

# MSX Article

**MARMSX**

*Graphos III*

## Resumo

O objetivo desse artigo é mostrar o funcionamento do excelente editor gráfico Graphos III, versão 1.2, criado no Brasil nos anos 80 pelo designer gráfico Renato Degiovani [1].

## 1- Introdução

Muitos MSXzeiros que dispunham do micro nos anos 80 e 90 certamente já usaram o editor gráfico Graphos III para criar imagens no formato de screen 2. Ele possuía diversos recursos interessantes para a época, como editor de alfabetos, zoom, shapes e desenho de formas geométricas.

O programa foi criado em 1984 para a linha de micros TRS 80, por Renato Degiovani. No final de 1985, o autor necessitava de um bom programa para auxiliar na criação de imagens para o adventure Angra-I. Nessa época, surgiram micros mais modernos e com melhores recursos gráficos, como o MSX. Então, ele desenvolveu durante 3 meses uma nova versão do software, aproveitando boa parte do código anterior. Em 1987, o programa foi finalmente comercializado. Adaptado de [2].

Em 1992, as revistas Micro Sistemas de 111 a 120 publicaram o código fonte do programa, só que convertido para PC.

## 2- O programa

O programa Graphos III é um poderoso editor gráfico para MSX, possuindo diversos recursos que auxiliam na criação de desenhos. Ele incorpora um editor de fontes, no qual é possível criar ou modificar fontes de caracteres no formato 8x8. Outro recurso interessante são os shapes, que são uma espécie de cliparts do MSX e eram utilizados para compor cenários diversos, principalmente de jogos.

Ao carregar o programa, surge a tela principal, conforme mostra a figura 2.1.



Figura 2.1. Tela principal.

O menu principal é composto das seguintes opções:

- Display tela – escolhe um dos quatro displays para a imagem.
- Edita tela – editor principal de imagens no formato screen 2.
- Arquivo tela – salva a imagem em disco ou fita K7.
- Recupera tela – carrega a imagem do disco ou fita K7.
- Edita alfabeto – editor de fontes do MSX.
- Arquivo alfabeto - salva a fonte em disco ou fita K7.
- Recupera alfabeto - carrega a fonte do disco ou fita K7.
- Cria shapes – cria shapes.
- Arquivo shapes - salva os shapes em disco ou fita K7.
- Recupera shapes - carrega os shapes do disco ou fita K7.
- Diretório – lista arquivos de determinados tipos: tela, layout, shapes etc.
- Versão do sistema – informa a versão do sistema.
- Basic – retorna ao Basic (exceto no MSX 2, que trava o micro).

Para entender melhor o funcionamento do programa, serão apresentados alguns conceitos utilizados no Graphos III.

### *Imagem*

A imagem do Graphos III é no padrão da screen 2 do MSX. Ela é armazenada no formato binário e carregada a partir do seguinte comando em Basic:

```
BLOAD "TELA.GRP",R
```

A imagem incorpora o display escolhido durante a edição.

### *Display*

É um algoritmo que tem como função realizar a apresentação da imagem na tela utilizando algum tipo de animação. São quatro opções:

Display	Descrição
ROT:A	Instantâneo. Carrega a imagem diretamente na tela.
ROT:B	Persiana. Divide a tela em 24 linhas horizontais, com 8 pixels cada, carregando a imagem de cima para baixo em cada uma dessas linhas.
ROT:C	Esquerda. Carrega a imagem da esquerda para a direita.
ROT:D	Chuveiro. Carrega a imagem em grupos de 8x1 pixels aleatoriamente.

Ao selecionar um dos quatro displays, ele já é automaticamente incorporado ao arquivo que será salvo. O display *default* é o ROT:A.

### *Alfabeto*

São padrões de caracteres de 8x8 pixels utilizados em todos os modos de tela do MSX. É possível editar e redefinir o formato de cada um dos 256 caracteres.

O editor principal de imagens irá utilizar o alfabeto que foi editado ou carregado em memória.

### *Shapes*

São coleções de desenhos destinados a compor uma tela, como os cliparts. Eles podem ter qualquer tamanho múltiplo de 8, variando de 8x8 a 240x176.

Os shapes são criados a partir de imagens desenhadas no editor principal e podem ser monocromáticos ou coloridos.

## **3- Editor principal**

Ao selecionar a opção “Edita tela” no menu principal, o editor é aberto, conforme mostra a figura 3.1.

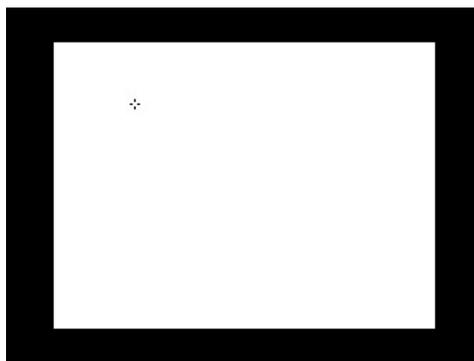


Figura 3.1. Editor de imagens.

A tela principal é branca com borda preta, e possui somente o cursor que servirá de base para desenhar. O cursor pode ser movimentado através das setas direcionais. Ao teclar “Shift” juntamente com os direcionais, o cursor movimenta-se mais rápido.

O editor possui 5 menus, que podem ser acessados através das teclas funcionais F1, F2, F3, F4 e F5. Outras teclas também ajudam na edição da imagem.

Lista de teclas do editor:

- Setas direcionais – movimentam cursor.
- Shift + direcional – aumenta velocidade do cursor.
- Control + direcional – diminui a velocidade do cursor.
- Select – abre menu de cores: Ink (cor de frente) e Paper (cor de fundo).
- LGRA – muda a cor do cursor.

- Stop – muda a cor da borda da tela.
- Esc – cancela a edição atual.
- Enter – valida edição atual / sai do editor, salvando tela no buffer.
- Home – recupera a tela do buffer (undo).
- Tab – move o cursor horizontalmente para o primeiro pixel do bloco de 8x8.
- F1 – abre o menu de desenho.
- F2 – abre o menu de texto.
- F3 – abre o menu de tela.
- F4 – abre o menu de ajuste.
- F5 – abre o menu de miscelânea.

### 3.1. Menu de desenho

O menu de desenho é acessado através da tecla F1, conforme mostra a figura 2.1.



Figura 3.2. Menu de desenho.

