March Number 15 1976

Contributions and correspondence should be sent to:

> Bob Hassinger, Coordinator
> OS/8 Special Interest Group
> c/c DECUS
> 146 Main Street
> Maynard, Massachusetts 01754

USER INPUT TO DEC SOFTWARE PLANNING

One of the most positive outcomes of the Fall DECUS Symposium in Los Angeles was an expanding interest on the part of DEC in user inputs to their software development planning. The managers of software development expressed a desire to hear from the users through the special interest groups. As a first step in that direction I am including a survey we hope will influence decisions DEC is making on what software development for the PDP-8 they should pursue in the next couple of years.

It is important for us to get a strong, representative response to this initiative from a broad cross section of the user community. If we get a sufficient response we will be able to discuss the questions and answers with DEC management at the Spring Symposium in Atlanta.

The survey is based on the "hypothetical" solution that DEC has available the resources to pursue one major set of software development projects for the PDP-8 family over the next two to three years.

One possible set of projects involves Operating System Development either along the lines of an upgraded "OS/8 type" of single user program development system which removes existing architectural limitations or else along the lines of an expanded "RTS-8 type" system for real-time, multi-tasking with support of single user program development (i.e., system CUSPs would be Tasks, etc.).

The second possible set of projects would involve Language Development. One possible line for this to take would be a FORTRAN IV that would run under RTS-8. A second possibility would be a BASIC implemented as an incremental compiler/interpreter which might possibly be able to use the floating point processors.

The attached form attemps to give you a chance to express your opinion on which major line of development to pursue and within it which way the work should go.

The hypothetical situation is probably fairly close to what is really going to happen and it does not allow for "one of each". You really have to face the real world situation of limited resources and decide what is most important to you.

EUROPEAN OS/8 SPECIAL USER GROUP

I recently received information from Lars Palmer about the Fall European DECUS Symposium. At that time consistent with the new DECUS Bylaws an OS/8 Special Interest Group for Europe was formed. An announcement of the formation was included along with a report of what went on at the meeting. Also Lars has sent along some additional information on MULTI-8. All of these items will be attached to the Newsletter. We can expect the DECUS Europe part of the OS/8 community to be forwarding material for this Newsletter which we plan to continue as one international publication for the foreseeable future.

"12-BIT SIG" vs. "OS/8 SIG"

Ernst Lopes Cardozo has written that he thinks the common opinion at the European DECUS Symposium in September was in favor of a general 12-bit SIG as opposed to one limited to OS/8. He thinks that with the new hardware (PDP-8A, etc.) and software trends (multi-user system like ETOS and MULTI-8) a wider formal scope may cover our members' range of interests better. If there is no significant objection between now and the Spring DECUS/US Symposium I think we will probably move in this direction. The concept of a "Mainframe" group as exemplified by the DEC system-10 group seems to be the most logical way to meet the 12-bit users' full range of needs for communication and cooperation in the future.

How about ideas for a "catchy" name that expresses the expanded area of interest?

NEWSLETTER DEADLINES

If the turn-around time and frequency of publication of this Newsletter are going to be improved I think I will have to establish a publication schedule with a deadline. Anything received "camera ready" by the deadline will make it into that issue. Anything not "camera ready" that I can get re-done by the deadline will also make it into that issue. Anything not making it under the deadline will be held for the next issue. To start I want to try an every-other-month schedule until further notice. Therefore, the deadlines will be the last Friday of the even numbered months (i.e., April 29, 1976, June 25, 1976, etc.)

MACREL UPDATE

The indication from DEC at the Fall Symposium was that work would begin again in earnest on MACREL (the OS/8 MACRO/Relocating assembler and linking loader for you new comers) as soon as everyone got back to Maynard. Since then, however, the DEC NET/8 development has taken over some of the MACREL personnel so progress is still awaiting the availability of personnel. Incidentally, I noticed an ad for PDP-8 System software people in the Sunday paper. DEC has been looking for one or two capable people to beef up the staff in our area. This reflects the increased software effort they promised in Los Angeles.

SOFTWARE SUPPORT

There were a lot of people at the Fall Symposium who were unhappy with the support OS/8 software has been receiving. We were by no means alone in our complaint, incidentally. Most of the other machine/operating system areas were saying similar things.

Bob Bean came ready to meet the OS/8 complaints. He told us that since he has taken charge of the area that includes OS/8 development he has been making a major effort to reduce the SPR backlog. He wants to answer all SPRs in 30 days or less. This has not been achieved as yet, but he says a lot of progress has been made.

To improve the quality of the answers Bob is personally reviewing each one. Until the situation is completely under control he invited users to write him directly if they are dissatisfied with the quality of an answer. This is in contrast to the normal indirect route through Software Communications. The traditional route will again be the rule after a few months when the situation is stabilized. Bob is located in Maynard in Single Users Systems, ML 5-5/E76.

In other developments a presentation was given on DEC's companywide re-evaluation of the way they handle software support. The "gist" of the session was that they now will state how long they will warranty software and under what categories of support. (The cynics saw this as an effort to limit a previously open ended liability.) After the warranty period some sort of charge for support is anticipated (to help pay for support, which is costing much more than previously anticipated, perhaps). This new policy does not presently have much impact on existing ("old") products but watch out for it to be phased in with new announcements. The main thrust seems to be towards the bigger, more expensive systems but we are bound to be affected eventually.

MATERIAL FROM STANLEY RABINOWITZ

Stan has forwarded a number of items since the last Newsletter.

1. Running BATCH in 32K. When BATCH runs on a 32K machine only 28K is made available to user programs. Users who have 32K and have no TD8E ROM may wish to install the following optional patch which allows programs to access all 32K while BATCH is running. Change location 711 in BATCH from 1370 to 7200.

2. Corrections to previously published information. The item published in the October Digital Software News on PAL8 V9G /W option failing is Sequence 10 not 8. You should add a change to location 1532 from 0307 to 0310 to change the version number to V9H. Also in the same issue of DSN, also applying to PAL8 V9G, regarding switching between handlers. The sequence number for this is 11 not 9. There will be a replacement for this item which is now sequence 11. The problem is that PAL8 blows up when switching between non-resident handlers. The following patch corrects this problem and changes the version number to V9I. Change location 5334 from 1176 to 7200 and 1532 from 0310 to 0311.

3. An OS/8 quiz and puzzle that I will attach.

4. Some advance copies of articles for the Digital Software News that I will condense and attach.

SPR's FORWARDED BY LARS PALMER

1. This is an explanation of the forward reference problem in RALF which causes problems that are undocumented. Problems arise because RALF needs to know during pass 1 how many words each statement will assemble into so that it can define symbols. Since it doesn't know whether an FPP direct reference instruction containing a forward reference should assemble as one word or two it plays safe by assigning it two words and flagging the symbol causing the forward reference so that during pass 2 all direct reference instructions containing this symbol are assembled as two word instructions. The way around this is to explicitly indicate how many words are needed for each direct instruction containing a forward reference to a base page variable by following the OPCODE by a single quote character (') for a one word instruction or a number sign (#) for a two word instruction. The only mention of single quote appears in the Fortran IV Users' Manual page 8-2 which has been superseded by the OS/8 Handbook. The number sign is given a cursory mention at the bottom of page 5-22 in the OS/8 Handbook as well as in the same table as is single quote but none of these explain where the symbols need to be used.

2. The OS/8 EDIT V9 rub out algorithy does not always work correctly since text is stored as 6 bit characters with 77 being used as an escape code and question mark (?) being stored as 77 77. When the rub out algorithm meets something like ?A stored as 77 77 01 it will treat the second 77 as an escape code for the 01 and deletes both of them corrupting the line. The solution that the submitter A. Windrom suggests is to ensure that 77 77 never occurs by re-coding question mark as 77 00 because the 200 code is presently ignored by EDIT so that 00 should not be a code that is stored.

3. Mr. Windrom also submitted a report on several Fortran IV bugs in Version 3.04. Because some of them appear to have been corrected in the new Version to be released with OS/8 Version 3C. I will not try to include them in this Newsletter.

