DECUS
PROGRAM LIBRARY

DECUS NO.	8-513
TITLE	DEBUG 8
AUTHOR	Michael S. Cole and C. W. Richardson Submitted by: W. R. Myers
COMPANY	Aerojet Nuclear Company Idaho Falls, Idaho
DATE	September 23, 1971
SOURCE LANGUAGE	PAL-III

DECUS Program Library Write-up DECUS NO. 8-513

DESCRIPTION OF PROGRAM AND HOW TO USE IT

The program is entered with a "JMS" to ØIØA - upon entering the program a pointer (ØIPT) is set to the contents of the AC, this pointer points to the location to be examined by the debug routine. Another location (ØIØAC) is set to zero, this location holds numbers input to the debug routine. Next the program executes a "JMS I" to a user supplied routine for a carriage-return line-feed (assumed loc. 66 in the listing). Next ØIPT is placed in the AC and then the AC is printed on the teletype (octal nos.) a space is printed then the contents of the location pointed to by ØIPT are printed (octal nos.). Next the program goes to a "READ" routine (supplied by the user and pointed to by location 3 in the listing) which will return to the following location with the ASCII code of the character entered from the keyboard in the AC (and in location 125 in the listing - TS1). The program now checks the input character to see if it is one of the following:

1) C.R. - if it is a C.R. the program exits to the call + 1.

2) Space - a space opens the next location showing the location and its contents.

3) L - an L will load the octal numbers typed in into the last opened location.

4) N - N takes the numbers typed in and uses them as a pointer to the next location to open.

5) G - takes the octal number typed in an begins execution there. (A slight modification would allow placing a value in the AC and Link before beginning execution.)

6) B - Inserts a break-point instruction in the location pointed to by the numbers typed in (the break point instruction is a JMS I 127 in the listing).

7) R - Restores the last breakpoint instruction to what it was originally.

8) S - Executes a "JMS" to the address just typed in.

9) V - Uses the current contents as a pointer for the next location to be opened.

10) C - Uses the current contents as a pointer to a floating-point number. (The user must provide a routine [INFØRA] to look at floating-point numbers - an example is shown in the listing.)

11) F - Uses the number that has been entered as a pointer to a floating-point number. (See C above.)

If the entered character was not one of the above the ASCII code is "ANDed" with an octal 7 and the number is added to a binary accumulator that has been rotated 3 places to the left (no carry) and the routine returns to read another character and check it as above.

