January   Number 20   1977

Contributions and correspondence should be sent to:

Robert Hassinger, Coordinator
12 Bit SIG
c/o DECUS   Liberty Mutual Research Center
146 Main Street   . . or . .   71 Frankland Road
Maynard, MA 01754   Hopkinton, MA 01748

DECUS/Europe contributions are solicited through:

Lars Palmer
DECUS/Europe 12 Bit SIG Newsletter Liaison
Hassle
Fack
S-431 20 MOLNDAL 1
SWEDEN

(Please include reference to Newsletter number and page when inquiring about
material published.)

NEWSLETTER DEADLINE

Deadline for ready-to-use material for the next Newsletter is February 25,
1977. Material requiring editing/re-typing must be in earlier. Ready-to-use
material should be prepared on 8½ x 11 inch white bond paper. A one inch mar-
gin should be maintained on both sides, on the top and on the bottom. Material
should be reasonably clean, legible, sufficiently dark copy for printing.
Materials prepared on electrostatic printers (e.g., Versatec printers, Xerox
machines, etc.) are often unsuitable for photographic reproduction.

International contributors please note that due to the Newsletter being printed
on U.S. sized paper (8½ x 11") please limit camera ready contributions to use
an area of 6½" x 9" (16.5 cm x 23 cm) approximately. Material on the larger
European size paper is difficult to fit in our format.

LAS VEGAS NOTES

A report on the Las Vegas Symposium from Tom McIntyre is included. There are
a few additional items I would like to note.

1. The program exchange project which Tom organized and managed impressed
    everyone and it and the demonstration of it for the Library Committee were

invaluable in getting the committee to recommend that DECUS go ahead
with acquisition of the PDP-12 for the Library. This progress is
particularly pleasing to me because I have been working for it for
about four years. It will give the Library facilities to expand its
services in the area of media conversion as well as expanding the "in-
house" reproduction capabilities for faster and more reliable program
reproduction. Tom omitted the fact that he or one of the other program
exchange committee members were on duty all night each night doing program
reproduction work for the attendees at the meeting. It was a tremendous
effort and everyone owes them a big Thank You.

Very special thanks are also due Maryann Oskirko who did a super job
gathering the financial support needed to bring the PDP-12 to the meet-
ing. She got contributions from the following DEC organizations:
Laboratory Data Product, PDC-8 Product Line, Traditional Products Line,
PDP-15 Product Line, Educational Products Group, Large Computer Group,
Large Business Systems Group, Small Business Products Group and System 11
OEM Marketing. You can't imagine how much it costs to send that machine to
a symposium!

2. We have been trying for some time to find a way for DEC to take a more
    active role in promoting and distributing a few of the most popular user
    written programs. I think real progress was made during the meeting in
    this area. There is still nothing definite but we are moving in the
    ⌐ direction. DEC recognizes that it does not have the resources to
    .e all the software that everyone needs. It is starting to believe
    that at least a few of the things our users have done rank or a par with
    what they do in terms of quality and utility. Finding a way to take better
    advantage of this work will benefit the users and it will help DEC through
    the availability of more comprehensive software for the machines they sell.
    The sort of thing we are looking at does not involve DEC in actually
    taking on the software as supported products because that is a very ex-
    pensive and troublesome thing in terms of DEC's support commitment. We
    are looking at DEC involvement in distribution and promotion however.
    If we succeed, the details will be in a future Newsletter.

3. There was a lot of informal discussion of the development of a scheme
    for extending the PDP-8 memory addressing capability beyond 32K. There
    were no formal announcements or commitments but don't be surprised if
    something on this line develops. The discussions were centered on a rather
    nice and fairly sophisticated approach. It would appear that if DEC offers
    a product it would probably be in the form of a new extended memory control
    option board for the Omnibus machines. A compatible option for the older
    machines looks very unlikely at this time.

4. For the first time a substantial contingent from the DEC Service organi-
    zations was active at the meeting. Software Services, Field Service, and
    Customer Spares were all well represented. They actively exchanged
    thoughts with the symposium attendees. Field Service ran a successful
    suite where everyone got a chance to discuss their problems with Field
    Service management. I anticipate this trend will continue and expand based
    on the initial reactions of the DEC people involved.

5. The last item should help in our program to draw in more of the OEM community by offering activities and services that address their particular needs. The OEM's represent a very large part of DEC's 12 Bit business and they have a lot of influence in DEC's decisions. In the past DECUS and the 12 BIT SIG have not served them as well as I would like. We are hoping to attract an expanded attendance from the OEM community at the next Symposium. Papers and other sessions of special interest to OEM's are particularly solicited. We (the SIG) will be working with the PDP-8 Product Line in this area in the coming months. If you are interested or have any ideas, Tom McIntyre and I can help you. Give us a call. The deadline for submissions is very soon so don't delay.

MACREL DEMONSTRATION AT LAS VEGAS

Papers on the MACREL assembler and its relocating linking loader were given at the Las Vegas meeting which clarified much of the planning and progress on them. The areas that are still vague due to the fact that they have not been finished yet included overlays and some of the other more exotic features. The current development version was demonstrated by assembling a set of RTS-8 modules and linking them to run an RTS-8 demonstration program. It was hard to appreciate MACREL's power from the demonstration but the important point was that MACREL really does exist and it really is up to the point where it seems to be able to do useful work. I think there is still some hard work left to be done to finish filling it out in time for the desired release schedule, but it looks as though the project is finally getting the attention and resources it needs to be successful.

FROM TOM McINTYRE

        The Las Vegas meeting was a very busy one for the 12 bit SIG. Our experiment in library management and distribution was continued using the TPL PDP-12. We gathered a total of 37 library files which filled 1.5 RK disk cartridges and contained offerings from BASIC games to RTS-8 user tasks. There was an attempt to formalize the distribution procedure by having users submit either material for distribution or requests for material which were then serviced by the "staff" in the late evening hours. We tried to keep the days open for handling submissions and running demos. The volunteer IO clerks were Jim Van Zee, Jim Crapuchettes, Jim Coryell, Tim Clarke and Tom McIntyre. We were still using the programs Doug Wrege wrote for the Fall 1974 meeting in San Diego. Although we didn't keep close track, I believe we distributed around 75 tape or diskette volumes in the two and a half days we were running. Several users also got RK05 copies of the full set of programs. The experiment was valuable in pinpointing the weaknesses of the system and suggesting areas for improvement. Tim Clarke has promised to work on GET and PUT (formerly DUMP and ꞋOAD) which turned out to have some bugs in them, and I am going to write a high core DECtape handler for the 12. We are looking at the possibility of incorporating a BASIC dialogue program in the system to ask the operator for file names and output devices and optimize the device packing. The BASIC program would generate a BATCH file which would supervise the actual copies, tape mounts, directory print outs and so forth. Anyone who would like to participate in these efforts should contact me so that the work can have some minimal coordination.

Partially as a result of our efforts, the DECUS board has voted to acquire the PDP-12 as the nucleus of a media exchange facility. To really make this work we need quite a bit of additional software. Right now we can support the OS/8 community pretty fully, but can offer only minimal media copying support for PDP-11 users and slightly more for the 18 and 36 bit machines. We use a pair of utilities at WVU to transfer files between our PDP-12 and PDP-11 that use a PDP-8 DECtape as the intermediate medium. The problem is that the tape really only holds one file. If anyone has a general utility that supports PDP-11 file structures on the PDP-12 using TC12F hardware or anything else we would like to hear from them.

