

1982/27. DSKCPY - A disk copying utility for the Department of Mines
Perkin-Elmer minicomputer (Revision 1)

R.G. Richardson

Abstract

This program is intended to increase the security of data stored in the Perkin-Elmer minicomputer. Disks are copied with optional deletion of selected files, compression of waste space, elimination of overflow cylinders, and default write protection of all files copied.

THE PROGRAM

DSKCPY (Appendix 1)

The program creates a file containing the directory of the disk to be copied on that disk. This file requires two cylinders. The other disk is initialised, which takes approximately one minute, then the user may give the names of any files not to be copied. The program will copy any random access or sequential file. At the completion of copying, a directory of the new disk is output on the line printer. If files were not explicitly closed when used previously, some overflow cylinders may be allocated.

Non-standard subroutines are:

SYSCOM (I, ACTION) - pass the command packed in ACTION to the operating system for implementation. This is used by DOS in the following manner:

AS 0622, 0899 - assign the printer to logical unit 6 and the core buffer to logical unit 9.
PA C6, A - initialise the disk on physical unit C6.
AL NNNNNN, LPU, RRR, IIII - allocate a file NNNNNN as logical unit L on physical unit PU with record length IIII allowing RRR cylinders.
AC NNNNNN, LPU - activate file NNNNNN on physical unit PU as logical unit L.
AT NNNNNN, AA, PU - set the attributes of file NNNNNN on physical unit PU.

AA = 00 sequential read/write
= 01 sequential read only
= 02 sequential write only
= 10 random access read/write
= 11 random access read only
= 12 random access write only
= 20 direct access read/write

LI LPU - print a directory listing of physical unit PU on logical unit L.

POSITN (ILUN, IREC) - position logical unit ILUN ready to read or write record IREC.

USER INSTRUCTIONS

- (i) Run DSKCPY by typing RU DSKCPY
- (ii) At the pause mount your cartridge disk

- (iii) When both disks are write enabled type CO
- (iv) Enter the disk to copy from. This must be C6 or C7, depending on whether you are copying to or from the cartridge disk. Once you enter this the disk to be copied to is cleared.
- (v) If you wish all files copied to be write protected answer YES. By doing this you allow the system to perform some security checks before writing to a file.
- (vi) If you wish to copy all files on a disk answer YES. If you answer NO the names of the files not to be copied will be requested and checked. Once all names have been entered type NOMORE.
- (vii) At the end of the copy a directory of the newly created disk will be printed. Check this and also check for warning of direct access files not copied.

[22 July 1982]

