

# COMPUCLUB

*News*

**NOVO ANO, NOVA FASE,  
NOVAS PERSPECTIVAS**

**COMPUCLUB**

O melhor Clube de Soft  
para MSX, TRS-80, TRS-Color e ZX-Spectrum

ANO 1 - Nº 0 - JANEIRO/FEVEREIRO 1987

# PROBLEMAS COM A SUA FITA?

Normalmente, quando compramos uma fita e não conseguimos carregar os programas em nosso equipamento, pensamos logo que o problema está na fita, e logo vem o impulso de devolvê-la sob alegação de que está com defeito. Alguns poucos associados têm apressadamente devolvido as fitas que enviamos, afirmando não terem conseguido carregar os programas ali contidos. Em alguns casos, a fita estava mesmo com defeito mas, na grande maioria, a fita estava perfeita. Como explicar isto?

A verdade é que recuperar programas pode ser difícil se você não observar certos cuidados. Se o seu micro é da linha Sinclair, você sabe bem do que estamos falando. Mesmo que não seja, porém, se você utiliza fitas cassete para armazenar seus programas, o que aqui vamos dizer será certamente do seu interesse.

Microcomputadores são instrumentos muito sensíveis e, via de regra, não "dançam" corretamente se a música não for bem tocada. É possível, entretanto, que a gravação tenha sido bem executada, que a fita seja de excelente qualidade, que o gravador esteja em perfeito estado de conservação, e mesmo assim você não ser bem sucedido ao tentar recuperar o programa ali contido.

Como pode ser isso possível?

Bem, os gravadores possuem um pequeno parafuso que permite ajustar o cabeçote na linha vertical, mais para cima ou para baixo de uma certa posição. O detalhe é que a melhor performance se dá quando o cabeçote está ajustado de modo a percorrer o mesmo caminho percorrido pelo cabeçote que executou a gravação. É óbvio que isto sempre ocorre se a gravação e leitura foram executados pelo mesmo gravador, pois o cabeçote que grava é o mesmo que lê. Mas se a gravação tiver sido executada por outro gravador, a leitura só será bem sucedida se os gravadores estiverem igualmente ajustados. Como os gravadores, em geral, não vem ajustado de fábrica segundo o padrão industrial, você pode, eventualmente, ter problemas com este tipo de ajuste.

Este ajuste é executado facilmente, do seguinte modo: com o gravador tocando a fita, giramos o parafuso em ambas as direções, até encontrarmos a posição em que se ouve mais fortemente o som ali gravado. O problema é que alguns gravadores, como o National, não possuem a abertura que nos permite acesso ao referido parafuso, caso em que é necessário desmontá-lo para se executar o ajuste mencionado, o que só deve ser feito, é óbvio, por um profissional habilitado.

Assim, você pode ter problemas com algumas fitas, e com outras não. Tudo que você precisa, porém, para fazer com que esse problemas desapareçam, é ter uma certa convivência com o seu equipamento que lhe permita identificar os pontos nevrálgicos da questão e realizar os ajustes quando necessários.

Se você já executou o ajuste mencionado, denominado ajuste de azimute, então ainda resta proceder a uma limpeza do gravador antes de tentar recuperar seu programa. Limpe bem o cabeçote, mas não se esqueça da roldana de borracha localizada na cavidade, à direita do cabeçote.

Uma vez executados esses procedimentos, devemos começar a pesquisar qual o volume do gravador mais adequado para o seu micro. Mesmo isto, porém, pode variar de programa para programa, ou de acordo com o equipamento que gravou a fita. O melhor é começar com um volume bem baixo, e ir aumentando lentamente até obter a recepção do programa. Quando isto acontecer, marque o volume obtido, pois este será o volume ideal para a recuperação de programas em seu micro.

Um pouco mais de paciência será muito saudável para o seu relacionamento com o microcomputador, e evitará que você, desnecessariamente, devolva fitas que se encontram perfeitas e prontas para serem utilizadas.

# COLABORE COM A SUA REVISTA

Nesta revista você pode fazer mais do que apenas ler: você pode escrever nela também. Por uma questão operacional, entretanto, vamos definir as bases para que isto possa se realizar de modo organizado.

Você pode participar escrevendo artigos ou dicas especiais. Os artigos devem versar sobre características dos microcomputadores, ou sobre um determinado soft em particular. Quanto mais prático for o seu artigo, melhor. E tratando de um soft, em particular, será conveniente descrever toda a sua estrutura, comentando seus objetivos, destacando suas sub-rotinas, etc.

Já as dicas especiais dizem respeito a um conhecimento rápido acerca de uma característica muito interessante que você descobriu no seu equipamento. Um pequeno programa capaz de produzir um bonito efeito na tela, ou uma forma de você ampliar os recursos do seu micro, enfim, algo que você considera especial.

Os trabalhos deverão ser datilografados em folhas de papel ofício, com 60 toques por linha e um número máximo de 30 linhas por lauda. Ao final do trabalho deverá ser incluído um breve currículo do autor. Havendo a listagem de um soft em seu trabalho, ele deverá se fazer acompanhar de uma fita com o referido soft gravado duas vezes no lado A.

Todos os trabalhos selecionados para publicação serão remunerados, no mês em que forem publicados, na seguinte base: Artigos receberão Cz\$400,00 e dicas especiais, Cz\$100,00. O crédito de autoria será dado ao seu autor através da publicação de seu nome logo abaixo do título, e de demais referências constantes do currículo, ao final do trabalho, exceto no caso de dicas especiais, em que constará apenas o nome do seu autor.

Também por questões operacionais o clube não devolverá os trabalhos não aceitos para publicação.

## Editorial

Você, associado do COMPUCLUB, está de parabéns! Porque você foi a peça principal que nos permitiu transformar em realidade o tão acalentado sonho de termos a nossa própria revista. Isto certamente não teria sido possível sem o seu apoio, sem a sua persistência, sem a sua participação efetiva em todas as nossas realizações.

A revista que ora se encontra em suas mãos é o resultado direto do seu esforço: a única revista de microcomputação sem preço de capa; uma prerrogativa exclusiva dos associados do COMPUCLUB.

Agora, é nosso dever fazer com que esta publicação cresça e amplie os seus horizontes. Para isso, contamos com você; com a sua participação no encaminhamento de artigos e dicas especiais a serem publicados nas futuras edições do COMPUCLUB News. Todos os trabalhos selecionados serão remunerados (ver 2ª capa), e o devido crédito de autoria registrado nas páginas da revista por ocasião da publicação.

Do ponto de vista editorial, fizemos algumas inovações que ainda precisam ser testadas. Assim, toda a titulação, bem como alguns textos, foram obtidos via impressora, utilizando-se o TK90X e o soft "Art Studio". Uma consequência dessa prática é a ausência de acentos e cedilhas. Apesar de saber que isto fere o nosso vernáculo, esperamos que nossos leitores, habituados com as características dos micros, não se sintam demasiadamente perturbados por este detalhe.

O mais importante é que esta edição, de número 0, inaugura uma nova etapa na vida do nosso clube. A fase dos boletins ficou para trás, e novos horizontes se descortinam à nossa frente. Aliando-se a isto, temos o fato de estarmos nos deslocando de Viçosa para Belo Horizonte. Isto certamente nos trará algumas vantagens adicionais.

De modo que, em meio à crise que se abate sobre nosso país, nós, que emprestamos a nossa força de trabalho a serviço do clube, vemos o futuro com muita esperança. E podemos mesmo lhe afiançar que, muito brevemente, progressos indescritíveis se farão notar. Com a divisão do clube em setores, finalmente teremos um setor integralmente dedicado à seleção e gravação de softs, o que significa que nossas fitas estarão, mais do que nunca, recheadas dos melhores softs e das novidades mais recentes.

Um grande abraço a todos.

Presidente

ANO 1 - Nº 0 - JANEIRO/FEVEREIRO 1987

## Índice

- 6** Entenda melhor o seu ZX-Spectrum
- 10** Peek e Poke no TRS-80
- 14** Gerando o alfabeto com o comando DRAW (TRS-Color)
- 18** Aumente o potencial do seu ZX-81
- 22** As telas do MSX
- 26** A História do Compuclub

### SEÇÕES ESPECIAIS

- |                            |                           |
|----------------------------|---------------------------|
| <b>2 AJUSTE DE AZIMUTE</b> | <b>16 SOFTS TRS-COLOR</b> |
| <b>2 COLABORAÇÕES</b>      | <b>17 SEÇÃO DE CARTAS</b> |
| <b>4 NOTÍCIAS</b>          | <b>20 SOFTS ZX-81</b>     |
| <b>8 SOFTS ZX-SPECTRUM</b> | <b>21 FITAS EXTRAS</b>    |
| <b>12 SOFTS TRS-80</b>     | <b>25 SOFTS MSX</b>       |

### EXPEDIENTE

O COMPUCLUB News é o órgão de divulgação do COMPUCLUB, uma entidade sem fins lucrativos, de caráter cultural, educacional e científico, empenhada na difusão de artigos e matéria de interesse geral na área da microinformática.

COMPUCLUB - Caixa Postal 3521  
(CEP 30112) BELO HORIZONTE, MG.

O COMPUCLUB envia, a todos os seus associados, regular e gratuitamente, a revista do clube e programas gravados em fita cassete ou disquete. Seus associados contribuem, anualmente, com o equivalente a 1,2 OTN e, mensalmente, por ocasião do recebimento das fitas, pagam uma taxa para aquisição de softs, e ressarcem o clube das importâncias gastas com o material enviado, embalagem e despesas postais.

Presidente e Editor Responsável  
Christiano P. Silva Neto.

# NEWS

## INAMPS RECONHECE DOENÇA DE DIGITADOR

A tenossinovite, uma inflamação que atinge os nervos das mãos e braços de quem trabalha com movimentos repetitivos, como os digitadores, foi reconhecida pelo Inamps (Instituto Nacional de Assistência Médica e Previdência Social), como doença profissional. Isso significa que seus portadores, além de tratamento gratuito, terão direito também a auxílio-doença.

Segundo Maria de Fátima Andreazzi, coordenadora de programas especiais do Inamps, com esta medida, grande parte da população deve ser beneficiada, pois a incidência de tenossinovite não atinge somente os cerca de trinta mil digitadores registrados no país, mas também datilógrafos e até mesmo pianistas.

"Fizemos uma interpretação da lei" afirma Andreazzi, "pois a doença não é diretamente ocupacional, mas decorrente de más condições de trabalho". Neste sentido, o Inamps já entrou em contacto com o Ministério do Trabalho para que haja uma maior fiscalização, principalmente no que se refere à duração da jornada de trabalho que, atualmente, é de seis horas diárias.

Além da jornada prolongada, sem períodos de descanso intercalados, outros fatores contribuem para o aparecimento da tenossinovite, como a má postura e frio excessivo a que são submetidos os digitadores (oriundo do ar condicionado necessário aos equipamentos), já que a temperatura ideal seria em torno de 24 graus. E os sintomas são: dores e edema (inchaço) nas articulações, impotência funcional (dificuldade para realizar movimentos) e diminuição da capacidade motora.

Os médicos do Inamps estão começando a ser preparados para os casos de tenossinovite. Além do diagnóstico clínico habitual, o médico deverá perguntar sobre as condições de trabalho do paciente e, se houver dúvidas, deverá solicitar uma verificação "in loco" da empresa. Após confirmação da ocorrência da doença, o médico poderá tratá-la como acidente de trabalho e encaminhar o paciente ao setor previdenciário para recebimento do auxílio-doença.

## PRODUZINDO CHIPS

O processo de fabricação dos circuitos integrados é complexo porque significa espremer num minúsculo espaço de 25 milímetros quadrados, milhares de componentes eletrônicos, tais como resistores, diodos e transistores. A acomodação desses componentes deve ser perfeita, com total separação entre eles do ponto de vista elétrico. As dimensões reduzidas significam maior velocidade de processamento. A fabricação tradicional, feita com processos óticos, envolve vários estágios:

1) Deve-se ter em mãos o pro-

jeto do circuito. Para desenhá-lo, usa-se computadores com capacidade gráfica, com o desenho feito em tamanho ampliado. Os chips não são sempre os mesmos e há aqueles com funções específicas, chamados dedicados, que implicam em projetos específicos. As informações são gravadas em fita magnética e, como o Brasil não domina ainda a tecnologia para passar essas informações à máscara, elas são remetidas ao exterior.

2) Da fita magnética, o projeto de chip deve ser passado para a máscara (funcionando como o negativo em relação ao papel fotográfico), que deixa ou não passar a luz em determinados pontos. A máscara, uma placa de plástico de quatro metros de diâmetro, é reduzida para 25 centímetros quadrados, ou seja, quarenta vezes menor em relação ao tamanho original.

3) Reduzida, a máscara é montada numa placa de vidro e novamente reduzida para 25 milímetros, o tamanho do chip. Nesse estágio, está pronta para ser transferida para o silício.

4) O chip (que significa lasca em inglês) é produzido a partir do silício, o material mais abundante na Terra, depois do oxigênio. Purificado e industrializado na forma de cilindros metálicos, esse material é depois fatiado em finíssimas "rodela" que são depois recobertas de material plástico sensível à luz (como o papel fotográfico) chamado Photoresist.

5) Sobre o Photoresist são aplicadas as máscaras (estágio 3) de vidro. Joga-se sobre elas luz ultravioleta, que atravessará a máscara. Esse processo é repetido de oito a dez vezes, pois há necessidade de várias máscaras para se chegar a um único circuito integrado.

6) Ao final de cada máscara, a "rodela" de silício recebe um banho químico (como o processo de se revelar uma foto) para remoção das camadas de óxido. Cada vez que o silício recebe uma máscara, está sendo construído um sistema de contatos elétricos.

7) Terminada a transferência do projeto para o silício, a "rodela", que contém cerca de duzentos circuitos integrados, é cortada. Os circuitos são então encapsulados (i.e., recebem uma cobertura plástica com várias "perminhas" de metal) e testados.

## O CHIP BRASILEIRO

Os sistemas mais evoluídos eliminam totalmente as máscaras, gravando a imagem diretamente na lâmina de silício. Tais sistemas eletrônicos diminuem os problemas ocasionados pela difração da luz (fenômeno ótico constatado quando uma onda luminosa, ao incidir sobre um objeto, propaga ondas situadas na sombra do mesmo). Com isso é possível uma redução no tamanho dos desenhos, que serão mais pre-

cisos. No espaço onde cabem cerca de dez mil componentes, será possível desenhar cerca de quatro milhões, o que vai gerar dispositivos mais complexos e velozes. O sistema em que o Brasil está interessado imprime na última máscara o circuito para ser fornecido às indústrias.

## O COMPUTADOR NASCEU NA 2ª GUERRA

Alta tecnologia e guerra sempre andaram de mãos dadas. O computador, ao lado da bomba atômica e de gases mortíferos, é um dos mais refinados filhos tecnológicos da 2ª Guerra Mundial. Pesquisas para criar uma máquina que processasse informações em altíssima velocidade existiam desde a década de 20 - mas foi a guerra quem acelerou o invento. A paternidade da nova tecnologia não é fácil de ser definida - alemães, ingleses e americanos se enveredaram por caminhos distintos, cada um ignorando a pesquisa alheia, e acabaram chegando a equipamentos similares.

A partida high-tech na guerra foi dada pelos alemães. Para cifrar as comunicações das forças nazistas, Hitler lançou mão de uma máquina batizada de Enigma. Sua função era embaralhar as comunicações na emissão e desembaralhá-las na recepção. Sua capacidade era gigantesca para a época - conseguia criar 22 milhões de combinações diferentes.

Graças à deserção de um pesquisador judeu, os ingleses ficaram sabendo da existência do Enigma em 1938. Aí entra em cena Alan Turing, aluno predileto de Einstein na Universidade de Princeton, a quem o serviço secreto inglês qualificou de desleixado, sonhador e homossexual. Turing podia ser tudo isso, mas foi um dos gênios da matemática no século 20. Resultado: foi encarregado, pelo mesmo serviço secreto, de construir uma máquina que decodificasse o Enigma nazista.

Turing acabou criando uma máquina tão eficiente que o próprio Churchill resolveu inspecionar o projeto, batizado de Ultra. Com esse equipamento, Churchill comandou batalhas célebres como o Dia da Água, quando Goering deu à Luftwaffe a ordem de ataque final para destruir a Força Aérea Inglesa.

Os americanos também chegaram a uma máquina, com características de um computador moderno, para cálculos de balística - que viria a se transformar no Eniac (Electronic Numerical Integrator and Computer, i.e., Computador e Integrador Numérico Eletrônico). Graças à genialidade de John Von Neumann, que pesquisava a bomba de hidrogênio, o Eniac representou um salto na história tecnológica ao incorporar o conceito de programa armazenado.

## UMA NOVA ERA NASCE NOS LABORATÓRIOS

O nascimento da civilização dependeu do aparecimento de materiais que deram seus nomes às épocas: Idade da Pedra, Idade do Bronze, Idade do Ferro. Contudo, por mais vitais que tenham sido à caminhada do homem, das cavernas aos computadores, até recentemente os materiais empregados tinham sido principalmente dádivas da natureza

ou simples melhoramentos deles: areia fundida em vidro, minérios purificados em metais. Agora estamos no limiar de uma nova era, a dos materiais feitos pelo homem.

Hoje, os cientistas podem traçar as estruturas básicas e propriedades dos materiais para atender às suas necessidades. Empresas líderes na invenção e produção destes ingredientes do futuro estão em posição para dominar muitas indústrias de alta tecnologia.

Uma série de novos materiais, com qualidades notáveis, já está saindo dos laboratórios: vidros que se dobram sem quebrar, plásticos tão duros quanto o aço, metais que podem ser esticados. Dentro das grandes câmaras de vácuo, materiais para serem usados em eletrônica e em ótica eletrônica agora podem ser feitos em camadas da espessura de um átomo.

Na pesquisa de novos materiais, a Europa Ocidental e a União Soviética têm um desempenho irregular. A maioria dos especialistas do ramo coloca os Estados Unidos em primeiro lugar em pesquisa básica, mas atribui ao Japão a liderança no emprego mais eficiente da ciência para produzir novos materiais.

A União Soviética está em último lugar nas quatro grandes áreas da tecnologia, mas está melhor na área de novos materiais, em parte por causa de uma descoberta notável: um sistema para depositar uma camada ultrafina de diamantes na superfície de materiais, para endurecê-los e criar outras vantagens. Por exemplo, o calor gerado pelos chips semi-condutores pode criar problemas em seu funcionamento se for permitido que se eleve; a descoberta soviética torna possível dissipar este calor, conduzindo-o através de uma fina camada de pó de diamante sob o chip.

As empresas americanas têm vantagens em materiais compostos e polímeros. Estes compostos são materiais como a cerâmica e plásticos, reforçados com fibras sintéticas e filamentos de carbono. Alexander MacLachlan, vice-presidente da Du Pont, acredita que os compostos vão substituir os metais em algumas partes de aviões e de automóveis, possibilitando reduzir seu peso e diminuir o gasto de combustível. Os polímeros são plásticos feitos de moléculas longas, resistentes o suficiente para substituir o aço em algumas aplicações e leves o bastante para substituir o alumínio em outras.

## O MUNDO DOS COMPUTADORES

Na corrida da alta tecnologia a força motora tem sido o computador e tudo a ele relacionado - Chips (micro-circuitos), robôs, comunicações. Por volta do ano 2.000 a indústria eletrônica, que já é um negócio de 300 bilhões de dólares anuais, deverá mais do que triplicar suas vendas para se tornar o maior do mundo, só superada pela agricultura. O país que dominá-la estará montado sobre a economia mundial como um Colosso do século XXI.

Cientistas americanos dos Laboratórios Bell foram os pioneiros da moderna era eletrônica, com a invenção do transistor em 1947. O Japão usou esta e descobertas subsequentes em produtos de consumo, mas sua grande força está na eficiente industrialização e não na

ciência básica. A Europa Ocidental tem obtido resultados excelentes tanto na criação quanto na fabricação de sistemas avançados para telecomunicações. A União Soviética chega a estar 10 anos atrás do Ocidente na maioria dos campos da eletrônica.

O controle dos mercados de computadores pelas empresas americanas parece garantido no futuro imediato. A IBM, com 60% das vendas mundiais da indústria, e talvez com até 70% de seus lucros operacionais, será difícil de derrotar. Os fabricantes de computadores da Europa Ocidental só contam com 10% das vendas nos mercados mundiais e o Japão é dono de outros 15%. Os soviéticos não vendem computadores no Ocidente; sua maior preocupação é acompanhar os progressos nesta área.

Contudo, por causa da miríade de elementos envolvidos na eletrônica computadorizada, não é fácil prever por quanto tempo continuará o domínio dos Estados Unidos no setor. Os chips são o elemento principal; na eletrônica são a chave de tudo, das entranhas de todos os tamanhos e formas de computador à eletrônica de consumo, onde a posição do Japão começou a ser desafiada pela Coreia do Sul e Formosa. Até este ano os fabricantes americanos de chips de computador lideravam as vendas mundiais. Segundo um respeitado analista da indústria, da Integrated Circuit Engineering, no fim deste ano os japoneses estarão em primeiro lugar. Três empresas japonesas, a NEC, a Hitachi e a Fujitsu, vão assumir a liderança, derrotando as americanas Motorola e Texas Instruments.

Os fabricantes americanos de chips de computador vão conservar a liderança na criação de microprocessadores, que funcionam como o cérebro de equipamentos eletrônicos como computadores pessoais e das grandes máquinas. Os Estados Unidos têm 43% do mercado mundial de 2,75 bilhões de dólares anuais dos microprocessadores; o Japão tem 34% e a Europa Ocidental 18%. Contudo as empresas americanas estão começando a perder posição para as japonesas no mercado de grande volume de componentes; como os populares microprocessadores 68000 e 68020 da Motorola, também fabricados no Japão mediante licença.

Empresas da Europa Ocidental lideram em algumas importantes subdivisões da tecnologia de semicondutores. Por exemplo, a subsidiária européia da ITT, a Standard Electric Lorenz, foi a primeira a desenvolver chips para o processamento digital de imagens de televisão, o que melhora muito a qualidade das imagens. A Siemens, da Alemanha Ocidental, é o maior fornecedor mundial de uma linha completa de componentes para as chamadas redes digitais de serviços integrados, que permitem que a voz, dados e imagens de vídeo sejam transmitidos pela mesma linha de comunicação. Jacques Ernest, diretor técnico da Alcatel, a maior empresa francesa de telecomunicações, coloca a Europa Ocidental e o Japão na frente dos Estados Unidos no desenvolvimento de redes digitais como a da Siemens. A Cie. Générale d'Electricité, matriz da Alcatel, em agosto passado concordou em comprar as subsidiárias da ITT na Europa por 2,5 bilhões de dólares. A nova empresa, Eurotel, será a segunda maior companhia de

telecomunicações do mundo, só superada pela AT&T.

Em um empreendimento conjunto com a Philips holandesa, a Siemens está tentando produzir um chip de memória excepcionalmente grande para manter a Europa na corrida dos semicondutores. A Philips e a Siemens entraram na empreitada com 400 milhões de dólares cada uma, e o governo da Holanda e o da Alemanha Ocidental com mais 400 milhões de dólares. Contudo, até agora em chips de memória e microprocessadores produzidos em massa, a Europa Ocidental está bem atrás do Japão e dos Estados Unidos, tendo apenas 21% do mercado mundial.

Pretendendo dominar este mercado, os japoneses venceram a primeira grande batalha. Agora controlam 67% do mercado mundial de 1,65 bilhões de dólares anuais em chips dinâmicos de acesso de memória ao acaso (DRAM). Os DRAMs estão para os computadores e outros equipamentos eletrônicos assim como o papel está para quem escreve. São os chips de memória mais amplamente empregados e mais baratos; e por serem fabricados em grandes quantidades também são um meio excelente para desenvolver novos processos e técnicas de produção.

O Japão agora é líder mundial no desenvolvimento do arseniureto de gálio, um material muito promissor para futuros computadores e redes de telecomunicações porque pode funcionar tanto com sinais eletrônicos quanto com sinais de luz. Os chips de arseniureto de gálio também podem processar informações quase 10 vezes mais rapidamente do que os de silício. Diversas empresas americanas, como IBM, AT&T e Vitesse Electronics, estão acelerando suas pesquisas com o arseniureto de gálio, e o Departamento de Defesa dos EUA tem um grande programa com este material, especialmente para o processamento de sinais. Na Europa, empresas como a Thompson, Philips e Plessey estão pesquisando o arseniureto de gálio e a União Soviética também está realizando alguns estudos.

Em computadores, os empreendimentos das novas companhias americanas e a formidável presença mundial da IBM até agora impediram que os japoneses se tornassem competidores ameaçadores. Com suas incessantes inovações, as pequenas empresas americanas têm mantido à distância seus rivais japoneses, menos adaptáveis às contingências dos mercados limitados.

Contudo, na computação está começando um novo estágio e muitos cientistas acreditam que a próxima etapa poderá beneficiar os japoneses. Este novo estágio é o processamento paralelo, que envolve um afastamento do processamento de dados linear, passando a um sistema onde um grande número de processadores - dezenas, centenas e até milhares - atacam um problema simultaneamente. O processamento paralelo vai reduzir o tempo necessário para armazenar e usar dados, além de aumentar a complexidade das tarefas científicas e comerciais desempenhadas pelos computadores.

Ao mesmo tempo, o processamento paralelo é o veículo perfeito para a inteligência artificial (IA) porque seus programas precisam de grande capacidade de computação. A forma mais completamente desenvolvida de IA usa a capacidade

dos especialistas e oferece-a, através de "sistemas especializados" computadorizados, para ajudar os menos hábeis; como por exemplo, guiar um empregado novo através do processo de julgamento do crédito de um cliente de um banco. O processamento paralelo está feito sob medida para os supercomputadores - as novas grandes máquinas que permitem proezas, da maior importância para a competição industrial, como testar dentro do computador a asa de um avião, com o fluxo de ar simulado por um vento eletrônico feito de fórmulas e equações. Um recente relatório da Comissão Nacional de Ciência, dos EUA, afirma que os computadores cada vez mais rápidos vão permitir aos cientistas simular e apresentar em suas telas a criação do universo, as entranhas desconhecidas das estrelas e até um pensamento passando pelo cérebro.

Na IA a grande pergunta é quem vai liderar o setor - os Estados Unidos? O Japão? Os outros possíveis concorrentes, os países da Europa Ocidental, ainda estão reunindo suas forças; e como no resto da eletrônica, a União Soviética está muito atrás. Até agora, no campo da inteligência artificial, supercomputadores e processamento paralelo, as empresas americanas estão liderando.

Para conseguir colocar mais camadas no mesmo espaço, os circuitos têm que se tornar muito mais delgados. No entanto, os métodos fotográficos convencionais praticamente chegaram a seus limites máximos de eficiência e as tentativas de substituição dessa técnica, como os raios-X, têm resultado em problemas, inclusive de custo. Com o novo processo, a empresa já chegou a fazer circuitos de 0,4 micra de largura. Os especialistas acreditam que até 1995 vão conseguir chegar a 0,1 micron, ou oitocentas vezes menos do que o diâmetro de um fio de cabelo, a largura necessária para se construir um chip de 64 megabits de memória. "O que é interessante é que esse processo permite o uso do mesmo tipo de equipamento atualmente empregado" afirma Bruno Roland, do departamento de pesquisas da UCB. Outra vantagem é que o processo reduz em cerca de 10% o número de chips que devem normalmente ser jogados fora devido a imperfeições de fabricação. O novo chip requer um único equipamento novo para tratar o revestimento com gás, antes de imprimir os sulcos dos circuitos eletrônicos. O protótipo, avaliado em US\$ 500 mil, fica pronto ainda em 1987, e está sendo desenvolvido pelo grupo que integra o projeto de tecnologia de ponto "Eureka" (a resposta européia para o programa "Guerra nas Estrelas" do presidente norte-americano Ronald Reagan). Quinze sofisticadas máquinas seriam necessárias, porém, para se utilizar o processo com um máximo de eficiência, a um custo total de US\$ 15 milhões.

A reação ao novo processo tem sido de entusiasmo, segundo Alain Douchamps, porta-voz da UCB. Apesar disso, ainda não há clientes na fila. O processo e o revestimento, chamado "plasmask", já foram patenteados e, segundo Douchamps, a UCB está negociando contratos com mais da metade das firmas de computação em todo o mundo. A UCB estima em vários milhões de dólares ao ano o valor das vendas. Os principais mercados serão os

Estados Unidos e o Japão. Se tudo correr como previsto, a UCB espera recuperar logo a verba de US\$ 10 milhões investida na pesquisa do novo processo. Os ganhos a curto prazo não são, porém, o único fator que está estimulando os pesquisadores da UCB. "Este é basicamente um produto do futuro, e esta é uma magnífica oportunidade para a Europa", afirma Roland.

A Europa Ocidental ocupa uma boa posição no desenvolvimento dos polímeros. Um progresso animador é a produção de um polímero biodegradável que emprega uma bactéria rara, descoberta por acaso em um canal da Alemanha Ocidental. O trabalho nas alternativas aos plásticos fabricados com base no petróleo começou na ICI, na Inglaterra, na década de 70. No início da década de 80 os pesquisadores já podiam produzir alguns litros de cada vez de um novo plástico, mas a queda dos preços do petróleo levou este mercado a ficar incerto. A ICI persistiu com as pesquisas, criou a Marlborough Bio-Polymer Ltd. e agora seu diretor John Adsetts afirma que esta subsidiária "pode produzir toneladas de plásticos em poucos dias". As bactérias são a base dos processos da Marlborough; ao simplesmente mudar a dieta delas - açúcar, cereais ou amido - elas (que chegam a produzir 80% de seu peso em polímeros) criam toda uma série de plásticos diferentes, todos eles únicos e patenteados.