If the switch register is "0" locations and contents are printed (with CRLF) until the switch register is non-zero.

```
          *3
          / THIS SHOULD POINT TO A USER INPUT ROUTINE
0003  7402  READ,   HLT
          / THIS SHOULD POINT TO A USER OUTPUT ROUTINE
0004  7402  PRINT,  HLT
          *66
          / THIS SHOULD POINT TO A USER CARRAGE RETURN L.F. ROUTINE
0066  7402  CRLF,   HLT
          *77
0077  0654  INFORA, INFOR
          *125
0125  0000  TS1,    0        /HOLDS INPUT FROM READ ROUTINE
          *127
0127  2715  ADDBRT, DBRT
          *654
          OUTPUT=14
          INPUT=13
          /ROUTINE TO UPDATE CONSTANTS
          /ENTER WITH ADDRESS OF CONSTANT IN THE AC
0654  0000  INFOR,  0
0655  3304          DCA     XAD       / SAVE ADDRESS OF FLOATING NUM
0656  4407          JMS I   7
0657  5704          FGET I  XAD       / GET FLOATING NUMBER
0660  0014          OUTPUT            / LOOK AT IT
0661  0000          FEXT
0662  1303          TAD     P273      / FOR ";"
0663  4404          JMS I   PRINT
0664  4407          JMS I   7
0665  0013          INPUT             / WANT TO CHANGE IT?
0666  6305          FPUT    TEMP
0667  0000          FEXT
0670  1060          TAD     60        / ANY INPUT?
0671  7640          SZA CLA
0672  5275          JMP     .+3       / NO
0673  3062          DCA     62        / "E" FORMAT
0674  5654          JMP I   INFOR     / NO - GO BACK
0675  4407          JMS I   7         / YES - REPLACE IT
0676  5305          FGET    TEMP
0677  6704          FPUT I  XAD
0700  0000          FEXT
0701  3062          DCA     62
0702  5654          JMP I   INFOR     / AND GO BACK
0703  0273  P273,   273               / ";"
0704  0000  XAD,    0                 / STORAGE FOR ADDRESS OF FLOATIN
0705  0000  TEMP,   0                 / STORAGE FOR INPUT -
0706  0000          0                 / IF ANY.
0707  0000          0
          *2600
2600  0000  OIOA,   0                 / OCTAL INPUT/OUTPUT
2601  3355          DCA     OIPT
2602  3356          DCA     OIOAC
2603  4466          JMS I   CRLF
2604  1355          TAD     OIPT
2605  4331          JMS     OOUT
2606  1755          TAD I   OIPT
2607  4331          JMS     OOUT
2610  1375          TAD     SWITCH
2611  7650          SNA CLA
2612  5261          JMP     OILF
```

```
2613  4403  OIO1,   JMS I   READ      / GET CHARACTER
2614  0371          AND     P0007     / CONVERT TO OCTAL
2615  3357          DCA     OCH
2616  4360          JMS     MATCH     / FIND OUT WHAT IS WANTED
2617  7563          -215
2620  5600          JMP I   OIOA      / C.R.
2621  7540          -240
2622  5261          JMP     OILF      / SPACE BAR
2623  7464          -314
2624  5257          JMP     OILO      / L FOR LOAD
2625  7462          -316
2626  5267          JMP     OINX      / N FOR NEXT
2627  7456          -322
2630  5323          JMP     OIDBRS    / R FOR RESTORE BREAK POINT
2631  7471          -307
2632  5271          JMP     OIGO      / G FOR GO
2633  7476          -302
2634  5303          JMP     OIDB      / B FOR INSERT BREAK POINT
2635  7455          -323
2636  5273          JMP     OIOJMS    / S FOR JMP TO SUBROUTINE
2637  7472          -306
2640  5300          JMP     OIOFLK    / F FOR LOOK AT FLOATING POIN
 2641  7452         -326
2642  5255          JMP     OIVECT    / V FOR VECTOR
2643  7475          -303
2644  5276          JMP     OIFVEC    / C FLOATING VECTOR
2645  0000          0
2646  1356          TAD     OIOAC
2647  7104          CLL RAL
2650  7104          CLL RAL
2651  7104          CLL RAL
2652  1357          TAD     OCH
2653  3356          DCA     OIOAC
2654  5213          JMP     OIO1
2655  1755  OIVECT, TAD I   OIPT      / GET ADDRESS
2656  5201          JMP     OIOA+1
2657  1356  OILO,   TAD     OIOAC     / STORE IN CORE
2660  3755          DCA I   OIPT
2661  2355  OILF,   ISZ     OIPT      / MOVE TO NEXT WORD
