

KIT 2+

- MANUAL DE INSTRUÇÕES



Manual KIT 2.0 e 2+
=====

Por Ademir Carchano

Publicado pela ACVS Eletrônica Ltda.

Este manual contém informações sobre as novas instruções e as modificações feitas nas já existentes no BASIC MSX 1.0.

SUMARIO =====

Apresentação.....	5
Descrição da sintaxe usada no manual.....	6
Instruções gráficas.....	7
SCREEN	9
COLOR	11
COLOR=	15
COLOR=NEW	16
COLOR=RESTORE	17
COLOR SPRITE	18
COLOR SPRITE\$	19
COPY	20
SET PAGE	22
SET SCROLL	23
WIDTH	24
CIRCLE, DRAW, LINE, PAINT, POINT, PRESET E PSET	25
Relógio interno	27
GET DATE	29
SET DATE	30
GET TIME	31
SET TIME	32
SET ADJUST	33
SET BEEP	34
SET PASSWORD	35
SET PROMPT	36
SET SCREEN	37
SET TITLE	38
RAM-Disk interna	39
CALL MEMINI	41
CALL MFILES	42
CALL MKILL	43
CALL MNAME	44
Turbo-BASIC	45
CALL AC	47
CALL RUN	48
CALL TURBO ON/OFF	49
#C	50
#N	51
#I	52
Miscelânea e VDP	53
PAD	55
PDL	57
BASE	58
VDP	59

Esta página foi deixada em branco propositalmente.

APRESENTAÇÃO

=====

As informações deste manual permitem que seja usada toda potencialidade do novo BASIC 2.0 e do TURBO-BASIC incorporados nos KIT 2.0 e KIT 2+ fabricados pela ACVS Eletrônica Ltda.

As principais características do computador, depois da instalação do KIT 2.0 ou do KIT 2+, são:

- Texto de até 80 colunas (mesmo pela televisão).
- Resolução gráfica máxima de 512 x 424 pontos com 16 cores de 512 combinações.
- Resolução máxima de cores, definidas ponto a ponto, de 256 com o KIT 2.0 e 19.260 com o KIT 2+.
- Memória ROM do BASIC com 48 KBytes (KIT 2.0) e 64 KBytes (KIT2+)
- Memória RAM de VIDEO de 128 KBytes.
- Trabalha com várias páginas gráficas.
- SCREEN 0 a 8 (KIT 2.0) - SCREEN 0 a 12 (KIT 2+)
- Memória ROM de 16 KBytes com TURBO-BASIC que acelera a execução de um programa em até seis vezes ou mais.
- Relógio interno mantido por baterias que atualiza hora e data mesmo com o computador desligado.
- Velocidade de resposta do teclado aumentada.
- Processador de vídeo que executa a maior parte das funções gráficas acelerando o processamento.
- Expansão de SLOT interna para 4 sub-slots:

	KIT 2.0	KIT 2+
SLOT 1.0	- Cartridge A	Cartridge A
SLOT 1.1	- SUBROM+TURBO-BASIC	SUBROM
SLOT 1.2	- Livre	TURBO-BASIC
SLOT 1.3	- Livre	Livre

- Saída de RF melhorada para acompanhar as novas características de resolução de pontos e cores.
- RAMDISK interna com até 32 KBytes.
- Instruções SET PASSWORD, SETTITLE, SETPROMPT, etc.
- Aceita a instalação da MEGA-MAPPER interna de 256 KBytes de memória RAM para o usuário.

ACVS Eletrônica Ltda.

Av. Paulista, 2001 - 9º andar - conj. 912 - CEP 01311 - São Paulo - SP.
Tel. (011) 289-7694

Descrição da sintaxe usada neste manual

NOME [(X)]	(TIPO)
------------	--------

- 1 - O nome do comando, instrução ou função está em letras maiúsculas e deverá ser escrito exatamente como aparece.
- 2 - Itens entre parênteses () devem ser definidos pelo usuário.
- 3 - Itens entre colchetes [] são opcionais.
- 4 - Todas as pontuações devem ser escritas como aparecem no formato. Isto se aplica a pontos finais, vírgulas, parênteses, dois pontos, ponto e vírgula, etc.
- 5 - X, Y e Z representam expressões numéricas.
- 6 - X\$, Y\$ e Z\$ representam expressões alfanuméricas.

(TIPO) - Indica se é um COMANDO, uma INSTRUÇÃO ou FUNÇÃO. A diferença entre os tipos é a seguinte:

- COMANDO: É a ordem que depois de executada coloca o computador no modo de comando direto via teclado. Por exemplo: LIST.
- INSTRUÇÃO: É a ordem que pode ser dada via teclado ou dentro de um programa. Por exemplo PRINT, GOTO, GOSUB, etc.
- FUNÇÃO: só é válida acompanhada por uma INSTRUÇÃO. Por exemplo: PRINT BASE(n)

INSTRUÇÕES GRAFICAS

SCREEN	9
COLOR	11
COLOR=	15
COLOR=NEW	16
COLOR=RESTORE	17
COLOR SPRITE	18
COLOR SPRITE\$	19
COPY	20
SET PAGE	22
SET SCROLL	23
WIDTH	24
CIRCLE, DRAW, LINE, PAINT, POINT, PRESET E PSET	25

Esta página foi deixada em branco propositalmente.

```
=====
SCREEN [X [,Y [,Z [,XX [,YY [,ZZ ]]]]]
=====
```

Instrução

Define o modo de tela que será usado, o tamanho dos SPRITES, o chaveamento do CLICK do teclado, a velocidade (baud rate) do gravador cassete e o tipo de impressora utilizada.

X é o modo de tela que será usado. Podendo ser:

- 0 - modo texto1 - 40x24 ou 80x24 (dependendo da instrução WIDTH)
- 1 - modo texto2 - 32x24
- 2 - modo gráfico - 256 x 192 - 16 cores de 512 - SPRITE monocromático
- 3 - modo gráfico - 64 x 48 - 16 cores de 512 - SPRITE monocromático
- 4 - modo gráfico - 256 x 192 - 16 cores de 512 - SPRITE multicolorido
- 5 - modo gráfico - 256 x 212 - 16 cores de 512 - " "
- 6 - modo gráfico - 512 x 212 - 4 cores de 512 - " "
- 7 - modo gráfico - 512 x 212 - 16 cores de 512 - " "
- 8 - modo gráfico - 256 x 212 - 256 cores de 256 - " "
- 9 - não existe
- 10 - modo gráfico - 256 x 212 - 12.499 cores / 16 de 512 / 0 a 15
- 11 - modo gráfico - 256 x 212 - 12.499 cores / 16 de 512 / 0 a 255
- 12 - modo gráfico - 256 x 212 - 19.268 cores

As SCREENs 10, 11 e 12 só são ativas no computador com o KIT 2+.

O número de cores na tabela corresponde ao número máximo que pode ser mostrado simultaneamente por página de vídeo.

A resolução vertical pode ser dobrada para 424 usando-se o modo entrelaçado de exibição (ver item ZZ).

Y determina o tamanho dos SPRITES:

- 0 - 8x8 magnitude 1 (8x8) 2 - 16x16 magnitude 1 (16x16)
- 1 - 8x8 magnitude 2 (16x16) 3 - 16x16 magnitude 2 (32x32)

Z ativa ou desativa o CLICK do teclado:

- 0 - desativa o CLICK 1 - ativa CLICK

XX define a velocidade de transmissão dos dados para o gravador cassete (baud rate):

- 1 - 1200 baud 2 - 2400 baud

YY define o tipo de impressora que está sendo usada:

- 0 - MSX diferente de 0 - não MSX

ZZ define o modo de exibição da tela. Podendo ser:

- 0 - Normal
- 1 - Entrelaçado
- 2 - Não entrelaçado, alternando páginas ímpares e pares
- 3 - Entrelaçado, alternando páginas ímpares e pares

Nos modos de exibição 2 e 3, o número da página de exibição deve ser ímpar (ver instrução SETPAGE).

