



DECUS 12 BIT SPECIAL INTEREST GROUP  
NEWSLETTER

Fall Number 41 1982

Contributions and correspondence should be sent to:

Robert Hassinger, Coordinator - 12 Bit SIG  
c/o DECUS MR2-3/E55 ..or.. Liberty Mutual Research Center  
One Iron Way 71 Frankland Road  
Marlboro, MA 01752 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 SUBMISSIONS

Submissions are accepted at all times and are normally used in the next issue to go to press regardless of date of receipt. Ready-to-use material should use an area 7 inches (18 cm) wide by no more than 9 inches (23 cm) long on each page. It should be single spaced on white bond paper whenever possible and must be reasonably clean, legible and sufficiently dark for good photographic reproduction.

Material submitted in machine readable form is particularly desirable because it can be edited and incorporated into the newsletter format more easily. Higher quality reproduction is also possible this way. Contact the editor (Bob Hassinger) for further details on acceptable media and formats if you plan to make a submission in machine readable form.

HOW ABOUT THAT

Another Newsletter. Amazing! The reports of the death of the Newsletter are exaggerated, as you can see. However, I do not have the time I once had to write a large fraction of the Newsletter. As a result, I can only get an issue out when I get enough contributions from you. A flurry of activity recently in the area of PASCAL for the 8 family and a review covering the DECmates (which I had hoped to write ever since the last issue but never had time) finally got this issue off the ground.

Since the last Newsletter the SIG has continued to participate in DECUS leadership activities such as the various committees. Symposium activity has dropped off due to lack of submissions. We need help organizing sessions for the Symposia. Give me a call!

Copyright © Digital Equipment Corporation 1982  
All Rights Reserved

It is assumed that all articles submitted to the editor of this newsletter are with the authors' permission to publish in any DECUS publication. The articles are the responsibility of the authors and, therefore, DECUS, Digital Equipment Corporation, and the editor assume no responsibility or liability for articles or information appearing in the document. The views herein expressed are those of the authors and do not necessarily express the views of DECUS or Digital Equipment Corporation.

DECUS 12 BIT SPECIAL INTEREST GROUP NEWSLETTER  
Number 41 - Fall 1982

PAGE 2

SIG COMMITTEES AND WORKING GROUPS

Steering Committee:

Robert Hassinger - address above - (617) 435-3452

Jim Crapuchettes  
Omnex Corp.  
801 E. Charleston Rd. Suite F  
Palo Alto, Calif. 94303  
(415) 494-3170

Lee Nichols  
E. I. Du Pont  
Experimental Station  
Building 357  
Wilmington, DE 19898  
(302) 772-3839

Lawrence H. Eisenberg  
17141 Nance Street  
Encino, California 91316  
(213) 788-0354

RTS-8 Working Group  
Lee Nichols - see above

WPS-8 Working Group  
Open

COS-310 Working Group  
Lawrence H. Eisenberg - see above

PDP-8 DIGITAL SOFTWARE NEWS

Recently, while talking to my local DEC software service manager, I learned that DEC had finally resolved the issue of the PDP-8 Digital Software News. Many readers will never have heard of this publication because it has not been available to most of us for a long time. DSN was the equivalent of the Software Dispatch, etc. in the PDP-11 world. Software problems and patches were published in it each month or two.

Up until a few years ago, purchasers of OS/8, etc. were put on the DSN mailing list. You automatically got copies of DSN without a special charge. This was in contrast to the PDP-11 world where software support services always cost considerable amounts. Eventually, the cost of this service got to be too much for DEC. Sales of 12 bit systems and software could not continue to support the free service. At that point DSN stopped being sent to most of us, and no means was provided to deal with the problem. Over the past several years the 12 Bit SIG leadership has worked with various DEC representatives to try to resolve the problem but progress came very slowly.

Now we learn indirectly that the issue seems to be resolved. Our DEC Software service contact tells me that DEC order number QF097-2Z gets you twelve months of DSN for \$185.

Wally Kalinowski wrote to say he had discovered the same information. He says that it took him many calls to get information on DSN from his DEC office and that it finally took his DEC contact 14 more calls internally to find out the details and confirm the order number. When Wally wrote, it had been quite some time since his order went in and he had not received any issues of DSN as yet. He thinks this may have been due to an address mix up. Based on my experience with RT-11 and RSX-11M this seems very likely. DEC has never gotten all of our software support addresses and contacts right at the same time, ever! I have learned that every time they make a

"correction" the situation can be expected to get worse. In any case, the renewed availability of DSN could be a very welcome development. If you order it, let me know how you make out.

