1500

A linha 1500 é a próxima instrução
DATA lida (default=primeiro
DATA)

RETURN Termina uma sub-rotina; retorna para
a instrução seguinte a **GOSUB**

② RETURN 130

Termina uma sub-rotina (usualmente
um evento de erro); retorna para a
linha especificada

WHILE A>7 Executa as instruções seguintes se a
expressão for verdadeira; deve ter
uma instrução **WEND** correspondente

WEND Volta para a instrução **WHILE**; se a

expressão WHILE for falsa, executará a próxima instrução após WEND

Conversão de Dados

I%=inteiro; S!=precisão simples; D#=precisão dupla

S!=2.54:PRINT CDBL(S!)

**Converte para precisão dupla
(2.539999961853027)**

D#=1.23456789012345:PRINT CSNG (D#)

Converte para precisão simples (1.234568)

I%=2.5:PRINT I%

Retorna 3; arredondado

S!=2.5:PRINT CINT

**Converte para inteiro; retorna 3;
arredondado**

HEX\$(256*256-1)

Retorna "FFFF" como um string hexadecimal

OCT\$(24)

Conversão octal; retorna o string "30"

STR\$(24)

Conversão de string; retorna o string "24"

VAL(" - 024")

Retorna -24 como um número decimal

CBD/I/S e MKD\$/I/S

*Veja Seção de Arquivos aleatórios de disco
na p. 58.*

Operadores

Operadores Aritméticos			Operadores Relacionais		
Operador	Função	Exemplo	Operador	Função	Exemplo
^	Exponenciação	2^3=8	=	igual	3=5 <i>falso</i> (0)
*	Multiplicação	7*4=28	<>	diferente	3<>5 <i>verd.</i> (-1)
/	Divisão	9/6=1.5	<	menor que	3<5 <i>verd.</i>
\	Divisão Inteira	9\6=1	>	maior que	3>5 <i>falso</i>
MOD	Módulo	9MOD 6=3	<=	menor ou igual	3<=5 <i>verd.</i>
+	Adição	9+3=12	>=	maior ou igual	3>=5 <i>falso</i>
-	Subtração	9-3=6			

Operadores Lógicos							
X	Y	NOT X	AND	OR	XOR	IMP (implica)	EQV
T	T	F	T	T	F	T	T
T	F	F	F	T	T	F	F
F	T	T	F	T	T	T	F
F	F	T	F	F	F	T	T
2	7	-3	2	7	5	-1	-6 comparação bit por bit

Funções de String*

Combina as séries com “+” (ex., “ABC”+“DEF”= “ABCDEF”)

ASC(“ABC”)

Retorna 65 como o código ASCII para o primeiro caractere na série

CHR\$(66)

Retorna “B”, converte o código ASCII em caractere

INSTR(3,“ABCABC”, “BC”)

Retorna 5 como a primeira ocorrência de

* Série de caracteres.

“BC” em “ABCABC”, começando na
terceira posição (default=início da série)

LEFT\$(“abcd”,2)

Retorna “ab” como os dois caracteres mais
à esquerda da série

LEN(“AB CD”)

Retorna 5 como o número de caracteres
na série

MID\$(“ABCD”,3,1)

Retorna “C” como um caractere iniciado na
posição 3

RIGHT\$(“ABCDE”,2)

Retorna “DE” como os últimos dois
caracteres no lado direito

SPACE\$(5)

Retorna uma série com cinco espaços

SPC(5) Imprime cinco **espaços**; usado com PRINT,
LPRINT e PRINT #

STRING\$(2,42)

Retorna “**” como uma série de dois
caracteres de código ASCII 42

STRING\$(3,“ABC”)

Retorna “AAA” como o primeiro caractere
repetido três vezes

SWAP A\$,F\$

Troca o conteúdo das duas séries (não
precisam ser do mesmo comprimento)

Modos de Texto e de Gráficos

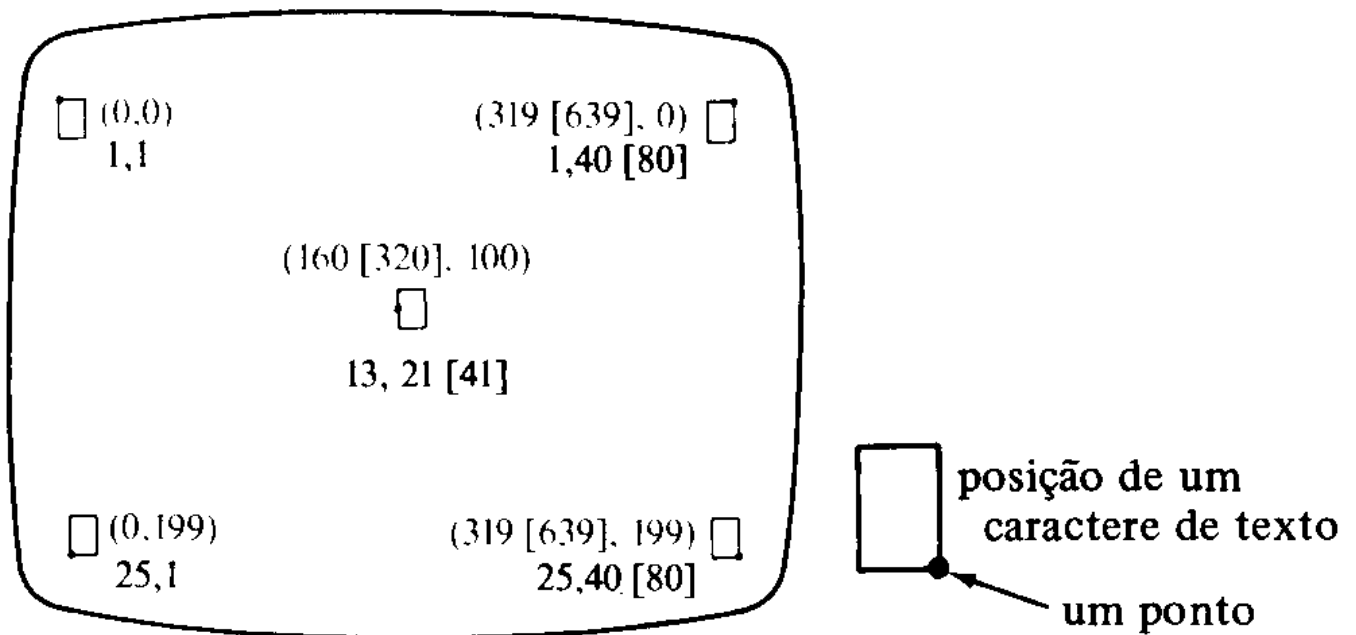
Mostra o dado (chadado por)	Adaptador de Monocromático	Adaptado de Gráficos/Cores		
	(SCREEN 0)	(SCREEN 0)	(SCREEN 1)	(SCREEN 2)
Texto				
40-Colunas	✓	✓	✓	n/d
80-Colunas	✓	✓	n/d	✓
Cores	Preto e Branco	16	6	Branco
Intensificado	✓	✓	n/d	n/d
Piscante	✓	✓	n/d	n/d
Sublinhado	✓	n/d	n/d	n/d
Gráficos				
Caracteres			Definido pelo usuário	Definido pelo usuário
Gráficos	✓	✓		
Res. Média (3 cores)	n/d	n/d	320x200 pontos	n/d
Alta res. (Preto,branco)	n/d	n/d	n/d	640x200 pontos
Cores disponíveis (habilitadas por)	n/d	(SCREEN 0,1)	(SCREEN 1,0)	n/d
Cores do "background"	Preto e branco	8+	16	Preto
Cores da borda	(="background")	16	(="background")	Preto
+ 16 cores do "background" em 40(80) colunas com OUT & H3D8,8(9) ✓ = Disponível n/d = Não disponível				