Exemplo:

```
10 COLOR 15,0,0
20 FOR I= 2 TO 8 : SCREEN I
30 FOR N=0 TO 255:LINE (0,0)-(N,211),C
40 IF I=8 THEN C=N ELSE C=C+1:IF C=16 THEN C=1
50 NEXT
60 FOR T=0 TO 1000:NEXT
70 NEXT
```

```
10 SCREEN 7,,,,,3
20 LINE(80,80)-(400,200),4,BF
30 SETPAGE1,1:CLS
40 LINE(100,100)-(380,180),4,BF
50 GOTO50
```

=====

COLOR [X] [,Y] [,Z]

=====

Instrução

Define as cores que serão utilizadas na tela nos modos texto e gráfico.

X define a cor do caractere.

Y define a cor do fundo da tela.

Z define a cor da borda da tela.

X, Y e Z devem ser números inteiros.

Nas SCREENs 0, 1, 2, 3, 4, 5 e 7, os números X, Y e Z devem estar compreendidos entre 0 e 15.

Nas SCREENs 5 e 7 cada byte guarda a cor de dois pontos em sequência.

A cor correspondente a cada número na paleta de cores é mostrada na tabela a seguir:

Nº na paleta	Nome da cor	Intensidade:		
		Verm.	Verde	Azul
0	transparente	0	0	0
1	preta	0	0	0
2	verde média	1	6	1
3	verde clara	3	7	3
4	azul escura	1	1	7
5	azul clara	2	3	7
6	vermelha escura	5	1	1
7	ciano	2	6	7
8	vermelha média	7	1	1
9	vermelha clara	7	3	3
10	amarela escura	6	6	1
11	amarela clara	6	6	4
12	verde escura	1	4	1
13	magenta	6	2	5
14	cinza	5	5	5
15	branca	7	7	7

A intensidade do vermelho, verde e azul na paleta de cores pode ser alterada com a instrução COLOR=.

Na SCREEN 6, X, Y e Z devem estar entre 0 e 3. Neste modo gráfico, se a cor da borda da tela estiver entre 0 e 3, será respeitada a tabela da paleta de cores. Porém, se a cor estiver entre 16 e 31, será feita uma combinação das cores conforme a tabela a seguir:

Nº da cor da borda	Combinação entre as cores:
16	0 e 0
17	0 e 1
18	0 e 2
19	0 e 3
20	1 e 0
21	1 e 1
22	1 e 2
23	1 e 3
24	2 e 0

25	2	e	1
26	2	e	2
27	2	e	3
28	3	e	0
29	3	e	1
30	3	e	2
31	3	e	3

Na SCREEN 8, X, Y e Z devem estar entre 0 e 255. Neste modo gráfico cada byte corresponde a cor de um ponto. Para saber o número da cor de acordo com a intensidade de vermelho, verde e azul, use a equação abaixo:

$$R * 32 + G * 4 + B$$

onde R, G e B são respectivamente as intensidades de vermelho, verde e azul.

O vermelho e o verde podem variar entre 0 e 7. O azul só pode variar entre 0 e 3.

A formação do byte da cor na SCREEN 8 é a seguinte:

bit - 7 6 5 4 3 2 1 0

cor - R R R G G G B B

Nas SCREEN 10, 11 e 12 é usado o sistema YJK para mostrar as cores. Neste modo especial 4 (quatro) bytes são usados para compor as cores de quatro pontos em sequência na mesma linha horizontal:

BIT	7	6	5	4	3	2	1	0	Pontos:
Byte 1	-	Y1	Y1	Y1	Y1	KL	KL	KL	1 2 3 4 --- 0 0 0 0
Byte 2	-	Y2	Y2	Y2	Y2	KH	KH	KH	-----+ + +
Byte 3	-	Y3	Y3	Y3	Y3	JL	JL	JL	-----+ +
Byte 4	-	Y4	Y4	Y4	Y4	JH	JH	JH	-----+ +

Os (Y) são a intensidade da cor (brilho) e podem variar de 0 a 31. Cada valor de Y corresponde a 8 (oito) níveis na escala de intensidade (0=0, 1=8, 2=16,..., 31=248).

Os (J e K) são os vetores que determinam qual cor será usada. Podem variar de -32 a 31 (6 bits). O gráfico a seguir permite visualizar como é feita a seleção da cor:

		laranja		
		J(31)		amarelo
	+	+	+	
	+	+	+	
	+	+	+	
magenta K(-32)	+	+	+	
	+	0	+	K(31) amarelo claro
	+	+	+	
	+	+	+	
	+	+	+	
azul		J(-32)		verde
		ciano		

A conversão de um sistema para outro é feita com as seguintes equações:

De YJK para RGB

$$\begin{aligned} R &= Y + J \\ G &= Y + K \\ B &= 5/4 Y - 1/2 J - 1/4 K \end{aligned}$$

de RGB para YJK

$$\begin{aligned} Y &= B/2 + R/4 + G/2 \\ J &= R - Y \\ K &= G - Y \end{aligned}$$

Nas SCREENs 10 e 11 os bytes das cores têm o seguinte formato:

BIT	-	7	6	5	4	3	2	1	0
PONTO 1	-	Y1	Y1	Y1	Y1	A	KL	KL	KL
PONTO 2	-	Y2	Y2	Y2	Y2	A	KH	KH	KH
PONTO 3	-	Y3	Y3	Y3	Y3	A	JL	JL	JL
PONTO 4	-	Y4	Y4	Y4	Y4	A	JH	JH	JH

O Y de cada ponto pode variar de 0 a 15 (0 a 240 na intensidade), K e J permanecem como na SCREEN 12 variando de -32 a 31.

SCREEN 10 - as cores são formadas como na SCREEN 11, porém, as instruções gráficas só trabalham com valores para as cores entre 0 e 15.

SCREEN 11 - as cores são formadas no sistema YJK ou com a paleta normal de 16 cores, dependendo do bit 3 (A). Se for 0, as cores são mostradas no sistema YJK, considerando o Y de 4 bits (0 a 240 de intensidade), obtendo-se assim as 12.499 variações possíveis. Se o bit 3 (A) for 1, as cores são as da paleta conforme os bits Y (0 a 15). Neste caso, os bits J e K são ignorados. As instruções gráficas nesta SCREEN podem usar valores para as cores entre 0 e 255.

As informações referentes às SCREENs 10, 11 e 12 só são válidas se o KIT 2+ estiver instalado no computador.

Quando não for dado nenhum valor a X, Y ou Z, a instrução COLOR utilizará os últimos valores aplicados às cores dos caracteres, fundo e borda da tela.

Exemplo 1:

```
10 SCREEN7
20 FORX=0TO511
30 LINE(255,0)-(X,211),C
40 C=C+1:IFC=16THENC=0
50 NEXT
60 GOTO60
```

Exemplo 2:

```
10 SCREEN8
20 FORX=0TO255
30 LINE(0,0)-(X,211),X
40 LINE(255,211)-(255-X,0),X
50 NEXT
60 GOTO60
```

Exemplo 3: SOMENTE PARA KIT 2+.

```
100 SCREEN12
110 Y=93
120 FOR JH=0 TO 3
130 FOR JL=0 TO 7
140 FOR KH=0 TO 3
150 FOR KL=0 TO 7
160 LINE (X,Y)-(X,Y+2),KL,BF
170 LINE (X+1,Y)-(X+1,Y+2),KH,BF
180 LINE (X+2,Y)-(X+2,Y+2),JL,BF
190 LINE (X+3,Y)-(X+3,Y+2),JH,BF
200 X=X+4 : IF X=128 THEN X=0 : Y=Y-3
210 NEXT KL,KH,JL,JH
220 COPY (0,0)-(127,95) TO (128,0)
230 COPY (0,0)-(255,95) TO (0,96)
240 FOR X=0 TO 127 STEP 4
250 LINE (X+1,0)-(X+1,191),4,,OR
260 NEXT X
270 FOR X=0 TO 255 STEP 4
280 LINE (X+3,96)-(X+3,191),4,,OR
290 NEXT X
300 FOR N=0 TO 31
310 LINE (0,0)-(255,191),7,BF,AND
320 LINE (0,0)-(255,191),CC,BF,OR
330 FOR T=0 TO 300 : NEXT T
340 CC=CC+8
350 NEXT N
360 IF INKEY$="" THEN 300
```

=====

```
COLOR = ( X, R, G, B )
```

=====

Instrução

Altera as intensidades do vermelho, verde e azul na paleta de cores.