APPENDIX 1
Program DSKCPY

```

FITL DSKCOPY - TO COPY AND TIDY SELECTED PORTIONS OF A DISK
THIS PROGRAM COPIES DISKS UNDER THE PERKIN-ELMER OPERATING
SYSTEM AND TIDIES THEM UP BY REMOVING OVERFLOW CYLINDERS
AND EXCESSIVE ALLOCATIONS OF CYLINDERS.
THE ONLY RESTRICTION ON FILES TO BE COPIED IS THAT THEY NOT
BE DIRECT ACCESS (ATTRIBUTES 20, 21).
THE OPTION IS AVAILABLE FOR SELECTED FILES NOT TO BE COPIED
AND FOR THE FILES AS THEY ARE COPIED TO BE WRITE PROTECTED
AT THE COMPLETION OF THE COPYING.
LOGICAL UNITS ARE CONTROLLED BY THE PROGRAM.
CONTROL INPUT IS FROM LOGICAL UNIT 5 (THE CONSOLE)
THE PRINTER (LU 6) IS ALSO USED.
    INTEGER*2 ACTION(11), AOK, YES, NO, ALINE(33), ATTRIB,
    . OVFLD, CYL, FFEED, BUFF(10000)
    IMPLICIT INTEGER*2 (I-N)
    DOUBLE PRECISION ANAME, FNAME, NOMORE, ASTAR, INDX
    INTEGER*2 ANUM(7), POINT
    LOGICAL IER, PROTCT
    EQUIVALENCE (ACTION(1), BUFF(1)), (ANAME, BUFF(40)),
    . (FNAME, BUFF(45)), (ANUM(1), BUFF(50))
TO SAVE SPACE
    DATA IC/1HC/, I1/1H1/, I2/1H2/, IC6/2HC6/, IC7/2HC7/, I6/1H6/,
    . I7/1H7/, IA/1HA/, IT/1HT/, IL/1HL/, ISPCE/1H /, ICOMMA/1H, /,
    . ISTAR/1H*/, YES/1HY/, NO/1HN/, FFEED/3072/, NOMORE/8HNOMORE /,
    . POINT/1H. /,
    . ASTAR/8H***** /, INDX/8HZZZ083 /
    CALL SYSCOM(I, 'AS 0622, 0899*')
ASSIGN PRINTER TO LOGICAL UNIT 6, CORE BUFFER TO LU 8
    CALL SYSCOM(I, 'PA C6*')
    CALL SYSCOM(I, 'PA C7*')
ENSURE ALL DISK FILES ARE CLOSED
    WRITE(5, 102)
102 FORMAT(' MOUNT YOUR OWN DISK'// ' WHEN BOTH DISKS ARE WRITE',
    . ' ENABLED CONTINUE RUN')
    PAUSE
    10 WRITE(5, 103)
103 FORMAT(' COPY FROM C6 OR C7?')
    READ(5, 104) IDSK
104 FORMAT(A2)
    IF (IDSK .NE. IC6 .AND. IDSK .NE. IC7) GOTO 10
NOT A VALID DISK NUMBER
NOW ASK IF THE DEFAULT IS TO WRITE PROTECT ALL FILES COPIED
    WRITE(5, 300)
300 FORMAT(' ALL FILES COPIED TO BE WRITE PROTECTED Y OR N?')
    READ(5, 110) AOK
    PROTCT=AOK .NE. NO
IF ANYTHING BUT AN EXPLICIT NO TYPED PROTECT THE FILES
    IPUI=I6
    IPUO=I7
    IF (IDSK .EQ. IC6) GOTO 11
    IPUI=I7
    IPUO=I6
SET INPUT AND OUTPUT PHYSICAL UNITS

NOW SET UP TEMPORARY OUTPUT FILE
    11 WRITE(8, 105) IPUI
105 FORMAT('AL ZZZ083, 4C', A1, ', 2, 80*')
    READ(8, 106) ACTION
106 FORMAT(11A2)
    CALL SYSCOM(I, ACTION)

```

```

ALLOCATE A FILE TO HOLD THE DIRECTORY
  WRITE(8,107) IPUI
107 FORMAT('AT ZZZ083,10,C',A1,'*')
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)
SET UP THE DIRECTORY FILE AS RANDOM ACCESS
  REWIND 4
REWIND THE DIRECTORY FILE
  WRITE(8,108) IPUI
108 FORMAT('LI 4C',A1,'*')
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)
PUT THE DISK DIRECTORY ONTO LU 4
  ENDFILE 4
FINISH WITH A FILEMARK
  WRITE(8,99) IPUD
  99 FORMAT('PA C',A1,'A*')
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)
INITIALISE OUTPUT DISK

NOW EDIT FILENAMES
  12 WRITE(5,109)
109 FORMAT(' COPY ALL FILES?')
  READ(5,110) AOK
110 FORMAT(A1)
  IF (AOK .EQ. YES) GOTO 17
COPYING ALL FILES
  IF (AOK .NE. NO) GOTO 12
CHOSEN AN INVALID OPTION
  14 WRITE(5,111)
111 FORMAT(' FILE TO BE DELETED - IF NO MORE TYPE NOMORE')
  READ(5,112) ANAME
112 FORMAT(A8)
  IF (ANAME .EQ. NOMORE) GOTO 17
FINISHED EDITTING FILENAMES SO OF TO THE COPYING

LOOK FOR THE NAME
  DO 15 I=2,300
  CALL POSITN(4,I)
  READ(4,112,END=14) FNAME
  IF (FNAME .EQ. ANAME) GOTO 16
FOUND A MATCHING NAME
  15 CONTINUE
CANT FALL OUT HERE AS DONT HAVE 300 FILES ON 1 DISK
  16 CALL POSITN(4,I)
  READ(4,113) ALINE
113 FORMAT(33A2)
  WRITE(5,114) ALINE
114 FORMAT(' DELETE'/1X,33A2/' Y OR N?')
  READ(5,110) AOK
  IF (AOK .EQ. NO) GOTO 14
DIDNT REARLY WANT TO DELETE IT
  IF (AOK .NE. YES) GOTO 16
DIDNT ANSWER YES OR NO
  CALL POSITN(4,I)
  WRITE(4,112) ASTAR
WRITE A FILENAME OF ALL* TO INDICATE NOT TO COPY
  GOTO 14
GET NEXT FILENAME