OBS.: Background - Segundo Plano
 Foreground - Primeiro Plano

Sumário dos Comandos

Modo	Coordenadas formato e variação	Define a posição do cursor ou ponto	Determina a posição do cursor	Determina o caractere ou a cor	Para exibir dados na tela
Texto	linha, col linha 1-25 col 1-40 ou 1-80	LOCATE	linhas: CSRLIN col: POS(0)	SCREEN	PRINT
Gráficos	(col, linha) col 0-319 ou 0-639 linha 0-199	PSET ou PRESET		POINT	ponteiros: PSET, PRESET linhas e quadrados: LINE círculos: CIRCLE áreas preenchidas: PAINT formas: DRAW imagem de uma matriz: PUT

Coordenadas para Pontos-Gráficos e Colocação de Texto



Com o Adaptador de Gráficos/Cores, **gráficos** (indicados em cor aqui, criados com PSET, LINE, DRAWm etc.) podem ser misturados com texto (indicado em tipo preto aqui, criado com LOCATE e PRINT).

Note a total diferença em coordenadas de sistema.

Monitor no Modo Texto

Adaptador Monocromático e de Gráficos/Cores

CLS **Limpa a página ativa; coloca o cursor no canto superior esquerdo**

WIDTH 40

Define o tamanho da tela com 40 colunas

WIDTH 80

Define o tamanho da tela com 80 colunas

LOCATE LINHA, COLUNA, ON.OFF, START, STOP

Coloca o cursor na linha e coluna

determinada; ON.OFF=0 para o cursor desativado, 1 para o cursor ativado; START, STOP variação de 0-31 para o tamanho do cursor

LOCATE ,,1,7,7

Define o cursor com a forma normal (i.é., sublinhado)

COLUMN=POS (0)

Retorna a coluna em que o cursor está (1-40 ou 1-80)

ROW=CSRLIN

Retorna a linha em que o cursor está (1-25)

A%=SCREEN (LINHA, COLUNA, 0)

Retorna o código ASCII nas coordenadas (linha, coluna)

C%=SCREEN (LINHA, COLUNA, 1)

Retorna os atributos de cor do caractere; deriva o número da cor dos seguintes atributos:

F%=C% MOD 16 F%= cor do "foreground"
(C%>127 se piscante)

$B\% = (C\% \text{ MOD } 128) \setminus 16$

$B\%$ = cor do "background"

SCREEN 0, BURST, ACTP=AGE, VISPAGE

Define o modo de texto; *itens opcionais para a placa de gráficos/cores somente*: em cor se $BURST > 0$; $ACTPAGE$ = página para a saída do texto

$VISPAGE$ = página visível (0-7 para $WIDTH 40$, 0-3 para $WIDTH 80$)

COR FRENTE, FUNDO, MARGEM

Define "foreground" (texto), "background", margem de cores (default=valor anterior); veja tabela abaixo

#	Cor	Monocromático		#	Cor	Monocromático	
		Fundo	Frente			Fundo	Frente
0	Preto	B	B	8	Cinza	HiW	B
1	Azul	W/ und	B	9	Azul-claro	HiW/ und	B
2	Verde	W	B	10	Verde-claro	HiW	B
3	Azul-escuro	W	B	11	Azul-claro	HiW	B
4	Vermelho	W	B	12	Verm.-claro	HiW	B
5	Magenta	W	B	13	Mag.-claro	HiW	B
6	Marrom	W	B	14	Amarelo	HiW	B
7	Branco	W	W	15	Branco	HiW	B

(Frente # + 16) pisca o "foreground" para colorido ou monocromático
Cores 8-15 disponíveis somente se uma instrução OUT for executada (veja p. 30)

PRINT A B,C\$;D;

Imprime a tela da página ativa; itens

separados pelo espaço ou ponto e vírgula são impressos próximos uns dos outros; a

vírgula provoca a impressão no início de cada zona de impressão (a cada 14 espaços); *se houver uma vírgula ou ponto e vírgula no final de uma instrução, não haverá um retorno de carro*

? A B,C\$;D;

O mesmo que a instrução **PRINT**? será LISTada como PRINT

PRINT USING "SS###.##";A,B

Imprime as variáveis no formato especificado pela série; *todas as variáveis serão impressas no mesmo formato*

Formato do Caractere	Definição	Exemplo		
		PRINT USING		Resultado
		Formato	Variável	
!	Primeiro caractere da série	"!"	"color"	c
\n espaços\	2+n séries de caracteres	"\ \"	"color"	col
&	Série inteira	"&"	"color"	color
#	Um dígito de cada #; arredondada	"###"	678.7	679
.	Ponto decimal	"##.##"	34	34.00
,	Vírgulas a cada três posições	"#####,"	123456	123,456
+ no início	+ ou - antes do número	"+###"	34,-90	+34,-90
+ no final	Sinal + após; se positivo	"##+"	78.3	78.3+
- no final	Sinal - após; se negativo	"##-##-"	+34,-82	34 82-
**	Preenche espaços deixados com *'s	"*#####"	1234,66	*1234***66
\$	Cifrão alinhado à esquerda	"\$##"	12,6	\$12\$ 6
\$\$	Cifrão alinhado à direita	"\$\$"	12,6	\$12\$ 6
\$	Preenche com **, então imprime \$	"\$"	12,6	\$12*\$6
^ ^ ^ ^	Formato exponencial	"##.# ^ ^ ^ ^"	752	7E+02
-(sublinha)	Próximo caractere como literal	"##-%"	98	98%

PRINT TAB(5);

Tabula na quinta coluna; na próxima linha,
se na atual já passou a linha 5

WRITE 5.4, "hi"

Exibe 5.4, "hi" com vírgulas e aspas

Gráficos/Cores

Com o Adaptador de Gráficos a Cores

SCREEN 1,16

Resolução média; *desativa a cor* (default=
cor acionada); sem vídeo inverso ou
piscante no modo de texto com SCREEN
1 ou 2

SCREEN 2 Alta resolução (sem cor)

COLOR BACKGROUND, PALETTE

Define média resolução; *define a cor do*
"background" (default=atual) e *seleciona*
o palette (default=0)

DEF SEG:POKE &H4E, COLR

Define a cor do texto para 1, 2, ou 3 em
média resolução

Cores do background			
0 preto	4 vermelho	8 cinza	12 1. vermelho
1 azul	5 magenta	9 1. azul	13 magenta
2 verde	6 marrom	10 1. verde	14 amarelo
3 azul	7 branco	11 1. azul-esverdeado	15 branco luminoso
Cor #	Palette=par #	Palette=ímpar #	
0	Background	Background	
1	Verde	Azul-esverdeado (azul-verde)	
2	Vermelho	Magenta (vermelho-azul)	
3 ("foreground")	Marrom	Branco	

Pontos

PSET (X,Y),COLR

Ponto em (x,y) com COLR(0-3)
(defaults="foreground")