X é o número da cor na paleta e deve estar entre 0 e 15.

R é a intensidade do vermelho e deve estar entre 0 e 7.

G é a intensidade do verde e deve estar entre 0 e 7

B é a intensidade do azul e deve estar entre 0 e 7

Esta instrução pode ser usada em qualquer SCREEN, exceto em SCREEN 8 que não usa a paleta de cores.

Exemplo:

```
100 SCREEN5
110 OPEN"GRP:"AS#1
120 FORN=0TO7
130 PRESET(N*32+6,40):PRINT#1,"COR"
140 PRESET(N*32+6,50)
150 PRINT#1,N+1
160 NEXT
170 FORX=0TO7
180 LINE(X*32,60)-(X*32+32,140),X+1,BF
190 LINE(X*32+32,141)-(X*32+64,211),15-X,BF
200 LINE(X*32,35)-(X*32+32,211),10,B
210 NEXT
220 PRESET(0,10)
230 FORN=0TO7
240 PRINT#1," RGB";
250 NEXT
260 FORC=0TO7
270 COLOR=(C+1,R,G,B)
280 R$=RIGHT$(STR$(R),1)
290 G$=RIGHT$(STR$(G),1)
300 B$=RIGHT$(STR$(B),1)
310 PRESET(C*32+8,20):PRINT#1,R$;G$;B$
320 B=B+1:IFB=8THENB=0:G=G+1:IFG=8THENG=0:R=R+1:IFR=8THENR=0
330 NEXT
340 GOTO260
```

```
=====
COLOR = NEW                                     Instrução
=====
```

Faz com que as intensidades de vermelho, verde e azul da paleta de cores retornem aos seus valores originais.

Os valores originais de vermelho, verde e azul são aqueles mostrados na tabela junto à instrução COLOR.

Exemplo:

```
10 SCREEN 5 : COLOR 15,0,0
20 OPEN "GRP:" AS #1
30 FOR C=2 TO 15
40 COLOR C
50 PRINT#1,"ALTERAÇÃO DAS CORES DA PALETA"
60 NEXT C
70 FOR N=2 TO 15
80 R=INT(RND(1)*7)
90 G=INT(RND(1)*7)
100 B=INT(RND(1)*7)
110 COLOR=(N,R,G,B)
120 NEXT N
130 FOR T=0 TO 1000 : NEXT T : BEEP
140 COLOR = RESTORE
150 IF INKEY$="" THEN 150
```

=====

COLOR = RESTORE

=====

Instrução

Transfere da memória de vídeo para a paleta de cores os valores das intensidades do vermelho, verde e azul.

Esta instrução é usada quando os valores das intensidades do vermelho, verde e azul são carregados para a memória de vídeo com a instrução BLOAD "nome do arquivo",S.

A tabela da paleta de cores pode ser encontrada nos seguintes endereços na memória de vídeo:

SCREEN 0 (40 colunas)	&H0400 - &H0420
SCREEN 0 (80 colunas)	&H0F00 - &H0F20
SCREEN 1	&H2020 - &H2040
SCREEN 2	&H2020 - &H2040
SCREEN 3	&H2020 - &H2040
SCREEN 4	&H2020 - &H2040
SCREEN 5	&H7680 - &H76A0
SCREEN 6	&H7680 - &H76A0
SCREEN 7	&HFA80 - &HFAA0
SCREEN 8	&HFA80 - &HFAA0

Nas SCREENs 5, 6, 7 e 8 pode-se trabalhar com mais de uma página de vídeo (ver instrução SET PAGE). Para descobrir o endereço da tabela da paleta de cores use os seguintes cálculos:

SCREEN 5 e 6 --> número da página * &H08000 + &H7680

SCREEN 7 e 8 --> número da página * &H10000 + &HFA80

Exemplo:

```
10 SCREEN 0 : WIDTH 40 : COLOR 15,0,0
20 LOCATE 10,10 : PRINT "COR 15 NORMAL"
30 FOR T=0 TO 500 : NEXT : BEEP
40 COLOR= (15,3,2,1)
50 LOCATE 10,10 : PRINT "COR 15 ALTERADA"
60 BSAVE "TESTECOR.PIC",&H0400,&H0420,S
70 COLOR=NEW
80 BLOAD "TESTECOR.PIC",S
90 FOR T=0 TO 500 : NEXT : BEEP
95 COLOR = RESTORE
```

=====

COLOR SPRITE (X) = Y

=====

Instrução

Define a cor e algumas características de um SPRITE.

Esta instrução só funciona nos modos gráficos de SCREEN 4 em diante.

X é o número do SPRITE, e deve ser igual ao número na instrução SPRITE\$ que define as características do SPRITE.

Y deve ser um número inteiro compreendido entre 0 e 111. E tem o seguinte significado:

- Entre 0 e 15 o Y é o número da cor da paleta de cores.

- Se for somado 32 ao número da paleta de cores, além da cor escolhida, é desativado o sensor de coincidência de SPRITES. Ou seja, se este SPRITE colidir com outro na tela, a instrução ON SPRITE GOSUB não será executada.

- Somando-se 64 ao número da paleta de cores, além da cor escolhida, é desativado o sensor de coincidência de SPRITES e é feita uma operação lógica OR com as cores dos SPRITES que estão sobrepostos.

As cores dos SPRITES podem ser determinadas pelas instruções: PUTSPRITE, COLOR SPRITE ou COLOR SPRITE\$. A instrução que for executada por último é que determinará a cor do SPRITE. Quando forem usadas as instruções COLOR SPRITE ou COLOR SPRITE\$, não se deve definir o número da paleta de cores na instrução PUT SPRITE.

Detalhes da instrução PUT SPRITE podem ser encontrados no manual do MSX 1.

Exemplo:

```
100 SCREEN5,1:COLOR15,0,0
110 FORX=0TO15:LINE(X*16,40)-STEP(16,120),15-X,BF:NEXT
120 B$=""
130 FORI=1TO8:READA$:B$=B$+CHR$(VAL("&b"+A$)):NEXT
140 SPRITE$(0)=B$
150 SPRITE$(1)=B$
160 COLORSPRITE(1)=33 'SOMBRA
170 COLORSPRITE$(0)=CHR$(12)+CHR$(10)+CHR$(5)+CHR$(8)+CHR$(3)+CHR$(3)
180 FORI=0TO212
190 PUTSPRITE0,(I,I),,0
200 PUTSPRITE1,(I+8,I+8),,1
210 NEXT
220 GOTO220
230 DATA 10011001
240 DATA 00111100
250 DATA 01111110
260 DATA 11111111
270 DATA 11111111
280 DATA 01111110
290 DATA 01000010
300 DATA 10000001
```

COLOR SPRITE\$ (X) = Y\$	Instrução
--------------------------	-----------

Define a cor das linhas de um SPRITE.

Esta instrução só funciona nos modos gráficos da SCREEN 4 em diante.

X é o número do SPRITE, e deve ser igual ao número na instrução SPRITE\$ que define as características do SPRITE.

