```
LINE# LOC   CODE        LINE

0001  0000             ;*****************************************
0002  0000             ;*  EEEEE X   X PPPP   AAA  N   N DDDD   *
0003  0000             ;*  E     X   X P   P A   A NN  N D   D  *
0004  0000             ;*  E      X X  P   P A   A N N N D   D  *
0005  0000             ;*  EEEE    X   PPPP  AAAAA N N N D   D  *
0006  0000             ;*  E      X X  P     A   A N  NN D   D  *
0007  0000             ;*  E     X   X P     A   A N  NN D   D  *
0008  0000             ;*  EEEEE X   X P     A   A N   N DDDD   *
0009  0000             ;*                                       *
0010  0000             ;*  AAA    BBBB   AAA   SSS  IIIII  CCC  *
0011  0000             ;* A   A   B   B A   A S   S   I   C   C *
0012  0000             ;* A   A   B   B A   A S       I   C     *
0013  0000             ;* A   A   B   B A   A  S      I   C     *
0014  0000             ;* AAAAA   BBBB  AAAAA   S     I   C     *
0015  0000             ;* A   A   B   B A   A     S   I   C     *
0016  0000             ;* A   A   B   B A   A S   S   I   C   C *
0017  0000             ;* A   A   BBBB  A   A  SSS  IIIII  CCC  *
0018  0000             ;*                                  *
0019  0000             ;*****************************************
0020  0000             ;
0021  0000             ;
0022  0000             ;*****************************************
0023  0000             ;*                                       *
0024  0000             ;* EXPANDED BASIC                        *
0025  0000             ;*                                       *
0026  0000             ;* PROGRAM TO APPEND TO THE CBM BASIC    *
0027  0000             ;* VERSION 4.0, 8032, CHRGET ROUTINES    *
0028  0000             ;* AND ADD COMMANDS TO CACHE PROGRAMS    *
0029  0000             ;* IN THE 64K ADD-ON MEMORY.             *
0030  0000             ;*                                       *
0031  0000             ;* 14JUL81 RJF (BOB)  30JUL81 (JOHN)     *
0032  0000             ;*                                       *
0033  0000             ;*****************************************
```

```
LINE# LOC   CODE        LINE

0035  0000               ; SYSTEM VARIABLE DEFINITIONS
0036  0000               ;
0037  0000               STATUS =$96             ;SYSTEM STATUS VARIABLE
0038  0000               DS     =$13             ;DISK STATUS DESCRIPTOR
0039  0000               CHRGET =$70             ;BASIC ROUTINE TO GET CHARS
0040  0000               CHRGOT =$76             ;GET LAST CHARACTER
0041  0000               GETPTR =$77             ;POINTER TO CHARACTERS
0042  0000               CHDGOT =$D39F           ;ROM COPY OF CHRGOT
0043  0000               NEWSTT =$B74A           ;NEW STATEMENT EXEX
0044  0000               LNKPRG =$B4B6           ;LINK BASIC LINES
0045  0000               STXTPT =$B622           ;SET START TEXT POINTER
0046  0000               OP94   =$F4A5           ;OPEN A FILE ON IEEE
0047  0000               READY  =$B3FF           ;RE-ENTER BASIC
0048  0000               FLOAD  =$B60B           ;RUN AN OVERLAY
0049  0000               RUNC   =$B5E9           ;CLEAR VARIABLES
0050  0000               CLRCH  =$FFCC           ;CLEAR OPEN CHANNEL
0051  0000               FCLOSE =$F2E2           ;CLOSE A FILE
0052  0000               TALK   =$F0D2           ;TELL DEVICE TO TALK
0053  0000               ACPTR  =$F1C0           ;GET A BYTE
0054  0000               CLSEI  =$F72F           ;CLOSE A FILE
0055  0000               UNTLK  =$F1AE           ; UNTALK
0056  0000               TKSA   =$F193           ;SEND SECONDARY ADDR
0057  0000               WSW    =$B3
0058  0000               T1     =$B4
0059  0000               T2     =$B5
0060  0000               T3     =$B6
0061  0000               FNLEN  =$D1             ;FILE NAME LENGTH
0062  0000               LA     =$D2             ;LOGICAL FILE NUMBER
0063  0000               SA     =$D3             ;SECONDARY ADDRESS
0064  0000               FA     =$D4             ;PRIMARY ADDRESS
0065  0000               FNADDR =$DA             ;FILE NAME ADDRESS
0066  0000               INDEX1 =$1F             ;TEMP POINTER
0067  0000               INDEX2 =$21             ;TEMP POINTER
0068  0000               EAL    =$C9             ;END ADDRESS POINTER
0069  0000               VARTAB =$2A             ;VARIABLE TABLE POINTER
0070  0000               MEMSIZ =$34             ;TOP OF MEM POINTER
0071  0000               ;
0072  0000               ; PROGRAM VARIABLES AND CONSTANTS
0073  0000               ;
