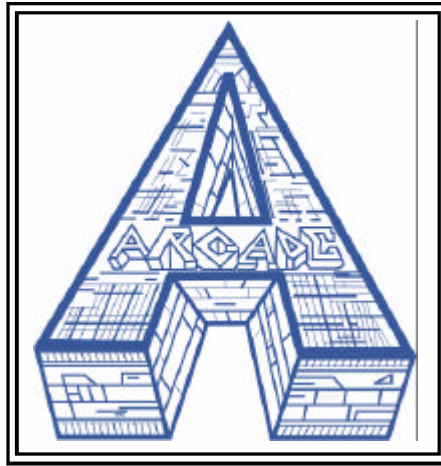


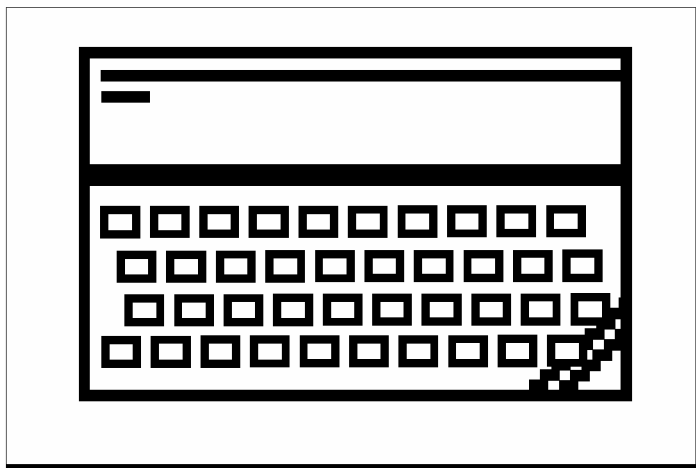
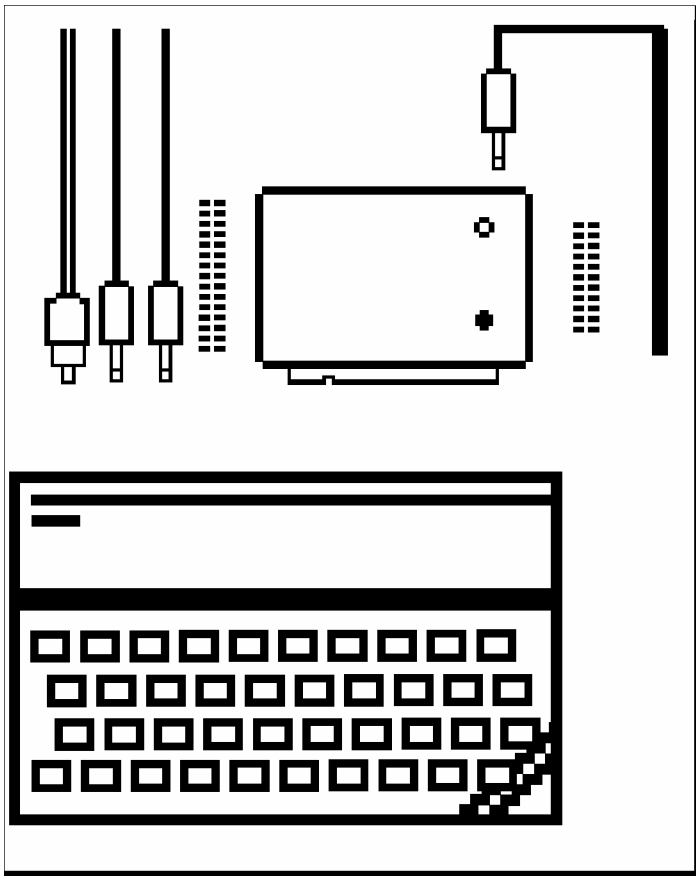
## Explorer

O seu micro pode ter agora 3 canais de som, poderão assim rodar seus programas com música simultaneamente, com a nova interface Explorer de som que a Arcade está lançando.



## Programas para disco

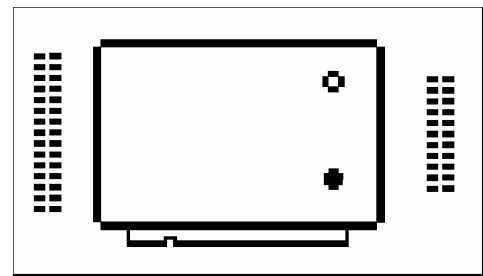
A Arcade é a única que tem programas de fase adaptados para disco padrão Beta. Todos os utilitários e aplicativos também estão adaptados neste padrão.



A Interface de Drive Padrão Beta da Arcade, é um excelente periférico para o extraordinário micro criado por Sir Clive Sinclair, o ZX Spectrum, lançado aqui no Brasil pela Microdigital, possibilitando ao usuário, além de jogos que são lançados mensalmente na Europa, a possibilidade de produzir serviços de alta qualidade comparados aos micros de 16 bits. Com saída para controlar até 4 drives, desde que estes estejam configurados corretamente, sem se importar com o tipo de drive, ou seja, face simples ou dupla, 360Kb ou 720Kb, 5 1/4 ou 3 1/2, facilitando assim a troca de padrões sem complicações. Além das vantagens do drive, também tem uma saída no lado direito para a ligação de qualquer impressora paralela padrão Centronics, gráfica ou não. As impressoras Amália, Emília, Elgin, NLQ, MTA, etc, funcionam sem qualquer complicação com a Interface Beta. A Interface contém também dois botões, o da parte superior, que é chamado "Botão Prático", transfere um programa bloqueado ou em High Speed da fita para o disco, com o único inconveniente de que sempre que for acionado, todos os 48Kb de memória RAM do micro são transferidos para as 12 primeiras trilhas do disco, o segundo botão é o "Botão de Reset" da Interface, que reseta todo o sistema, limpando toda a memória, e reinicializando o sistema de disco.

Ao carregar um programa na memória do micro, você tem então varias formas de colocá-lo no disco, a saber:

- 1- Pelo Botão Prático, como descrito neste manual.
- 2- Com o utilitário que acompanha a Interface.
- 3- Abrir o programa e transportá-lo bloco por bloco,



isto é, se o programa tiver Header (cabeçalho), o que dificulta o seu trabalho.

- 4- Utilizando a M1, que transfere o programa da fita para o disco sem se importar com a existência ou não dos Headers.

Solicite informações sobre drives de 360Kb, 720Kb, de 5 1/4 e 3 1/2 polegadas para o seu TK, MSX, PC e AMIGA.

# I. CARACTERÍSTICAS DA AR-20

---

Estamos finalmente lançando a Interface de Drive compatível com a BETA, mundialmente famosa, a AR-20.

A INTERFACE AR-20 possui características avançadas, tais como:

- *Utiliza drives do tipo compatível com IBM-PC, CP500, MSX, CP400 e outras marcas de qualidade.*
- *Aceita praticamente qualquer drive existente no mercado, desde o de 40 trilhas, 1 face (160 Kbytes) até o de 2 faces (640 Kbytes) de 5 ¼ ou 3 ½ polegadas.*
- *Rapidez de acesso: o tempo de carga de um bloco de 48 Kbytes é de 15 segundos, fazendo você esquecer dos carregamentos comuns.*
- *Suporta até 4 drives de TIPOS DEFERENTES, permitindo a conversão entre formatos.*
- *Interface para IMPRESSORA PARALELA (padrão CENTRONICS) incorporada, com software gravado em EPROM, permitindo a emissão de listagens, gráficos, etc...*
- *Este manual contém todas as informações que você precisará para utilizar ao máximo sua INTERFACE AR-20. Após uma pequena introdução para aqueles que nunca operaram com disk drive serão apresentados, um a um, todos os comandos do D.O.S. (Disk Operating System – Sistema Operacional de Disco), bem como será dada uma breve descrição dos tipos de arquivos úteis sobre a adaptação de programas a partir da fita cassete, no final deste manual.*

# II. INSTALAÇÃO

---

Acompanhando a AR-20, você encontrará:

- *Este manual.*
- *A interface propriamente dita.*
- *O disco com o sistema e utilitários.*

De posse destes itens, procedemos a seqüência de instalação da interface:

1. *Verifique se o micro está desligado, então, conecte a interface no TK 90X / TK95, encaixando o CONECTOR EDGE CARD no pente de contatos existentes na parte traseira do micro. Observe que só há uma posição de encaixe, graças a guia existente no conector.*
2. *Insira o cabo plano que vem do disk drive no conector lateral marcado "DISK – DRIVE" na Interface.*
3. *SEM LIGAR O MICRO, ligue o drive (ele deve ter sua própria fonte de alimentação, com respectiva chave liga-desliga). Se o LED do seu drive acender, então o conector está ao contrário. Retire o cabo do conector lateral, vire-o de lado e tente novamente, se o LED apagar está tudo ok.*
4. *Conecte a fonte do seu TK no plugue localizado atrás da interface e ligue a chave. Quando ligar a fonte, o LED do drive poderá piscar, isto é normal.*
5. *Após o teste de memória, o micro indica a versão firmware que está na interface. Isto significa que tudo está funcionando bem.*
6. *Insira o disco de DOS ou outro previamente formatado no drive. Após o RESET inicial ou através da chave, o firmware tentará EXECUTAR UM PROGRAMA chamado "boot" que esteja no disco, Se não houver nenhum PROGRAMA BASIC chamado "boot" ou se não houver um disquete no drive quando for ligado, aparecerá uma mensagem de erro, juntamente com a linha ( A>RUN "boot" ). Se desejar executar outro comando, simplesmente apague a linha e introduza o comando desejado.*
7. *O sinal " A > " serve para lembrá-lo de que você não está no BASIC TK, e sim os comandos deste manual estarão ativos, sendo todos os outros ignorados.*