Y\$ deve ser uma expressão com 1 até 16 caracteres. Cada caractere correspondendo a uma linha do SPRITE. O código dos caracteres de acordo com o seu valor tem o seguinte significado:

Entre 0 e 15 o Y\$ é o número da cor da paleta de cores.

Se for somado 32 ao número da paleta de cores, além da cor escolhida, na linha do SPRITE é desativado o sensor de coincidência de SPRITES. Ou seja, se a linha do SPRITE colidir com outra na tela, a instrução ON SPRITE 60SUB não será executada.

Somando-se 64 ao número da paleta de cores, além da cor escolhida, na linha do SPRITE é desativado o sensor de coincidência de SPRITES e é feita uma operação lógica OR com as cores dos SPRITES que estão sobrepostos.

Se for somado 128 ao número da paleta de cores, a linha do SPRITE se deslocará 32 pontos para a esquerda.

As cores dos SPRITES podem ser determinadas pelas instruções: PUTSPRITE, COLOR SPRITE ou COLOR SPRITE\$. A instrução que for executada por último é que determinará a cor do SPRITE. Quando forem usadas as instruções COLOR SPRITE ou COLOR SPRITE\$, não se deve definir o número da paleta de cores na instrução PUT SPRITE.

Detalhes da instrução PUT SPRITE podem ser encontrados no manual do MSX 1.

Exemplo: ver instrução COLOR SPRITE

COPY origem TO destino	Instrução
COPY (X1,Y1) - (X2,Y2) [,XX] TO (X3,Y3) [,YY] [,op. lógica]	
COPY (X1,Y1) - (X2,Y2) [,XX] TO matriz	
COPY (X1,Y1) - (X2,Y2) [,XX] TO "periférico: nome do arquivo"	
COPY matriz [,sentido] TO (X3,Y3) [,YY] [,operação lógica]	
COPY matriz TO "periférico: nome do arquivo"	
COPY "periférico: nome do arquivo" [,sentido] TO (X3,Y3) [,YY] [,oper. lóg.]	
COPY "periférico: nome do arquivo-1" TO "periférico: nome arquivo-2"	

Copia dados de uma tela gráfica para outro ponto da tela, para outra página de tela gráfica, para uma matriz ou para um arquivo em disquete. Traz dados de uma matriz ou que estejam gravados em disquetes para a tela gráfica. Copia um arquivo de um disquete para outro, ou faz cópia do arquivo no mesmo disquete com outro nome.

Esta instrução só é válida nas SCREENS 5, 6, 7 e 8.

X1 e Y1 são as coordenadas do início do bloco que será copiado.

X2 e Y2 são as coordenadas do fim do bloco que será copiado.

X3 e Y3 são as coordenadas de destino, ou seja, o ponto na tela gráfica para onde os dados começarão a ser transferidos.

X1, X2 e X3 são números inteiros e podem variar entre 0 e 255 nas SCREENS 5 e 8, e entre 0 e 511 nas SCREENS 6 e 7.

Y1, Y2 e Y3 podem variar entre 0 e 211 nas SCREENS 5, 6, 7 ou 8.

Se a palavra STEP for usada antes das coordenadas, X e Y passam a indicar valores que serão somados à última coordenada usada por algum comando gráfico.

Nas SCREENS 5, 6, 7 e 8 pode-se trabalhar com mais de uma página de vídeo (ver instrução SET PAGE). Nestes casos:

XX e YY representam o número da página de origem e o número da página de destino, respectivamente. Se XX e YY não forem determinados, será assumida a página que está sendo usada.

[Sentido]: determina o sentido em que os dados serão transferidos. Pode variar entre 0 e 3 e tem o seguinte significado:

- 0 - da esquerda superior para a direita inferior.
- 1 - da direita superior para a esquerda inferior.
- 2 - da esquerda inferior para a direita superior.
- 3 - da direita inferior para a esquerda superior.

Matriz: é o nome da variável (matriz numérica) que deve ter tamanho suficiente para receber os dados da tela gráfica. O tamanho da matriz pode ser calculado com a seguinte equação:

$$\text{INT} ((\text{pixel} * (\text{ABS}(X2-X1)+1) * (\text{ABS}(Y2-Y1)+1) + 7) / 8) + 4$$

Pixel nas SCREENs 5, 7 e 8 vale 4 (quatro) e na SCREEN 6 vale 2 (dois).

[Operação]: é uma operação lógica que pode ser feita entre as cores de origem e as cores de destino da área copiada. O resultado da operação é visto na região de destino. As operações lógicas podem ser:

OR - $C2 = C1 + C2$

XOR - $C2 = \text{NOT}(C1) * C2 + C1 * \text{NOT}(C2)$

AND - $C2 = C1 * C2$

PSET - $C2 = C1$

PRESET - $C2 = \text{NOT}(C1)$

TOR, TXOR, TAND, TPSET e TPRESET, executam as mesmas operações lógicas descritas acima com a diferença de não afetarem a cor transparente (0).

C1 é o número da cor de origem.

C2 é o número da cor de destino.

NOT(cor) significa a inversão bit a bit do número da cor.

Periférico: é a letra que indica qual controlador de disquete será usado, podendo ser: A, B, C, D, E ou F dependendo de quantas nterfaces de drives estão ligadas ao computador.

Nome do arquivo-1 e nome do arquivo-2 são, respectivamente, os nomes dos arquivos de origem e de destino gravados no disquete.

Exemplo:

```
100 SCREEN7,0:COLOR15,0,0
110 X1=0:Y1=0:X2=64:Y2=32
120 TM=INT((4*(ABS(X2-X1)+1)*(ABS(Y2-Y1)+1)+7)/8)+4
130 DIM A(TM)
140 FORX=1TO7
150 CIRCLE(16,16),X*2-1,X:CIRCLE(16,16),X*2,X
160 CIRCLE(48,16),X*2-1,8+X:CIRCLE(48,16),X*2,8+X
170 NEXT
180 FORC=1TO7
190 COLOR=(C,0,0,7-C/2):COLOR=(C+8,7-C/2,0,0)
200 NEXT
210 COPY(0,0)-STEP(64,32)TOA
220 FORN=1TO100
230 X=RND(N)*256:Y=RND(N)*200
240 COPYA,0TO(X,Y),,TPSET
250 NEXT
260 COPY(0,0)-(350,211)TO(200,0),,TPSET
270 GOTO270
```

=====

Seleciona a página da tela que será mostrada e a página que será utilizada para desenhos (página ativa).

Esta instrução só é válida nas SCREENs 5, 6, 7, 8, 10, 11 e 12

X determina a página que será mostrada. Deve ser um número inteiro entre 0 e 3 dependendo da SCREEN selecionada.

Y determina a página que será utilizada (ativa). Deve ser um número inteiro entre 0 e 3 dependendo da SCREEN selecionada.

As páginas de cada SCREEN são:

SCREEN 5 - 4 páginas (0 a 3)
SCREEN 6 - 4 páginas (0 a 3)
SCREEN 7 - 2 páginas (0 e 1)
SCREEN 8 - 2 páginas (0 e 1)

SCREEN 10 - 2 páginas (0 e 1)
SCREEN 11 - 2 páginas (0 e 1)
SCREEN 12 - 2 páginas (0 e 1)

As SCREENs 10 a 12 só são acessadas se o Kit 2+ estiver instalado no computador.

Exemplo:

```
10 SCREEN 5: COLOR 15,0,0
20 LINE (20,20)-(220,200),4,BF
30 SETPAGE 0,1:CLS
40 LINE (40,40)-(200,180),8,BF
50 SETPAGE 0,2:CLS
60 LINE (60,60)-(180,160),12,BF
70 SETPAGE 0,3:CLS
80 CIRCLE (127,105),80,4
90 PAINT (127,105),10,4
100 FOR P=0 TO 3
110 SETPAGE P,P
120 FOR T=0 TO 300 :NEXT T:BEEP
130 NEXT P
140 GOTO 100
```

=====

```
SET SCROLL [X] [,Y] [,Z] [,XX]
```

=====

Instrução

Determina uma coordenada a ser usada como ponto inicial a partir do qual a tela será mostrada no vídeo.