NOTE FROM JIM CORYELL

Jim says that in TECO Version 3 if location 5126 is changed from 0011 to 1637 than you have a "V" command which is like the T command except that output goes to the output file rather than the terminal. He says this was once done with a "W" command. Jim also sends a copy of an SPR that he submitted in July regarding a problem with OS/8 BASIC where SAVE programs destroy BATCH. The problem seems to be that an effort is made to restore BATCH with some words when they were saved by BRTS or BLOAD. If this problem has not been solved by a published fix and you need Jim's patch let me know.

Jim also notes that in the December Digital Software News that in article BRTS #19 an instruction is shown as 5053 when it should be 3053. Have you

noticed the most consistent thing about DEC is the error rate in their printed matter. Since at least 1968 it seems to have stayed constant at about "one critical error per page".

ADDRESS FOR JOHN ALGEO

Recently I mentioned that John Algeo had sent me information on some random access routines for Fortran II. At that time I did not have an address for him. Since then I have received the following address: Santa Ynez Research Farm, PO Box 688, Santa Ynez, CA 93460.

MORE ON BOB PHELPS' USR ROUTINE AND OTHERS

I talked to Bob a few days ago. He is getting close to submitting his USR routine to DECUS; however, he wants to make some improvements first. In the meantime, if you need to reach him the address I have is University of Rochester Medical Center, Department of Radiation Biology and Biophysics, Rochester, NY 14642.

In addition to his USR routine which was mentioned last time and which I have been using with great success, the other programs he forwarded are as follows:

MODE 8 - Allows PDP 8 mode subroutines to be easily called from Fortran programs. This routine allows your subroutine to be written strictly as a PDP 8 subroutine without having to go through all of the overhead of providing a RALF coded interface. This minimizes the space required for such linkage for each PDP 8 mode routine you use.

RLINK - This allows a single LINCtape block to be read into a 256 variable array. for LINCtape this routine will provide the ability to access data stored in the usual 12 bit way with PDP 12 programs.

IOT - This routine allows PDP 8 mode instructions (IOT's in particular) to be issued with a specified value in the accumulator. The new value is returned. This could be used with an LAS instruction to return the value of the switch register.

TV - This routine allows ASCII formatted information to be displayed on the PDP 12 scope in background mode while executing Fortran programs. The routine is designed for more rapid interaction with the user than is possible using the teletype as an output device. The FORTRAN function backspace allows over-writing a line and rewind erases the screen. Input from the keyboard using the scope to display the typed line is possible using the TV in handler. To use TVIN the NULL handler must be used as the output device associated with the TV display and the buffers for both NULL and TVIN must be in the same field. All standard formatted READ commands are allowed.

READB and WRITEB - These allow reading and writing binary files from FORTRAN. They read and write the next block of 256 12 bit binary words on a particular logical unit to and from a 256 variable array. The values read are between 0 and 4095.

FMTX - This is a FORTRAN multiasking executive which allows simultaneous execution of several subroutines controlling real-time tasks.

PLOTIT and PREVU - These are additions to the Fortran IV plotting capabilities which permit spooling of Calcomp plotter output to a file and then they provide the ability to pull back the contents of such a file and plot it as a background task under Fortran IV.

SCIENTIFIC SUBROUTINE PACKAGE

The version of the SSP that is accessible to OS/8 users that I converted from the PDP 10 SSP package has now been submitted to the DECUS library. If you need a copy of it before it is formally announced you could contact the library. The library prefers that you wait until an item has been formally accepted and announced, so you should only make a special request such as this if you really need something in a hurry before it is fully accepted into the library. Special requests for new programs that are not processed yet cost the library a lot of time and trouble.

IMPROVED EDITORS FROM GEORGE GONZALEZ

George Gonzalez recently forwarded to me information on several variations on editors that he has worked on for the PDP 8. Since most of the professors and graduate students where he is don't have the time and patience to learn how to use EDIT let alone TECO he has made several changes to the BASIC interactive editor to give it better editing capabilities. The most important enhancement aside from fixing "DEC bugs" is the addition of a FOCAL like MODIFY command (or EDIT's "S" command). This allows one to modify a line without having to re-type the whole line. Other convenient features include LNH and RNH commands. The LNH command allows you to specify start listing without a heading at a given line number and list a particular number of lines. He has also ensured that control-C always returns to the editor.

He has also modified the BASIC editor so that it will edit FORTRAN programs with the addition of line numbers. It is capable of chaining to either the FORT or F4 compiler when RUN is typed. This allows the novice programmer to work with FORTRAN without ever learning the subtlies of EDIT or TECO. The normal work flow with this modification is the same as it is with OS/8 BASIC, that is, the compiler is chained to, the program is loaded and executed, and when it terminates or when control-C is typed the return is made to the editor which re-loads the text of the program you're working on.

Mr. Gonzalez also sends along information on a version of a program called XEDIT which is similar to one of the same name which runs on his University's CDC time sharing systems. He likes this editor much better than either EDIT or TECO for most editing purposes. It is certainly an interesting and potentially useful editor with many features too complicated to detail in the Newsletter. Mr. Gonzalez is willing to supply anyone with copies of his BASIC 3.1 editor, his FORTRAN editor or XEDIT on paper tape along with documentation for $2.00, $2.00 and $4.00, respectively. His address is: George Gonzalez, 2626 Keller Parkway, St. Paul, Minn. 55100.

NOTE FROM BRIAN WHARTON

Mr. Wharton sent along a note saying that he has developed a patch for PAL 8 V9
enabling any single logical edit page to be listed on the output device rather
than printing a listing of an entire file. The selection of what page is printed
is a run time option so that you can get around a lot of the problem of having
to include large numbers of XLIST commands to control listings when you have
only slow output devices and/or small storage devices and you are working on
large programs. He says that he will be happy to send a copy of the patch to
anyone who is interested. His address is: ITT Components Group, Europe,
Standard Telephones and Cables, Ltd., Capacitor Division, Brixham Road,
Paignton, Devon, TQ47BE, ENGLAND.

NOTE FROM PHILIP SIEMENS

Phil indicates that he is submitting a new Fortran II library to DECUS that
performs 23 bit arithmetic using mode A of the EAE. On a matrix inversion
benchmark that he has been using the speed improvement is 110% over the stand-
ard OS/8 version 3 Fortran II and 41% over Fortran IV with the EAE. Phil also
included a copy of a straight forward, rather comprehensive speed testing bench-
mark for Fortran programs along with some results. I plan to work with it a
bit more and perhaps publish some results in the future.

NOTE FROM THOMAS E. MERRICK

Mr. Merrick as well as a number of other members of the Special Interest Group
have responded to the problem raised by Lars Palmer regarding a FORTRAN statement
of the form:

$$DO\ 10\ I = J - 1$$

Mr. Merrick and others have noted that this should not be considered as a
source error for the following reason: Spaces may be inserted anywhere in an
executable statement in FORTRAN to improve readibility, including in variable
name thus "DO 10 J" is the same as "DO10J", which is a legal variable name
and "DO10J = J - 1" is a legal executable statement. Note also that the follow-
ing is legal and not a DO statement:

$$DO\ 10\ J = 1.10$$

Mr. Merrick suggests that this type of human typing error can be found by look-
ing at the symbol map and being suspicious of "DOXXJJ" type of variable names
where XX is any number and JJ is an integer.

NOTE FROM A. W. FREDIANI REGARDING RTS-8

Mr. Frediani is a new user of RTS-8 and expresses an interest in getting to-
gether with other people interested in using it in academic and scientific
areas. If you are interested you can contact him at the College of Liberal
Arts, Department of Psychology, Pennsylvania State University, 417 Bruce V.
Moore Building, University Park, PA 16802.

NOTE FROM ROBERT D. COLVETT

Mr. Colvett wrote some time back inquiring as to whether DECUS program 8-397
(8K EDITOR FOR DISC MONITOR) had ever been adapted to OS/8 because it contains
a number of conveniences that are not available in the OS/8 editor. As far
as I know, no one has done this and unfortunately the DECUS library submission
did not include a source. If anyone has any information regarding the subject,
please forward it to me.

