ERLANGEN
9. MRZ 1972

DECUS
PROGRAM LIBRARY

DECUS NO.	8-187
TITLE	KEYBOARD CONTROLLED BINARY PUNCH
AUTHOR	Edward A. Taft, III
COMPANY	Saint Mark's School Southborough, Massachusetts
DATE	April 7, 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.

KEYBOARD CONTROLLED BINARY PUNCH

DECUS Program Library Write-up DECUS No. 8-187

INTRODUCTION

The Keyboard Controlled Binary Punch makes binary tapes of selected areas of core. Its main features are as follows:

1. Keyboard controlled making all operations specified by typed commands.

2. Several blocks of data from separated areas of core can be punched on one tape.

3. Punches field marks for extended memory programming.

4. Occupies only one page of core.

5. Easy to use and especially good for FORTRAN programming. For example, an operating program may be punched onto tape (complete with library routines, operating system, and monitor). Often-used programs saved in this manner may be later run simply by loading the tape with the BIN loader and running them, with no need to fuss with special FORTRAN loaders.

REQUIREMENTS

This program requires a PDP-5, 8, 8/I or 8/S with an ASR-33 and high-speed punch. Extended memory is optional, and greatly increases the convenience of saving large programs.

OPERATION

There are three binary versions of this program available. Version A (LOW) occupies page 1, with a starting address of 0200. Version B (HIGH) occupies page 36, with a starting address of 7400. Version C (SPECIAL) occupies page 37, with a starting address of 7600.

The ASCII version may be reassembled to load into any page by setting SA=the first location on that page.

Load and start the program. It will operate in any field and respond with a CR/LF. Turn on the high-speed punch.

The following commands are now available:

1. L (leader)
   Typing the letter L will cause about six inches of BIN leader (code 2ØØ) to be punched. The program will then respond auto-

matically with a CR/LF. L also clears the checksum in preparation for punching a program.

2. F n (field)
Typing F, a field number (Ø-7), and a CR will cause a field mark of the form 3nØ to be punched on the tape. All data thereafter will be taken from the specified field.

3. P mmmm-nnnn (punch)
Typing P, two octal core limits separated by a dash, and CR will cause the specified material to be punched in binary. The data will be taken from the field most recently specified by F, or, if no field has been given, by the field specified by the DATA FIELD register when the program was started.

4. C (checksum)
Typing C will cause the accumulated checksum to be punched, followed by six inches of leader code 2ØØ. The checksum will then be cleared. The program will respond with CR/LF.

All spaces are ignored on input. Any character besides L, F, P, or C typed at the beginning of a line (including CR) will be ignored. No punching will take place, and the program will restart, responding with a CR/LF. Pressing rubout while typing a line will also cause a restart, and the line will be ignored.

No field mark is punched unless an F command is given. Data to be punched is taken from the field indicated by the DATA FIELD register. If it is desired to change the field without punching a field mark, restart the program and set the DATA FIELD register to the desired field.

When punching a tape designed to be loaded into any field, field marks should not be punched, except when punching a tape which must later be loaded into a particular field (or sections of which must be loaded into different fields.

The checksum must be punched immediately after the data. The leader punched after the checksum will cause the BIN loader to compare this checksum to that computed while the tape has been read in.

Enough loader (L) should be punched at the beginning of the tape to make handling easy.

For further information on BIN format, refer to the DEC documentation for the Binary Loader.

Some suggestions for different systems:

1. Without Extented Memory - A free page must be found in which to place the punch program. The SPECIAL version cannot be used for there is no way to load it over the loader. A space between the program and the data is found by running SYMBOLPRINT and can be used to locate the punch routine when saving FORTRAN.

2. Extended Memory - If the SPECIAL version is used, it may be placed in the last page of the field which does not contain the RIM and BIN loaders.

3. 8K Memory and Disk/DECtape Monitor - The monitor head occupies the last page of field Ø, and the RIM and BIN loaders are usually placed in field 1. To save the RIM and BIN loaders on the systems device, the following is used: .SAVE BINL ! 176ØØ;

The SPECIAL version may then be loaded into field 1 and used (this may also be saved). To restore the RIM and BIN loaders, call .BINL from the monitor.