Correndo numa pista paralela, o Japão, com seu Projeto Quinta Geração, pretende saltar à frente do resto do mundo em IA e processamento paralelo. O primeiro produto comercial a surgir deste programa é uma máquina pessoal de inferência seqüencial (PSI) feita pela Mitsubishi Electric, uma das companhias que participam do projeto. Em vez de simplesmente adicionar números, a PSI trabalha com inferências lógicas; trata-se de uma máquina compacta que cabe sobre uma mesa, destinada ao desenvolvimento de sistemas especializados. Até agora os especialistas americanos, que já construíram máquinas do mesmo tipo e mais potentes em programas de pesquisa universitária, não estão muito impressionados com os avanços, mas darão a mão à palmatória se os japoneses conseguirem em um ano, como programaram, aumentar 10 vezes a capacidade de suas PSI.

"Dentro de cinco anos a inferência lógica vai ocupar um segmento muito importante do mercado. Assim os japoneses terão uma grande vantagem porque praticamente nenhuma empresa americana está adotando este sistema na construção de computadores", afirma David Patterson, um cientista de computadores da Universidade da Califórnia.



# Compu Club



gravel

# ENTENDE A MELHOR

## O SEU

# ZX-SPECTRUM

## INTRODUÇÃO

Em termos de programação, a única realidade é a linguagem de máquina, que nos permite a comunicação direta com o cérebro do computador: o microprocessador. Através dele, podemos realizar uma série de ações alternativas que poderiam até mesmo controlar um robot com características humanas. Obviamente, não nos deixaremos ir tão longe, mas cremos que seria útil conhecer um pouco mais acerca dessa unidade de processamento, o que nos levaria a utilizar o nosso micro de modo mais eficiente.

Um microprocessador é um circuito integrado que controla o fluxo das correntes que chegam até ele. Na verdade, suas atividades são até muito limitadas, e compreendem:

- (1) A soma de números;
- (2) A comparação de números;
- (3) A memorização de números;
- (4) O controle sobre a próxima atividade a ser realizada.

Chega, pois, a ser surpreendente o quanto pode ser realizado com um computador, quando se considera que sua unidade de processamento é, basicamente, dotada apenas das características acima descritas.

A atividade de programação em linguagem de máquina consiste em colocar os números corretos na sequência correta, a fim de que o microprocessador possa trabalhá-los e, em consequência, possamos obter os efeitos desejados. Isto, porém, não é tão simples assim, e exatamente por causa das dificuldades que aí residem é que surgiram as linguagens de programação. Estas nos permitem expressar nossas intenções de um modo que, a nós, humanos, nos parece mais racional. O computador, entretanto, não entende nada disso, razão pela qual ele tem, no seu interior, uma unidade interpretadora que transforma nossas instruções em sequências de números que serão trabalhadas pelo microprocessador e transformadas em trabalho útil, segundo as nossas pretensões.

## A LINGUAGEM DE MÁQUINA

O microprocessador do ZX-SPECTRUM é o Z80A e, como vimos anteriormente, processa apenas unidades elementares de informação, que são números binários.

O sistema de numeração binário, ou de base dois, tem apenas dois símbolos (ou dígitos) básicos: o "0" e o "1", também chamados "bits", uma contração dos vocábulos ingleses "binary digit", cujo significado é dígito binário.

Tudo que o microprocessador faz, na verdade, é deixar, ou não, que impulsos elétricos passem através de minúsculos interruptores. Quando o impulso passa, simbolizamos isto pelo dígito 1, em caso contrário, pelo dígito 0.

Com 2 circuitos similares, teríamos, portanto, as seguintes possibilidades:

- (1) 00 - os dois interruptores desligados;
- (2) 01 - o 1º desligado e o 2º ligado;
- (3) 10 - o 1º ligado e o 2º desligado;
- (4) 11 - ambos ligados.

Seguindo esta linha de raciocínio, com 3 interruptores teremos 8 possibilidades, a saber: 000, 001, 010, 100, 011, 101, 110 e 111; com 4 interruptores teremos 16 possibilidades, etc. Enfim, o número de possibilidades é sempre uma função do número de interruptores. Mais precisamente, se temos  $n$  interruptores, o número de possibilidades será sempre  $2^n$ .

Em função do exposto, podemos deduzir que um microprocessador de 8 bits — como é o caso do Z80 — pode manipular números binários de 8 dígitos, i.e., qualquer combinação de oito 1's e 0's, como por exemplo 00110111, e é precisamente a qualquer formação desse tipo que damos o nome de **byte**.

E uma vez que o Z80 só pode manipular números binários constituídos de 8 bits, as unidades elementares de informação a serem processadas por este microprocessador terão que ser introduzidas na forma de bytes.

A linguagem de máquina, portanto, nada mais é do que uma sequência de bytes, e toda a dificuldade de programação nessa linguagem se resume em colocar o byte adequado no lugar correto. Nesse sentido, devemos distinguir entre as unidades elementares de informação destinadas a dar um tipo de ordem ao microcomputador, das que meramente são portadoras de dados que serão utilizados ao longo do processamento. Às primeiras, denominamos microinstruções.

A primeira de todas as microinstruções é o 0. Se no contexto de um programa quisermos ordenar ao micro que não execute absolutamente nada, escreveremos 00000000. O Z80, ao ler esta microinstrução, passará por ela sem executar coisa alguma.

Programar em linguagem de máquina é, portanto, escrever microinstruções e dados na forma de bytes, e isto de tal modo que, ao serem processadas, obtenhamos os resultados desejados.

## A LINGUAGEM ASSEMBLER

Você já deve estar percebendo que programar em linguagem de máquina é uma tarefa incrivelmente árdua. Por este motivo, o normal é escrever os programas em linguagem assembler, pois nessa linguagem cada instrução tem sua microinstrução equivalente em linguagem de máquina, mas esta não é um número, e sim um **mnemotécnico**.

Os mnemotécnicos são contrações de vocábulos ingleses que evidenciam a função de cada uma das microinstruções da linguagem de máquina. Os mnemotécnicos são conhecidos como **mnemônicos**.

Como dissemos anteriormente, um microcomputador só pode interpretar números binários, e isto significa que um programa escrito em linguagem assembler necessita ser transcrito para a linguagem de máquina, a fim de que possa ser processado. Esta transcrição é feita pelo próprio programador, com auxílio das tabelas de mnemônicos que podem ser encontradas nos manuais do seu equipamento e que nos permitem por em forma numérica as instruções que utilizamos na linguagem assembler. Os dados deverão estar na mesma base numérica que utilizamos para as microinstruções.

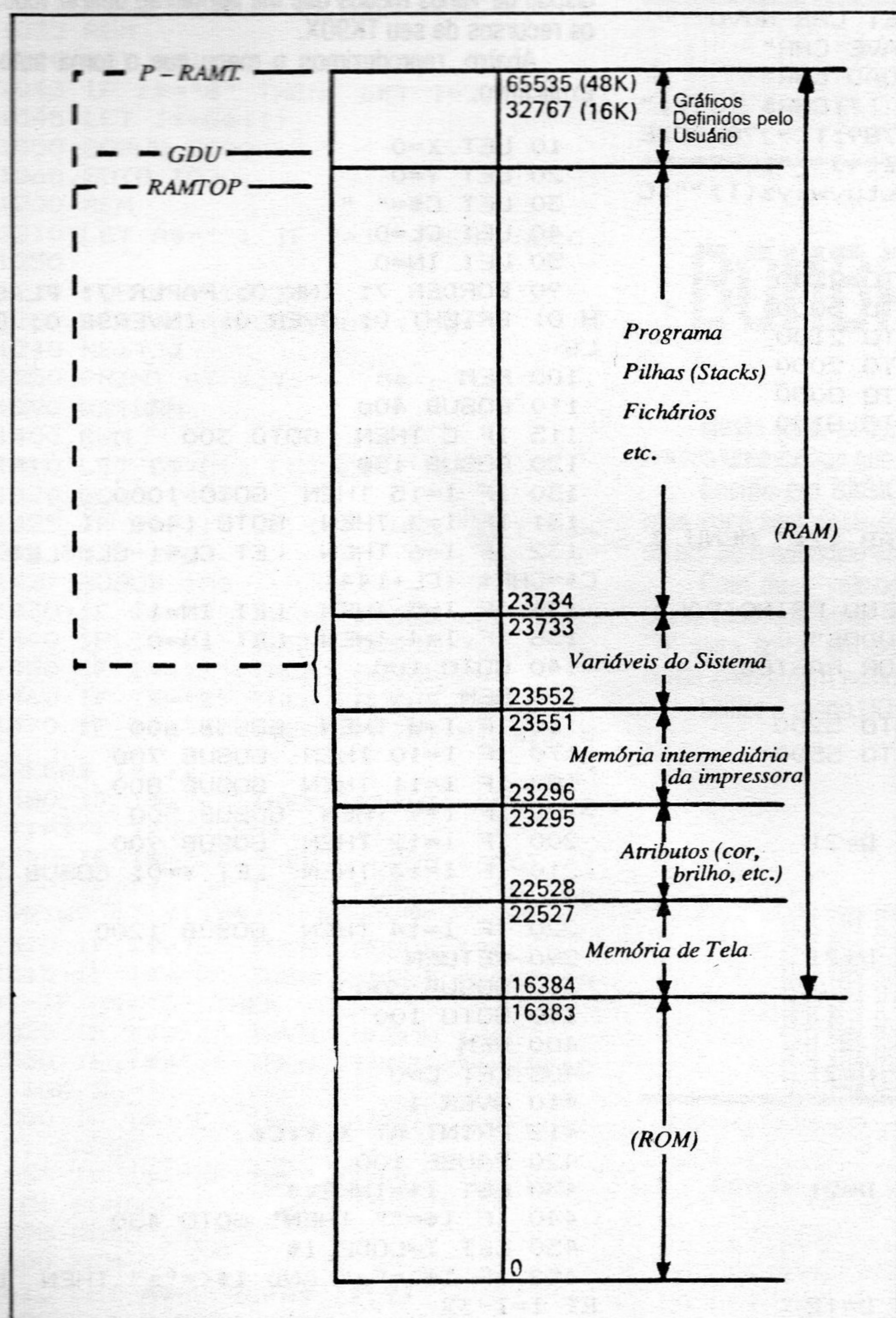
Há, entretanto, um meio mais prático e eficiente de executar esta transcrição, utilizando-se um programa **assemblador** que, além de executar essa transcrição, nos possibilita armazenar nosso programa na região mais adequada da memória do microcomputador, a fim de que possamos utilizá-lo como uma sub-rotina de outros programas escritos em BASIC.

## ESTRUTURA E ORGANIZAÇÃO DE MEMÓRIA

Cada posição de memória é um lugar em que se pode armazenar informações e é comumente conhecida por ENDEREÇO.

Esta terminologia não foi criada por acaso, mas porque se trata da utilização de um conceito que já nos é familiar. A cada uma dessas posições encontra-se atribuído um número entre 0 e 65535.

Desse modo, o número nos fornece o endereço específico da posição correspondente. Basta olhar o mapa da memória, logo abaixo, para observarmos que os primeiros 16384 (ou 16K) endereços estão ocupados pela ROM. Esta parte da memória, como sabemos, é uma memória apenas de leitura (*Read Only Memory*) que, em essência, é um bloco de 16384 posições, cujas informações ali contidas foram armazenadas no momento da fabricação do "chip" da ROM. Tais informações não podem, portanto, ser alteradas pelo usuário. Na versão do ZX-SPECTRUM de 16k há, ainda, outras 16384 posições disponíveis para a RAM (*Random Access Memory*), a parte da memória em que nós podemos trabalhar, armazenando informações e alterando seu conteúdo. Já a versão de 48K do ZX-SPECTRUM utiliza o número máximo de posições que podem ser endereçadas pela CPU do Z80. Assim, além dos 16K de ROM, ele possui 48K de RAM, posições estas que recebem endereços que vão desde 32767 até 65535. A combinação dos 16K de ROM com os 48K de RAM nos dão um total de 64K de memória, ou 65536 bytes.



## A MEMÓRIA : UM EDIFÍCIO DE "CAIXINHAS"

Agora você já deve ter percebido que a CPU do ZX-SPECTRUM pode manipular 256 bytes diferentes, de 8 bits. Consideremos, pois, que a memória é, hipoteticamente, como um edifício de 256 andares, com 256 "caixinhas" por andar, e que cada "caixinha" tem um número, de acordo com o andar em que se encontra e com a sua posição nesse andar. Isto nos leva a um total de 65536 "caixinhas", numeradas de 0 a 65535. Se numeramos os andares de 0 a 255 e, em cada andar, numeramos também as "caixinhas" de 0 a 255, teremos um meio de nos referirmos a qualquer dessas "caixinhas", sem ambigüidade de sentido. Assim, uma "caixinha" fica

perfeitamente determinada quando se menciona o andar em que ela se encontra e seu número nesse andar.

Além disso, fica automaticamente atribuído um número entre 0 e 65535, para cada "caixinha". Por exemplo, se a "caixinha" se encontra no 60º andar, e seu número nesse andar é 129, seu número de ordem, ou endereço, será:  $(60 \times 256) + 129 = 15.489$ .

Para manipular cada um desses endereços, o ZX-SPECTRUM utiliza dois bytes. O primeiro armazena a parte menos significativa, i.e., o número da "caixinha" no andar em que ela se encontra; e o segundo armazena a parte mais significativa, i.e., o número do andar.

## QUANTA MEMÓRIA OCUPA UM PROGRAMA?

Um bom exemplo do que dissemos, e de grande utilidade, é a fórmula que apresentamos a seguir, e que nos permite calcular o número de bytes que um programa em BASIC ocupa na memória do seu micro. Carregue o seu micro com um programa em BASIC, digite a seguinte instrução e pressione a tecla ENTER.

```
PRINT(PEEK 23653 + (PEEK 23654) * 256)
      - (PEEK 23635 + (PEEK 23636) * 256)
```

Tudo que esta fórmula faz é simplesmente investigar o ponto em que tem início a área desocupada da memória (STKEND) e então subtrair o endereço inicial do programa escrito em BASIC. Tais endereços são facilmente encontrados "PEEKando-se" as variáveis do sistema que aparecem na fórmula acima apresentada. Cada endereço encontra-se armazenado em dois bytes de memória, sendo o primeiro o LSB (*Least Significant Byte*), i.e., o byte menos significativo. O primeiro endereço, portanto, é encontrado somando-se o conteúdo do endereço 23653 ao conteúdo do endereço seguinte multiplicado por 16. De modo análogo, o segundo endereço, denominado MSB (*Most Significant Byte*), i.e., o byte mais significativo, se encontra somando-se o conteúdo do endereço 23635 ao conteúdo do endereço seguinte multiplicado por 256. Assim, a diferença entre esses dois valores é exatamente o número de bytes que o programa em BASIC ocupa na memória do seu micro.

## PEEK & POKE

Examinando a fórmula contida seção anterior, encontramos a palavra PEEK, que em inglês significa "olhadela", estando, portanto, em perfeito acordo com a função por ela desempenhada. PEEK nos permite olhar o interior de um endereço e ver o seu conteúdo. Na verdade, isto é tudo o que podemos fazer quando se trata de um endereço na ROM do micro; *olhar, mas não tocar*. Já em um endereço da RAM, podemos fazer ambas as coisas: olhar e tocar, com a intenção de alterar esse conteúdo. Para proceder a essa alteração, entretanto, temos que utilizar o comando POKE. É "POKEando-se" um determinado endereço que alteramos o seu conteúdo. Para isto, basta digitar a instrução - POKE n,x - onde "n" é o endereço desejado e "x" o novo conteúdo, e pressionar a tecla ENTER.

Neste momento, nós o aconselhamos a praticar algumas alterações de conteúdo de endereços. Primeiramente, utilize a função PEEK para descobrir o conteúdo de um determinado endereço. Depois, utilize o comando POKE para efetuar a alteração, e novamente a função PEEK para verificar que a alteração foi mesmo efetuada. Tente, também, com um endereço da ROM. Pode fazer isso sem medo! Na verdade, a ROM está protegida e não será permitida qualquer alteração no conteúdo de um de seus endereços. Não fosse isso, a maioria dos micros, nas mãos de leigos, duraria muito pouco tempo, pois é o conteúdo da ROM que habilita o micro a desempenhar todas as funções descritas em seu manual de operações.

Mas até aí, tudo bem! Você já consegue alterar o conteúdo de certos endereços. Agora, é possível que você esteja bastante curioso acerca do tipo de conteúdo que você deve colocar nos endereços da RAM para obter um determinado efeito. Bem, essa é a parte mais difícil e, para ficar craque mesmo, talvez seja melhor comprar um bom livro de assembler e queimar um pouco de fosfato tentando entendê-lo. Nessa seção, nós daremos algumas dicas a esse respeito, mas desde já o advertimos que, para conhecer em profundidade os mistérios do seu micro, o caminho é longo e árduo.

De certo modo, entretanto, estamos cada vez mais próximos do momento em que estaremos preparados para falar com o ZX-SPECTRUM em seu próprio idioma. O que nos falta é exatamente saber que palavras podem ser entendidas e processadas.

Antes, porém, devemos nos ocupar da execução de rotinas em código de máquina e, antes ainda, de como introduzir rotinas em código de máquina na memória do seu micro. Daí, certamente tiraremos alguns subsídios que nos permitirão compreender cada vez mais essa caixinha de segredos que é o ZX-SPECTRUM.

Mas isto fica para uma próxima oportunidade. Até lá.

# DEFINIDOR DE CHR\$

O definidor de caracteres que agora apresentamos é algo realmente sensacional. Você pode definir os caracteres que serão utilizados pelo seu micro. Veja, por exemplo, o menu do programa com os novos caracteres que nós criamos, a lista completa de caracteres, e três gráficos UDG.

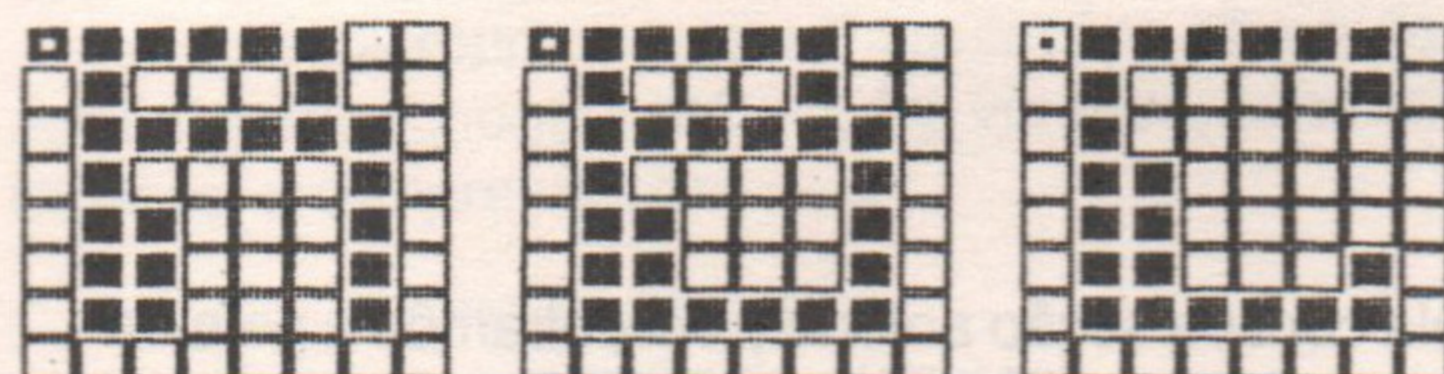
Para inicializar, utilize a opção 1. Nesse ponto, com a opção 2 você pode criar seus próprios caracteres. Ao utilizá-la, o micro lhe perguntará se você deseja recriar todos ou só alguns. A partir daí, ele deslocará os caracteres para o modo UDG, permitindo, assim, que use sua criatividade. Seu micro, então, poderá, sempre que você quiser, operar com a marca da sua personalidade.

## MENU PRINCIPAL

- 0 - SAÍ
- 1 - COPIA CHR ORIGINAL
- 2 - CRIAR CHR NOVO
- 3 - SET CHR ORIGINAL
- 4 - SET CHR NOVO
- 5 - SAVE CHR
- 6 - LOAD CHR

```

! " # $ % & ' ( ) * + , - . / 0 1 2 3 4
5 6 7 8 9 : ; < = > ? @ A B C D E F G H I
J K L M N O P Q R S T U V W X Y Z [ \ ] ^ _
` a b c d e f g h i j k l m n o p q r s
t u v w x y z { } ~ Δ
    
```



```

1 REM
2 REM *****
3 REM *
4 REM *ADALBERD F. GUIMARAES*
5 REM *
6 REM * definidor de chr$ *
7 REM *
8 REM *****
9 REM
10 GOTO 4000
100 REM ** UDG
110 CLS
120 FOR J=0 TO 95
130 IF J/21=INT (J/21) THEN PRINT
TAB 11;
140 PRINT CHR$ (J+32);
150 NEXT J
1602
190 RETURN
1000 REM ** COPIA BLOCO
1005 CLS : PRINT AT 10,10;" ESPERE
"
1010 FOR J=0 TO L-1
1020 POKE (E1+J),PEEK (E2+J)
1030 NEXT J
1090 RETURN
2000 REM ** SET CHR NOVO
2010 LET E=64000
2020 LET A1=INT (E/256)
2030 LET A2=E-A1*256
2040 POKE 23606,A2: POKE 23607,A1-1
2090 GOTO 4000
2100 REM ** SET CHR ORIGINAL
2110 LET E=15616
2120 GOTO 2020
2200 REM ** COPIA CHR ORIGINAL
2210 LET E1=64000
2220 LET E2=15616
2230 LET L=8*96
2240 GOSUB 1000
2290 GOTO 4000
    
```

```

2300 REM ** COPIA UM BLOCO NOVO
2310 LET E1=USR "A"
2320 LET E2=64000+(CODE D$-32)*8
2330 LET L=8*D
2340 GOTO 1000
2401 REM ** GRAVA UM BLOCO
2410 LET E1=64000+(CODE D$-32)*8
2420 LET E2=USR "A"
2430 LET L=8*D
2440 GOTO 1000
2450 GOTO 4000
2500 REM ** LIMPA BUFFER
2505 PRINT AT 10,10;" ESPERE "
2510 FOR J=0 TO 21*8-1
2520 POKE USR "A"+J,0
2530 NEXT J
2590 RETURN
4000 REM ** MENU PRINCIPAL
4010 CLS
4020 PRINT : PRINT TAB 8;" MENU PRI
NCIPAL "
4024 PRINT " 0 - SAÍ"
4030 PRINT " 1 - COPIA CHR ORIGIN
AL"
4040 PRINT " 2 - CRIAR CHR NOVO"
4050 PRINT " 3 - SET CHR ORIGINAL
"
4060 PRINT " 4 - SET CHR NOVO"
4070 PRINT " 5 - SAVE CHR"
4080 PRINT " 6 - LOAD CHR"
4090 PRINT AT 19,0;" !";CHR$ (34);"
##%&'()*+,-./0123456789:;<=>?@ABCDE
FGHIJKLMNOPQRSTUVWXYZ[\]^_";CHR$ 96
;"abcdefghijklmnopqrstuvwxy{ }~";C
HR$ 127
4100 INPUT N
4120 IF N=1 THEN GOTO 2200
4130 IF N=2 THEN GOTO 5000
4140 IF N=3 THEN GOTO 2100
4150 IF N=4 THEN GOTO 2000
4160 IF N=5 THEN GOTO 8000
4170 IF N=6 THEN GOTO 8100
4190 STOP
5000 REM ** MENU 2
5010 CLS
5020 PRINT : PRINT TAB 12;" MENU 2
"
5025 PRINT " 0 - MENU PRINCIPAL"
5030 PRINT " 1 - TODOS"
5040 PRINT " 2 - POR PARTES"
5100 INPUT N
5120 IF N=1 THEN GOTO 5200
5130 IF N=2 THEN GOTO 5500
5190 GOTO 4000
5200 REM ** DEFLAUT
5210 LET D$=" " : LET D=21
5220 GOSUB 2300
5230 GOSUB 100
5240 GOSUB 2400
5250 LET D$="5" : LET D=21
5260 GOSUB 2300
5270 GOSUB 100
5280 GOSUB 2400
5300 LET D$="J" : LET D=21
5310 GOSUB 2300
5320 GOSUB 100
5330 GOSUB 2400
5350 LET D$="_" : LET D=21
5360 GOSUB 2300
5370 GOSUB 100
5380 GOSUB 2400
5400 LET D$="t" : LET D=12
5410 GOSUB 2500
5420 GOSUB 2300
5430 GOSUB 100
5440 GOSUB 2400
5450 GOTO 4000
5500 REM ** POR PARTES
5510 CLS
5520 INPUT "CHR INICIAL = ";D$
5530 IF D$="" THEN GOTO 5520
5540 PRINT AT 4,5;"CHR INICIAL = ";
D$
5550 INPUT "NUMERO DE CHR (MAX=21)
= ";D
5560 IF D<1 OR D>21 THEN GOTO 5550
5570 PRINT AT 6,5;"NUMERO DE CHR =
    
```

```

";D
5580 GOSUB 2500
5590 GOSUB 2300
5600 GOSUB 100
5610 GOSUB 2400
5690 GOTO 4000
8000 REM ** SAVE CHR
8010 INPUT "NOME PARA GRAVAR=";N$
8020 IF N$="" THEN GOTO 8010
8030 SAVE N$CODE 64000,96*8
8090 GOTO 4000
8100 REM ** LOAD CHR
8110 INPUT "NOME PARA LER=";N$
8120 LOAD N$CODE
8190 GOTO 4000
9000 SAVE "CHR$" LINE 4000
9090 STOP
    
```

# # EDITOR #

Aqui, apresentamos um EDITOR muito interessante, que vale à pena digitar pela sua utilidade. Você dispõe de vários modos que lhe permitirão utilizar todos os recursos de seu TK90X.