## **IMPORTANTE:**

- *Nunca desconecte a Interface de trás do micro quando este estiver ligado, como descreve o item 1 anterior.*
- *Nunca ligue ou desligue o Micro / Interface ou os drives quando um disquete estiver dentro do drive.*
- *Sempre, antes de encaixar a interface no micro, verifique se a guia do conector EDGE está no seu lugar. A falta do guia permitirá ao conector ser encaixado em qualquer posição, o que poderá danificar permanentemente o Micro e/ou a Interface.*
- *Nunca segure o disquete pelo lado da ranhura, o que poderá causar erros na leitura.*

## **III. UMA NOTA SOBRE DRIVES E DISQUETES**

---

O disk-drive é o meio de armazenamento externo mais popular nos microcomputadores atuais, pois consegue aliar rapidez, baixo custo e boa capacidade de armazenamento a um volume e peso bastante pequenos.

Dentro de um disquete, os programas e blocos de dados (chamados Arquivos) são gravados magneticamente em um número fixo de TRILHAS concêntricas (como em um disco musical comum), número este dependente do tipo de drive utilizado, cada trilha é dividida em pedaços chamados SETORES, sendo que no caso da AR-20 D.O.S. cada setor pode armazenar 256 bytes de informação.

Todos os arquivos (de qualquer tipo) são organizados em uma lista, uma para cada disquete, chamado Catálogo ou Diretório. Nele estão armazenados, em ordem de gravação, o nome, tipo, tamanho, etc... de cada um dos arquivos existentes no disquete, sendo consultado pelo D.O.S. cada vez que for dado um comando de SAVE ou LOAD.

Por outro lado, quando se compra um disquete novo, nenhuma destas informações de trilhas e setores está gravado no disquete, ele necessitará ser Formatado. Formatar um disquete acidentalmente apagado (por um ímã ou campo magnético próximo), sua formatação é parcial ou totalmente perdida, fazendo com que o drive acuse erros de leitura e/ou escrita: neste caso, o único recurso é formatá-lo novamente. Note, porém, que o processo de formatação destrói todos os arquivos presentes no disquete. Para evitar isso, a primeira recomendação é:

- *Nunca deixe seus disquetes próximos a campos magnéticos intensos, tais como televisores, alto-falantes, motores, etc.*
- *Imprevistos sempre podem acontecer (uma queda na voltagem da rede elétrica, por exemplo), pode colocar por terra várias horas de trabalho. A segunda recomendação portanto é:*

***MANTENHA SEMPRE um BACK-UP em outro disquete dos arquivos importante que possuir.***

## **IMPORTANTE:**

Como todo equipamento, o seu drive é muito delicado, necessitando de cuidados especiais para que possa funcionar corretamente durante muito tempo.

- *Não submeta o drive a choques mecânicos fortes.*
- *Não segure do lado da ranhura.*
- *Não opere em locais muito úmidos ou quentes.*
- *Instale o drive em uma rede elétrica confiável, não sujeita a variações.*
- *Guarde o disquete na capa quando não tiver sendo usado.*
- *Envie o drive a uma assistência técnica para alinhamento.*

## IV. COMANDOS DO D.O.S.

---

Na própria AR-20 existe um programa que gerencia todas as operações realizadas pelo disk-drive: é o D.O.S. (do Inglês Disk Operating System – Sistema Operacional de Disco). Por estar contido em uma memória permanente, ele não ocupa muito espaço na memória do micro (apenas 112 bytes).

O D.O.S. possui seus próprios comandos, muitos dos quais são versões com sintaxe modificada dos comandos do BASIC TK. Ao ser ligado o sistema, o D.O.S. é imediatamente ativado, assim permanecendo até que recebe um comando RETURN. Para lembrá-lo de que o D.O.S. está ativo, é colocado na linha de disco a letra do drive atualmente selecionado (inicialmente o drive A) seguido do símbolo " > ". Neste caso apenas os comandos listados neste manual estarão ativos, sendo os outros ignorados.

Para desativar o D.O.S. e entrar diretamente no BASIC TK, introduza: RETURN <ENTER>

E para voltar ao D.O.S. a partir do BASIC TK, introduza: RAND USR 15360 <ENTER>

Todavia, este ponto de entrada do D.O.S. devolve o comando do D.O.S. a partir do BASIC sem interrupção na execução do programa, o formato é: RAND USR 15363: REM: comando

A palavra-chave REM presente na linha impõe duas restrições ao programador:

- *Só pode haver um comando do D.O.S. por linha do BASIC.*
- *E este comando deverá ser o último da linha em questão.*

Assim que é ligado o sistema ou na primeira vez que um determinado drive é acessado (após um NEW ou RESET), o D.O.S. inicia uma série de testes que determinam o tipo de drive instalado (número de trilhas e de faces) e qual o tempo mínimo de acesso trilha a trilha. Desta forma, a interface sempre pode utilizar ao máximo as características do drive, bem como permite que diferentes tipos de drive operem conjuntamente.

## V. A seguir uma descrição detalhada sobre cada um dos comandos.

---

### Comando 40

Existem certos tipos de drives mais antigos que possuem um mecanismo de movimentação da cabeça de leitura / gravação diferente de todos os outros (chamado de CAM). Ao serem efetuados os testes iniciais acima descritos, o D.O.S. pode assumir erroneamente que o drive em questão possui 80 trilhas, o que não é verdade. Para sanar este inconveniente, o comando 40 determina que o drive atualmente selecionado (de acordo com a letra que aparece na linha de edição) deve ser 40 trilhas.

**Exemplo:**     A > 40                   **seleciona 40 trilhas para o drive corrente: A**

### Comando 80

Este comando é contrário do anterior, ou seja, informa ao D.O.S. que o drive corrente deve ser assumido como de 80 trilhas.

**Exemplo:**     A > 80                   **seleciona 80 trilhas para o drive corrente: A**

## Comando \* (selecionar)

A Interface AR-20 é capaz de comandar ao mesmo tempo até 4 drives, nomeados A, B, C, e D. Ao ser ligada a interface, é assumido como drive corrente o drive A, e assim permanece até que seja modificado.

Existem dois meios para isto:

1. Indicar o nome do drive juntamente com o nome do arquivo, como por exemplo: `LOAD "b: teste"`. Neste caso, após ser realizada no drive b:, o D.O.S. torna a assumir o drive anteriormente selecionado como corrente.
2. Usar o comando \*. Neste caso, o drive especificado no comando passará a ser o drive corrente e a letra correspondente aparecerá na linha de edição juntamente com o símbolo.

### SINTAXE: \* Es

Onde, **Es** pode ser uma string simples (por exemplo: "C:"), uma variável string (simples, indexada ou fracionada, p.ex. A\$(TO 2), ou uma expressão string, p.ex. (CHR\$ 65+":"). O resultado da expressão deverá ser uma letra A, B, C, ou D, seguida de dois pontos.

Se não houver disco no drive correspondente ou se o drive correspondente não existir, será enviada a mensagem de erro "S / D i s c o".

**Exemplo:**     A > \* "b: "             **acessa o drive B**

## Comando **FORMAT**

Conforme foi explicado na seção III, o disquete precisa ser formatado previamente antes de ser usado. O comando **FORMAT** automaticamente verifica as especificações do drive e formata o disquete de acordo com elas.

### SINTAXE: **FORMAT Es**

Onde, **Es** pode ser qualquer uma das opções como no comando "\*" e deve conter o título do disquete. Se o nome do drive for especificado, a formatação será feita no referido drive. Se não for, será feito no drive corrente.

Muito cuidado com esquecimentos, pois o comando não pede para confirmar se o disquete a ser formatado está no drive. Se existir um disquete no drive selecionado, a formatação será imediatamente iniciada.

Após a operação, a tela mostrará o seguinte:     **NOME**  
   **N1 / N2**

Onde **N1** é o número de setores livres no disco e **N2** é o número total de setores. Cada trilha é gravada com 16 setores e cada setor contém 256 bytes. A trilha 0 do lado 0 do disco é reservada para uso do D.O.S. (ela contém o diretório) e desta forma é possível formatar um disquete de dupla face e 40 trilhas por face com 79 trilhas x 16 setores x 256 bytes = 323584 bytes, ou simplesmente 320 K.

É possível forçar um drive dupla face a formatar como face simples. Isto é fundamental para uso com **BOTÃO PRÁTICO** (ver seção VI). Para isto, deve-se colocar um \$ como primeiro caractere do título do disco, por exemplo: **FORMAT "\$teste"**.

