COMPUTATION CENTER
January 21, 1964

FROM: Louis Pouzin

SUBJECT: Miscellaneous Programming Tools

This paper depicts some tools developed for personal use which could be made available now through common file 1. They do not intend to be of general design and usage. They are as they are, and result from the particular needs for which they have been implemented. Some of them do not work properly when arguments do not match assumed specifications. They also happen to be modified whenever it is desired.

* * *

1. Commands, started by RESUME

PRINTF ALFA BETA N. Prints in BCD form the file ALFA BETA. Words 13 and 14 of every line are printed on the left, as they are supposed to be the serialization field. Then the first N words of every line are printed in alphabetic. Accepts variable length records.

PRBIN ALFA BETA N. Prints file ALFA BETA in octal BCD form. Printing is done by groups of N words, followed by a carriage return. Whenever N is greater than 6, the program prints as many 6 word lines to reach the number N.
N may not be greater than 28.
Words are separated by one space.

PRBSS ALFA. Prints a summary of all BSS routines which are supposed to be contained in file ALFA BSS.
1st line Entry names, and relative address.
2nd line Common Length, program length, transfer vector leng
3rd line (if any) Names in transfer vector.

EXTBSS ALFA BETA. Extracts from a file ALFA BSS (a library) the first BSS routine with an entry name BETA, and creates a file BETA BSS. Previous BETA BSS is deleted if possible.

UPDBSS ALFA BETA. Searches file ALFA BSS, and replaces the first routine with an entry name BETA by the file BETA BSS. A new version of ALFA BSS then replaces the old one.

PRLIN ALFA BETA SYMB.
Prints the line number of the line which contains the symbol SYMB anywhere in the 6 first characters, in file ALFA BETA.

PERMUT ALFA BETA N1 N2 N3.
Takes the piece of the file ALFA BETA, going from after line N1 through (including) line N2, and moves this piece after line N3. N3 may not be between N1 and N2.

N3 > N2 or N3 < N1

old ALFA BETA is deleted.

INSERT ALFA BETA GAMA N.
Inserts the file ALFA BETA into the file GAMA BETA, after line N. Old GAMA is deleted.

DØLØD ALFA BETA GAMA.
Simulates the loading of programs ALFA BSS, BETA BSS, etc.. and creates a file (LØAD FILE) with the same characteristics as the file created by the modular system loader.

* * * * * *

2. Function and Subroutines

All entry names listed in the same paragraph belong to the same routine. When usable as functions, the result is in accumulator (S, 1-35), in order to fit Fortran and Mad conventions.

1. ADD(X) returns address field of X
   DEC(X) " decrement " "

2. BCDEC(X) converts the unsigned BCD number X into a binary integer right-justified.

3. BCØCT(X) converts the unsigned BCD octal number X into a binary integer right-justified.

4. CØMARG returns a specified argument from the command buffer usabe as following:

   EXECUTE CØMARG.(3,A)
   put 3rd argumrnt in A

   A = CØMARG.(3) same result.

   A = CØMARG.(3,B) both A and B set to argument 3.

Ex. of typical use.

```
AA      THRØUGH AA, FØR A=1,1,A.E.19.ØR.A(A).E.$$.ØR
          1 CØMARG.(A,A(A)).E.FENCE
```

A(1)...A(A-1) will be the significant arguments.

5. DA(A,B) returns A.ANB.B
   DXV(A,B) returns A.EXØR.B
   DV(A,B) returns A.ØR.B

6. GCLC(A,B) A= command location counter
   B= number of last command to be executed.

   return in AC:A in address field, B in decrement field.

   SCLC(A,B) A= command location counter setting
   B= number of last command to be executed.

   GCLS(A,B) A(0), A(1)...A(N) $N \leq 19$

   Vector A is set to the content of the Bth command buffer (including fence).

   SCLS(A,B) Sets Bth command buffer with content of vector A(0)...A(N) until A(19) or fence.

N.B. calls another S/R: MØVE1, MØVE2, MØVE3

7. LJUST(A) returns A left justified with trailing blanks. Only leading blanks (not leading zeros) are dropped out.

8. LS(A,N) returns A shifted N bits to the left.
   RS(A,N) returns A shifted N bits to the right.

9. MNEM(A) returns BCD mnemonic operation code of the binary word A.

10. ØCABC(A) returns BCD octal value of the address field of A.
    ØCDBC(A) returns BCD octal value of decrement field of A.
    ØCRBC(A) returns BCD octal value of right half of A.
    ØCLBC(A) " " " " " left " "

11. PRNTP(A) prints A
    where VECTØR VALUES A= hollerith string of any length, 777777777777K

12. SID(A,B) stores address of A in decrement of B. Other fields are not modified.
    STA(A,B) stores address of A in address of B.
    STD(A,B) stores decrement of A in decrement of B.
    STL(A) stores location counter (from calling program) in A.
    STP(A,B) stores prefix of A in prefix of B.
    STT(A,B) stores tag of A, in tag of B.

13. TØCT(A) returns 1 if A is BCD octal (blanks or digits 0-7)
0 if not.
also usable as: EXECUTE TØCT.(A,B) result in B
or C = TØCT.(A,B) result in B and C

14. MØVE1,MØVE2,MØVE3
Utility routine needed by SCLS,GCLS.

C.A.S.