Opções:

- Traço – traço à mão-livre.
- Bloco – define o tamanho de um bloco e desenha à mão livre.
- Linha – desenha uma linha ou polígono.
- Retângulo – desenha um retângulo.
- Raio – desenha raios, a partir de uma origem em comum.
- Círculo – desenha um círculo ou círculos concêntricos.
- Pintura – muda a cor de fundo do desenho, como o traço à mão-livre.
- Spray – desenha padrão de spray à mão-livre.
- Fill – preenche uma área a partir da posição do cursor.

Todas as opções, exceto a pintura, utilizam a cor de frente (Ink). Já a pintura utiliza a cor de fundo (Paper). Para abrir o menu de cores, pressione a tecla “Select”. A figura 3.3 mostra o menu de cores do Graphos III.

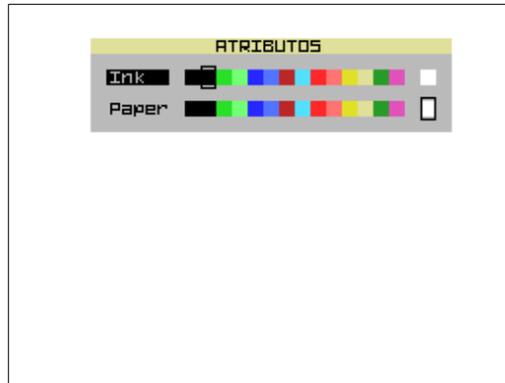
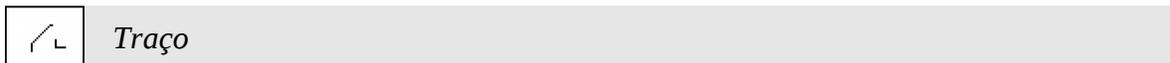


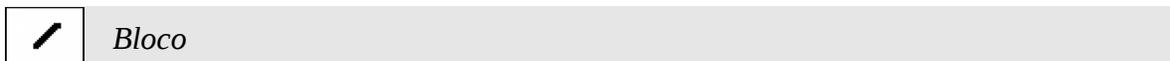
Figura 3.3. Seleção de cores.

Utilize as setas do cursor para navegar e escolher as cores. A primeira linha escolhe a cor de frente (Ink), enquanto que a segunda linha escolhe a cor de fundo (Paper). As teclas para cima e para baixo escolhem entre Ink e Paper, enquanto que as teclas esquerda e direita escolhem a cor. Um retângulo preto em torno da cor indica qual é a cor selecionada. Tecle “Esc” para cancelar as modificações ou “Enter” para validá-las.

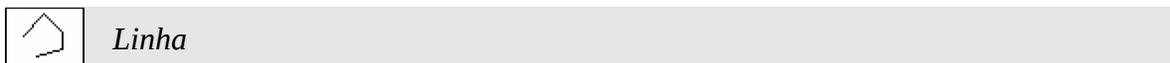
Na figura 3.3, observamos que o retângulo no Ink está sobre a cor preta (índice 1), enquanto que o retângulo no Paper está sobre a cor branca (índice 15).



Para desenhar à mão livre, utilize as teclas direcionais juntamente com a barra de espaços. A linha é desenhada, somente quando a barra está sendo pressionada. Pressione a tecla “Esc” caso deseje cancelar o desenho dos traços.



O primeiro passo é definir o tamanho do bloco. Utilize os cursores para definir o tamanho e tecla “Enter” para validar o tamanho do bloco. Ele pode ter de 1x1 até 8x8 pixels de tamanho. O passo seguinte é desenhar na tela exatamente como foi feito no traço.



Para desenhar uma linha ou polígono, mova o cursor para o local do primeiro ponto e tecla “Espaço”. Uma “âncora” é desenhada na tela indicando o local de origem da linha. Mova o cursor até a posição de destino desejada (figura 3.4) e tecla “Espaço” novamente para traçar a linha. Tecle “Enter” para terminar a linha ou siga desenhando o polígono, criando uma nova linha a partir da última posição (figura 3.5).

A tecla “Esc” cancela o desenho da linha ou polígono.

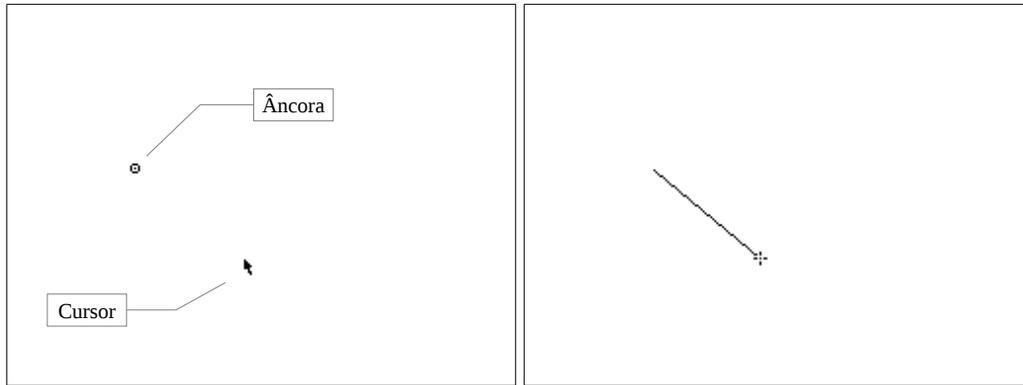


Figura 3.4. Desenhando uma linha.

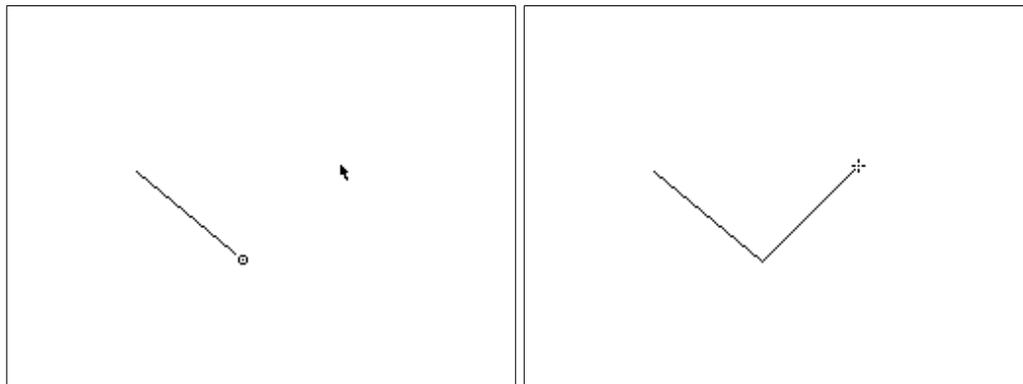


Figura 3.5. Desenhando um polígono.