**Exemplos:**     A > **FORMAT "teste - 01"**             **formata no drive corrente: A**  
                  A > **FORMAT "b: teste - 01"**       **formata no drive B estando no drive A**

## Comando CAT

Este comando mostra o diretório do disquete, fornecendo também informações adicionais sobre cada arquivo do mesmo.

**SINTAXE:** CAT #n Es

Onde, #n é opcional e pode ser um número, uma variável numérica (normal ou indexada) ou uma expressão numérica com valor entre 1 e 15. Representa o número do canal para onde o diretório será enviado.

Existem inicialmente 3 canais abertos para envio de informações no TK: CANAL #1 (Imprime na parte de baixo da tela), CANAL #2 (Imprime na parte de cima da tela – normalmente utilizado no comando PRINT) e CANAL #3 (envia os dados para a impressora – normalmente utilizado pelo comando LPRINT). Se for omitido, o diretório será enviado para o CANAL #2 (tela).

Es é opcional e pode ser qualquer das opções descritas para o comando “\*” sendo o nome do drive do qual se quer o diretório. Se for omitido, será enviado o diretório do drive corrente.

Após alguns segundos, serão enviadas as seguintes informações:

```
Título: NOME
N1 Arquivo (s)
N2 Apagado (s)

D: Arquivo 1 <T> S1: Arquivo 2 <T> S2
D: Arquivo 3 <T> S3: Arquivo 4 <T> S4
.
.
.
H3 Livre (s)
```

Onde “NOME” é o título do disquete, definido na formatação. “N1” é o número total de setores do disco. “N2” é o número de arquivos apagados (ver comando MOVE). “D” é o drive selecionado para o diretório. “T” é o tipo de arquivo (B para BASIC, C para código de máquina, D para matrizes e £

para arquivo de dados). “S1, S2, S3, S4” é o número de setores ocupados pelo arquivo. “H3” é o número de setores restantes do disquete.

**Exemplos:** A > CAT

A > CAT “b: ”

A > CAT #3

**lista conteúdo no drive corrente: A**

**lista conteúdo do disco no drive B.**

**Imprimi o diretório do drive corrente: A**

## Comando LOAD

Este comando funciona da mesma maneira que para o cassete, ou seja, carrega um arquivo do disquete para a memória. Todas as variações deste comando são possíveis, tais como:

**SINTAXE:** LOAD Es  
LOAD Es CODE  
LOAD Es CODE endereço  
LOAD Es DATA a\$(), etc ...

Onde, Es é uma expressão STRING ou STRING simples, exatamente como nos comandos anteriores. “endereço” também podem ser expressões ou variáveis.

Se o programa BASIC tiver sido salvo com número de linha, este começará a ser executado automaticamente. Se não, será listado assim que terminar o carregamento. Introduzir LOAD pura e simplesmente equivale a LOAD “boot”.

**Exemplos:** A > LOAD

A > LOAD “Asterix”

**arrega o programa “boot” do disco, se tiver!**

**carrega o programa chamado Asterix.**

## Comando **SAVE**

Este comando funciona basicamente como no cassete, (vide o save da AR-22 compatível com a M1) ou seja, salva um programa BASIC, bloco de bytes ou matrizes no disquete. Quase todas as variações deste comando são válidas, como:

**SINTAXE:**    **SAVE Es**  
                  **SAVE Es LINE e número da linha**  
                  **SAVE Es CODE início, comprimento**  
                  **SAVE Es DATA a\$( ), etc ..., onde:**

Es é uma STRING ou expressão como nos comando anteriores.

A única exceção é o “token” SCREEN\$, que deve ser substituído por CODE 16384, 6912. Note que a menor unidade de armazenamento em um disquete é um SETOR (256 bytes). Desta forma, ao se salvar um programa de, digamos, 10 bytes, este ocupará UM SETOR completo no disco. Isto é válido para qualquer fração de 256 bytes que o arquivo possuir (p.ex., um programa de 513 bytes ocupará 3 setores). Apesar disto, ao ser carregado o arquivo somente os bytes correspondentes ao que foi salvo serão carregados na memória.

## Comando **RUN**

Este comando é equivalente ao comando LOAD seguido de RUN. Desta forma, se o programa foi salvo com número de linha, a execução será a partir desta linha. Se não, a execução será a partir da primeira linha do programa.

**SINTAXE:**    **RUN Es**  
                  **RUN Es CODE**

A segunda opção permite a execução imediata de um programa em código de máquina a partir do endereço inicial do bloco. Se o endereço de execução for outro, então deve ser usado o comando RANDUSR endereço. Em ambos os casos, Es é definido como nos comandos anteriores.

## Comando **MERGE**

Este comando funciona exatamente como no cassete, acrescentando um programa BASIC a outro já existente. Se houverem duas linhas com o mesmo número, a mais nova substituirá a primeira.

**SINTAXE:**    **MERGE Es**

Onde Es é definida como nos comandos anteriores. Este comando impede a auto-execução de um programa salvo com o comando SAVE ... LINE ...

## Comando **NEW**

Este comando serve para mudar o nome de um arquivo. Se não for um programa BASIC, o tipo de arquivo deve ser especificado.

**SINTAXE:**    **NEW “novo”, “antigo”**  
                  **NEW “novo”, “antigo” CODE etc ...**

Onde “antigo” é o nome antigo e “novo”, o nome novo. Os tipos de arquivo são CODE, DATA e £, além do programa em BASIC.

**Exemplos:**    A > NEW “boot”, “buut”                    troca o nome “buut” pelo “boot”. Basic  
                  A > NEW “code”, “cod” CODE        troca o nome “cod” pelo “code”. Code

## Comando **ERASE**

Este comando apaga arquivos no disquete. Como no comando **NEW**, o tipo de arquivo deve ser especificado, se este não for um programa **BASIC**.

Note que o apagamento de arquivos deixa um espaço vazio no disquete. Este espaço vazio não será reaproveitado, pois o comando **SAVE** sempre salva no espaço após o último arquivo. Portanto, deve-se utilizar o comando **MOVE** para compactar os arquivos e acabar com tais “buracos”.

Quando houver um espaço vazio no disquete, o diretório acusará um ou mais espaços (arquivos apagados).

### **SINTAXE: ERASE Es TIPO**

Onde **Es** é definida como nos comandos anteriores e **TIPO** é o tipo de arquivo (**CODE**, **DATA** ou **£**), se este não for **BASIC**.

## Comando **COPY**

Este comando é usado para copiar arquivos de um drive para outro. Como nos dois comandos anteriores, o tipo de arquivo deve ser especificado se este não for um programa **BASIC**. O formato mais comum é:

### **SINTAXE: COPY “B:destino”, “A:origem” TIPO**

Que copia um arquivo do tipo **TIPO** chamado **origem** do drive A para o drive B, com o nome de **destino**. “**origem**” e “**destino**” podem ser iguais. Se A ou B for omitido, será assumido como o drive corrente.

Se quiser copiar **TODOS** os arquivos de um drive para outro, deve-se usar a forma reduzida:

**COPY “B:\*”, “A:\*”** ... ou seja, copiar para o drive B tudo que está no drive A.

Se houver no drive B um arquivo do mesmo nome e tipo que no drive A, o usuário será perguntado se deseja substituir o antigo pelo novo.

Se quiser copiar arquivos de um disquete para outro com apenas um drive ou quando desejar fazer uma cópia-imagem do disquete (para Backup), então deve ser utilizado o programa utilitário **FILER** que está no disco que acompanha a interface.

### **IMPORTANTE:**

*Este comando necessita de pelo menos 4 Kbytes livres na memória para ser executado. Se houver menos do que isso, a mensagem “RAM lotada” será impressa e o comando abortado.*

## Comando **MOVE**

Conforme foi descrito no comando **ERASE**, o apagamento de arquivos deixa um espaço vazio no disquete. Desta forma, é possível que não exista espaço livre no fim do disco para salvar um determinado arquivo, devido a estes espaços livres, chamados “buracos”.

O comando **MOVE** é utilizado para compactar os arquivos já existentes, mudando-os de posição no disquete, de modo a eliminar tais “buracos” e liberar espaço no fim do disco para outros arquivos.

### **SINTAXE: MOVE**



## Comando PEEK

Este comando permite ler qualquer parte de um arquivo no disquete e transferi-lo para uma utilização na RAM. Somente um setor pode ser lido de cada vez.

**SINTAXE:** PEEK Es endereço, setor

Es contém o nome do arquivo e é definido como nos comandos anteriores. Endereço é o local da RAM onde o setor será transferido e setor é o número do setor no arquivo.

## Comando POKE

Este comando escreve o conteúdo de uma área da RAM em qualquer parte de um arquivo contido no disco.

**SINTAXE:** POKE Es endereço, setor

### IMPORTANTE:

*Para cada arquivo aberto, é criado um "buffer" de 336 bytes na memória. Certifique-se de que existe memória disponível antes de abrir o arquivo. Após o arquivo ser fechado, este "buffer" é perdido.*

## ARQUIVOS SEQUENCIAIS E ALEATÓRIOS

---

Além dos tipos de arquivos já conhecidos pelo usuário do TK (BASIC, blocos de bytes e matrizes) a interface AR-20 suporta outro tipo de arquivo: o ARQUIVO DE DADOS. Neste tipo de arquivo, os itens são armazenados individualmente, e podem constar de números, variáveis numéricas e STRINGS.