X é um número inteiro entre 0 e 511 e define a coordenada X (horizontal)

Y é um número inteiro entre 0 e 255 e define a coordenada Y (vertical).

Z determina se será feito SCROLL nas 8 colunas da esquerda. Podendo ser:

0 - as 8 colunas da esquerda da tela são mostradas.

1 - as 8 colunas da esquerda da tela não aparecem.

XX determina se haverá troca de páginas de vídeo ou não. Podendo ser:

0 - somente uma página. Neste caso quando a tela desaparece em uma das laterais do vídeo a mesma tela reaparece no lado oposto.

1 - duas páginas. Neste caso quando a primeira página desaparece em uma das laterais do vídeo a segunda página aparece no lado oposto.

Para movimentar duas telas a página ativa deve ser ímpar. Para isso use a instrução SETPAGE.

A instrução SET SCROLL só pode ser executada se o Kit 2+ estiver instalado no computador.

Exemplo:

```
10 SCREEN 8,0
20 COPY (0,0)-(255,80) TO (0,211)
30 FOR X=0 TO 255
40 LINE (255,0)-(X,211),X
50 NEXT
60 SETPAGE 1,1:CLS
70 COPY (0,0)-(255,80) TO (0,211)
80 FOR X=0 TO 255
90 LINE (0,0)-(X,211),X
100 NEXT
110 FOR X=0 TO 511
120 SET SCROLL X,0,1,1
130 NEXT
140 FOR X=0 TO 511
150 SETSCROLL X,X/2,1,1
160 NEXT
170 GOTO 110
```

=====

WIDTH X	Instrução
---------	-----------

=====

Determina o número de caracteres por linha de texto.

X é um número inteiro que pode variar de 1 a 80 na SCREEN 0 e de 1 a 32 na SCREEN 1.

Na SCREEN 0 se X for menor ou igual a 40, serão utilizados caracteres grandes para texto. Se X estiver entre 41 e 80 os caracteres serão pequenos.

Os parâmetros originais de fábrica são:

KIT 2.0

KIT 2+

SCREEN 0 - 40 colunas

37 colunas

SCREEN 1 - 29 colunas

29 colunas

Exemplo:

```
10 SCREEN0
20 FOR N=1 TO 80
30 WIDTH N
50 PRINT "Esta tela tem ";N;" colunas."
60 FOR T=0 TO 300:NEXT
70 NEXT
```

=====

CIRCLE, DRAW, LINE, PAINT, POINT, PRESET e PSET

=====

Todas estas instruções passam a trabalhar com os novos valores das cores e coordenadas das telas gráficas.

Para obter os valores máximos das coordenadas ver instrução SCREEN.

Para saber as cores que podem ser utilizadas veja as instruções COLOR e SCREEN.

Nas SCREENs de 5 a 8 os desenhos podem ter uma cor para o traçado e ser preenchidos com outra cor diferente. A instrução PAINT passa a aceitar a seguinte sintaxe:

PAINT C1,C2

onde C1 é a cor que irá preencher a figura e C2 a cor do traçado.

Esta página foi deixada em branco propositalmente.

REL6610 INTERNO

GET DATE	29
SET DATE	30
GET TIME	31
SET TIME	32
SET ADJUST	33
SET BEEP	34
SET PASSWORD	35
SET PROMPT	36
SET SCREEN	37
SET TITLE	38

Esta página foi deixada em branco propositalmente.

```
=====
GET DATE X$ [,A]                                     Instrução
=====
```

Obtém a data armazenada no relógio/calendário interno.

A data no relógio interno é transferida para a variável X\$. O formato dos dados na variável é o seguinte:

DD/MM/AA

onde DD é o dia, MM o mês e AA o ano.

Se for usada a extensão opcional [,A], será transferida a data do alarme.

A data armazenada no relógio/calendário interno é atualizada mesmo com o computador desligado.

Exemplo:

```
GETDATE A$
PRINT A$
```

=====

```
SET DATE X$ [,A]
```

=====

Instrução

Acerta a data no relógio interno.

X\$ é uma expressão alfanumérica com o seguinte formato:

DD/MM/AA

onde:

DD é o dia (entre 1 e 31)

MM é o mês (entre 1 e 12)

AA é o ano.

A data é armazenada no relógio interno e será incrementada mesmo com o computador desligado.

Se for usado o complemento opcional [,A], a data armazenada será a do alarme, e esta data não é incrementada automaticamente.

Exemplo:

```
SETDATE "14/04/61"
```

```
GETDATE A$
```

```
PRINT A$
```

=====

GET TIME X\$ [,A]

=====

Instrução

Fornece a hora armazenada no relógio interno.

A hora no relógio interno é transferida para a variável X\$. O formato dos dados na variável é o seguinte:

HH/MM/SS

onde HH é a hora, MM o minuto e SS o segundo.

Se for usada a extensão opcional [,A], será transferida a hora do alarme.

A hora no relógio interno é atualizada mesmo com o computador desligado.

Exemplo:

```
10 SCREEN 1
20 GETTIME A$
30 LOCATE 10,10
40 PRINT A$
50 GOTO 20
```

=====

SET TIME X\$ [,A]

=====

Instrução

Acerta a hora do relógio interno.

X\$ é uma sequência de caracteres com o seguinte formato:

HH:MM:SS

onde:

HH é a hora (entre 00 e 23)

MM o minuto (entre 00 e 59)

SS o segundo (entre 00 e 59).

Se for usado o complemento opcional [,A], os dados serão considerados como hora do alarme. Neste caso somente as horas e minutos serão armazenados.

Os dados são guardados na memória do relógio interno que é mantido por baterias. Assim, mesmo com o computador desligado, eles serão preservados e atualizados.

Exemplo:

```
10 SET TIME "13:30:20"  
20 GET TIME A$  
30 LOCATE 10,10  
40 PRINT A$  
50 GOTO 20
```

Ajusta a posição da tela para se adequar ao televisor ou monitor.

X deve ser um número inteiro entre -7 e 8. Quando for negativo a tela se deslocará para a esquerda. Quando for positivo a tela se deslocará para a direita.

Y deve ser um número inteiro entre -7 e 8. Quando for negativo a tela se deslocará para cima e quando for positivo a tela se deslocará para baixo.

Os valores de X e Y são armazenados na memória do relógio. Assim, quando o computador for ligado, os parâmetros armazenados serão usados automaticamente.

Exemplo:

```
10 SCREEN1:COLOR 15,4,7
20 PRINT "USE AS TECLAS DO CURSOR"
30 C=STICK(0)
40 X=X+(C=6 OR C=7 OR C=8)-(C=2 OR C=3 OR C=4)
50 Y=Y+(C=8 OR C=1 OR C=2)-(C=4 OR C=5 OR C=6)
60 IF X=-8 THEN X=-7
70 IF X=9 THEN X=8
80 IF Y=-8 THEN Y=-7
90 IF Y=9 THEN Y=8
100 LOCATE 4,4
110 PRINT "X' = ";X,"Y = ";Y
120 SETADJUST (X,Y)
130 GOTO 30
```

Seleciona o tipo e o volume do BEEP.

X é o tipo de BEEP e deve ser um número inteiro entre 1 e 4.

Y é o volume do BEEP que varia entre 1 (volume mínimo) e 4 (volume máximo).

Os valores de X e Y são armazenados na memória do relógio interno. Assim, quando o computador for ligado, os parâmetros armazenados serão usados automaticamente.

O BEEP também pode ser gerado pressionando-se as teclas CONTROL+G.