```

NOW FOR THE COPYING

17 CONTINUE

J=1

18 J=J+1

CALL POSITN(4, J)

READ(4, 113, END=30) ALINE

WRITE(8, 113) ALINE

COPY A LINE FORM THE DISK DIRCTORY FILE TO THE CORE BUFFER

READ(8, 112) ANAME

READ THE FILENAME

IF (ANAME .EQ. ASTAR .OR. ANAME .EQ. INDX) GOTO 18

DONT COPY THIS FILE

READ(8, 115) ICYL, IRECL, ATTRIB, OVFL0

115 FORMAT(8X, I3, I5, 7X, I4, I2)

READ THE NUMBER OF CYLINDERS ALLOCATED, THE RECORDLENGTH (BYTES),

THE FILE ATTRIBUTES AND THE NUMBER OF OVERFLOW CYLINDERS NEEDED

IF (ATTRIB .LT. 20) GOTO 19

NOT DIRECT ACCESS FILES

WRITE(6, 116) ALINE

116 FORMAT(1X, 33A2, ' DIRECT ACCESS AND NOT COPIED')

GOTO 18

NOW GET THE FILE ACTIVATED AND ALLOCATED

19 CONTINUE

ICYL=ICYL+OVFL0

THE ACTUAL NUMBER OF CYLINDERS IS THE SUM OF THE ALLOCATED SPACE AND ANY OVERFLOWS

ALINE(1)=IA

ALINE(2)=IC

ALINE(3)=ISPCE

HAVE "AC "

READ(8, 117) (ALINE(I), I=4, 9)

117 FORMAT(6A1)

READ THE FILENAME IN A1

DO 20 I=5, 9

IF (ALINE(I) .EQ. ISPCE) GOTO 21

LOOK FOR A SPACE AT THE END OF THE FILENAME

20 CONTINUE

I=10

IF DONT FIND SPACE WANT TO ENSURE AT END OF FILENAME

21 ALINE(I)=ICOMMA

HAVE "AC NNNNNN, "

ICMA=I

NOTE WHERE THE COMMA IS FOR USE WITH ACTIVATE

ALINE(I+1)=I1

HAVE "AC NNNNNN, 1"

ALINE(I+2)=IC

HAVE "AC NNNNNN, 1C"

ALINE(I+3)=IPUI

AC NNNNNN, 1C6 9AY

ICMA2=I+4

NOTE WHERE THE SECOND COMMA OF ALOCATE HAS TO GO

ALINE(ICMA2)=ISTAR

END OF ACTIVATE HAVE "AC NNNNNN, 1C6*"

READ(8, 124) ANUM

124 FORMAT(16X, 7A1)

GET THE FIELD FOR THE NUMBER OF RECORDS

ANUM(7)=POINT

ADD DECIMAL POINT AT END

```

WRITE(8,125) ANUM
125 FORMAT(7A1)
READ(8,126) USED
126 FORMAT(F7.0)
READ IN FLOATING POINT
THE NUMBER OF RECORDS IN THE FILE WHICH MAY BE UP TO 65535 WITH DOS
BUT MAY BE MORE THAN THIS BECAUSE THE COUNT IS RESET ALTHOUGH THERE
MAY BE 1. E6 OR MORE RECORDS FOR VERY SHORT RECORD LENGTHS
CYL=AIN(T(USED/(48.*AIN(T(FLOAT(256/IRECL))))))+1.0
CALCULATE THE THEORETICAL NUMBER OF CYLINDERS REQUIRED.