Na recuperação dos itens na leitura, deve-se tomar cuidado quanto ao tipo e ordem dos itens armazenados, para não tentar ler um item STRING em uma variável numérica ou vice-versa.

Existem dois tipos de arquivos de dados: SEQUENCIAL e ALEATÓRIO. O seqüencial, como o nome já diz, obtém os itens na ordem em que foram originalmente armazenados, de modo que após ler o décimo item, o único meio de ler novamente o nono item é recomeçar a leitura desde o primeiro. O arquivo aleatório, porém, não tem o mesmo tipo de limitação do seqüencial.

Neste, o acesso aos itens é feito indicando-o pelo seu número dentro do arquivo. Portanto, a qualquer momento pode-se obter qualquer item do arquivo, tornando o acesso muito mais fácil e rápido. Porém como limitação deste tipo de arquivo tem-se que TODOS os itens devem ter o mesmo comprimento, especificado no momento em que é criado o arquivo. Se for armazenado um item menor que o comprimento do registro, o item será acrescido de um ENTER (CHR\$ 13) ao seu final. Se for maior, serão "cortados" todos os excedentes.

Um arquivo de dados necessita inicialmente ser ABERTO (para escrita ou leitura), processado, e ao final do processamento, ser FECHADO. O fechamento do arquivo é de suma importância, pois se ainda restarem dados no "buffer" a serem armazenados, estes serão perdidos se o arquivo não for fechado. Além disso, o D.O.S. não poderá encontrar o fim do arquivo nas operações de leitura.

Para abrir um arquivo é necessário “ligá-lo” a um dos CANAIS DE INFORMAÇÃO do TK. Conforme foi dito na explicação do comando CAT, existem 16 canais no TK destinados à troca de informações entre s diversos periféricos do computador. Destes dezesseis, quatro são inicialmente usados e correspondem a alguns periféricos padrão:

Canal #0: Entrada de dados via teclado, saída via parte inferior da tela.

Canal #1: O mesmo do anterior. Normalmente usado pelo EDITOR BASIC.

Canal #2: Não possui rotina para entrada de dados. Saída via parte superior da tela. Normalmente usada pelo comando PRI NT.

Canal #3: Não possui rotina para entrada de dados. Saída via impressora ZX (com a AR-20 conectada, saída via porta paralela CENTRONICS).

Os canais #4 até #5 estão livres para o usuário e devem ser usados com arquivos de dados. Até 12 arquivos simultâneos podem ser abertos desta maneira.

### **IMPORTANTE:**

*Para cada arquivo aberto é criado um “buffer” de 292 bytes na memória. Certifique-se de que existe memória disponível antes de abrir o arquivo. Após o arquivo ser fechado, este “buffer” é destruído.*

### **Comando OPEN #**

Este é o comando para abrir arquivos de dados. Os arquivos seqüenciais podem ser abertos para escrita ou leitura, mas nunca para ambas ao mesmo tempo. Para abrir um arquivo seqüencial para escrita, o formato é:

**SINTAXE: OPEN #N, “nome”,W**

Onde N é o nº do canal (um para cada arquivo) e “nome” é o nome do arquivo que pode ser também uma expressão ou variável, como em todos os comandos. Uma vez aberto um arquivo para escrita, usa-se o comando PRI NT para escrever no arquivo.

Após o fim da escrita, o arquivo deve ser fechado. Para abri-lo novamente para leitura, o formato é:

**OPEN #N, “nome”,R**

Quando o canal #N for aberto para leitura, deve-se usar os comandos I NPUT ou I NKEY\$ para ler os dados do arquivo. Após o término da leitura, deve-se fechar o arquivo.

Para abrir um arquivo aleatório (tanto para escrita como para leitura) o formato do comando é:

**OPEN #N, “nome”RND, tam**

Onde N é o nº do canal, “nome” é o nome do arquivo e tam é o tamanho do registro a ser usado. O valor máximo para tam é 255.

O registro é armazenado como uma STRING simples. Se esta for menor que o tamanho do registro, é armazenada normalmente com um “CARRIAGE RETURN” (CHR\$ 13) no fim. Se for maior, será truncada e armazenada sem o byte CHR\$ 13.

## Comando **CLOSE #**

Este é o comando para fechar arquivos.  
Para fechar um arquivo, basta fechar o canal no qual ele foi aberto.

**SINTAXE:**    **CLOSE #N**

## Comando **PRINT #**

Este comando escreve dados no arquivo e deve ser usado apenas no BASIC TK.  
Portanto, **NENHUM DOS COMANDOS PRINT, INPUT ou INKEY\$** devem ser precedidos da seqüência **RAND USR 15363: REM:** como em todos os outros comandos do D.O.S.

**SINTAXE:**

O formato para escrever em um arquivo seqüencial é:

**PRINT #N;.....**

As reticências indicam que quaisquer itens que possam ser enviados à tela podem ser enviados a um arquivo. Observe, porém, que na leitura o comando **INPUT #** espera um ENTER (CHR\$ 13) no final de cada item. Portanto, se quiser escrever mais de um item em um mesmo comando **PRINT #**, tenha o cuidado de separá-los com uma apostrofe.

O formato para escrever em um arquivo aleatório é:

**PRINT #N: (NR),.....**

Onde **NR** é o número do registro desejado. Para as reticências vale tudo o que foi dito para o arquivo seqüencial, porém, com uma ressalva: só se pode escrever em um registro de cada vez.

## Comando **INPUT # e função INKEY\$ #**

O comando **INPUT #** obtém do arquivo um item e o coloca na variável especificada, como se tivesse sido introduzido pelo teclado.

**SINTAXE:**

Para o arquivo seqüencial, o formato do comando é:

**INPUT #N: var**

Onde **N** é o nº do canal do arquivo e **var** é a variável onde será lido o item.

É preciso tomar cuidado para que itens numéricos sejam lidos em variáveis numéricas e itens string em variáveis string. Desta forma evita-se inconsistência na leitura dos dados.

Para ler os registros de arquivos aleatórios, o formato do comando é:

**INPUT #N: (NR), var**

**NR** é o nº do registro que se quer acessar e deve estar entre parênteses. **Var** e **N** são como no arquivo seqüencial.

A função `INKEY$ #` devolve um carácter lido do arquivo, tal e qual o `INKEY$` normal, que devolve um carácter do teclado. Seu formato é:

**INKEY\$ #N**

Onde **N** é o nº do canal ao qual está ligado o arquivo.

Em um arquivo seqüencial, quando não houver mais caracteres a serem lidos por `INPUT #` ou `INKEY$ #`, será emitida a mensagem de erro “`8 Fim de arquivo`” e a execução do programa se interromperá. Para evitar isso, deve-se simular a função EOF (end of File – ver exemplo) que devolve o valor 1 se o último carácter do arquivo já foi lido.

Com o arquivo aleatório isto não acontece, pois é desenvolvida uma `STRING` nula se for feita uma tentativa de leitura após o fim do arquivo (p.ex., tentar obter o 30º item de um arquivo de 20 itens)

## EXEMPLOS

### ARQUIVO SEQUENCIAL

Este programa escreve 10 itens no arquivo e os numera para futura referência. Após a escrita, o programa lê o arquivo e imprime os itens na tela:

```
10 RAND USR 15363: REM: OPEN #4, "ARCADE", W
20 FOR I=1 TO 10: PRINT #4; "Este e o item"; I
30 NEXT I: RAND USR 15363: CLOSE #4
40 RAND USR 15363: REM OPEN #4, "ARCADE", R
50 FOR I=1 TO 10: INPUT #4; A$
60 PRINT A$: NEXT I
70 RAND USR 15363: REM: CLOSE #4
```

### ARQUIVO ALEATÓRIO

Este programa escreve 10 itens numerados no arquivo e depois os recupera aleatoriamente 20 vezes, usando a função “`RND`” do BASIC TK

```
10 RAND USR 15363: REM: OPEN #4, "ARCADE 1" RND, 20
20 FOR I=1 TO 10: PRINT #4; (I), "Este e o item"; I
30 NEXT I: RAND USR 15363: REM: CLOSE #4
40 RAND USR 15363: REM OPEN #4, "ARCADE 1" RND, 20
50 FOR I=1 TO 20: INPUT #4; (INT (RND*9) +1), A$
60 PRINT A$: NEXT I
70 RAND USR 15363: REM: CLOSE #4
```

### SIMULAÇÃO DA FUNÇÃO EOF

As seguintes linhas em BASIC permitem ao programador evitar a leitura após o fim do arquivo, que interrompe o programa com o erro “`O Fim de arquivo`”:

```
1 DEF FN S(N)=PEEK 23631+256*PEEK 23632+PEEK (23574+2*N) +
256*PEEK (23575+2*N) - 1
2 DEF FN E(N)=PEEK (FN S(N) +13) =PEEK (FN S(N) +27) AND PEEK
(FN S(N) +14) =PEEK (FN S(N) +28)
```

Com isso, a função “`FN E(N)`” onde **N** é o nº do canal ao qual está ligado o arquivo, devolverá o valor 1 se o último carácter do arquivo tiver sido lido, e 0 se não.