 *Retângulo*

Para criar um retângulo não preenchido ou vários na tela a partir de uma origem em comum, mova o cursor até a posição desejada na tela e tecle “Espaço” para definir a posição de origem, que será marcada com uma “âncora”. Mova o cursor até a posição de destino e tecle “Espaço” novamente para desenhar o retângulo. É possível continuar a desenhar retângulos, sempre baseado na origem em comum (âncora). A tecla “Enter” termina a edição e a tecla “Esc” cancela a operação de desenho.

 *Raio*

Para desenhar linhas na tela a partir de uma origem em comum, mova o cursor até a posição desejada na tela e tecle “Espaço” para definir a posição da origem em comum. Mova o cursor até uma nova posição, e tecle “Espaço” novamente para traçar uma linha. Mova o cursor para uma nova posição e trace uma nova linha, quantas vezes quiser. A tecla “Enter” termina a edição, enquanto que a tecla “Esc” anula a operação de desenho de raios.

As linhas terão a mesma origem, que foi o primeiro ponto marcado (âncora).



### *Círculo*

Para desenhar um círculo ou vários círculos concêntricos na tela, mova o cursor até a posição desejada na tela e tecle “Espaço” para definir a posição da origem em comum dos círculos. Mova o cursor até uma nova posição, e tecle “Espaço” novamente para desenhar o círculo. Continue a edição, desenhando quantos círculos concêntricos quiser. A tecla “Enter” termina a edição e a tecla “Esc” cancela a operação de desenho.



### *Pintura*

Pinta a tela sob o cursor com a cor de fundo (Paper), semelhante ao que o traço faz. No entanto, preenche um grupo de 8x1 pixels de uma vez, ou até mesmo 16x1 pixels, quando o cursor estiver sobre dois grupos de 8x1 pixels.



### *Spray*

Desenha exatamente como o traço faz, só que com o padrão de spray.



### *Fill*

Para preencher uma área na tela, delimitada pela mesma cor da tinta de preenchimento, posicione o cursor sobre a área que deseja pintar e tecle “Espaço” para preenchê-la.

## 3.2. Menu de texto

Escreve na tela, utilizando o alfabeto editado ou carregado.

Há seis opções de escrita para o mesmo alfabeto:

- Normal – não há modificações.
- Itálico – letras no formato *itálico*.
- Bold – letras no formato **bold**.
- Duplo – dobra a resolução vertical de cada caractere (8x16).
- Duplo bold - dobra a resolução vertical de cada caractere, utilizando o bold.
- Largo – dobra a resolução horizontal de cada caractere (16x8).

Para escrever, selecione um tipo de escrita e tecle um caractere (letra, número ou símbolo). O cursor em forma de colchetes “[ ]” indica a posição onde a letra será escrita na tela. Não há opção de apagar a letra com o “Backspace”. Para corrigir, utilize a tecla “Home”, que cancela toda a escrita.

### 3.3. Menu de tela

O menu de tela realiza operações sobre toda a tela.

Opções:

- Salva tela – salva estado atual da tela, impedindo o posterior cancelamento.
- Inverte vídeo – inverte as cores do vídeo (Ink e Paper).
- Inverte atributo – inverte as cores do vídeo (Ink e Paper).
- Retira vídeo – remove partes da tela baseadas no Ink.
- Retira atributo – remove partes da tela baseadas no Paper.
- Repõe vídeo – cancela operação de retira vídeo.
- Repõe atributo – cancela operação de retira atributo.
- Limpa a tela – limpa a tela com a cor de fundo definida.
- Imprime tela – imprime a imagem em uma impressora.

Obs: as opções de inversão de tela parecem sempre utilizar a cor branca como base.

### 3.4. Menu de ajuste

Ajusta a tela. Afeta somente os desenhos feitos com o Ink.

Opções:

- Scroll – move a tela em qualquer direção, pixel a pixel. As bordas são apagadas.
- Scroll 8x8 - move a tela em qualquer direção, de 8 em 8 pixels.
- Rotação - move a tela em qualquer direção, pixel a pixel. As bordas dão a volta.
- Rotação 8x8 - move a tela em qualquer direção, de 8 em 8 pixels.

Utilize as setas direcionais para mover a tela. Tecle “Enter” para terminar.

### 3.5. Menu de miscelânea

O último menu possui algumas funções:

- Zoom – permite ampliar um trecho da tela a nível de pixels.
- Shape – permite selecionar um desenho contido na galeria dos shapes.
- Corte – permite o corte de parte da tela.
- Grid – gera um grid de blocos de 8x8 na tela, com as cores branca e cinza. A cor cinza é de fundo (Paper), não alterando o desenho feito com o Ink.

Obs: a ferramenta grid é ideal para demarcar as regiões de 8x8 pixels características da screen 2.

## Zoom

Ao selecionar a opção de zoom, o cursor modifica para o formato de lupa (figura 3.6). Ele deverá ser movimentado para o local da tela que se deseja ampliar, teclando-se “Espaço” para abrir um editor com tela ampliada (figura 3.6).

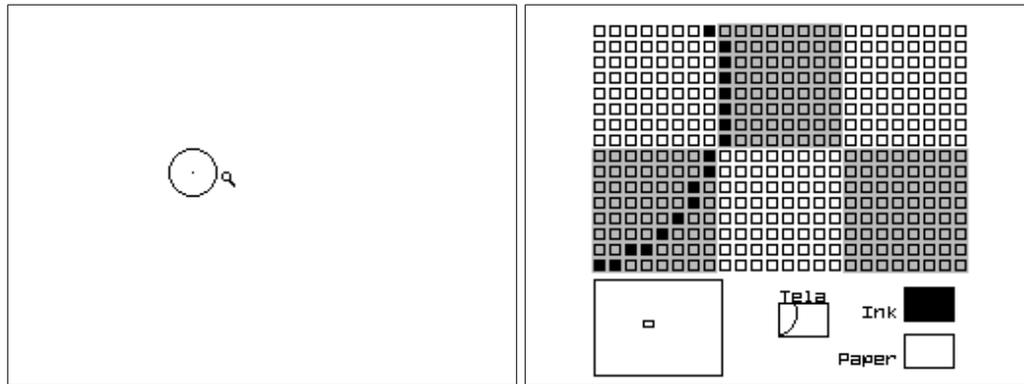


Figura 3.6. Ferramenta de zoom.

O editor é composto da área em zoom (acima), do local da tela que está sendo ampliado (embaixo, à esquerda), do desenho da tela na seleção atual (centro) e das cores do Ink e Paper (direita).