STEP Precede qualquer (x,y) para movimentos relativos

PSET STEP (X,Y),COLR 2

Movimento relativo; coloca unidades dos pontos x e y distantes do último ponto referenciado na cor *especificada por COLR2* (default=3)

PRESET (X,Y),COLR

O mesmo que **PSET**, mas o default é o "background" com cor (0)

COLR=POINT(X,Y)

Retorna a cor do ponto em (x,y); = -1 se (x,y) estiver fora da tela

Linhas e Blocos

LINE –(X2,Y2),C

Desenha a linha (x2,y2) na cor C
(default="foreground")

LINE(0,0) –(32,18)

Desenha a linha de (0,0) até (32,18) na cor do "foreground"

LINE(10,10) –(160,180),2,BF

Desenha um bloco na cor 2 (default=cor do "foreground") com o canto superior esquerdo em (10,10) e o canto inferior direito em (160,180); *F preenche o bloco*

Círculos

② CIRCLE(X,Y),14

Desenha o círculo com raio 14 centrado em (x,y)

② CIRCLE(X,Y),14,2,0,1.5

Desenha o arco na cor 2, de 0 a 1.5 radianos; (default de cor="foreground"); valores de arcos negativos desenharão raios

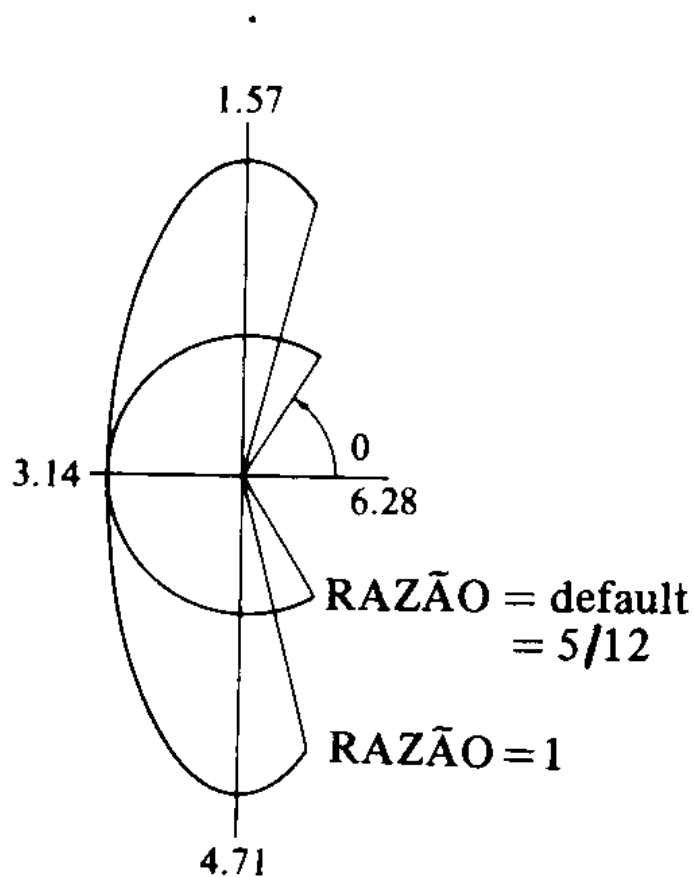
2 CIRCLE (X, Y), 14, ..., 1/4

Desenha uma elipse com raio de 1/4;
(default da proporção das faces: 5/6
médio, 5/12 de alta resolução; se
proporção < 1 – raio no eixo x; se
proporção > 1 – eixo y)

SCREEN 2: CIRCLE (480, 100), 80, 1-1, -5.28 RATIO

Em média resolução
(SCREEN 1):
10 unidades na vertical
têm o mesmo comprimento
que 12 unidades na
horizontal.

Em alta resolução
(SCREEN 2):
10 unidades na vertical
têm o mesmo comprimento
que 24 unidades na
horizontal.



Áreas

2 PAINT (X, Y), COLOR1, EDGECOLOR

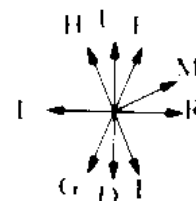
Preenche a área com COLOR1 (default=cor do “foreground”), alinhado com

EDGECOLOR0-3 (default=COLOR1);
 (x,y) devem estar dentro da área a ser
 pintada (PAINT)

2 DRAW P\$

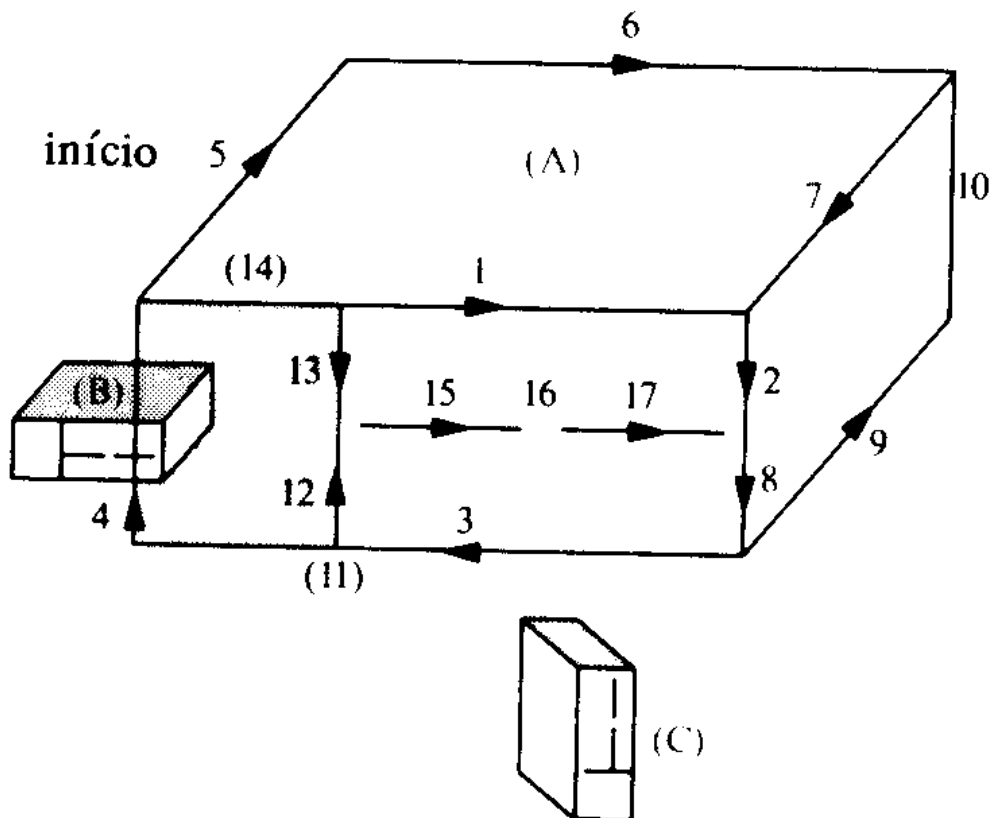
Desenha um objeto dado através de comandos
 de linguagem gráfica na série P\$