Exemplo:

```
10 FOR A=1 TO 4
20 FOR B=1 TO 4
30 SETBEEP (A,B)
40 FOR N=1 TO 10
50 BEEP
60 NEXT
70 FOR T=0 TO 500:NEXT
70 NEXT
80 NEXT
```

=====

SET PASSWORD X\$

=====

Instrução

Estabelece uma senha ou palavra de acesso ao computador.

X\$ é uma sequência alfanumérica de até 255 caracteres.

A senha é armazenada na memória do relógio interno, que é mantido por baterias. Assim, mesmo com o computador desligado, a senha continua armazenada.

Ao se ligar o computador será apresentada a palavra "PASSWORD:" esperando que a senha seja digitada.

Caso a senha digitada não coincida com a armazenada, o computador continuará esperando a senha correta.

Para eliminar a senha use a instrução SET TITLE"".

Só se pode utilizar uma das instruções "SETPASSWORD", "SETPROMPT" ou "SETTITLE" de cada vez. A última instrução executada será a instrução armazenada, pois ambas utilizam a mesma área na memória do relógio interno.

Exemplo:

SET PASSWORD"TESTE"

desligue o computador e ligue-o novamente.

digite TESTE seguido da tecla RETURN

=====	=====
SET PROMPT X\$	Instrução
=====	=====

Estabelece um PROMPT ou palavra de aviso de espera de um comando.

X\$ é uma sequência alfanumérica de até 6 caracteres.

O PROMPT é armazenado na memória do relógio interno, que é mantido por baterias. Assim, mesmo com o computador desligado, ele será preservado.

O PROMPT normal do computador é "Ok". Quando este PROMPT é selecionado, a área na memória de relógio fica livre para ser usada pelas instruções: SETPASSWORD ou SETTITLE.

Só se pode utilizar uma das instruções "SETPASSWORD", "SETPROMPT" ou "SETTITLE" de cada vez. A última instrução executada será a instrução armazenada, pois ambas utilizam a mesma área na memória do relógio interno.

Exemplo:

SET PROMPT "TESTE"

Transfere os parâmetros da tela e do modo de trabalho do computador para a memória do relógio.

Os parâmetros armazenados são:

Tipo de SCREEN de texto: 0 ou 1.

Número de colunas (WIDTH): 1 a 80.

Número da cor do caractere: 0 a 15.

Número da cor do fundo do caractere: 0 a 15.

Número da cor da borda da tela: 0 a 15;

Exibição das teclas de funções: ligada ou desligada.

Click do teclado: ligado ou desligado.

Tipo de impressora: MSX ou não MSX

Velocidade dos dados para o gravador cassete: 1200 ou 2400 bauds.

Modo de exibição da tela: 1 a 4

Estes dados são armazenados na memória do relógio interno, que é mantido por baterias. Assim, mesmo com o computador desligado, eles serão preservados.

Exemplo:

Para começar com 80 colunas automaticamente:

SCREEN0

WIDTH80

SETSCREEN

Desligue o computador e ligue-o novamente.

Armazena um nome que será mostrado na apresentação quando o computador é ligado.

X\$ é uma sequência alfanumérica com até 6 caracteres.

Se X\$ tiver exatamente 6 caracteres, o nome aparecerá na apresentação, e o computador ficará esperando que qualquer tecla seja pressionada. Se tiver até 5 caracteres o nome ficará na tela durante por alguns segundos e então continuará o processamento.

Y, que varia de 1 a 4, seleciona qual combinação de cores será usada na apresentação quando o computador for ligado. Podendo ser:

- 1 - Azul / Preta
- 2 - Verde / Azul clara
- 3 - Vermelha / Magenta
- 4 - Amarela / Vermelha clara

Estes dados são armazenados na memória do relógio interno, que é mantido por baterias. Assim, mesmo com o computador desligado, eles serão preservados.

Só se pode utilizar uma das instruções "SETPASSWORD", "SETPROMPT" ou "SETTITLE" de cada vez. A última instrução executada será a instrução armazenada, pois ambas utilizam a mesma área na memória do relógio interno.

SET TITLE "" é usado para apagar os dados armazenados por SET PASSWORD.

Exemplo:

SET TITLE "TESTE ",4

Desligue o computador e ligue-o novamente.

RAM-DISK

CALL MEMINI	41
CALL MFILES	42
CALL MKILL	43
CALL MNAME	44

Esta página foi deixada em branco propositalmente.

CALL MEMINI [(X)]	Instrução
-------------------	-----------

Prepara uma área da memória RAM para trabalhar como RAMDISK (disco em memória).

X é o tamanho do disco em memória em bytes e tem que ser um número inteiro entre 1023 e 32767, ou 0 (zero).

CALL MEMINI (0) cancela o modo RAMDISK.

Quando se executa a instrução CALL MEMINI todo conteúdo do disco em memória será limpo.

No lugar da palavra CALL pode-se utilizar o sinal de sublinhado (_). Por exemplo: MEMINI

A instrução CALL MEMINI deve ser executada antes das instruções: CALL MFILES, CALL MNAME ou CALL MKILL.

As seguintes instruções podem ser utilizadas após a instrução CALL MEMINI para se trabalhar com arquivos em RAMDISK: SAVE, LOAD, RUN, MERGE, OPEN, CLOSE, PRINT#, PRINT#USING, INPUT#, LINEINPUT#, EOF, LOC, LOF.

A RAMDISK deve ser tratada como se fosse um disquete normal. O único detalhe é que o nome do dispositivo deve ser MEM:.

Exemplo:

```

MEMINI
10 PRINT "TESTE ";
20 GOTO 10

SAVE "MEM:TESTE.BAS"

NEW

MFILES

LIST

LOAD "MEM:TESTE.BAS"

LIST

RUN
  
```

CALL MFILES	Instrução
-------------	-----------

Lista na tela os nomes dos arquivos armazenados na RAMDISK e quantos bytes ainda estão livres para uso.

No lugar da palavra CALL pode-se utilizar o sinal de sublinhado (_). Por exemplo: _MFILES

Este comando só será aceito depois da execução da instrução CALL MEMINI. Caso a instrução CALL MEMINI não tenha sido executada ou a RAMDISK tenha sido eliminada pela instrução CALL MEMINI (0), será emitida a mensagem de erro "disk offline".

Exemplo:

```
_MEMINI
10 PRINT "TESTE ";
20 GOTO 10

SAVE "MEM:TESTE.BAS"

NEW

_MFILES

LIST

LOAD "MEM:TESTE.BAS"

LIST

RUN
```

=====

```
CALL MKILL ("nome do arquivo")
```

=====

Instrução

Apaga um arquivo armazenado na RANDISK.

No lugar da palavra CALL pode-se utilizar o sinal de sublinhado (_). Por exemplo: _MFILES

Este comando só será aceito depois da execução da instrução CALL MEMINI. Caso a instrução CALL MEMINI não tenha sido executada ou a RANDISK tenha sido eliminada pela instrução CALL MEMINI (0), será emitida a mensagem de erro "disk offline".

Exemplo:

```
_MEMINI
```

```
10 PRINT "TESTE ";  
20 GOTO 10
```

```
SAVE "MEM:TESTE.BAS"
```

```
_MFILES
```

```
_MKILL "TESTE.BAS"
```

```
_MFILES
```

```
=====
CALL MNAME ("nome do arquivo 1" as "nome do arquivo 2")           Instrução
=====
```

Troca o nome de um arquivo gravado na RAMDISK.

O "nome do arquivo 1" é aquele já gravado na RAMDISK e o "nome do arquivo 2" é o que substituirá o antigo.

No lugar da palavra CALL pode-se utilizar o sinal de sublinhado (_). Por exemplo: _MNAME

Este comando só será aceito depois da execução da instrução CALL MEMINI. Caso a instrução CALL MEMINI não tenha sido executada ou a RAMDISK tenha sido eliminada pela instrução CALL MEMINI (0), será emitida a mensagem de erro "disk offline".