Another idea that surfaced in the library committee was the notion of DECUS interns. The thought was that some projects could be handled by individuals who would work for the Summer at Maynard (or some other suitable facility) and receive a modest stipend to be provided by as yet unknown sources. We would be interested in knowing how much interest there would be in such a program if funds were found.

Our steering committee had a very frank discussion with the Executive Board on the subject of 12 bit representation on the Executive Board. The net result was that the SIG will not receive a position on the board nor will any other group except the DECsystem 10/20 users. The reasons for this decision are primarily historical but the session was valuable in any case since it brought to light the various problems which we perceive our SIG members to have. One problem is that the communication channel from the Executive Board to the SIG has not been functioning very effectively. The Board has requested that the SUG coordinator make regular timely reports to the SIG chairmen of the Board proceedings. Another outcome is that the Board will investigate the possibility of allocating more of DECUS' resources to the development of Local User Groups and other mechanisms of providing services to small installations whose users cannot afford the expense of the symposia. If we feel that the boards actions in these areas do not meet our needs our dissatisfaction can be communicated again. For the time being the steering committee felt that the Board was acting in good faith to meet our problems.

PDP8
more than 30,000 installed worldwide

digital

January 5, 1977

Mr. Bob Hassinger
Liberty Mutual Research Center
71 Frankland Road
Hopkinton, MA   01748

Dear Bob:

SUBJECT: Decus Review

All in all I think that we had a good Decus session in
Las Vegas. I was disappointed that the attendance wasn't
as good as we would have liked, but the group that we did
have (about 60) was very active.

The key message that we were trying to get across was that
the PDP-8 is an important product to Digital and is being
very actively developed. To some extent this is a "catchup"
year where a lot of the emphasis has gone to completing the
projects that had been previously mentioned. Specifically,
the MACREL assembler and DECNET/8 are now real and will be
released this Spring.

I hope our intention to concentrate on the further development
of RTS/8 was clearly understood. I am very happy to see that
several users took an active role in forming an RTS/8 User's
group. It is my understanding that Steve Root of our
Software Engineering Group and Lee Nichols have already begun
discussions, and that the inputs on the RTS/8 manual which users
made subsequent to Decus have been incorporated in the manual.
It is our goal to encourage such involvement in our PDP-8
software product development.

One of the frustrations which I have with Decus in general, and the PDP-8 section in specific, is that it is not a representative cross-section of our User base. Since the majority of PDP-8's are shipped to OEM's, Decus cannot represent the voice of the User's unless it has significant OEM representation. For Spring Decus I want to ensure that we make an explicit attempt to involve OEM Users in the meeting, and provide sessions which will be receptive to their needs.

I understand that Jamie Milne and Dave Rogers, of PDP-8 Product Line, are packing their skis for the Canadian Decus and are looking forward to meeting many PDP-8 owners who we rarely have contact with up in Canada. Jamie is becoming the coordinator for Decus in the PDP-8 group and will take a lead role in planning our involvement in the Spring and subsequent meetings.

Sincerely,

Gary M. Cole,
PDP-8 Product Development Manager

GMC/cac

PDP-8 Products, Parker Street, Maynard, Massachusetts 01754

FORMATION OF AN RTS/8 WORKING GROUP

Since the release of the RTS/8 Operating System, no special interest group has been formed. Almost all user communication has been at DECUS meetings. The recent Fall Symposium in Las Vegas provided several excellent workshops given by Digital, and user application sessions. But these meetings are only twice a year, and not all of the RTS/8 user community can attend.

The RTS/8 user community needs to improve communication, particularly with Digital's PDP-8 software development personnel. For this reason, the embryo of an RTS/8 Working Group was formed at the Fall DECUS Symposium. The purpose of this working group, made up of experienced RTS/8 users and Digital software personnel, is to provide a bidirectional path to foster communication of ideas, problems, needs, SPR's, new offerings, documentation, etc. Specifically, this group can provide:

- A focal point for input to guide the development and enhancement of the RTS/8 system.

- A forum to share ideas and software developments
  and to demonstrate the feasibility of user concepts
  worthy of implementation by Digital.

- Assistance in the creation, maintenance and
  distribution of usable system documentation.

- A group to critique specifications for new
  software and to field test software developed
  both by Digital and the user community.

The RTS/8 Working group will be a subset of the 12 Bit
SIG and use this newsletter for communication. Lee Nichols
organized the working group and volunteered to serve as its
coordinator. He also volunteered to collect and edit RTS/8
items for publication in this newsletter. You can send
ideas, problems, or "wish list" items to him. Lee's address
is:

> L. H. Nichols
> E. I. du Pont deNemours & Co.
> Experimental Station
> Building 357
> Wilmington, DE 19898

One area Lee would like to start work on is the
publication of SPR's for RTS/8 V2. You may have noticed
the absence of SPR reports for RTS/8 in the Digital
Software News. Since RTS/8 is a source release, users
can easily fix a problem and not take the time to document
it for the benefit of others. Please send a copy of SPR's
who have submitted or a note describing a problem (and
solution if possible) you have encountered to Lee. He
will add them to his own and submit them for the next
newsletter. If we can collect V2 SPR's now, we can
publish a list of the bugs which have not been fixed
in V2B when its released.

One of the system features missing from RTS/8 is a
task level ODT. Lee is currently generating the software
specifications for this and would appreciate ideas.
Looking ahead to MACREL this ODT will probably resemble
a subset of ODT-11 and should include features like
relocation registers.

WASHINGTON AREA 12 BIT LOCAL USERS GROUP

For those readers in the Maryland, Washington, DC and northern Virginia area
there is an informal PDP-8 & 12 users group that meets once every four or
five weeks, usually on a Tuesday, Wednesday or Thursday evening from 7:30 PM
until we get tired of talking. The members are from universities, medical/
health organizations and various federal government organizations. There are
no dues, by-laws, etc. At the meetings we discuss various topics such as:
software system, what bugs are where, whose not to buy and why, what works
with which; hardware such as I/O busses, A/D converters, tape and disk units,
extra memory, printers again with the nitty-gritty info on lemons, good deals,
etc. In fact, we cover just about anything that has to do with 8's and 12's.
We also discuss the types of programming, data collection systems, etc., we
have and use.

If you use an 8 or a 12 and if you would like to talk about the little mental
midgets (good, bad or indifferent) why not join us! For information on the
next meeting write or call: Edward Franzosa, Special Testing & Research
Lab, 7704 Old Springhouse Road, McLean, Virginia 22101.

NEW SOFTWARE RELEASES COMING

Current thinking at DEC regarding 12 Bit software releases in 1977 seems to
go as follows: Version 2B of RTS-8 is scheduled for release in April or
May, DECNET/8 Level 2 is also scheduled at the same time. There has to be a
new release of OS/8 (tentatively referred to as Version 4) before the end of
1977 because of the date problem, and MACREL should be released during 1977.

Details on the RTS-8 and DECNET/8 releases are included in a separate article.
These plans are quite firm, I think. The software is in good shape and in
fact it has been waiting for the PDP-11 DECNET projects to catch up to ensure
compatibility and it has also been waiting for improved documentation to be
written. The OS/8 and MACREL situation is less firm. There is now a strong
determination in DEC to get a release of MACREL out as soon as possible. As
a result, one possible sequence of events that has been suggested goes as
follows:

1. A "preliminary release" of a slightly incomplete version of MACREL could
   be made at an early date (i.e., maybe no overlays, etc.).

2. Late in the year OS/8 V4 could be released along with a more finalized
   version of MACREL.

I don't think this scenario or any other is definitely set yet, but it is my
current best guess about what is going to happen this year.