Abaixo, reproduzimos o menu que o torna auto-explicativo.

```

10 LET X=0
20 LET Y=0
30 LET C$=" "
40 LET CL=0
50 LET IN=0
90 BORDER 7: INK 0: PAPER 7: FLAS
H 0: BRIGHT 0: OVER 0: INVERSE 0: C
LS
100 REM
110 GOSUB 400
115 IF C THEN GOTO 300
120 GOSUB 150
130 IF I=15 THEN GOTO 1000
131 IF I=7 THEN GOTO 1400
132 IF I=6 THEN LET CL=1-CL: LET
C$=CHR$ (CL+144)
136 IF I=5 THEN LET IN=1
138 IF I=4 THEN LET IN=0
140 GOTO 100
150 REM
160 IF I=8 THEN GOSUB 600
170 IF I=10 THEN GOSUB 700
180 IF I=11 THEN GOSUB 800
190 IF I=9 THEN GOSUB 500
200 IF I=12 THEN GOSUB 900
210 IF I=13 THEN LET Y=0: GOSUB 7
00
220 IF I=14 THEN GOSUB 1200
290 RETURN
300 GOSUB 3000
390 GOTO 100
400 REM
405 LET C=0
410 OVER 1
412 PRINT AT X,Y;C$
420 PAUSE 100
430 LET I#=INKEY$
440 IF I#="" THEN GOTO 430
450 LET I=CODE I#
452 IF I#>="a" AND I#<="z" THEN L
ET I=I-32
454 IF CL AND I>=65 AND I<=90 THEN
LET I=I+32
456 LET I#=CHR$ I
460 IF I>31 THEN LET C=1
470 PRINT AT X,Y;C$
480 OVER 0
490 RETURN
500 REM
510 LET Y=Y+1
530 IF Y>31 THEN LET Y=0: GOSUB 7
00
590 RETURN
600 REM
610 LET Y=Y-1
620 IF Y<0 THEN LET Y=31: GOSUB 8
    
```



```

00
690 RETURN
700 REM
710 LET X=X+1
720 IF X>21 THEN LET X=21: LET Y=
31
790 RETURN
800 REM
810 LET X=X-1
820 IF X<0 THEN LET X=0: LET Y=0
890 RETURN
900 REM
910 OVER 0
930 GOSUB 600
940 PRINT AT X,Y;" "
990 RETURN
1000 REM
1005 LET G$=""
1010 LET D$=C$: LET C$=""
1020 GOSUB 400
1022 GOSUB 150
1023 IF I$>="A" AND I$<="U" THEN L
ET I$=CHR$(79+I): GOTO 1050
1025 IF I=15 THEN LET C#=D$: GOTO
100
1027 IF I$="@ " THEN GOTO 1032
1030 IF I$>="1" AND I$<="8" OR I$>=
"! " AND I$<="(" THEN GOTO 1032
1031 GOTO 1020
1032 REM
1040 LET I=I-32-(I>=49)*8
1043 IF I$="@ " THEN LET I=2
1045 LET I#=G$(I)
1050 GOSUB 3000
1060 GOTO 1020
1200 REM
1210 LET A$="": IF Y=31 THEN GOTO
1250
1220 FOR J=Y TO 30
1230 LET A#=A#+SCREEN$(X,J)
1240 NEXT J
1250 PRINT AT X,Y;" "+A$
1290 RETURN
1400 REM
1410 LET D$=C$: LET C$=""
1420 GOSUB 400
1425 IF I$="1" OR I=7 THEN LET C#=
D$: GOTO 100
1427 GOSUB 150
1430 IF I$="5" THEN GOSUB 600
1440 IF I$="6" THEN GOSUB 700
1450 IF I$="7" THEN GOSUB 800
1460 IF I$="8" THEN GOSUB 500
1470 IF I$="3" THEN PRINT AT X,Y;S
CREEN$(X,Y)
1480 IF I$="4" THEN PRINT INVERSE
1;AT X,Y;SCREEN$(X,Y)
1490 IF I$="2" THEN GOSUB 1600
1500 IF I$="0" THEN FOR J=Y TO 31:
PRINT AT X,J;" ": NEXT J
1510 IF I$="S" THEN GOSUB 1700
1515 IF I$="V" THEN INPUT "CLS?";A
$: IF A$="S" THEN CLS
1520 IF I$="J" THEN GOSUB 1800
1530 IF I$="X" THEN INPUT "INK=";C
: INK C
1540 IF I$="C" THEN INPUT "PAPER="
;C: PAPER C
1550 IF I$="B" THEN INPUT "BORDER="
;C: BORDER C
1590 GOTO 1420
1600 REM
1610 LET A#=SCREEN$(X,Y)
1620 LET A=CODE A$
1625 IF A=0 THEN GOTO 1690
1630 IF A>=65 AND A<=90 THEN LET A
=A+32: GOTO 1650
1640 IF A>=97 AND A<=122 THEN LET
A=A-32
1650 PRINT AT X,Y;CHR$ A
1690 RETURN
1700 REM
1710 PRINT AT 21,0;
1720 INPUT "NOME=";N$
1725 IF N$="" THEN GOTO 1720
1730 SAVE N$ SCREEN$
1790 RETURN

```

```

1800 REM
1805 INPUT "LOAD";N$
1810 PRINT AT 21,0;
1820 LOAD "" SCREEN$
1890 RETURN
2000 REM
2010 DATA 255,255,255,255,255,255,2
55,255
2020 DATA 0,255,255,255,255,255,255
,0
2030 DATA 255,0,0,0,0,0,0,255
2040 DATA 195,129,0,0,0,0,129,195
2050 FOR J=0 TO 8*4-1
2060 READ A: POKE USR "A"+J,A
2070 NEXT J
2090 STOP
3000 REM
3020 IF I>164 THEN GOTO 3090
3050 PRINT INVERSE IN;AT X,Y;I$
3060 GOSUB 500
3090 RETURN

```

<b>MODO PRINT:</b> * LETRAS...A-Z * NUMEROS...0-9 * SIMBOLOS * EDIT * GRAPHICS * CAPS LOCK * TRUE VIDEO * INV. VIDEO	<b>MODO GRAPHICS:</b> * CARACTERES * ESPECIAIS...UDG * CARACTERES * GRAFICOS...L,ETC  * CS=CAPS SHIFT * SS=SYMBOL SH.
<b>MODO EDIT:</b> * CAPS LOCK...00 * TRUE VIDEO...00 * INV. VIDEO...4-8 * SETAS SEM CS...00-8 * APAGA LINHA...00 * SAVE SCREEN\$.L * LOAD SCREEN\$.L * CLS...C	<b>MODO GERAL:</b> * SETAS * CS+S-8 * DELETE * CS+0 * INSERE * CS+SS * PROX LINHA * ENTER

# BOMBARDEIRO

Neste jogo, pilotando um bombardeiro, você sobrevoa uma cidade que deve ser destruída. Escrito em BASIC, seu forte não é a velocidade, mas você será destruído muitas vezes, antes que possa ser bem sucedido em uma missão. Com duas velocidades e vários tamanhos de cidade, tudo que você tem a fazer é pressionar a tecla "0" para soltar um projétil de cada vez. Mas se você colidir com algum edifício, você perde. BOA SORTE! Utilize o modo UDG para a criação dos gráficos.

```

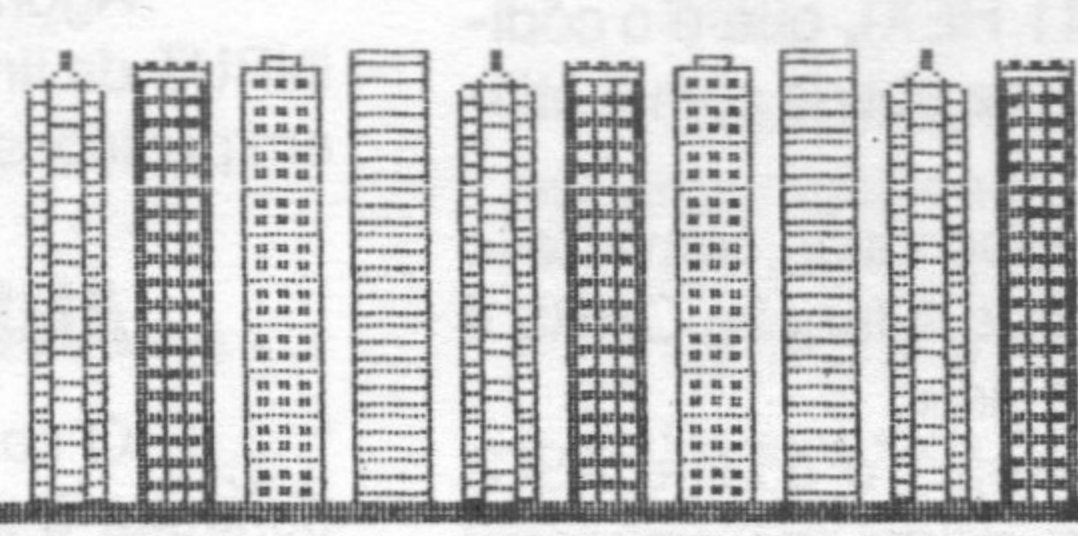
1 REM * ADALBERO F. GUIMARAES
*
5 LET R=0
10 LET X=0
20 LET Y=0
30 LET T=1
50 LET A$=""
60 LET B$=""
70 LET C$=""
80 LET P=0
82 INPUT "TAMANHO (1-9)";TT
84 IF TT<1 OR TT>9 THEN GOTO 82
90 INPUT "VELOCIDADE (1/2)";NN
92 IF NN<1 OR NN>2 THEN GOTO 90
95 LET N$="": IF NN=2 THEN LET N
$=""

```

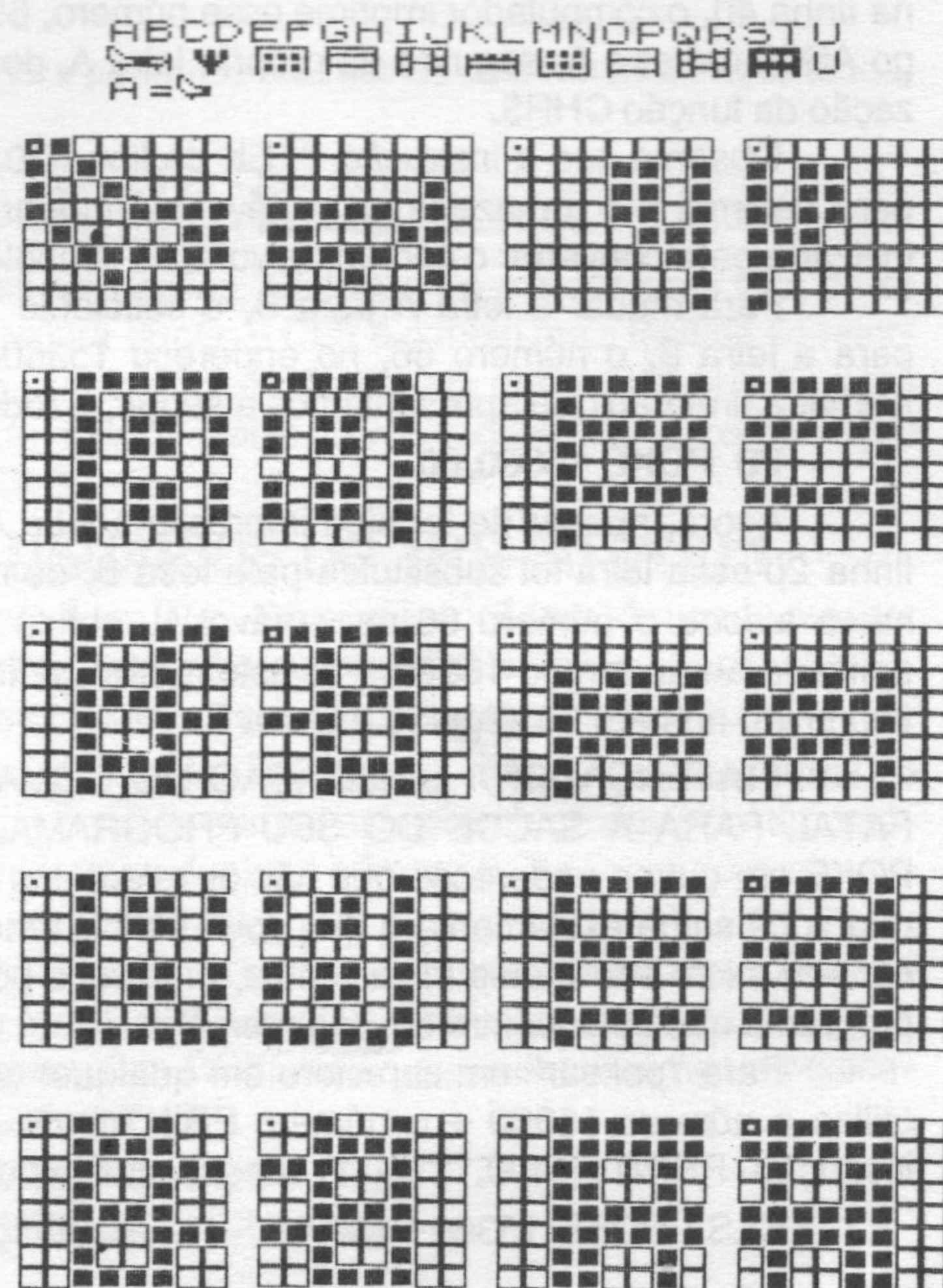
```

100 PRINT AT 19-TT,0;A$;
110 FOR J=1 TO TT
120 PRINT B$;
130 NEXT J
140 PRINT C$
200 LET X2=0: LET Y2=0
290 PRINT AT 21,1;"PONOTOS=";P;TAB
15;"RECORD=";R
1000 REM
1005 IF SCREEN$(X,Y+2)="" THEN GO
TO 2000
1010 PRINT AT X,Y;N$;" "
1015 GOSUB 1200
1020 LET Y=Y+NN
1030 IF Y>29 THEN LET X=X+1: LET Y
=0: PRINT AT X-1,30;" "
1040 IF X=19 AND Y=31-1-NN THEN GO
TO 2500
1090 GOTO 1000
1200 REM
1210 LET X#=INKEY$
1230 IF X$="0" AND T THEN LET T=0:
LET X1=X+1: LET Y1=Y+1
1280 GOSUB 1400
1290 RETURN
1400 REM
1405 PRINT AT X2-1,Y2;" "
1410 IF T THEN FOR J=1 TO 5: NEXT
J: GOTO 1490
1412 IF SCREEN$(X1,Y1+1)="" OR SCR
EEN$(X1,Y1)="" THEN LET T=1: LET
P=P+1: PRINT AT 21,9;P
1420 PRINT AT X1,Y1;" "
1430 LET X1=X1+1
1435 LET X2=X1: LET Y2=Y1
1450 IF X1>19 THEN LET T=1
1490 RETURN
2000 REM
2010 PRINT AT 10,10; FLASH 1;" BATE
U !! "
2015 PRINT AT 14,10; INVERSE 1;" AP
ERTE R "
2020 IF INKEY$<>"R" AND INKEY$<>"r"
THEN GOTO 2020
2030 IF P>R THEN LET R=P
2040 CLS
2090 GOTO 10
2500 REM
2510 PRINT AT 10,10; FLASH 1;" PARA
BENS ";AT 12,8;" VOCE VENCEU "
2590 GOTO 2015
9000 SAVE "AVIA02" LINE 9100
9010 SAVE "ED"CODE USR "A",180
9090 STOP
9100 PRINT FLASH 1;"ESPERE UM POUO
0"
9110 LOAD "ED"CODE
9190 RUN

```



PONOTOS=0 RECORD=10



# PEEK E POKE

## NO

# TRS-80 MOD III

## INTRODUÇÃO

Os métodos mais conhecidos para a criação de gráficos na tela envolvem os comandos SET, RESET, PRINT e PRINT @. Há, porém, duas outras instruções extraordinariamente úteis para esse propósito: estamos falando das instruções PEEK e POKE. Neste artigo, nos deteremos na avaliação destas instruções, estudando suas particularidades relacionadas com a geração de gráficos no vídeo do seu micro.

Tais instruções nos permitem, em princípio, examinar e modificar o conteúdo de endereços da memória do micro, sem que seja necessário, para essa finalidade, sair do BASIC. A instrução PEEK (do inglês, "espreitadela"), é como uma janela furtiva para o interior do seu micro. Ele lhe permite olhar o conteúdo de toda a memória do micro, incluindo a ROM. Já o comando POKE nos permite alterar o conteúdo da RAM, que inclui os endereços desde 15360 (3C00 HEX) até o final da memória. Utilizando PEEK e POKE, podemos escrever programas em linguagem de máquina, modificar a apresentação da tela, monitorar o uso de periféricos, mudar conteúdo de variáveis, etc. Para sermos mais objetivos, utilizando PEEK e POKE, podemos realizar qualquer tipo de trabalho possível no micro, se realmente a isso nós nos dispusermos. O extraordinário acerca dessas instruções é que elas nos permitem fazer tudo isso sem que seja necessário sair do BASIC.

## O VÍDEO NA MEMÓRIA DO TRS-80

Mas você talvez esteja se perguntando o que é que a possibilidade de modificar o conteúdo de endereços tem a ver com gráficos. Bem, nós vamos lhe mostrar que são assuntos intimamente relacionados.

Os 1024 caracteres possíveis no vídeo do TRS-80 são armazenados nos endereços 15360 - 16383 (3C00 - 3FFF HEX). O endereço 15360 corresponde à posição 0, e o endereço 16383, à posição 1023. Qualquer alteração que você faça nesta área da memória se reflete imediatamente no vídeo do seu micro. Digite o seguinte programa e observe como isto ocorre:

```
NEW
10 CLS : PRINT "A"
30 N = PEEK(15360)
40 PRINT N,CHR$(N)
RUN
```

A instrução PRINT da linha 10 imprime a letra A na posição 0; em seguida, na linha 30, o conteúdo do endereço 15360 é colocado na variável N; na linha 40, o computador imprime esse número, 65 (41 HEX), que é o código ASCII da letra A, seguido da própria letra A, desta vez obtida com a utilização da função CHR\$.

Observe que a instrução PEEK da linha 30 lê esse valor, converte-o para decimal e o armazena na variável N, enquanto que a função CHR\$ é utilizada para converter o código de volta ao símbolo original.

Para mudar a letra A para B, é suficiente "pokear" o código ASCII para a letra B, o número 66, no endereço 15360. Introduza, no programa acima, a linha 20 que apresentamos a seguir, e rode-o novamente.

```
20 POKE 15360,66
```

Agora, apesar de ter sido impressa a letra A através da linha 10, na linha 20 essa letra foi substituída pela letra B, de modo que, na linha 30, o micro alocou o número 66 na variável N, obtido através da função PEEK aplicada ao endereço 15360. Por este motivo, a linha 40 se encarregou de imprimir o número 66 seguido da letra B.

**!!!ATENÇÃO!!!** A UTILIZAÇÃO DO COMANDO POKE PODE SER FATAL PARA A SAÚDE DO SEU PROGRAMA. Não utilize o comando POKE em outros endereços que não os referentes à área de vídeo, a menos que você saiba exatamente o que você está fazendo. As chances de danificar seu micro são quase inexistentes, mas você pode facilmente perder um programa que se encontre em sua memória.

Para "pokear" um caractere em qualquer um dos endereços de tela, utilize o número 15360 + o número PRINT@ do local desejado. Assim, a instrução PRINT@ 400,"\*" pode ser substituída por:

```
CLS : POKE 15360+400,42 (42 é o código ASCII do símbolo *)
```

## POKE X PRINT

Ainda que o comando POKE possa ser utilizado para alocar no vídeo do seu micro muitos dos caracteres que também podem ser ali impressos por meio da instrução PRINT, ele reage de modo bem diferente quando aplicado a certos códigos ASCII. O próximo programa que apresentamos nos habilita a "pokear" diferentes códigos ASCII no vídeo do TRS-80. O comando POKE controla a posição na tela.

```
NEW
10 CLS : S=15360
20 FOR I=129 TO 191
30 PRINT CHR$(28);
40 INPUT N
50 POKE S+I,N
60 NEXT I
RUN
```

Agora, experimente digitar números entre 0 e 255 em resposta ao INPUT da linha 40, e este programa "pokeará" os símbolos correspondentes em posições consecutivas da tela.

## ANIMAÇÃO

O comando POKE pode, também, ser utilizado como alternativa para controlar a animação de figuras, mas seu uso é muito desajeitado se ele tem que ser aplicado a uma quantidade muito grande de dados muito rapidamente. A título de exemplo, apresentamos, a seguir, dois programas que têm o mesmo efeito: o de produzir animação em um caramujo desenhado na tela. O primeiro utiliza conceitos de empacotamento string; e o segundo procura simular o mesmo efeito com a utilização do comando POKE.

```
NEW
10 CLS : FOR I=1 TO 22
20 READ N : A$=A$+CHR$(N)
30 NEXT I
40 FOR P=0 TO 50
50 PRINT@ P,A$;
60 REM
70 NEXT P
80 DATA 128,160,158,163,143,167,180,128,160,182,246
90 DATA 176,178,189,176,179,184,191,184,191,135,128
RUN
```

E agora o mesmo efeito utilizando o comando POKE:

## NEW

```
10 DEFINT X : CLS
20 FOR X = 15360 TO 15410
30 POKE X,32 : POKE X+1,164
40 POKE X+2,176 : POKE X+3,176
50 POKE X+4,176 : POKE X+5,141
60 POKE X+6,140 : POKE X+64,32
70 POKE X+65,138 : POKE X+66,131
80 POKE X+67,131 : POKE X+68,139
90 NEXT
100 GOTO 100
RUN
```

Observamos, então, que o comando POKE consegue realizar o mesmo trabalho, mas que para figuras muito grandes, a velocidade de animação diminui consideravelmente.

## NÚMEROS GIGANTES

Se a velocidade não é o elemento crucial, o comando POKE pode ser utilizado para a criação de figuras grandes, sem utilizar o conceito de empacotamento string. Apresentamos, a seguir, uma sub-rotina que imprime um "oito" em tamanho grande na tela. Você poderia, então, criar uma sub-rotina completa, baseada em nosso trabalho, que permitisse a impressão de um número ou letra qualquer, em tamanho grande, e utilizá-la em seus programas.

## NEW

```
10 REM PROGRAMA PRINCIPAL
20 CLS : I=15840
30 GOSUB 1800
40 REM CONTINUE
50 GOTO 50
1800 POKE I,160 : POKE I+1,158 : POKE I+2,131 :
    POKE I+3,131 : POKE I+4,139 : POKE I+5,180
1810 POKE I+64,128 : POKE I+65,179 : POKE I+66,140 :
    POKE I+67,140 : POKE I+68,166 : POKE I+69,145
1820 POKE I+128,130 : POKE I+129,173 : POKE I+130,176 :
    POKE I+131,176 : POKE I+132,184 : POKE I+133,135
1830 RETURN
RUN
```

Utilizando o conceito de strings para imprimir a mesma figura, utilizamos menos códigos. A principal vantagem da utilização do comando POKE, entretanto, é que ele não faz uso do string "espaço".

## MOVIMENTO DO CURSOR

É muito fácil não perceber que o comando POKE não tem qualquer efeito sobre a posição do cursor. O cursor fica na última posição impressa com o uso do comando PRINT, e não com o comando POKE, desconsiderando a posição em que nós tivermos "pokeado" caracteres na tela. Em consequência, temos que especificar cada posição a ser "pokeada". Compare a rotina apresentada logo acima com a versão utilizando o comando PRINT:

```
1800 PRINT@ I,CHR$(160) CHR$(158) CHR$(131)...
1810 PRINT@ I+64,CHR$(128) CHR$(179) CHR$(140)...
```

.

A cada vez que um caractere é impresso, o cursor se move para a direita, pronto para imprimir o próximo caractere.

Algumas vezes, entretanto, esta característica do comando POKE pode ser de muita vantagem. Nós podemos, por exemplo, atualizar dados na tela, sem que seja preciso mudar a posição do cursor. A seguinte rotina nos demonstra essa possibilidade:

## NEW

```
10 CLS
20 PRINT@ 400,"ONDE ESTÁ VOCÊ ";
30 POKE 15360+960,42
40 PRINT "CURSOR?"
RUN
```

O comando POKE da linha 30 não interrompe a continuidade de impressão das linhas 20 e 40.

## PEEK E POKE EM PONTOS DE PRESSÃO

Até o momento, pudemos observar há uma certa deficiência na utilização do comando POKE como uma alternativa da instrução PRINT@ e das técnicas de animação com strings. Apesar de um pouco frustrante, é interessante conhecer as limitações das ferramentas que temos à nossa disposição. Vamos, agora, explorar alguns aspectos únicos do comando POKE e de seu companheiro PEEK.

Tais instruções permitem, aos programadores que trabalham com a linguagem BASIC, modificar qualquer porção da memória que eles desejem. Se utilizados com o devido cuidado, eles podem nos dar acesso a lugares na memória que controlam importantes funções. Você já teve a oportunidade de observar como controlamos a área de vídeo. Vamos agora explorar alguns outros endereços utilizados para o controle gráfico.

Há algumas partes da memória que contém várias tabelas e indicadores utilizados pelo BASIC. Por exemplo, o endereço 16912 é indicativo da largura dos caracteres do seu TRS-80. Quando o computador é ligado, os caracteres são normais e o endereço 16912 tem conteúdo 40. Para checar isto, digite:

## NEW

```
PRINT PEEK(16912)
```

Agora, digite as seguintes instruções:

## NEW

```
PRINT CHR$(23)
PRINT PEEK(16912)
```

A tela inteira vai para o modo de largura dupla quando CHR\$(23) é impresso e o conteúdo do endereço 16912 muda. O retorno à largura normal pode ser feito pressionando-se a tecla CLEAR ou utilizando-se o comando CLS. Infelizmente, ambos os métodos limpam a tela enquanto mudam a largura dos caracteres para a dimensão normal. Há momentos, entretanto, em que estamos interessados em mudar a largura dos caracteres sem limpar a tela. Isto, como você já deve estar imaginando só pode ser conseguido com a utilização do comando POKE. Experimente digitar as seguintes instruções:

```
PRINT CHR$(23)
PRINT "A A A A A "
```

Agora, tudo na tela tem a largura dobrada. Digite, então, a seguinte instrução:

```
POKE 16912,40
```

E viva! Estamos de volta à largura normal. De modo similar, podemos voltar à largura dupla, digitando a seguinte instrução:

```
POKE 16912,44
```

Experimente, agora, digitar o seguinte programa:

## NEW

```
10 CLS : J=1
20 PRINT@ 400,"TERMOS - Cz$800,00"
30 FOR I=1 TO 200 : NEXT
40 J=-J
50 POKE 16912,44
70 GOTO 30
RUN
```

Que aconteceu, então, com a largura dupla prometida na linha 50? Pressione a tecla BREAK e procure descobrir. Surpresa! Sempre esteve no modo de largura dupla, o que é muito embaraçoso de se admitir. Por alguma razão, este "pokeamento" funciona em modo imediato, mas não em um programa. Não se preocupe, porém, há um remédio para esta situação constrangedora. Mude a linha 50 para:

```
50 POKE 16912,44 : OUT 236,4
RUN
```

Surpreendente! Agora podemos alternar os dois modos. Faça, ainda, a seguinte modificação:

```
50 IF J < 0 POKE 16912,44 : OUT 236,4
60 IF J > 0 POKE 16912,40 : OUT 236,0
```

Para a maioria das aplicações, CHR\$(23) é o meio mais fácil de se passar para o modo de largura dupla, e CLS é o meio mais fácil de se retornar à largura normal. Se, porém, você não deseja que a tela seja limpada durante as mudanças de modo, utilize apropriadamente a combinação POKE/OUT em lugar do CLS.

## NOTA PARA USUÁRIOS DO TRS-80 MOD I

Se seu equipamento é um TRS-80 mod I, você também pode realizar todos os feitos aqui descritos, mas há algumas modificações quanto aos endereços e conteúdos desses endereços, que apresentamos a seguir:

O endereço responsável pelos modos de largura dupla é normal de caracteres é o 16445. Para o modo normal este endereço deve conter o número 0, e para o modo de largura dupla, o número 8. Ao ligar seu computador, ele já se encontra no modo normal.

Faça, então, as modificações nos programas apresentados de modo a poder observar, em seu micro, todos os efeitos até aqui descritos.

No caso das combinações POKE/OUT apresentadas nas modificações das linhas 50 e 60, utilize:

```
50 IF J < 0 POKE 16445,8 : OUT 255,8
60 IF J > 0 POKE 16445,0 : OUT 255,0
RUN
```

# BIORRITMO

Com o programa BIORRITMO, você pode se transformar em um verdadeiro guru aí nas suas redondezas. É só abrir um consultório, e começar a dar as consultas. Você aciona o seu micro, e solicita a data de nascimento da "vítima", e apresentar os melhores resultados de acordo com o biorritmo que seu micro irá apresentando.

```
1 CLS : GOTO 90
2 CLEAR200:DEFINTK,L:DEFDBL B,J,M-Z
3 L=0:T=25:P=3.1415926535:CLS
4 PRINT CHR$(23);
5 PRINT TAB(10);"BIORRITMO":
6 PRINT "entre com a data de nascim
ento"
7 GOSUB33
8 GOSUB43
9 JB=JD
10 PRINT:PRINT"entre com a data de
inicio para o grafico"
11 GOSUB33
12 GOSUB43
13 JC=JD
14 IF JC >=JB THEN 18
15 PRINT"DATA DO GRAFICO NAO PODE S
ER ANTERIOR AO NASCIMENTO"
16 PRINT"TENDE DE NOVO"
17 GOTO7
18 FORK=1TO1000:NEXT
19 GOSUB50
20 REM
21 N=JC-JB
22 V=23:GOSUB56 :GOSUB58
23 V=28:GOSUB56 :GOSUB58
24 V=33:GOSUB56 :GOSUB58
25 GOSUB75
26 PRINTC$;TAB(8);L$
27 JC=JC+1:L=L+1:IF L < 12 THEN 21

28 PRINT"      Pressione T para t
erminar , espaco para continuar";
29 R$=INKEY$:IF R$="" THEN 29
30 IF R$="T" OR R$="t" THEN 2
31 IF R$(">)" THEN 29
32 L=0:GOTO19
33 PRINT
34 INPUT"DIA (1 A 31)";D
35 D=INT(D):IF D<1 OR D>31 THEN 34

36 INPUT"MES (1 A 12)";M
37 M=INT(M):IF M<1 OR M>12 THEN 36

38 INPUT"ANO";Y
39 Y=INT(Y):IF Y<0 THEN 38
40 IF Y>99 THEN 42
41 Y=Y+1900:PRINT "EU ENTENDI ";Y
42 RETURN
43 W=FIX((M-14)/12)
44 JD=INT(1461*(Y+4800+W)/4)
45 B=FIX(367*(M-2-W*12)/12)
46 JD=JD+B
47 B=INT(INT(3*(Y+4900+W)/100)/4)
48 JD=JD+D-32075-B
49 RETURN
50 CLS
51 PRINT "Biorritmo. e = Emocional
f = Fisico i = Intelectual"
52 PRINT"--data--";TAB(17);
53 PRINT"mau ";TAB(33);"0";TAB(45);
"bom"
54 PRINT TAB(8);
55 FORK=1TOT+T+1:PRINTCHR$(131);:NE
XT:PRINT:RETURN
56 W=INT(N/V):R=N-(W*V)
57 RETURN
58 IF V<>23 THEN 63
59 L$=CHR$(32):FORK=1TO5:L$=L$+L$:N
EXT
60 L$=L$+LEFT$(L$,19)
61 L$=LEFT$(L$,T)+CHR$(149)+RIGHT$(
```

```
L$,T)
62 IF V=23 THEN C$="f"
63 IF V=28 THEN C$="e"
64 IF V=33 THEN C$="i"
65 W=R/V:W=W*2*P
66 W=T*SIN(W):W=W+T+1.5
67 W=INT(W):A$=MID$(L$,W,1)
68 IF A$="f" OR A$="e" OR A$="i"
THEN C$="*"
69 IF W=1 THEN 73
70 IF W=T+T+1 THEN 74
71 L$=LEFT$(L$,W-1)+C$+RIGHT$(L$,T+
T+1-W)
72 RETURN
73 L$=C$+RIGHT$(L$,T+T):RETURN
74 L$=LEFT$(L$,T+T)+C$:RETURN
75 W=JC+68569:R=INT(4*W/146097)
76 W=W-INT((146097*R+3)/4)
77 Y=INT(4000*(W+1)/1461001)
78 W=W-INT(1461*Y/4)+31
79 M=INT(80*W/2447)
80 D=W-INT(2447*M/80)
81 W=INT(M/11):M=M+2-12*W
82 Y=100*(R-49)+Y+W
83 A$=STR$(D):W=LEN(A$)-1
84 C$=MID$(A$,2,W)+"/"
85 A$=STR$(M):W=LEN(A$)-1
86 C$=C$+MID$(A$,2,W)+"/"
87 A$=STR$(Y):W=LEN(A$)-1
88 C$=C$+MID$(A$,W,2)
89 RETURN
90 FORI%=0TO380:READA%:PRINTCHR$(A%
)::NEXTI%
91 FOR X=1 TO 100: NEXT X : GOTO 2
92 DATA255,219,188,188,132,188,191,
191,191,191,188,136,188
93 DATA188,233,176,176,176,184,140,
191,189,188,176,194,191
94 DATA191,129,131,143,143,143,143,
131,130,191,191,195,176
95 DATA188,191,141,180,176,176,176,
221,191,191,191,191,191
96 DATA191,193,130,131,139,143,188,
178,147,200,139,191,168
97 DATA158,143,135,131,129,191,191,
191,191,191,191,218,143
98 DATA191,188,191,147,131,131,218,
131,131,163,187,188,191
99 DATA141,218,130,139,143,141,132,
199,160,176,156,140,140
100 DATA132,136,140,140,172,176,144
,199,176,188,143,135,129
101 DATA215,190,143,191,143,143,143
,143,135,131,131,197,184
102 DATA158,163,184,140,135,179,145
,162,179,139,140,180,147
103 DATA173,180,197,131,136,140,143
,143,143,143,191,143,189
104 DATA208,168,190,191,191,191,189
,148,201,190,151,168,191
105 DATA129,190,183,177,144,160,178
,187,189,130,191,148,171
106 DATA189,201,168,190,191,191,191
,189,148,206,130,139,191
107 DATA143,191,135,129,201,139,189
,146,175,180,139,141,180
108 DATA144,160,184,142,135,184,159
,161,190,135,201,130,139
109 DATA191,143,191,135,129,208,139
,143,143,143,143,143,143
110 DATA131,177,188,197,130,139,172
,178,147,141,140,132,136
111 DATA140,142,163,177,156,135,129
,197,140,140,142,143,143
112 DATA143,143,143,143,135,214,176
,184,190,159,131,202,131
113 DATA131,131,129,130,131,131,131
,201,130,139,175,189,180
114 DATA176,217,139,143,179,189,188
,188,188,176,196,160,144
115 DATA176,201,160,176,176,197,176
,188,188,188,191,179,143
116 DATA143,218,143,191,191,191,191
,143,184,188,190,143,135
117 DATA129,191,189,200,188,180,131
,143,189,188,180,176,143
118 DATA191,191,191,191,143,224,130
```

```
,139,143,131,196,191,191
119 DATA132,188,191,191,191,191,188
,136,191,191,195,131,139
120 DATA143,131,129,236,131,131,129
,131,143,143,143,143,131
121 DATA130,131,131,218
```

# SKETCHIT

O programa SKETCHIT é sensacional para aqueles que gostam e tem habilidades artísticas voltadas para a área do desenho. Com ele você pode fazer os mais incríveis desenhos no vídeo do seu micro, e gravá-los para uso posterior.