```

```

NOW ACTIVATE THE INPUT FILE
WRITE(8,118) ALINE
118 FORMAT(33A1)
READ(8,186) ACTION
CALL SYSCOM(I,ACTION)
ACTIVATE THE INPUT FILE
REWIND 1
AND REWIND IT READY FOR READING
IF (USED.EQ.0.) GOTO 22
DONT KNOW HOW MANY RECORDS SO COUNT THEM
IF ((ICYL-CYL).LE.1) GOTO 36
HAVE RIGHT NUMBER OF CYLINDERS

```

```

OTHERWISE COUNT THE NUMBER OF RECORDS
USED=0.
COUNTER FOR RECORDS
22 READ(1,ERR=23,END=23) ITMP
USED=USED+1.0
GOTO 22
GO DOWN THE FILE COUNTING THE RECORDS
23 CYL=AIN(T(USED/(48.*AIN(T(FLOAT(256/IRECL))))))+1.0
CALCULATE NUMBER OF CYLINDERS NEEDED - COPE WITH VERY LONG FILES
IF ((ICYL-CYL).GT.1) ICYL=CYL
REDUCE THE NUMBER OF CYLINDERS ALLOCATED TO WHAT IS NEEDED

```

```

NOW ALLOCATE THE NEW FILE
36 ALINE(2)=IL
"AL NNNNNN,1C6"
ALINE(ICMA+1)=I2
"AL NNNNNN,2C6"
ALINE(ICMA2-1)=IPUO
"AL NNNNNN,2C7"
ALINE(ICMA2)=ICOMMA
NOW HAVE NAME, LU AND PHYS. UNIT OK
"AL NNNNNN,2C7, "
CALL DIGIT(NDIG,ICYL)
GET THE NUMBER OF DIGITS TO WRITE ICYL AND WRITE TO THE CORE BUFFER
IN THE APPROPRIATE FORMAT
IUP=ICMA2+NDIG
THE LOWEST ORDER DIGIT OF ICYL WILL BE HERE
IDN=ICMA2+1
THE HIGHEST ORDER DIGIT OF ICYL WILL BE HERE
READ(8,117) (ALINE(I),I=IDN,IUP)
NOW HAVE "AL NNNNNN,2C6,34"
ALINE(IUP+1)=ICOMMA
"AL NNNNNN,2C7,34, "
CALL DIGIT(NDIG,IRECL)
DO THE SAME FOR THE RECORDLENGTH
IDN=IUP+2

```

```

IUP=IUP+NDIG+1
READ(8,117) (ALINE(I), I=IDN, IUP)
READ THE RECORDLENGTH "AL NNNNNN, 207, 34, 80"
ALINE(IUP+1)=ISTAR
"AL NNNNNN, 207, 34, 80*"
WRITE(8,118) ALINE
READ(8,106) ACTION
CALL SYSCOM(I, ACTION)
ALLOCATE THE OUTPUT FILE
REWIND 1
REWIND THE INPUT FILE
REWIND 2
REWIND THE OUTPUT FILE
NBUFF=20000/IRECL
NUMBER OF RECORDS THAT WILL FIT IN THE BUFFER ARRAY IN A2 FORMAT
IF (MOD(IRECL, 2) .NE. 0) GOTO 31
A FILE WITH AN ODD NUMBER OF BYTES PER RECORD
ISTEP=IRECL/2
THE NUMBER OF WORDS IN THE ARRAY TO FILL AT EACH CALL
27 ISTAT=1
USED TO REMEMBER IF AT END OF FILE
IDN=1
DO 24 I=1, NBUFF
CALL INBIN(BUFF(IDN), ISTEP, ISTAT)
USE A BINARY READ
IF (ISTAT .EQ. 0) GOTO 25
HIT THE END OF FILE
IDN=IDN+ISTEP
INCREMENT UP THE BUFFER ARRAY
24 CONTINUE
I=NBUFF+1
ENSURE IT IS DEFINED IF COME OUT END OF LOOP
25 IF(I .EQ. 1) GOTO 28
HIT END OF FILE ON FIRST READ OF LOOP
I=I-1
IDN=1
DO 26 K=1, I
CALL OUTBIN(BUFF(IDN), ISTEP)
IDN=IDN+ISTEP
26 CONTINUE
END OF LOOP FOR WRITING OUT WHAT WAS PUT IN
IF (ISTAT .EQ. 1) GOTO 27
HADNT HIT EOF SO BACK ROUND
GOTO 28
HAD HIT AN EOF

NOW TO COPE WITH FILES HAVING AN ODD NUMBER OF BYTES IN EACH RECORD
31 IDN=1
ISTAT=1
IUP=IDN+IRECL-1
NBUFF=NBUFF/2
ONLY GOING TO GET ONE BYTE PER WORD INSTEAD OF TWO
DO 32 I=1, NBUFF
READ(1, 200, END=33, ERR=33) (BUFF(K), K=IDN, IUP)
200 FORMAT(256A1)
IDN=IDN+IRECL
IUP=IUP+IRECL
INCREMENT START AND END COUNTRS
32 CONTINUE
I=NBUFF+1