OS/8 V4 will be necessary for everyone concerned with the date continuing to
work. It will also be required for the full capabilities of MACREL (currently
the thinking about overlays seems '   ', ire certain new internal monitor
functions for full implementation). _her things that might be included in
V4 are presently in the planning stage. Considering the product development
cycle, however, there isn't much time for any major development projects. I
have not heard of any plans for significant enhancements to any of the existing
languages for this release.

RTS/8 VERSION V2B and DECNET/8

The V2B version of the RTS/8 will be released in April or May. While the primary reason for V2B is to support DECNET/8, some bugs have been fixed and a few new things have been added. The DECNET/8 package should also be released in April or May. DECNET/8 will be level 2 and is not compatible with current DECNET released. Level 2 DECNET for RT-11 is due to be released in the April-May time frame with RSX-11 versions to follow later.

Two new executive requests (ER's) and two new tasks have been added for V2B and two ER's have expanded functions. The RTS/8 Users Manual has been extensively revised and will be about 50% larger than version 2. A parameter for version number has been added to the task setup definitions (TASK, CUR, START, INIWT). When defined, VERS becomes the initial MQ value. System-wide support for the KL8A has been added. Power fail code has been added and/or corrected in device drivers, particularly the TTY handler. And the OS/8 support task can use the RTS/8 LPT handler instead of doing direct line printer output.

The new ER's added to V2B are RESCHD and WAITX, and are both in "advanced programming" category. RESCHD forces the RTS/8 executive to perform a reschedule looking for the highest priority runnable task. This ER is used if a user task has inhibited task switching (by setting TSWFLG to zero) to execute a lot of delicate code or to modify data in another task. Task switching should be inhibited rather than running with interrupts disabled for long periods of time to avoid losing interrupts such as the clock. The WAITX ER is similar to the WAITE ER, but WAITX requires that the specified event flag be posted. When multiple even flags are active within a task (such as the TTY handler), a WAITE for event flag A will be satisfied if event flag B is posted. WAITX solves this problem by allowing a task to run only after the specified event flag is posted.

The WAITM and RECEIVE ER's have been functionally expanded. A non-resident task using WAITM can now free its partition by putting a 4000 in the AC when the WAITM is evoked. The RECEIVE ER has been changed to allow returning to the calling task if no messages are pending. If AC bit 0 is a 1 when the RECEIVE ER is executed and no messages are in the queue, control is returned to the calling task with the AC equal to zero (rather than a CDF instruction to the field of a message).

The new tasks added for V2B are EXIT and NULL8A. The EXIT task is run by MCR when the EXIT command is requested. User tasks can send EXIT messages which are dequeued and processed when EXIT runs. The messages can specify the address of an EXIT subroutine or can request that the message be posted. This allows user tasks to close files, turn off outputs, etc. and prepare for an orderly shutdown before EXIT returns to OS/8. The NULL8A task displays a running counter from 1 to 9999 in the AC and from 1 to 7777 in the MQ. This display provides a relative index of the system activity.

NEW DECUS PROGRAM LIBRARY SUPERVISOR

Since Ferne Halley transferred out of DECUS this last spring, the Program Library has not had a full time supervisor. At the Las Vegas meeting we learned that Chuck Conley will be taking on this job. Old time 12 Bit users will remember Chuck as a long time friend of the 12 Bit world. His contributions to our area at DEC go back as far as I can remember. Chuck will also add a badly needed technical capability that DECUS has lacked.

SYK2PH: Two-Page, 16-Bit-Packing OS/8 System and

* Auxiliary Handlers for the Sykes 7100

and 7200 Floppy Disk

M. Peterson,[†] P.M. Holtham and I.M. Templeton

National Research Council of Canada,

Ottawa, Ontario, Canada

The standard OS/8 system and auxiliary handlers for the (unbuffered) Sykes 7100 series floppy disk are single-page handlers which store a 12-bit core word in two 8-bit disk bytes, thus wasting 25% of the available storage capacity. These new handlers overcome the storage problem and utilize the full 16-bit capability. The system handler also contains a special format bootstrap requiring only 13_{10} toggled instructions. Appropriate patches for FRTS (unnumbered and V4) and BASIC (V3.0 and V4A), which as supplied only recognize and treat correctly the TD8E 2-page system handler, are included in the listing. An extra program which converts the contents of an existing disk to the new format is also supplied.

* This is an independent 2-page handler with entry points for units 1 and 2, i.e., not co-resident with SYS

[†] no longer at this address.

PROGRAMS FROM PETER LEMPKIN

Peter says he is making some more DECUS submissions. He sent along copies of the documentation (very well done, by the way). The following are his abstracts: LPTSPL - lists up to 5 files specially formated on a line printer or other output device. The program is called through the CCL "PRINT" command using the Command Decoder. Formated output contains, on each page, a page header consisting of input file name, date, and page number followed by the file text with consecutive line numbers. Several Command Decoder and text-embedded switches are available for greater operating flexibility.

MAG1∅ - a PDP8e utility program which uses the Command Decoder to specify commands to manipulate and transfer files between MTA∅: and MTA1: and OS/8 devices. The magtape files have associated file headers permitting access of particular files by name. Using 9-track mode, it uses either the TC58 or TM8e magtape controllers for the TU2∅-1∅ drives. The TC58 may be used with an OS/8 system which is "built" for TM8e magtape devices. That is, MAG1∅ has its own magtape handler but uses the fact MTA∅: and MTA1: exist in the OS/8 configuration to permit the names of the drives to be specified.

Files are written in 9-track 8-bit byte odd parity, 800 BPI format. This format may be read by machines which can read IMP compatible byte mode (such as PDP-1∅ KL1∅).

A companion program MAG8E.SAI (written in SAIL) for the PDP-1∅ is available for performing the same functions as MAG1∅. Thus files may be transferred between the two machines via 9-track tape.

Peter also included information on his Buffer Memory Monitor System for Interactivw Image Processing. This is a large program package with over 70 operations available. There are 8 buffer memories implemented with each storing two 8 bit 256 x 256 pixel images which may be accessed on the order of less than 1 microsecond per pixel on his system. This seems to be one of the bigger projects implemented on a 12 bit machine. (These days everyone seems to think they need a bigger machine to do things like this.)

NOTE FROM JOHN YOUNGQUIST

John wrote to say he has been working on a project to add logical IF operators (i.e., .EQ., .NE., .LT., .GT., .LE., .GE.) to OS/8 FORTRAN II. At the moment the modified version is in "field test". He plans to make this version of the compiler available for a nominal copying charge when testing is complete.

He has documented a bug that has shown up in the distributed FORTRAN II compiler involving bad compilation of GOTO's that reference line numbers that occur following the GOTO. A short example is enclosed.

John also included a copy of a flyer from Dewar Information Systems that includes a patch to OS/8 V3C to modify the Keyboard Monitor and Command Decoder to erase the last character on the screen when a rubout is typed (nice when the system console is a CRT). The flyer also has information on current and planned development of DISC's proprietory software. DISC has not sent any information directly to me about any of this so I assume they do not want it published. Their address is: 622 Forest Avenue, River Forest, IL 60305. John's address is: Verus Instruments Inc., Box 122, Fort Erie, Ontario.

NOTE FROM JIM VAN ZEE

Jim's latest U/W-FOCAL newsletter came recently with more news. He now provides a way to get rid of all the 'IOF' instructions so you can be sure the interrupt is always on. He has added an easy way to trap calls to "internal handlers" (like TTY:) and has built in code to check for the PLTR: and LPT: handlers. He has a simple internal line printer handler for the "LP$V" type printer (not the "L645" type). With this you no longer have to use the OS/8 output file to run the line printer. It can be saved for some sort of file output.