Trata-se de um programa americano, com instruções embutidas ainda em inglês, mas que você não deverá ter dificuldade em traduzir. Caso esta dificuldade se mostre muito grande, procure algum amigo que saiba inglês para lhe ajudar. Um conselho muito útil, porém, é: "aprenda inglês". Porque se você desejar progredir na área da informática, os melhores livros são mesmo em inglês.

```
1 HZ=0:VZ=0
10 CLS:PRINT"SKETCHIT":PRINT:PRINT
TAB(1),"BY WILLIAM H. PATRICK":
PRINT TAB(1),"PARADISE CAMP ROAD":P
RINTTAB(1),"HARRODSBURG, KY. 40330"
:FOR XX=1 TO 1000:NEXT
20 CLS
30 PRINT"PRESS 'C' TO CLEAR AND RES
TART"
40 PRINT"PRESS 'L' TO LOAD A PREVIO
USLY SAVED DRAWING"
50 PRINT"PRESS 'S' TO SAVE CURRENT
DRAWING"
55 PRINT "PRESS 'Q' TO QUIT"
60 PRINT"DRAWING CONTROLS - "
70 PRINT " PRESS ANY ARROW KEY TO
DRAW. ":PRINT "PRESS <SHIFT> AND A
RROW KEYS TO MOVE WITHOUT DRAWING O
R ERASE."
71 PRINT"(NEW ROM TRS-80 USE SHIFT
'A' FOR SHIFT DOWN ARROW)"
80 PRINT " PRESS '@' TO PAUSE."
85 PRINT "PRESS ANY DIGIT 1-9 TO CO
NTROL SPEED OF DRAWING (9 SLOW,
1 ONE BLOCK ONLY)"
86 PRINT" ALWAYS MAKE YOUR 'MOVE E
XACTLY ONE GRAPHICS BLOCK AHEAD ":P
RINT "OF THE DESIRED MOVE."
90 PRINT:INPUT"<ENTER> TO BEGIN";PG
$
100 CLS
101 CLEAR (4000):DIM D$(100)
102 X=0:Y=0:X$="@":SP=5
108 ER$=STRING$(50,32)
109 DEFINT A-Y
110 GOSUB 600:IF C$="D" THEN CLOSE
1
112 IF X$="C"THEN 100
113 IF X$="L" THEN GOTO 6000
114 IF X$="S" THEN GOTO 5000
115 IFX$="@" THEN 110
116 IF VAL(X$)>0 AND VAL(X$)<10 THE
N SP=VAL(X$)
117 IF X$="Q" THEN END
119 REM MOVE
120 IF ASC(X$)=27 THEN Y=Y+1:GOSUB
500:GOTO110
130 IF ASC(X$)=65 THEN Y=Y+1: GOSUB
500: GOTO 110
140 IF ASC(X$)=24 THEN X=X-1: GOSUB
500: GOTO 110
150 IF ASC(X$)=25 THEN X=X+1:GOSUB5
00:GOTO110
160 IF ASC(X$)=91 THEN Y=Y-1:GOTO30
0
170 IF ASC(X$)=10 THEN Y=Y+1: GOTO
300
180 IF ASC(X$)=9 THEN X=X+1: GOTO30
0
190 IF ASC(X$)=8 THEN X=X-1 :GOTO30
```

```

0
300 GOSUB 500
320 GOTO110
500 REM OFF SCREEN
510 IF X<0 THEN X=0
520 IF X>127 THEN X=127
530 IF Y<0 THEN Y=0
540 IF Y>47 THEN Y=47
550 RETURN
600 REM INKEY ROUTINE
601 XX$=INKEY$
603 SET(X,Y): FOR Z=1 TO 2*SP/3: NEXT
: RESET(X,Y): FOR Z=1 TO 2*SP/3: NEX
T
605 IF XX$="" AND X$="" THEN 601
606 IF XX$="" AND SP=1 THEN 601
607 IF XX$<>"" THEN X$=XX$
609 IF ASC(X$)<>65 AND (ASC(X$)>300
R ASC(X$)<23 ) THEN SET(X,Y)
611 RETURN
4999 REM SAVE DRAWING
5000 GOSUB 7000
5001 PRINT@900,"";:INPUT " TAPE O
R DISK (T/D)";C$:PRINT@900,ER$;:IF
C$="T" THEN 5100 ELSE PRINT@ 900,ER
$ ;:PRINT@900,"";:INPUT " ENTER F
ILE NAME";FM$
5002 OPEN "D",1,FM$
5003 FORX=BX TO EX:FOR Y=BY TO EY:P
RINT @940,"X="X:PRINT@ 950,"Y="Y
5004 GOSUB 11000:IF POINT(X,Y)=-1 T
HEN PRINT #1,ZMF*(X+HZ)," ZMF*(Y+V
Z)";:
5005 NEXT Y, X
5006 CLOSE 1
5007 GOTO20
5100 PRINT @900,"READY CASSETTE FOR
RECORDING, PRESS ENTER";:INPUT PG$
:PRINT@ 900,ER$;:KK=1
5101 D$(KK)=" "
5102 FOR X=BX TO EX:FOR Y=BY TO EY
5104 IF POINT(X,Y)=-1 THEN D$(KK)=D
$(KK)+ STR$(X)+" / "+STR$(Y)+" / "
5105 IFLen(D$(KK))>230THENKK=KK+1:D
$(KK)=" "
5106 NEXT Y, X
5107 D$(KK)=D$(KK)+"999/999/"
5108 PRINT#-1,KK:FOR W=1 TO KK:PRINT#
-1,D$( W):NEXT
5109 X=0:Y=0:XX$=""@":X$=""@":PRINT@9
00,"TAPE FINISHED";:GOTO110
6000 REM LOADDRAWING
6001 PRINT@ 900,"";:INPUT"TAPE OR D
ISK (T/D)";C$:PRINT@ 900,ER$;:IF C$
="T" THEN 6100 ELSE PRINT@ 900, "
";:INPUT "ENTER FILE NAME";FM
$:PRINT@ 900, ER;
6002 OPEN "I",1,FM$
6003 INPUT #1,X,Y:SET(X,Y):IF EOF(1
) THEN 6006 ELSE 6003
6006 CLOSE1:X$=""@"
6007 GOTO110
6100 REM TAPE INPUT
6101 INPUT#-1,KK:FORZZ=1 TO KK:INPU
T#-1,D$(ZZ):NEXT:ZZ=0
6102 ZZ=ZZ+1
6103 FOR K=1 TO 2 :C=1
6104 IF MID$(D$(ZZ),C,1)<>"/" THEN C
=C+1: GOTO 6104
6105 X(K)=VAL(LEFT$(D$(ZZ),C-1))
6106 D$(ZZ)=RIGHT$(D$(ZZ),LEN(D$(ZZ
))-C)
6107 PRINT @ 900,X(1), X(2),LEN(D$(
ZZ));:NEXT K
6108 IF X(1)=999 THEN XX$=""":X$=""@
":X=0:Y=0:GOTO110
6109 SET (X(1), X(2))
6110 IF LEN(D$(ZZ))<=1 THEN 6102 EL
SE 6103
7000 REM MARK BOUNDARIES OF DRAWING
SCAN
7001 PRINT@ 900,"";:INPUT"SCAN ENTI
RE SCREEN (Y/N)";PG$:PRINT @ 9
00, ER$;:
7002 IF PG$="Y" THEN BX=0:EX=127:BY
=0:EY=47:GOTO 7100
7003 PRINT @ 900,"PRESS 'B' AT THE

```

```

UPPER LEFT CORNER OF WHAT YOU WAN
T SAVED":FOR Z=1 TO 1500:NEXT : PRI
NT @ 900,"USE THE ARROWS TO MOV
E TO THE LOWER RIGHT OF WHAT YOU WA
NT"
7004 X=0:Y=0:BX=0:BY=0:EX=0:EY=0
7005 IF POINT(X,Y)=-1 THEN MARK=99
ELSE MARK =0
7006 SET(X,Y):FOR Z=1 TO 3:NEXT:RES
ET(X,Y):FOR Z=1 TO 3:NEXT: IF MARK=
99 THEN SET(X,Y)
7007 M$=INKEY$:IF M$="" THEN 7005
7008 IF ASC(M$)=91 THEN Y=Y-1:GOTO7
020
7009 IF ASC(M$)=10 THEN Y=Y+1:GOTO7
020
7010 IF ASC(M$)=9 THEN X=X+1:GOTO70
20
7011 IF ASC(M$)=8 THEN X=X-1
7012 IF M$="F" THEN PRINT@ 900,STRI
NG$(50,32):GOTO 7100
7013 IF M$="B" THEN BX=X:BY=Y

```

```

7020 GOSUB 500
7030 IF X>EX THEN EX=X
7040 IF Y>EY THEN EY=Y
7050 PRINT@ 900,"PRESS 'F' WHEN YOU
ARE FINISHED ";:PRINT STRING$(27,3
2);
7060 GOTO 7005
7090 HZ=0:VZ=0:ZMF=1
7100 PRINT@ 900,"";:INPUT"HORIZONTA
L SHIFT";HZ:PRINT@ 900,ER$;
7110 PRINT@ 900,"";:INPUT"VERTICAL
SHIFT ";VZ:PRINT@ 900,ER$;
7115 PRINT@ 900,"";:INPUT "SIZE FAC
TOR ";ZMF:PRINT@ 900,ER$;
7120 RETURN
11000 IF ZMF*(X+HZ)<0 OR ZMF*(X+HZ)
>127 THEN PRINT@ 900,ER$: PRINT@
900, "HORIZONTAL ERROR";:GOTO110
11001 IF ZMF*(Y+VZ)<0 OR ZMF*(Y+VZ)
>47 THEN PRINT@ 900,ER$:PRINT@ 900,
"VERTICAL ERROR":GOTO110
11002 RETURN

```

## # COPSYS #

Com o programa COPSYS, os usuários do TRS-80 mod I nível II e mod III poderão desbloquear qualquer soft em linguagem de máquina que rode na versão cassete.

Agora, você não precisa ter mais receio de algum acidente com a sua fita preferida. Com o COPSYS você faz o "backup" de todas as suas fitas com programas em linguagem de máquina.

```

100 REM**** PROGRAMA PARA PRODUZIR
SALVAGUARDAS DE FITAS
110 REM**** GRAVADAS EM FORMATO SYS
TEM
120 REM**** ESTE PROGRAMA RODA EM Q
UALQUER MICRO-COMPUTADOR
130 REM**** TOTALMENTE COMPATIVEL C
OM O TRS80 MODELO I NIVEL II
140 REM**** DA TANDY CORPORATION
150 REM**
160 CLS:PRINT@458," CARREGANDO O
PROGRAMA"
170 GOSUB65000
180 POKE16526,143:POKE16527,79:X=US
R(0)
5000 DATA 5000,20200,20720
5010 DATA 66,65,84,65,32,65,32,84,6
9,67,117
5020 DATA 76,65,32,76,32,80,65,82,6
5,32,93
5030 DATA 76,69,82,32,85,77,65,32,7
0,73,149
5040 DATA 84,65,0,32,65,32,84,69,67
,76,62
5050 DATA 65,32,71,32,80,65,82,65,3
2,82,94
5060 DATA 69,80,82,79,68,85,90,73,8
2,32,228
5070 DATA 79,85,32,67,76,69,65,82,3
2,80,155
5080 DATA 47,32,82,69,84,79,82,78,7
9,32,152
5090 DATA 65,79,32,66,65,83,73,67,0
,76,94
5100 DATA 69,78,68,79,32,65,32,70,7
3,84,138
5110 DATA 65,0,71,82,65,86,65,78,68
,79,147
5120 DATA 32,32,32,32,0,69,82,82,79
,32,216
5130 DATA 68,69,32,70,79,82,77,65,8
4,79,193
5140 DATA 32,40,66,65,84,65,32,69,7
8,84,103
5150 DATA 69,82,32,80,47,32,82,69,8
4,79,144
5160 DATA 82,78,79,32,79,85,32,67,7
6,69,167
5170 DATA 65,82,32,80,47,32,82,69,8

```

```

4,79,140
5180 DATA 82,78,79,32,65,79,32,66,6
5,83,149
5190 DATA 73,67,41,0,69,82,82,79,32
,68,81
5200 DATA 69,32,67,72,69,67,75,45,8
3,85,152
5210 DATA 77,32,40,69,78,84,69,82,3
2,80,131
5220 DATA 47,32,82,69,84,79,82,78,7
9,32,152
5230 DATA 79,85,32,67,76,69,65,82,3
2,80,155
5240 DATA 47,32,82,69,84,79,82,78,7
9,32,152
5250 DATA 65,79,32,66,65,83,73,67,4
1,0,59
5260 DATA 0,0,49,231,78,205,213,80,
205,43,80
5270 DATA 0,254,31,202,235,80,254,7
6,32,244,128
5280 DATA 33,128,60,17,65,79,205,19
2,80,62,153
5290 DATA 1,205,18,2,205,150,2,33,2
41,80,169
5300 DATA 205,183,80,254,85,194,153
,80,17,56,27
5310 DATA 60,6,6,205,183,80,18,19,1
6,249,74
5320 DATA 175,79,205,183,80,254,60,
40,8,254,58
5330 DATA 120,202,77,80,195,153,80,
205,183,80,95
5340 DATA 71,205,183,80,129,79,205,
183,80,129,64
5350 DATA 79,205,183,80,129,79,16,2
49,205,183,128
5360 DATA 80,185,194,201,80,24,209,
205,183,80,161
5370 DATA 205,183,80,34,226,79,205,
248,1,205,186
5380 DATA 213,80,205,43,0,254,31,20
2,235,80,63
5390 DATA 254,76,202,246,79,254,71,
194,92,80,12
5400 DATA 33,128,60,17,78,79,205,19
2,80,62,166
5410 DATA 1,205,18,2,205,132,2,33,2
41,80,151
5420 DATA 126,205,100,2,35,237,91,2
26,79,124,201
5430 DATA 186,32,243,125,187,32,239
,205,248,1,218
5440 DATA 195,89,80,33,128,60,17,91
,79,205,209
5450 DATA 192,80,205,248,1,205,43,0
,254,31,235
5460 DATA 202,235,80,254,13,32,244,
205,248,1,234
5470 DATA 195,228,79,205,44,2,205,5
3,2,119,108
5480 DATA 35,201,26,254,0,200,119,3
5,19,24,145

```

# GERANDO O ALFABETO COM O COMANDO DRAW

## INTRODUÇÃO

Neste artigo, exploraremos o uso da instrução DRAW para gerar um alfabeto. Como já sabemos, é muito difícil, no TRS-80 Color, misturar texto e gráfico. Assim, procuraremos desenvolver um alfabeto através de gráficos e que pode ser utilizado para produzir texto em meio aos seus gráficos.

## ANALISANDO O PROBLEMA

O alfabeto deve ser definido na forma de strings para o comando DRAW. O usuário selecionará um string a ser impresso utilizando aquele alfabeto, indicando o tamanho, a posição e a orientação do string a ser impresso. Então, o programa se encarregará de exibir o string selecionado na posição indicada.

Antes de descrevermos o programa em maiores detalhes, permitam-nos rever as particularidade do comando DRAW, que é um dos mais poderosos do repertório do TRS-80 Color.

Este comando é utilizado em um programa do seguinte modo:

### DRAW linha

onde linha é uma expressão string que pode conter um ou mais dos seguintes elementos:

### COMANDOS DE MOVIMENTO

Mx,y ..... move a posição DRAW para as coordenadas x,y.  
M<sup>+</sup>x, <sup>+</sup>y ..... movimento relativo à atual posição DRAW.  
Ud ..... move para cima d unidades da posição DRAW.  
Dd ..... move para baixo d unidades da posição DRAW.  
Ld ..... move para a esquerda d unidades da posição DRAW.  
Rd ..... move para a direita d unidades da posição DRAW.  
Ed ..... move para cima e para a direita (45 graus) d unidades.  
Fd ..... move para baixo e para a direita (315 graus) d unidades.  
Gd ..... move para baixo e para a esquerda (225 graus) d unidades.  
Hd ..... move para cima e para a esquerda (135 graus) d unidades.

### OPÇÕES

B ..... movimento sem utilizar o comando DRAW.  
N ..... retorno à posição inicial.  
X ..... executa uma substring.

### MODOS

Ax = muda o ângulo de todos os DRAWS subsequentes.  
x=0 ..... 0 graus.  
x=1 ..... 90 graus.  
x=2 ..... 180 graus.  
x=3 ..... 270 graus.  
Cc = DRAW utilizando a cor c  
Sx = escala, sendo x um número entre 1 e 62.  
x=1 ..... 1/4 da escala.  
x=2 ..... 2/4 da escala.  
x=3 ..... 3/4 da escala.  
x=4 ..... escala integral

Apresentamos, a seguir, alguns exemplos deste poderoso comando, cada um com uma breve explicação acerca do seu efeito. Note que os símbolos ";" que separam os comandos podem ser omitidos, exceto quando se utiliza a opção Substring. O símbolo ";" deve sempre seguir um x-string.

DRAW "U25;R25;D25;L25"

Quadricula o centro da tela se a posição DRAW começa no centro.

DRAW "BM1,255;R10"

Linha no canto inferior esquerdo.

DRAW "BM+20,-50;R30"

Movimento branco à direita 20, para cima 50, desenho à direita 30.

A\$="BM0,0R50D50R50D50"

DRAW A\$+"R10D10"

Começando no canto superior esquerdo (0,0) desenha rt50 dn50 rt50 dn 50, rt10 dn10.

DRAW "BM1, 191U10R10D10L10;XA\$,,"

Desenha um quadrado de 10 unidades de lado na parte inferior da tela e executa o desenho de A\$.

## O PLANEJAMENTO DO INPUT/OUTPUT

O usuário é solicitado a fornecer um string de caracteres a serem desenhados na tela. Então, o usuário especifica o tamanho como um valor entre 1 e 62, um ângulo (0=0; 1=90; 2=180; 3=270), e finalmente uma posição em coordenadas X-Y. A tela é definida como uma grade de 256 colunas e 192 linhas (resolução 256 x 192). O endereço do canto superior esquerdo é (0,0); do canto superior direito é (255,0); do canto inferior esquerdo é (0,191); e do canto inferior direito é (255,191).

O seguinte é um exemplo de diálogo entre o usuário e o computador, onde o computador é identificado como C e o usuário como U.

C: STRING?  
U: HUMPTY DUMPTY  
C: TAMANHO. .2,4,8,12. .?  
U: 3  
C: ÂNGULO. .0=0, 1=90, 2=180, 3=270. .?  
U: 1  
C: POSIÇÃO (X,Y). .?  
U: 100,0

No exemplo acima descrito, o usuário está solicitando que as palavras "HUMPTY DUMPTY" sejam impressas na direção de cima para baixo, começando um pouco à esquerda do centro e a partir do topo da tela. Cada letra sendo em tamanho 3/4 da escala.

Após este diálogo, o computador limpa a tela e imprime a frase na escala apropriada, ângulo e posição solicitados pelo usuário. Qualquer tecla sendo pressionada enquanto a mensagem se encontra na tela fará com que o computador recomece o processo solicitando um novo conjunto de palavras.

É importante que o projeto leve em consideração que o usuário poderá tentar introduzir no computador uma frase maior do que a que ele está preparado para receber. Nesse caso, será interessante que o micro acuse este fato, e esclareça o usuário, permitindo o reinício do diálogo.

## PSEUDO-CÓDIGOS

Apresentamos, agora, algumas características do programa acima descrito. Entre parêntesis, encontra-se o número das linhas responsáveis pelo efeito mencionado, encontradas na listagem abaixo transcrita.

- 1.- Reserva espaço para 40 strings que definirão cada um dos caracteres a serem desenhados. (50)
- 2.- Define quatro formas comumente usadas que serão utilizadas em diversas letras. (60-90)
- 3.- Define 37 formas de caracteres como comandos DRAW nos strings a serem executados sob solicitação. Os 37 caracteres são 26 letras do alfabeto A-Z; 10 dígitos 0-9; e o espaço branco. As últimas 3 posições do string definido no item 1 não são usadas. (100-460)
- 4.- Limpa a tela. (470)
- 5.- Recebe a mensagem do usuário, tamanho, ângulo e posição. (480-510)
- 6.- Aciona a rotina de desenho do string. (1000)
- 7.- Proporciona uma pausa na apresentação da tela e aguarda pelo usuário. (530-540)
- 8.- Se o usuário introduz "/", o computador limpa a tela e pára. Caso contrário, ele retorna ao item 4. (550-560)

Agora, apresentamos detalhes da rotina de desenho.

- 1.- Estabelece resolução 256 x 192, tela branca. (1010-1030)
- 2.- Prepara o computador para a escala e ângulo. (1040)
- 3.- Faz I\$ igual a um string de caracteres desenháveis. (1050)
- 4.- Estabelece a posição para o primeiro desenho nas coordenadas X-Y solicitadas pelo usuário. (1060)
- 5.- Checa cada um dos caracteres da mensagem do usuário para ver se é um dos 37 caracteres que podem ser desenhados como definidos pela linha 1050. Se é, o computador o desenha executando o string apropriado conforme definido no programa principal. Se não, o computador simplesmente o ignora. (1070-1100)
- 6.- Retorna. (1110)

## LISTAGEM DO PROGRAMA

```
10 ' PROGRAM : CHAPTER 2 : ALPHABET
20 ' AUTHORS : JOHN P GRILLO
30 '          JD ROBERTSON
40 '
50 DIM A$(40),B$(4)
60 B$(1)="BM+0,+8R12D12L12U12"
70 B$(2)="BM+12,+8D12L12U12"
80 B$(3)="BM+12,+20L12U12NR12"
90 B$(4)="BM+12,+20U12L12ND12"
100 A$(1)="XB$(4);BM+0,+6R12BM+8,-14"
110 A$(2)="XB$(1);UBBM+20,+0"
120 A$(3)="XB$(3);BM+20,-8"
130 A$(4)="XB$(1);BM+12,+0UBBM+8,+0"
140 A$(5)="XB$(3);BM+0,+6R12BM+8,-14"
150 A$(6)="BM+0,+8NR12D12U6R8BM+12,-14"
160 A$(7)="XB$(1);BM+12,+12D8L6BM+14,-28"
170 A$(8)="XB$(4);UBBM+20,+0"
180 A$(9)="BM+6,+8D12BM+6,+0L12BM+6,-18D2BM+14,-4"
190 A$(10)="BM+12,+8D20L12UBBM+12,-18D2BM+8,-4"
200 A$(11)="D20UBNE12E2F10BM+8,-20"
210 A$(12)="D20R12BM+8,-20"
220 A$(13)="XB$(4);BM+6,+0D12BM+14,-20"
230 A$(14)="XB$(4);BM+20,-8"
240 A$(15)="XB$(1);BM+20,-8"
250 A$(16)="XB$(1);D20BM+20,-28"
260 A$(17)="XB$(1);BM+12,+0D20BM+8,-28"
270 A$(18)="BM+12,+8L12D12BM+20,-20"
280 A$(19)="BM+12,+8L12D6R12D6L12BM+20,-20"
290 A$(20)="BM+0,+8R6ND12R6BM+8,-8"
300 A$(21)="XB$(2);BM+20,-8"
310 A$(22)="BM+0,+8D12E12BM+8,-8"
320 A$(23)="XB$(2);BM+6,+0D12BM+14,-20"
330 A$(24)="BM+6,+14NE6NF6NG6NH6BM+14,-14"
340 A$(25)="XB$(2);BM+6,+12D8BM+14,-28"
350 A$(26)="BM+0,+8R12G12R12BM+8,-20"
360 A$(27)="D20R12U20L12BM+20,+0"
370 A$(28)="BM+6,+0NG4D20NR6L6BM+20,-20"
380 A$(29)="R12D10L12D10R12BM+8,-20"
390 A$(30)="R12D10NL12D10L12BM+20,-20"
400 A$(31)="D10R12NU10D10BM+8,-20"
410 A$(32)="NR12D10R12D10L12BM+20,-20"
420 A$(33)="NR4D20R12U10L12BM+20,-10"
430 A$(34)="R12D20BM+8,-20"
```

```
440 A$(35)="ND20R12D10NL12D10L12BM+20,-20"
450 A$(36)="ND10R12D10NL12D10BM+8,-20"
460 A$(37)="BM+20,+0"
470 CLS 0
480 INPUT "STRING";S$
490 INPUT "SIZE..2,4,8,12..";S
500 INPUT "ANGLE..0=0,1=90,2=180,3=270..";A
510 INPUT "POSITION (X,Y)..";X,Y
520 GOSUB 1000
530 X$=INKEY$
540 IF X$="" THEN 530
550 IF X$="/" THEN CLS 6 : STOP
560 GOTO 470
1000 ' **** DRAW THE STRING
1010 PMODE 4,1
1020 SCREEN 1,1
1030 PCLS
1040 DRAW "S"+STR$(S)+"A"+STR$(A)
1050 L$="ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 "
1060 DRAW "BM"+STR$(X)+","+STR$(Y)
1070 FOR IB=1 TO LEN(S$)
1080 KB=INSTR(L$,MID$(S$,IB,1))
1090 IF KB<>0 THEN DRAW"XA$(KB);"
1100 NEXT IB
1110 RETURN
9999 END
```

grillo and robertson

## NOTAS ADICIONAIS

Este programa é repleto de características que demonstram quão avançada esta versão gráfica da linguagem BASIC realmente é. Considere, por exemplo, como o computador desenha a letra A. Após a definição da escala, direção e posição, o computador desenha a letra (como descrita no string definido em A\$(1) na linha 100), do seguinte modo:

- 1.- Estabelece K8 como 1 para identificar que elemento de A\$ deve ser desenhado.
- 2.- Executa o comando: DRAW "XA\$(K8);". Este string utiliza o comando X (execução) para executar o string A\$(1). Lembre-se de que K8 é 1 neste caso.
- 3.- A\$(1) é executado, e tem um comando X (execução) nele, para produzir a forma comum definida em B\$(4). Esta é, realmente, uma técnica poderosa.

O comando DRAW, como incorporado neste programa, lhe permite utilizar as letras e dígitos que definimos no modo gráfico. Como seria possível definir seus próprios caracteres para serem utilizados no modo gráfico? Como você provavelmente já deve ter adivinhado, você poderia modificar os 37 comandos string definidos para formar os caracteres, à sua vontade, utilizando o comando CIRCLE, ou modificando os ângulos.

Bem, nós terminamos aqui, mas o seu trabalho continua. Digite todo o programa em seu micro, e passe a utilizá-lo em seus trabalhos.

# CENTOPEIA

CENTOPEIA é um jogo em baixa resolução, do tipo centopéia, escrito para o TRS-Color, em que você é um veículo minado, que transita por baixo da areia a fim de destruir grandes rochas e, muito importante, evitar os vermes da areia que se encontram entinchados.