Há dois modos no editor de zoom: a movimentação na tela principal e a edição de pixels. Os modos são trocados através da tecla “Enter”. O primeiro modo permite movimentar a área de zoom pela tela principal, através das setas. Este modo é iniciado quando o editor de zoom é aberto. O segundo modo transfere o controle para a janela superior, permitindo alterar a região pixel a pixel. As setas movimentam o cursor e a tecla “Espaço” acende ou apaga um pixel. Há também três letras no canto superior direito da tela, quando este modo está ativo: A, S e R. O “A” é ativado pela tecla “Home”, e permite acender ou apagar um pixel. O “S” é ativado pela tecla “Insert” e permite somente acender os pixels. O “R” é ativado pela tecla “Delete” e permite somente apagar os pixels.

A tecla “Esc” permite cancelar as modificações durante a edição do zoom. E é essa mesma tecla que é utilizada para sair desse editor e voltar para tela de desenho. Dessa forma, tecla “Enter” para validar as modificações no zoom antes de sair do editor.

A tecla “Select” abre o diálogo de escolha de cores, quando estiver no modo de edição do zoom. Desta vez, as setas esquerda e direita selecionam entre Ink e Paper, enquanto que as setas cima e baixo selecionam a cor. Ao modificar a cor do fundo (Paper), qualquer alteração de pixel no zoom irá modificar a cor de fundo da linha 8x1 ao qual pertence esse pixel.

## Shape

Permite selecionar uma figura da coleção de shapes criada ou carregada de um arquivo, e colar em qualquer posição da tela.

Para navegar entre as figuras, utilize as setas esquerda ou direita. Teclre “Enter” para escolher a figura. Em seguida, utilize os cursores para movimentar a figura pela tela e escolher uma posição. Teclre “Enter” novamente para desenhar a figura nessa posição.

Os shapes do tipo 1 podem realizar operações lógicas sobre a área de destino. Para isso, teclre “Select” quando for desenhar a imagem. As operações são: PSET, AND, OR e XOR.

### Corte

Permite selecionar uma área da tela para recortá-la ou movimentá-la.

Ao selecionar a ferramenta, o usuário deverá marcar a posição inicial e final do retângulo de corte (figura 3.7). A tecla “Espaço” marca o início e o fim da área de corte, que é delimitada por um retângulo tracejado. Em seguida, utilize as setas do cursor, de modo com que a figura seja deslocada até sair completamente dessa área. Esta ferramenta serve também para fazer um *scroll* em uma determinada região da tela.

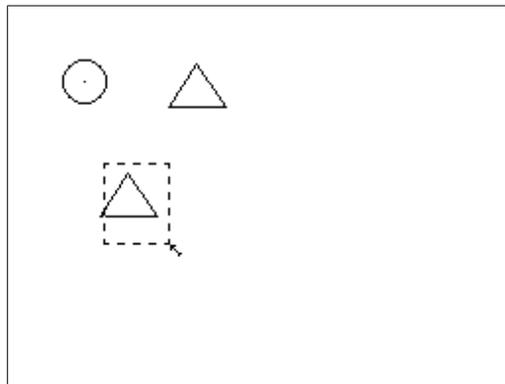


Figura 3.7. Ferramenta de corte.

### 3.6. Salvando a tela

Terminada a edição da imagem, teclre “Enter” e retorne ao menu principal.

A opção “Arquiva tela” permite salvar a imagem em diversos formatos:

- Display – salva a tela em formato binário, juntamente com o display selecionado.
- Layout – salva somente a tabela de padrões (Ink) da imagem.
- Compac – salva as tabelas de padrão (Ink) e cores (Paper) separadas.
- Editor – salva a tela no formato do Sistema Editor de Adventures [1].

A imagem salva na opção “Display” é a única que pode ser aberta fora do programa Graphos III sem a necessidade de um programa que carregue a imagem.

O programa base para carregar telas desse tipo é apresentado a seguir.

```
10 SCREEN 2
20 BLOAD "TELA.SCR",R
30 GOTO 30
```

## 4- Editor de alfabetos

O editor de alfabetos permite alterar o mapa de caracteres do MSX e salvar o mapa modificado em disco. Esse mapa pode ser utilizado nas screens 0, 1 e 2.

Para abrir o editor, selecione a opção “Edita alfabeto” e tecla “Enter”. A figura 4.1 apresenta o editor.

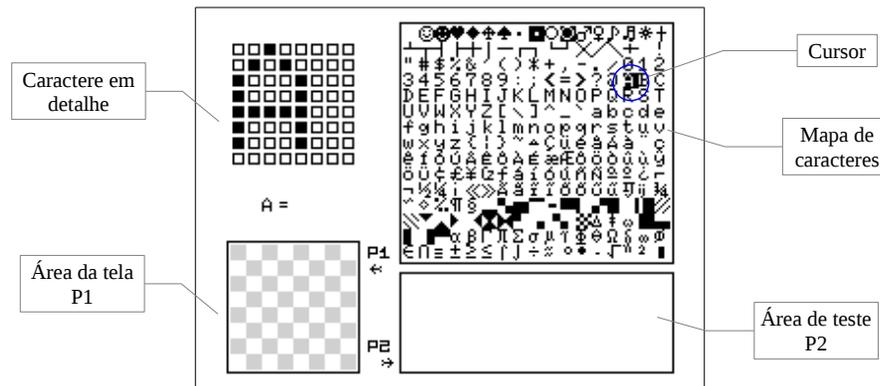


Figura 4.1. Editor de alfabetos.

Inicialmente o cursor se encontra no mapa de caracteres, no qual possui o formato de um bloco que inverte as cores do caractere selecionado. Para navegar, utilize as setas direcionais. Conforme o caractere selecionado muda, a janela de “caractere em detalhe” é atualizada.

Para editar um determinado caractere, coloque o cursor sobre ele e tecla “Enter”, de modo que o controle passe para a janela “caractere em detalhe”. Agora, você pode editar o caractere pixel a pixel, assim como é feito na ferramenta de zoom do editor de imagens. A barra de espaços acende ou apaga um pixel, enquanto que a tecla “Esc” cancela a edição e “Enter” valida, retornando ao mapa de caracteres.

A tecla “Select”, quando no “mapa de caracteres”, abre um menu com quatro funções:

- Clear P2 – limpa a área P2.
- Edita P2 – permite acessar a área P2.
- Clear fonte – apaga fonte corrente, retornando a fonte padrão do MSX.
- Edita P1 – edita a área P1.

A tecla “Select”, quando no “detalhe do caractere”, abre um menu com quatro funções:

- Clear – limpa caractere.
- Inverte – inverte as cores do caractere.
- Espelha – espelha horizontalmente o caractere.
- Rotação – rotaciona o caractere 90 graus anti-horário.
- Edita P1 – edita a área P1.