#### PASCAL OS/8 FIELD TESTS

I recently learned that the U of M version of PASCAL for the PDP-8 family was finally up and working. Since then I have had a limited opportunity to try it on our PDP-8/I and our DECstation. Although I am not a PASCAL expert, it seems to run as expected on our systems. The documented features are very impressive. They have gone to a great deal of effort to make their system all OS/8 family systems (version 3 or later). For example they have made a determined effort to not rely on any specialized hardware, they are trying to be sure they can run on the older pre-Omnibus machines, EAE is not required, etc, and while a minimum of 20K memory is needed for compilation, programs will then run in as little as 12K memory (for example the test programs I have tried run on our 16K DECstation). This is accomplish with run time system overlays and PASCAL program separate code segments.

Wally Kalinowski sent the following report: "Our installation has been testing the University of Minnesota's very impressive implementation of PASCAL. The compiler compiles the source program into an intermediate P-code. The run time system then interprets the P-code. Some of the features of this PASCAL are:

- (1) It has it's own handlers, but can also use OS8 handlers.
- (2) I/O is device independent.
- (3) It will use the FPP if available (similar to FRTS).
- (4) It supports a segmented virtual code space, eliminating the need for overlays.
- (5) The debugging features are superb - postmortem dumps indicating exactly where the problem occurred and showing the values of the variables at the time of the failure.
- (6) It compiles essentially all full blown PASCAL.

"Compilation is a little slow on our floppy based system, even using Jim Van Zee's floppy handlers which are considerably faster than the stock handlers supplied by DEC. All in all, this is one of the most impressive programs ever made available for the PDP/8."

John Easton at U of M indicates that they anticipate making their PASCAL available in the near future for a moderate fee through SSRFC. His address is Social Science Research Facilities Center, 25 Blegen Hall, 269 19th Avenue South, Minneapolis, MN 55455.

#### PASCAL-S

A version of PASCAL called PASCAL-S has been circulating primarily in Europe for some time. It has not become well know in the US, partly because most of the documentation is in German and it has not been submitted to the DECUS Program Library. Jim van Zee passed on a copy of what he got when he was in Europe some time back. It appears to be a compile and go system although I suspect it would be possible to document how to save the compiled image.

Wally Kalinowski included the following in his note about the U of M PASCAL:  
"This is a very useful subset of PASCAL. The compiler is an implementation in PAL of

a compiler written by N. Worth. It is an extremely fast compiler but does not have all the features of the U of M system. We really get a kick out of compiling .SV files because of the far out error messages which are all in German."

If someone could help a little with documentation in English, I am sure many users would be interested PASCAL-S. Maybe someone can help identify what, if anything needs to be done to make it run the pre-8/e machines. If possible, I would like to encourage the submission of this program to the DECUS Program Library to make it more available (also, I find that frequently after someone sends out the first five or ten copies of a program, they start to get tired of it and wish someone like DECUS could help with the reproduction work).

The following are some excerpts from the English documentation I have:

#### NOTES ON PASCAL-S COMPILER FOR PDP-8/E

#### COMPILER/INTERPRETER LIMITS:

-----  
512 IDENTIFIERS  
63 ARRAYS  
63 BLOCKS  
1980 STATEMENTS OF INTERMEDIATE CODE  
16 LEVELS  
8 CHARACTER VALID IN IDENTIFIERS  
80 CHAR'S/LINE MAXIMUM FOR COMPILER INPUT (NOT PROTECTED!)

#### DIFFERENCES TO "WIRTH'S" PASCAL-S:

-----  
MAXINT = 2<sup>35</sup> - 1 = 34359738367  
REALS BETWEEN 2.78E-309 AND 8.98E+307, PRECISION ABOUT 5.0E-11

MAX. ARRAY-BOUNDS  
MAX. CASE-ITEMS : -2048 < N < 2048

EOF AND EOLN W I T H O U T (INPUT)

#### ADDITIONAL PREDEFINED PROCEDURES

HALT  
ASCII(ARG1, ..., ARG1) - OUTPUTS 8 BIT ASCII CODES  
RANDOM - FUNCTION - RETURNS RANDOM NUMBERS BETWEEN 0 AND 1

NO LINE-SPACING CONTROL CHARACTERS PROVIDED!  
USE SPECIAL PREDEFINED PROCEDURE ASCII

OUTPUT LINE LENGTH NOT LIMITED (USERS RESPONSIBILITY!)

PASCAL-S for the PDP-8/e was written by:

Prof. Heinz Stegbauer  
HTL - Modling  
Austria

and has been distributed by the German 12-BIT SIG. For more information please contact:

Wolfgang Leber  
Chairman, German 12-BIT SIG  
Max Planck Institute fur Hirnforschung  
Deutschordenstrasse 46  
D-6000 Frankfurt am Main 71  
West Germany

HARDWARE REVIEW FROM JIM VAN ZEE

During the past two years (since the last issue of the Newsletter), both Digital and other vendors have produced a number of new '8' products. Several issues ago there was a list of manufacturers of '6100' systems (systems built with the Intersil IM6100 microprocessor), however many of those companies have since 'disappeared' (does anyone out there know, for instance, what happened to TLF?), and new ones have subsequently come along.