0074  0000               LOC    =$7800
0075  0000                      *=LOC
0076  7800  4C 2E 79            JMP GOOD         ;START PROGRAM
0077  7803               ;
0078  7803               ;
0079  7803               TABHI
0080  7803  79                 .BYT >ZZ1
0081  7804  79                 .BYT >ZZ3
0082  7805  7A                 .BYT >ZZ4
0083  7806  7A                 .BYT >ZZ5
0084  7807  79                 .BYT >ZZ8
0085  7808               ;
0086  7808               TABLO
0087  7808  91                 .BYT <ZZ1
0088  7809  FD                 .BYT <ZZ3
0089  780A  11                 .BYT <ZZ4
```

```
LINE# LOC   CODE        LINE

0090  780B  22                .BYT <ZZ5
0091  780C  7C                .BYT <ZZ8
0092  780D                    ;
0093  780D                    NCMD    =5
0094  780D  52 4C             CMD     .BYT 'RLOEG'
0095  7812                    FILE    *=*+22
0096  7828  00                .BYT $00
0097  7829                    ENTRY   =25
0098  7829                    ENTMAX  =225            ;ENTRY*9
0099  7829                    FITAB   *=*+256
0100  7929  81 B7             WHERET  .WOR $B782-1    ;GONE3-3-1
0101  792B                    MASTER  =$BB            ;AVAILABLE SPACE (THREE BYTES
0102  792B  4C 51 79          JUMP    JMP START
```

```
LINE# LOC    CODE          LINE

0104  792E                ;
0105  792E                ;  INIT THE MEMORY MANAGER (SYS TO HERE)
0106  792E                ;
0107  792E                GOOD
0108  792E  A2 02                LDX #$02        ;MOVE THREE BYTES
0109  7930  BD 2B 79     WEDGE   LDA JUMP,X
0110  7933  95 79                STA CHRGOT+3,X
0111  7935  CA                   DEX
0112  7936  10 F8                BPL WEDGE
0113  7938                ;
0114  7938                TABSET
0115  7938  A2 00                LDX #0
0116  793A  8A                   TXA
0117  793B  9D 29 78     TABLP   STA FITAB,X
0118  793E  E8                   INX
0119  793F  D0 FA                BNE TABLP
0120  7941                ;
0121  7941                ;  SETUP THE START ADDRESS
0122  7941                ;
0123  7941  A9 80                LDA #$80
0124  7943  85 BC                STA MASTER+1    ;HIGH BYTE
0125  7945  0A                   ASL A
0126  7946  85 BB                STA MASTER      ;LOW BYTE
0127  7948  85 BD                STA MASTER+2    ;BANK SELECT
0128  794A  85 34                STA MEMSIZ      ;SET MEMSIZ TO LOC
0129  794C  A9 78                LDA #>LOC
0130  794E  85 35                STA MEMSIZ+1
0131  7950  60                   RTS
```

```
LINE# LOC    CODE          LINE


0133  7951                ;
0134  7951                ; THIS IS WHERE WE COME TO DO THE WORK
0135  7951                ;
0136  7951                START
0137  7951  C9 21               CMP #'!          ;LOOK FOR THE ESCAPE CHARACTER
0138  7953  D0 21               BNE NOTME
0139  7955  20 C9 7A            JSR SAVREG       ;SAVE THE REGISTERS
0140  7958                ;
0141  7958                ; TEST HERE FOR QUOTE MODE ETC.........
0142  7958                ;
0143  7958  20 70 00            JSR CHRGET       ;GET THE NEXT CHARACTER
0144  795B  48                 PHA
0145  795C  F0 14               BEQ NOTCMD
0146  795E  A2 04               LDX #NCMD-1
0147  7960                FINDC                  ;FIND THE COMMAND
0148  7960  DD 0D 78            CMP CMD,X
0149  7963  D0 0A               BNE AGAIN
0150  7965                ;
0151  7965                ; PUT COMMAND ADDRESS ON STACK
0152  7965                ;
0153  7965  68                 PLA              ;FIX THE STACK
0154  7966                ;
0155  7966  BD 03 78            LDA TABHI,X      ;FOUND COMMAND...GO THERE
0156  7969  48                 PHA
0157  796A  BD 08 78            LDA TABLO,X
0158  796D  48                 PHA
0159  796E  60           STRTS RTS
0160  796F                ;
0161  796F                AGAIN
0162  796F  CA                  DEX
0163  7970  10 EE               BPL FINDC
0164  7972                ;
0165  7972                NOTCMD
0166  7972  20 D0 7A            JSR RESREG
0167  7975  68                 PLA              ;FIX THE STACK
0168  7976                NOTME
0169  7976  C9 3A               CMP #':
0170  7978  B0 F4               BCS STRTS
0171  797A  4C 7D 00            JMP CHRGOT+7
```

```
LINE# LOC    CODE          LINE

0173  797D                ;
0174  797D                ; QUIT COMMAND
0175  797D                ;
0176  797D                ; THIS COMMAND EXITS THE MEMORY MANAGER
0177  797D                ; AND RESTORES THE CHRGET ROUTINE.