A área P2 é uma área destinada a testar o alfabeto. Utilize as letras, números e símbolos para escrever, sempre movendo o cursor com as setas. O cursor não se movimenta automaticamente. Tecla “Enter” para sair do modo P2.

Obs: Há um bug no modo P2, onde é possível escrever sobre o mapa de caracteres, sem alterá-lo.

A área P1 é um recorte da tela principal (editor de imagens), começando pelo canto superior esquerdo. Essa ferramenta é útil para criar mosaicos na tabela de caracteres. Quando P1 é acessado pelo “mapa de caracteres”, permite se movimentar pela tela principal através das setas. A tecla “Enter” termina a navegação. Quando acessado através do “caractere em detalhe”, permite selecionar uma área de 8x8 pixels da imagem (indicada na janela P1) e copiá-la para o caractere editado. Vejamos como.

### *Criação de mosaicos de caracteres*

A área P1 é útil na criação de mosaicos de caracteres. Entretanto, necessitamos antes abrir o editor de telas e colocar ali o desenho que desejamos utilizar no mosaico. A figura 4.2 apresenta a imagem utilizada como exemplo.

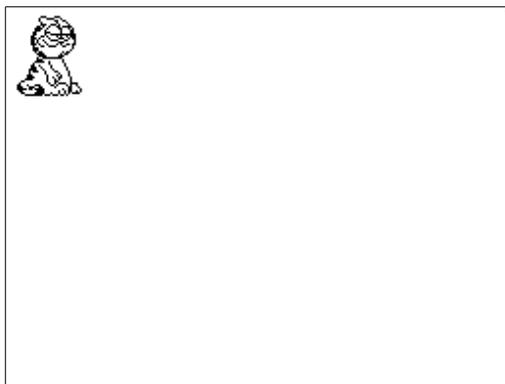


Figura 4.2. Desenho utilizado.

Voltando ao editor de alfabetos, abriremos a área P1 pelo “mapa de caracteres”. O desenho surge então nessa área (figura 4.3).

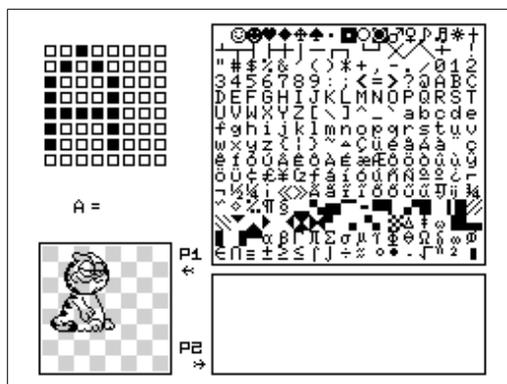


Figura 4.3. Desenho na área P1.

Cada retângulo em cinza e branco dessa região representa um bloco de 8x8 pixels na imagem. Observando a figura 4.3, percebemos que o desenho ocupa 5 blocos de largura por 6 de altura. Então, serão necessários 30 espaços do mapa de caracteres para armazenar todo o desenho. Nota-se que há blocos vazios no mosaico. Eles também serão armazenados, uma vez que a montagem do mosaico será facilitada se considerarmos todos os blocos.

O primeiro passo para montar o mosaico é escolher uma área da tabela de caracteres para armazenar o desenho. O ideal é escolher a área de símbolos, preservando o código ASCII (índices de 0 a 127). A área escolhida no exemplo está assinalada com o retângulo verde na figura 4.4.

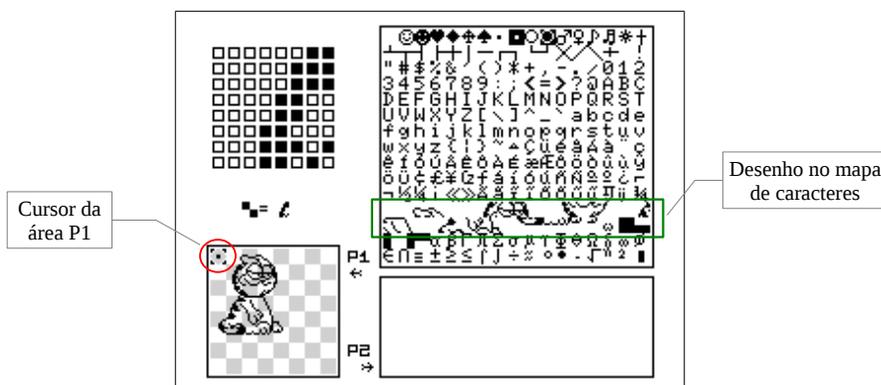


Figura 4.4. Mosaicagem.

Os trinta caracteres deverão ser editados um a um, utilizando a área P1 para copiar o padrão. A correspondência de blocos de caracteres entre a área P1 e o “mapa de caracteres” é apresentada a seguir.

Mapa de caracteres	Área P1
...	01 02 03 04 05
...	06 07 08 09 10
01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17	11 12 13 14 15
18 19 20 21 22 23 24 25 26 27 28 29 30 xx xx xx xx	16 17 18 19 20
...	21 22 23 24 25
...	26 27 28 29 30

Para copiar um padrão, primeiro selecione o caractere do “mapa de caracteres” correspondente ao bloco da área P1 a ser copiado. Entre em seguida no modo de “caractere em detalhe”. Abra o menu, pressionando “Select”. Escolha a opção “Edita P1”. Um cursor aparece na área P1, permitindo a navegação pelos blocos (na imagem 4.4, o cursor está sobre o bloco 01). Ao pressionar “Enter”, o padrão do bloco selecionado é copiado para o “caractere em detalhe”. Ao teclar “Enter” novamente, a mudança do caractere é confirmada. A tecla “Esc” cancela qualquer uma das duas operações durante a edição.

Se salvarmos o alfabeto, podemos carregá-lo na screen 1 e montar o mosaico do desenho, conforme mostra a figura 4.5.



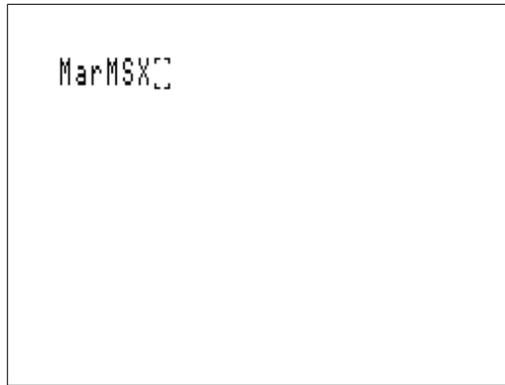


Figura 5.1. Texto utilizado como exemplo.