NOTE FROM PETER LEMPKIN

Mr. Lempkin writes to say that he has a package running for OS/8 FORTRAN II
that provides functions similar to John Algeo's random access IO package. It
allows one or two page simultaneous input and output files, reading and setting
(like USETI, USETO on the PDP 10) of file block pointers and file sizes so as
to implement random access I/O. Devices may be specified by either number or
ASCII name. The actual I/O is performed either in 8 bit character packed mode
and/or block transfer mode. The package allows the intermixing of these modes.
The I/O package has been used in much of his software over the past year in-
cluding a PDP8E based image processing system called PROCES. A write-up on
PROCES is in the NITS Depository in Arlington Virginia. (The NITS number
is PB24264/AS, $7.00 paper copy; $2.25 microfiche). He plans to submit both
I/O and PROCES to DECUS in the near future.

Mr. Lempkin notes the difficulty of generating binary files and exchanging them
between Fortran II and Fortran IV due particularly to the difficulty of access-
ing the last word in blocks in FORTRAN IV and the fact that FORTRAN IV always
uses a floating point type representation for its data. He is interested in
anyone's thoughts or inputs on the subject. (Perhaps Bob Phelps' READB and
WRITEB are a good solution.) Mr. Lempkin's address is Image Processing Unit,
Division of Cancer Biology and Diagnosis, National Cancer Institute, NIH,
Bethesda, MD 20014.

FLAP. I have recently been interested in FLAP so I started trying to track
it down and found out a few interesting things. First, in spite of what the
documentation says FLAP does not require an FPP-12 to operate. It requires
only an OS/8 configuration the same as RALF. As some of you know, FLAP generates
absolute binary output that can be loaded by ABSLBR the same as PAL8 does, but
FLAP is able to assemble both PDP-8 instructions and FPP-12 instructions.
It was originally intended to be used by people using an FPP-12 without
FORTRAN IV. I have yet to find anyone who has ever used it, however; in fact, I
haven't been able to find anyone who knows whether it is even distributed. It
is not distributed with OS/8 or Fortran IV and the last time I checked with
people who are now in charge of the FPP-12 as a product at DEC they were not
familiar with it either so that it may be that it is not being shipped with
FPP-12's either. It is, however, documented in the OS/8 Handbook and as it
turns out it is generated from the very same source file that RALF comes from.
The difference is a conditional assembly which substitutes appropriate sections
of code to implement the extra pseudo-ops and relocatable binary output for RALF
or else the absolute binary output for FLAP. I tried assembling the current
sources for RALF/FLAP and discovered that through oversights during some of the
more recent additions of new features to RALF; the FLAP conditionals were not

able to assemble a valid version of FLAP, however, a few minor editing changes did produce an apparently functional version of FLAP. Does anyone else have any experience with this?

A couple of interesting things that it might be possible to do with FLAP are:

1. FLAP could be used to assemble RTS-8 tasks that utilize the FPP-12.

2. Consider the possibility of extracting the FPP-12 simulator from the FORTRAN IV runtime system and using it along with FLAP in a manner similar to the old floating point software interpreter that has been used since the first days of PDP-8. You would get a much more powerful floating point package this way and it would be possible to interchange support packages for all the possible configurations from a plain PDP-8 all the way to a machine with an FPP-12. In fact, I have already done some of this. The floating point processer emulator is fairly easy to remove from the run time system and use stand alone.

THE 1975 DECUS EUROPE SYMPOSIUM

A summary of points of interest to the 12 bit Users Group.

As can be seen from a separate protocol in this paper it was decided to form a 12 bit SIG in Europe. We hope to build up a good communication with our American counterpart and that we will have a meaningful exchange of ideas between us. In the session part of the symposium the following points of interest were discussed all of which will be referred in greater depth in the symposium proceedings to be issued later.

V J Blackmore of Christie Hospital, Manchester, England, talked about modifying OS/8 to use a KW 8 display. By modifying the system in such a way that keyboard characters are sent to a special program reciding in the top part of core it was possible to modify the system to use a KW 8 display instead of a teletype. This approach had to be used as a character generated program is too large to fit in a normal handler. The character generated is used as a normal OS/8 handler and calls into core the core resident program for character generation.

Alan Charlesworth of University of Bath, England, talked about a FOCAL based hybrid operating system. In reality a FOCAL interpreter modified so as to be the driver to an analogue computer thereby creating a computer system.

P Handler of University of Madrid, Spain, talked about a lapstick OS/12 FOCAL-12 integrated software. A system whereby it is possible to utilize common file handling conventions in the three operating systems and to boot from one system to the other by keyboard commands.

A special session was given to the Multi-8 system, which is described in a separate bulletin inclosed with this Newsletter I hope. As I have now implemented another of the timesharing monitors for the PDP 8 (the ETOS timesharing system from Educomp Corporation) I shall give a short comparison between these two systems.

The Multi-8 is a complex series of monitors from a small core residence monitor for several tasks up to a background/foreground monitor with the same kind of general idea as the RTS-8 in the foreground except that the tasks are

    a) swapable on disc

    b) relocatable to any page of core within the core allocated for the foreground

The background is the timesharing system which will run up to 4 terminals for normal OS/8 usage. The ETOS system is a pure timesharing system running up to 8 terminals. It does not support the very wide range of hardware that the Multi-8 supports. I have no absolute criteria

for such a statement but I have a feeling that the ETOS system is probably faster in the timesharing part than the Multi-8. The ETOS system utilizes a special hardware which the Multi-8 does not. Both system support almost all OS/8 software. We run in our system FORTRAN IV, TECO, OS/8 BASIC and I know that these things are runable under the Multi-8 system too. However, as always is the case when running a single user program in a multi-user environment some patching in the program is required. Specifically the teletype services routines usually have to be looked into.

The Multi-8 system allocates each user a separate SYS area on the disc while the ETOS system uses a common disc SYS area write locked to all users. This second approach has the advantage of making program development done at a central console faster available to all users and also is probably a bit more foolproof for the normal user who has no chance of destroying his SYS area by a program running wild. However, it does have a disadvantage. The programs that modify the SYS area have to be changed. FORTRAN IV compiler e.g. has to be changed so as to use DSK area for scratch storage and the batch monitor can not be run at all as it heavily modifies the SYS area. It will be very interesting to see in the years to come which timesharing OS/8 system becomes the standard in the OS/8 community. Let us hope that the systems becomes standardized and we will not run into the situation where we have several systems with slightly incompatible programs, which is not a situation to look forward to.

AFTER THE SOMEWHAT FRAGMENTARY INFORMATION IN THE LAST OS/8 SIG NEWSLETTER I WILL TRY TO BE MORE SPECIFIC AS TO THE STATUS OF THE MULTI8 PROJECT.

AT THE DECUS EUROPE SYMPOSIUM HELD 10-12 SEPT '75 IN THE HAGUE, THE FIRST RELEASE OF THE MULTI8 FOREGROUND/BACKGROUND SYSTEM WAS ANNOUNCED AND DEMONSTRATED. BY COURTESY OF THE DUTCH BLAST FURNACES THE IMPLEMENTERS OF MULTI8 WERE ABLE TO DEMONSTRATE A TWO USER VERSION ON A 16K PDP8/E WITH RK8E AND TD8E DECTAPE (1). APART FROM HARDWARE BREAKDOWN THE SYSTEM BEHAVED WELL AND SOME ATTENDEES WERE ABLE TO DEMONSTRATE VARIOUS OS/8 PROGRAMS, NOTABLY A MINIATURE COBOL RUNNING IN THE 8K BACKGROUND OF MULTI8. THIS FIRST RELEASE DOES NOT REQUIRE ANY SPECIAL HARDWARE AND SUPPORTS UP TO FOUR USER OS/8 TIMESHARING ON A 16K MACHINE. SUPPORTED PERIPHERALS INCLUDE RK8E, RF08, DF32, TC01, TC08, TD8E, LPT, PTP, PTR, AND VARIOUS TERMINALS OF DIFFERENT SPEED AND FILLER CHARACTER NEEDS.