0178  797D                ;
0179  797D                QUIT
0180  797D  20 70 00             JSR CHRGET      ;CLEAN UP THE LINE
0181  7980  D0 FB                BNE QUIT
0182  7982  A2 02                LDX #$02
0183  7984                REST
0184  7984  BD A2 D3             LDA CHDGOT+3,X
0185  7987  95 79                STA CHRGOT+3,X
0186  7989  CA                   DEX
0187  798A  10 F8                BPL REST
0188  798C  20 D0 7A             JSR RESREQ
0189  798F  4C 76 00             JMP CHRGOT
```

```
LINE# LOC   CODE          LINE


0191  7992                ;
0192  7992                ;LOAD FILE FROM DISK INTO EXPANSION MEMORY
0193  7992                ;
0194  7992                RECAL
0195  7992  20 D7 7A             JSR GETFN       ;GET THE FILENAME
0196  7995  A5 96               LDA STATUS
0197  7997  D0 53               BNE RECOUT      ;BAD STATUS
0198  7999  20 10 7B           JSR GTYPE       ;GET THE FILE TYPE
0199  799C  A5 96               LDA STATUS
0200  799E  D0 4C               BNE RECOUT
0201  79A0  A9 52               LDA #'R         ;SET UP FOR FILE READ
0202  79A2  9D 12 78           STA FILE,X

0204  79A5                ;OPEN THE FILE

0206  79A5  A9 0E               LDA #14         ;SET THE LOGICAL FILE NUMBER
0207  79A7  85 D2               STA LA
0208  79A9  09 60               ORA #$60        ;SET THE SECONDARY ADDRESS
0209  79AB  85 D3               STA SA
0210  79AD  20 A5 F4           JSR OP94        ;OPEN THE FILE
0211  79B0  20 38 7B           JSR ENTER       ;ENTER FILENAME INTO TABLE
0212  79B3  B0 3F               BCS RECERR      ;BAD...
0213  79B5                ;
0214  79B5  A5 D4               LDA FA
0215  79B7  20 D2 F0           JSR TALK
0216  79BA  A5 D3               LDA SA
0217  79BC  20 93 F1           JSR TKSA
0218  79BF                REC10
0219  79BF  20 C0 F1           JSR ACPTR       ;GET A CHARACTER
0220  79C2  78                 SEI             ;KILL THE IRQ
0221  79C3  20 AE 7B           JSR SELBNK      ;SET UP EXP. RAM
0222  79C6  A0 00               LDY #0
0223  79C8  91 BB               STA (MASTER),Y
0224  79CA  A9 00               LDA #$00        ;RESET THE BANKS
0225  79CC  8D F0 FF           STA $FFF0
0226  79CF  58                 CLI             ;RESTORE THE IRQ
0227  79D0  20 C6 7B           JSR INCMAS
0228  79D3  A5 96               LDA STATUS      ;TERMINATE ON STATUS <> 0
0229  79D5  D0 04               BNE REC20
0230  79D7  90 E6               BCC REC10       ;GO DO MORE BYTES
0231  79D9  B0 19               BCS RECERR      ;OUT OF MEMORY
0232  79DB  C9 40         REC20 CMP #64          ;EOI IS OK
0233  79DD  D0 15               BNE RECERR      ;NOT EOI...
0234  79DF  20 AE F1           JSR UNTLK
0235  79E2  20 2F F7           JSR CLSEI       ;CLOSE THE FILE
0236  79E5  A0 15               LDY #21
0237  79E7  20 6B 7B           JSR ENTLST      ;PUT IN THE END ADDRESS
0238  79EA  A9 00               LDA #0
0239  79EC                RECOUT
0240  79EC  85 96               STA STATUS
0241  79EE  20 D0 7A           JSR RESREG      ;RESTORE REGISTERS
0242  79F1  4C 70 00           JMP CHRGET      ;BACK TO BASIC
0243  79F4                ;
0244  79F4                RECERR
0245  79F4  20 AE F1           JSR UNTLK       ;CLOSE THE CHANNEL
```

```
LINE# LOC   CODE        LINE

0246  79F7  20 2F F7          JSR CLSEI
0247  79FA  A9 20             LDA #32          ;SEND AN ERROR BACK
0248  79FC  D0 EE             BNE RECOUT
```

```
LINE# LOC    CODE          LINE

0250  79FE                LOAD
0251  79FE                ;
0252  79FE                ; THIS ROUTINE LOADS A FILE INTO THE EXPANSION MEMOR
0253  79FE                ;
0254  79FE  20 3A 7A             JSR LOADER      ; DO THE LOAD STUFF
0255  7A01  A6 20                LDX INDEX1+1
0256  7A03  86 2B                STX VARTAB+1
0257  7A05  A6 1F                LDX INDEX1
0258  7A07  86 2A                STX VARTAB
0259  7A09  20 E9 B5             JSR RUNC        ;CLEAR THE VARIABLES
0260  7A0C  20 B6 B4             JSR LNKPRG      ; LINK BASIC LINES
0261  7A0F  4C FF B3             JMP READY       ;SAY READY
0262  7A12                ;
```

```
LINE# LOC    CODE          LINE

0264  7A12                OVRLY
0265  7A12                ;
0266  7A12                ; OVERLAY AN EXISTING PROGRAM (I.E. LOAD FROM BASIC)
0267  7A12                ;