A seleção permite delimitar áreas múltiplas de 8 pixels. Dessa forma, uma área em torno do texto será selecionada, de acordo com essa limitação (figura 5.2).

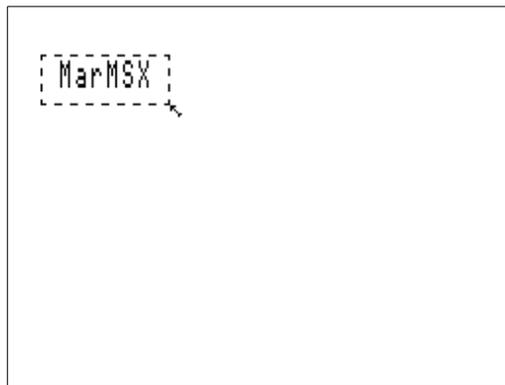


Figura 5.2. Seleção do objeto de interesse.

A seleção do ponto inicial e final da área do shape é feita com o “Espaço”. Ao criar a seleção, é possível mover a figura dentro do retângulo tracejado com as setas de modo a ajustá-la. Tecele “Enter” para terminar a criação do shape.

Para utilizar o shape, volte ao editor principal e escolha a opção “Shape” no menu de miscelânea. A imagem pode ser copiada quantas vezes se desejar na tela, como mostra a figura 5.3.

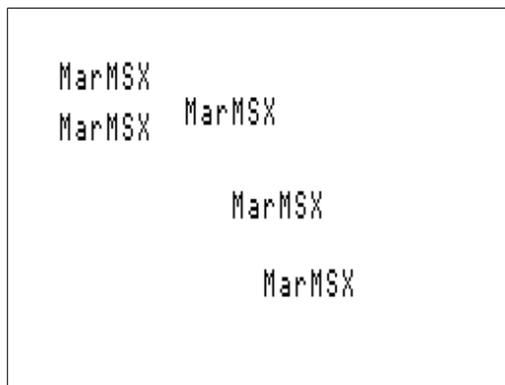


Figura 5.3. Utilizando o shape criado.

Ao ser criado um segundo shape, ele será empilhado junto com o primeiro. A coleção de imagens do shape pode conter figuras de tamanhos variados, bem como tipos diferentes. Não é possível apagar um determinado shape, mas somente todos eles de uma vez.

### 5.1. Máscara

Máscara é um recurso que permite que o shape seja colado sobre a tela, copiando apenas os pixels desejados.

Os shapes dos tipos 1 e 2 copiam para a tela toda a área shape, inclusive o fundo de tela. Dessa forma, toda a área do retângulo do shape é substituída na tela quando ele é copiado, conforme mostra a figura 5.4.

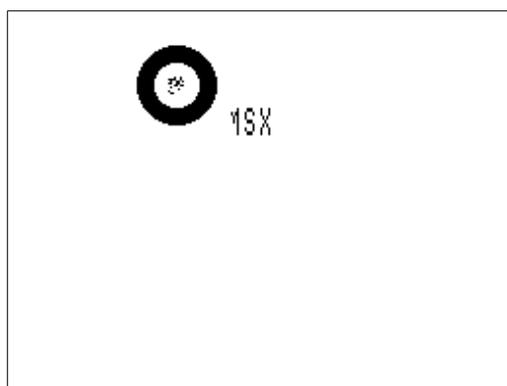


Figura 5.4. Shape sem máscara.

Quando é feita a opção pelo shape do tipo 3 ou 4, um editor surge após a definição da área do shape, contendo a imagem do shape monocromática (Ink) com os pixels invertidos. Essa imagem é a máscara que será utilizada para definir a área útil do shape. Ela pode ser modificada, de acordo com o interesse do usuário.

Os pixels apagados da máscara indicam a região do shape que será copiada para a tela e os pixels acesos não. A figura 5.5 mostra como o shape do tipo 3 preserva a área da tela definida pela máscara, não copiando os pixels brancos do entorno do círculo.



Figura 5.5. Shape com máscara.

## 6- Formatos de arquivo do Graphos III

### 6.1. Alfabeto

O arquivo de alfabeto contém um *header* comum de arquivos do MSX, mais a tabela de caracteres contendo o padrão das letras. Assim temos:

Offset (hexa)	Tamanho (bytes)	Descrição	
0000	1	Tipo de arquivo do MSX	Header
0001	2	Endereço inicial	
0003	2	Endereço final	
0005	2	Endereço de execução	
0007	8	Padrões do caractere índice 0	Dados
000F	8	Padrões do caractere índice 1	
...	...	...	
07FF	8	Padrões do caractere índice 255	

Os padrões do caractere são formados por um bloco de 8x8 pixels, onde cada linha é representada por um byte. Por exemplo, para a letra “A”, localizado no endereço x do arquivo, temos:

Offset	Valor
x	&B00100000
x+1	&B01010000
x+2	&B10001000
x+3	&B10001000
x+4	&B11111000
x+5	&B10001000
x+6	&B10001000
x+7	&B00000000

Para abrir o alfabeto na screen 0, deve-se somar ao valor do endereço inicial do *header* um determinado valor de deslocamento, que resulte em 2048 (&H800). Assim, temos:

```
deslocamento = 2048 - endereco_inicial
```

Por exemplo, se o endereço inicial for &H9200, temos:

```
deslocamento = &H800 - &H9200
```

```
deslocamento = &H7600
```

Em Basic, digite:  
PRINT HEX\$(&H800 - &H9200)

Então, carregue o alfabeto na screen 0:

```
10 SCREEN 0
20 BLOAD "ALFABETO.ALF",S,&H7600
```

Para as screens 1 e 2, o endereço da tabela de caracteres se localiza no endereço 0 da VRAM. Assim ,temos:

```
deslocamento = 0 - endereco_inicial
```

Para o mesmo endereço &H9200, o deslocamento a ser utilizado no BLOAD é de &H6E00. Entretanto, a screen 2 exige que a tabela de cores seja alterada com as cores de frente e fundo, do contrário, os caracteres não poderão ser impressos. Além disso, deve-se alterar a tabela de nomes toda para o valor igual a 32 (espaço) e, então, modificar essa tabela de modo que apresente os caracteres desejados.

