1. Publicação nº	2. Versão	3. Data	5. Distribuição
INPE-3114-PRE/511		*Maio, 1984*	☐ Interna ☒ Externa ☐ Restrita

4. Origem	Programa
DAS/DBL	*ASTRO*

6. Palavras chaves - selecionadas pelo(s) autor(es)
PROGRAMA
DIRETÓRIO
BOAS VINDAS

7. C.D.U.: *681.322;621.38*

8. Título *INPE-3114-PRE/511*

COMO IMPLEMENTAR O PROGRAMA DE BOAS VINDAS EM MICROCOMPUTADORES COMPATÍVEIS COM O APPLE II

10. Páginas: *09*

11. Última página: *06*

12. Revisada por

Jeronimo O. D. Jardim

9. Autoria *Ricardo Varela Corrêa*

Assinatura responsável

13. Autorizada por

Nelson de Jesus Parada
Diretor Geral

14. Resumo/Notas

Este programa altera o diretório de diskette identificando cada programa com uma letra. Para rodar qualquer programa, basta digitar a letra correspondente. Quatro comandos do DOS são identificados pelos números de 1 a 5, onde o número cinco identifica a saída para o BASIC.

15. Observações *Este trabalho foi parcialmente subvencionado pelo Fundo Nacional de Desenvolvimento Científico e Tecnológico, FNDCT, através de contrato FINEP. Este trabalho será submetido para publicação na revista MICRO SISTEMAS.*

Revista: MICROSISTEMAS

Análise, Teleprocessamento e Informática Ltda.

Al. Gabriel Monteiro da Silva, nº 1227
Jardim Paulistano - São Paulo - SP
01441

RESUMO DO CURRICULUM VITAE:

Engenheiro Eletrônico formado em 1982 pela Universidade de Brasília. Desenvolve projetos utilizando microcomputadores MAXXI-Polymax no tratamento de dados da Telemetria de Vôos de Balões Estratosféricos no Departamento de Astrofísica do Instituto de Pesquisas Espaciais, INPE do Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPq em São José dos Campos, SP.

COMO IMPLEMENTAR O PROGRAMA DE BOAS VINDAS EM MICROCOMPUTADORES COMPATÍVEIS COM O APPLE II

Ricardo Varela Corrêa

Este programa chama o Diretório do Diskette, identificando cada programa com uma letra. Reconhece o tipo de programa (se é arquivo ou "assembly"), seleciona e roda qualquer programa do Diretório pela letra associada inicialmente.

Algumas sub-rotinas e certas posições de memória do APPLESOFT são utilizadas; portanto, antes de analisar o programa, faz-se necessário explicá-las.

CALL - 868: limpa o texto da posição atual do cursor até o final da linha. A posição do cursor permanece inalterada.

CALL - 198: é o alto-falante que emite um "bip".

PEEK (-16384): é a posição de memória onde é armazenado o caractere digitado no teclado.

CALL - 936: equivale ao HOME do BASIC.

As posições de memória referentes à pagina 1 do modo texto são utilizadas para identificar qualquer caractere escrito no vídeo. Assim ao chamar o CATALOG de um diskette, pode-se ler através destas posições qualquer caractere escrito no vídeo.

Estas posições variam do decimal 1024 a 2047. A tela é dividida em três partes: as posições de 1024 a 1063 correspondem à primeira linha do vídeo na primeira parte; as posições de 1064 a 1103 correspondem à primeira linha do segundo terço: as posições de 1104 a 1143 correspondem ao último terço da tela. A segunda linha do primeiro terço estará entre 1144 e 1183 e assim sucessivamente. Cada terço do vídeo terá oito linhas de quarenta colunas. Esta divisão é esquematizada na figura 1.

---

* Trabalho parcialmente subvencionado pelo Fundo Nacional de Desenvolvimento Científico e Tecnológico, FNDCT, através de contrato FINEP.