## VI. O Botão Prático

---

Quando o botão vermelho localizado na lateral esquerda da interface é pressionado, toda a memória do micro (48Kbytes) é transferida para o disquete, de modo que o programa que está rodando no momento pode ser continuado mais tarde.

A seqüência a seguir mostra a operação correta do “Botão Prático”:

1. Formate um disquete de Face Simples (ver comando `FORMAT`),
2. Coloque o disquete no Drive A.
3. Retorne ao BASIC (comando `RETURN`),
4. Limpe toda a memória do TK, introduzindo `PRINT USR 0` [ENTER], ou botão de Reset.
5. Carregue e rode o programa do cassete normalmente.
6. Após o programa estar carregado e rodando, aperte o botão por um momento. A cópia do programa contido na RAM será salva no disco.

Note que o disquete precisa estar formatado e deve estar no drive A. o nome do arquivo salvo é “ ” e seu tipo é `CODE`. O arquivo é salvo nas 12 primeiras trilhas do disquete. Se haviam outros arquivos guardados nas trilhas 1 a 12 do disquete, estes serão apagados. Este arquivo pode ser renomeado pelo comando `NEW` ou pode ser copiado para outro disquete, usando o comando `COPY` ou o programa `FILER`.

NOTA: Alguns programas usam uma rotina para ler o teclado diferente do tipo normalmente usado. Se um programa não funcionar após ser carregado do disquete, deve-se mudar o nome do arquivo para que comece com um “\$”.

Para rodar o programa salvo pelo “Botão Prático”, use o comando `GOTO`.

### Comando **GOTO**

Este comando é exclusivamente usado para carregar um programa salvo pelo “Botão Prático”. Durante a carga, algumas linhas de vídeo poderão não fazer sentido. Este efeito é normal e desaparecerá assim que terminar a carga.

**SINTAXE:** `GOTO “nome” CODE`

## PROGRAMA ARCOPY

---

Este programa auxilia o usuário na transferência de programa do cassete para o drive. Após a carga, a tela mostra o nome e a versão do programa, bem como uma série de informações a serem preenchidas durante a leitura do programa.

Existem certas restrições ao programa a ser transferido: Seu tamanho não deve exceder a 40 Kbytes. O programa deve conter "header", ou seja, deve ser precedido de um pequeno grupo de 17 bytes que informam o nome, tipo, tamanho e endereço inicial do mesmo. A grande maioria dos jogos não corre "header", o que torna sua transferência um pouco mais difícil.

Após a carga do programa, este é imediatamente transferido para o drive, com o mesmo nome da versão cassete e o ciclo se reinicia para a carga do próximo bloco ou programa.

### IMPORTANTE:

*O programa ARCOPY apenas copia o programa conforme foi lido do cassete. Desta forma, é bem possível que seja necessária uma adaptação dos comandos LOAD para o formato do D.O.S., p. ex. LOAD "" CODE deve ser mudado para RAND USR 15363: REM: LOAD "nome" CODE. Cuidado com blocos de mesmo nome. Durante a transferência, o bloco anterior será substituído se houver igualdade entre os primeiros oito caracteres dos nomes dos dois blocos. Desta forma, um bloco chamado "programa 1" será apagado se o próximo bloco se chamar "programa 2".*

## PROGRAMA ARCFILES

---

Após a carga do programa, é apresentada uma lista dos comandos disponíveis. Serão descritos somente os comandos novos, pois os outros já foram descritos neste manual.

CAT Utilizado do mesmo modo que o comando normal do D.O.S., este envia ao periférico, informações adicionais sobre o drive utilizado sobre os arquivos que ele contém. Estas informações estão agrupadas em colunas, com os nomes dos arquivos linha após linha.

## APÊNDICE A – COMO TRANSFERIR PROGRAMAS DA FITA CASSETE

---

Existem três maneiras principais de se realizar a transferência, cada uma com sua seqüência particular de operações:

- *Manualmente, ou seja, carregando na memória cada um dos blocos componentes do programa e salvando-os (com endereço inicial e tamanho) no disquete. Este é talvez o mais eficaz, mas certamente o mais difícil de todos, pois depende do programa e do tipo de proteção empregado neste para bloquear cópias. Uma boa dose de conhecimento de linguagem de máquina e do sistema operacional do TK são necessários para se usar este método, que torna sua descrição impossível de ser feita neste manual.*
- *Usando a AR-22 - Multiface 1 que salva automaticamente.*
- *Usando o "Botão Prático", que é o mais simples e eficiente para transferir jogos, porém não funciona com programas de BASIC e com utilitários que utilizam o modo 1 de interrupção do microprocessador Z-80.*

- Usando o programa *ARCOPY*, que acompanha a interface no disco de utilitários. Este método tem como limitação só copiar programas que possuem *Header* (um pequeno grupo de 17 bytes que contém informações acerca dos programas e que é enviado ao gravador quando se dá o comando *SAVE* do *BASIC*), o que reduz os tipos de transferências a programas *BASIC*, utilitários e alguns jogos mais antigos.

## **APÊNDICE B – INFORMAÇÕES ADICIONAIS SOBRE A INTERFACE**

---

### **B.1) UTILIZAÇÃO COM MAIS DE UM DRIVE**

Como já foi dito, a interface AR-20 pode operar com até 4 drives ao mesmo tempo. Como a interface possui uma só saída para os drives, estes devem ser ligados em paralelo, ou seja, o cabo plano para mais de um drive deve conter um EUROCONNECTOR FÊMEA em uma das pontas (para ligação à interface) e de um a quatro conectores de Borda da Placa na outra ponta. A pinagem do conector da interface está listada na página seguinte. Este cabo pode ser adquirido pronto na ARCADE que possui outros suprimentos para informática.

A configuração de qualquer um dos drives será A:, B:, C: ou D: e deve ser feita no próprio drive, através dos Jumpers normalmente marcados como DS0 (A:), DS1 (B:), DS2 (C:) e DS3 (D:). A ligação do Jumper correspondente configurará o drive conforme desejado.

A ARCADE poderá fornecer todo o suporte técnico de instalação e configuração para o usuário que desejar ampliar o seu sistema.

### **B.2) TRATAMENTO DE ERROS**

A AR-20 trata os erros que porventura apareçam de duas formas diferentes: uma quando o D.O.S. está ativo e outra quando o BASIC está ativo.

Quando o D.O.S. está ativo (sinal de ">" na linha de edição), quaisquer erros são imediatamente reportados e o comando abortado. Por exemplo, uma mensagem de "Arq. Inexistente" é mostrada e o comando aparece novamente para ser corrigido.

Porém quando o BASIC está ativo, existem erros que levam à produção de uma mensagem (sem disco no drive, p. ex.) e outros que são simplesmente abortados, não produzindo mensagem de erro.

Para estes tipos, a função `USR 15363` devolve um número que pode ser armazenado para teste posterior de ocorrência de algum erro. Para isto, basta usar os comando do D.O.S. na forma `LET var=USR 15363: REM: comando`.

Após o retorno, a variável `var` conterá um número que deve ser interpretado.

*0 – Comando executado sem problemas.*

*1 – Arquivo inexistente no disquete.*

*2 – Arquivo já existente no disquete (quando tenta-se salvar um arquivo de mesmo nome e tipo de outro anteriormente salvo).*

*3 – Sem espaço no disquete.*

*5 – Número do registro acima do especificado (em operações com arquivos aleatórios).*

*7 – Erro no disco.*

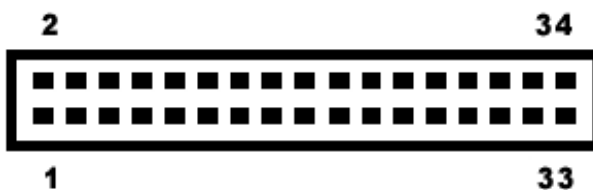
*8 – Erro de sintaxe nos comandos.*

Desta forma, pode-se utilizar estes números para tomar decisões e/ou imprimir mensagens no programa.

### B.3) PINAGEM DOS CONECTORES

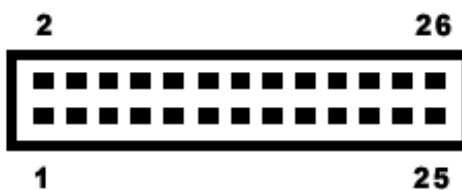
#### CONECTOR DE DRIVE (Visto de Frente)

Pinos 01 a 33	TERRA
Pino 02	N/C
Pino 04	N/C
Pino 06	Drive Select 3
Pino 08	INDEX
Pino 10	Drive Select 0
Pino 12	Drive Select 1
Pino 14	Drive Select 2
Pino 16	Motor On
Pino 18	Direction
Pino 20	STEP
Pino 22	Write Data
Pino 24	Write Gate
Pino 26	Track 00
Pino 28	Write protected
Pino 30	Read Data
Pino 32	Side Select
Pino 34	Ready



#### CONECTOR DA IMPRESSORA (Visto de Frente)

Pinos 01 a 25	TERRA
Pino 02	SELECT
Pino 04	PE (sem papel)
Pino 06	BUSY
Pino 08	ACK
Pino 10	Dado 7
Pino 12	Dado 6
Pino 14	Dado 5
Pino 16	Dado 4
Pino 18	Dado 3
Pino 20	Dado 2
Pino 22	Dado 1
Pino 24	Dado 0
Pino 26	STROBE





## B.4) USO DA INTERFACE DE IMPRESSORA

A interface de impressora paralela opera segundo o padrão Centronics e possui o software de operação já contido na interface. A única configuração necessária deve ser feita na impressora, que deve estar configurada de modo a não dar salto de linha (LF) quando receber o código de retorno do Carro (CR).