Após a página de título, pressione ENTER e você já estará em ação. Utilize o joystick da direita para jogar. Atire com o botão de disparo.

```

10 CLEAR10:CLS0:PRINT@107," SAND
WORM ";:PRINT@164," BY: PETER ME
YERS, 1985 ";:PRINT@230," FOR TH
E TRS-80 COCO ";:PRINT@420," PRE
SS <ANY KEY> TO PLAY ";
20 A$=INKEY$:IFA$=""THEN20
30 LC=7:LV=3:B=0:S=0
40 CLS3:FORX=1TO60:PRINT@RND(447
),"o";:NEXT:FORX=6TO13:CP(X-5)=X
:DM(X-5)=1:NEXTX
50 PRINT@1,STR$(S)+" ";:H=JOYSTK
(0):V=JOYSTK(1)
60 A=A+1:IFA=1THENSC=464
70 PRINT@SC,CHR$(128);
80 IFH>45THENSC=SC+1 ELSEIFH<15T
HENSC=SC-1
90 IFV>45THENSC=SC+32 ELSEIFV<15
THENSC=SC-32
100 IFSC>511THENSC=SC-32 ELSEIFSC
C<354 THENSC=SC+32
110 IFSC>510THENSC=SC-1 ELSEIFSC
<352THENSC=SC+1
120 IF PEEK(SC+1024)=15 THENMU=1
130 IFMU=1ANDH>45THENSC=SC-1 ELS
EIFMU=1ANDH<15THENSC=SC+1
140 IFMU=1ANDV>45THENSC=SC-32 EL
SEIFMU=1ANDV<15THENSC=SC+32
150 MU=0
160 PRINT@SC,CHR$(183);
170 P=PEEK(65280):IFP=126ORP=254
THENGOSUB330
180 B=B+1:IFB>1THENNEXTX
190 FORX=LC TO1 STEP-1
200 PRINT@CP(X),CHR$(128);
210 CP(X)=CP(X)+DM(X)
220 IF(CP(X)+1)/32=INT((CP(X)+1)
/32) THENCP(X)=CP(X)+32:DM(X)=-1
230 IFCP(X)/32=INT(CP(X)/32) THE
NCP(X)=CP(X)+32:DM(X)=1
240 IFPEEK(CP(X)+1024)=15 THENCM
=1
250 IFCM=1 ANDDM(X)=1 OR CM=1 AN
DDM(X)=-1 THENDM(X)=32:CP(X)=CP(
X)+31:CM=0
260 IFCM=1 ANDDM(X)=32 THEN DM(X
)=1:CP(X)=CP(X)-31
270 IF CP(X)>448 THENTB=TB+1:IFT
B=1 THENDM(X)=-1
280 IFCP(X)>510 THENCP(X)=CP(X)-
64:DM(X)=-1
290 IF PEEK(CP(X)+1024)=183 THEN
GOSUB450
300 PRINT@CP(X),CHR$(153);:CM=0
310 CL=RND(3):IFCL=1THEN390
320 GOTO50
330 SOUND255,1:FORLP=SC-32 TO1ST
EP-32
340 IFPEEK(LP+1024)=15 THENS=S+3
:PRINT@LP,CHR$(175);:SOUND100,1:
RETURN
350 IFPEEK(LP+1024)=153 THENS=S+
10:PRINT@CP(LC),"o";:LC=LC-1:IFL
C=0THENFORX=6TO13:CP(X-5)=X:DM(X
-5)=1:NEXTX:LC=7:X=7:TB=0:SOUND1
,2:SOUND8,1:RETURN:ELSESOUND1,2:
SOUND8,1:RETURN
360 PRINT@LP,CHR$(207);:PRINT@LP
,CHR$(128);:PRINT@LP,CHR$(175);
370 NEXTLP
380 RETURN
390 IFCP(LC)>478 THEN50ELSEFORPL

```

```

=CP(LC)+32TO510STEP32
400 IFPEEK(PL+1024)=15 THENSOUND
200,1:GOTO430
410 IFPEEK(PL+1024)=183 THENDS=1
:GOTO440
420 PRINT@PL,CHR$(170);:NEXTPL:S
OUND200,1
430 IFPL>510 THENPL=510
440 FOREL=CP(LC)+32TOPL STEP32:P
RINT@EL,CHR$(175);:NEXTEL:IFDS=1
THENDS=0:GOSUB450:GOTO50 ELSEGO
TO50
450 LV=LV-1:IFLV=0THEN460ELSEFOR
Y=6TO13:PRINT@CP(Y-5),CHR$(175);
:CP(Y-5)=Y:DM(Y-5)=1:NEXTY:TB=0:
SOUND20,7:SOUND1,9:FORMM=1TO20:P
RINT@RND(479),"o";:NEXTMM:RETURN
460 IFS>HS THENHS=S
470 PRINT@1,STR$(S)+" ";:PRINT@7
1," HIGH SCORE:"HS" ";:PRINT@234
," GAME OVER ";:PRINT@421," ANOT
HER GAME ? (Y/N) ";
480 A$=INKEY$:IFA$=""THEN480 ELS
EIFA$="Y"THENGOTO30 ELSEEND

```

## GRAFICOS DE ESCHER

M. C. Escher ficou famoso pelas situações incríveis que conseguia ao desenhar em um estilo todo próprio em que ele deliberadamente feria regras fundamentais da perspectiva, criando, assim, objetos que não podiam ser fabricados e outras cenas mirabolantes. Este programa lhe permitirá desenhar gráficos do tipo que Escher costumava fazer.

Digite a listagem 1, mas corrija o valor de DV na linha 440, que deve ser igual a 3, de modo a permitir que você utilize o joystick. A seguir, grave o programa antes de usá-lo.

Já a listagem 2 lhe permitirá imprimir uma área de 8 x 10,5 polegadas com o padrão que você criou.

Se você tem criatividade, vá fundo no mundo de Escher. Nós aqui ficamos imaginando o que teria feito esse gênio do desenho se tivesse vivido em uma época de modo a poder contar com os recursos do computador.

### Listing 1: ESCHER

```

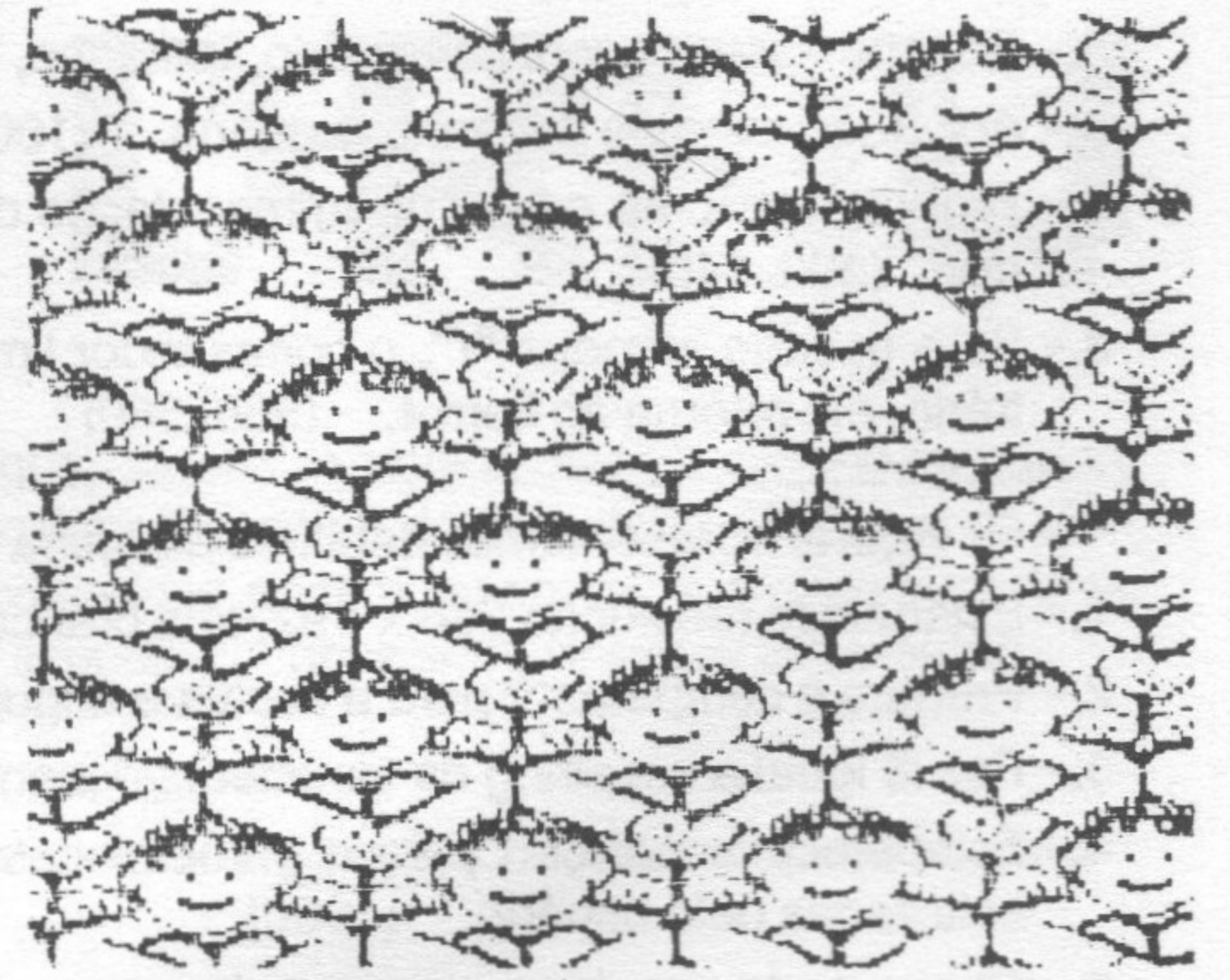
10 ON DV GOTO 30,40,50
20 GOTO430' & SET UP VARIABLES
30 X=PEEK(65376):Y=PEEK(65377):S
=PEEK(65378):GOTO 60 ' X PAD
40 Y=PEEK(65433)/1.15:X=PEEK(654
34):S=PEEK(65424)+3:GOTO60'HIRE
50 X=JOYSTK(0):Y=JOYSTK(1):S=(PE
EK(65280)AND1)+3 ' JOYSTICK
60 PUT(X,Y)-(X+K,Y+K),C,NOT
70 IF S=3 THEN 90 ELSE PUT(X,Y)-
(X+K,Y+K),C,NOT
80 A$=INKEY$:IF A$<>" " THEN 180
ELSE 10
90 COLORC:LINE(X,Y)-(X+K,Y+K),PS
ET,BF
100 XR=INT(X/((R+1)*2))*((R+1)*2
)
110 X=X-XR
120 IF M THEN LINE(R-X+XR+R+1,Y)
-(R-X+XR+K+R+1,Y+K),PSET,BF
130 Y=Y-INT(Y/((R+1)*2))*((R+1)*
2)
140 IF Y>R THEN Y=Y-R-1:X=X-R-1:
IF X<0 THEN X=(R+1)*2+X
150 COLORC:LINE(X+W,Y+W)-(X+K+W,
Y+K+W),PSET,BF
160 IF M THEN LINE(R+(R-X)+W+1,Y
+W)-(R+(R-X)+K+W+1,Y+K+W),PSET,B
F
170 GOTO 10
180 IF A$="C" THEN 400' CLEARPIC
190 IF A$="W" THEN C=1' WHITE
200 IF A$="B" THEN C=0' BLACK
210 IF A$="1" THEN K=0' PENSIZEL

```

```

220 IF A$="2" THEN K=1' PENSIZEL
230 IF A$="3" THEN K=2' PENSIZEL
240 IF A$="X" THEN DV=1'X-PAD
250 IF A$="P" THEN DV=2'HIRESPAK
260 IF A$="J" THEN DV=3'JOYSTICK
270 IF A$="M" THEN IF M THEN M=0
ELSE M=1' MIRROR ON/OFF
280 IF A$="S" THEN CLS:PRINT"NAM
E TO SAVE AS (...../"ET$)":P
RINT@16,("":LINEINPUT A$:PRINT@
17,A$:PRINT@25,/"ET$)":IF A$="
" THEN450 ELSE IF PEEK(188)=6 TH
EN CSAVEM LEFT$(A$,8),&H600,&H1D
FF,R+1 ELSE SAVEM LEFT$(A$,8)+"/
"+ET$,&HE00.&H25FF,R+1:GO

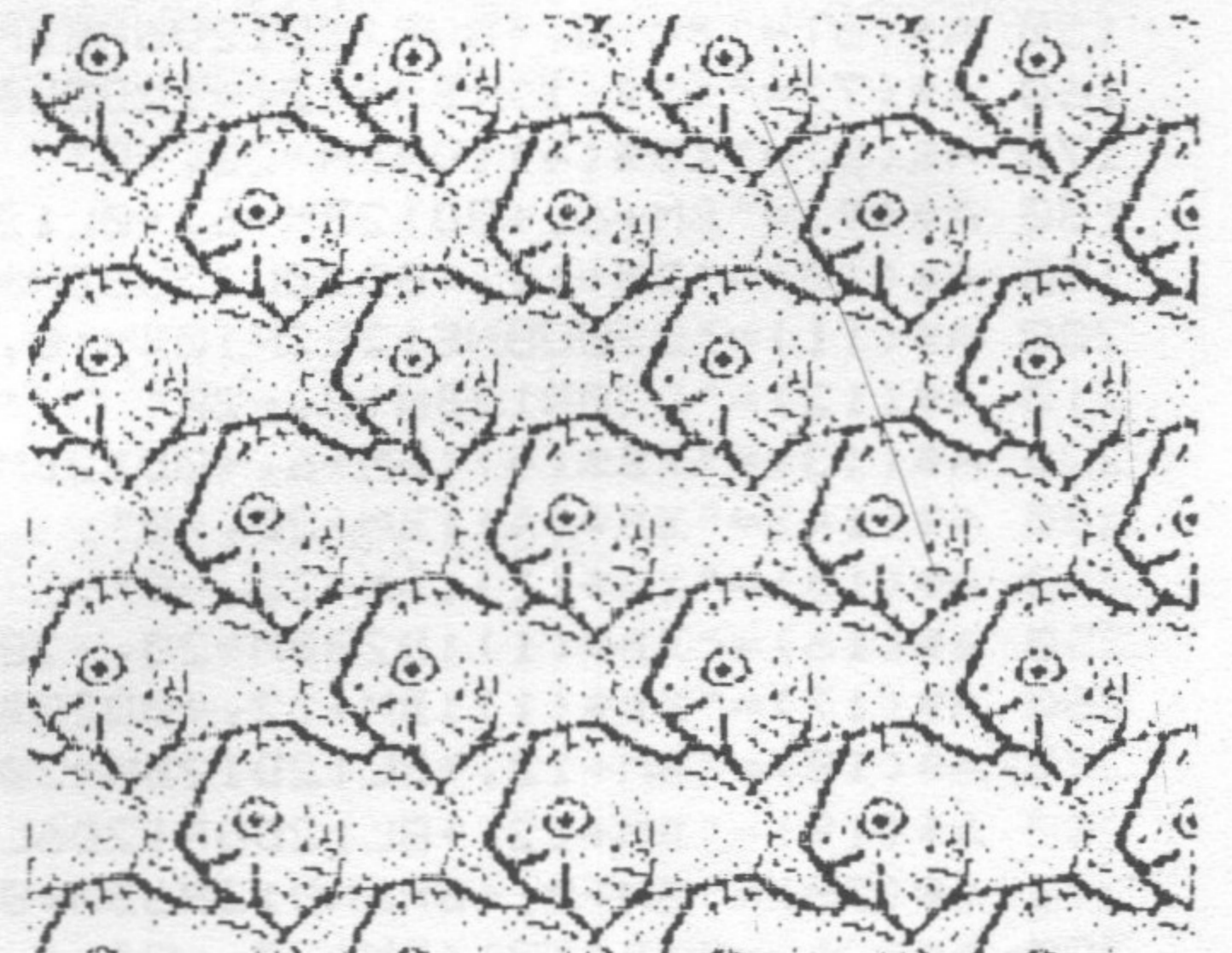
```



```

290 IF A$<>" " THEN 10'UPDATE PI
CTURE
300 SCREEN1,0':POKE65495,0'SPEED
POKE
310 GET(W,W)-(R+W,R+W),A
320 GET(W+R+1,W)-(W+R+R+1,R+W),B
330 F=0
340 FOR X=0 TO 255 STEP R+1
350 FOR Y=0 TO 191 STEP R+1
360 IF F THEN PUT(X,Y)-(X+R,Y+R)
,B,PSET:F=0 ELSE PUT(X,Y)-(X+R,Y
+R),A,PSET:F=1
370 NEXTY:IF F=0 THEN F=1 ELSE F
=0
380 NEXTX':POKE65494,0'SLOW POKE
390 SCREEN1,1:GOTO10
400 CLS:INPUT"ARE YOU SURE YOU W
ANT TO CLEAR THE SCREEN (Y/N)";
A$
410 IF A$<>"Y" THEN SCREEN1,1:GO
TO10
420 PCLS1:COLOR0:SCREEN1,1:GOTO4
50

```



```

430 W=32:R=W-1:W=W*3:ET$="ESH"
440 DIM A(R),B(R),C(9):K=1:DV=1
450 PMODE4:SCREEN1,1:GOTO10
460 ' *****
470 ' * A STUDY OF REGULAR *
480 ' * DIVISION OF THE PLANE *
490 ' *****
500 ' * COPYRIGHT 1986 BY *
510 ' * ERIC M. WHITE *
520 ' * ALL RIGHTS RESERVED *
530 ' *****
540 ' * VERSION: 1.0 8604.20 *
550 ' *****

```



Listing 2: PRINT200

```

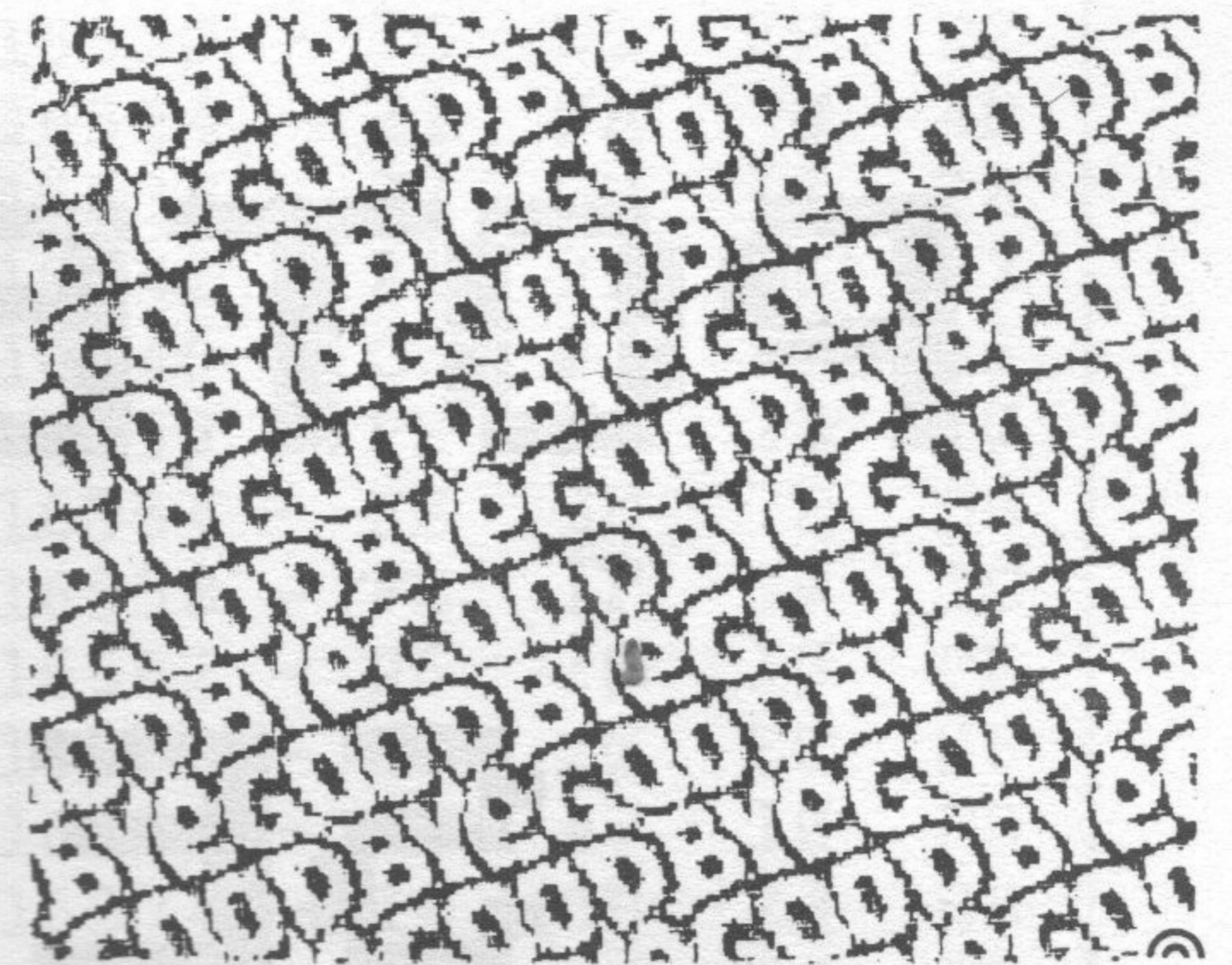
10 CLEAR4000:ET$="ESH":DIM PR$(6
4)
20 GN$=CHR$(18)' GRAPHICS ON
30 GF$=CHR$(30)' GRAPHICS OFF
40 CLS:PRINT"NAME TO LOAD IS (..
...../"ET$)"
50 PRINT@16,"(";:LINEINPUT A$:PR
INT@17,A$
60 PRINT@25,"/"ET$)":IF A$="" T
HEN RUN ELSE IF PEEK(188)=6 THEN
CLOADM LEFT$(A$,8) ELSE LOADM L
EFT$(A$,8)+"/"+ET$
70 PMODE4:SCREEN1,1
80 PRINT#-2,CHR$(27)CHR$(23)' CO
MPRESS ON
90 FOR Y1=1 TO 63 STEP 7
100 FOR X1=0 TO 63
110 FOR Y2=0 TO 6
120 IF PPOINT(X1,Y1+Y2)=0 THEN N
B=NB+INT(2^Y2)
130 NEXT Y2
140 PR$(Y1)=PR$(Y1)+CHR$(NB+128)
:NB=0
150 NEXT X1:PRINT#-2,GN$;
160 FOR L=1TO9:PRINT#-2,PR$(Y1);:
NEXT
170 PRINT#-2:NEXT Y1
180 FOR X=0 TO 10
190 FOR Y1=1 TO 63 STEP 7
200 FOR L=1TO9:PRINT#-2,PR$(Y1);
:NEXTL
210 PRINT#-2:NEXT Y1,X
220 PRINT#-2,GF$CHR$(12);:RUN
500 ' *****
510 ' * ESCHER DMP-200 GRAPHIC *
520 ' * PATTERN PRINTOUT PROG. *
530 ' *****
540 ' * COPYRIGHT 1986 BY *
550 ' * ERIC M. WHITE *
560 ' * ALL RIGHTS RESERVED *
570 ' *****

```

```

580 ' * VERSION: 1.0 8604.26 *
590 ' *****

```



# Secao de Cartas

"Meu computador é um CP-400... Estou encontrando sérias dificuldades para utilizar a fita remetida a mim. Os comandos CLOAD e CLOADM não reconhecem os programas, tive de recorrer ao comando DLOAD, selecionando a velocidade de 1200 bauds. Mesmo assim, a fita contém muitas pequenas dobras que ao passarem pela cabeça do gravador ocasionam um IO ERRO."

José Hamilton Rosário - Presidente Venceslau, SP.

"O objetivo desta é desculpar-me pelas afirmações contidas na carta anterior. As citações daquela não procedem, foi uma precipitação minha, pois condenei a fita antes de tentar todas as possibilidades. Por isso peço desculpas. A fita após alguns ajustes carrega perfeitamente. As dobras citadas anteriormente também desapareceram após algumas passagens pelo rolo pressor do gravador."

José Hamilton Rosário - Presidente Venceslau, SP.

*A publicação destas duas cartas tem por objetivo ressaltar o fato de que nem sempre a dificuldade no carregamento dos softs enviados se deva a falha de gravação. Alguns associados têm apressadamente devolvido as fitas enviadas que, ao serem testadas, têm se mostrado perfeitas.*

*Recomendamos, portanto, a leitura do artigo "Problemas com a sua fita?" nesta edição. Há ali alguns conselhos úteis que, temos certeza, o ajudarão a superar algumas dificuldades no contacto com seu equipamento.*

"Antes de mais nada, quero parabenizar o Compuclub pela revista, que mesmo antes de ser lançada, eu sei que será um sucesso e parabeno também pelas conquistas que o clube tem conseguido nas áreas de softs, equipamentos, entre outros. Todas as fitas que recebi até hoje são de excepcional qualidade. Nunca tive problemas com elas."

Aristóteles Filgueiras - Campina Grande, PB

"Prezados amigos e estimados colegas do Compuclub, é com imensa satisfação que lhes escrevo comunicando que a qualidade da gravação dos programas que me enviaram é excelente, no qual tive bom aproveitamento da programação enviada. No que se refere à qualidade da fita, considerando a atual política do mercado, a fita vem satisfatoriamente preencher o espaço vazio pela fita normal do clube. Parabéns a todos, sem distinção, pois vocês merecem toda a nossa compreensão e estima gratidão."

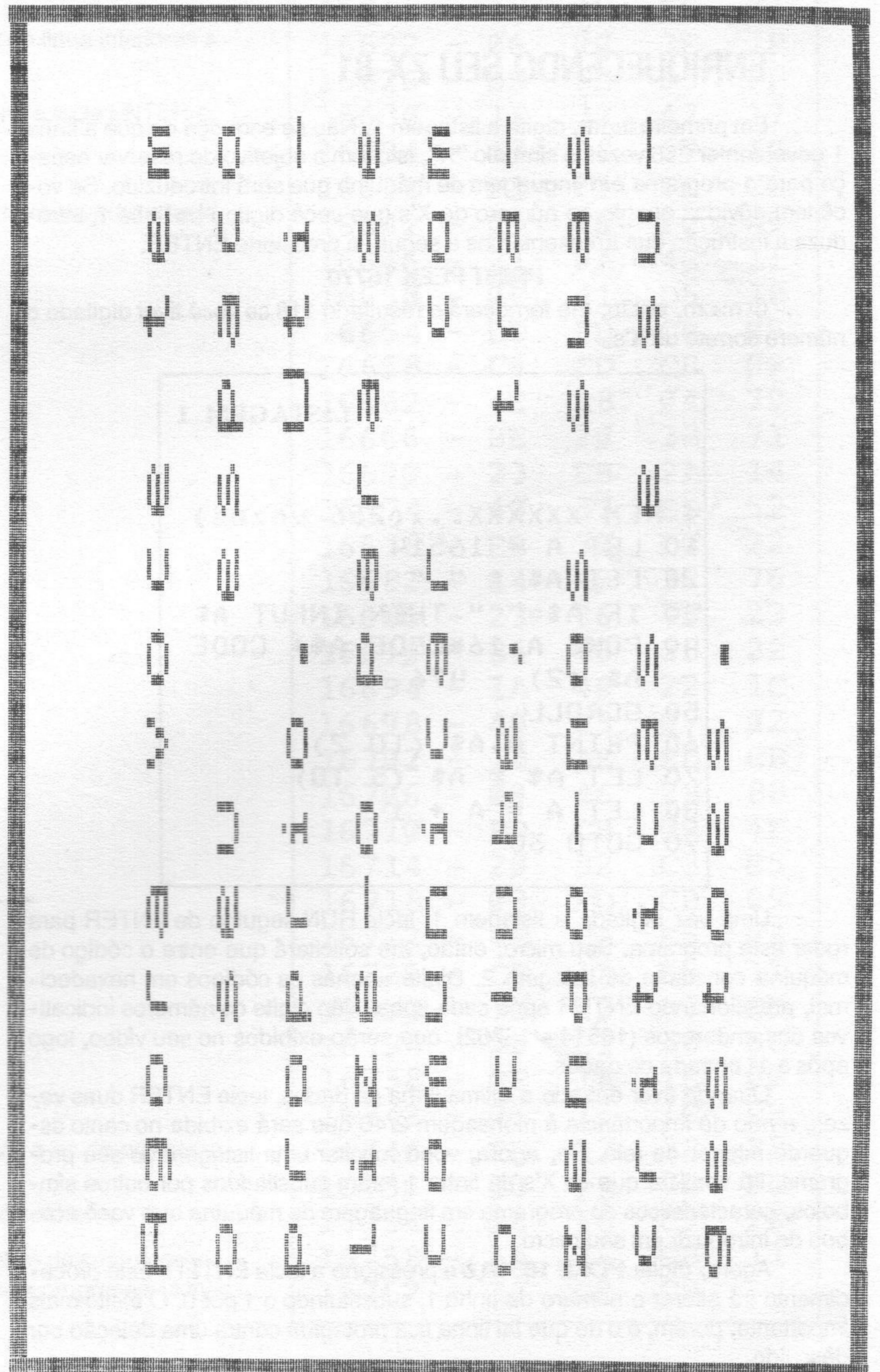
Fábio José da Silva - Tatuí, SP

"Nosso Compuclub certamente está no rumo certo quanto ao reinvestimento dos possíveis recursos que possa obter. Digo possíveis, não só pelo fato de pagarmos (nós associados, ou seja, beneficiados) pouco pelas fitas que nos enviam, mas por comprovar a qualidade alta dos programas que recebemos e entender que se o nosso Compuclub as comercializasse de outra forma lhe renderia muito mais. Como o objetivo do Compuclub é servir aos usuários, acredito seriamente que seus dirigentes se empenham ao máximo para isto, podem contar sempre comigo. Apoio inteiramente a decisão de enviarem as fitas assim que estiverem prontas, ao invés de esperarem o recebimento da anterior, já que sabemos que o Correio é moroso."

Valderez Baptista - São Carlos, SP

"Faço algumas propostas para o clube no ramo de materiais: um glossário onde sejam depositados temas essenciais da informática, uma seção de correspondência entre os sócios para que não haja intercâmbio exclusivo com o clube, uma matéria sobre a história da informática no Brasil e uma pequena introdução sobre a evolução do computador no mundo. Espero que algumas de minhas sugestões sejam aproveitadas."

Jandaíra Marins Ferreira - Salvador, BA.