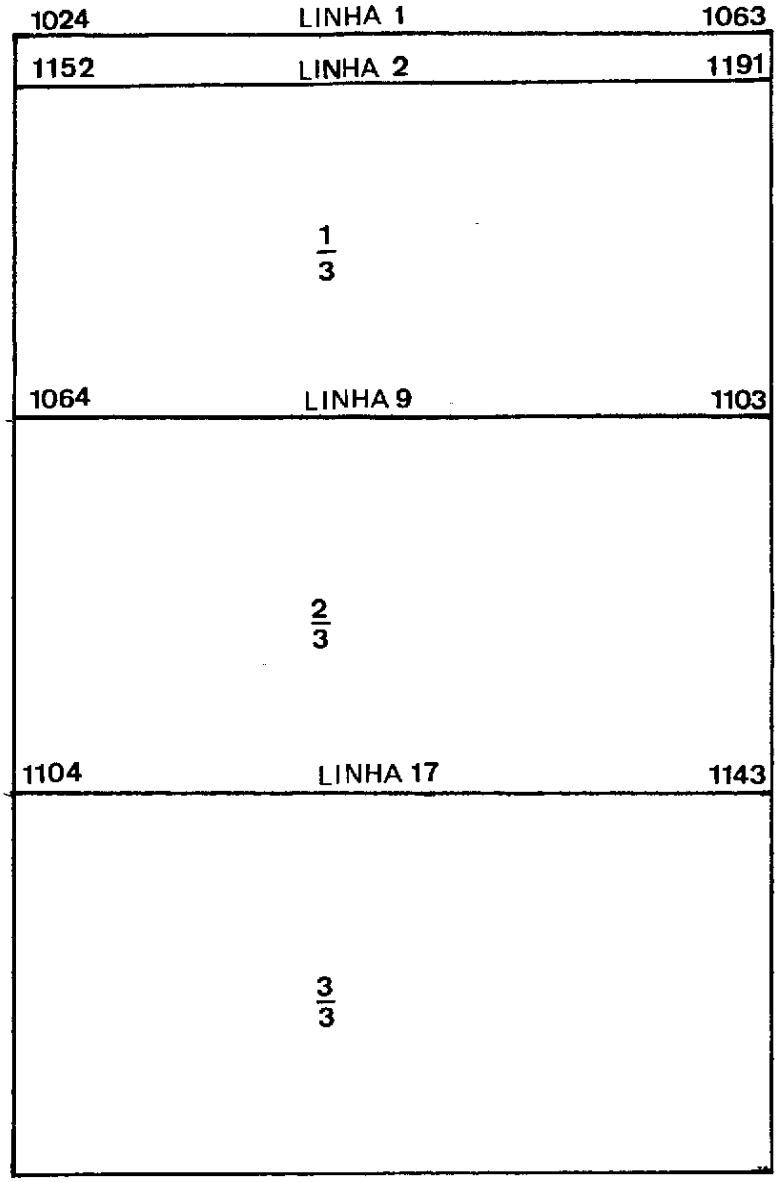

Figura 1 - Representação da divisão da tela do vídeo

Ao chamar o Diretório, verifica-se que na segunda coluna (1025) do vídeo é escrito o tipo de arquivo e nas colunas 1027 a 1029 é mostrada a quantidade de setores utilizados pelo programa.

Para identificar os programas existentes no Diretório, serão usadas letras entre colchetes escritas nas posições de 1027 a 1029. Em ASCII, o abre colchete e fecha colchete correspondem respectivamente a 219 e 221.

O programa de boas vindas será analisado passo-a-passo para um melhor entendimento. A primeira operação será chamar o CATALOG usando o artifício de imprimir um caractere de controle antes do comando CATALOG do DOS.

```
100 HOME: D$=CHR$ (4): PRINT D$"CATALOG"
```

A linha 1000 corresponde a uma sub-rotina de identificação do terço do vídeo e da linha em que se encontra escrito o programa.

```
1000 C1=INT(CV/8): C2=CV-C1*8:
     P=1024*C2+40*C1+CH:
     C=PEEK(P):
     RETURN
```

A variável CV conta as linhas do vídeo. Cada terço da tela é identificado pela variável C1 cujos valores serão 0,1 e 2. C2 identifica a linha de cada terço, assim terá valores de 0 a 7. Por exemplo, os valores C1=0 e C2=3 identificarão o primeiro terço do vídeo e a quarta linha.

Experimente variar CV de 0 a 23 calculando os valores de C1 e C2.

A posição do vídeo será identificada por P. A posição 1028 foi escolhida para colocar as letras de identificação. A posição 1025 identificará o tipo do programa que está sendo selecionado. Assim com CH=4 tem-se a primeira posição em 1028.