Jim's "big news" is the availability of DISPLAY ROUTINES to use the scope (on a PDP-12 or LAB 8e for example) for doing program development. The Output Scope command will send all output (including error messages) to the internal scope handler. Since this is so much faster and nicer it is the assumed mode when you load this version. Up to 85 characters are displayed on each line using Tim Clark's compression scheme. Rubout actually erases the character from the screen, control-L clears the screen anytime and the screen scrolls to show the last 10 to 20 lines. A VIEW command is also provided to plot points on the scope. 12 to 16 K is needed for this feature to be really useful. The 8K version does still do useful work, though.

Jim says he has built stand alone (non-OS/8) paper tape versions of U/W FOCAL. Not as nice without the file I/O but still a big improvement over FOCAL '69 and upward compatible with the full U/W FOCAL.

If the version of U/W FOCAL you are using came from DECUS it is out of date. You might want to inquire of Jim about what basis he will make the latest version available on. In the past he has asked for a modest contribution to help support the thesis work he has been doing due to a severe shortage of research funds. It may be well worth it to keep up and get the best configuration for your system. Jim's address is: Dept. of Chemistry, BG-10, University of Washington, Seattle, Washington 98195.

NOTE FROM DAVE M. BROCKMAN

Dave sent flyers on some OS/8 programs that he is marketing for 8080 microprocessor software development under OS/8. CALOS-80 is an 8080 Cross Assembler that runs under OS/8 which uses the standard Intel mnemonics. SIMOS-80 is an interactive 8080 simulation which runs on a PDP-8 under OS/8. The full 8080 architecture is simulated and all instructions may be executed. Input/Output operations are simulated by printing the device number and requesting/printing data. An emulation of the terminal is also available. The prices are very modest by DEC standards and are in keeping with the trends towards low cost per copy for micro computer software. Sources are available. The address is: FBE Research Company, P.O. Box 68234, Seattle, Washington 98168.

A note from Ernst Lopes Cardozo, Laboratory of Physiology, Vondellaan 2
Utrecht, The Netherlands.

AT LAST!

Finally the super Memory Management Unit for the PDP8 is here!
This piece of ingenious hardware replaces and extends the Memory Extension
Control (KM8E or KM8-AA) and provides the capabilities necessary to run
foreground/background systems like RTS-8 and MULTI8 at full speed. It may
be considered a major extension to the PDP8 architecture.
The new unit consists of two modules that plug into the Omnibus replacing
the memory extension control. It provides two extra functions, dynamic
instruction trapping en dynamic field relocation.
The general problem with foreground/background systems like RTS-8, MULTI8 and
ETOS is that programs executing in the background are seriously slowed down
by the fact that all memory extension instructions (e.g. CDF, CIF, RDF, etc.)
have to be emulated by monitor software. This slow down depends on the way
the program is coded but may be more than a factor 10 for some Fortran II
programs. This is caused by the fact that all IOT's are inevitably trapped
by the existing timesharing hardware.
Dynamic instruction trapping means that certain IOT's (certain device codes)
may be trapped or 'untrapped' under software control. Monitor software may
instruct the Memory Management Unit not to trap CDF, CIF and CDI instructions
to certain fields (the field number is part of the device code). For instance,
to run an 8K OS/8 system in the background, the fields 0 and 1 should be made
accessible. So the background program may execute CDF 0 en CDF 10 (CIF, CDI)
instructions at full speed, without any emulation overhead. However, if the
background program executes a CDF 30, this instruction is trapped and monitor
software may generate an appropriate error message. In no way the background
program can escape from its assigned memory area. Of course this alone is not
enough, as it would require the background program to run in the real fields
0 and 1. Now the field relocation logic comes into play. By appropriate
instructions the monitor software can set the Memory Management Unit to map
all field addresses generated by the background program onto any real field
in the machine. So if the program references data or instructions in field 0,
the actual machine address accessed may be in field 5 or any other. In this
way the virtual 32K address space of the background program may be mapped
on any real addresses. This mapping is only effective when the processor is
in User Mode.
Currently an adapted version of the MULTI8 realtime foreground/timesharing
background system is available that makes use of the new unit. An adapted
version of the OS/8 support task of RTS8 will be made available too.
The device has been developed in cooperation between the Laboratory of
Physiology of the University of Utrecht and DIGICOS b.v., a small Dutch
systems house. The latter will produce the device, with first deliveries
scheduled for April 1977. Further details can be obtained from DIGICOS,
P.O. Box 24, Ouderkerk a/d Amstel, The Netherlands.

SNOBOL-8.2

Bill Lenon had a copy of SNOBOL-8.2 at the Las Vegas meeting. The author is
Fred Dalrymple I believe. It is an improper subset of SNOBOL-3 for the PDP-8
under OS/8. The restrictions from SNOBOL-3 involve an absence of functions
and extended arithmetic statements. On the other hand, SNOBOL-8.2 is ex-
tended by allowing in-line assembly language code (similar to the "S" fea-
ture in OS/8 FORTRAN II) and access to OS/8 for opening, closing, and manipu-
lating OS/8 files.

The compiler generates PAL8 source code that you run through PAL8 and ABSLDR
to get a runable program. Hopefully this could result in relatively efficient
run time code. Due to the refurbishing of my 8 I have not had a chance to
try the system out yet but it looks very interesting from the write-up. I
don't know what the current plans are for making it available outside of the
program exchange system at the DECUS/US Symposia.

NOTE FROM GEOFFREY CHASE OSB

A couple of programs for dealing with crashed directories are discussed:

1. REWDIR - In DECUS, he says, with an improved version now in existence
   He uses it to create system DECtape directories on his RK system so back
   up tapes can have a 12 K system head on them for booting a system when
   the system on the disk is crashed.

2. SAVDIR (not in DECUS - he will submit it if people want it). Does an
   image save of the directory blocks. Syntax is like PIP/Y in that if no
   file name is given the directory blocks are assumed while a file name
   specifies a file with one of these directory images in it.

   Notes on COBOL-8 (sold by EDUComp). This dialect is (a) a subset,
   (b) somewhat slow, and (c) somewhat limited. He has made the following
   improvements:

   a) Relax user interface restrictions (i.e., 9 input files rather
      than just one, etc.

   b) Speed up compiler by a factor of 5.

   c) Simplify use and improve error messages.

   d) Speed up and debug run-time support.

His address is: Portsmouth Abbey School, Portsmouth, RI 02871

LOW COST PAPER TAPE READER

DEC finally announced the PRS∅1 Serial Paper Tape Reader. There were accessory
bulletins available at the Las Vegas meeting in the Traditional Products booth.
It is a small portable unit that will plug in series with any terminal that has
the standard DEC Mate-N-Lok connectors. There is a version for 300 band and

another for 2400 band. This way no additional interface is needed. I don't have the exact price, but I have heard it is in the area of $700. If you have a system with a DECwriter or DECscope as the terminal rather than a teletype, you probably have no way to read paper tape. This could be a cheap way to solve that problem.

NOTE FROM ERIC KIRTLAND OLSON

Eric suggests we formalize a "wish book" section in the Newsletter to help get users needs together with what is being or has been done. He asks for the following:

1. A routine, called at bootstrap time, which does the LOGIN operations of asking for a name, password, and whatever. Also, a LOGOUT program which re-initializes the "need to login" state. The LOGIN program should suppress printing of the password, at least. It should also not let the user by until he gives a valid login. Ex:

   (Bootstrap)

   (Calls in LOGIN)

   LOGIN:ERIC (CR)
   PASSWORD: (suppressed) GUEST
   BAD.
   LOGIN: ↑ C
   LOGIN FIRST.
   LOGIN:ERIC
   PASSWORD:OLSON
   THANK YOU.