Normalmente, existe uma chave acessível ao usuário que faz esta configuração na impressora (CR=CR ou CR=CR+LF). Consulte o manual da sua impressora.

### USO COM O BASIC:

Quando é ligado o sistema, a saída do canal #3 (impressora) é enviada para a interface paralela. Todavia, um comando NEW ou PRINT USR 0 a partir do BASIC redirecionará a saída para a impressora ZX. Neste caso deve-se usar o comando RAND USR 15363: REM: LPRI NT 1 para acionar novamente a impressora paralela.

O comando LIST (bem como o comando LIST #3) funcionará com as seguintes características:

- *O driver de controle de atributos incluído na listagem (normalmente para destacar certos trechos do programa) não serão considerados.*
- *Os caracteres gráficos e UDGs serão impressos em modo gráfico com comando padrão EPSON, de modo a aparecer na tela.*
- *A função CHR\$ envia o caráter diretamente, a exceção dos CHR\$ 8 (BACKSPACE) e CHR\$ 6 (a vírgula do BASIC). CHR\$ 8 é transformado em um retorno de carro (CHR\$ 13) seguido de uma série de espaços que posicionam a impressora na coluna anterior. Esta característica permite a acentuação com programas de processamento de texto estrangeiros, mesmo em impressora que não aceitem o BACKSPACE (como a GRAFIX MTA).*
- *O CHR\$ 6 funciona como tabulação. Sendo este fixa de 8 em 8 colunas.*
- *Ao fim do comando, e enviando um CR seguido de LF, para a mudança de linha. O mesmo ocorre quando o caráter "" encontrado como separador de itens no comando.*

### USO EM LINGUAGEM DE MÁQUINA:

A instrução ASSEMBLY "CALL 3CF5h" ou RST 10h (este já com o canal #3 aberto anteriormente) imprime o caráter cujo código está no acumulador com as seguintes características:

- *O CARRIAGE RETURN (CR) é enviada com OR seguida da LINE-FEED.*
- *BACKSPACE é enviado conforme descrito para o BASIC.*
- *Caracteres gráficos e UDGs acionarão o modo gráfico padrão EPSON.*
- *TOKENS (ASCII 165 e 255) serão expandidos em seus comandos correspondentes no BASIC (p. ex., o código 234 é impresso como REM).*

A Impressora que se comportou perfeitamente quando ligada a AR-20 foi a LADY 80 da ELGIN. Com a configuração de fábrica produzindo gráficos de qualidade.

# GUIA PARA PEQUENOS DEFEITOS

---

## 1) SISTEMA NÃO LIGA.

- A chave ON/OFF da fonte do TK está na posição OFF.
- O plugue de 9 volts da fonte não está encaixando na tomada "DC" localizada atrás da interface.
- O plugue AC (2 pinos) da fonte não está ligado a uma tomada de força.
- Verifique se a tomada de força está boa, encaixando nela um aparelho que esteja comprovadamente funcionando.
- A interface AR-20 não está encaixada no TK

## 2) AO SER LIGADO O DRIVE, O LED VERMELHO ACENDE PERMANENTEMENTE.

- Retire o cabo do drive na interface e inverta a sua posição.

## 3) CARACTERES ESTRANHOS E BLOCOS COLORIDOS NA TELA, MENSAGEM INICIAL NÃO APARECE.

- Mau contato nos terminais do conector, desligue o sistema, retire cuidadosamente a interface e torne a encaixá-la.
- Se não funcionar, verifique a posição da interface em relação ao micro. AMBOS DEVEM ESTAR NA MESMA HORIZONTAL

## 4) O DRIVE NÃO é ACIONADO. MENSAGEM "sem disco" APARECE APÓS 5 SEGUNDOS.

- A fonte (externa) do drive não está ligada.
- O cabo de conexão do drive não está ligado à interface ou ao drive.
- O drive não está configurado corretamente.

## 5) O DRIVE É ACIONADO, MAS A MENSAGEM "sem disco" APARECE.

- Não existe disquete dentro do drive.
- A porta do drive não foi fechada corretamente.
- O disquete foi encaixado do lado errado no drive.

## 6) APÓS 20 SEGUNDOS, A MENSAGEM "Erro no Disco" APARECE.

- Um disquete apagado acidentalmente ou não previamente formatado foi inserido. Tente outro disquete ou formate este.

## 7) A MENSAGEM "Sem Arquivo" APARECE QUANDO SE TENTA CARREGAR UM PROGRAMA EXISTENTE NO DISCO.

- O tipo do arquivo não foi especificado ( CODE, DATA ou E )
- O disquete está com sua formatação defeituosa.

## 8) O COMANDO "LPRINT" EXECUTA NORMALMENTE, PORÉM NÃO SAI NADA NA IMPRESSORA PARALELA.

- A impressora está desligada.
- O cabo do conector está invertido. Retire-o da interface e inverta-o.
- O sistema está configurado para a impressora.

## 9) AO SEREM EXECUTADOS OS COMANDOS "LPRINT" OU "LLIST", O SISTEMA FICA "SUSPENSO", SÓ SAINDO COM A TECLA BREAK.

- A impressora está fora de linha em modo local. Pressione o botão LINHA/LOCAL na impressora.

# GARANTIA

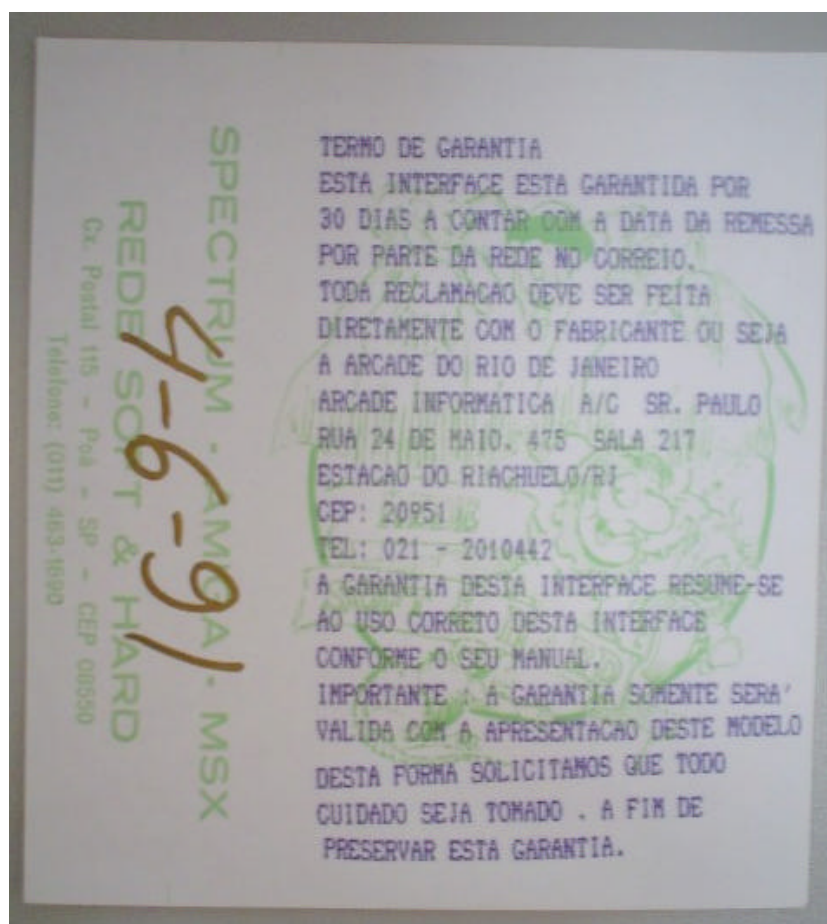
---

A ARCADE INFORMÁTICA & SISTEMAS LTDA, concede garantias, com o carimbo aposto na Nota Fiscal com validade de 90 dias a contar da emissão da Nota Fiscal.

Não damos garantias ao produto que foi violado por pessoas não autorizadas pela ARCADE.