2662  3375          DCA     SWITCH
2663  7404          OSR
2664  7640          SZA CLA
2665  2375          ISZ     SWITCH
2666  5202          JMP     OIOA+2
2667  1356  OINX,   TAD     OIOAC     / RESET POINTER
2670  5201          JMP     OIOA+1    / GO LOOK
2671  4466  OIGO,   JMS I   CRLF      / CARRIAGE RETURN LINE FEEC
2672  5756          JMP I   OIOAC     / GO START AT OIOAC ADD.
2673  4466  OIOJMS, JMS I   CRLF      / POSITION CARRIAGE
2674  4756          JMS I   OIOAC     / SOCK IT TO 'EM
2675  4315          JMS     DBRT      / LOOK AT THE GOOD STUFF
2676  1755  OIFVEC, TAD I   OIPT      / GET POINTER
2677  7410          SKP
2700  1356  OIOFLK, TAD     OIOAC
2701  4477          JMS I   INFORA
2702  5267          JMP     OINX
```

```
2703  1756  OIDB,   TAD I  OIOAC
2704  3327          DCA      OIDBSV     / SAVE COMMAND
2705  1356          TAD      OIOAC
2706  3330          DCA      OIDBAD     / AND ADDRESS
2707  1326          TAD      JMSDB
2710  3756          DCA I    OIOAC      / INSERT BREAK POINT INSTRUCTS
2711  5261          JMP      OILF       / AND GO LOOK AT NEXT
2712  2713  AOIOACS,        OIOACS
2713  0000  OIOACS, 0                  / STORAGE FOR AC
2714  0000  OIOLS,  0                  / AND LINK
2715  0000  DBRT,   0
2716  3313          DCA      OIOACS     / SAVE AC
2717  7004          RAL
2720  3314          DCA      OIOLS      / AND LINK
2721  1312          TAD      AOIOACS
2722  5201          JMP      OIOA+1     / GO LOOK AT IT
2723  1327  OIDBRS, TAD      OIDBSV     / PICK UP COMMAND
2724  3730          DCA I    OIDBAD     / RESTORE IT
2725  5261          JMP      OILF       / LOOK AT NEXT
2726  4527  JMSDB,  JMS I    ADDBRT
2727  0000  OIDBSV, 0
2730  0000  OIDBAD, 0
2731  0000  OOUT,   0
2732  3354          DCA      OOT
2733  1372          TAD      M4         / COUNT 4 DIGITS
2734  3353          DCA      OOC
2735  1354          TAD      OOT
2736  7004          RAL                 / ISOLATE ONE OCTAL DIGIT
2737  7006          RTL
2740  3354          DCA      OOT
2741  1354          TAD      OOT
2742  7004          RAL
2743  0371          AND      P0007
2744  1374          TAD      P260       / CONVERT TO ASCII
2745  4404          JMS I    PRINT
2746  2353          ISZ      OOC        / FINISHED ?
2747  5335          JMP      .-12       / NO
2750  1373          TAD      K240
2751  4404          JMS I    PRINT
2752  5731          JMP I    OOUT       / YES
2753  0000  OOC,    0                  / COUNTER
2754  0000  OOT,    0                  / TEMP
2755  0000  OIPT,   0                  / OCTAL POINTER
2756  0000  OIOAC,  0                  / OCTAL ACCUMULATOR
2757  0000  OCH,    0                  / OCTAL CHARACTER
2760  0000  MATCH,  0
2761  1760          TAD I MATCH        / GET NUMBER
2762  2360          ISZ      MATCH       / MOVE POINTER
2763  7450          SNA                 / FINISHED ?
2764  5760          JMP I    MATCH      / YES
2765  1125          TAD      TS1        / MATCH?
2766  7650          SNA CLA
2767  5760          JMP I    MATCH      / YES - GO TO IT
2770  5361          JMP      MATCH+1    / NO - TRY AGAIN
2771  0007  P0007,  7
2772  7774  M4,     -4
2773  0240  K240,   240
2774  0260  P260,   260
2775  0001  SWITCH, 1

ERROR COUNT = 0000
```

```
ADDBRT  0127
AOIOAC  2712
CRLF    0066
DBRT    2715
INFOR   0654
INFORA  0077
INPUT   0013
JMSDB   2726
X240    2773
MATCH   2760
M4      2772
OCH     2757
OIDB    2703
OIDBAD  2730
OIDBRS  2723
OIDBSV  2727
OIFVEC  2676
OIGO    2671
OILF    2661
OILO    2657
OINX    2667
OIOA    2600
OIOAC   2756
OIOACS  2713
OIOFLK  2700
OIOJMS  2673
OIOLS   2714
OIOI    2613
OIPT    2755
OIVECT  2655
OOC     2753
OOT     2754
OOUT    2731
OUTPUT  0014
PRINT   0004
P0007   2771
P260    2774
P273    0703
READ    0003
SWITCH  2775
TEMP    0705
TS1     0125
XAD     0704
```