THIS DECUS SYMPOSIUM WAS THE LAST OCCASION THAT WE HAD FLOOR ANTHONI WITH US IN HOLLAND AS HE LEFT EUROPE AT THE END OF SEPTEMBER TO START ALL OVER AGAIN IN NEW ZEELAND. I THINK WILL HEAR MORE OF HIM FROM THE OTHER SIDE OF THE WORLD.

IN THE MULTI8 WORKSHOP SOME POSSIBLE FUTURE EXTENSIONS WERE DISCUSSED, E.G. DEMAND-PAGING (PER 4K), HARDWARE SUPPORT OF CDF CIF ETC., IMPLEMENTATION OF SOME DECNET TASKS, EXTENSION TO 7 USERS, ETC. SOME OF THESE DEVELOPMENTS HAVE ALREADY BEEN STARTED.

MULTI-8 IS DISTRIBUTED BY WESTFRIES CONSULTING BV, A SMALL DUTCH SOFTWARE HOUSE AND I THINK I HAVE TO EXPLAIN WHY WE DON'T SAY PUT IT IN THE DECUS LIBRARY.

WE THINK THE DECISION TO INSTALL A NEW OPERATING SYSTEM HAS ABOUT THE SAME IMPORTANCE AS THE CHOICE OF A MAINFRAME. ONES AN OPERATING SYSTEM IS IN USE A LARGE INVESTEMENT IN PROGRAMMING IS BUILT UPON IT AND ONE HAS TO BE SURE THAT THE OS WILL CONTINUE TO SERVICE FOR A NUMBER OF YEARS AND HAS THE ABILLITY AND SUPPORT TO INCORPORATE NEW PERIFERALS, CORE EXTENSIONS AND APPLICATIONS. IN THIS LIGHT IT SEEMS RESONABLE TO SPEND A FEW PERCENT OF THE HARDWARE INVESTMENT ON SOFTWARE.

ADDRESSES:

TECHNICAL-
ERNST LOPES CARDOZO,
PHYSIOLOGY LABORATORY STATE UNIVERTITY
VONDELLAAN 24 UTRECHT, THE NETHERLANDS.

DISTRIBUTION-
WESTFRIES COMPUTER CONSULTING B.V.
P.O. BOX 20
OOSTZAAN, THE NETHERLANDS

CORRECTION TO [3N ARTICLE ON PAGE 47 OF OCTOBER 1975 ISSUE
SUBMITTED TO DSN BY STANLEY RABINOWITZ

ON THE THIr) LINE OF THE PATCH "4532" SHOULD READ "4352"
THE ENTIRE LINE SHOULD READ:

        2561/XXXX 4352; 4313; 5211; 7706

THIS IS REPLACEMENT ARTICLE 1 TO SEQUENCE ARTICLE 2 FOR CREF V3, OS/8 V3
==
OS/8   V3C   CREF   V4
FIXES TO CREF
SUBMITTED 2-FEB-76 TO DSN BY STANLEY RABINOWITZ

THE FOLLOWING PATCH INCORPORATES PATCHES TO CREF V3A WHICH WERE FOUND
TOO LATE TO BE INCLUDED IN THE OS/8 V3C RELEASE. THESE PATCHES UPGRADE
CREF V4 TO CREF V4A.

        . GET SYS CREF
        . ODT
        2576/0240 101
        4353/XXXX 1356; 3042; 5235; 35
        4507/5767 5755
        4555/XXXX 4353
        6016/1036 5310
        6110/XXXX 4713; 1036; 5217; 741
        0107/XXXX 3512; 3513; 5506; 413; 431; 7564
        3242/5564 5355
        3355/XXXX 4106; 5564
        5623/3776 4106
        4300/0145 201; 1144
        3254/7650 7450
        3256/1010 1161; 7650; 5273; 1010; 1372; 7700
        3264/1023 5267; 1023; 4505; 1023; 1114
        ^C
        . SAVE SYS CREF
==
OS/8   V3   F4
PROBLEM WITH MULTIDIMENSIONAL ARRAYS
SUBMITTED 18-JAN-76 TO DSN BY STANLEY RABINOWITZ

DATA INITIALIZATION OF MULTI-DIMENSIONAL ARRAYS DOES NOT WORK PROPERLY.
THE F4 COMPILER IS ERRONEOUSLY COMPUTING THE SUBSRIPTS OF THE ARRAY.

THE FOLLOWING PATCH CORRECTS TIS PROBLEM:

        . GET SYS F4
        . ODT
        2066/7126 4; 1256
        2121/7240 7201
        ^C
        . SAVE SYS F4
==

CONDITIONALS INSIDE ITERATIONS
SUBMITTED 28-JAN-76 TO DSN BY STANLEY RABINOWITZ

TECO DOES NOT PROPERLY HANDLE UNSATISFIED CONDITIONALS IF OTHER
CONDITIONALS ARE ENCOUNTERED WITHIN AN INNER ITERATION WHILE SCANNING
FOR THE TERMINATING SINGLE QUOTE.
THIS IS BECAUSE TECO INCORRECTLY BUMPS THE CONDITIONAL COUNT WHENEVER
IT SEES A DOUBLE QUOTE. THIS SHOULD NOT BE DONE AT NON-ZERO
NEST LEVELS.

THE FOLLOWING PATCH FIXES THIS PROBLEM AND UPGRADES TECO TO VERSION 304

```
. GET SYS TECO
. ODT
2560/7240 4144
145/XXXX  1752; 7640; 5512; 7240; 5744; 2711
1372/0457 460
^C
. SAVE SYS TECO
```

==

OS/8   V3   BASIC   V3.0
LONG INPUT LINES
SPR SUBMITTED 25-NOV-1975 BY WALTER C. DAUGHERITY

IF MORE THAN ABOUT 100 CHARACTERS ARE READ BY AN 'INPUT' STATEMENT
IN BASIC WITHOUT A CARRAGE RETURN, BRTS GETS LOST. CARRAGE RETURN,
CONTROL-C THEN WHIPES OUT THE MONITOR HEAD (07600-07777) REQUIRING
REBOOTSTRAPPING.

==

OS/8   V3   ODT
ODT USE OF LOCATIONS 07744-07746
SPR SUBMITTED 25-NOV-76 BY WALTER C. DAUGHERITY

DESPITE THE INFORMATION ON PAGE 1-114 OF THE OS/8 HANDBOOK, ODT DOES NOT
APPEAR TO USE LOCATIONS 07744-07746 AS DOCUMENTED. SEE EXAMPLE:

```
07744/6203
07745 /0000 200
07746 /6003 3401
00000/0007 1
00001 /7402
07746/3401
^C
. SAVE SYS TEST 00000-00001

. GET SYS TEST

. ODT

07744/6203
07745 /0000
07746 /2000
00000/0001
00001 /7402
07746/2000
```

(NOTE: I CAN NOT REPRODUCE THIS PROBLEM WITH THE OS/8 V3C MONITOR -R. H.)
==

OS/8 HANDBOOK AND DIGITAL SOFTWARE NEWS
SPR SUBMITTED 25-NOV-75 BY WALTER C. DAUGHERITY

1) ON PAGES 1-58/59 THE HANDBOOK SHOULD INDICATE THAT 'COMPA' IS
REQUIRED FOR 'COMPARE'. ('COMP' IS INTERPRETED AS 'COMPILE') ALSO P. 1-53
2) DSN FEB 75 - CHAINING TO 'LOADER' - SHOWS THE OLD CONTENTS
OF LOCATION 35 INCORRECTLY.
3) DSN FEB 75 - MEANINGLESS CODE GENERATED BY 'LIST XXXX' - SHOWS THE
OLD CONTENTS OF LOCATION 01427 INCORRECTLY.
4) DSN APRIL 75 - MAGNETIC TAPE - REFERS TO THE WRONG PAGE
5) DSN APRIL 75 - SHORTENING THE LENGTH OF THE RF08 SYSTEM DISK - WILL
NOT WORK IF ANOTHER SYSTEM DEVICE IS ACTIVE WHEN BUILD IS RUN, AS WOULD
ORDINARILY BE THE CASE FOR A USER WHO HAD GENERATED A SYSTEM WITH A
SYSTEM DEVICE OTHER THAN THE RF08 AND THEN RESAVED 'BUILD'.
A MORE GENERAL SOLUTION IS AS FOLLOWS:

```
. RUN SYS BUILD
$DELETE SYS
$INSERT RF08, SYS
$SIZE SYS
2000/1777
$ALTER RF08, 7
0001/2
$DELETE SYS
$^C
. SAVE SYS BUILD
```