Qualquer esclarecimentos, escreva para:

ARCADE INFORMÁTICA & SISTEMAS LTDA.  
RUA 24 DE MAIO, 475/217-218  
RIACHUELO-RJ  
CEP: 20711  
TEL (021) 201-0442



## Observações:

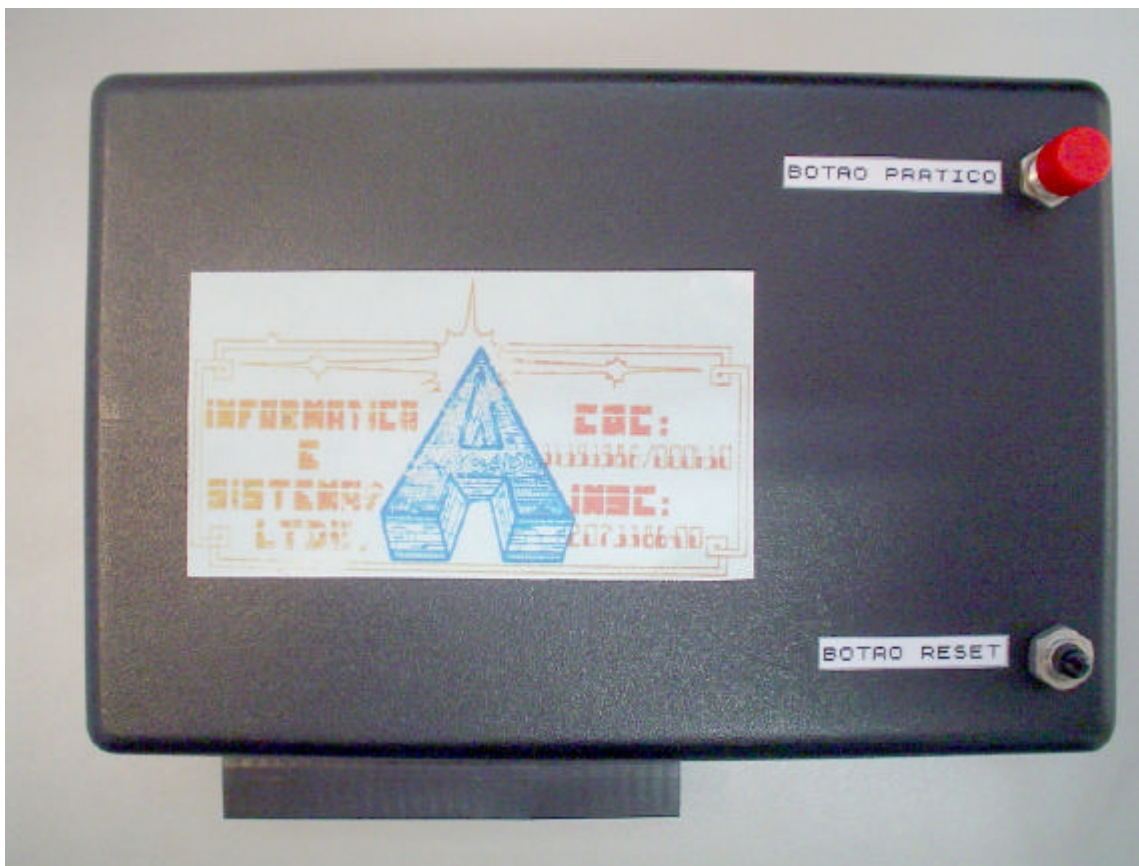
- No manual original da Arcade AR-20, todos os lugares em que aparecem o símbolo £ (libra) foram substituídos pelo símbolo # (sustenido). Mantido quando usados em descrições de tipo de arquivos (Code, Data ou £).
- Na página 16 foram acrescentados as pinagens dos conectores de drive e impressora, pois no manual original apesar de serem citados, não haviam!

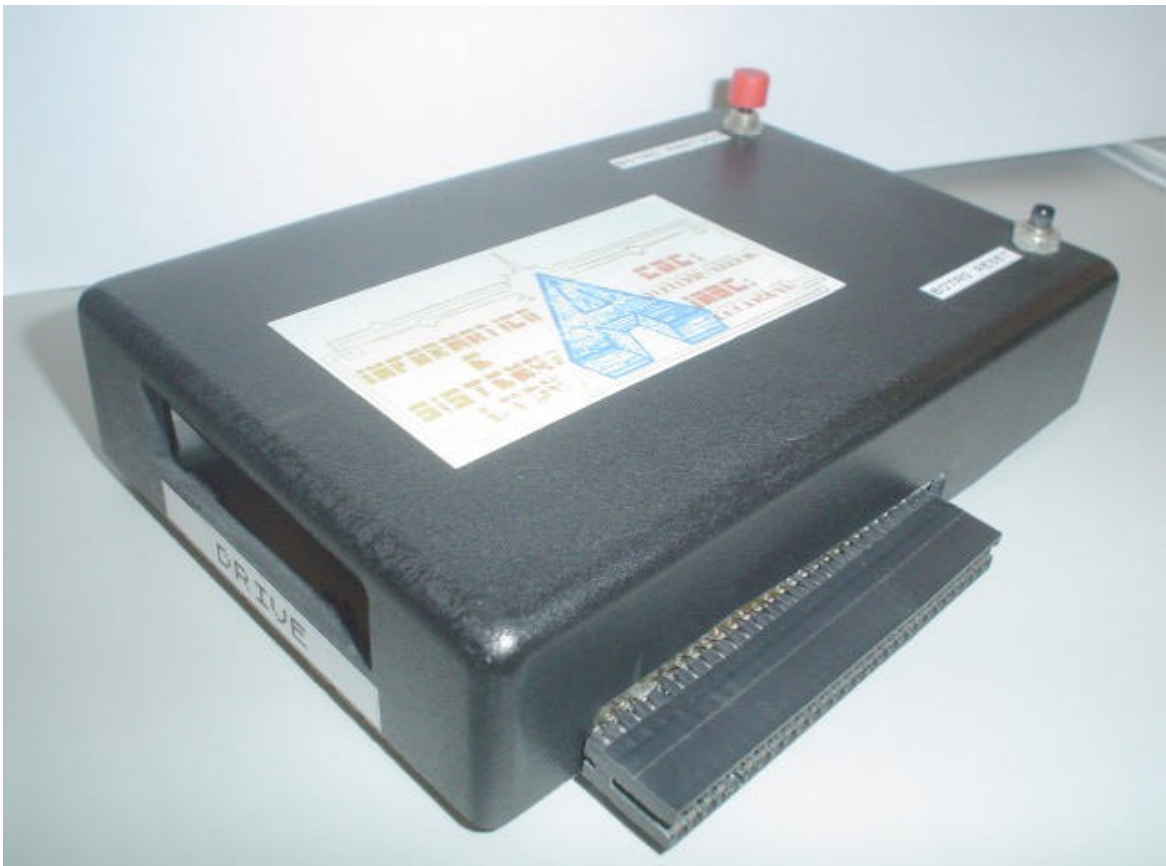
*Este manual foi digitado e corrigido por Marcelo Martins do Carmo.  
Março-Abril de 2004.  
Taubaté-SP.*