0268  7A12  20 3A 7A            JSR LOADER      ;LOAD THE FILE
0269  7A15  20 22 B6            JSR STXTPT      ;PICK UP THE FIRST LINE
0270  7A18  AD 2A 79            LDA WHERET+1    ;PUT CORRECT ADDRESS ON STACK
0271  7A1B  48                  PHA
0272  7A1C  AD 29 79            LDA WHERET
0273  7A1F  48                  PHA
0274  7A20  4C 0B B6            JMP FLOAD       ;GO RUN IT
0275  7A23                ;
```

```
LINE# LOC    CODE          LINE

0277  7A23                EXEC
0278  7A23                ;
0279  7A23                ; LOAD AND RUN WITH A CLEAR SO VARIABLES ARE GONE
0280  7A23                ;
0281  7A23  20 3A 7A            JSR LOADER        ;LOAD THE FILE
0282  7A26  A6 20               LDX INDEX1+1
0283  7A28  86 2B               STX VARTAB+1
0284  7A2A  A6 1F               LDX INDEX1
0285  7A2C  86 2A               STX VARTAB
0286  7A2E  20 E9 B5            JSR RUNC          ;CLEAR VARIABLES
0287  7A31  20 B6 B4            JSR LNKPRG        ;LINK LINES
0288  7A34  20 22 B6            JSR STXTPT        ;SET POINTER TO FIRST LINE
0289  7A37  4C 4A B7            JMP NEWSTT        ;EXEC IT
0290  7A3A                ;
```

```
LINE# LOC    CODE          LINE

0292  7A3A                ;
0293  7A3A                ; ROUTINE TO LOAD FILES INTO LOW RAM FROM 64K
0294  7A3A                ;
0295  7A3A                LOADER
0296  7A3A  20 D7 7A             JSR GETFN       ;READ THE FILENAME
0297  7A3D  A5 96                LDA STATUS
0298  7A3F  D0 5C                BNE LDERR       ;BAD FILENAME....
0299  7A41  20 7D 7B             JSR LUKUP       ;LOOKUP THE FILE IN DIRECTORY
0300  7A44  B0 57                BCS LDERR       ;NOT FOUND...
0301  7A46                ;
0302  7A46  98                   TYA             ;INDEX INTO THE DIR FOR ADDRES
0303  7A47  18                   CLC
0304  7A48  69 12                ADC #18
0305  7A4A  A8                   TAY
0306  7A4B  B9 29 78             LDA FITAB,Y     ;START LO
0307  7A4E  85 21                STA INDEX2
0308  7A50  C8                   INY
0309  7A51  B9 29 78             LDA FITAB,Y     ;START HI
0310  7A54  85 22                STA INDEX2+1
0311  7A56                                       ;
0312  7A56  C8                   INY
0313  7A57  B9 29 78             LDA FITAB,Y     ;GET THE MAPPER CONTROL BYTE
0314  7A5A  85 D1                STA FNLEN
0315  7A5C                ;
0316  7A5C  C8                   INY
0317  7A5D  B9 29 78             LDA FITAB,Y     ;END LO
0318  7A60  85 C9                STA EAL
0319  7A62  C8                   INY
0320  7A63  B9 29 78             LDA FITAB,Y     ;END HI
0321  7A66  85 CA                STA EAL+1
0322  7A68                ;
0323  7A68                ; LOOP TO LOAD DATA INTO LO RAM
0324  7A68                ;
0325  7A68                LODLP
0326  7A68  78                   SEI             ;NO IRQ ALLOWED
0327  7A69                LODLP0
0328  7A69  A0 00                LDY #$0
0329  7A6B  A6 D1                LDX FNLEN       ;SET UP THE BANKS
0330  7A6D  D0 04                BNE LODLP1
0331  7A6F  A2 80                LDX #$80        ;SELECT 0,2
0332  7A71  D0 02                BNE STLP
0333  7A73                LODLP1
0334  7A73  A2 88                LDX #$88        ;SELECT 1,3
0335  7A75                STLP
0336  7A75  8E F0 FF             STX $FFF0
0337  7A78                ;
0338  7A78                ;
0339  7A78  B1 21                LDA (INDEX2),Y  ;LOAD ADDRESS LO
0340  7A7A  85 1F                STA INDEX1
0341  7A7C  20 9F 7A             JSR BUMPX
0342  7A7F                                       ;
0343  7A7F  B1 21                LDA (INDEX2),Y  ;LOAD ADDRESS HI
0344  7A81  85 20                STA INDEX1+1
0345  7A83  20 9F 7A             JSR BUMPX
0346  7A86                ;
```

```
LINE# LOC    CODE          LINE

0347  7A86                 LODLP2
0348  7A86  B1 21                 LDA (INDEX2),Y ;TRANSFER A BYTE
0349  7A88  91 1F                 STA (INDEX1),Y
0350  7A8A                 ;
0351  7A8A  E6 1F                 INC INDEX1      ;BUMP THE DESTINATION POINTER
0352  7A8C  D0 02                 BNE LODLP3
0353  7A8E  E6 20                 INC INDEX1+1
0354  7A90                 LODLP3
0355  7A90  20 9F 7A              JSR BUMPX       ;BUMP SOURCE POINTER
0356  7A93  90 F1                 BCC LODLP2      ;DONE YET?