6) DSN JULY 75 - BASIC GETS LOST - SHOULD HAVE AN '.ODT' COMMAND
AFTER THE '.GET'.
7) DSN OCT 75 - MISSING FORM FEED - CONTAINS A SERIOUS ERROR WHICH
RESULTS IN 'CREF'S' PRINTING AN INFINITE NUMBER OF A'S AT THE END
OF ITS LISTING. THE CORRECT PATCH TO LOCATION 2561 IS 4352.
8) DSN OCT 75 - /W OPTION FAILS - SHOWS THE OLD CONTENTS OF LOCATION
2314 INCORRECTLY.
9) DSN DEC 75 - USING TTY HANDLER WITH BRTS - CONTAINS A SERIOUS
ERROR WHICH RESULTS IN THE FIRST BLOCK OF 'BRTS' BEING ZEROED BY A
'PRINT' STATEMENT. THE CORRECT PATCH TO LOCATION 5373 IS 3053.
10) OS/8 HANDBOOK ERRORS AND CHANGES:
P. 1-37 BIT 3=1 PROGRAM BEING RUN WILL NOT DESTROY THE BATCH MONITOR
         (5000-7577 OF THE HIGHEST FIELD)
P. 1-51 THE OPEN BRACKET ([) IS PRODUCED ON TELETYPEWRITERS BY ...
P. 1-84 CTRL/L(FORM)
P. 1-86 CTRL/L (FORM)
P. 1-88 IF NO OUTPUT FILE IS SPECIFIED, THE 'J' OR 'F' COMMAND
         READS THE NEXT ...
P. 1-91 TWO PLACES (E AND G): CTRL/L (FORM)
P. 1-93 TWO PLACES (A AND C): CTRL/L (FORM)
P. 1-108 PARAGRAPH 2. SEEMS INAPPROPRIATE FOR 'SORRY-NO INTERRUPTIONS'
            (PREVIOUS PAGE)
P. 1-110 "THE /I OPTION CAN BE USED WHEN MAKING PATCHES TO AN ALREADY
            SAVED PROGRAM WITHOUT REASSEMBLING THE ENTIRE
            PROGRAM. SEE EXAMPLE 1 BELOW. "
P. 1-111 /N LOAD ALL FILES SPECIFIED ON THIS INPUT LINE INTO FIELD N
            (WHERE N IS AN OCTAL INTEGER) UNTIL A 'FIELD' SETTING
            IS ENCOUNTERED.

P. 1-117 CONTROL COMMANDS
          NOTE: ADDRESSES IN THE FOLLOWING COMMANDS MAY BE 5 OCTAL DIGITS.
P. 1-118 A-OPEN AC
          L-OPEN L
P. 1-120 M LINE-FEED LINE-FEED - OPEN UPPER SEARCH LIMIT
          INITIALLLY, THE UPPER SEARCH LIMIT IS 7577.
P. 1-121 TELETYPE EXAMPLE HAS MANY ERRORS: SEARCH LIMITS ARE IN 41
          AND 42, AND THE FOUR 'ISZ' INSTRUCTIONS "FOUND" ARE NOT IN
          ADDRESSES WITHIN THE SEARCH LIMITS.
P. 2-40 REMOVE "BUILD" AND ADD "SIZE" BEFORE "SYSTEM"
P. 2-53 REMOVE 'BUILD' DESCRIPTION AND ADD 'SIZE' DESCRIPTION
          FROM DSN JUNE 75
P. 2-178 *◀STRING$$
          *N◀STRING$NENDSTRING$PWEF$$
          THE FIRST GOOD LINE OF YPUR FILE; IF SO, DELETE THEM.
P. 3-28 THE SECOND INSTRUCTION SHOULD BE 'CDF 00'
P. 3-30 "THE /F OPTION ... "
P. 3-31 "THE 'IFNDEF' PSEUDO-OP ... "
P. 3-41 THE RIGHTMOST COLUMN UDER GROUP 1 OPERATE MICROINSTRUCTIONS
          SHOULD BE HEADED "SEQUENCE".
          'BSW' IS SEQUENCE 4
P. 3-42 'OSR' IS SEQUENCE 2
          THE RIGHTMOST COLUMN UNDER COMBINED OPERATE MICROINSTRUCTIONS
          SHOULD BE HEADED "TIME". THE TIMES GIVEN SEEM INCONSISTENT
P. 6-41 (LINE NUMBER) 'UDEF' FUNCTION NAME (ARGUMENT)
P. 6-94 SECOND INSTRUCTION SHOULD BE 'CLL RTL'
P. 6-122 AFTER THE 'END' STATEMENT ADD:
          "HOWEVER, IF ANY ERROR MESSAGES ARE GENERATED THEY MAY NOT BE
          ABLE TO IDENTIFY THE STATEMENT IN ERROR IF IT IS UNNUMBERED. "
P. 7-2   /K KEEP THE FILE 'FORTRN. TM' AS A PERMANENT FILE.
P. 7-3   THE FILE 'FORTRN. TM' IS THEN DELETED ...
P. 7-46 THE FORTH THROUGH SIXTH LINES BELONG AT THE BOTTOM OF THE PAGE
          TO COMPLETE NOTE 7.
P. INDEX-3 CCL (NOT CCY)
P. INDEX-8 THIS PAGE IS NOT ALPHABETIZED
===

FROM T. WES SYKES

THE FOLLOWING PATCHES MAY BE HELPFUL TO USERS OF OS/8 BASIC VERSION 3:

1. BRTS.SV

    CHANGE 01773 FROM 1077 TO 7000 TO INHIBIT THE ADDITION OF THE
    HIGH ORDER BIT ON MOST OUTPUT CHARACTERS(THE EXCEPTIONS BEING
    CR, LF, AND CONTROL/Z).   THIS PROVIDES A METHOD OF OUTPUTTING
    CONTROL CHARACTERS AND X-Y COORDINATES FOR PLOTTING ON
    TEKTRONIX TERMINLS BY USING THE "PRINT PNT()"
    INSTRUCTION

2. BASIC.FF

    .GE SYS:BASIC.FF

    .OD

    ```
    14246/1336 7325 ( TAD K3 BECOMES NL0003 )
    14336/0003 7605 (K3 BECOMES POINTER TO 7605)
    14345/3773 3736 (NEW STORE INDIRECT TO 7605)
    14373/7605 7344 (POINTER TO 7605 BECOMES NL7776)
    14374/XXXX 1305 (ADD USR FUNCTION WD TO -2
                     TO DECIDE IF CALL IN ERROR WAS LOOKUP OR ENTER)
    14375/XXXX 7650 (SNA CLA - IF NOT ZERO IT WAS NOT LOOKUP)
    14376/XXXX 5543 ( JMP TO SET EOF FOR THIS FILE)
    14377/XXXX 4516 ( JMS TO ERROR ROUTINE)
    14310/4516 5373 ( CHANGE ERROR CALL TO JMP TO PATCH)
    ```

    CTRL/C
    .SA SYS:BASIC.FF

    .GE SYS:BASIC.SF

    .OD
    ```
    12635/3467 3400 ( CHANGE -(OLD ERROR CALL)-1 TO -(NEW ERROR CALL)-1)
    ```

    CTRL/C
    .SA SYS:BASIC.SF

    THIS PATCH PROVIDES THE USER WITH THE ABILITY TO ENQUIRE
    ABOUT THE PRESENCE OF A FILE WITHOUT THE HASSLE OF THE FATAL
    ERROR "EN".   ALL THAT IS NECESSARY TO DO IS PERFORM
    THE "FILE #N:" OR "FILEN #N:" INSTRUCTION AND THEN TEST
    WITH THE " IF END #N THEN XXX".   SINCE I HAVE A RK8E DISK
    I WAS NOT ABLE TO CHECK THAT THE ERROR FOR THE ENTER
    OPERATION PERFORMS AS BEFORE BECAUSE I CAN USUALLY OPEN OUTPUT.