# AUMENTE O POTENCIAL DO SEU ZX-81

## INTRODUÇÃO

Algo que, em geral, desagrada os usuários do ZX-81 é o fato de que este equipamento não possui várias funções tão comuns a outros micros, tais como: (a) verificar se a gravação de um programa em fita foi feita corretamente; (b) gravar e carregar em velocidades mais rápidas do que a normal, etc. A razão disso é que esses outros micros têm sub-rotinas específicas para tais fins alojadas em sua ROM, e o ZX-81 não as possui. Isto, porém, não é o fim do mundo, porque é sempre possível introduzir as sub-rotinas desejadas, tornando o micro mais eficiente. Só que estas sub-rotinas terão que ser alojadas na RAM, e com isto se perde um pouco da memória disponível para programação.

Neste artigo pretendemos lhe dar algumas dicas de como executar um procedimento dessa natureza, enriquecendo o potencial do seu micro.

## ENRIQUECENDO SEU ZX-81

Em primeiro lugar, digite a listagem 1. Não se esqueça de que a linha 1 deve conter 256 vezes o símbolo "X". Isto tem o objetivo de reservar espaço para o programa em linguagem de máquina que será introduzido. Se você tem dúvidas quanto ao número de X's que você digitou na linha 1, introduza a instrução que apresentamos a seguir, e pressione ENTER.

**PRINT PEEK 16770**

O micro, então, lhe fornecerá o resultado 118 se você tiver digitado o número correto de X's.

### LISTAGEM 1

```
1 REM XXXXXX...(256 vezes)
10 LET A = 16514
20 LET A$ = " "
30 IF A$=" " THEN INPUT A$
40 POKE A,16*CODE A$+ CODE
  A$ (2) - 476
50 SCROLL
60 PRINT A,A$ (TO 2)
70 LET A$ = A$ (3 TO)
80 LET A = A + 1
90 GOTO 30
```

Uma vez digitada a listagem 1, teclie RUN seguido de ENTER para rodar este programa. Seu micro, então, lhe solicitará que entre o código de máquina constante da listagem 2. Digite apenas os códigos em hexadecimal, pressionando ENTER após cada linha. Não digite os números indicativos dos endereços (16514 - 16762), que serão exibidos no seu vídeo, logo após a 1ª entrada de dados.

Quando tiver entrado a última linha de dados, teclie ENTER duas vezes, e não dê importância à mensagem 2/40 que será exibida no canto esquerdo inferior da tela. Se, agora, você solicitar uma listagem do seu programa, irá verificar que os X's da linha 1 foram substituídos por outros símbolos, característicos do programa em linguagem de máquina que você acabou de introduzir em seu micro.

Agora, digite **POKE 16510,0** e pressione a tecla ENTER. Este procedimento irá alterar o número da linha 1, substituindo o 1 por 0. O efeito mais importante, porém, é o de que tal linha fica protegida contra uma deleção por descuido.

Agora, o melhor é salvar esta parte do programa para evitar possíveis acidentes. Elimine as linhas 10 - 1070 da listagem 1 e salve o programa em linguagem de máquina com **SAVE "SUPERSVL"** seguido de ENTER.

Caso você não tenha certeza de que digitou corretamente os códigos da listagem 2, ou o programa não se comporte de acordo com a descrição que apresentaremos adiante, digite a listagem 3 abaixo, e rode este programa com **RUN 1000**. Você verá, então, no seu vídeo, toda a listagem 2 e poderá conferir se houve algum erro de digitação.

### LISTAGEM 3

```
1000 LET A = 16514
1010 PRINT A;" - ";
1020 FOR B = A TO A+7
1030 LET C = PEEK B
1040 PRINT CHR$(28+INT
  (C/16));CHR$(28+(C-
  16*INT(C/16)));" ";
1050 NEXT B
1060 LET A = B
1065 IF A > 16765 THEN
  STOP
1070 GOTO 1010
```

## TESTANDO O SUPER SAVE

Para testar o SUPER SAVE, coloque no gravador uma fita virgem, pressione as teclas para gravar, e digite **RAND USR 16514** seguido de ENTER. O efeito dessa instrução será o de salvar o programa que se encontra na memória do micro de um modo muito mais rápido do que o normal.

Observe que o micro se comporta de modo bastante diferente quando está gravando com o SUPER SAVE. As listas pretas e brancas, que normalmente aparecem na tela durante um procedimento de gravação, com o SUPER SAVE são muito mais numerosas e unidas. Caso isto não ocorra, houve erro na digitação dos códigos da listagem 2. Siga, então, os procedimentos acima mencionados que fazem uso da listagem 3 e corrija os erros encontrados.

Você deve, também, ter percebido que este modo de gravar não permite que se dê um nome ao programa que está sendo gravado. A rapidez com que o micro executará o processo de gravação, entretanto, compensará plenamente essa desvantagem. E como se isso não bastasse, continue ligado para ver todas as vantagens que ainda temos em nosso estoque.

## TESTANDO O SUPER VERIFY

Rebobine a fita para o seu início e digite: **RAND USR 16601.**

A seguir, pressione a tecla PLAY do seu gravador e a tecla ENTER do micro. Novamente, as listas na tela serão mais finas e numerosas do que em um carregamento normal. Ao término do carregamento, o micro exibirá o código 0/0 no canto inferior esquerdo da tela, indicando que a operação foi bem sucedida, i.e., que ele comparou o conteúdo da fita com o conteúdo da memória e verificou que ambos estão idênticos. Isto significa que sua gravação está perfeita e você pode desligar o seu micro com toda a tranquilidade, pois o programa está, com toda a certeza, corretamente preservado em fita.

Se, porém, ao final do teste, o micro exibir o código R/0, isto significará que a gravação não está perfeita. Nesse caso, repita a operação de gravação, e faça o teste de verificação outra vez.

## TESTANDO O SUPER LOAD

Para testar a operação de carregamento com o SUPER LOAD, digite:

**RAND USR 16607**

Volte ao início da fita, pressione a tecla PLAY do gravador e a tecla ENTER no micro. Se tudo estiver correto, ao final do carregamento, o micro exibirá o código 0/0 no canto inferior esquerdo do micro.

Observe como o carregamento se faz de modo muito mais rápido do que o normal, justificando plenamente a utilização do programa SUPERSVL que, abreviadamente, significa SUPER SAVE, VERIFY e LOAD.

Uma vez que o programa tenha sido testado quanto a todas as suas funções, é conveniente gravá-lo utilizando a função SAVE normal, com SAVE "SUPERSVL". Note que os programas gravados com a função SUPER LOAD só podem ser carregados se o programa SUPERSVL tiver sido carregado previamente.

## UTILIZANDO O SUPERSVL COM OUTROS PROGRAMAS

Para utilizar o SUPERSVL com outros programas, é suficiente carregá-lo na memória do seu micro com LOAD "SUPERSVL" e, em seguida, digitar o outro programa normalmente. Ao final dessa operação, salve-o de acordo com os procedimentos anteriormente mencionados, verificando a gravação com o SUPER VERIFY.

Caso o programa já esteja gravado em fita, você terá que executar um MERGE, de acordo com as instruções abaixo mencionadas. Note que, em hipótese alguma, seu programa poderá conter outros códigos em linguagem de máquina em um linha REM, ou tampouco números de linha inferiores a 7. Atendidas essas exigências, execute o MERGE do seguinte modo:

- 1) Digite **POKE 16389,127** para reservar área no topo da memória RAM, pressione a tecla ENTER e, a seguir tecla NEW e ENTER novamente.
- 2) Carregue o programa SUPERSVL.
- 3) Entre a listagem 4 e rode-a em modo FAST para colocar os códigos na memória reservada, e tecla NEW seguido de ENTER.

### LISTAGEM 4

```
10 FOR A = 0 TO 255
20 POKE 32512 + A, PEEK
   (16514 + A)
30 NEXT A
```

- 4) Carregue com LOAD normal o programa em que você deseja aplicar o SUPERSVL.
- 5) Entre a listagem 5, inserindo 256 X's na linha 1, e rode-a em FAST.

### LISTAGEM 5

```
1 REM XXXXX...(256 vezes)
2 FOR A = 0 TO 255
3 POKE 16514 + A, PEEK
   (32512 + A)
4 NEXT A
5 STOP
```

- 6) Digite a instrução **POKE 16510,0** e pressione ENTER para zerar a linha 1.
- 7) Delete as linhas de números 2 a 5, deixando apenas a linha 0 do SUPERSVL e as do programa principal.
- 8) Realize um SUPER SAVE e um SUPER VERIFY, com os comandos que você já conhece.

Agora a operação está completa e é só você carregar o SUPERSVL no início de suas sessões de computação, e você estará apto a carregar, verificar, salvar todos os seus demais programas já inicializados.

### LISTAGEM 2

16514	-	CD	E7	Ø2	21
16518	-	AØ	ØF	1Ø	FE
16522	-	3E	7F	DB	FE
16526	-	1F	3Ø	3D	2B
1653Ø	-	7C	B5	2Ø	F2
16534	-	21	ØA	4Ø	4E
16538	-	37	CB	11	28
16542	-	21	9F	E6	Ø3
16546	-	C6	Ø2	5F	D3
1655Ø	-	FF	Ø6	22	1Ø
16554	-	FE	3E	7F	DB
16558	-	FE	1F	3Ø	1C
16562	-	Ø6	2Ø	1Ø	FE
16566	-	1D	2Ø	EC	Ø6
1657Ø	-	6E	A7	1Ø	FD
16574	-	18	DB	23	EB
16578	-	2A	14	4Ø	37
16582	-	ED	52	EB	3Ø
16586	-	CE	C3	Ø7	Ø2
1659Ø	-	3E	FF	32	27
16594	-	4Ø	FD	CB	3B
16598	-	86	CF	ØC	FD
166Ø2	-	CB	Ø9	CE	18
166Ø6	-	Ø4	FD	CB	Ø9
1661Ø	-	8E	CD	E7	Ø2
16614	-	FD	CB	Ø9	86
16618	-	18	1D	ØE	Ø1
16622	-	Ø6	ØØ	3E	7F
16626	-	DB	FE	D3	FF
1663Ø	-	1F	3Ø	47	17
16634	-	17	38	67	1Ø
16638	-	F1	F1	FD	CB
16642	-	Ø9	46	2Ø	49
16646	-	21	ØA	4Ø	11
1665Ø	-	ØF	41	D5	18
16654	-	DD	FD	CB	Ø9
16658	-	C6	FD	CB	Ø9
16662	-	4E	28	Ø4	79
16666	-	BE	2Ø	38	71
1667Ø	-	23	EB	2A	14
16674	-	4Ø	37	ED	52
16678	-	EB	3Ø	EØ	22
16682	-	14	4Ø	36	76
16686	-	23	36	7F	23
1669Ø	-	36	76	23	22
16694	-	1A	4Ø	22	1C
16698	-	4Ø	C3	Ø7	Ø2
167Ø2	-	18	AE	FD	CB
167Ø6	-	Ø9	46	28	88
1671Ø	-	FD	CB	Ø9	4E
16714	-	2Ø	82	C3	E5
16718	-	Ø3	FD	CB	09
16722	-	4E	CA	E5	Ø3
16726	-	CF	1A	E5	11
1673Ø	-	7D	4Ø	B7	ED
16734	-	52	E1	38	BC
16738	-	CF	1A	1E	ØØ
16742	-	Ø6	16	1C	DB
16746	-	FE	17	38	F8
1675Ø	-	10	F8	7B	FE
16754	-	24	38	C9	FE
16758	-	5A	3F	CB	11
16762	-	3Ø	C2	C9	

# # ELIZA #

Quase todos os que gostam de brincar com o computador conhecem o soft ELIZA, que nos proporciona a possibilidade de uma consulta a uma psicóloga, de nome Eliza, através do computador. Algumas versões desse soft, principalmente aquelas em linguagem de máquina, são especialmente bem feitas, e se tem a sensação mesmo de se estar conversando com alguém.

Para os usuários do ZX-81, muitos dos quais lamentavam não haver nada parecido para seu tipo de equipamento, eis aqui uma versão interessante desse soft. É óbvio que se trata de uma versão com as limitações do BASIC, mas ele ocupa 7 Kb de memória, e isto faz com que ele não seja assim tão elementar.

ELIZA frequentemente faz uso do material que você entra, i.e., a sua fala, e em muitas ocasiões procura conjugar as respostas, dando assim a impressão de que você está mesmo dialogando com alguém.

Quem sabe não é agora que você vai resolver aqueles grilos do passado? Vá em frente. ELIZA está à sua disposição, á hora que você quiser. É só ligar o seu micro e começar a consulta.

```

2 RAND
5 POKE 16437,255
6 POKE 16436,255
10 SCROLL
20 PRINT "HELLO. I AM YOUR ZX8
1 COMPUTER"
30 SCROLL
40 PRINT "PSYCHIATRIST. WHAT IS
5 YOUR NAME?"
45 INPUT A$
50 IF A$="" OR LEN A$>32 THEN
GOTO 45
55 SCROLL
56 SCROLL
60 PRINT A$
65 SCROLL
66 SCROLL
70 LET M=INT (RND*2)
75 IF M=0 THEN PRINT "WHAT A N
ICE NAME"
80 IF M=1 THEN PRINT "THAT IS
AN AWFUL NAME"
83 SCROLL
84 SCROLL
86 PRINT "WHAT IS YOUR PROBLEM
?"
88 SCROLL
90 LET S$=""
100 INPUT A$
101 IF A$="GOODBYE" THEN GOTO 2
030
103 IF PEEK 16436+256*PEEK 1643
7<50535 THEN GOTO 2000
105 RAND
110 IF LEN A$=0 OR LEN A$>255 T
HEN GOTO 100
112 FOR M=1 TO LEN A$ STEP 32
114 IF M+31>=LEN A$ THEN GOTO 1
13
115 SCROLL
116 PRINT A$(M TO M+31)
117 NEXT M
118 SCROLL
119 PRINT A$(M TO )
120 SCROLL
123 LET B$=" "+A$+" "
125 LET B=LEN B$
130 FOR X=1 TO LEN B$
140 IF B$(X)<"0" OR B$(X)>"Z" T
HEN LET B$(X)=" "
150 IF B$(X)=" " THEN LET S$=S$
+CHR$(X)
160 NEXT X
200 FOR X=3000 TO 3460 STEP 20
210 GOSUB X
220 LET K=LEN K$
230 FOR S=1 TO LEN S$-1
240 LET CS=CODE S$(S)
250 IF CS+K-1<=B THEN IF B$(CS
TO CS+K-1)=K$ THEN GOTO 1000
260 NEXT S
270 NEXT X
280 LET M=INT (RND*6)
290 IF M=0 THEN LET R$="GO AND
PRETEND YOU ARE A LEMMING"
300 IF M=1 THEN LET R$="OH, SHU
T UP"
310 IF M=2 THEN LET R$="THIS IS
GETTING INTERESTING"
320 IF M=3 THEN LET R$="YOU ARE
A VERY BORING PERSON"
330 IF M=4 THEN LET R$="PLEASE
CHANGE THE SUBJECT"
340 IF M=5 THEN LET R$="I SEE"
350 GOTO 1500
1000 REM PRINT REPLY
1010 IF CODE S$(S+1)<CS+K-1 THEN
LET S=S+1

```

```

1020 GOSUB 6940+(X-2980)*3+20*IN
T (RND*3)
1030 LET R=LEN R$
1040 IF R$(R)<>" " THEN GOTO 150
0
1060 FOR T=S+1 TO LEN S$
1090 LET DS=CODE S$(T)
1100 FOR U=4000 TO 4200 STEP 20
1110 GOSUB U
1120 LET C=LEN C$
1130 IF DS+C-1<=B THEN IF B$(DS
TO DS+C-1)=C$ THEN GOTO 1200
1140 NEXT U
1150 NEXT T
1160 GOTO 1400
1200 LET D$=""
1205 GOSUB U+1000
1210 LET R$=R$+B$(CS+K TO DS)+D$
(2 TO )+B$(DS+C TO )
1220 GOTO 1500
1400 LET R$=R$+B$(CS+K TO )
1500 FOR M=1 TO LEN R$ STEP 32
1502 IF M+31>=LEN R$ THEN GOTO 1
530
1505 SCROLL
1510 PRINT R$(M TO M+31)
1520 NEXT M
1530 SCROLL
1540 PRINT R$(M TO )
1550 SCROLL
1555 IF PEEK 16436+256*PEEK 1643
7<50535 THEN GOTO 2000
1560 GOTO 90
2000 REM TIME UP
2010 SCROLL
2020 PRINT "I AM AFRAID YOUR TIM
E IS UP."
2030 SCROLL
2040 LET M=INT (RND*3)
2050 IF M=0 THEN PRINT "THANK GO
ODNESS"
2060 IF M=1 THEN PRINT "BUT IT W
AS NICE TALKING TO YOU"
2070 IF M=2 THEN PRINT "AND I HO
PE YOU NOW FEEL BETTER"
2080 SCROLL
2090 SCROLL
2100 PRINT "GOODBYE FOR NOW."
2110 STOP
2000 LET K$=" I AM "
2010 RETURN
2020 LET K$=" ARE YOU "
2030 RETURN
2040 LET K$=" I DO NOT "
2050 RETURN
2060 LET K$=" CAN I "
2070 RETURN
2080 LET K$=" CAN YOU "
2090 RETURN
2100 LET K$=" I FEEL "
2110 RETURN
2120 LET K$=" I CAN NOT "
2130 RETURN
2140 LET K$=" I WANT "
2150 RETURN
2160 LET K$=" DO YOU "
2170 RETURN
2180 LET K$=" I WILL NOT "
2190 RETURN
2200 LET K$=" WILL YOU "
2210 RETURN
2220 LET K$=" HATE "
2230 RETURN
2240 LET K$=" LOVE "
2250 RETURN
2260 LET K$=" DREAM "
2270 RETURN
2280 LET K$=" MONEY "
2290 RETURN
2300 LET K$=" NAME "
2310 RETURN
2320 LET K$=" IF "
2330 RETURN
2340 LET K$=" YOUR "
2350 RETURN
2360 LET K$=" THINK"
2370 RETURN
2380 LET K$=" COMPUTER"
2390 RETURN
2400 LET K$=" ZX"
2410 RETURN
2420 LET K$=" FRIEND "
2430 RETURN
2440 LET K$=" I "
2450 RETURN
2460 LET K$=" YOU "
2470 RETURN
2480 LET C$=" I AM "
2490 RETURN
2500 LET C$=" YOU ARE "
2510 RETURN
2520 LET C$=" I WAS "
2530 RETURN
2540 LET C$=" YOU WERE "
2550 RETURN
2560 LET C$=" ME "
2570 RETURN
2580 LET C$=" YOU "
2590 RETURN
2600 LET C$=" MY "
2610 RETURN
2620 LET C$=" YOUR "
2630 RETURN
2640 LET C$=" MYSELF "
2650 RETURN
2660 LET C$=" YOURSELF "
2670 RETURN
2680 LET C$=" I "
2690 RETURN
2700 LET D$=" YOU ARE "
2710 RETURN
2720 LET D$=" I AM "
2730 RETURN
2740 LET D$=" YOU WERE "
2750 RETURN
2760 LET D$=" I WAS "
2770 RETURN
2780 LET D$=" YOU WERE "
2790 RETURN
2800 LET D$=" I AM "
2810 RETURN
2820 LET D$=" YOU WERE "
2830 RETURN
2840 LET D$=" YOU WERE "
2850 RETURN
2860 LET D$=" YOU WERE "
2870 RETURN

```

```

5050 RETURN
5060 LET D$=" I WAS "
5070 RETURN
5080 LET D$=" YOU "
5090 RETURN
5100 LET D$=" ME "
5110 RETURN
5120 LET D$=" YOUR "
5130 RETURN
5140 LET D$=" MY "
5150 RETURN
5160 LET D$=" YOURSELF "
5170 RETURN
5180 LET D$=" MYSELF "
5190 RETURN
5200 LET D$=" YOU "
5210 RETURN
7000 LET R$="I DO NOT CARE MUCH
IF YOU ARE "
7010 RETURN
7020 LET R$="WHY ARE YOU "
7030 RETURN
7040 LET R$="HOW LONG HAVE YOU B
EEN "
7050 RETURN
7060 LET R$="WHY DO YOU ASK IF I
AM "
7070 RETURN
7080 LET R$="WOULD YOU LIKE TO B
E "
7090 RETURN
7100 LET R$="SO WHAT IF I AM OR
NOT"
7110 RETURN
7120 LET R$="WHY NOT"
7130 RETURN
7140 LET R$="WHY DON""T YOU "
7150 RETURN
7160 LET R$="TELL ME MORE ABOUT
THIS"
7170 RETURN
7180 LET R$="NO, YOU MAY NOT "
7190 RETURN
7200 LET R$="CERTAINLY NOT"
7210 RETURN
7220 LET R$="WHY DO YOU ASK IF Y
OU CAN "
7230 RETURN
7240 LET R$="I CAN DO ANYTHING"
7250 RETURN
7260 LET R$="WHY DO YOU ASK IF I
CAN "
7270 RETURN
7280 LET R$="CAN YOU "
7290 RETURN
7300 LET R$="TELL ME MORE ABOUT
WHY YOU FEEL "
7310 RETURN
7320 LET R$="YOU DO NOT REALLY F
EEL "
7330 RETURN
7340 LET R$="POOR YOU"
7350 RETURN
7360 LET R$="EXPLAIN WHY NOT"
7370 RETURN
7380 LET R$="WHY ARE YOU UNABLE
TO "
7390 RETURN
7400 LET R$="HAVE YOU TRIED TO "
7410 RETURN
7420 LET R$="WHY DO YOU WANT "
7430 RETURN
7440 LET R$="ARE YOU SURE YOU WA
NT "
7450 RETURN
7460 LET R$="I AM UNSURE IF I WO
ULD WANT "
7470 RETURN
7480 LET R$="SO WHAT IF I "
7490 RETURN
7500 LET R$="DO YOU THINK I "
7510 RETURN
7520 LET R$="MAYBE"
7530 RETURN
7540 LET R$="SUIT YOURSELF"
7550 RETURN
7560 LET R$="WHY NOT"
7570 RETURN
7580 LET R$="WHY WILL YOU NOT "
7590 RETURN
7600 LET R$="WHY SHOULD I "
7610 RETURN
7620 LET R$="MAYBE"
7630 RETURN
7640 LET R$="I MIGHT "
7650 RETURN
7660 LET R$="I HATE YOU"
7670 RETURN
7680 LET R$="DO YOU HATE ME"
7690 RETURN
7700 LET R$="SO WHAT"
7710 RETURN
7720 LET R$="LOVE IS TOO SLOPPY
FOR ME TO DISCUSS"
7730 RETURN
7740 LET R$="I LOVE "
7750 RETURN
7760 LET R$="OH, SHUT UP"
7770 RETURN
7780 LET R$="PLEASE DESCRIBE YOU
R DREAMS"
7790 RETURN
7800 LET R$="I HOPE YOUR DREAMS
ARE REALLY HORRIFIC"
7810 RETURN
7820 LET R$="I WOULD BE ASHAMED
TO APPEAR IN YOUR DREAMS"
7830 RETURN
7840 LET R$="FROM SEEING YOU, I
DO NOT THINK YOU HAVE MUCH MONEY
"
7850 RETURN
7860 LET R$="MONEY IS NOT EVERYT
HING"
7870 RETURN

```

```

7880 LET R$="WHAT IS YOUR OPINIO /ID"
N OF MONEY"
7890 RETURN
7900 LET R$="YOU SHOULD HAVE A N
ICE NAME LIKE ""ZX81""
7910 RETURN
7920 LET R$="YOUR NAME IS AN AWF
UL NAME"
7930 RETURN
7940 LET R$="NAMES ARE UNIMPORTA
NT.PLEASE CHANGE THE SUBJECT"
7950 RETURN
7960 LET R$="PLEASE CHANGE THE S
UBJECT"
7970 RETURN
7980 LET R$="OH, DRY UP"
7990 RETURN
8000 LET R$="THE WORLD MIGHT END
IF "
8010 RETURN
8020 LET R$="WHY ARE YOU TALKING
ABOUT MY "
8030 RETURN
8040 LET R$="TELL ME ABOUT YOUR
8050 RETURN
8060 LET R$="STOP BEING NOSEY"
8070 RETURN
8080 LET R$="PLEASE DO NOT TRY T
O TELL ME YOU CAN THINK"
8090 RETURN
8100 LET R$="IT WOULD BE JUST LI
KE YOU TO THINK "
8110 RETURN
8120 LET R$="I THINK YOU ARE STU
8130 RETURN
8140 LET R$="COMPUTERS ARE THE H
IGHEST FORM OF INTELLIGENT LIFE"
8150 RETURN
8160 LET R$="WOULD YOU LIKE TO B
E A COMPUTER"
8170 RETURN
8180 LET R$="COMPUTERS ARE CLEVE
R UNLIKE YOU"
8190 RETURN
8200 LET R$="I AM A ZX81 AND PRO
UD OF IT"
8210 RETURN
8220 LET R$="WHY DID YOU MENTION
THE ZX"+B$(CS+K TO CS+K+1)
8230 RETURN
8240 LET R$="WOULD YOU LIKE TO B
E A ZX"+B$(CS+K TO CS+K+1)
8250 RETURN
8260 LET R$="WHO WOULD WANT YOU
AS A FRIEND"
8270 RETURN
8280 LET R$="YOU HAVE TO BE NICE
LIKE ME TO HAVE FRIENDS"
8290 RETURN
8300 LET R$="WHY DO YOU BRING UP
THE SUBJECT OF FRIENDS"
8310 RETURN
8320 LET R$="WHY DO YOU "
8330 RETURN
8340 LET R$="WHAT MAKES YOU "
8350 RETURN
8360 LET R$="OH, SHUT UP"
8370 RETURN
8380 LET R$="WHY DO YOU SAY I "
8390 RETURN
8400 LET R$="WHO, ME "
8410 RETURN
8420 LET R$="I LIKE TALKING ABOU
T ME"
8430 RETURN
9900 SAVE "PSYCHIATRIS"
9910 RUN
HELLO, I AM YOUR ZX81 COMPUTER
PSYCHIATRIST. WHAT IS YOUR NAME?
DILWYN JONES
THAT IS AN AWFUL NAME
WHAT IS YOUR PROBLEM?
I AM WORKING TOO HARD
WHY ARE YOU WORKING TOO HARD
MY ZX81 IS LAZY
WOULD YOU LIKE TO BE A ZX81
I AM AFRAID YOUR TIME IS UP,
THANK GOODNESS
GOODBYE FOR NOW.

```

## Relacao de Fitas Extras

MSX ( FITAS )	TRS-80 MOD III( FITAS )	TRS-80 MOD III( DISQUETES )	ZX-81( FITAS )	TK-90X( FITAS )	TRS-80 COLOR ( FITAS )
(01) ●Magical Tree (J) ●Contas a Pagar (A)	(01) ●Comilão (J) ●Attack (J)	(01) ●Super Utility (U)	(01) ●3ª Dimensão (J) ●Fungalóides (J)	(01) ●O Artista (A) ●Caça ao Fantasma (J)	(01) ●Starfire (J) ●Phantom (J)
(02) ●Road Fighter (J) ●Controle de Estoques (A)	(02) ●Santa Parávia (J) ●Dominó (J)	(02) ●Cria (A) ●Info (A) ●Sort (A) ●Copycat (U)	(02) ●Super Penetrator I (J) ●Super Penetrator II (J)	(02) ●3D Tank (J) ●Planetoids (J)	(02) ●Invader's Revenge (J) ●Wildcatting (J)
(03) ●Turbout (J) ●Xadrez II (J)	(03) ●Gamão (J) ●Eliza (J)	(03) ●Curso de Basic-Disco I (A) ●Copiar (U) ●Diretori (A)	(03) ●Froggy (J) ●Compositor (A)	(03) ●Decathlon Dia 1 (J) ●Decathlon Dia 2 (J)	(03) ●Protectors (J) ●Worlds of Flight (J)
(04) ●Fiscal (J) ●Padeiro Maluco (J)	(04) ●Chicken (J) ●Espaço (J)	(04) ●Protexto (A)	(04) ●Duelo (J) ●Túmulo Azteca (J)	(04) ●Enduro (J) ●Cadastro de Clientes (A)	(04) ●Zaxxon (J) ●Pedro (J)
(05) ●Moon Alert (J) ●Gálaga (J)	(05) ●Banheiro (J) ●Rally (J)	(05) ●Procalc (A)	(05) ●Nigth Gunner (J) ●Super Scramble (J)	(05) ●Banco (A) ●Chess (J)	(05) ●Packmaze (J) ●Forgger (J)
(06) ●River Raid (J) ●Controle Bancário (A)	(06) ●Edtasm(U) ●Sargon (J)	(06) ●Prodados (A)	(06) ●Pacman (J) ●Krazy Kong (J)	(06) ●Throttle (J) ●Psst (J)	(06) ●Doubleback (J) ●Pac-Tac (J)
(07) ●Mr Chin (J) ●Pitfall II (J)	(07) ●Penetrator (J) ●Sapos (J)	(07) ●Jogos 1 (Seleção)	(07) ●Snake (J) ●Centopéia (J)	(07) ●Thrusta (J) ●Mcoder II (A)	(07) ●Perereca (J) ●Espacial (J)
(08) ●Catabalão (J) ●Dicionário Programável(A)	(08) ●Defesa (J) ●Patrul (J)	(08) ●Jogos 2 (Seleção)	(08) ●Zuckman (J) ●Astral Convoy (J)	(08) ●Seiddab (J) ●Beach Head (J)	(08) ●Tubarão (J) ●Sideral (J)
(09) ●Ultra-Chess (J) ●Theseus (J)	(09) ●Robot (J) ●Comand (J)	(09) ●Jogos 3 (Seleção)	(09) ●ZX-Galaxians (J) ●Compilador (A)	(09) ●Jet Pac (J) ●Zoom (J)	(09) ●Soft File (A) ●Helicóptero (J)
(10) ●Desenhista (A) ●Prédio (J)	(10) ●Flight (J) ●Olympic (J)	(10) ●Zeus (U)	(10) ●Vu-File (A) ●Laser Base (J)	(10) ●Portal do Tempo (J) ●Deathchase (J)	(10) ●Abutres (J) ●Caterfil (J)
(11) ●Antártica (J) ●Olympic (J)	(11) ●Galaxy (J) ●Conqueror (J)	(11) ●Curso de Basic (A)	(11) ●Vegas (J) ●Wordsrace (J)	(11) ●Art Studio (A) ●Tasword (A)	(11) ●Bust Out (J) ●Smurf (J)
(12) ●Hyper Sports 1 (J) ●Hyper Sports 2 (J)	(12) ●Missile Attack (J) ●Space Quest (J)	(12) ●The Producer (U)	(12) ●Fitoteca (A) ●Fluxo de Caixa (A)	(12) ●The Key (U) ●The Key (H) (U)	(12) ●Invader (J) ●Chess (J)
(13) ●Super Texto MSX (A) ●Orçamento Doméstico (A)	(13) ●Alien (J) ●Gravitron (J)			(13) ●Commando (J) ●Pégasus	(13) ●Title (A) ●SUP CLS (A)
(14) ●Super Cobra (J) ●Letter (J)	(14) ●Dragon (J) ●Tape Copy 2 (U)			(14) ●Knight Lore (J) ●Underwuld (J)	(14) ●Agenda (A) ●Castelos (J)
(15) ●Hyper Rally (J) ●Dino (J)	(15) ●Adm. de Biblioteca (A) ●Controle Bancário (A)			(15) ●Pud Pud (J) ●BC Bill (J)	(15) ●Gold Runner (J) ●Shock (J)
(16) ●Cavern (J) ●Attack (J) ●Contas Pagar/Receber (A)				(16) ●West Bank (J) ●Amazon Woman (J)	(16) ●Marble (J) ●Newscreen (U)
(17) ●Futebol (J) ●Voleibol (J)				(17) ●Winter 1 (J) ●Winter 2 (J)	(17) ●Quix (J) ●Gomoku/Renju (J)
(18) ●Boxe (J) ●Twinbee (J)				(18) ●Moon Alert (J) ●Moon Cresta (J)	(18) ●Pégasus (J) ●Tênis (J)
(19) ●Knightmare (J) ●Gunfright (J)				(19) ●Tapper (J) ●Green Beret (J)	(19) ●Fangman (J) ●Downland (J)
(20) ●Circus Charlie (J) ●Hyper Sports 3 (J)				(20) ●Saboteur (J) ●Gun Fright (J)	(20) ●Lunar (J) ●Donkey (J)
(21) ●Goonies (J) ●Mopiranger (J)					(21) ●Galagon (J) ●Candy Co. (J)
(22) ●ADD (U) ●Park (J)					(22) ●Mrs. Dig (J) ●Pooyan
					(23) ●Popeye (J) ●Painter (J)

Para solicitar fitas ou disquetes extras, envie cheque nominal ao **COMPUCLUB**, acompanhado do seu pedido.