0357  7A95                 ;
0358  7A95  A2 00                 LDX #0          ;RESTORE THE ROM'S
0359  7A97  8E F0 FF              STX $FFF0
0360  7A9A  58                   CLI             ;WE ARE DONE LET IRQ GO
0361  7A9B  18                   CLC
0362  7A9C  60                   RTS
0363  7A9D                 LDERR                 ;LOAD ERROR
0364  7A9D  38                   SEC
0365  7A9E  60                   RTS
0366  7A9F                 ;
0367  7A9F                 ;SUBROUTINE TO HANDLE DATA POINTERS TO EXPANSION MEM
0368  7A9F                 ;
0369  7A9F                 BUMPX
0370  7A9F  E6 21                 INC INDEX2
0371  7AA1  D0 0F                 BNE TSTXX
0372  7AA3  E6 22                 INC INDEX2+1
0373  7AA5  D0 0B                 BNE TSTXX
0374  7AA7  A2 80                 LDX #$80        ;CROSSED OVER TO OTHER BLOCK
0375  7AA9  86 D1                 STX FNLEN
0376  7AAB  86 22                 STX INDEX2+1
0377  7AAD  A9 B8                 LDA #%10111000
0378  7AAF  8D F0 FF              STA $FFF0
0379  7AB2                 TSTXX                 ;CHECK FOR END OF FILE
0380  7AB2  A6 C9                 LDX EAL
0381  7AB4  E4 21                 CPX INDEX2
0382  7AB6  D0 08                 BNE DONEX
0383  7AB8  A6 CA                 LDX EAL+1
0384  7ABA  E4 22                 CPX INDEX2+1
0385  7ABC  D0 02                 BNE DONEX
0386  7ABE  38                   SEC             ;DONE WITH LOAD
0387  7ABF  60                   RTS
0388  7AC0                 DONEX
0389  7AC0  18                   CLC
0390  7AC1  60                   RTS
0391  7AC2                 ;
```

```
LINE# LOC    CODE        LINE

0393  7AC2                ;
0394  7AC2                ;INCREMENT POINTER TO TEXT
0395  7AC2                ;
0396  7AC2                INCPTR
0397  7AC2  E6 77                 INC GETPTR
0398  7AC4  D0 02                 BNE INCRTS
0399  7AC6  E6 78                 INC GETPTR+1
0400  7AC8                INCRTS
0401  7AC8  60                    RTS
0402  7AC9                ;
0403  7AC9                ;SAVE AND RESTORE ROUTINES FOR .A, .X, .Y
0404  7AC9                ;
0405  7AC9                SAVREG
0406  7AC9  85 B3                 STA WSW
0407  7ACB  86 B4                 STX T1
0408  7ACD  84 B5                 STY T2
0409  7ACF  60                    RTS
0410  7AD0                ;
0411  7AD0                RESREG
0412  7AD0  A5 B3                 LDA WSW
0413  7AD2  A6 B4                 LDX T1
0414  7AD4  A4 B5                 LDY T2
0415  7AD6  60                    RTS
```

```
LINE# LOC   CODE          LINE


0417  7AD7                ;
0418  7AD7                ; GET FILENAME FROM USER COMMAND
0419  7AD7                ;
0420  7AD7                GETFN
0421  7AD7  20 70 00             JSR CHRGET
0422  7ADA  F0 57                BEQ BADFI       ;NOT THERE ERROR
0423  7ADC  C9 22                CMP #'"         ;LOOK FOR OPEN QUOTE
0424  7ADE  D0 F7                BNE GETFN
0425  7AE0                ;
0426  7AE0  A9 00                LDA #0
0427  7AE2  85 96                STA STATUS
0428  7AE4  AA                   TAX
0429  7AE5  A8                   TAY
0430  7AE6  A9 12                LDA #<FILE
0431  7AE8  85 DA                STA FNADDR
0432  7AEA  A9 78                LDA #>FILE
0433  7AEC  85 DB                STA FNADDR+1
0434  7AEE                FLOOP
0435  7AEE  20 C2 7A             JSR INCPTR
0436  7AF1  B1 77                LDA (GETPTR),Y
0437  7AF3  C9 22                CMP #'"         ;LOOK FOR END QUOTE
0438  7AF5  F0 0A                BEQ ENDFI
0439  7AF7  9D 12 78             STA FILE,X
0440  7AFA  E8                   INX
0441  7AFB  E0 12                CPX #18
0442  7AFD  90 EF                BCC FLOOP       ;GO GET MORE
0443  7AFF  B0 32                BCS BADFI       ;TOO LONG
0444  7B01                ENDFI
0445  7B01  86 D1                STX FNLEN       ;SAVE LENGTH (INDEX)
0446  7B03  A9 00                LDA #0          ;PAD FILENAME WIH $00 ON RIGHT
0447  7B05  E0 12          FI10  CPX #18
0448  7B07  B0 06                BCS ENDFIA
0449  7B09  9D 12 78             STA FILE,X
0450  7B0C  E8                   INX
0451  7B0D  D0 F6                BNE FI10
0452  7B0F                ENDFIA
0453  7B0F  60                   RTS
0454  7B10                ;
0455  7B10                ; GET THE FILE TYPE
0456  7B10                ;
0457  7B10                GTYPE
0458  7B10                GCOMMA
0459  7B10  20 70 00             JSR CHRGET
0460  7B13  F0 1E                BEQ BADFI       ;NOT ENOUGH INFO.....