(Two partial solutions could be the LOGIN facility in DEC System-8 (available from DECUS) and Vernon Blackmore's LOG which should be submitted to DECUS soon. - R.H.)

2. Two terminals are available at my school, and we frequently run OS/8. Does anyone know of a way of running it Time-Shared with two terminals?

(4 user FOCAL may still be available from DEC but I don't know of an OS/8 version, one of DEC's EDU systems has been set up to run multi-user BASIC with limited OS/8 file support, and ETOS from EDUComp might be interesting except that Eric does not have a disk, only single DECtape, card reader, 2 teletypes, 16 K and no EAE. How about more information and ideas from our readers. - R.H.)

3. A card reader handler that reads EDU-15 Batch-Basic cards would be nice.

(Would the MSBAT (mark sense Batch) program that seems to be on some or all V3 (or is it V3C only?) tapes be useful? Almost nothing is generally known about this program by the general using public. - R.H.)

Eric's address is Bolton Road, Harvard, MA 01451.

FROM BRUCE A. ERNST

"I am running RTS-8 on a lab PDP-8e. The clock handler for the DK8-EP does not contain any method of handling the Schmitt triggers. I would like any information from anyone who has written a clock handler that does take into account the Schmitt triggers. The clock will be used to construct histograms from data acquired during experiments with single neuron cells". His address is: Southern Illinois University, School of Medicine, 611 North Klein, Springfield, IL 62702.

NOTE FROM GILBERT R. BOHUSLAV

"Brazosport College currently has a PDP 8/E which is running under OS/8.

"We use the system for both time-sharing and process control applications. Time-sharing is run in BASIC under EDU 25.

"I am looking for someone who (1) has developed some type of accounting system which would allow me to determine computer users and (2) has hooked up to EDU 25 through a remote terminal through BELL 103A modem.

"If anyone has information pertaining to the above points please have them contact me." The address is: Brazo Sport College, 500 College Drive, Lake Jackson, Texas 77566.

NOTE FROM RUDI STANGE

Ref.: SIG Newsletter No. 19, November 76 - OS/8 Date

I have read your tractate on the date handling scheme of OS/8, and i was really a bit surprised for this never worried me at all. The reasons are as follows:

1. OS/8 first appeared 1971, in other words, there is probably nobody who has kept OS/8 programs prior to 1972, and most of these will carry a much later date.

2. Until now OS/8 has masked the bits representing the year (or deducted 1970 from the repsective year). All that has to be done from 1978 on is to use the same process, mask off the repective bits, say 000 for 1978, or 001 for 1979 etc. and the reverse operation adds 270 (ASCII) to those bits which in return gives 1978 etc.

3. All programs with dates as early as 1972 will then come out as 1980 (if there are any). Later ones will be presented with a date of the future recognizeable at once, and all that has to be done to correct this is to deduct 8 years from it; that is, if any one really cares to do that. In all cases where the date is vital, the respective program itself can be patched to print out its creation date.

4. The more one writes about this subject, the more it can become a problem. Generally speaking, what is really so vital about old program dates? They generally sit on tapes or floppies with the old monitor version anyway and therefore the dates will remain.

5. What worries me is the presentation of the date the way it is done in the states. "MM,DD,YY,". Other countries do it different, i.e. DD,MM,YY. I would like to see provisions for this, not _nly with OS/8 but with all operating systems.

Rudi Stange, Sales Support, Munich

Comments on Rudi Stange's Note Regarding the OS/8 DATE (RH)

1. I, for one, have many 1971 files as well as many more files from 1972, etc. I would like to be able to retain as much information as possible about their creation dates when I print new directories of them and when I use programs like MEDIAI and MEDIAO to generate master directories of my entire library. Files of the same name and different creation dates will appear and I want to be able to distinguish the older versions from the newer ones reliably.

2. There may be a little more to it than Rudi describes, but I agree that it is obviously possible to code a change in the base year used by each program as it interprets the dates in directories or the current system date. Unfortunately every program must be re-coded or patched every time the base year is changed.

3. Here is the underlying difficulty in all these schemes. How does a given program know what "today's date" is so it can tell if a particular directory entry is a "date of the future"? Unless every program that accesses the date is patched regularly with the current information the only obvious way to get it is to have the monitor capture it and supply it the programs through a USR call or in bits somewhere in the monitor - most likely in the core resident part.

I think it is better for the monitor to capture the current date and supply it to programs because then we will only have to modify our programs once and from then on the system will work with no more patches every four years or one year, or whatever. I think the required coding is really not significantly greater to do it right once and for all.

4. Rudi is really saying that you can infer what base year to use with a given directory by other entries in that directory (i.e., you infer the base year for a directory entry by using a version of PIP saved on that particular tape to print that tape's directory). This ignores the problem of wanting to retain the correct date when a file is copied to another tape or disk. This approach also does not solve the case of a program like MEDIAI which in one run may look at directories of all ages.

5. I think OS/8 DIRECT has implemented DEC's company wide solution to the question of unambiguous presentation of dates worldwide (i.e., 27-JAN-77). OS/8 PIP still uses the US format (1/27/77) for directory listings but there is little reason to use PIP for that purpose now. The main place that the US format remains a problem is in the DATE command. DEC has already modified many of its other operating systems (i.e. RT-11) to use the new standard format in such places and perhaps they will put it in a future release of OS/8 also. In any case if it is important a user can go to CCL and put in his own version of the DATE command.

NOTES FROM TOM McINTYRE IN RESPONSE TO ITEMS FROM
THE NOVEMBER NEWSLETTER

Bill Lennon at Northwestern Tech Institute has the EDGRIN software or at least did two years ago. He may be the only one who ever bought it. With respect to Brian's problem of E and F stops on the PDP-12, he is probably aware that if you depress STOP and press continue the current instruction finishes and you can fully use the console exam and deposit switches. I agree however that Clayton should blush at the design. Having also worked a little on our 11/20 which tells you essentially nothing I am always glad to get back to the 12. On Chase Amblers problem with hard copy devices, we have had quite a bit of experience in this area. This copy was produced on a DIABLO HITYPE model I using a parallel interface which is not generally available. The model I's in terminal configurations are around and appearing in the used market. I saw one advertised in the last issue of Computerworld. The only drawback of the daisy printers like the HITYPE is a shortage of character fonts. The last time I checked, six months ago, italic and Greek symbol fonts were not available. Oddly enough the correspondence terminal is generally considered the most versatile terminal device. This is because standard office selectric type balls will work on the correspondence terminal whereas they will not on an EBCDIC 2741 terminal. These type balls give the widest variety of type fonts and the best print quality at somewhat reduced speed. It is my understanding that John Alderman of Digital Communications Associates in Atlanta has a handler for the 2741 that uses a table look up to translate ASCII to EBCDIC codes. It would not be much challenge to change the table for correspondence codes since there is a one to one equivalence. It does get a little hairy if you want to use the device both as a keyboard and as a

printer since the translation table takes about c page, but the high core handler approach could work in this application.