O exemplo a seguir, mostra como utilizar o alfabeto na screen 2:

```
10 SCREEN 2
20 BLOAD"alfabeto.alf",S,&H6E00
30 FOR E=&H2000 TO &H27FF
40 VPOKE E,&B11110000
50 NEXT E
60 FOR E=6144 TO 6144+255
70 VPOKE E,32
80 NEXT E
90 L=2:C=2
100 M$="MarMSX 2017"
110 GOSUB 500
120 GOTO 120
500 ' Escreve
510 E=6144+L*32+C
520 FOR I=1 TO LEN(M$)
530 VPOKE E+I,ASC(MID$(M$,I,1))
540 NEXT I
550 RETURN
```

Obs: a tabela de nomes da screen 2 divide a tela em três partes. Assim, para carregar o alfabeto nas outras partes, some 256 ou 512 aos valores das três tabelas.

## 6.2. Shapes

O arquivo de shapes é uma coleção de desenhos empilhados, onde cada desenho contém seu próprio *header*. O arquivo de shape não contém o *header* de arquivo do MSX.

O formato de arquivo de shape é descrito a seguir:

Offset (hexa)	Tamanho (bytes)	Descrição
0000	4	Header do shape 1
0004	N1	Dados do shape 1
0004 + N1	4	Header do shape 2
0008 + N1	N2	Dados do shape 2
...	...	...

O *header* de cada shape contém as seguintes informações:

Offset (hexa)	Tamanho (bytes)	Descrição
x	1	ID do shape
x+1	1	Tipo do shape (1-4)
x+2	1	Largura do shape em pixels
x+3	1	Altura do shape em blocos de 8x8

Exemplo de *header*: 01 01 30 06.

ID do shape: 1

Tipo do shape: 1

Largura do shape: &H30 ou 48 pixels

Altura do shape: &H06 ou 6 blocos de 8x8 pixels = 48 pixels

O tamanho dos dados depende do tipo de shape utilizado, uma vez que cada tipo de shape possui uma determinada quantidade de blocos de dados do mesmo tamanho, que varia de 1 a 3. O tamanho de cada bloco é calculado multiplicando-se diretamente os valores da altura pela largura contidos no *header*. Nesse exemplo, estamos utilizando shapes do tipo 1, que possui apenas um bloco. Dessa forma, temos:

$$48 \times 6 \times 1 = 288 \text{ bytes} \quad \text{ou}$$
$$\&H30 \times \&H06 \times 1 = \&H120 \text{ bytes}$$

Caso esse shape fosse o primeiro da lista, o *header* estaria localizado no arquivo nas posições de 0 a 3, enquanto que os dados estariam de 4 a 291.

A tabela a seguir informa a quantidade de blocos de informação, bem como o tipo de dados contidos em cada um deles e o tamanho final dos dados.

Tipo	Blocos	Tamanho
1	Tabela de padrão.	Largura x Altura
2	Tabela de padrão e cores.	2 x Largura x Altura
3	Tabela de padrão da máscara e padrão.	2 x Largura x Altura
4	Tabela de padrão da máscara, padrão e cores.	3 x Largura x Altura

O arquivo do tipo shape possui sempre o tamanho múltiplo de 128 bytes. Assim, é possível que o arquivo contenha lixo na área não usada. Entretanto, é fácil identificar o fim de arquivo: quando o próximo ID do shape possuir valor igual a 255 (&HFF), significa o final do arquivo.

### 6.3. Layout

Este arquivo contém apenas a tabela de padrões compactada de uma imagem, além do *header* padrão do MSX, descrito na seção 6.1.

O modo utilizado na compactação da tabela de caracteres é descrito a seguir.

Seja  $p$  o ponteiro de arquivo e  $val(p)$  o valor retornado de  $p$  com 1 byte de tamanho, tem-se:

- Se  $val(p) = 0$ , então  $val(p+1)$  indica a quantidade de linhas vazias seguidas.
- Se  $val(p)=255$ , então  $val(p+1)$  indica a quantidade de linhas cheias seguidas.
- Se  $val(p) \neq 0$  e  $val(p) \neq 255$ , então  $val(p)$  indica o padrão a ser desenhado.

Obs: um byte na VRAM equivale a uma linha de 8x1 pixels na tela.

Exemplos:

$p$ 00 02	$p$ 01 06 D7	$p$ FF 03
Há 2 linhas vazias na tela: 00000000 00000000	Padrões das próximas 3 linhas: 00000001 = 01 00000110 = 06 11010111 = D7	Há 3 linhas cheias na tela: 11111111 11111111 11111111

Quando o ponteiro  $p$  encontra os valores 00 ou FF, ele sabe que o byte seguinte corresponde ao número de linhas seguidas. Dessa forma, ele salta duas posições em busca da nova informação. No caso de ser diferente de 00 ou FF, o ponteiro salta apenas uma posição.

No Graphos III versão 1.2, todos os bytes que indicam o padrão da linha sofrem um deslocamento positivo de &H99, inclusive o 00 e FF. Assim, “00 02” fica “99 02”.

O programa a seguir exemplifica como carregar uma imagem no formato “.LAY” na screen 2 do MSX.

```

10 SCREEN 2
20 LINE(0,0)-(255,191),15,BF
30 BLOAD"TESTE.LAY"
40 E=&H9200 : V=0
50 P=PEEK(E)
60 IF PEEK(E)=0 AND PEEK(E+1)=0 THEN 180
70 IF P<>0 THEN 130
80 FOR F=1 TO PEEK(E+1)
90 V=V+1
100 NEXT
110 E=E+2
120 GOTO 50
130 N=1
140 IF P=255 THEN N=PEEK(E+1) : E=E+2 ELSE E=E+1
150 FOR I=1 TO N : VPOKE V,P : V=V+1 : NEXT
160 P=PEEK(E)
170 IF P<>0 THEN 130 ELSE 60
180 GOTO 180

```

Supondo que o endereço inicial do header seja &H9200.

Dica: a instrução da linha 20 substitui o preenchimento da tabela de cores com as cores de fundo e frente. Esta solução é bem mais rápida.

#### 6.4. Compac

Nesse tipo de formato, a tela é salva em dois arquivos: o primeiro é a tabela de caracteres no mesmo formato do Layout, mas com a extensão “.VDC”; o segundo é a tabela de cores da imagem compactada utilizando o algoritmo Run-Length [3], possuindo a extensão “.ATC”.

O modo de compactação Run-Length utilizado é descrito a seguir:

Seja  $p$  o ponteiro de arquivo e  $val(p)$  o valor retornado de  $p$  com 1 byte de tamanho, tem-se:

- $val(p)$  indica a quantidade de vezes que um padrão de cor será repetido sequencialmente na VRAM. Se quantidade for zero, fim de arquivo.
- $val(p+1)$  indica quais as cores de frente e fundo que serão repetidas na VRAM.