3. BASIC.UF

    THE ONLY PATCH I HAVE CONCERNING THE LAB 8/E FUNCTIONS
    IS FOR THE "JMS" VECTORS IN BRTS.SV
    THE OS/8 HANDBOOK(PAGE 6-126) INDICATES THAT 1565 SHOULD BE 3541
    AND 1567 SHOULD BE 3521(FOR ADC AND CLW FUNCTIONS). THEY HAVE
    CHANGED IN VERSION 3 BASIC TO 3542 AND 3522, RESP.
    I DON'T THINK THAT THIS HAS BEEN PUBLISHED ,SO I THOUGHT
    I WOULD PASS IT ON.

OS/8 Quiz -- Stanley Rabinowitz –

1. How can a user program print a message on the batch log?

2. What is a shorter form for the following command?

   .PAL SYS:PROG,LPT:<PROG

3. Which of the following cause disastrous results?

   (a) Typing CTRL/C while squishing SYS:

   (b) Write-locking SYS: while FOTP is in the middle of deleting a set of files

   (c) Punching a hole in your system DECtape with a hole puncher

   (d) Turning your system LINCtape off line while trying to write on SYS:

   (e) Trying to run OS/8 on a TC08 using a PDP-11 formatted DECtape

4. Is there a way to create a directory which has no additional information words, not even one for the date?

5. Your TECO macro is running amok and out of control. How can you stop it without losing the file?

6. How can you give today's date to all files on a DECtape?

7. How can you change the title at the top line of your PAL8 listing?

8. If drive 0 of your dual RK8E goes down, is there any way to run OS/8 using drive 1?

9. Can FOTP be aborted without returning to the monitor?

10. What cusps do anything with additional information words after the date?

11. Which of the following cusps can be chained to?

   (a) TECO

   (b) CCL

   (c) BOOT

   (d) CAMP

12. What does the /D option in FOTP stand for?

1. First, check if BATCH is running. (Bit 1 of location Ø7777 is 1)
   Get field of BATCH (bits 6-8 of location Ø7777). Then, for each
   character you want to print on BATCH log, execute the following:

        CDF userfield
        CIF batchfield
        TAD (CHAR
        JMS I (BATOUT

   where BATOUT=74ØØ.   (Ref: page 3-1Ø of software support manual.)

2. .PAL PROG-L

3. none of these.

   (a) PIP is good! It will respond: SORRY - NO INTERRUPTIONS and
       continue with the squish.

   (b) FOTP is good! FOTP prints: ORIGINAL DIRECTORY PRESERVED.

   (c) DECtapes are indestructable! Redundant recording of data
       enables the tape to still be read.

   (d) LINCtapes are good! The LINCtape controller will 'hang'
       until you reselect the drive; then the operation will continue.

   (e) TCØ8 controllers are good! The controller causes OS/8 to use
       the first 128 words in each DECtape block. All the extra
       words are ignored. [Don't try this with the TD8E controller.]

4. Yes. The command .ZERO dev:=Ø is treated the same as .ZERO dev:=1.
   However, the = option on ZERO and SQUISH is taken modulo 1ØØ
   to allow you to specify Ø additional information words, i.e., type

        .ZERO dev:=1ØØ

   This will give you a directory without room for dates, but you
   will have the capacity to handle more files.

5. Type CTRL/P .

6. With V3, one way to do this is to copy all the files to another
   DECtape using the /T option. A much simpler scheme is available
   with V3c:

                .RENAME DTAØ:*.*/T

7. The EJECT pseudo-op uses the text following it to specify a new title.

8. Yes. Use the bootstrap given on page 1-27 of the OS/8 handbook.

9. Yes. Type CTRL/P .

1Ø. PIP - ZERO and SQUISH create them.

    DIRECT - /I lists additional information words in octal

    FOTP - properly copies additional information words

    RESORC - tells you if you have room for additional information words (/E)

11. all of them

    (a) Pass legal TECO command to TECO by putting it in 176ØØ,
        one character per word.

    (b) Pass legal CCL command to CCL by putting it in 176ØØ, one
        character per word, terminated by a Ø.

    (c) Leave legal BOOT command in OS/8 KBM line buffer.

    (d) Leave legal CAMP command in OS/8 KBM line buffer.

12. /D stands for Don't copy. It definitely does not mean "Delete",
    since there are combinations of switches which can be used in FOTP
    which include the /D switch which do not delete any files.
    /N stands for No predelete. If both /D and /N are used, then there
    will be neither predeletion nor postdeletion. Consequently, no
    files will be deleted.

                    Scoring:

1Ø or more correct - You are an OS/8 Guru

8 or 9 correct - You are an OS/8 expert

5-7 correct - You have some familiarity with the internal workings
                of that inexplicable and wondrous product known as OS/8

2-4 correct - You are a novice, and should think twice before using
                programs such as FOTP, TECO.

less than 2 correct: limit yourself to commands like .DATE

ACROSS

1. Used for compression

2. Existential tester

7. Argument habitat

9. Transfer mode guaranteeing file integrity

10. Most cusps do this

11. Way to omit symbol table

12. Alphanumeric field specifier in FORMAT

13. Data eater

16. OS/8 can run on this

19. Command decoder mode

21. Switch allowing bit-mapping of .SV files

22. Type of display

23. Switch

25. Type of loop

26. Default extension used with 41 down

27. Command used by FORTRAN and BATCH

28. Way of starting a program

30. Default extension to be used by new assembler

32. Handler with many conditionals

34. Powerful language

36. Way to get bigger DIRECTORY and RESORC listings

37. How to fill your TTY with LG error messages

38. Way to use more memory

39. Special character processed by 32 across

DOWN

1. FORTRAN function

2. Switch not used by PAL 8

3. This consists of data

4. BRTS error message which is the same as a FORTRAN keyword

5. Greater than L option

6. Less than L option

7. Default transfer mode used by 16 down

8. USR function which BASIC can perform

11. Device used to protect injured body part

14. Way to get rid of user service routine

15. Usually fast non-file-structured-device

16. Cusp used to copy system heads

17. Typically first command typed

18. Affectionate name for 16 across

20. Type of block

22. Handler which reads holes

24. OS/8 patcher

26. Opposite of R (TECO)

29. BASIC statement

31. Name of location wherein contents of accumulator are stored upon a breakpoint

33. Reply to PIP

35. FORTRAN function which is also name of a rival PDP-8 operating system

36. DEC's inadequate substitution for FUTIL

38. basic part of OS/8

41. sign used by CCL

OS/8 Crossword Puzzle

— Stanley Rabinowitz —

A crossword grid with numbered cells 1–49.

ACROSS (cont.)

4Ø. This occurs many times in PAL8 and F4

43. A word has lots of these

45. Sympathetic sorrow

46. Way to get bigger CREF listings

47. FPP has one of these

48. Default extension to get

DOWN (cont.)

42. Handler which does not respond to CTRL/C

44. Mathematical name for FORTRAN (Ø.Ø,1.Ø)

45. Default extension preceding 3Ø across

47. Option used with multi-volume transfers

49. Used to get version numbers

DC02 Multiple Terminal Handler for OS/8

James H. Donnelly
Appalachian Laboratory for Occupational Respiratory Diseases
Medical Research Branch
Morgantown, W. Va.

Many application programs in our laboratory are written in FORTRAN II
and IV. To facilitate I/O from these programs to several remote terminals,
an OS/8 compatible DC02 handler has been developed. This handler is a
modification of the ASR33 handler.

To implement the DC02 handler, simply append the code listed below to
ASR33.PA, delete the ASR33 page zero code, the origin at 200, change the
contents of location TTYCTO to MKCC, and assemble with PAL8.

The added code will redefine the teletype nemonics. Since these nemonics
are part of PAL's permanent symbol table, error RD 101 will be typed on
the console terminal. Ignore the messages; they are there only to inform
you that the symbols have been redefined.