On the subject of handlers, I have been thinking about our conversations on high core resident handlers and have some ideas I would like to have some feedback on. I have been thinking of implementing a single one or two page handler called maybe HI: that would load a specific size core load from a specified area of the system device into high core and then go to it as an extended handler. The general mechanism would be that any devices that required or wanted this level of service would exist on the system as save images placed at the beginning of the system device so they would not be moved about by squashes. To use a particular handler to do a copy the procedure would be for the user:

```
.R DTA
.COPY HI:<DSK:*.PA/F
```

The .R DTA would cause the file DTA.SV to run which would locate the handler HI: in the system area, look up itself, patch the device control word table and store the DTA version of the HI: handler in the handler block of the system. The copy of the HI handler could either be part of the DTA save image or the copy already in the system area could be patched. In one case an extra read is required in the other the save image is one block longer. The .SV part could also size core and set software core size to protect itself from getting clobbered once loaded. A further problem arises if one wants to use PIP to zero or squash the HI device. In this case the image of PIP must be patched to reflect the current size of HI. Of course no one should use PIP for these purposes since DECsystem 8 squash and zero are much better. I expect to use this mechanism to implement the software version of the TC12F DECtape handler and another handler to read RK05 disk packs on a brand X controller. A problem with our system is that we have run out of device slots and need to conserve the number of handlers on the system. I don't see any real problems to the above approach unless one wanted to use two different HI handlers at the same time. In that case you might configure HI as a two entry point handler and have some devices become HI1 and others HI2.

NOTES FROM I.M. TEMPLETON

Enclosed are an article on "Patching RESORC Tables" and another on "Separating Segments from a FORTRAN II Library File". He says, "I found the LIB8 separation operation very useful in modifying Bill Kaufman's EAE LIB8 which has a very long UTILITY (11_8 blocks, 15 core pages) stored 4th in sequence, and thus makes for rather inefficient loading (I had a case in which UTILITY loaded in field 1 leaving 10 pages unused in field $\emptyset$). Not having any tape drives I didn't need his UTILITY anyway, so I built my own LIB8 in correctly descending order of size using just the EAE version of FLOAT and INTEGR and getting the rest from the original LIB8".

(Note: DECUS 12-48 contains a version of LOADER that will search two library files and which will permit multiple occurrences of the same module name in the library file. In this way the library can contain optimum size modules depending on how much is to be loaded. For example: FOO1 by itself needs 2 pages and a second 2 page module - FOO2 may sometimes be needed with it, thus normally using 4 pages. If FOO1 and FOO2 can be combined into a 3 page module then both the combined version and the FOO1 version can be in the library. By careful arrangement the correct version will load thus saving space - I thought 12-48 was also available on DECtape but I cannot find the listing any more) - RH)

The second note includes the following:

"RE the Friedemann Brauer note on p. 6 of #19 Newsletter, we have now submitted to DECUS a package for which I enclose a copy of the abstract.

"Further to my recent remarks about the Kaufman EAE LIB8 package, I have found a couple of restrictions: (1) the EAE is used exclusively in mode A, and no SWBA is performed on entry. Any previous use of mode B, and any use of mode B in SABR subroutines or inserts must be cancelled with a SWBA before starting or continuing in Fortran. (2) The use of a 23-bit divide can cause a noticeable loss of precision in the 6th decimal place. I have found little increase of speed compared with an original LIB8 in which the old FMP entry has been disabled and the Parker version (Decus 8-615) added.

"P.S. I have just found a serious error in my FORLIB 'SYNC' extension, p. 22 of Newsletter #18. I omitted the 'W' immediately before the R,11∅. Without this, of course, none of the preceding changes will be written into the disk file."

FROM STANLEY RABINOWITZ

REPLY TO NOTE ON .DEassign COMMAND

   In the last issue of this newsletter, Jim Van Zee complained that the .DEassign command does not work without CCL. I cannot understand this comment. I have tried .DEassign both under OS/8 V3 and V3C and it works properly when CCL has been disabled.
   DEassign is a keyboard monitor (KBM) command. When CCL is enabled (using the .R CCL command), CCL comes in and removes DE from the KBM's list of legal commands. When CCL is disabled (via the .CCL command), the DE command is restored to the KBM's repertoire of commands. Perhaps Mr. Van Zee merely removed CCL.SV from his system device and neglected to disable it via the .CCL command.

INITIAL COMMAND EXECUTION ON SYSTEM STARTUP

Here is a patch to the OS/8 monitor which might be
of interest to some users. (This is not a DEC-supported
patch.) This patch allows a user to enter a CCL command
into the 'C' remembering area (using the .UC CCL command)
and causes the system to execute this command whenever the
system is bootstrapped.

```
.R EPIC                     xxxx/6046↓
*SYS:</1$                   xxxx/5710↓
R,11                        xxxx/404↓
0,0                         xxxx/303
0000/325                    0,175
0,77                        xxxx/7677
xxxx/6211↓                  W
xxxx/1251↓                  R,0
xxxx/3775↓                  0,77
xxxx/6201↓                  0400/1077
xxxx/1311↓                  E
xxxx/3201↓                  *↑C
xxxx/3202↓                  .
```

Restrictions: Only CCL commands (and not KBM commands)
may be saved in the 'C' remembering area. (This restriction
will probably be lifted in the next release of OS/8.)
There is no easy way for a user to find out what command
is currently in the 'C' area. A much better scheme would
be to have the user enter his initial command into a file
called INIT.CM which would then be executed upon system
startup, if present. This is not easy to do now, but may
very well be a feature of the next release of OS/8.

NOTE FROM FRANK DIETER LEHMANN

Maybe the following proposition is useful for FOCAL users.

If you use OMSI-FOCAL and U/W-FOCAL there will be troubles because user-
programs for these two FOCAL-versions are not compatible.

Trying to run a program for OMSI with U/W-FOCAL or vice versa results in hang-
ing up FOCAL in a loop. One can no longer control FOCAL from the TTY or VT50.

Instead of bookkeeping by hand you may mark the programs used with U/W-FOCAL
by giving them another extension.

This was done with a very small patch to U/W-FOCAL.

Change locations
00134 / 0603 0625
07061 / 7001 1374

From now on U/W-FOCAL lists, calls, runs and saves user-programs with the
assumed extension: FU.

FROM I.M. TEMPLETON

Patching RESORC Tables

The 4-word 'TYPE' tables in field 0 of RESORC are dealt
with on p. B-7 of the OS/8 Software Support Manual. The third word is
the negative of the device size in blocks and the fourth word refers
in some way (has anyone unscrambled this?) to the 'KIND' table.
There is also a 'NAME' table in field 1 (14435-14474) which may be
modified to provide a match for the cryptic numeric labels which RESORC
give to user devices. The table contains 13 pairs of 6+6 - bit ASCII
names terminated by 0000 (this stops further testing for a match).
Beyond this point there is a dummy pair of names reading TE,ST,LI,ST
followed by a second 0000. All of the list is available for modification
- e.g. CDR, CDP, DEV, OUT, INP - and TEST LIST provide two spare slots.
A table of 'KIND' resides at 14506-14553 - I don't understand this one
yet!

Separating Segments from a Fortran II Library File

The individual modules in a library file produced by LIBSET
(e.g. LIB8.RL) cannot be added to or replaced as is the case for LIBRA
files, but must be separated and resubmitted to LIBSET. This separation
may be achieved relatively simply as follows. Pages A-7 and 8 of the
OS/8 Software Support Manual describe the library directory block
which may be examined with EPIC to determine the relative block numbers
corresponding to the individual entry points of the file. In the case
of LIB8, these may be correlated with the subroutines and entry point
names listed in Table 7-5 of the OS/8 handbook. In the 1974 version of

LIB8 we find IOH at relative block 1, FLOAT at 10, UTILTY at 15, POWER

at 20, TRIG at 23, ATAN at 25, LPOWRS at 27, INTEGR at 30, SQRT at 32,

RWTAPE at 33, and IOPEN at 34.   We can now delete the LIB8.RL

directory entry and, with PIP, insert in its place the entries for the

individual files thus:

```
. R PIP
* SYS:<SYS:/S/0         (to fill any empties)
* LIB8.RL</D
* L8INDX[1]</I=1
* IOH.RL[7]</I=7
* FLOAT.RL[5]</I=5
* UTILTY.RL[3]</I=3
* POWERS.RL[3]</I=3
* TRIG.RL[2]</I=2
* ATAN.RL[2]</I=2
* !POWRS.RL[1]</I=1
* INTEGR.RL[2]</I=2
* SQRT.RL[1]</I=1
* RWTAPE.RL[1]</I=1
* IOPEN.RL[1]</I=1
```

LIB8.RL can then be recreated via LIBSET to include any new or modified

modules.