Exemplo:

```
_MEMINI
10 PRINT "TESTE ";
20 GOTO 10
SAVE "MEM:TESTE.BAS"
_MFILES
_MNAME ("TESTE.BAS" AS "PROG.BAS")
_MFILES
```

TURBO-BASIC -----

CALL AC	47
CALL RUN	48
CALL TURBO ON/OFF	49
'#C	50
'#N	51
'#I	52

Esta página foi deixada em branco propositalmente.

=====

CALL AC (Auto-Charge)

Transfere o programa TURBO-BASIC para a memória RAM e prepara o computador para aceitar os novos comandos.

Esta instrução só é necessária no computador com o KIT 2+. No KIT 2.0 o programa já está instalado.

Antes de se utilizar qualquer das novas instruções do TURBO-BASIC a instrução CALL AC deve ser executada.

No lugar da palavra CALL pode-se utilizar a sinal de sublinhado (_). Por exemplo: AC

Executa um programa em BASIC com a velocidade do TURBO-BASIC.

A velocidade de execução do programa em BASIC varia de acordo com a complexidade das instruções encontradas.

No KIT 2+ esta instrução só funciona depois da execução da instrução CALL AC (AUTO-CHARGE).

Para instruções FOR...NEXT, o NEXT deve ser acompanhado pela variável a qual se refere. Por exemplo:

```
10 FOR X=0 to 100
```

```
90 NEXT X
```

Nunca deixe rotinas de execução contínua como:

```
100 GOTO 100
```

substitua pela seguinte sequência:

```
100 IF INKEY$="" THEN 100
```

Alguns comandos do BASIC normal do computador não são executados pelo TURBO-BASIC. A maioria deles se refere ao tratamento de arquivos e ao acesso a disquetes. Na lista a seguir estão os comandos e instruções mencionados:

AUTO, BASE, BLOAD, BSAVE, CALL, CDBL, CINT, CLEAR, CLOAD, CLOAD?, CLOSE, CONT, CSAVE, CSNG, CVD, CVI, CVS, DEFFN, DELETE, DRAW, DSKF, EOF, EQV, ERASE, ERL, ERR, ERROR, FIELD, FILES, FPOS, FRE, GET, IMP, INPUT\$, KEYLIST, KILL, LFILES, LINE INPUT\$, LIST, LLIST, LOAD, LOC, LOF, LPRINT USING, LSET, MAXFILES, MERGE, MKD\$, MKI\$, MKS\$, MOTOR, NEW, NAME, ON ERROR GOTO, OPEN, PLAY, PRINT\$, PRINT#USING, PRINT USING, RENUM, RESUME, RSET, SAVE, SPC, TAB, TRON, TROFF, WIDTH.

Exemplos:

```
10 SCREEN 8
20 FOR N=0 TO 500
30 X=RND(1)*256
40 Y=RND(1)*212
50 C=RND(1)*256
60 LINE(X,Y)-STEP(20,20),C,BF
70 NEXT N
80 IF INKEY$="" THEN 80
```

```
AC
RUN
```

```
10 TIME=0
20 FOR N=0 TO 10000
30 NEXT N
40 PRINT TIME
```

Execute uma vez com o RUN do BASIC normal e outra com CALL RUN

Ativa ou desativa o TURBO-BASIC durante a execução de um programa.

CALL TURBO ON, ativa a execução em TURBO-BASIC.

CALL TURBO OFF, desativa o TURBO-BASIC.

No KIT 2+ a instrução CALL AC deve ser executada antes de se utilizar estas instruções.

Exemplo 1:

```
10 SCREEN 5
20 FOR X=0 TO 255
30 LINE(0,0)-(X,211),X
40 NEXT X
50 TURBO ON
60 CLS
70 FOR X=0 TO 255
80 LINE (0,0)-(X,211),X
90 NEXT X
100 TURBO OFF
110 IF INKEY$="" THEN 110
```

Exemplo 2:

```
100 DEFINT A-Z
110 TIME=0
120 FOR N=1 TO 1000
130 A=A+1:B=B+1:C=C+1
140 NEXT N
150 PRINT "TEMPO SEM TURBO =";TIME
160 TURBOON
180 TIME=0
190 FOR N=1 TO 1000
200 A=A+1:B=B+1:C=C+1
210 NEXT N
220 PRINT "TEMPO COM TURBO =";TIME
```

```
=====
'#C-                                     Turbo-BASIC
=====
```

No BASIC normal as telas gráficas trabalham com coordenadas no eixo Y somente com valores entre 0 e 211. Com este comando em TURBO-BASIC as instruções gráficas passam a aceitar valores entre 0 e 255, permitindo que se trabalhe com a tela toda, inclusive a faixa que não é normalmente mostrada (linhas de 212 a 255).

'#C- liga a tela plena.

'#C+ desliga a tela plena.

Exemplo:

```
10 SCREEN 8
20 '#C-
30 FOR X=0 TO 255
40 LINE (0,0)-(X,255),X
50 NEXT X
60 IF INKEY$="" THEN 60
70 CLS
80 '#C+
90 FOR X=0 TO 255
100 LINE (0,0)-(X,255),X
110 NEXT X
120 IF INKEY$="" THEN 120
```

```
=====
'#N                                         Turbo-BASIC
=====
```

Permite que a instrução FOR...NEXT trabalhe com números inteiros negativos. No BASIC normal se um número inteiro passar de &H7FFF, o programa é interrompido com erro de sobrecarga (Overflow).

'#N+ habilita a função até &h7FFF.

'#N- trabalha com inteiros negativos.

Exemplo:

```
10 '#N+
20 FOR IZ=&H7FF0 TO &H7FFF
30 PRINT IZ
40 NEXT IZ
```

Execute o programa com RUN, CALL RUN e depois troque a linha 10 por:

```
10 '#N-
```

e execute com CALL RUN

```
=====
'01                                     Turbo-BASIC
=====
```

Permite que as informações, após este código, sejam executadas como uma rotina em assembler e que, variáveis do BASIC, sejam usadas pela rotina. Também admite que se use linhas do BASIC como subrotinas do programa em assembler.

Para utilizar variáveis do BASIC no programa em assembler basta colocar o nome da variável na posição desejada.

Uma linha do BASIC pode ser acessada no programa em assembler usando o caractere "@" (arroba) antes do número da linha. Cuidado quando usar a instrução RENUM, já que ela não altera a linha no bloco em assembler.

Exemplo:

```
100 AC
110 TURBOON
120 SCREEN5:YZ=0:YZ=0
130 PRESET(0,0)
140 FORCZ=0T0255
150 GOSUB190
160 NEXTCZ
170 IF INKEY$="" THEN 170
180 END
190 @I&H2A,CZ,&H29,&H29,&H29,&H11,&HBF,&H1B,&H19
200 @I&H06,&H0B,&H3A,&HB7,&HFC,&H32,XZ,&H3A,&HB9,&HFC,&H32,YZ
210 @I&H7E,&H23,&HC5,&H06,&H0B
220 @I&H17,&HFS,&HC5,&HE5,&HDC,@260,&HE1,&HC1,&H3A,XZ,&H3C,&H32,XZ,&HF1
230 @I&H10,&HED,&H3A,XZ,&HD6,&H0B,&H32,XZ,&H3A,YZ,&H3C,&H32,YZ,&HC1
240 @I&H10,&HD6,&H3A,XZ,&HC6,&H0B,&H32,&HB7,&HFC,&H3A,YZ,&H3B,2,&HD6
250 @I&H0B,&H32,&HB9,&HFC,&HC9
260 PSET(XZ,YZ):RETURN
```

Miscelânea E VDP

PAD	55
PDL	57
BASE	58
VDP	59

Esta página foi deixada em branco propositalmente.

Lê dados e coordenadas dos TOUCHPAD, LIGHTPEN, MOUSE e TRACKBALL que estejam ligados às entradas de JOYSTICK.