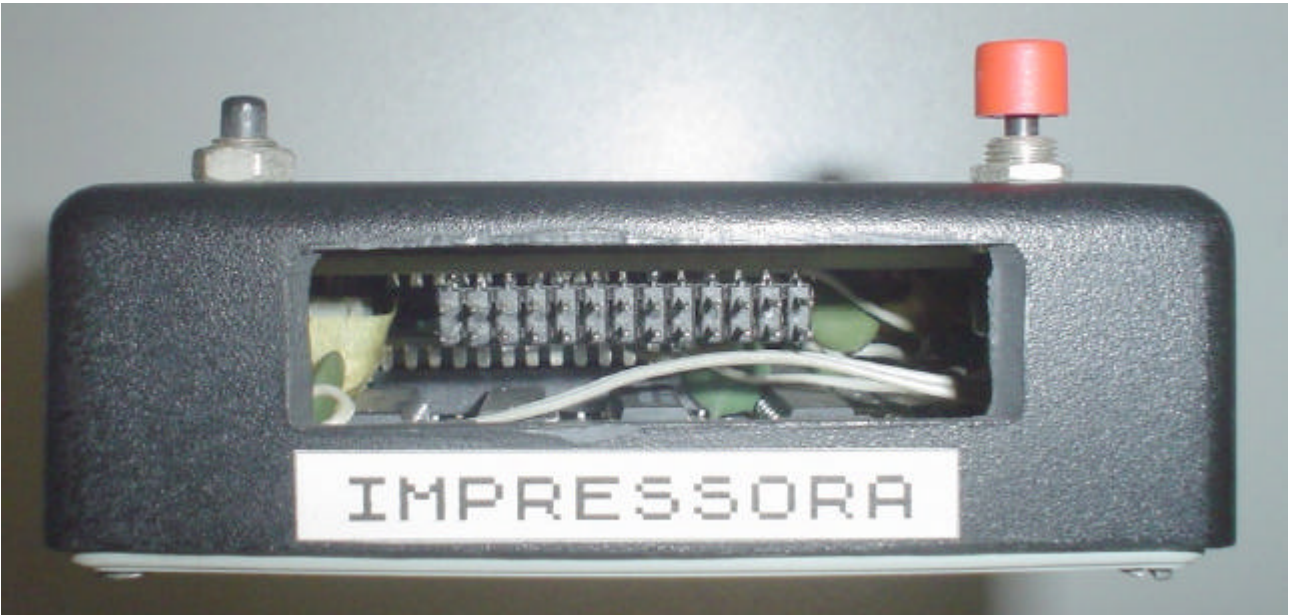
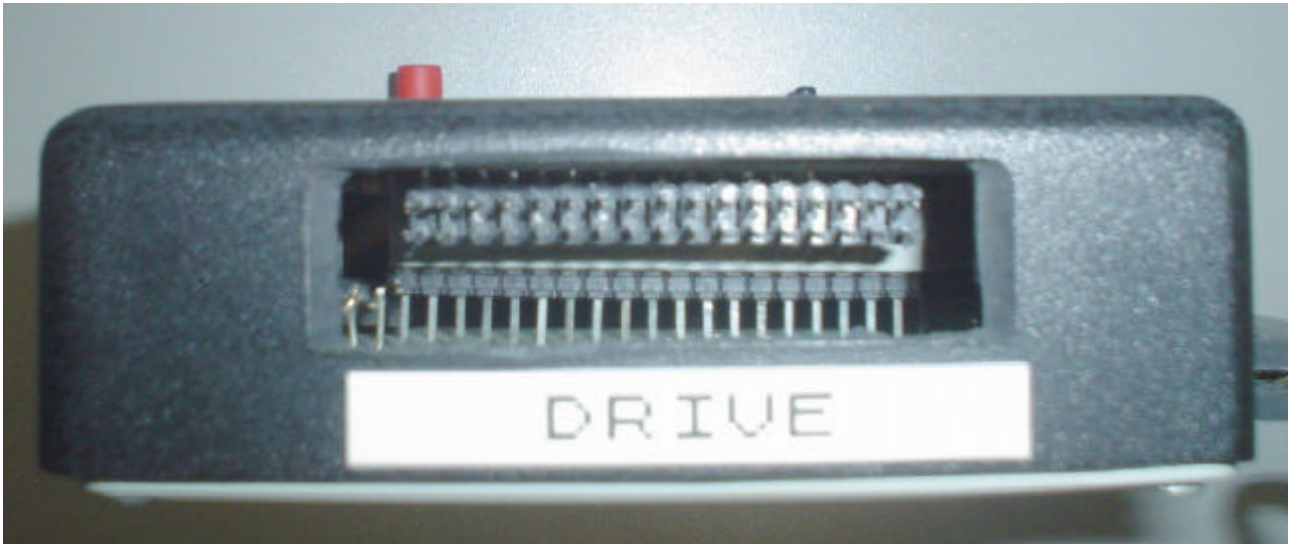
*E-mail: marcelo.m.carmo@uol.com.br  
Dedicado aos amigos da Lista do TK90X no Yahoo Grupos.*

## FOTOS DA INTERFACE AR-20 DA ARCADE

---





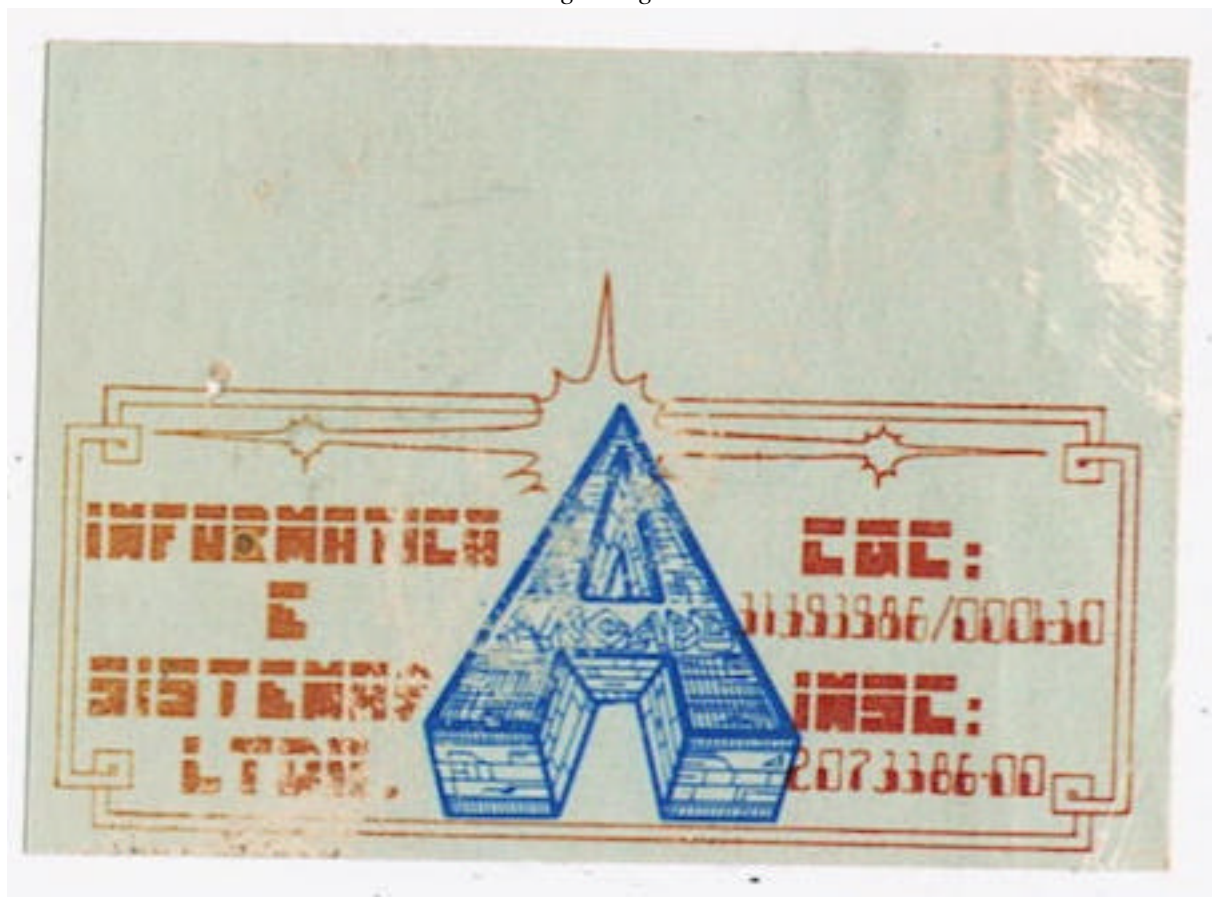


Outra opção de etiquetas!

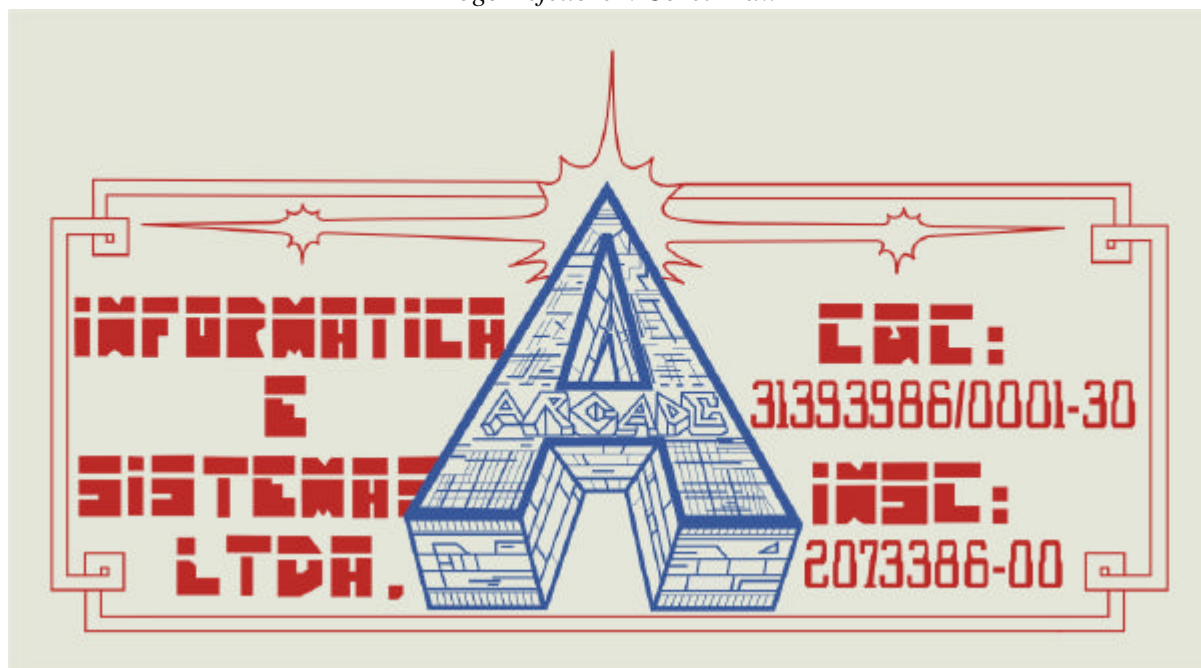


# LOGOMARCA DA ARCADE INFORMÁTICA & SISTEMAS LTDA.

*Logo Original*



*Logo Refeito em CorelDraw*



*Obs: Foram gastos 3 horas de trabalho neste logo "A" e mais umas 2 horas no restante.*