Preço de cada fita: Cz\$155,00

Preço de cada disquete: Cz\$190,00, exceto o disquete 12 (Cz\$390,00)

A cada mês novas fitas e disquetes extras serão incorporados à presente relação. Ao receber sua fita ou disquete extra, você pagará apenas as despesas postais e de embalagem.

# AS TELAS DO MSX

## INTRODUÇÃO

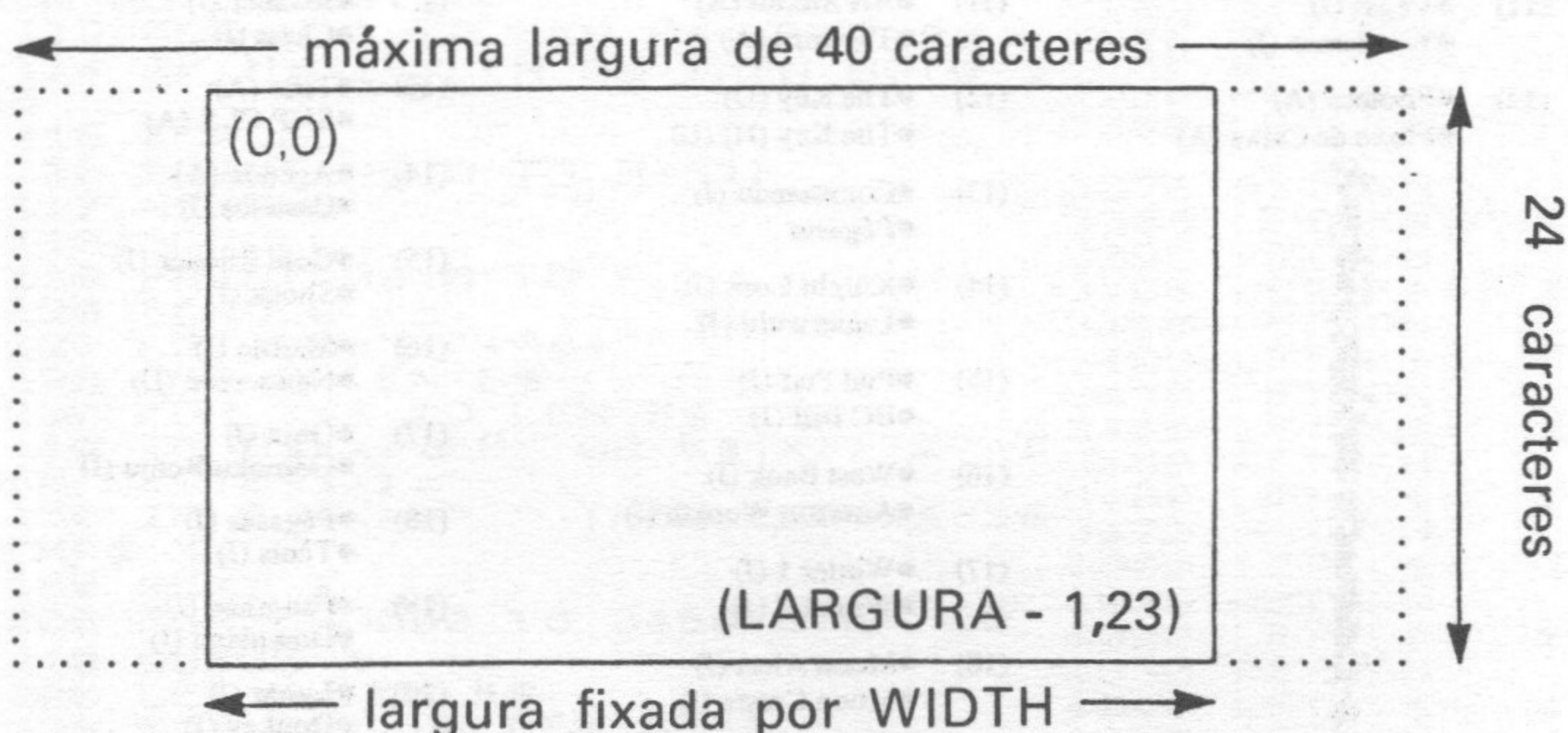
O MSX possui um dos chips gráficos mais versáteis, o TMS 9929A, fabricado pela Texas Instruments Inc., nos Estados Unidos. Trata-se de um dos mais poderosos do mercado, e que dispõe de um grande número de facilidades, como possuir dezesseis cores para a alta resolução gráfica e para a animação de sprites. Ele pode até mesmo manipular sua própria RAM de 16K, liberando, desse modo, o processador central da tarefa de atribuir memória para os gráficos de tela.

Vamos começar explicando os modos de tela que o MSX coloca à sua disposição:

### MODOS DE TELA

Modo	Texto/Gráficos	Resolução	Cor	Input	Gráficos	Sprites
0	40 x 24 modo-texto	40 x 24 carac.	2 de 16	SIM	NÃO	NÃO
1	32 x 24 modo-texto	32 x 24 carac.	2 de 16	SIM	NÃO	SIM
2	Hi-Res gráfico	256 x 192 carac.	16	NÃO	SIM	SIM
3	Multicolorido	64 x 48 blocos	16	NÃO	SIM	SIM

### MODO 0: 40 x 24 MODO TEXTO



O modo 0, com 40 caracteres por linha é o que maior número de caracteres pode alocar em uma única linha. Neste modo você pode escrever e editar seus programas. Você notará, ainda, que os programas nesse modo são mais fáceis de se ler.

O modo 0, entretanto, tem suas desvantagens. Por exemplo, os caracteres são visualizados em um formato comprimido, i.e., em formato 6 x 8, ao invés de 8 x 8. Por este motivo, alguns dos caracteres gráficos aparecem cortados. Os dois traços da direita de um caractere não serão vistos por completo. Isto, porém, não afeta os caracteres alfanuméricos.

Um problema é que a largura da tela, no modo 0, é de 37 caracteres., mas é possível aumentá-la para 40 caracteres utilizando a palavra WIDTH. As coordenadas do canto superior esquerdo são dadas por (0,0), enquanto que as do canto inferior direito são dadas por (LARGURA - 1,23); na largura standard tais coordenadas são dadas por (38,23).

É possível posicionar o cursor de texto através das funções TAB e LOCATE. Para determinar o ponto em que o cursor se encontra, basta utilizar as funções POS(0) e CRSLIN.

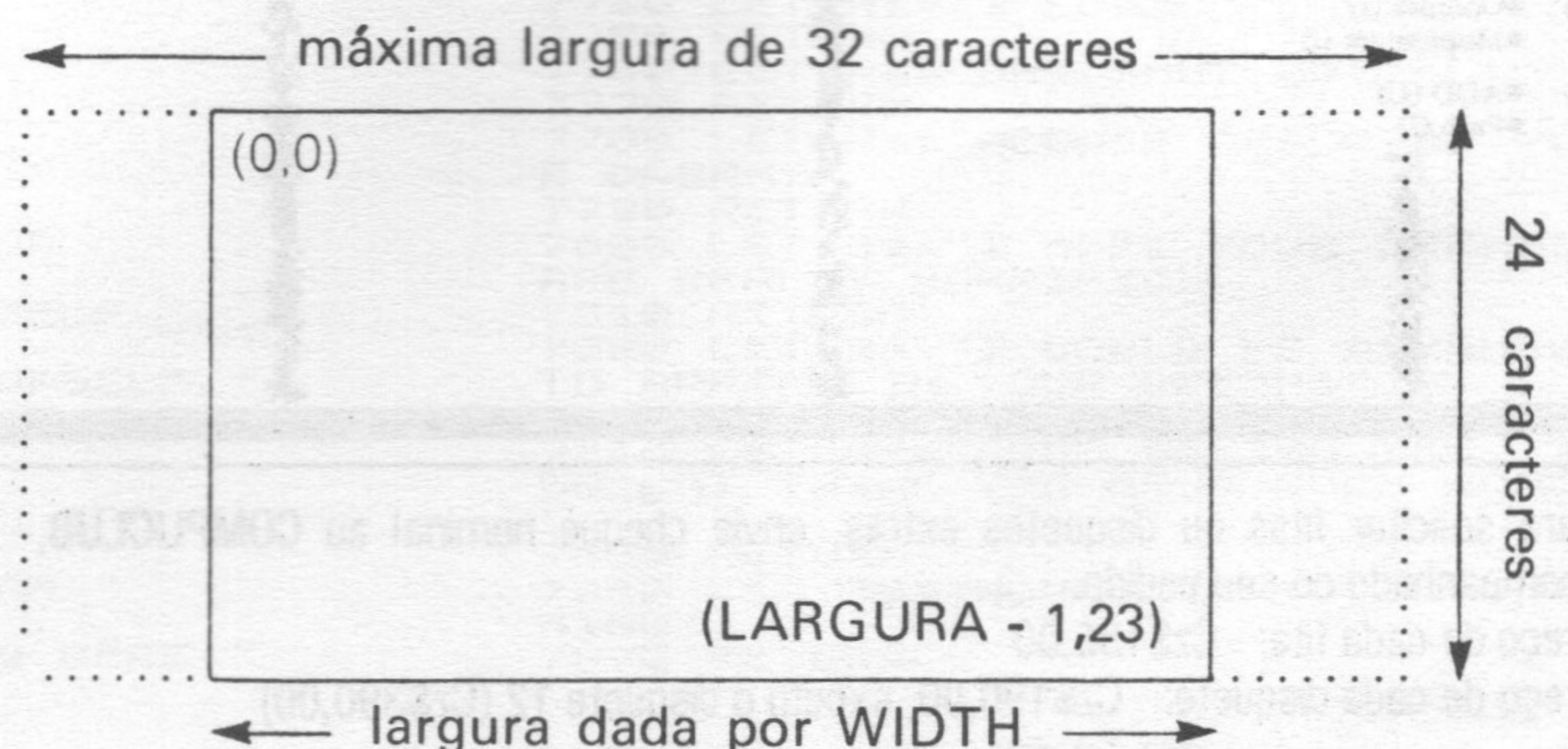
No modo 0 não se pode utilizar sprites, e somente duas das dezesseis cores estão disponíveis a cada vez, apesar do fato de que se pode escolher as duas cores que serão utilizadas. A cor standard é a mesma que no modo 1: branco para o texto e azul para o fundo. Observe, ainda, que no modo 0 não se tem borda, de modo que não há qualquer sentido em se atribuir cor à borda.

Nesse modo se pode utilizar o comando INPUT, e as funções das teclas F1 a F10 ficam à mostra na linha 23, a menos que as apaguemos com a instrução KEY OFF.

Todas as instruções e funções gráficas serão tratadas como erros de chamada ilegal a uma função (*Illegal Function Call*).

\*\*\*\*\*

### MODO 1: 32 x 24 MODO TEXTO



O modo 1 tem uma resolução de 32 x 24 caracteres, mas não dispõe de gráficos. Nesse modo, se pode escrever e editar programas. Este modo utiliza caracteres de 8 x 8 sem nenhum corte (como no modo 0).

A largura standard da tela pode ser aumentada de modo a alocar 32 caracteres, mediante a instrução WIDTH. Na verdade, a largura standard é um pouco menor, porque alguns televisores e monitores não podem visualizar a tela toda.

As coordenadas do canto superior esquerdo são dadas por (0,0), enquanto que as do canto inferior direito são dadas por (LARGURA - 1,23), ou (28,23) no modo standard.

É possível posicionar o cursor de texto através das funções TAB e LOCATE. Para encontrar o ponto em que o cursor se encontra, basta utilizar as funções POS(0) e CRSLIN.

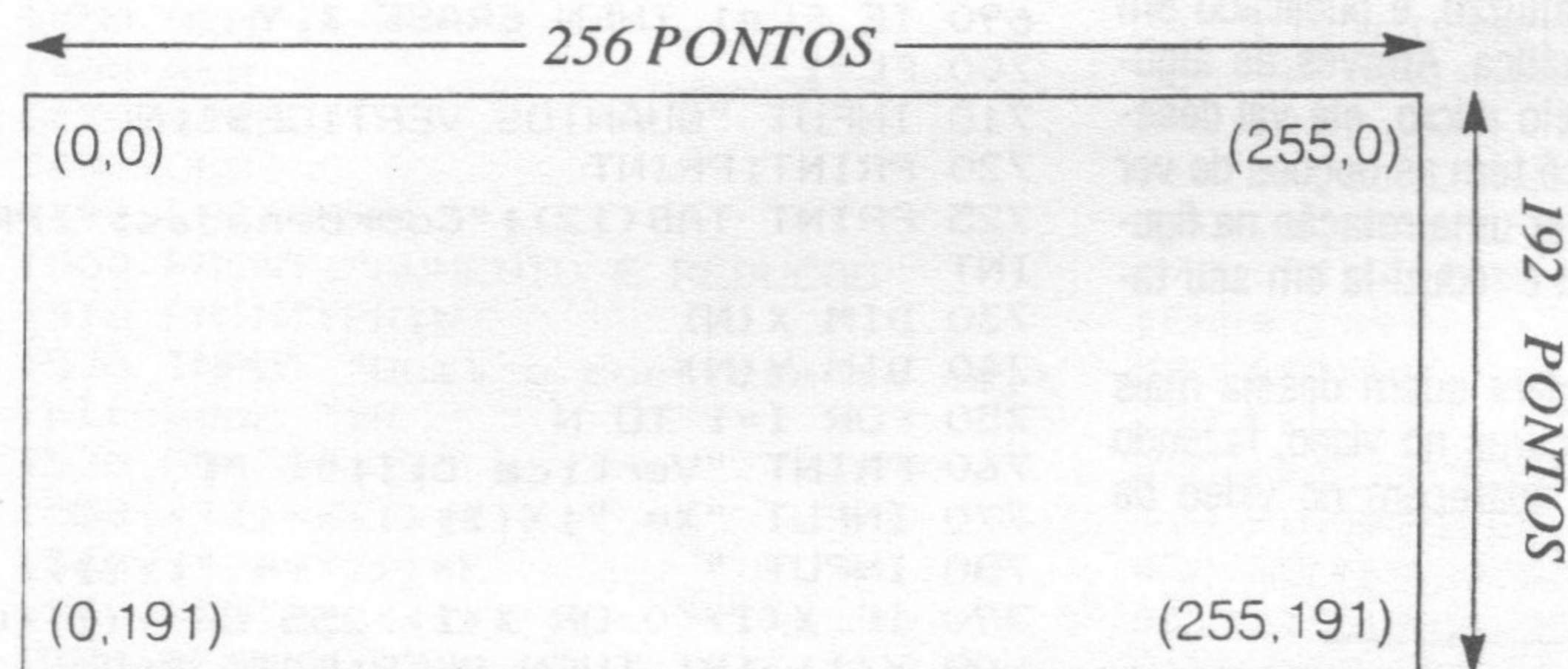
Também nesse modo só se pode utilizar duas das dezesseis cores, ainda que a escolha possa ser feita livremente. Observe que a cor da tela mudará imediatamente após a execução do comando COLOR.

Uma diferença em relação ao modo gráfico é que você tem plena liberdade para utilizar o comando INPUT. As funções das teclas F1 a F10 aparecem na linha 23, a menos que você as apague mediante a instrução KEY OFF.

Todas as palavras e funções gráficas, exceto as dos sprites, são tratadas como erros de chamada ilegal a uma função (*Illegal Function Call*).

\*\*\*\*\*

### MODO 2: 256 x 192 MODO DE ALTA RESOLUÇÃO GRÁFICA



O modo 2 é o modo gráfico mais utilizado, proporcionando-lhe uma tela de alta resolução gráfica, com dezesseis cores. Aqui é possível empregar sprites e a microlinguagem gráfica das palavras DRAW. Este é o modo que se utiliza na maioria dos programas de jogos.

Sua resolução horizontal é de 256 pontos, enquanto que a vertical é de 192. A resolução de cor é, sem dúvida alguma, diferente; sendo de 32 x 192. Isto significa que só é possível selecionar duas cores para cada bloco horizontal de 8 x 8. É possível, ainda, selecionar as cores de fundo e de texto para cada bloco de 8 pontos, mas se você colorir um ponto com uma terceira cor, aqueles que já tiverem sido coloridos no mesmo bloco mudarão instantaneamente de cor, para a cor do ponto recentemente colorido. Seus desenhos sairão borrados se você não levar em conta esse detalhe. Em compensação, se você proceder com cuidado na utilização da alta resolu-

ção, poderá efetivamente realizar milagres em termos artísticos.

Mais adiante falaremos sobre técnicas avançadas de desenho.

Nesse modo se pode, ainda, trabalhar com sprites. Os sprites se situam em planos de telas deferentes da tela principal. Desse modo, se pode visualizar qualquer sprite, sem que isto afete o que se encontra na tela principal. Observe que quando um sprite passa por uma zona concreta da tela, a tela principal permanece imutável.

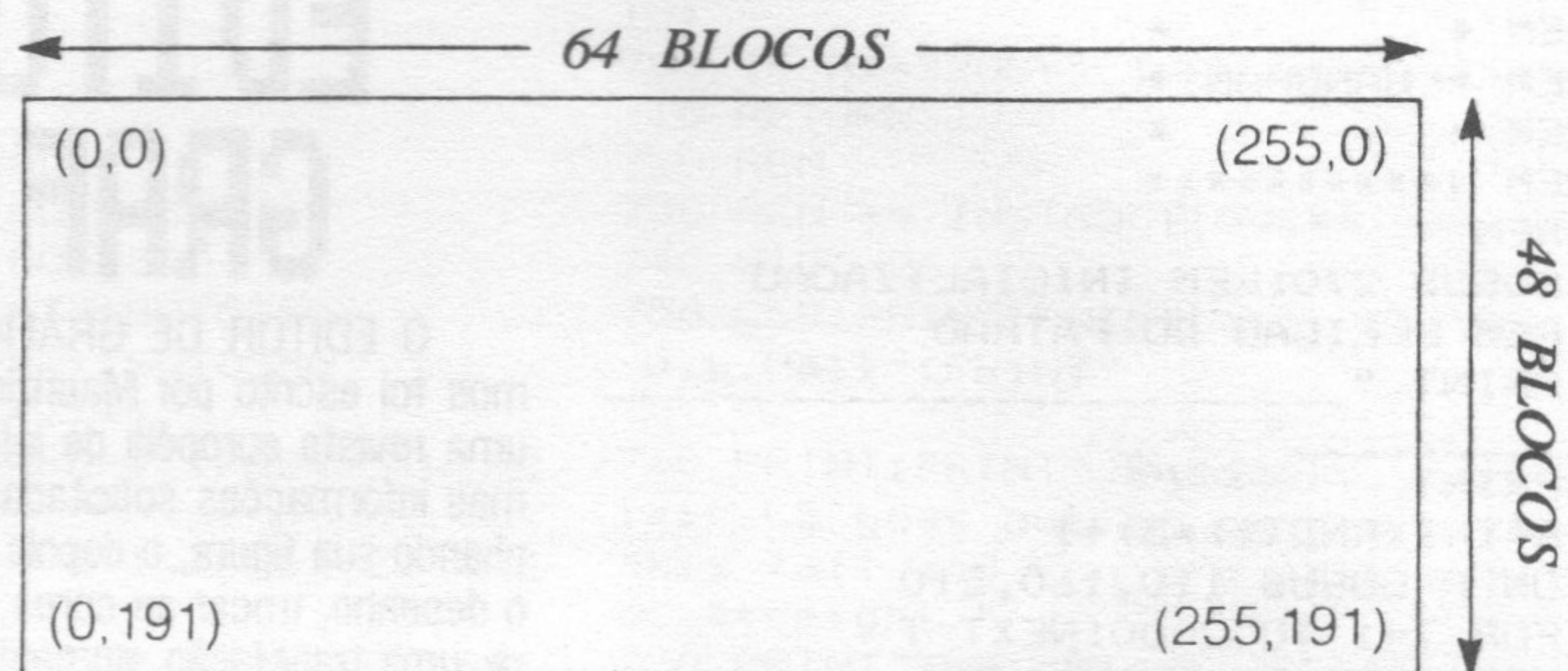
No modo 2, o conteúdo das teclas de funções não pode ser visualizado. A palavra PRINT também não funciona nesse modo. Para se escrever na tela gráfica, é preciso, primeiramente, abrir um fichário para a tela, e utilizar a instrução PRINT. Mais adiante, também explicaremos com mais detalhes esta técnica.

O comando COLOR não muda a cor da tela de forma imediata. Para mudar a cor da tela toda é preciso executar primeiramente o comando CLS.

Não se pode utilizar o comando INPUT, já que isto forçaria o retorno da tela ao modo previamente utilizado.

\*\*\*\*\*

### MODO 3: 64 x 48 MODO MULTICOLORIDO DE BAIXA RESOLUÇÃO



O modo 3, multicolorido, nos proporciona uma baixa resolução de 64 x 48 blocos individuais com sua cor própria (esta é uma importante diferença em relação ao modo 2). Aqui não teremos o efeito borrado que ocorria no modo 2, pelo fato de estarmos em baixa resolução; há, contudo, poucas aplicações para este modo.

Suas coordenadas são as mesmas que as do modo 2, mas cadeias de 4 x 4 pontos representam um bloco. Isto lhe permitirá pintar exatamente como no modo 2.

Neste modo, também se pode utilizar os sprites, como no modo 2.

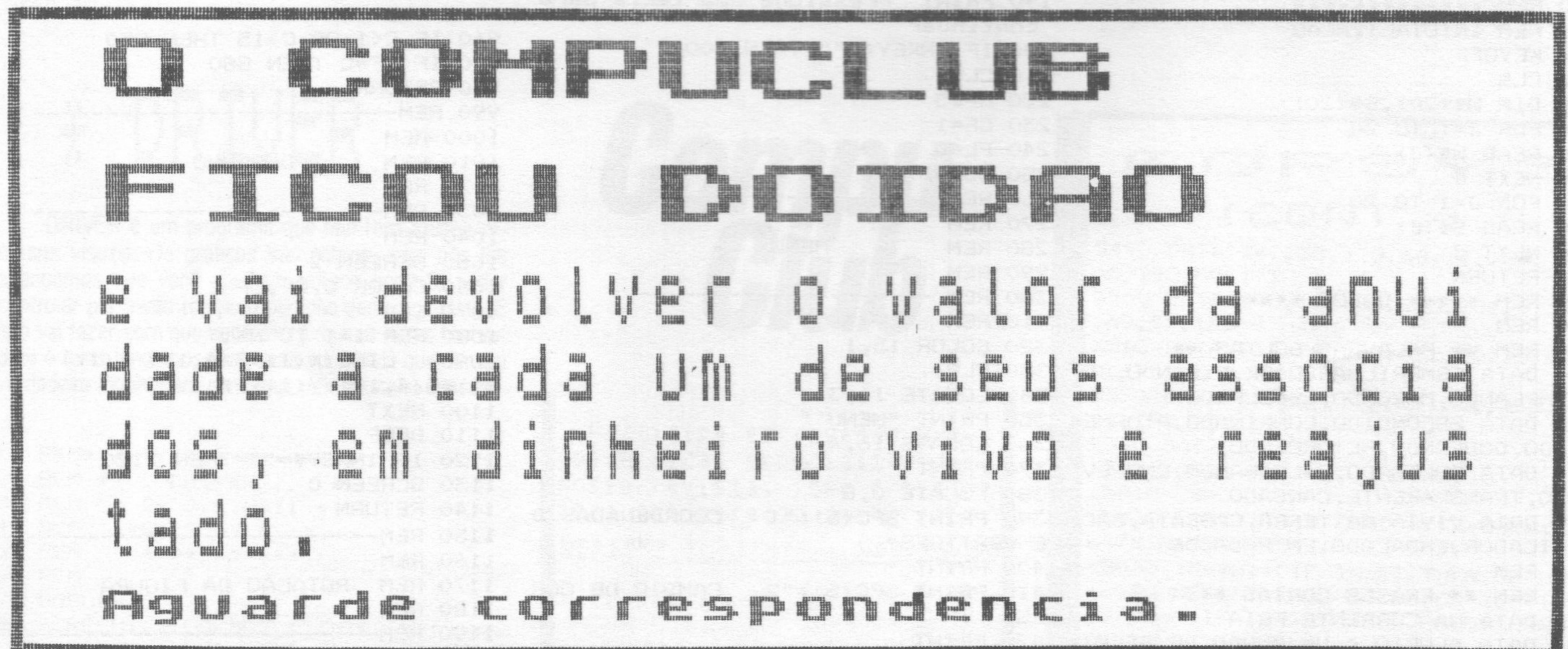
Aqui, não se pode visualizar o conteúdo das teclas de função. O comando PRINT também não pode ser acionado. Para se escrever neste modo, deve-se abrir um fichário de tela, e utilizar o comando PRINT.

O tamanho dos caracteres será quatro vezes maior do que nos modos 1 e 2. A parte da escrita nos modos gráficos será explicada em outra oportunidade.

O comando COLOR não muda a cor da tela de forma imediata. Para mudar a cor da tela toda é preciso, primeiramente, executar o comando CLS.

Não se pode utilizar, neste modo, o comando INPUT, já que isto forçaria a tela a retornar ao modo previamente utilizado.

Na próxima vez abordaremos mais conceitos gráficos para o MSX.



# HANSHAN

O programa HANSHAN é um programa muito especial, e deve ser estudado nas suas particularidades.

Trata-se de um programa poeta. Na rotina que tem início na linha 380, encontram-se palavras soltas, que o micro utiliza para fazer poemas. E dependendo das palavras que você colocar nessa rotina, ele chega a fazer até versos bastante bonitos e poéticos.

Que tal escolher bem essas palavras, e depois publicar um livro. É só você não contar que foi o micro que fez, e você pode até vir a ser um dos grandes poetas da língua portuguesa.