0461  7B15  C9 2C                CMP #',
0462  7B17  D0 F7                BNE GCOMMA
0463  7B19  A6 D1                LDX FNLEN       ;GET THE CURRENT FILENAME LENG
0464  7B1B  9D 12 78             STA FILE,X
0465  7B1E  9D 14 78             STA FILE+2,X    ;PUT COMMA HERE ALSO
0466  7B21  20 70 00             JSR CHRGET      ;GET TYPE BYTE (S,U,P,R)
0467  7B24  F0 0D                BEQ BADFI
0468  7B26  E8                   INX
0469  7B27  9D 12 78             STA FILE,X      ;NO VALUE CHECK IS DONE!!!
0470  7B2A  E8                   INX             ;SKIP TO PUT IN THE READ OR WR
0471  7B2B  E8                   INX             ;BUT DO NOT PUT IN HERE
```

```
LINE# LOC   CODE          LINE


0472  7B2C  86 D1             STX FNLEN
0473  7B2E                   ;
0474  7B2E                   ;  PUT IN THE DEVICE NUMBER CHECK LATER**************
0475  7B2E                   ; FOR NOW DEFAULT TO 8 ONLY
0476  7B2E                   ;
0477  7B2E  A9 08             LDA #8
0478  7B30  85 D4             STA FA
0479  7B32  60               RTS
0480  7B33                   ;
0481  7B33                   BADFI
0482  7B33  A9 01             LDA #$01
0483  7B35  85 96             STA STATUS
0484  7B37  60               RTS
0485  7B38                   ;
0486  7B38                   ; ENTER FILE INTO TABLE
0487  7B38                   ; PUT IN LOAD ADDRESS AND SET STATUS IN CARRY
0488  7B38                   ;
0489  7B38                   ENTER
0490  7B38  20 7D 7B          JSR LUKUP       ;LOOK FOR FILE IN TABLE
0491  7B3B  90 3E             BCC BADENT      ; DUP FOUND...
0492  7B3D                   ;
0493  7B3D                   ; LOOK FOR BLANK ENTRY
0494  7B3D                   ;
0495  7B3D  A0 E1             LDY #ENTMAX
0496  7B3F                   EMPTY
0497  7B3F  B9 29 78          LDA FITAB,Y
0498  7B42  F0 09             BEQ EMPFND
0499  7B44  98               TYA
0500  7B45  38               SEC
0501  7B46  E9 19             SBC #ENTRY
0502  7B48  A8               TAY
0503  7B49  D0 F4             BNE EMPTY
0504  7B4B  F0 2E             BEQ BADENT
0505  7B4D                   ;
0506  7B4D                   EMPFND                   ;FOUND AN ENTRY
0507  7B4D  98               TYA                     ;POINT INDEX1 TO ENTRY
0508  7B4E  18               CLC
0509  7B4F  69 29             ADC #<FITAB
0510  7B51  85 1F             STA INDEX1
0511  7B53  A9 78             LDA #>FITAB
0512  7B55  69 00             ADC #0
0513  7B57  85 20             STA INDEX1+1
0514  7B59                   ;
0515  7B59                   ;PLACE ENTRY INTO DIR
0516  7B59                   ;
0517  7B59  A0 00             LDY #0
0518  7B5B                   ENTMOR
0519  7B5B  B9 12 78          LDA FILE,Y
0520  7B5E  C9 2C             CMP #',         ;ONLY PUT IN DRIVE # AND NAME
0521  7B60  F0 07             BEQ ENTFST
0522  7B62  91 1F             STA (INDEX1),Y
0523  7B64  C8               INY
0524  7B65  C0 12             CPY #18
0525  7B67  90 F2             BCC ENTMOR
0526  7B69                   ;
```

```
LINE# LOC   CODE        LINE

0527  7B69              ENTFST
0528  7B69  A0 12               LDY #18
0529  7B6B              ;
0530  7B6B              ENTLST                    ;ENTER END ADDRESS
0531  7B6B  A5 BB             LDA MASTER      ;SAVE THE LOAD ADDRESS
0532  7B6D  91 1F             STA (INDEX1),Y
0533  7B6F  C8               INY
0534  7B70  A5 BC             LDA MASTER+1
0535  7B72  91 1F             STA (INDEX1),Y
0536  7B74  C8               INY             ;PUT IN THE BANK CONTROL BYTE
0537  7B75  A5 BE             LDA MASTER+3
0538  7B77  91 1F             STA (INDEX1),Y
0539  7B79  18               CLC
0540  7B7A  60               RTS
0541  7B7B              BADENT                    ;COME HERE FOR BAD EXIT
0542  7B7B  38               SEC