X deve ser um número inteiro entre 0 e 19. E tem o seguinte significado:

Se X estiver entre 0 e 3 presume-se que há uma TOUCHPAD ligada na tomada do JOYSTICK 1.

Se X estiver entre 4 e 7 presume-se que há uma TOUCHPAD ligada na tomada do JOYSTICK 2.

Se X estiver entre 8 e 11 presume-se que tenha uma LIGHTPEN ligada na tomada do JOYSTICK.

Se X estiver entre 12 e 15 presume-se que tenha um MOUSE ou TRACKBALL ligado na tomada do JOYSTICK 1.

Se X estiver entre 16 e 19 presume-se que tenha um MOUSE ou TRACKBALL ligado na tomada do JOYSTICK 2.

Nos casos acima se X for igual a:

0 ou 4 - retorna o estado da mesa da TOUCHPAD: 0 se não foi tocada ou -1 se foi tocada.

1 ou 5 - retorna a coordenada X do ponto onde a mesa foi tocada.

2 ou 6 - retorna a coordenada Y do ponto onde a mesa foi tocada.

3 ou 7 - retorna o estado da tecla da TOUCHPAD. O resultado será:
0 (tecla não pressionada) ou -1 (tecla pressionada).

8 - retorna o estado da LIGHTPEN: 0 se as coordenadas não estão prontas ou -1 se as coordenadas estão prontas.

9 - retorna a coordenada X da posição da LIGHTPEN.

10 - retorna a coordenada Y da posição da LIGHTPEN

11 - retorna o estado da tecla da LIGHTPEN: 0 (tecla não pressionada) ou -1 (tecla pressionada)

12 ou 16 - retorna o estado do MOUSE ou TRACKBALL: é sempre -1

13 ou 17 - retorna a coordenada X do MOUSE ou TRACKBALL.

14 ou 18 - retorna a coordenada Y do MOUSE ou TRACKBALL.

15 ou 19 - retorna sempre 0.

A leitura das teclas do MOUSE ou TRACKBALL é feita com a instrução STRIG.

Antes de ler as coordenadas do MOUSE ou TRACKBALL é obrigatória a execução da função PAD (12) ou PAD (16), dependendo da tomada onde o MOUSE ou TRACKBALL esteja ligado.

Antes de ler as coordenadas da LIGHTPEN verifique se a função PAD(8) retorna com o valor -1.

A função PAD só terá validade se algum dos acessórios (TOUCHPAD, LIGHTPEN, MOUSE ou TRACKBALL) estiver ligado ao computador.

- TOUCHPAD é uma pequena prancheta usada para desenhos com sensores em toda sua extensão que, ao serem tocados, transmitem as coordenadas X,Y da posição ao computador. Para acompanhar os traços do desenho é usada uma caneta com uma chave auxiliar. Uma segunda chave é acionada quando a caneta toca a superfície da prancheta. A função das chaves é similar as existentes nos JOYSTICKS.

- LIGHTPEN é uma caneta com um sensor de luz na ponta que, quando encostada na tela do monitor ou televisor, transmite sua posição em coordenadas X,Y ao computador. Possui duas chaves auxiliares. Uma é acionada quando a tecla existente no corpo da caneta é pressionada. A outra é um sensor que é acionado ao cruzar com um raio luminoso na tela.

- MOUSE é um controlador manual com uma esfera na sua parte inferior e duas chaves auxiliares como nos JOYSTICKS. A movimentação do MOUSE sobre qualquer superfície faz com que a esfera gire duas rodas perfuradas, uma para o eixo X e outra para o eixo Y, que geram uma sequência de pulsos transformados em coordenadas pelo computador.

- TRACKBALL é semelhante ao MOUSE, com exceção da esfera que fica na parte superior. Ao invés do controlador ser movimentado sobre uma superfície, a esfera é girada pela mão do operador para o lado desejado.

PDL (X)	Função
---------	--------

Retorna o estado de PADDLES ligados ao computador.

Até 12 (doze) PADDLES podem ser ligados às entradas de JOYSTICK (6 na entrada A e 6 na B).

Se X for 1, 3, 5, 7, 9 ou 11 presume-se que os PADDLES estejam ligados na entrada do JOYSTICK-1 (A). Se for 2, 4, 6, 8, 10 ou 12 presume-se que estejam ligados na entrada do JOYSTICK-2 (B).

Esta função retorna um valor entre 0 e 255.

PADDLES são controles rotativos utilizados em jogos ou programas que têm movimentos em duas direções (direita e esquerda por exemplo).

A função PDL(X) só será válida se estiverem um ou mais PADDLES ligados ao computador.

Exemplo:

```
10 PRINT PDL(1)
20 GOTO10
```

BASE (X) ou BASE (X) = Y	Função ou Instrução
--------------------------	---------------------

Mostra o endereço do primeiro byte das tabelas do processador de vídeo (VDP), ou escreve um endereço inicial nos registradores das tabelas do VDP.

(X) deve ser sempre uma expressão de números inteiros entre 0 e 44 que, dependendo da SCREEN utilizada, tem o seguinte significado:

SCREEN	0	1	2	3	4	5	6	7	8
TABELA NOMES	0	5	10	15	20	25	30	35	40
TABELA CORES	-	6	11	16	21	26	31	36	41
TABELA CARACTERES	2	7	12	17	22	27	32	37	42
TABELA ATRIBUTO DOS SPRITES	-	8	13	18	23	28	33	38	43
TABELA DE FORMATO DOS SPRITES	-	9	14	19	24	29	34	39	44

Quando se lê um endereço com a função BASE (X), o valor de X pode variar entre 0 e 44.

Para se escrever um valor nos registradores do VDP com a função BASE (X) = Y, o valor de X só pode variar entre 0 e 19, ou seja, somente as SCREENs 0, 1, 2 e 3 são afetadas. Quando se troca um endereço na SCREEN 3 os endereços das SCREENs 4, 5, 6, 7 e 8 também são alterados.

Os dados da SCREEN 8 valem também para as SCREENs 10, 11 e 12 no caso do KIT 2+.

Exemplo:

```

100 'Enche duas telas
110
120 SCREEN0:WIDTH40:KEYOFF
130
140 FORA=0TO959:PRINT"-";NEXT
150 BASE(0)=1024:SCREEN0
160 FORA=0TO959:PRINT"+";NEXT
170
180 'Alterna as páginas de texto
190
200 BASE(0)=0
210 FORT=0TO100:NEXT
220 BASE(0)=1024
230 FORT=0TO100:NEXT
240 GOTO200

```

Outro exemplo:

```
PRINT BASE(0)
```

OBS: Use esta função ou instrução somente quando estiver bem familiarizado com o modo de trabalho do processador de vídeo (VDP).

VDP (X) ou VDP (X) = Y	Função ou Instrução
------------------------	---------------------

Verifica ou altera o conteúdo dos registradores do Processador de Vídeo (VDP).

X é um número inteiro e pode variar entre:

- 9 e -1 - registradores de STATUS 1 a 9.
- 0 e 8 - mesmos do MSX1.
- 9 e 24 - registradores de 8 a 23.
- 33 e 47 - registradores de 32 a 46.

Esta instrução só deve ser utilizada quando se estiver bem familiarizado com o funcionamento do Processador de Vídeo (VDP).

Exemplo:

```
10 FOR R=-9 TO 8
20 A=VDP(R)
30 PRINT "Registro: "R;" = ";RIGHT$("00000000"+BIN$(A),8)
40 NEXT
```

Esta página foi deixada em branco propositalmente.



ACVS Eletrônica Ltda.

Av. Paulista, 2001 - 9º andar - Conj. 912
CEP. 01311 - SÃO PAULO - SP - T. 289-7694
R. Sandoval Ferreira Cabral, 369
CEP. 05135 - SÃO PAULO - SP - T. 834-5349