Comando	Resultado
"U10"	Desenha 10 unidades para cima a partir do último ponto referenciado
"E5"	Desenha para cima e para a direita (5 unidades para cima, 5 unidades para a direita)
"U=NA;"	Desenha para cima o número de unidades fornecidas pelo valor de NA
R,F,D,G,L,H	Outros comandos de desenho mostrados no diagrama à direita
"M10,15"	Desenha uma linha oblíqua a partir do último ponto de referência para o endereço absoluto (10,15)
"M+10,-15"	Desenha uma linha relativa para o último ponto referenciado através de 10 unidades horizontais e 15 unidades verticais
"BU35"	Move o último ponto referenciado 35 unidades para cima, mas não desenha
"NU35"	Desenha para cima 35 unidades, mas não altera o último ponto de referência
"C2"	Desenha os pontos seguintes na cor #2
"Sn"	Usa o fator de escala= $n/4$; n é a variação de 1-255 (default=1)
"An"	Gira para a esquerda com um ângulo de $90*n$ graus; $n=0/1/2/3$
"Xstring\$;"	Desenha o objeto definido pelo string (série de caracteres) dado; pode ser encadeado
"x"+VARPTR\$(string\$)	Para uso do compilador



Programa Exemplo

```
5 SCREEN 1:CLS:COLOR 0,0
10 F$="R150 D50 L150 U50 E50 R150 G50 D50
    E50 U50 BM-150,100 U50 D25 BR5 R40 BR10 R40"
20 DRAW"C2 BM50,75 XF$;":PAINT
    STEP(-100,0),2 'desenha PC; pinta a parte frontal
    esquerda (A)
30 DRAW"C1 BM20,100 S1 XF$;":PAINT STEP
    (0,-15),1 'desenha em menor escala; pinta o
    topo (B)
40 DRAW"C3 BM160,180 A1 XF$;":PAINT STEP
    (-5,-5),3 'gira; desenha; pinta a lateral (C)
```



Para salvar uma tela gráfica no disquete, use

```
DEF SEG = &HB800:BSAVE "FIGURA",0,&H4000
```

Para recuperar uma tela gráfica do disquete, use

```
DEF SEG = &HB800:BL0AD "FIGURA"
```

Get e Put

Permitem que blocos de imagem sejam transferidos para e de séries

O dimensionamento de uma série para armazenar uma imagem é feito como mostrado abaixo:

$$H=(X2-X1)+1:V=(Y2-Y1)+1$$

Dimensões horizontal e vertical

$$BYTES=4+V*INT((BITS*H+7)/8)$$

BITS=2 para média resolução; 1 para alta resolução

```
DIM ARRAY%(BYTES/2)
```

Série para armazenar a tela de uma área retangular; superdimensionamento é aceito

Para área quadrada, 'n' unidades de cada lado:

n	Dimensões da Matriz	
	Média Res.	Alta Res.
10	19	13
20	65	34
50	334	181
Tela completa	8002	8002

2 GET (X1,Y1)–(X2,Y2),ARRAY%

Salva a área da tela gráfica na matriz; para uma velocidade mais alta use $X1 \text{ MOD } 4 = 0$ em baixa resolução, $X1 \text{ MOD } 8 = 0$ em alta resolução

2 PUT (X3,Y3),ARRAY% ACTION

Coloca a imagem de volta na tela; (X3,Y3) = canto superior esquerdo; as possíveis AÇÕES são:

- XOR Inverte o ponto na tela se o ponto existir na imagem da matriz (também ação default)
- OR Superpõe a imagem da matriz sobre uma imagem já existente
- AND Transfere a matriz da imagem somente se a imagem já estiver oculta
- PSET Simplesmente transfere a matriz da imagem para a tela
- PRESET Simplesmente transfere o negativo da matriz da imagem para a tela

Cores resultantes (média resolução (SCREEN 1) e alta resolução (SCREEN 2)):

Número da Cor da Tela	Número da Cor da Matriz				
	AND 0 1 2 3	OR 0 1 2 3	XOR 0 1 2 3	PSET 0 1 2 3	PRESET 0 1 2 3
(Background) 0	0 0 0 0	0 1 2 3	0 1 2 3	0 1 2 3	3 2 1 0
(Foreground) 1	0 1 0 1	1 1 3 3	1 0 3 2	0 1 2 3	3 2 1 0
2	0 0 2 2	2 3 2 3	2 3 0 1	0 1 2 3	3 2 1 0
(Foreground) 3	0 1 2 3	3 3 3 3	3 2 1 0	0 1 2 3	3 2 1 0

Entrada de Teclado

A\$=INKEY\$

Procura por um pressionamento de tecla;

armazena o caractere na variável A\$ (série

(DEF SEG=0:POKE 1050,PEEK(1052))

Limpa o buffer do teclado

INPUT "PROMPT";A,B

Obtém a entrada do teclado para cada

variável na lista; primeiro *ponto e vírgula*

elimina o retorno do carro – alimentação

de linha após a entrada; mostra a

mensagem de entrada em "PROMPT";

vírgula após "PROMPT" elimina a "?"

A\$=INPUT\$(5)

Aguarda até que cinco caracteres sejam digitados; não é necessário

LINE INPUT;"PERGUNTA?";Q\$

Insera qualquer texto incluindo vírgulas e aspas; deve ser entrado em uma variável de série

Teclas de Função

KEY OFF

Apaga os valores das teclas de função da linha 25

KEY ON

Mostra os primeiros seis caracteres com os respectivos valores das teclas de função

KEY LIST

Lista todos os valores das dez teclas de função na tela

KEY 10,"LIST-100"+CHR\$(13)

Designa uma série (1-15 caracteres) para a tecla de função F10

KEY 5," "

Desativa a tecla de função especificada

KEY (n) ON

Ativa a ação de "armadilha" da tecla especificada

n	Tecla
1-10	F1 -F10
11	Cursor para Cima
12	Cursor para a Esquerda
13	Cursor para a Direita
14	Cursor para Baixo

[2] KEY (3) OFF

Desativa a ação de “armadilha” da tecla:
tecla pressionada não é armazenada

[2] KEY (11) STOP

Desativa a ação de “armadilha” da tecla;
armazena a tecla pressionada até a
condição KEY(11) ON

[2] ON KEY (5) GOSUB 5000

Após KEY(5) ON, GOSUB 5000 quando
[F5] é pressionada

Entrada/Saída de Periféricos em Geral

especificação do arquivo=periférico:nome do arquivo

periférico=KYBD:

Teclado

SCRN:

Vídeo

LPT1: *ou* LPT2:

ou LPT3:

Impressora

COM1: *ou* COM2: Placa de
 Comunicação
 CAS1: Gravador Cassete
 A: *ou* B: *ou* C: *ou* D: Acionador de
 Disquete

Comandos de arquivo assim como OPEN e PRINT # podem ser usados para comunicações em um sentido genérico com todos os periféricos mencionados acima.

2 BASIC Avançado permite interrupções ao invés de um “polling” com os periféricos de E/S, assim como porta de comunicação, caneta luminosa, teclas de função especial e adaptador de jogos (ex., KEY(n) ON/OFF/STOP e ON KEY(n) GOSUB).