0543  7B7C  60               RTS
0544  7B7D              ;
0545  7B7D              ; LOOKUP FILE IN DIRECTORY
0546  7B7D              ;
0547  7B7D              LUKUP
0548  7B7D  A0 E1             LDY #ENTMAX
0549  7B7F              LUK10
0550  7B7F  B9 29 78           LDA FITAB,Y
0551  7B82  D0 09             BNE LUKCMP
0552  7B84              LUK20
0553  7B84  98               TYA             ;MOVE TO NEXT FILE
0554  7B85  38               SEC
0555  7B86  E9 19             SBC #ENTRY
0556  7B88  A8               TAY
0557  7B89  D0 F4             BNE LUK10
0558  7B8B  F0 1F             BEQ NOTFND
0559  7B8D              ;
0560  7B8D              LUKCMP
0561  7B8D  84 B6             STY T3          ;SAVE .Y WHILE DOING THE COMPA
0562  7B8F  A2 00             LDX #0
0563  7B91              LUK30
0564  7B91  BD 12 78           LDA FILE,X
0565  7B94  C9 2C             CMP #',         ;COMMA MEANS DONE
0566  7B96  F0 0B             BEQ FOUND
0567  7B98  D9 29 78           CMP FITAB,Y
0568  7B9B  D0 0A             BNE TRYMOR
0569  7B9D  C8               INY
0570  7B9E  E8               INX
0571  7B9F  E0 12             CPX #18
0572  7BA1  90 EE             BCC LUK30
0573  7BA3              ;
0574  7BA3              FOUND                     ;I FOUND IT
0575  7BA3  A4 B6             LDY T3          ; .Y POINTS TO FILE
0576  7BA5  18               CLC
0577  7BA6  60               RTS
0578  7BA7              TRYMOR                    ;I AM TRYING TO FIND IT
0579  7BA7  A4 B6             LDY T3
0580  7BA9  4C 84 7B           JMP LUK20
0581  7BAC              NOTFND                    ;I LOST IT
```

```
LINE# LOC   CODE         LINE

0582  7BAC  38           SEC
0583  7BAD  60           RTS
0584  7BAE               ;
0585  7BAE               ;THIS ROUTINE TURNS ON THE PROPER BANK
0586  7BAE               ;
0587  7BAE               SELBNK
0588  7BAE  A6 BD        LDX MASTER+2
0589  7BB0  D0 06        BNE SEL10
0590  7BB2  A2 80        LDX #$80        ;TURN ON BANKS 0 AND 2
0591  7BB4  8E F0 FF     STX $FFF0
0592  7BB7  60           RTS
0593  7BB8               SEL10
0594  7BB8  A2 88        LDX #$88        ;TURN ON BANKS 1 AND 3
0595  7BBA  8E F0 FF     STX $FFF0
0596  7BBD  A6 BD        LDX MASTER+2    ;ONE BYTE HAS THE $40 FLAG
0597  7BBF  70 04        BVS SEL30
0598  7BC1  A2 80        LDX #$80
0599  7BC3  86 BD        STX MASTER+2
0600  7BC5               SEL30
0601  7BC5  60           RTS
0602  7BC6               ;
0603  7BC6               ; BUMP UP THE MASTER POINTER AND SET THE $40 FLAG
0604  7BC6               ;
0605  7BC6               INCMAS
0606  7BC6  E6 BB        INC MASTER      ;BUMP POINTERS
0607  7BC8  D0 10        BNE MASRTS
0608  7BCA  E6 BC        INC MASTER+1
0609  7BCC  D0 0C        BNE MASRTS      ;CHECK FOR OVERFLOW OF BANK
0610  7BCE  A6 BD        LDX MASTER+2    ;WE WENT OVER
0611  7BD0  D0 0A        BNE INC10       ;OUT OF MEMORY?
0612  7BD2  A2 40        LDX #$40        ;SET FLAG FOR OVERFLOW
0613  7BD4  86 BD        STX MASTER+2
0614  7BD6  A2 80        LDX #$80        ;RESET POINTER HI
0615  7BD8  86 BC        STX MASTER+1
0616  7BDA               MASRTS
0617  7BDA  18           CLC
0618  7BDB  60           RTS
0619  7BDC               INC10                ;OUT OF MEMORY
0620  7BDC  38           SEC
0621  7BDD  60           RTS
0622  7BDE               ;
0623  7BDE               ;
0624  7BDE               ;THESE MUST BE AFTER THE DEFINITION
0625  7BDE               ;
0626  7BDE               ZZ1     =RECAL-1        ;RECALL FROM DISK
0627  7BDE               ZZ3     =LOAD-1         ;LOAD LOW MEMORY FROM EXPANSI
0628  7BDE               ZZ4     =OVRLY-1        ;OVERLAY PROGRAM IN MEMORY
0629  7BDE               ZZ5     =EXEC-1         ;EXECUTE WITHOUT OVERLAY
0630  7BDE               ZZ8     =QUIT-1         ;EXIT PROGRAM.