O programa a seguir exemplifica como carregar o arquivo de cores “.ATC”.

```

10 SCREEN 2
20 BLOAD"TELA.ATC"
30 E=&H9200 : V=&h2000
40 C=PEEK(E)
50 IF C=0 THEN 130
60 D=PEEK(E+1)
70 FOR F=1 TO C
80 VPOKE V,D
90 V=V+1
100 NEXT
110 E=E+2
120 GOTO 40
130 GOTO 130

```

Supondo que o endereço inicial do header seja &H9200.

## 7- Extra: leitor de shapes em Basic

O programa a seguir lê o header de todos os shapes em um arquivo e imprime na tela.

```
10 SCREEN 0
20 OPEN"SHAPE.SHP" FOR INPUT AS#1
30 SH=1 : LID=255
40 TM(1)=1:TM(2)=2:TM(3)=2:TM(4)=3
50 GOSUB 1000
60 SH=SH+1
70 IF ID<>255 THEN 50
80 CLOSE 1 : END
1000 ' Lê header
1010 ID=ASC(INPUT$(1,#1))
1015 IF LID<>255 THEN IF ID <> LID+1 THEN PRINT "Erro no arquivo." :
ID=255
1016 IF ID=255 THEN RETURN
1020 TP=ASC(INPUT$(1,#1))
1030 L=ASC(INPUT$(1,#1))
1040 A=ASC(INPUT$(1,#1))
1050 C=L*A*TM(TP)
1060 PRINT "Shape #";SH
1070 PRINT " ID:";ID
1080 PRINT " Tipo:";TP
1090 PRINT " Largura:";L;"pixels"
1100 PRINT " Altura:";A;"blocos"
1110 PRINT " Tamanho:";C;"bytes"
1120 PRINT
1130 IF C - 100 < 0 THEN DM$=INPUT$(C,#1) : RETURN ELSE
DM$=INPUT$(100,#1):C=C-100:GOTO1130
```

O próximo programa lê os shapes e desenha na screen 2.

```
10 SCREEN 2
15 LINE(0,0)-(255,191),15,BF
20 OPEN"SHAPE.SHP" FOR INPUT AS#1
30 LID=255
40 TM(1)=1:TM(2)=2:TM(3)=2:TM(4)=3
50 GOSUB 1000
60 IF ID<>255 THEN GOSUB 1500 ELSE 90
70 A$=INPUT$(1) : LINE(0,0)-(255,192),15,BF
80 GOTO 50
90 CLOSE 1
100 GOTO 100
1000 '
1001 ' Lê header
1002 '
1010 ID=ASC(INPUT$(1,#1))
1015 IF LID<>255 THEN IF ID <> LID+1 THEN PRINT "Erro no arquivo." :
ID=255
1016 IF ID=255 THEN RETURN
1020 TP=ASC(INPUT$(1,#1))
1030 L=ASC(INPUT$(1,#1))
1040 A=ASC(INPUT$(1,#1))
```

```

1050 C=L*A*TM(TP)
1060 RETURN
1500 '
1501 ' Lê shape
1502 '
1510 IF TP=1 OR TP=2 THEN 1550
1520 FOR F=1 TO L*A
1530 D=ASC(INPUT$(1,#1))
1540 NEXT F
1550 E=0 : GOSUB 1580 : ' Lê tabela de caracteres
1560 IF TP=1 OR TP=3 THEN RETURN
1570 E=&H2000 : ' Lê tabela de cores
1580 FOR F=1 TO L*A
1590 V=ASC(INPUT$(1,#1))
1600 VPOKE E,V
1610 E=E+1
1620 IF E MOD 256 >= L THEN E=E+256-L
1630 NEXT F
1640 RETURN

```

Obs: o comando INPUT para leitura de dados binários está sujeito ao caractere de fim de arquivo (&H1A), interrompendo a leitura com a mensagem de erro "Input past end". Caso isto venha a ocorrer, deve-se substituir o método de leitura do programa de INPUT por GET, apesar deste ser mais lento que o outro.

O programa anterior foi reescrito utilizando o método de leitura GET.

```

10 SCREEN 2
15 LINE(0,0)-(255,191),15,BF
20 OPEN"SHAPE.SHP" AS#1 LEN=1
25 FIELD #1, 1 AS C$
30 P=1 : LID=255
40 TM(1)=1:TM(2)=2:TM(3)=2:TM(4)=3
50 GOSUB 1000
60 IF ID<>255 THEN GOSUB 1500 ELSE 90
70 A$=INPUT$(1) : LINE(0,0)-(255,192),15,BF
80 GOTO 50
90 CLOSE 1
100 GOTO 100
1000 '
1001 ' Lê header
1002 '
1010 GET #1,P : ID=ASC(C$)
1015 IF LID<>255 THEN IF ID <> LID+1 THEN PRINT "Erro no arquivo." :
ID=255
1016 IF ID=255 THEN RETURN
1020 GET #1,P+1 : TP=ASC(C$)
1030 GET #1,P+2 : L=ASC(C$)
1040 GET #1,P+3 : A=ASC(C$)
1050 C=L*A*TM(TP) : P=P+4
1060 RETURN
1500 '
1501 ' Lê shape
1502 '
1510 IF TP=1 OR TP=2 THEN 1550

```

```
1520 FOR F=1 TO L*A
1530 GET #1,P : D=ASC(C$) : P=P+1
1540 NEXT F
1550 E=0 : GOSUB 1580 : ' Lê tabela de caracteres
1560 IF TP=1 OR TP=3 THEN RETURN
1570 E=&H2000 : ' Lê tabela de cores
1580 FOR F=1 TO L*A
1590 GET #1,P : V=ASC(C$) : P=P+1
1600 VPOKE E,V
1610 E=E+1
1620 IF E MOD 256 >= L THEN E=E+256-L
1630 NEXT F
1640 RETURN
```

## 8- Créditos e bibliografia

O artigo foi escrito por Marcelo Silveira, Engenheiro de Sistemas e Computação, formado pela Universidade do Estado do Rio de Janeiro.

Data: dezembro de 2017.

E-mail: flamar98@hotmail.com  
Homepage: marmsx.msxall.com

Referências bibliográficas:

- [1] – Tilt Net - <http://www.tilt.net>
- [2] – Tilt Net - <http://www.tilt.net/clip/bits04.htm>
- [3] – Run-Length – [http://en.wikipedia.org/wiki/Run-length\\_encoding](http://en.wikipedia.org/wiki/Run-length_encoding)