Comunicações

1 OPEN “COM2:BAUD,PARIDADE,BITS.DE.DADOS,BITS.DE.PARADA” COMO #5

Aloca o buffer para a saída/entrada 1 ou 2 como arquivo 5; opções de variáveis (*defaults em itálico*) são:

Baud	Paridade	Bits.de.Dados	Bits.de.Parada
75 1200	S (espaço)	4	1 (default para 150-9600 baud)
110 1800	0 (ímpar)	5	2 (para 1½; usado para 4 ou 5 bits de dados)
150 2400	M (marca)	6	2 (default para 75-110 baud)
300 4800	E (par)	7	
600 9600	N (nenhuma)		

1 OPEN "COM1:300,,,,,RS,CS100,DS50,CD5000,LF" AS #2 LEN=64

- RS** Elimina a linha de RTS (REQUEST-TO-SEND)
- CS100** Espera 100 mseg pelo CTS (CLEAR-TO-SEND) antes do Timeout (default=CS1000); CS ou CS0 eliminam a linha CTS
- DS50** Semelhante a CS para a linha DSR (DATA-SET-READY) (default=DS1000)
- CD5000** Semelhante a CS para a linha CD (CARRIER-DETECT) (default=CD ignorado)
- LF** Envia uma alimentação de página após cada retorno do carro (para impressoras somente)
- LEN-64** Máximo número de bytes lidos do buffer do COMM (default=128)

Lê o dado da porta Comm:

Use INPUT#, LINE INPUT# e INPUT\$ com arquivos de dados seqüenciais.

2 COM(n) ON

Ativa a ação de "armadilha" do adaptador COMM #1 ou #2

2 COM(n) OFF/STOP

Desativa a ação de “armadilha” (use
COM(n)OFF ou COM(n)STOP)

2 ON COM(n) GOSUB 900

GOSUB se o dado vem para o buffer de
entrada; uma linha de número 0 desativa
a ação de “armadilha”

1 WHILE NOT EOF (5)

EOF verdadeiro se o buffer do arquivo #5
estiver vazio; NOT EOF é verdadeiro se
existir dado no buffer

1 NC=LOC (5)

Número de caracteres no buffer #5

1 NF=LOF (5) Libera espaço no buffer

CHAR\$=INPUT\$(NC, #5)

Lê NC caracteres do buffer

INPUT#5,Q\$

Retorna os caracteres até a primeira vírgula
ou retorno de carro

LINE INPUT#5,P\$

Retorna os caracteres até o primeiro retorno
de carro

XOFF\$=CHR\$(19):XON\$=CHR\$(17)

Freqüentemente utilizado para controlar
periféricos remotos

Envia o dado para a porta Comm:

Use PRINT#, PRINT# USING, WRITE# com arquivos de dados seqüenciais

1 WIDTH "COM2:",128

Envia um retorno de carro/alimentação de linha após *128 caracteres (default=255)*; ativado por subseqüentes instruções OPEN

1 WIDTH #5,80

Troca WIDTH imediatamente se o arquivo #5 estiver aberto

Controle de Impressora

LLIST 149-180

Lista as linhas 149-180 (default=programa inteiro) na impressora

X=LPOS(2)

Posição atual de LPT2:imprime o cabeçalho dentro do buffer da impressora

LPRINT A B,HS;

Imprime o dado na primeira impressora; o ponto e vírgula elimina a alimentação de linha

WIDTH "LPT1:",120

Define a primeira impressora com uma largura de página para impressão de 120 colunas

OPEN "LPT3:" AS #2:WIDTH #2,255

255 elimina a alimentação de linha após o retorno de carro

LPRINT USING FMT\$;A B,C\$

Impressão formatada, *veja* Monitor no Modo Texto p. 31 para formatação

Adaptador para Jogos

AX=STICK(0)

Coordenada x do "joystick" A; 0-100 para 100kOhm

AY=STICK(1):BX=STICK(2):BY=STICK(3)

Recuperado pela função do stick(0)

STRIG ON/OFF

Ativa/desativa o teste dos botões do "joystick"

IF STRIG(n) THEN CLS

Determina se os botões foram pressionados; n=4-7 para o BASIC Avançado somente (*veja* abaixo)

n	Valor (falso)	Valor (verdadeiro)
0 (4)	0: A1 (A2) não pressionado desde o último teste	-1: A1 (A2) foi pressionado
1 (5)	0: A1 (A2) nada pressionado até agora	-1: A1 (A2) sendo pressionado agora
2 (6)	0: B1 (B2) não pressionado desde o último teste	-1: B1 (B2) foi pressionado
3 (7)	0: B1 (B2) nada pressionado até agora	-1: B1 (B2) sendo pressionado agora

2] STRIG (n) ON

Ativa a função de “armadilha” do botão do “joystick”; $n=0,4$ para A1, A2 e $n=2,6$ para B1, B2

2] STRIG (n) OFF

Desativa a função de “armadilha” e não lembra

2] STRIG (n) STOP

Desativa a função de “armadilha”; mas lembre-se: a função ocorrerá com ON

2] ON STRIG (n) GOSUB 1120

Vai para a sub-rotina se um botão é pressionado (linha=0 desativa a função); uma parada automática é executada por GOSUB; um reinício automático (ON) é executado por RETURN

Caneta Luminosa

PEN ON

Ativa a caneta luminosa para uma captação (polling) (e não “armadilha”)

PEN OFF

Desativa a captação (polling) para melhorar a velocidade quando a caneta luminosa não estiver sendo utilizada

Z=PEN (n)

Lê a coordenada n da caneta luminosa:

n	Z
0	-1 se a caneta é pressionada desde a última captação; 0 se não
1	coordenada x(0-319 ou 0-639) onde a caneta estava na última ativação
2	coordenada y (0-199) onde a caneta estava na última ativação
3	-1 se a caneta estiver sendo pressionada agora; 0 se a caneta não estiver sendo pressionada agora
4	coordenada x se a caneta agora estiver para baixo, ou quando não mais pressionada
5	coordenada y se a caneta agora estiver para baixo; ou quando não mais pressionada
6	linha do caractere (1-24) onde a caneta foi ativada pela última vez
7	coluna do caractere (1-40 ou 80) onde a caneta foi ativada pela última vez
8	linha do caractere se a caneta estiver agora na descendente, ou quando da última utilização
9	coluna do caractere se a caneta estiver agora na descendente; ou quando da última utilização

[2] PEN ON/OFF/STOP

Armadilha, semelhante a **[2] STRIG(n) ON;**
etc. (p. 51)

[2] ON PEN GOSUB 2520

Semelhante a ON STRIG(n)

Porta Lógica de Entrada/Saída

BYTE=INP (P)

Lê uma palavra de 8 bits da porta P, com P variando entre 0-65535

OUT P,BYTE

Envia uma palavra de 8 bits (0-255) para a porta P

WAIT P,A%,X%

Espera até que o byte lido da porta P não seja zero *quando realizada uma operação lógica XOR com X%* e a seguir uma operação lógica AND com A%

Motor de Gravador Cassete

MOTOR 0

Desliga o motor; qualquer outro número diferente de zero liga o motor (default= estado da chave)

Som

BEEP Som de 800Hz de tom por 1/4 de segundo; o mesmo que PRINT CHR\$(7)

SOUND 500,36

Som de 500Hz de tom por 36 ticks (2 segundos); variação da frequência de

37-32767 Hz; variação dos ticks de
0-65535 (0 ticks=sem som)

SOUND 440,T*18.2

Som de uma nota de 400Hz por T segundos

SOUND 32767,T*18.2 Silêncio por T segundos

2 PLAY STRING\$

Toca a música especificada pela linguagem
da série como segue:

Caractere	Função	Variação (default)	Exemplo	
			Série	Resultado
A-G # ou + - Ln ou L=var; An-Gn A.-G. On pn P. Tn	Nome das notas Sustenido Bemol Duração da nota Nota mais duração Staccato Define uma oitava Pausa Semi pausa Tempo	1-64 (16), duração=1/n 1-64, duração=1/n 1-255, 1.5*duração 0-6, inicia em C 1-64, pausa=1/n 1-255, 1.5*duração 32-255 (120)	C G#;G+ B- L4 L=XY; A2 A2. 03 P2 P2.. n 1/4 de nota /min N1	C natural G sustenido B bemol Semínima A;1/2 A;3/4 C médio 1/2 nota 9/8 C na oitava 0
Nn MN ML MS MF MB XA\$;	Notas por números Música normal Música legato Música staccato Música no "foreground" Música no "background" Executa a subsérie A\$	1-84 (0=resto) Cada nota com 7/8 de L Cada nota com a última duração de L Cada nota com 3/4 de L BASIC espera por cada nota (default=MF) 32 notas no buffer; BASIC continua		

Programa Exemplo

A\$ = "O2 CEFG... CEFG... CEFG.. E..C..
E..D1"

:B\$ = "MLXA\$;P1;MS XA\$;":PLAY"T255XB\$;"

Linguagem de Máquina

Endereço absoluto = $16 \times \text{Segmento Atual} + \text{OFFSET}$
(ex., DEF SEG=&HB123:POKE 15,99 põe o decimal 99 na posição hex B123F).

CLEAR,32400

Memória acima do decimal 32400 disponível para programa em Linguagem de Máquina

DEF SEG = &HB800

Define o segmento atual (default=Segmento de dado do BASIC (DS))

PRINT PEEK(OFFSET)

Retorna o byte do endereço dado; OFFSET 0-65535

POKE OFFSET, BYTE

Grava o BYTE (0-255) no endereço

OFFSET=VARPTR(Q\$)

Retorna o endereço da variável entre parênteses

S\$=VARPTR\$(Q3)

Retorna o tipo e o endereço da variável como uma série de 3 bytes; da esquerda para a

direita, S\$ contém: tipo (2=inteiro, 3=série, 4=simples precisão, 8=dupla precisão);
byte inferior do endereço; byte superior do endereço

CALL XYZ (A%,B,C\$)

Chama uma sub-rotina em linguagem de máquina com endereço do offset dado pelo valor de XYZ, passando os parâmetros A%, B e C\$

DEF USR9=32768!

Offset inicial para a sub-rotina em linguagem de máquina USR9 (default=USR0)

XY=USR7 (ARGUMENTO)

Chama rotina em linguagem de máquina especificada, passando ARGUMENTO (pode ser simulado); retorna o resultado em XY

BLOAD "ML1",OFFSET

Carrega o arquivo binário na memória no OFFSET dado (default=0)

BSAVE "EX2",OFFSET,COMPRIMENTO

Salva o bloco de memória, de número de bytes dado pelo COMPRIMENTO, começando no endereço indicado

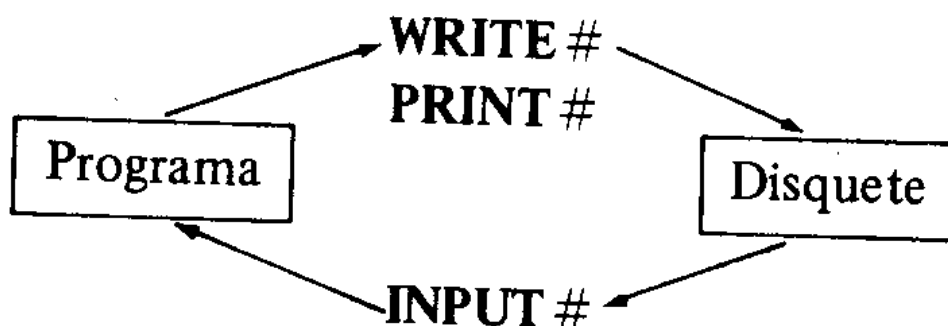
Mapa de Memória

Item	Endereço		Tamanho do Segmento Hex (Decimal)
	Decimal (K=1024)	Hex (segmento)	
64 K RAM	0-64 K	00000-0FFFF	10000 (64 K)
Buffer do monitor Monocromático	704-720 K	B0000-B3FFF	4000 (16 K)
Buffer do monitor de Gráficos/Cores	736-752 K	B8000-BBFFF	4000 (16 K)
ROM {	Do usuário	F4000-F5FFF	2000 (8 K)
	Interpretador BASIC em cassette	F6000-FDFFF	8000 (32 K)
	DOS BIOS	FE000-FFFFF	2000 (8 K)

Arquivos de Dados

Somente um arquivo de fita cassete; somente três arquivos de disquete a menos que seja usado /F:n no DOS.

Arquivos Seqüenciais São mais fáceis de usar que arquivos randômicos, mas não são tão versáteis



Gerando arquivos (gravando dados no disquete):

OPEN "DADO1" FOR OUTPUT AS#3

Abre um arquivo de nome "DADO1" para escrever no arquivo usando o buffer#3

OPEN "O",#3, "DADO1"

O mesmo que o comando acima

1 OPEN "DADO1" FOR APPEND AS#3

Abre o arquivo "DADO1" para acrescentar mais dados

PRINT #3,A;B;C\$;" ";D\$

Grava o dado seqüencialmente no arquivo #3

PRINT #3, USING" ##.##";F;G

Grava de acordo com o formato (*veja* "PRINT USING" em Monitor no Modo Texto, p. 31)

WRITE #3,A\$,B\$,C

Grava o dado no arquivo, delimitando o string com aspas, separando os itens com vírgulas

1 REGISTROS=LOC(3)

Número de registros escritos no arquivo #3

Acessando arquivos (lendo o dado):

OPEN "DADO1" FOR INPUT AS#2

Abre o arquivo para ler o buffer #2

OPEN "I",#2, "DADO1" O mesmo que acima

1 BYTES=LOF(2)

Comprimento do arquivo #2, em múltiplos de 128

IF EOF(2) THEN CLOSE

Teste de verificação de fim de arquivo; retorna -1 se verdadeiro

INPUT #2,A,B,C\$,D\$

Lê o dado do arquivo seqüencial ou do periférico

LINE INPUT #2,Q\$

Lê todos os caracteres (<=254) até um retorno de carro em Q\$

I\$=INPUT\$(C,#2)

Retorna uma série de caracteres C do arquivo #2

Fechando arquivos:

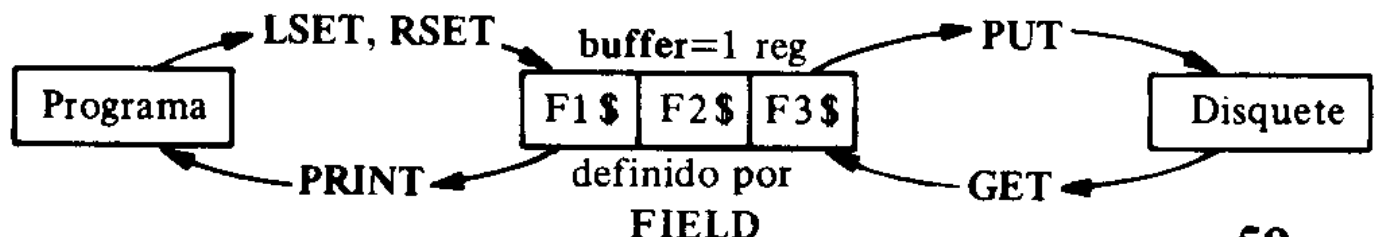
CLOSE #1,#3 Fecha os arquivos #1 e #3

CLOSE Fecha todos os arquivos e periféricos

1 **RESET**

Fecha todos os arquivos de disquete e limpa o buffer do sistema

Arquivos de Disquete Randômicos



1 OPEN "DADO4" AS #1 LEN=512

Abre o arquivo randômico para entrada ou saída *com comprimento de registro de 512 bytes* (default=128); não pode exceder a quantidade definida no DOS com opção /S:

1 OPEN "R",#1,"DADO4",64

O mesmo que o anterior

1 FIELD #1,12 AS F1\$,28 AS F2\$,3 AS F3\$

Aloca espaço no buffer do arquivo randômico para variáveis de série; não use estes nomes de série em instruções INPUT ou LET

1 ADD.FCB=VARPTR(#1)

Endereço do bloco de controle do arquivo

1 REC.NUM=LOC(3)

Último número do registro gravado ou lido

1 BYTES=LOF(4)

Comprimento do arquivo #4, em múltiplos de 128

Criando arquivos (gravando dados no disquete):

1 I\$=MKI(1%)

Converte tipos numéricos em séries, 2 bytes

1 S\$=MKSS(S!)

Converte tipos numéricos em séries, 4 bytes

1 **D\$=MKD\$(D#)**

Converte tipos numéricos em séries, 8 bytes

1 **LSET F1\$=D\$**

Coloca a série D\$ no buffer randômico F1\$,
com justificação à esquerda

1 **RSET F2\$=S\$**

Coloca a série S\$ no buffer randômico F2\$,
com justificação à direita

1 **PUT #1,REG.NÚMERO**

Escreve o registro do buffer no arquivo;
*default REG.NÚMERO é o próximo
número disponível (1-32767)*

Acessando arquivos (lendo os dados):

1 **GET #1,REG.NUM2**

Lê o registro do arquivo no buffer

1 **D\$=CVD(F1\$)**

Converte uma série de 8 bytes em um
número de **precisão dupla**

1 **PRINT CVS(F2\$)**

Converte uma série de 4 bytes em um número
de **precisão simples**

1 **I%=CVI(I\$)**

Converte uma série de 2 bytes em um número
inteiro

Fechando arquivos:

CLOSE e **RESET** como mostrado em Arquivos Sequenciais de dados na página 57.

Detecção de Erro

Se ocorrer um erro no BASIC, o programa parará, a menos que o erro seja detectado.

ON ERROR GO TO 60

Ativa o detector de erro, a sub-rotina de tratamento do erro começa na linha 60

ON ERROR GO TO 0 Desativa o detector de erro
CODE=ERR

Retorna o número do código do erro

IF ERR=24 THEN PRINT "timeout do periférico"

Mensagem do usuário sem parar o programa

LINE=ERL

Retorna o número da linha onde o erro ocorreu

ERROR N

Provoca erro no BASIC (N=código de erro padrão do BASIC)

ERROR 212

Provoca erro definido pelo usuário (preferível 200 ou maior)

RESUME 0

Continua (após uma rotina de erro) na
instrução que causou o erro

RESUME NEXT

Continua na instrução seguinte à que causou
o erro

RESUME 2980

Continua na linha dada pelo número





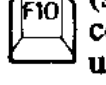





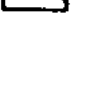
Erro: Mensagens e Números

Message	#	Message	#	Message	#
Advanced Feature	73	Division by zero	11	Out of data	4
Bad file mode	54	Duplicate DIM	10	Out of memory	7
Bad file name	64	FIELD overflow	50	Out of string space	14
Bad file number	52	FOR without NEXT	26	Overflow	6
Bad record number	63	File already exists	58	RESUME w/o error	20
Can't continue	16	File already open	55	RETURN w/o GOSUB	3
Comm. buffer overflow	69	File not found	53	String formula too complex	16
Device Fault	25	Illegal direct	12	String too long	15
Device I/O Error	57	Illegal function	5	Subscript out of range	9
Device Timeout	24	Input past end	62	Syntax error	2
Device unavailable	68	Internal error	51	Too many files	67
Direct statement in file	66	Line buffer overflow	23	Type mismatch	13
Disk Media Error	72	Missing operand	22	Undefined line number	8
Disk Write Protected	70	NEXT without FOR	1	Undefined user function	18
Disk full	61	No RESUME	19	WEND w/o WHILE	30
Disk not Ready	71	Out of Paper	27	WHILE w/o WEND	29






























Códigos de teclado expandido

Códigos de dois bytes retornados pela instrução
INKEY\$ A\$=INKEY\$:IF LEN(A\$)=2 THEN
CÓDIGO=ASC (RIGHT\$(A\$,1))

Código Tecla Pressionada

3	caractere nulo
15	 
16-25	 Q, W, E, R, T, Y, U, I, O, P
30-38	 A, S, D, F, G, H, J, K, L
44-50	 Z, X, C, V, B, N, M
59-68	 ...  (se desativada como Soft-keys usando KEY n, "")
71	
72	
73	
75	
77	
79	
80	
81	

Código Tecla Pressionada

82	
83	
84-93	  ...  
94-103	  ...  
104-113	  ...  
114	 
115	 
116	 
117	 
118	 
119	 
120-131	 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, -
132	 

Conjunto de caracteres ASCII

Gravar com CHR\$(n), ler com SCREEN (linha, coluna)

Dec	Hex	Caractere	Dec	Hex	Caractere	Dec	Hex	Caractere
0	0	(nulo)	42	2A	*	84	54	T
1	1	☺	43	2B	+	85	55	U
2	2	●	44	2C	,	86	56	V
3	3	♥	45	2D	-	87	57	W
4	4	♦	46	2E	.	88	58	X
5	5	♣	47	2F	/	89	59	Y
6	6	♠	48	30	0	90	5A	Z
7	7	(beep)	49	31	1	91	5B	[
8	8	(retrocesso)	50	32	2	92	5C	\
9	9	(tab)	51	33	3	93	5D]
10	A	(avanço de linha)	52	34	4	94	5E	^
11	B	(home)	53	35	5	95	5F	
12	C	(avanço de papel)	54	36	6	96	60	
13	D	(retorno de cursor)	55	37	7	97	61	a
14	E	♪	56	38	8	98	62	b
15	F	⚙	57	39	9	99	63	c
16	10	▶	58	3A	:	100	64	d
17	11	◀	59	3B	:	101	65	e
18	12	↕	60	3C	<	102	66	f
19	13	!!	61	3D	=	103	67	g
20	14	π	62	3E	>	104	68	h
21	15	§	63	3F	?	105	69	i
22	16	—	64	40	@	106	6A	j
23	17	‡	65	41	A	107	6B	k
24	18	!	66	42	B	108	6C	l
25	19	!	67	43	C	109	6D	m
26	1A	—	68	44	D	110	6E	n
27	1B	—	69	45	E	111	6F	o
28	1C	(cursor à direita)	70	46	F	112	70	p
29	1D	(cursor à esquerda)	71	47	G	113	71	q
30	1E	(cursor para cima)	72	48	H	114	72	r
31	1F	(cursor para baixo)	73	49	I	115	73	s
32	20	(espaço)	74	4A	J	116	74	t
33	21	!	75	4B	K	117	75	u
34	22	"	76	4C	L	118	76	v
35	23	#	77	4D	M	119	77	w
36	24	\$	78	4E	N	120	78	x
37	25	%	79	4F	O	121	79	y
38	26	&	80	50	P	122	7A	z
39	27	'	81	51	Q	123	7B	{
40	28	(82	52	R	124	7C	
41	29)	83	53	S	125	7D	}