0631  7BDE                       .END

ERRORS = 0000
```

SYMBOL TABLE

SYMBOL	VALUE						
ACPTR	F1C0	AGAIN	796F	BADENT	7B7B	BADFI	7B33
BUMPX	7A9F	CHDGOT	D39F	CHRGET	0070	CHRGOT	0076
CLRCH	FFCC	CLSEI	F72F	CMD	780D	DONEX	7AC0
DS	0013	EAL	00C9	EMPFND	7B4D	EMPTY	7B3F
ENDFI	7B01	ENDFIA	7B0F	ENTER	7B38	ENTFST	7B69
ENTLST	7B6B	ENTMAX	00E1	ENTMOR	7B5B	ENTRY	0019
EXEC	7A23	FA	00D4	FCLOSE	F2E2	FI10	7B05
FILE	7812	FINDC	7960	FITAB	7829	FLOAD	B60B
FLOOP	7AEE	FNADDR	00DA	FNLEN	00D1	FOUND	7BA3
GCOMMA	7B10	GETFN	7AD7	GETPTR	0077	GOOO	792E
GTYPE	7B10	INC10	7BDC	INCMAS	7BC6	INCPTR	7AC2
INCRTS	7AC8	INDEX1	001F	INDEX2	0021	JUMP	792B
LA	00D2	LDERR	7A9D	LNKPRG	B4B6	LOAD	79FE
LOADER	7A3A	LOC	7800	LODLP	7A68	LODLP0	7A69
LODLP1	7A73	LODLP2	7A86	LODLP3	7A90	LUK10	7B7F
LUK20	7B84	LUK30	7B91	LUKCMP	7B8D	LUKUP	7B7D
MASRTS	7BDA	MASTER	00BB	MEMSIZ	0034	NCMD	0005
NEWSTT	B74A	NOTCMD	7972	NOTFND	7BAC	NOTME	7976
OP94	F4A5	OVRLY	7A12	QUIT	797D	READY	B3FF
REC10	79BF	REC20	79DB	RECAL	7992	RECERR	79F4
RECOUT	79EC	RESREG	7AD0	REST	7984	RUNC	B5E9
SA	00D3	SAVREG	7AC9	SEL10	7BB8	SEL30	7BC5
SELBNK	7BAE	START	7951	STATUS	0096	STLP	7A75
STRTS	796E	STXTPT	B622	T1	00B4	T2	00B5
T3	00B6	TABHI	7803	TABLO	7808	TABLP	793B
TABSET	7938	TALK	FOD2	TKSA	F193	TRYMOR	7BA7
TSTXX	7AB2	UNTLK	F1AE	VARTAB	002A	WEDGE	7930
WHERET	7929	WSW	00B3	ZZ1	7991	ZZ3	79FD
ZZ4	7A11	ZZ5	7A22	ZZ8	797C		

END OF ASSEMBLY

EXPANDED-BASIC

```
100 REM*********************************
110 REM* EXPANDED BASIC DEMO           *
120 REM*                              *
130 REM* BOOT EXPANDED BASIC           *
140 REM* AND THEN CACHE DEMO PROGRAM*
150 REM*********************************
160 IF A THEN SYS 7*16↑3+8*256
170 IF A THEN:!R,"STTEST",P:!E,"STTEST"
180 A=1:DLOAD "EXPANDED-BASIC"
```

STTEST

```
10 REM DEMO FOR MEMORY MANAGER
20 REM FIRST WE MUST GET THE STUFF INTO MEMORY
30 :!R,"TEST1",P
40 :!R,"TEST2",P
50 :!R,"TEST3",P
60 :!R,"TEST4",P
70 REM EXECUTE THE FIRST PROGRAM IN THE CHAIN
80 :!E,"TEST1"
```

TEST1

```
10 REM THIS IS A TEST
15 PRINT"TEST1":FORI=1TO200:NEXT
17 PRINT"PRINT THE NUMBERS FROM 1 TO 20"
19 FORI=1TO200:NEXT
20 FORI=1TO20
30 PRINT,I
40 NEXT
50 :!E,"TEST2"
```

TEST2

```
10 REM THIS IS A TEST
15 PRINT"TEST2":FORI=1TO200:NEXT
17 PRINT"PRINT THE NUMBERS FROM 20 TO 1"
19 FORI=1TO200:NEXT
20 FORI=20 TO 1 STEP -1
30 PRINT,I
40 NEXT
50 :!E,"TEST3""
```

TEST3

```
10 REM THIS IS A TEST
15 PRINT"■TEST3":FORI=1TO200:NEXT
17 PRINT"PRINT THE NUMBERS FROM 51 TO 70"
18 PRINT"ALONG WITH THE SQUARE-ROOT
19 FORI=1TO200:NEXT
20 FORI=51TO70
30 PRINT,I,SQR(I)
40 NEXT
50 :!E,"TEST4"
```

TEST4

```
10 REM THIS IS A TEST
15 PRINT"■TEST4":FORI=1TO200:NEXT
17 PRINT"YOU HAVE SEEN THE SYSTEM LOAD AND"
18 PRINT"EXECUTE THREE PROGRAMS FROM THE
20 PRINT"EXPANSION MEMORY.  IF YOU LOOK AT
30 PRINT"THE FILES WITH THE WORD 'TEST' IN THEM
40 PRINT"YOU WILL SEE HOW THIS IS DONE
45 PRINT"■■TYPE ■SPACE■ TO CONTINUE OR ■Q■ TO QUIT"
50 GETA$:IFA$=""THEN50
60 IFA$="Q"THEN:!Q:END
65 IFA$=" "THEN:!E,"TEST1"
70 IFA$="2"THEN:!E,"TEST2"
80 IFA$="3"THEN:!E,"TEST3"
```