/ DC02 HANDLER

```
         0012                 OUTDC=12
         0011                 INDVC=11
         6117                 MTON=6117
         6112                 MKCC=6112

         0000          *0
00000    7770                 -10
00001    0403                 DEVICE DC02;DEVICE DC0;410;DC0&177+4000;ZBLOCK 2
00002    6062
00003    0403
00004    6000
00005    0410
00006    4110
00007    0000
00011    0403                 DEVICE DC02;DEVICE DC1;410;DC1&177+4000;ZBLOCK 2
00012    6062
00013    0403
00014    6100
00015    0410
00016    4107
00017    0000
00021    0403                 DEVICE DC02;DEVICE DC2;410;DC2&177+4000;ZBLOCK 2
00022    6062
00023    0403
00024    6200
00025    0410
00026    4106
00027    0000
00031    0403                 DEVICE DC02;DEVICE DC3;410;DC3&177+4000;ZBLOCK 2
 0032    6062
00033    0403
00034    6300
00035    0410
00036    4105
00037    0000
00041    0403                 DEVICE DC02;DEVICE DC4;410;DC4&177+4000;ZBLOCK 2
00042    6062
00043    0403
00044    6400            .
00045    0410

00046    4104
00047    0000
00051    0403                 DEVICE DC02;DEVICE DC5;410;DC5&177+4000;ZBLOCK 2
00052    6062
00053    0403
00054    6500
00055    0410
00056    4103
00057    0000
00061    0403                 DEVICE DC02;DEVICE DC6;410;DC6&177+4000;ZBLOCK 2
00062    6062
00063    0403
00064    6600
00065    0410
00066    4102
00067    0000
```

```
00071   0403        DEVICE DC02;DEVICE DC7;410;DC7&177+4000;ZBLOCK 2
00072   6062
00073   0403
00074   6700
00075   0410
00076   4101
00077   0000


        6111            KSF=10^INDVC+6001
        6112            KCC=10^INDVC+6002
        6114            KRS=10^INDVC+6004
        6116            KRB=KCC KRS
        6121            TSF=10^OUTDC+6001
        6122            TCF=10^OUTDC+6002
        6124            TPC=10^OUTDC+6004
        6126            TLS=TCF TPC



        0001            DC0VERSION="A&77



        0200        *200

00200   0010    DC02C.  10                      /UNIT SELECT, UNIT 1
00201   1310    DC0TAD. TAD     DC0

00202   2203    DC0ISZ. ISZ     DCNMB

00203   0000    DCNMB.  0

        0300        *300
00300   0001    DC0V.   DC0VERSION

00301   2203    DC7,    ISZ     DCNMB
00302   2203    DC6,    ISZ     DCNMB
00303   2203    DC5,    ISZ     DCNMB
00304   2203    DC4,    ISZ     DCNMB
00305   2203    DC3,    ISZ     DCNMB
00306   2203    DC2,    ISZ     DCNMB
00307   2203    DC1,    ISZ     DCNMB
00310   2203    DC0,    ISZ     DCNMB

00311   7200            CLA
00312   6214            RDF                     /GET USERS DATA FIELD
00313   1315            TAD     CDF0    /MAKE A CDF TO USERS FIELD
00314   3363            DCA     CDFU    /SAVE IT
00315   6201    CDF0.   CDF     0       /SET DATA FIELD TO 0
                                        /HANDLERS ALWAYS IN FIELD 0.
00316   1203            TAD     DCNMB   /RESTORE ENTRY TO MAKE HANDLER REUSABLE
00317   7041            CIA
00320   1201            TAD     DC0TAD  /MAKE A TAD DC0 + N
00321   3326            DCA     DCFUN   /SAVE IT
00322   4366            JMS     LOC     /FIND OUT WHERE THE TTY ROUTINES ARE
00323   7305            CLA CLL I&C RAL /   AC= 2
00324   1377            TAD     DC0X    /MAKE AN ENTRY TO THE DC02 HANDLER
00325   3377            DCA     DC0X    /TO BE USED FOR ARG PICKUP AND EXIT
00326   7402    DCFUN.  HLT             /BECOMES TAD DC0 + N
00327   3777            DCA I   DC0X    /POINTER FOR ARG PICKUP AND EXIT
00330   7332            CLA STL RTR     /MAKE AN ISZ TO REPAIR ENTRY SKIP CHAIN
                                        / 2000 > AC
```

```
00331   1326            TAD     DCFUN   /DCFUN = TAD DC0 + N
00332   3354            DCA     REP
00333   1202            TAD     DC0ISZ  /MAKE AN ISZ TO FIX DESTROYED ENTRY POINT

00334   7402    REP,    HLT             /BECOMES DCA TO DESTROYED ENTRY POINT ISZ
00335   7300            CLA CLL
00336   1203            TAD     DCNMB   /PICKUP DEVICE NUMBER
00337   7041            CIA             /MAKE A JMP TO DEVICE SELECT TABLE
00340   1365            TAD     JMPROR
00341   3343            DCA     JROL

00342   7320            CLA CLL STL
00343   7402    JROL,   HLT             /BECOMES JMP TO DEVICE SELECT TABLE
                /
                /  MOVE DEVICE SELECT BIT TO BITS 0-7
                /  BIT 0 IS SET FOR DEVICE ZERO ON DC02
                /  BIT 7 IS SET FOR DEVICE 7.

00344   7010            RAR             / 7
00345   7010            RAR             / 6
00346   7010            RAR             / 5
00347   7010            RAR             / 4
00350   7010            RAR             / 3
00351   7010            RAR             / 2
00352   7010            RAR             / 1
00353   7010    JROR0,  RAR             / 0
00354   1200            TAD     DC02C   /UNIT SELECT BIT
                                        /CURRENTLY SET TO SELECT UNIT 1
                                        /BITS 8 THRU 11 SELECT DC02 UNIT
00355   6117            MTON            /ENABLE (SELECT) UNIT AND DEVICE
00356   6112            MKCC            /CLEAR THE KEYBOARD FLAG
                /
                / FINISH CONSTRUCT OF A PSEUDO JMS TO THE TTY HANDLER
                /
00357   7301            CLA CLL IAC
00360   1377            TAD     DC0X
00361   3377            DCA     DC0X
00362   3203            DCA     DCNMB   /ZERO DEVICE NUMBER SO
                                        /SO WE CAN USE IT AGAIN
00363   7402    CDFU,   HLT             /BECOMES CDF TO USERS FIELD
                                        /SO THE TTY HANDLER WILL KNOW WHERE THE
                                        /USER IS LOCATED
00364   5777            JMP I   DC0X    /INDIRECT TO TTY HANDLER


00365   5353    JMPROR, JMP JROR0


00366   0000    LOC,    0
00367   7200            CLA
00370   1366            TAD     LOC
00371   0375            AND     A76
00372   1376            TAD     A177
00373   3377            DCA     DC0X
00374   5766            JMP I   LOC
```

```
00375  7600  A76,    7600
00376  0177  A177,   0177


       0377  *377
00377  0000  PCOX,   0         /INDIRECT JMP TO TTY HANDLER


       0400          PAGE
                /
                /
```

Rudolf ALBRECHT

Helmut JENKNER

University-Observatory, Vienna, Austria.

Because of its nature as an interactive language, FOCAL is not very well suited for astronomical data reduction work including extensive arithmetics: the calculating speed is just too slow. Thus, we have adopted OS/8-FORTRAN for the data reduction.

The data to be worked on are usually produced by assembly-language data acquisition programs. As long as they are integers they can be read from LINCtape by both FOCAL and FORTRAN (1, 2). It is, however, not possible to use FOCAL for looking at FORTRAN-produced floating-point data on LINCtape. The reason is the different floating-point format of the two languages.