NOTE FROM BRIAN CONVERSE

Brian wrote to tell about a custom "Collimation" keyboard he has been using
that he is very pleased with. It patches into an ASR-33 teletype look with
an opto-isolator. His keyboard was ordered with additional clusters of keys
including: an adding machine pad, cursor moving set coded for SCROLL and
another set coded for LAP6W. He got a special arrangement where the alpha-
betical keys are normally upper case and the shift key goes to lower case!
With all this plus a power supply and custom case the cost was around $600
which he thinks might sound high, but have you ever priced DEC's stand alone
keyboard for the VT-11! He expects to let us know more about how to order
these keyboards in a couple of months after he gets more experience with it.

He says that in reference to his complaint about the missing entry points for
LTA4 through 7 he did not realize that they were actually all in the same
handler. He is still having trouble though because he runs out of slots while
building his system and does ɔ understand what is happening. The explana-
tion goes something like th' ɔS/8 holds each handler in a reserved block
in the system area. Due to limited number of handler blocks there is a
limit on the number of separate handlers that may be configured into the system
at the same time. Each handler is allowed to have multiple entry points so
one handler can handle multiple logical units (i.e., the LINCtape handler has
entry points for at least LTA∅ through LTA3 available). Due to the size of
certain system tables and a limited number of bits for identifying handlers
there is also a limit on the total number of entry points (i.e., logical
units) that can be active in a system. You could have 15 entry points all
in just two or three handlers for example. RESORC/E will show where you
stand on both these limitations. Another limitation can be that BUILD has
a limit on the size of its tables and buffer space for holding handlers.
Sometimes you have to remove a handler (not just de-activate it) to make room
for another.

FROM ALAN CLEARY
<u>   </u>

THE UNIVERSITY OF NEWCASTLE UPON TYNE

HEAD OF DEPARTMENT
Professor M. Hammerton

DEPARTMENT OF PSYCHOLOGY
RIDLEY BUILDING
CLAREMONT PLACE
NEWCASTLE UPON TYNE NE1 7RU
Telephone 28511 STD Code 0632

Lars Palmer,
DECUS/EUROPE 12 BIT SIG
   Newsletter Coordinator,
Hassle,
Fack,
S-431 20 Molndal 1,
Sweden.

15th November 1976.

Dear Lars,

        We have a VT50 on one of our OS/8 systems and in the absence of
any known DEC support I have devised patches and software to use
some of the nice features of the terminal. They may be of interest to
other OS/8 users having this terminal:-

(i) I have patches to the monitor and command decoder which
     implement correct deletion from the screen (except for tabs)
     instead of the usual backslash and echo. It is also a good
     idea to filter out CTRL/Q and CTRL/S which are used for
     output control, as these can accidentally get into a command
     line. Patches are appended.

(ii)   I have written a two page handler for the VT50, based on the
       KL8E handler. On output it sets the VT50 into scroll mode
       allowing the SCROLL and SHIFT/SCROLL keys to control the
       display of one additional line and a complete new screen
       respectively. On input the handler keeps the VT50 in normal
       mode. It deletes correctly from the screen, displaying tabs
       as underscore to ensure that deletion will always work.

(iii) For many years we have used West Virginia University's
       SCROLL editor on our 8E with a VC8E refreshed scope, and
       on our LINC-8. I have now modified it to display on the VT50
       terminal. For those unfamiliar with SCROLL, it is a TECO-
       like editor which operates on a complete file, allowing both
       forward and backward searches. It also has auxilliary input
       and output capabilities. The display on the scope shows the
       current state of the lines around the cursor. It has proved
       very popular with our users. The VT50 version allows input
       translation to lower case and also displays the octal value of
       the character to the left of the cursor.

       One of these days I hope to get around to implementing VT50 deletion
for BASIC and the OS/8 cusps. If anyone has done this, it would be useful
to have details. I will supply copies of my work to anyone who sends a
formatted Dectape or Linctape.

       Regarding the VT50 problem reported in Newsletter 18, I think that
it is based on a misunderstanding of the switch controls on the base of
the VT50. I agree that all OS/8 programs should be written to force the
8th ASCII bit, but meanwhile you can make the VT50 hardware force
the 8th bit. The parity slide switch on the base of the VT50 is marked
NONE and EVEN. In the NONE position it should force the 8th bit.
I suspect that people who have experienced problems have this switch
in the EVEN position. The VT50 user's manual, however, does say
that it is possible to rewire the terminal to give a space in the 8th bit.
It is therefore possible that the switch is in the correct position, but
someone has been at the internal wiring.

                                        Yours sincerely,

                                        Alan Cleary.

PATCHES TO OS/8 SYSTEM FOR VT50

.R FUTIL

```
7.1224/ 7403   7555
0007.01225\ 1302   1207
0007.01226\ 7402   7557
0007.01227\ 1302   1207


7.1263/ 2020   7000
0007.01264\ 5247
0007.01265\ 1070   210
0007.01266\ 4423   240


7.1307/ 1013   1265
0007.01310\ 1071   4423
0007.01311\ 7650   1013
0007.01312\ 5300   1071
0007.01313\ 1070   7650
0007.01314\ 2020   5201
0007.01315\ 4423   1266
0007.01316\ 7240   4423
0007.01317\ 3020   1265
0007.01320\ 1013   4423
0007.01321\ 3040   7000
0007.01322\ 1440   7000
0007.01323\ 4423   7000
WRITE


51.1033/ 7403   7555
0051.01034\ 5321   5207
0051.01035\ 7402   755?
0051.01036\ 5321   5207


51.1135/ 1102   1360
0051.01136\ 2024   7000


51.1142/ 1015   1361
0051.01143\ 3020   4466
0051.01144\ 1420   1360


51.1151/ 2024   1360
0051.01152\ 5300   4466
0051.01153\ 1102   5202
0051.01154\ 5277   7000
0051.01155\ 0000
0051.01156\ 2024   7000
0051.01157\ 5362
0051.01160\ 1102   210
0051.01161\ 4466   240
WRITE


^C
```

KEYBOARD MONITOR

VT50 always gives 233 for altmode, so replace 376 and 375 support (for teletypes) by ignore CTRL/Q and CTRL/S.

The rest changes the support of deletion, replacing backslash and echo of the deleted character by backspace, space, backspace.

COMMAND DECODER

CTRL/Q and CTRL/S are implemented in a similar manner to the monitor patches above.

Deletion changed similarly to the monitor patches above.