Dec	Hex	Caractere	Dec	Hex	Caractere	Dec	Hex	Caractere
126	7E	~	170	AA	ˆ	214	D6	π
127	7F	◻	171	AB	½	215	D7	#
128	80	ϕ	172	AC	¼	216	D8	+
129	81	ü	173	AD	ı	217	D9	┘
130	82	é	174	AE	«	218	DA	┘
131	83	â	175	AF	»	219	DB	■
132	84	ä	176	B0	˙	220	DC	■
133	85	à	177	B1	⋯	221	DD	■
134	86	á	178	B2	⋮	222	DE	■
135	87	ç	179	B3		223	DF	■
136	88	ê	180	B4	┘	224	E0	ı
137	89	è	181	B5	┘	225	E1	β
138	8A	ë	182	B6	┘	226	E2	┘
139	8B	ı	183	B7	ı	227	E3	π
140	8C	ı	184	B8	ı	228	E4	Σ
141	8D	ı	185	B9	ı	229	E5	σ
142	8E	À	186	BA		230	E6	μ
143	8F	Á	187	BB	ı	231	E7	τ
144	90	Ê	188	BC	ı	232	E8	ϕ
145	91	α	189	BD	ı	233	E9	ϕ
146	92	Æ	190	BE	┘	234	EA	Ω
147	93	ô	191	BF	┘	235	EB	δ
148	94	ó	192	C0	┘	236	EC	∞
149	95	ò	193	C1	┘	237	ED	∅
150	96	û	194	C2	┘	238	EE	€
151	97	ü	195	C3	┘	239	EF	∩
152	98	v	196	C4	—	240	F0	≡
153	99	O	197	C5	+	241	F1	±
154	9A	U	198	C6	┘	242	F2	≥
155	9B	e	199	C7	┘	243	F3	≤
156	9C	£	200	C8	ı	244	F4	ı
157	9D	¥	201	C9	ı	245	F5	J
158	9E	Pts	202	CA	ı	246	F6	÷
159	9F	ı	203	CB	ı	247	F7	≈
160	A0	á	204	CC	┘	248	F8	◦
161	A1	ı	205	CD		249	F9	•
162	A2	ó	206	CE	┘	250	FA	•
163	A3	ú	207	CF	ı	251	FB	√
164	A4	ñ	208	D0	┘	252	FC	n
165	A5	Ñ	209	D1	ı	253	FD	z
166	A6	a	210	D2	ı	254	FE	■
167	A7	o	211	D3	ı	255	FF	(espaço em branco 'FF')
168	A8	ç	212	D4	ı			
169	A9	┘	213	D5	ı			

Índice Geral dos Comandos em BASIC

A

ABS, 23
AND, 28, 42
ASC, 28
ATN, 23
AUTO, 18

B

BEEP, 53
BLOAD, 16, 56
BSAVE, 16, 56

C

CALL, 56
CDBL, 27
CHAIN, 16
CHR\$, 28
CINT, 23, 27
CIRCLE, 37
CLEAR, 19, 55
CLOSE, 59
CLS, 31
COLOR (text), 33
COLOR (gr.), 35
COM, 47
COMMON, 21
CONT, 19
COS, 23
CSNG, 27
CSRLIN, 32
CVD/I/S, 61

D

DATA, 25
DATE\$, 21
DEF FNxx, 21, 24
DEFDBL, 21
DEFINT, 21
DEFSNG, 21
DEFSTR, 21
DEF SEG, 35, 55
DEFUSR, 21, 56
DELETE, 18
DIM, 21
DRAW, 39

E

EDIT, 18
ELSE, 25
END, 25
EOF, 48, 59
EQV, 28
ERASE, 22
ERL, 62
ERR, 62
ERROR, 62
EXP, 23

F

FIELD, 60
FILES, 17
FIX, 23
FNxxx, 24

FOR, 25
FRE, 19

G

GET (files), 61
GET (gr.), 41
GOSUB, 25
GOTO, 25

H

HEX\$, 27

I

IF, 25
IMP, 28
INKEY\$, 43, 65
INP, 53
INPUT, 43
INPUT #, 47, 57
INPUT\$, 47, 59
INSTR, 28
INT, 23

K

KEY, 44
KILL, 17

L

LEFT\$, 29
LEN, 29, 47, 60
LET, 20
LINE, 37
LINE INPUT, 44
LINE INPUT #, 47, 59
LIST, 18
LLIST, 18, 49

LOAD, 17
LOC, 48, 58, 60
LOCATE, 32
LOF, 48, 58, 60
LOG, 23
LPOS, 49
LPRINT, 49
LPRINT USING, 50
LSET, 59

M

MERGE, 17
MID\$, 29
MKD/I/S\$, 61
MOD, 28
MOTOR, 53

N

NAME, 18
NEW, 19
NEXT, 25
NOT, 28

O

OCT\$, 27
OFF, 44, 48, 50, 52
ON, 44, 48, 50, 52
ON COM, 48
ON ERROR, 62
ON GOSUB, 26
ON GOTO, 26
ON KEY, 45
ON PEN, 52
ON STRIG, 51
OPEN, 50, 58, 60
OPEN "COM, 46, 47

OPTION BASE, 22
OR, 28, 42
OUT, 53

P

PAINT, 38
PEEK, 55
PEN, 51
PLAY, 54
POINT, 30, 37
POKE, 55
POS, 32
PRESET, 30, 36, 42
PRINT (?), 30, 34
PRINT USING, 34
PRINT #, 57
PRINT # USING, 58
PSET, 30, 36, 42
PUT(files), 61
PUT(gr.), 41

R

RANDOMIZE, 22
READ, 26
REM ('), 26
RENUM, 20
RESET, 59
RESTORE, 26
RESUME, 63
RETURN, 26
RIGHT\$, 29
RND, 23
RSET, 59
RUN, 18

S

SAVE, 18
SCREEN, 30, 32, 35
SGN, 23
SIN, 23
SOUND, 53
SPACE\$, 29
SPC(, 29
SQR, 23
STEP, 25, 36
STICK, 50
STOP, 20, 51
STR\$, 27
STRIG, 50
STRING\$, 29
SWAP, 23, 29
SYSTEM, 20

T

TAB(, 35
TAN, 23
THEN, 25
TIME\$, 22
TO, 25
TROFF, 20
TRON, 20

U

USR, 56

V

VAL, 27
VARPTR, 55, 60
VARPTR\$, 39, 55

W
WAIT, 53
WEND, 26
WHILE, 26
WIDTH, 32, 49

WRITE, 35
WRITE#, 57

X
XOR, 28, 42

OUTROS LIVROS NA ÁREA

Guias do Operador

- Gifford — Apple II — Comandos Básicos
- Ingraham — CP/M — Comandos Básicos
- Wilson — VisiCalc — Comandos Básicos

Guias do Usuário

- Baras — Lotus 1 - 2 - 3 — Guia do Usuário
- Castlewitz — VisiCalc — Guia do Usuário
- Hoffman — MS - DOS — Guia do Usuário
- Sachs — IBM PC — Guia do Usuário
- Townsend — dBase II — Guia do Usuário