4. 8K FORTRAN - After loading the operational program, punch the contents of each field up to and including the page determined by subtracting the number of free pages in that field (printed by the symbol table routine in the loader) from 368. Field marks should be punched on the tape at the beginning of each field.

ERRORS

There is no error detection system. The user should note the following:

1. If the first (non-space; spaces are ignored) character is not L, F, P, or C, it is ignored and the program restarts.

2. Numbers are processed up to the first non-digit. No attempt is made to check for overflow, therefore, a field number not in the range Ø-7 may cause severe errors.

3. It is important that the punching strictly follow BIN format, that is, leader must precede data. Several blocks of data may be punched consecutively, but there must be no leader between them. Finally, the checksum must be punched, followed by leader in any amount. The entire sequence may be repeated several times on the same tape, in which case the BIN loader will stop at the end of each section for checksum comparison. The program does not check for illegal sequences.

If a mistake is made, punch leader to clear the checksum and start over.

A typing error may be corrected by pressing rubout and retyping the line if the command has not yet been executed.

```
/KEYBOARD CONTROLLED BINARY PUNCH
/
/BY     EDWARD A. TAFT, III
/       SAINT MARK'S SCHOOL
/       SOUTHBOROUGH, MASS.  Ø1772
/
/FOR 8-FAMILY COMPUTERS, WITH ASR-33 AND HIGH SPEED PUNCH
/
/OCCUPIES ONE PAGE
SA=76ØØ
/
/
```

```
          *SA
7600 1341 START, TAD C215       /PRINT CR
7601 4370       JMS COUT
7602 1342       TAD M3          /LF
7603 4370       JMS COUT
7604 4350       JMS CIN         /GET 1ST CHARACTER
7605 1343       TAD CLEAD       /TEST FOR L,C,F,P
7606 7450       SNA
7607 5264       JMP LEADER      /L=PUNCH 6 INCHES LEADER
7610 1344       TAD CSUM
7611 7450       SNA
7612 5261       JMP CHKSUM      /C=PUNCH CHECKSUM AND LEADER
7613 1342       TAD M3
7614 7450       SNA
7615 5242       JMP FLD         /F=NEW FIELD SETTING
7616 1345       TAD CPUN
7617 7640       SZA CLA
7620 5200       JMP START       /NONE OF THOSE, IGNORE
7621 4317       JMS OCIN        /P=PUNCH BINARY DATA
7622 3251       DCA POINT       /LOWER LIMIT
7623 4317       JMS OCIN        /GET UPPER LIMIT
7624 7140       CLL CMA
7625 1251       TAD POINT
7626 3370       DCA COUT        /SET COUNTER
7627 1251       TAD POINT
7630 7120       STL
7631 4303       JMS WOUT        /PUNCH ORIGIN SETTING
7632 7100       CLL
7633 1651       TAD I POINT     /PUNCH THE DATA
7634 4303       JMS WOUT
7635 2251       ISZ POINT
7636 7700 C7700, SMA CLA        /DOES NOTHING HERE
7637 2370       ISZ COUT
7640 5232       JMP .-6
7641 5200       JMP START
7642 4317 FLD,  JMS OCIN        /NEW FIELD, GET FIELD #
7643 7104       CLL RAL         /MUST BE 0-7
7644 7006       RTL
7645 3303       DCA OCNUM
7646 1303       TAD OCNUM       /CHANGE DATA FIELD
7647 1346       TAD XCDF
7650 3251       DCA .+1
7651 6201 POINT, CDF
7652 1347       TAD CHECKS      /SAVE CHECKSUM
7653 3251       DCA POINT       /FIELD SETTING NOT A PART OF IT
7654 1303       TAD OCNUM       /PUNCH FIELD SETTING
7655 1236       TAD C7700
7656 4274       JMS POUT
7657 1251       TAD POINT       /RESTORE CHECKSUM
