DECUS

PROGRAM LIBRARY

DECUS NO.	8-177
TITLE	COPY
AUTHOR	Alexander Symthe Submitted by: Theodore Green
COMPANY	Taft School Watertown, Connecticut
DATE	March 1, 1969
SOURCE LANGUAGE	PAL-D

Although this program has been tested by the contributor, no warranty, express or implied, is made by the contributor, Digital Equipment Computer Users Society or Digital Equipment Corporation as to the accuracy or functioning of the program or related program material, and no responsibility is assumed by these parties in connection therewith.

COPY

DECUS Program Library Write-up

DECUS No. 8-177

ABSTRACT

COPY is in effect an extension of PIP. It is used to copy disk files onto paper tape or from paper tape onto disk. COPY will copy any of the five file types--ASCII, Binary, FORTRAN Binary, User, and System. COPY's major advantage is that it saves time in putting files on disk and in getting them off. Its primary disadvantage is that everything is punched in COPY fromat; if one wishes to know what his tape says, he must reload it with COPY onto disk and use a more conventional method to translate it. In short, tapes punched with COPY must be loaded with COPY.

The author's primary goal in developing COPY was to expand the effective size of the disk by making a more flexable FAILSAFE program. COPY helps to do this in two ways: (1) it makes it easy to reload systems programs allowing the programmer to delete system sections of the disk and not have to spend an inordinate amount of time reloading them. (2) COPY makes it possible for the programmer to easily and simply punch and load ASCII and other user type files. For example, it the programmer has a long PAL program for which he needs PALD, DDT, EDIT, and lots of room, he can easily load the long DDT tape, and delete other systems tapes. When he has finished his intermediate run, he can easily punch his ASCII files with COPY and reload the system programs he deleted. Then, when he wishes to continue work on his program, he can delete the system programs again and reload his ASCII files. As an example of the speed which COPY gives the user, the author reloaded the entire disk with ten systems tapes in five minutes (using a high speed reader).

REQUIREMENTS

Storage

COPY uses locations 0-2777, but only 0-1577 need be saved on disk; the rest are used by the Command Decoder and for various internal buffers.

Other Programs Needed

COPY requires that the Disk Monitor I/O routine be in core and that the command decoder be stored on disk starting in block 15.

Equipment

Copy was written to run on a PDP-8/S with up to four disks, a high speed reader and punch and a Teletype; it could be easily adapted to utilize the extra features of a larger configuartion. At present, however, it will only recognize and use those parts of the system previously mentioned.

USAGE

Loading

COPY is loaded with the binary loader or with LOAD. It can be saved on disk and called through CALL/SAVE of the Disk Monitor System.

Start-up and/or Entry

The starting address of COPY is 0200_8.

Errors in Usage

If an error occurs, the program will type an error message in the form of "ERROR" followed by a four digit number. On a checksum error in reading, the machine will halt with the discrepancy displayed in the accumulator.

0303	Error in reading in Command Decoder
0434	Input/Output device unrecognizable (probably not implemented.)
0504	Disk storage exceeded
0551	Error in reading SAM block
0616	Error in Disk read
0642	Error in Disk write
1235	Error in writing Directory Name block
1261	Disk output file name not in directory
1270	Error in reading Directory Name block

One may also encounter errors when the computer is under the control of the Command Decoder. For a list of those error messages, see the appropriate manual.

Recovery form Such Errors

In the case of the four-digit error message, control is returned to the beginning of the program. On a checksum error, if the operator presses CONTINUE, the error will be ignored. No other recoveries are possible.

DISCRIPTION

Discussion

One first loads the program and then starts it. The program will type:

FILE-TYPE -

To this, the operator must reply with a letter standing for one of the five possible file types--A, B, F, U, S; these letters signify, respectively, ASCII, Binary, FORTRAN Binary, User, and System files. (The program will ignore any illegal characters.) The program then calls the Command Decoder which types:

*OUT-

and

*IN-

The replies to these questions are the same as the usual replies to the Command Decoder.

After the Command Decoder has finished, control returns to COPY, which performs the copy. If the copy is from disk to tape, COPY will give leader, then the file, then trailer. It will then stop punching and wait for either a CONTROL/C which takes control back to the monitor, or a CONTROL/P, which takes control back to the beginning of the program. This delay is to permit the operator to switch the punch off. If the input is from punched tape, the machine types an up-arrow and waits for a CONTROL/P before proceeding. If the copy is disk to disk, it is done and the program automatically starts over.

Examples (underlined letters typed by computer)

```
.LOAD                          /load binary tape
*IN-R:
*OPT-1
SA=7600
↑↑

.SAVE COPY!0-1577;200          /save COPY
.COPY                          /call COPY

FILE-TYPE - A                  /ASCII file

*OUT-T:                        /output to Teletype
*IN-S:ACS1                     /input from file named ACS1

FILE-TYPE - S                  /systems program

*OUT-S:FOCL                    /reload FOCAL
*IN-R:                         /from paper tape
↑

FILE-TYPE - (control/c         /job finished; (control/c) takes
.                              /control back to monitor
```

Applications

COPY is very useful in taking long ASCII files on and off disk; but, it is even more useful for reloading disk in event of its being erased. COPY provides a speedy way of getting files on and off disk.

METHODS

Discussion

COPY utilizes the disk monitor I/O routine and the Command Decoder to find and place the files on and off disk. The program starts by asking file type so that it can convey this information to the Command Decoder. Then it calls on the Command Decoder which returns with input and output devices. The program decodes these, sets the proper internal switches and flags and then starts to make the copy. The data is read in, translating if necessary, and then written out, translating if necessary, until the program encounters a link of zero. This signals the end of the input, at which time the program takes appropriate measures, and then restarts.

FORMAT

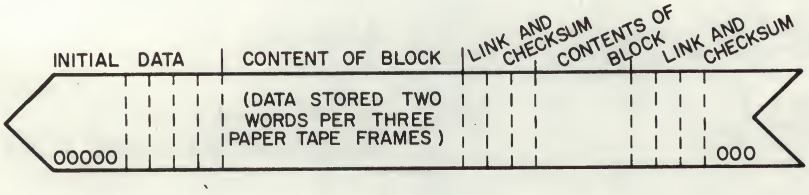
Input Data

Input data is read and translated into core form. If the input is from disk, no translation is necessary; however, off punched tape, translation is necessary.

Core Data

Data is stored 200_8 positions at a time in core in the same fashion it would be stored if it were on disk.

Output Data



EXECUTION TIME

The machine takes much less than a quarter of a second to read or write a block on/off disk (on an 8/S). A block of tape consists of 303_8 frames and is subject only to the speed of the input or output device for speed of transfer.