```

```

GOTO 34
IF FALL THROUGH HAVENT HIT EOF
33 ISTAT=0
HIT EOF SO REMEMEBER IT
  IF (I .EQ. 1) GOTO 28
HIT EOF IN THE FIRST RECORD READ IN THE LOOP
34 IDN=1
  IUP=IDN+IRECL-1
LIMITS
  I=I-1
TAKE INTO ACCOUNT THAT HAVE INCREMENTED I BY 1
  DO 35 II=1, I
  WRITE(2,200) (BUFF(K),K=IDN,IUP)
  IDN=IDN+IRECL
  IUP=IUP+IRECL
35 CONTINUE
  IF (ISTAT .EQ. 1) GOTO 31
FINISH COPYING RECORDS WITH ODD NO OF BYTES

28 CONTINUE
TO HERE IF HIT EOF
  ENDFILE 2
WRITE A FILE MARK
AND SET THE ATTRIBUTES
  ALINE(2)=IT
HAVE "AT NNNNNN,207,34,80"
  IF (PROTCT .AND. MOD(ATTRIB,2) .EQ. 0) ATTRIB=ATTRIB+1
IF THE OPTION OF PROTECTING ALL FILES COPIED IS CHOSEN CHANGE
THE ATTRIBUTE IF IT IS EVEN (I. E. 0,10 BECOME 1,11)
  CALL DIGIT(NDIG,ATTRIB)
  IDN=ICMA+1
  IUP=ICMA+NDIG
WANT TO KEEP THE COMMA AFTER THE FILENAME
  READ(8,117) (ALINE(I),I=IDN,IUP)
"AT NNNNNN,10"
  ALINE(IUP+1)=ICOMMA
" AT NNNNNN,10, "
  ALINE(IUP+2)=IC
  ALINE(IUP+3)=IPUD
"AT NNNNNN,10,C7"
  ALINE(IUP+4)=ISTAR
"AT NNNNNN,10,C7*"
  WRITE(8,118) ALINE
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)
PROTECT THE OUTPUT FILE
WITH THE ORIGINAL ATTRIBUTE
  CALL SYSCOM(I,'CL*')
CLOSE ALL OPEN FILES
  GOTO 18
GET NEXT FILENAME

TO HERE WHEN ALL FILES COPIED
30 CONTINUE
  WRITE(8,121) IPUI
121 FORMAT('DE ZZZ083,C',A1,'*')
"DE ZZZ083,C6"
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)

```

```

: DELETE THE FILE HOLDING THE DIRECTORY
  CALL SYSCOM(I, 'PA C6*')
  CALL SYSCOM(I, 'PA C7*')
: DE-ALLOCATE ALL DISK FILES
  WRITE(6,122) FFEED
122 FORMAT(A2)
: THROUGH A NEW PAGE ON THE PRINTER
  WRITE(8,123) IPUD
123 FORMAT('LI 6C',A1,'*')
  READ(8,106) ACTION
  CALL SYSCOM(I,ACTION)
"LI 6C7" PRODUCE A DIRECTORY LISTING OF THE NEW FILE
  CALL SYSCOM(I, 'AS 0620*')
REASSIGN LU 6 TO VDU FROM PRINTER
  STOP
  END
  SUBROUTINE INBIN(IBUF,N, ISTAT)
  INTEGER*2 IBUF(N),N, ISTAT
  READ(1,END=10,ERR=10) IBUF
  RETURN
10 ISTAT=0
  RETURN
  END
  SUBROUTINE OUTBIN(IBUF,N)
  INTEGER*2 IBUF(N),N
  WRITE(2) IBUF
  RETURN
  END
  SUBROUTINE DIGIT(NDIG,N)
  INTEGER*2 NDIG,N
  NDIG=1
  IF (N .GE. 10) NDIG=2
  IF (N .GE. 100) NDIG=3
  IF (NDIG .EQ. 1) WRITE(8,100) N
  IF (NDIG .EQ. 2) WRITE(8,200) N
  IF (NDIG .EQ. 3) WRITE(8,300) N
100 FORMAT(I1)
200 FORMAT(I2)
300 FORMAT(I3)
  RETURN
  END

```