FROM DAN SMITH
       Eye Research Institute
       20 Staniford St.
       Boston, MA 02114   617 742-3140 x. 260

WARNING TO RALF BIT-PICKERS:
LEFT SHIFTS MAY BE HAZARDOUS TO YOUR HEALTH

       Some of us have already discovered that right shifts (via
ALN) leave the last bit shifted out in an overflow register (AC1),
from which a left shift can retrieve it; thus, for example, shifting
right N bits and then shifting left N bits masks off, not the lowest
N bits, but the lowest N-1.
       There is, however, a further complication. There seem to be
various ways of introducing garbage into AC1. For example, certain
formatted I/O operations leave garbage in AC2, which can be transferred
to AC1 by a floating add. At present, then, left shifts via ALN
must be assumed to have an undefined effect unless the contents of AC1
are explicitly established by a preceding instruction.
       There are several alternatives:
       1)   Instead of shifting left N, use a pair of ALN's to
            shift right one and then shift left N+1. This will
            have the desired effect. However, if the one-bit save
            is a bug rather than a subtle feature (nobody seems to
            know) the program will bomb later if FRTS is ever fixed.
       2)   FSTA the number, FCLA, shift right to clear AC1, FLDA
            the number back, then shift it. This should work and
            be revision-proof. Takes a few cycles, though.
       3)   Use a multiply instead.
       4)   Do bit-picking via 8-mode traps, not RALF.
       5)   Fix FRTS.
       It may have occurred to somebody that "leaving garbage in AC2"
might be undesirable for reasons other than left shifts. This is true.
I have documented and reported a reproducible case where a formatted
I/O operation causes a last-bit error in a subsequent floating add.
The case is a FORTRAN program, and the add is an "integer" add. The
effect is that I=2+2 sets I to a quantity which prints as 4, but is
actually not quite .EQ. 4   .

FLIST (advertisement)
       I've written a tiny program to call PASS3 of F4, so as to get a
listing with ISN's without recompiling. I've submitted it to DECUS,
but will supply copies to anyone in a hurry. There's a trick to calling
pass 3; .R PASS3 won't work, nor, apparently, will chaining.

SPR's

Jim Van Zee has recently submitted the following SPR's:

1. "DIRECT: This problem was reported to me by Tim Clark of FRELAN Associates; he may also be submitting a SPR. The difficulty is that if the output file created by DIRECT has either 1 or 2 characters beyond the end of a block (for instance, 385 or 386 characters) the ENDCHK routine fails since it looks only at OUWDCT which is only incremented on the 3rd character. The result is that the formfeed and CTRL/Z do not get sent to the output. - Those being the last 2 characters. In version 05 (the user improved version submitted to DECUS) the formfeed can be omitted, so in that case the LF at the end of the last line would also be omitted.

   "Well, so what? This doesn't seem very serious until you consider what happens if the output is to a file. Then since DIRECT always closes the file with a multiple of 5 blocks one may have up to 4 blocks of real 'garbage' which nicely appears when you TYPE or LIST this file later on. Also if you use PIP to condense it to its real size you may easily encounter a problem with a 'line too long' because of the omission of the EOF.

   "Anyway the fix is straight-forward enough. I suggest initializing RPOS to RPOS-1 rather than RPOS1 and then adding the test in OLOOP to check if RPOS=RPOS-1. The first change isn't really necessary, but it saves a literal (possibly)."

2. "PIP: The /Y option in PIP cannot transfer a system-area file back to the system head when the file is at or near the end of a device. My specific problem occurred with a file which happened to be 1 block from the end of a LINCtape, i.e., there was 1 free block after the file. (This was happenstance, but I usually do put the system-area files near the end since they are big and seldom used.) Clearly the problem is that PIP is specifying too many blocks to read and then ignoring the extra ones when it writes (at least it does the write correctly!). In this case simply moving the file to another tape (with FOTP) allowed it to be transferred back to the system area (so there was nothing intrinsically wrong with the file), and of course PIP did write it out properly in the first place."

Jim also submitted an SPR on CREF producing junk at the end of a listing but it seems too complex to include here. He notes that the answer to his problem with the new version of PAL8 not handing as many symbols as previous versions was due to a subtle change that DEC slipped in. The /K option used to mean "use extra core". In the new version of PAL8 it now means "don't keep the USR in core". In the past an 8K system never needed the /K option but now if you want the maximum symbol table space on an 8K system (or any size system) you must use /K to recover the space otherwise used for USR. Jim and I agree that when this sort of change is made the documentation should feature it in terms of change from the previous way of doing something rather than simply updating the documentation to reflect the change. In the latter case, everyone has to read every word of the new documentation like a Philadelphia lawyer looking for the occasional critical change. A good way to handle this is often to use "change bars" in the margins to point out where changes have been made. Unfortunately the 12 Bit software documentation has not been organized enough for this to work in the past, though.

FROM JOHN YOUNGQUIST

```
C          FORTRAN TEST PROGRAM
C
           GOTO 10
           I=1
10         GOTO 20
20         I=2
           END
```

```
                /        FORTRAN  TEST  PROGRAM
                /
                /        GOTO 10
0204  5777      JMP \10              <<<<< NOTE BAD ASSEMBLY!!!
                /        I=1
0205  7001      IAC
0206  3200      DCA \I
                /10      GOTO 20   <<<<< NOTE STATEMENT IGNORED!!!
                /20      I=2
0207  1376      \20, TAD (2
0210  3200      DCA \I
                /        END
0211  4033      CALL 0,EXIT
0212  0003 06
0213  0000      [0, BLOCK 2
0214  0000
0376  0002
                END
```

```
EXIT      0000EXT
MAIN      0201EXT
OPEN      0000EXT
[0        0213
\I        0200
\10       0000UNDF            <<<<< NOTE UNDEFINED!!!
\20       0207
```

The following fixes this problem in FORT.SV

```
13133/ 5177 5371       /JMP FIX

13171/ XXXX 2024       /ISZ LINEPT      /BUMP POINTER
13172/ XXXX 7000       /NOP             /PROTECT SKIP
13173/ XXXX 4776       /JMS DMPLIN      /DUMP THE COMPILED CODE
13174/ XXXX 3024       /DCA LINEPT      /SET EMPTY FLAG
13175/ XXXX 5177       /JMP START       /BACK TO WORK
13176/ XXXX 1554       /DMPLIN          /POINTER TO DUMP ROUTINE
```

The following letter was sent in response to letters from Andrew Short and Chase Ambler. To date no response has been received.

November 9, 1976

Mr. William H. Munson
Software Product Manager
DEC
146 Main Street
Maynard, MA 01754

Dear Mr. Munson:

As Chairman of the DECUS 12-BIT Special Interest Group and editor of the 12-BIT Newsletter I hear from many users about problems they are having with DEC. A current and continuing problem is DEC's policy (or more accurately, lack of policy) regarding 12-bit software maintenance. I am writing to you because you responded to Mr. Norman Dotti's complaints in this area in June. I am enclosing copies of two letters I have recently received which demonstrate that a serious problem continues to exist in this area. My own experience confirms that in spite of statements by DEC to the contrary, no comprehensive policy has as yet been implemented with respect to the maintenance and updating of OS/8 and other 12-bit software.

I, as a user of the software, have never received any notice of a policy. In my frequent contacts with DEC I have never been able to elicit a firm policy statement. All this in spite of the fact that a full year has passed since the announcement of the software maintenance scheme at last Fall's DECUS Symposium.

I would like to have a firm statement of policy specifically with respect to OS/8 and other 12-bit software in time to be able to read it to my membership at this Fall's Symposium the first week of December. I also intend to disseminate your response through the 12 BIT Newsletter to all our members. The deadline for the next issue is the end of December.

Very truly yours,

Robert Hassinger
Senior Research Associate

CORRECTED ADDRESS FOR VERNON BLACKMORE

The address given in the last Newsletter should be corrected to 18 Brookfold Road rather than Brookfield.