O parâmetro C lerá através do comando PEEK o que está escrito na posição P. Como teste, escreva o programa abaixo:

```
10 POKE 1027, 219
15 POKE 1028, 193
20 POKE 1029, 221
25 POKE 1155, 219
30 POKE 1156, 194
35 POKE 1157, 221
```

Este programa escreverá no vídeo as letras A e B entre colchetes, uma abaixo da outra. Experimente localizar todas as posições do vídeo para melhor entender o programa.

A linha 110 é responsável em escrever no vídeo as letras de identificação dos programas do Diretório.

```
110 T=0: CH=4: FOR CV=0 TO 23: GOSUB 1000
    : IF C<> 160 THEN POKE P-1, 219: POKE P, T+193:
    POKE P+1, 221: T=T+1: S=CV
```

O decimal 160 corresponde em ASCII ao BRANCO; assim a linha 110 compara com 160 todas as linhas da coluna 1028. Com isto só serão atribuídas letras aos programas existentes. O decimal 193 corresponde à letra A, e a cada incremento de CV, T é incrementado, escrevendo as letras em ordem crescente (A, B, C, D...). O parâmetro S guarda o último valor de CV em que existem caracteres escritos.

A linha 130 define o STRING A$. Através dos comandos MID$ e LEFT$ retira-se a primeira letra de A$, escrevendo no vídeo o restante. A cada retorno a esta linha (GOTO 130), a nova primeira letra é retirada, dando a impressão de movimentação do STRING.

```
100 NEXT VC: UTAB24: A$="DIGITE A
    LETRA P/ RODAR O PROGRAMA
    CORRESPONDENTE, OU LOAD=1 LOCK=2
    UNLOCK=3 DELETE=4 EXIT=5 (SAIDA/BASIC)..."
```

```
130 B$="RUN": HTAB1: PRINT LEFT$(A$, 39);:
    A$=MID$(A$,2)+LEFT$(A$,1): K=PEEK(-16384)
    IF K<128 THEN FOR K=1 TO 50: NEXT K:
    GOTO 130
```

A linha 140 verifica se um número ou letra foi digitado. Caso seja letra, o programa segue para a linha 300; caso seja número (de 1 a 5), o programa segue para a linha 200.

```
140 POKE -16368,0: K=K-176: IF K<1 OR K>5  THEN 300
200 HTAB 1: CALL-868: IF K=5 THEN CALL-936: NEW: END
```

Esta linha interrompe o programa caso o número digitado seja o decimal 5.

As linhas de 210 a 240 identificam o número digitado com o comando DOS a ser executado.

```
210 PRINT"DIGITE LETRA DO PROGRAMA...";: IF K=1 THEN B$="LOAD": FLASH
220 IF K=2 THEN B$="LOCK": FLASH
230 IF K=3 THEN B$="UNLOCK": FLASH
240 IF K=4 THEN B$="DELETE": FLASH
```

Como exemplo, caso seja digitado o número 2, será escrito no vídeo:

```
DIGITE LETRA DO PROGRAMA...LOCK
```

Nas linhas 250 e 300, através do comando ASC(K$), a variável K assumirá o valor ASCII correspondente da letra digitada. Digitando um número ou letra não existente, o programa retorna à linha 130.

Na próxima linha, CH e CV recebem novos valores que permitem que as posições da coluna 1025 possam ser lidas, pois nesta coluna estão escritos os tipos de programas existentes no Diretório. A letra A corresponde a arquivo e a letra B corresponde a programa em linguagem de máquina.

```
250 PRINT B$;: CALL-198: NORMAL:
    GET K$: K=ASC(K$)-48
300 IF K<17 OR K>T+16 THEN 130
310 CH=1: CV=S-T+K-16: GOSUB1000
    IF C=194 AND (B$="RUN" OR B$="LOAD)
    THEN B$="B"+B$
```

Dependendo do tipo de programa, B$ adota dois nomes: BRUN caso seja um programa em assembly; RUN no caso de arquivos.

A linha 320 lerá os valores ASCII existentes na linha selecionada, convertendo-os nos caracteres correspondentes. A variação de CH (de 6 a 39) fará com que os valores de P correspondam à coluna onde estão escritos os nomes dos programas do Diretório. Assim o STRING B$ será o nome do programa selecionado.

```
320 FOR CH=6 TO 39: GOSUB 1000:
    B$=B$+CHR$(C): NEXT CH: HTAB1:
    CALL-868: PRINT B$: PRINT D$; B$
    NEW

350 END
```

Agora só resta digitar o programa e gravá-lo em diskette através do comando INIT.

```
INIT ALO, D1
```