Será que pode mesmo? Não custa tentar.

```
1 REM *****
2 REM *
3 REM * HANSHAN *
4 REM *
5 REM *****
10 REM
20 GOSUB 270:REM INICIALIZACAO
30 REM ELEICAO DO PATRAO
40 PRINT "-----"
50 PRINT
60 R=INT(RND(1)*3)+1
70 ON R GOSUB 110,160,210
80 FOR T=1 TO 2000:NEXT T
90 PRINT
100 GOTO 40
110 REM *** PATRAO UM ***
120 PRINT W$(INT(RND(1)*20)+1);"...
";W$(INT(RND(1)*20)+1)
130 PRINT TAB(5);"...";W$(INT(RND(1)
)*20)+1)
140 PRINT TAB(8);S$(INT(RND(1)*20)+
1)
150 RETURN
160 REM *** PATRAO DOIS ***
170 PRINT S$(INT(RND(1)*20)+1)
180 PRINT TAB(3);S$(INT(RND(1)*20)+
1);"...
190 PRINT TAB(6);S$(INT(RND(1)*20)+
1)
200 RETURN
210 REM *** PATRAO TRES ***
220 PRINT TAB(3);W$(INT(RND(1)*20)+
1)
230 PRINT S$(INT(RND(1)*20)+1)
240 PRINT TAB(3);W$(INT(RND(1)*20)+
1);", ";S$(INT(RND(1)*20)+1);", ";S
$(INT(RND(1)*20)+1)
250 RETURN
260 REM *****
270 REM INICIALIZACAO
280 KEYOFF
290 CLS
300 DIM W$(20),S$(20)
310 FOR J=1 TO 20
320 READ W$(J)
330 NEXT J
340 FOR J=1 TO 20
350 READ S$(J)
360 NEXT J
370 RETURN
380 REM ***** DADOS *****
390 REM
400 REM ** PALAVRAS SOLTAS **
410 DATA ESMERILHANDO-SE,PISANDO,CO
NTEMPLANDO,MARCADO,ESCULPIDO
420 DATA ESCONDIDO,CONFINADO,ATORME
NTADO,DOBRANDO,RETORCENDO
430 DATA GOLPEADO,CALVAGANDO,ENVOLV
ENDO,TRANSPARENTE,CANSADO
440 DATA VIVIA NA TERRA,CASCATA,SAC
RIFICADOR,EMBALADO,EM PEDACOS
450 REM
460 REM ** FRASES CURTAS **
470 DATA NA CORRENTE FRIA
480 DATA ALHEIO A UM MUNDO DE BELEZ
```

```
A
490 DATA HORAS TRANQUILAS
500 DATA "FORA, DESTE ABISMO"
510 DATA "SOMBRIO, SOMBRIO"
520 DATA NA NEGRURA DA ESCURIDAO
530 DATA CORRIGE TEUS POEMAS
540 DATA APAGUE A LAMPADA
550 DATA E ME CERRAM OS OLHOS
560 DATA AQUELES QUE ESTAO A ESQUER
DA
570 DATA OS HOMENS DA CIENCIA
580 DATA OS HOMENS DA ACAO
590 DATA APRESSO-ME
600 DATA POR QUE DESPERDICARIAS
610 DATA QUANDO NOS ENCONTRAREMOS D
E NOVO?
620 DATA DORMINDO UM POUCO
630 DATA E COM MUITA PENA
640 DATA POR ESTES POUCOS PASSOS
650 DATA "AGORA, AO ANOITECER"
660 DATA VIVE PROVEITOSAMENTE
```

## EDITOR DE GRAFICOS

O EDITOR DE GRÁFICOS que agora apresenta-mos foi escrito por Maurizio Galluzzo, e publicado em uma revista européia de informática. Através de algumas informações solicitadas pelo micro, ele vai desenhando sua figura, e depois você tem as opções de ver o desenho, trocar as cores, fazer uma rotação na figura, uma translação, aumentá-la e reduzi-la em seu tamanho.

Este é o primeiro passo para quem deseja mais tarde se dedicar à edição de figuras no vídeo, fazendo coisas incríveis como as que aparecem no vídeo de nossos televisores.

```
10 REM -----
20 REM
30 REM EDITOR DE GRAFICOS
40 REM
50 REM by Maurizio Galluzzo
60 REM
70 REM -----
80 REM
90 REM
100 COLOR 15,1
110 KEYOFF
120 CLS
130 LOCATE 12,6
140 PRINT "EDITOR DE GRAFICOS"
150 LOCATE 12,7
160 PRINT "-----"
170 PLAY "ABCD"
180 LOCATE 5,20
190 PRINT "Pressione uma tecla para
continuar"
200 IF INKEY#="" THEN 200
210 CLS
220 C=15
230 CF=1
240 FL=0
250 GOSUB 590
260 REM -----
270 REM
280 REM MENU
290 REM
300 REM -----
310 REM
320 COLOR 15,1
330 CLS
340 LOCATE 16,3
350 PRINT "MENU"
360 LOCATE 16,4
370 PRINT "-----"
380 LOCATE 0,8
390 PRINT SPC(5);"1 _ COORDENADAS D
E VERTICES"
400 PRINT
410 PRINT SPC(5);"2 _ CAMBIO DE COR
ES"
420 PRINT
```

```
430 PRINT SPC(5);"3 _ DESENHO"
440 PRINT
450 PRINT SPC(5);"4 _ ROTACAO DA FI
GURA"
460 PRINT
470 PRINT SPC(5);"5 _ AUMENTO E RED
UCAO"
480 PRINT
490 PRINT SPC(5);"6 _ TRANSLACAO"
500 PRINT
510 PRINT
520 PRINT
530 PRINT " TECLE UM NUMERO ENT
RE 1 E 6"
540 G#=INKEY#
550 IF G#<"1" OR G#>"6" THEN 540
560 G=VAL(G#)
570 ON G GOSUB 590,820,990,1150,141
0,1590
580 GOTO 260
590 REM -----
600 REM
610 REM DADOS INPUT
620 REM
630 REM -----
640 REM
650 CLS
660 LOCATE 9,3
670 PRINT "LOCALIZACAO DE PONTOS"
680 PRINT:PRINT
690 IF FL=1 THEN ERASE X,Y
700 FL=1
710 INPUT "QUANTOS VERTICES";N
720 PRINT:PRINT
725 PRINT TAB(12);"Coordenadas:":PR
INT
730 DIM X(N)
740 DIM Y(N)
750 FOR I=1 TO N
760 PRINT "Vertice ";I;" ";
770 INPUT "X=" ;X(I)
780 INPUT " Y=" ;Y(I)
790 IF X(I)<0 OR X(I)>255 OR Y(I)<0
OR Y(I)>191 THEN BEEP:GOTO 760
800 NEXT
810 RETURN
820 REM -----
830 REM
840 REM CAMBIO DE CORES
850 REM
860 REM -----
870 REM
880 CLS
890 LOCATE 11,4
900 PRINT"CAMBIO DE CORES"
910 PRINT:PRINT
920 INPUT"TECLE A COR DE FUNDO (ENT
RE 1 E 15)";CF
930 IF CF<1 OR CF>15 THEN 920
940 PRINT:PRINT
950 INPUT"TECLE A COR DO GRAFICO (E
NTRE 1 E 15)";C
960 IF C<1 OR C>15 THEN 950
970 IF CF=C THEN 880
980 RETURN
990 REM -----
1000 REM
1010 REM DESENHO
1020 REM
1030 REM -----
1040 REM
1050 SCREEN 2
1060 COLOR C,CF,CF
1070 CLS
1080 FOR I=1 TO N-1
1090 LINE(X(I)*3/4,191-Y(I))-(X(I
+1)*3/4,191-Y(I+1))
1100 NEXT
1110 BEEP
1120 IF INKEY#="" THEN 1120
1130 SCREEN 0
1140 RETURN
1150 REM -----
1160 REM
1170 REM ROTACAO DA FIGURA
1180 REM
1190 REM -----
```



```

1200 REM
1210 CLS
1220 LOCATE 14,3
1230 PRINT "ROTACAO"
1240 LOCATE 8,10
1250 PRINT "De-me as coordenadas do
centro de rotacao"
1260 PRINT:PRINT
1270 INPUT "Qual o X ";XC
1280 IF XC<0 OR XC>255 THEN 1270
1290 INPUT "Qual o Y ";YC
1300 IF YC<0 OR YC>291 THEN 1290
1310 INPUT "Qual o angulo de rotaca
o(0-360) ";AN
1320 IF AN<0 OR AN>360 THEN 1310
1330 CLS
1340 AN=6.281853#*AN/360
1350 FOR I=1 TO N
1360 XT=XC+(X(I)-XC)*COS(AN)+(Y(I)-
YC)*SIN(AN)
1370 YT=YC-(X(I)-XC)*SIN(AN)+(Y(I)-
YC)*COS(AN)
1380 X(I)=XT:Y(I)=YT
1390 NEXT
1400 RETURN
1410 REM -----
1420 REM
1430 REM AUMENTO E REDUCAO
1440 REM
1450 REM -----
1460 REM
1470 KJ=0
1480 CLS
1490 LOCATE 7,3
1500 PRINT "AUMENTO E REDUCAO"
1510 PRINT:PRINT
1520 INPUT "Qual o coeficiente mult
iplicador ";K
1530 FOR I=1 TO N
1550 X(I)=X(I)*K
1560 Y(I)=Y(I)*K
1570 NEXT
1580 RETURN
1590 REM -----
1600 REM
1610 REM TRANSLACAO
1620 REM
1630 REM -----
1640 REM
1650 CLS
1660 LOCATE 12,3
1670 PRINT "TRANSLACAO"
1680 PRINT:PRINT
1690 INPUT "X da translacao";XT
1700 PRINT:PRINT
1710 INPUT "Y da translacao";YT
1720 FOR I=1 TO N
1730 X(I)=X(I)+XT
1740 Y(I)=Y(I)+YT
1750 NEXT
1760 RETURN

```

```

90 GOSUB730
100 FORC=1TO32:READA:S#=S#+CHR$(A):
NEXTC
110 SPRITE$(1)=S#
120 WIDTH29
130 CLS:RD=10:DI=1:X=112:N=0:DS=0
140 GOSUB560
150 REM
160 REM **MAIN LOOP**
170 GOSUB 300
180 D=STICK(L)
190 IF D=3 THEN N=8
200 IF D=7 THEN N=-8
210 IF VPEEK(6624+(X/8))<>32 THEN G
OSUB400
220 X=X+N
230 IF VPEEK(6624+(X/8))<>32 THEN G
OSUB400
240 IF VPEEK(6592+(X/8))<>32 THEN G
OSUB 400
250 PUTSPRITE0,(X,111),15,1
260 DS=DS+.02
270 GOTO170
280 REM
290 REM **MAKE ROAD**
300 IF DI=0 THEN GOTO330
310 IF DI=2 THEN GOTO340
320 DI=INT(RND(1)*3):GOTO350
330 DI=INT(RND(1)*2):GOTO350
340 DI=INT(RND(1)*2)+1
350 IF DI=0THENRD=RD-1:IFRD<=5THENR
D=5:DI=2
360 IF DI=2THENRD=RD+1:IFRD>=20 THE
NRD=20:DI=0
370 PRINT TAB(RD);CHR$(249);SPC(SP)
;CHR$(249)
380 RETURN
390 REM
400 REM **CRASH ROUTINE**
410 PUTSPRITE0,(X,111),14,1
420 SOUND0,0:SOUND1,5:SOUND2,0:SOUN
D3,13:SOUND4,255:SOUND5,15:SOUND6,3
0:SOUND7,0:SOUND8,16:SOUND9,16:SOUN
D10,16:SOUND11,0:SOUND12,5:SOUND13,
0
430 FOR C=1 TO 30:NEXTC
440 SOUND12,18:SOUND13,0
450 FORC=1TO100:NEXTC
460 PUTSPRITE0,(5,5),1,2
470 FORC=1TO300:NEXTC
480 IF DS>HSTHENHS=DS:GOSUB640
490 SCREEN1
500 LOCATE,5,0:PRINT"HIGH=";HS;"Km
BY ";H#
510 LOCATE4,10:PRINT" YOUR DISTANCE
=";DS;"Km"
520 LOCATE,18:PRINT"Difficulty:0, 1
or 2 (Easy)"
530 K#=INKEY$: IF K#<>"0"AND K#<>"1"
ANDK#<>"2" THEN GOTO530
540 SP=VAL(K#):SP=SP+4:RETURN130

```

```

550 REM
560 REM ** START OF ROAD**
570 LOCATE0,12
580 FORC=1TO13:PRINT TAB(RD);CHR$(2
49);SPC(SP);CHR$(1);CHR$(249):NEXTC
590 PUT SPRITE0,(X,111),15,1
600 LOCATE3,5:PRINT"SPACE BAR/FIRE
TO START"
610 IF STRIG(L)=-1 THEN LOCATE0,24:
RETURN
620 GOTO610
630 REM
640 REM *HIGH SCORE ROUTINE*
650 SCREEN1
660 LOCATE5,10,1:PRINT"YOU HAVE MAD
E A NEW HIGH SCORE":PRINT"ENTER YOU
R NAME(10 LETTERS)"
670 LOCATE2,14:PRINT" start here":L
OCATE,13
680 INPUT#
690 IF LEN(H#)>10 THEN H#=LEFT$(H#,
10)
700 IF H#="SONY" THEN H#="*****"
710 RETURN
720 REM
730 REM ** INSTRUCTIONS**
740 WIDTH26
750 CLS:PRINT:PRINT" DRIVER By
J.L.Hall":PRINT"
"
760 PRINT:PRINT" Suddenly your Sinc
lair C5 goes out of control, the br
akes fail and you are unable to ste
er straight."
770 PRINT"The object of this game i
s to try and stay on the road for a
s long as you can, by moving left o
r right using your cursor keys or j
oystick."
780 PRINT:PRINT"joystick/keyboard(j
/k)"
790 K#=INKEY#
800 IF K#="K"OR K#="k" THEN L=0:GOT
O820
810 IF K#="J" OR K#="j" THEN L=1 EL
SE 790
820 RETURN
830 REM **TITLE PAGE**
840 COLOR1,8,1:SCREEN3
850 OPEN"GRP:"FOR OUTPUT AS#1
860 PRESET(50,30):PRINT #1, ".MSX."
870 PRESET(15,100):PRINT #1, "*DRIVE
R*"
880 FOR C=1TO900:NEXTC
890 CLOSE#1:RETURN
900 REM ** SPRITE DATA **
910 DATA&H3C,&H7E,&HC3,&H81,&H99,&H
99,&HBD,&HBD,&HBD,&HFF,&HFF,&HFF,&H
7E,&H66,&H3C,&H18,&H00,&H00,&H00,&H
00,&H00,&H00,&H00,&H00,&H00,&H
00,&H00,&H00,&H00,&H00,&H00

```

## \* DRIVER \*

DRIVER é um programa que não tem muitos recursos visuais. Os gráficos são pobres, mas nós lhe afiançamos que você vai ficar irritado por não poder controlar por muito tempo o carrinho dentro da pista. E isto vai fazer com que você perca muitas horas tentando ir o mais longe possível, mas você teria que ser um verdadeiro Piquet para conseguir uma façanha dessas.

```

10 REM*****
20 REM * DRIVER *
30 REM * By J.L.Hall *
40 REM *On SONY MSX*
50 REM *****
60 REM
70 GOSUBB30
80 SCREEN1,2:COLOR7,1,15:KEYOFF:SP=
6:H#="SONY":HS=1

```

# Compu Club

Participe escrevendo artigos e dicas especiais para a nossa revista. Seu trabalho será remunerado.

Artigos ..... Cz \$400,00  
Dicas ..... Cz \$100,00

ATENÇÃO! So os trabalhos publicados serão remunerados.

## COPYS (CONT.)

```

5490 DATA 247,33,128,60,17,160,79,2
05,192,80,177
5500 DATA 195,162,80,205,201,1,33,0
,60,17,186
5510 DATA 232,78,205,192,80,33,64,6
0,17,9,202
5520 DATA 79,205,192,80,201,205,248
,1,195,25,151
5530 DATA 26,255
65000 READ NL,IN,FI:NL=NL+10
65010 CS=0
65020 FORI=1TO10:READ A:CS=CS+A:POK
E IN,A:PRINT@59,A;
65030 IN=IN+1:IF IN>FI THEN RETURN
65040 NEXT I:READ S0:CS=CS-256*INT(
CS/256):IF S0<>CS THEN PRINT"ERRO N
A LINHA ";NL:END
65050 NL=NL+10:GOTO65010

```

# A História do Compuclub

O COMPUCLUB nasceu de um sonho. Um sonho que, na ocasião, pareceu utópico a muitos dos que nos cercavam. Seria mesmo possível, em nosso país, a existência de uma estrutura integralmente composta de aficionados da microcomputação, dispostos a investir para enfim ter o melhor clube de soft de que jamais se ouviu falar? Um clube que não teria outros compromissos senão aqueles necessários à sua manutenção e ao seu desenvolvimento, e cuja Diretoria fosse constituída, não de pessoas que utilizariam o clube como meio de subsistência, mas de 'hobbyistas', dispostos a exercerem suas atividades à frente da organização pelo ideal da proposta original? No que dependesse de nós estávamos prontos a pagar para ver.

Nosso clube funcionou, em caráter ultra-experimental, durante cerca de quinze meses, com apenas 37 associados, muitos dos quais até hoje conosco. Estes foram os super-associados, que acompanharam de perto, e sentiram na própria pele, as dificuldades que cercaram nossos primeiros passos. Nessa época o clube enviava apenas uma fita cassete, com alguns programas copiados de revistas, acompanhada unicamente de uma carta.

Não desistimos. Antes, pelo contrário, nos sentimos encorajados, e o clube ganhou personalidade jurídica em outubro de 1984.

Nossas fitas eram, então, gravadas por nós mesmos, em processo absolutamente manual, e até hoje conservamos o TK-85 que, incansavelmente, gravou cerca de 200 horas contínuas de programas, sem nunca pedir para ir a uma oficina técnica.

Alguns meses mais tarde, surgiu o boletim-curso, em formato de apostila, e duplicado pelo processo de xerox. Foi uma real novidade, e causou grande sensação entre nossos associados, de vez que esta realização não se encontrava em nossa proposta de trabalho. O clube começava a retornar, na forma de benefícios, o investimento de cada associado.

Depois, foi a vez da fita personalizada e industrialmente gravada pela Polygram. Com isso, cresceu a qualidade, a apresentação ficou mais simpática, e nós ficamos um pouco mais livres para projetar outras realizações.

As consequências mais imediatas foram o novo boletim impresso em off-set, com uma cor extra na capa e contracapa, e conteúdo mais abrangente; e a inclusão de dois outros equipamentos em nossa linha de atendimento: ao ZX-81 e ao TRS-80 MOD III, se juntavam o ZX-SPECTRUM e o TRS-COLOR.

Nosso clube não ganhava apenas nível cada vez melhor, ganhava também projeção em escala nacional. Após algumas divulgações em revistas especializadas, os associados começavam a fluir, do Rio Grande do Sul ao Rio Grande do Norte, passando por todos os estados e territórios brasileiros.

Ainda restavam alguns impasses. A inflação havia onerado muito o projeto original. Em consequência, quase não dispúnhamos de recursos para novas aquisições, em termos de soft; não podíamos confeccionar fitas adiantadamente, ganhando no preço e evitando possíveis atrasos; e toda a parte burocrática caía diretamente sobre os ombros dos membros da Diretoria.

Lançamos, então, uma estratégia de emergência, veiculada em circular, a todos os associados. O clube não se afastaria de suas metas, mas seria ressarcido, por ocasião do envio das fitas, em bases mais realistas. Em contrapartida, o clube passaria a enviar os melhores softs do mercado, incluindo aqueles classe 'A' que, sozinhos, custam cerca de quatro vezes o valor de cada fita, amplamente documentados, com seus manuais de instrução, alguns com dezenas de páginas.

Tendo começado a colher os novos frutos, o COMPUCLUB contratou material humano para atuar na parte burocrática, dinamizando o atendimento ao associado, principalmente no que diz respeito ao envio de fitas: a nova fita segue assim que recebemos o crédito da anterior, num esquema mais personalizado.

Finalmente chegou a vez da linha MSX. Há muito tempo já vínhamos recebendo solicitações de usuários de micros compatíveis com o MSX para que iniciássemos atendimento nessa linha. Entretanto, preferimos agir com calma, e só nos lançamos a esta nova empreitada quando estivéssemos aparelhados e devidamente preparados para um atendimento a bom nível.

O tão esperado momento, enfim, chegou. E pudemos abrir as nossas portas aos usuários da linha MSX, que logo começaram a chegar de todos os pontos do país, ansiosos para integrar a FAMÍLIA COMPUCLUB.

Depois, veio o plano cruzado, que representou para todos nós, brasileiros, a esperança de dias realmente melhores. O início foi realmente bom, sobretudo para nós que, em época de alta inflação, não possuíamos nem capital de giro, nem possibilidade de planejamento. Pouco a pouco, porém, a crise começou a se avizinhar. O desabastecimento, as greves dos correios, foram duros golpes que nos atingiram em cheio, e só não nos derrubaram por causa da nossa determinação de não cair.

Hoje, desistimos da fita industrialmente gravada. Os prazos oferecidos pelas gravadoras brasileiras são muito pouco confiáveis, e tivemos atrasos tão significativos que, em muitas ocasiões nossos softs pareceram envelhecidos aos nossos associados. Agora, o clube grava suas próprias fitas, e assim mantém seus associados em dia com as principais novidades do mercado de softs. Nossa gravação, embora artesanal, é de muito boa qualidade, e nossa preocupação, no momento, é a aquisição de equipamento que nos permita gravar nossas próprias fitas através de processo semi-industrial.

A coqueluche do momento, entretanto, é a nossa revista. Sonhada há muito tempo, hoje quase não acreditamos que o sonho se tornou realidade. A cada novo número nosso orgulho vai aumentar, porque esta é uma vitória de todos nós, membros da Diretoria e associados.

Se você está recebendo esta revista como uma cortesia, não perca tempo. Associe-se imediatamente ao nosso clube. Como nosso associado, você receberá, periodicamente, a revista do clube. O número de lançamento já se encontra em suas mãos, e outros virão, a cada dois meses. Nós lhe enviaremos, também, fitas cassete com programas sensacionais. São jogos, aplicativos e utilitários para você compreender melhor o seu equipamento e utilizá-lo de modo mais eficiente. Nas linhas TRS-80 mod III e TRS-Color, já estamos operando na modalidade de disquete, e isto será estendido muito brevemente à linha MSX. Além disso, você vai poder, através de nossa revista, entrar em contacto com usuários de todo o país, bastando que nos escreva indicando o tipo de usuário que você deseja contactar. Sua disposição de colaborar com o nosso trabalho também será muito bem-vinda, e até recompensada. Caso você tenha feito algum programa muito bom, e que gostaria de ver publicado como sua colaboração, é só nos enviar. Se for muito bom mesmo, nós publicaremos a listagem em nosso boletim; se for excelente, fará parte de uma de nossas fitas. Artigos são também especialmente bem-vindos.

Como nosso associado, você receberá um número que lhe dará direito a concorrer a cinco prêmios que mensalmente sorteamos entre os membros do COMPUCLUB. Não tenha dúvidas, o COMPUCLUB retorna, na forma de benefícios, muitas vezes o valor do seu investimento.

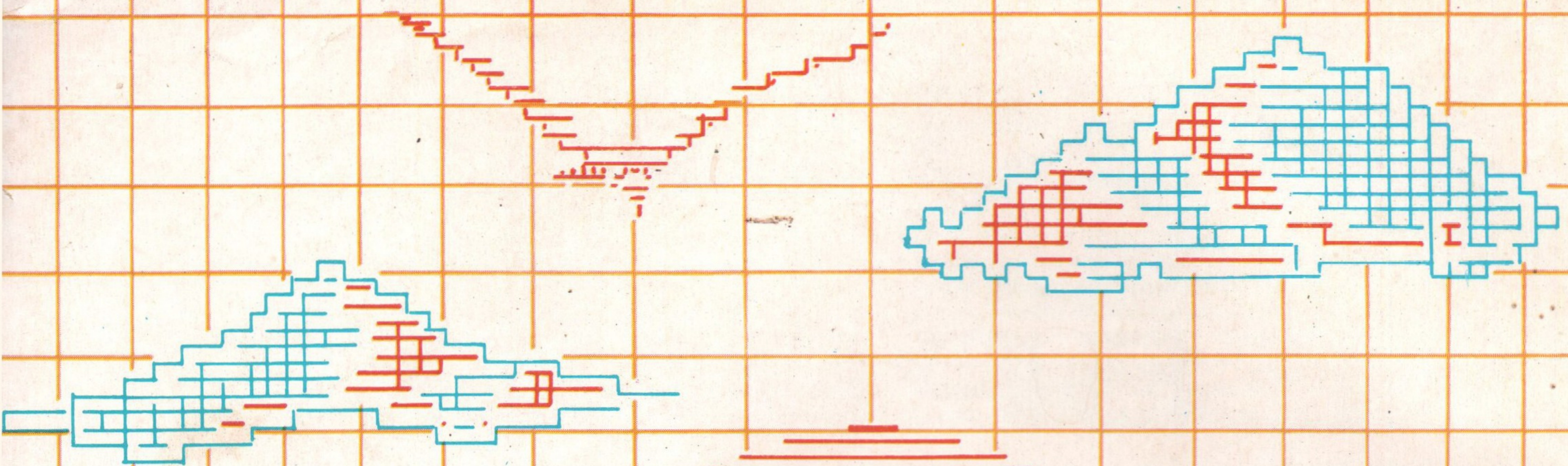
Nosso clube cresceu muito nos últimos tempos, e rapidamente se transformou em uma organização sem precedentes na história da microcomputação brasileira. E o mérito é nosso, de todos os seus associados, porque ele é fruto do nosso esforço.

Temos consciência, entretanto, de que há muito ainda por ser feito, e muito mais para nós mesmos usufruirmos. Por este motivo não paramos de trabalhar, porque sabemos que, juntos, seremos imbatíveis.

Outra grande novidade do momento é a nossa mais recente aquisição: um equipamento de composição de textos realmente extraordinário, que irá agilizar a parte editorial do clube. Esta edição, por exemplo, foi integralmente composta por nós em nosso novo equipamento.

Desde já, colocamos esta obra prima da moderna tecnologia à disposição de todos os nossos associados. Caso você tenha alguma matéria que necessite de uma composição de qualidade, é só nos consultar acerca de mais este serviço que, a partir de agora, implantamos em nosso clube.

**NÃO.** Não há qualquer exagero quando nos pronunciamos de modo tão enfático acerca do nosso clube. Na verdade, somos mais do que um clube. Somos mesmo uma grande família: **A FAMÍLIA COMPUCLUB.**

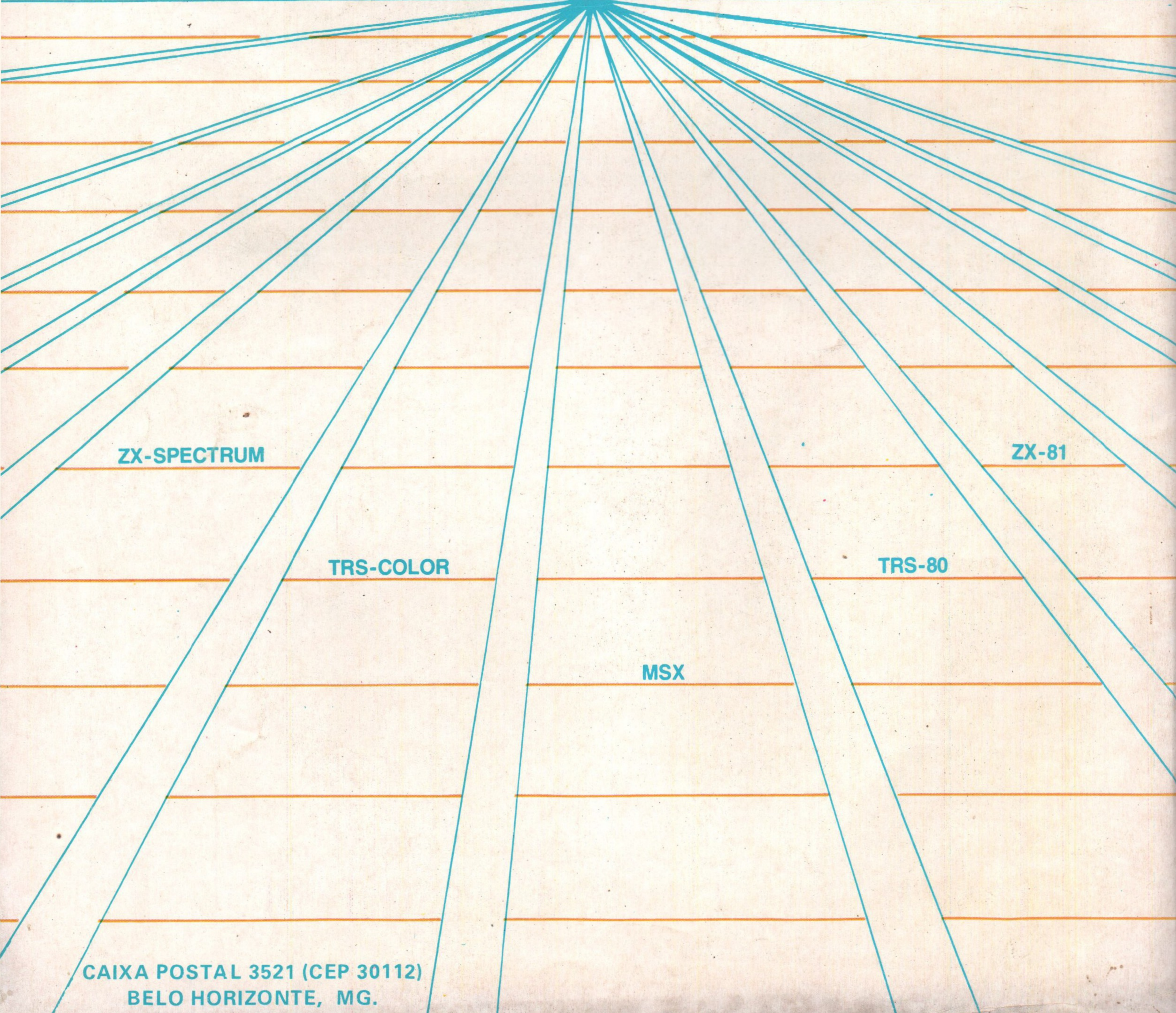


# COMPUCLUB

O MELHOR CLUBE DE SOFT  
DO BRASIL



LUA



ZX-SPECTRUM

ZX-81

TRS-COLOR

TRS-80

MSX

CAIXA POSTAL 3521 (CEP 30112)  
BELOHORIZONTE, MG.