FOCAL FP-format:

|← Exponent →| |←————— Mantissa ————→|
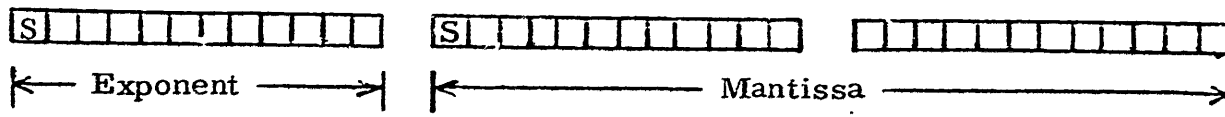
FORTRAN FP-format:
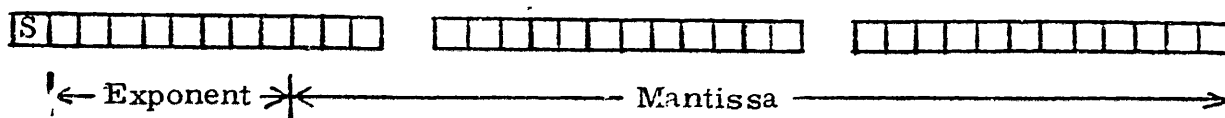
|← Exponent →|←————— Mantissa ————→|

S ... Sign-bit

The FORTRAN mantissa is a total of four bits longer than the FOCAL mantissa, trading a limitation in range for a higher numerical accuracy of numbers. FORTRAN uses an excess 128 exponent, so it never becomes negative and the only sign-bit applies for the whole FP-word.

A FOCAL-12 user function has been developed to convert
FORTRAN-written FP-data to FOCAL-format.

Example:

1.10 L O, FO, F,#n, 1

...

1.70 F I=0,10; S A(I)=FNEW(D, FO(I))

The first argument D in the list is a dummy argument and it has
the following purpose: upon returning the variable FO(I) FOCAL
goes through a normalisation routine, rearranging the FP-word
so the most significant bit occurs to the right of the mantissa
sign-bit and adjusting the exponent accordingly. This is obviously
not desirable in case FO(I) contains a FORTRAN variable. FNEW
therefore changes (and later restores) the FOCAL function return,
so there will be no normalizing. This, however, is only possible,
if the file-call occurs in the second argument, since the first
argument is evaluated by FOCAL before FNEW gains control.

The need to temporarily eliminate the normalisation also means
that the file-call has to occur in the argument list and that it
must be just a file-call. In other words, calls like:

1.30 S T=FO(I)
1.40 S A=FNEW(D, T)

or

1.20 S T=FNEW(D, FO(I)+25)

are not possible.

Several things have to be considered when writing the data onto
LINC tape: FOCAL only can read 256-word blocks and it expects a
block to start with the exponent of an FP-word, even though this
means wasting one integer word upon crossing block boundaries.

This is not necessarily the case in FORTRAN, causing the FP-words to be out of alignment after the first block of the respective file. The problem can be solved by writing data block by block. For writing we are generally using the routine WLINC (1).

It should be kept in mind that FORTRAN array subscrips start at 1 while FOCAL file subscrips start at 0.

A listing of the FNEW function is appended. Instructions on how to implement the user function to FOCAL can be found in section 7 of the FOCAL-12 manual.

References:

(1) Charles M. MOORE III, RWLINC.SB, PS/8 FORTRAN
    Library Routines, DECUS No. 12-48.

(2) Charles M. MOORE III, LTAPE.SB, PS/8 FORTRAN
    Library Routines, DECUS No. 12-48.

```
0000                              *20
0001                              /FOCAL-12 USER FUNCTION
0002                              /TO DECIPHER FORTRAN-
0003                              /WRITTEN FP-VALUES
0004                              /CALL
0005                              /    S A=FNEW(D,FO(I))
0006                              /
0007                              / D...DUMMY ARGUMENT
0010                              / FO..OPENED FORTRAN
0011                              /       FILE
0012                              /
0013                              /NOTE: THE FILE CALL MUST
0014                              /OCCUR IN THE ARGUMENT LIST
0015                              /
0016                              PMODE
0017                              *35
0020     0035   4467             FN-1
0021                              *410
0022     0410   4470             FN
0023                              *4470
0024     4470   7200    FN,      CLA
0025     4471   1373             TAD POINT
0026     4472   3374             DCA CPOINT
0027     4473   1375             TAD SW1          /CHANGE FOCAL
0030     4474   3774             DCA I CPOINT     /FUNCTION
0031     4475   2374             ISZ CPOINT       /RETURN
0032     4476   3774             DCA I CPOINT
0033     4477   2374             ISZ CPOINT
0034     4500   1376             TAD SW2
0035     4501   3774             DCA I CPOINT
0036     4502   4545             GETC
0037     4503   4540             PUSHJ            /GET ACTUAL
0040     4504   1613             EVAL             /ARGUMENT
0041     4505   7300             CLA CLL
0042     4506   3372             DCA SFLAG
0043     4507   1044             TAD EXP
0044     4510   7510             SPA              /SIGN?
0045     4511   2372             ISZ SFLAG
0046     4512   7010             RAR              /EXTRACT LOWER
0047     4513   7012             RTR
0050     4514   7012             RTR              /BITS
0051     4515   0364             AND MASK1        /FILTER
0052     4516   3367             DCA SAVEH
0053     4517   1044             TAD EXP
0054     4520   7012             RTR
0055     4521   7010             RAR              /MAKE
0056     4522   0365             AND MASK2        /FOCAL
0057     4523   1366             TAD M200         /EXPONENT
0060     4524   3044             DCA EXP
0061     4525   1045             TAD HORD
0062     4526   7010             RAR              /SHIFT
0063     4527   7012             RTR              /MANTISSA
0064     4530   7012             RTR
0065     4531   0371             AND MASK3
0066     4532   3370             DCA SAVEL
0067     4533   1045             TAD HORD
```

```
0070      4534   7012          RTR
0071      4535   7012          RTR
0072      4536   0365          AND MASK2
0073      4537   1367          TAD SAVEH
0074      4540   3045          DCA HORD
0075      4541   1046          TAD LORD
0076      4542   7012          RTR
0077      4543   7012          RTR
0100      4544   0365          AND MASK2
0101      4545   1370          TAD SAVEL
0102      4546   3046          DCA LORD
0103      4547   1372          TAD SFLAG       /NEG?
0104      4550   7640          SZA CLA
0105      4551   4451          JMS I MINSKI    /YES, NEGATE
0106      4552   1373          TAD POINT       /RESTORE FOCAL
0107      4553   3374          DCA CPOINT
0110      4554   1376          TAD SW1
0111      4555   3774          DCA I CPOINT
0112      4556   2374          ISZ CPOINT
0113      4557   1375          TAD SW1
0114      4560   3774          DCA I CPOINT
0115      4561   2374          ISZ CPOINT
0116      4562   3774          DCA I CPOINT
0117      4563   5536          JMP I EFUN3I
0120                     /
0121      4564   3400   MASK1,  3400
0122      4565   0377   MASK2,  0377
0123      4566   7600   M200,   -200
0124      4567   0000   SAVEH,  0
0125      4570   0000   SAVEL,  0
0126      4571   7400   MASK3,  7400
0127      4572   0000   SFLAG,  0
0130      4573   2022   POINT,  2022
0131      4574   0000   CPOINT, 0
0132      4575   6232   SW1,    6232
0133      4576   7000   SW2,    NOP
0134                     /
0135                           EFUN3I=136
0136                           EXP=44
0137                           HORD=45
0140                           LORD=46
0141                           MINSKI=51
0142                           EVAL=1613
0143                           GETC=4545
0144                           PUSHJ=4540
0145                           LISTAP-7
```

NO ERRORS

USER INPUT TO DEC SOFTWARE SURVEY
(Refer to article for background to these questions)

1. Which major line of development would you prefer DEC to pursue?

(   ) Operating System Development
(   ) Language Development

Comment:

2. If DEC elects to pursue Operating System Development which class of
    system would you prefer?

(   ) Extended "OS/8 type" single user development system
(   ) Extended "RTS-8 type" multi-tasking real time system.

Comment:

3. If, on the other hand, DEC pursues Language Development which would you
    prefer they develop?

(   ) RTS-8 FORTRAN IV
(   ) New BASIC
(   ) Other

Comment:

4. What characteristics would you want to see in such a language?

5. If DEC pursues Operating System Development could you accept a smaller
    number of supported configurations (i.e., less breadth for more depth)
    assuming users would still be able to write their own configuration depen-
    dent support. (This might mean distribution on a more limited set of media
    than at present).

(   ) Yes
(   ) No
Comment:
Name_____

Company_____