```

```
7660 5272           JMP LEADER+6
7661 1347 CHKSUM,   TAD CHECKS          /PUNCH CHECKSUM
7662 7100           CLL
7663 4303           JMS WOUT
7664 1236 LEADER,   TAD C7700           /PUNCH 6 INCHES OF LEADER
7665 3370           DCA COUT
7666 1336           TAD C200
7667 4274           JMS POUT
7670 2370           ISZ COUT
7671 5266           JMP.-3
7672 3347           DCA CHECKS          /ZERO OUT CHECKSUM
7673 5200           JMP START
                    /PUNCH OUTPUT ROUTINE
7674 0000 POUT,     0                   /PUNCH CHARACTER IN AC
7675 6026           PLS
7676 6021           PSF
7677 5276           JMP .-1
7700 1347           TAD CHECKS          /ADD TO CHECKSUM
7701 3347           DCA CHECKS
7702 5674           JMP I POUT
                    OCNUM=.
                    /WORD PUNCH (2 CHARACTERS)
7703 0000 WOUT,     0
7704 3350           DCA CIN
7705 1350           TAD CIN             /GET LEFT HALF
7706 0236           AND C7700
7707 7012           RTR
7710 7012           RTR
7711 7012           RTR
7712 4274           JMS POUT
7713 1350           TAD CIN             /GET RIGHT HALF
7714 0375           AND C77
7715 4274           JMS POUT
7716 5703           JMP I WOUT
                    /INPUT AN OCTAL NUMBER
7717 0000 OCIN,     0
7720 3303           DCA OCNUM           /INITIALIZE TO 0
7721 4350           JMS CIN             /GET A CHARACTER
7722 1376           TAD M270            /SEE IF DIGIT
7723 7500           SMA
7724 5336           JMP END             /NO
7725 1377           TAD C10
7726 7510           SPA
7727 5336           JMP END             /NO
7730 3350           DCA CIN             /YES, SAVE
7731 1303           TAD OCNUM           /ADD NEW DIGIT
7732 7106           CLL RTL
7733 7004           RAL
7734 1350           TAD CIN
7735 5320           JMP OCIN+1
```

```
          C2ØØ=.
7736 72ØØ END,    CLA
7737 13Ø3         TAD OCNUM
774Ø 5717         JMP I OCIN
7741 Ø215 C215,   215
7742 7775 M3,     -3       /FOR LF AND "F"
7743 7464 CLEAD,  -314     /FOR "L"
7744 ØØ11 CSUM,   314-3Ø3  /FOR "C"
7745 7766 CPUN,   3Ø6-32Ø  /FOR "P"
7746 62Ø1 XCDF,   CDF
7747 ØØØØ CHECKS, Ø        /CHECKSUM
          /INPUT A CHARACTER FROM THE TELETYPE
775Ø ØØØØ CIN,    Ø
7751 6Ø31         KSF
7752 5351         JMP .-1
7753 6Ø36         KRB
7754 437Ø         JMS COUT         /ECHO
7755 1365         TAD M377
7756 745Ø         SNA
7757 52ØØ         JMP START        /RUBOUT, RESTART
776Ø 1366         TAD TSPC
7761 745Ø         SNA
7762 5351         JMP CIN+1        /IGNORE SPACES
7763 1367         TAD C24Ø
7764 575Ø         JMP I CIN
7765 74Ø1 M377,   -377
7766 Ø137 TSPC,   377-24Ø
7767 Ø24Ø C24Ø,   24Ø
          /OUTPUT A CHARACTER TO THE TELETYPE
777Ø ØØØØ COUT,   Ø
7771 6Ø46         TLS
7772 6Ø41         TSF
7773 5372         JMP .-1
7774 577Ø         JMP I COUT
          C3ØØ=C77ØØ
7775 ØØ77 C77,    77
7776 751Ø M27Ø,   -27Ø     /DIGIT TESTS
7777 ØØ1Ø C1Ø,    1Ø
```

```
CHECKS  7747
CHKSUM  7661
CIN     775Ø
CLEAD   7743
COUT    777Ø
CPUN    7745
CSUM    7744
C1Ø     7777
C2ØØ    7736
C215    7741
C24Ø    7767
C3ØØ    7636
C77     7775
C77ØØ   7636
END     7736
FLD     7642
LEADER  7664
M27Ø    7776
M3      7742
M377    7765
OCIN    7717
OCNUM   77Ø3
POINT   7651
POUT    7674
SA      76ØØ
START   76ØØ
TSPC    7766
WOUT    77Ø3
XCDF    7746
```

/