In Germany the 'EURO-12' machine has been available for some time, and is constantly being improved. It is built around the 'Eurocard' format, and seems to have almost every conceivable feature that a programmer or hardware designer could wish for - the ODT in panel memory, for example is simply OTW: type 'H' and you get a 'help' slide showing how to use all the features! The panel display shows both binary and octal, and some models have an 'extender' socket on the top for super-easy prototype work. The system was demonstrated running at a clock rate of 9MHz (3-5MHz is typical for 6100 systems), which allows double-density disk copies to be performed successfully. The standard system device is a DEC-compatible floppy (DSD-440), but other options may be available. For more information, contact Wolfgang Leber at: Twelve-bit Systems, Am Dorne 14a, 6105 Ober-Ramstadt, West Germany.

(The 6100 processor is actually quite popular in Europe, and a number of different firms have designed it into their products, for instance PRAXICO in Wuppertal (West Germany) has a small system with a 2MB dual Remex floppy-disk drive (that's 2200 OS/8 blocks per diskette!). Unfortunately my information on this, and many other systems, is too incomplete to provide any kind of review. Perhaps the next issue of the Newsletter will have additional information. I know, for instance, that a number of 6100 systems have been built at Universities. It would be fun to have a list of all the 'home-brew' people.)

Here in the USA the PC/M company seems to have more-or-less captured the non-DEC 6100 market. They recently announced their new 'OMEGA' series which comes in a handsomely-designed box that both 'looks' and 'feels' almost exactly like a 'real' PDP-8e. The conscientious effort which has been put into making this machine '100%' DEC-compatible is quite obvious, and the level of documentation that comes with it is surpassed only by Hewlett Packard! The OMEGA system provides the programmer with a 'modern' PDP8 architecture, using all the latest technology, instead of one designed a decade ago. It has its own bus structure, with dozens of 'standard' interfaces available. Like the EURO-12, the OMEGA also offers a very advanced panel-memory ODT which makes program debugging an absolute breeze, with features such as trapping jumps outside of a defined region, program tracing (which prints the contents of the internal registers after each instruction), etc. PC/M offers a number of different

interfaces for DEC-compatible mass-storage devices, such as the RK05 disk and the RX02 floppy, as well as 'DEC-compatible' A/D boards, parallel I/O modules, etc. For more information, contact Bob Nelson at Pacific Cyber, Metrix, Inc., 6800 Sierra Court, San Ramon, CA 94566 (Tel: (415) 829-8700).

DEC, of course, has meanwhile been mass-producing the 6100-based VT/WT/78 DECstation series. Only a very few of these have gone to OEM's for hardware-software development due to the limited provisions for debugging (i.e. NONE, aside from the OS/8 ODT facility). This machine was essentially designed for the WPS (word-processing) market, from which it is now emerging, quite often at very attractive prices. The limitation to 16K, as well as the speed of the processor, will certainly restrict any subsequent usage, but as an example of what might be done with a VT/78, Eagle Research Corp. in St. Albans, West Virginia has developed a multi-channel A/D converter that can be plugged into the parallel I/O (printer) port on the back, and a rather nice version of FOCAL was also developed for this machine.

The VT/78 was replaced a year ago by the VT/278. Outwardly the obvious difference is that one is in a 'VT52' box, while the other is in a 'VT100' box. But inside the VT/278 ('DECmate') is a COMPLETELY NEW microprocessor! Let no one think the 12-bit era is over - DEC has actually created a NEW 'PDP8' processor, which they designated the '6120'. Since the 6120 has not been commercially available except as part of the '278, it's future may (unfortunately) be somewhat limited. DEC has again aimed the VT/278 toward the office market, but with some extremely interesting features which were made possible by the new processor.

For starters, the 6120 has a stack! Not just one stack, but two! The stack instructions use IOT's in the 'CDF' group, and hence would create problems for software designed for the KT8A. Even more interesting is the fact that the 6120 has two complete 32K address ranges: one for 'normal' memory, and one for 'panel' memory. The DECmate makes extensive use of this feature in implementing (most of) the functions of a VT100 terminal with 'panel code'. Field 6 of 'panel memory', for example, contains the screen buffer: put a character there and it will be displayed on the screen. The 12-bit word size works out just perfectly, allowing any of 256 different characters to be displayed, each with a possible attribute of 'reverse', 'bold', 'underline' or 'blink', according to whatever 4 most-significant bits are set.

In Field 0 of panel memory are the power-up diagnostics, the bootstrap, routines for emulating the normal TTY instructions (including a 'type-ahead' buffer!), and even a screen-dump facility. Field 1 contains additional code for handling the foreign keyboards and alternate character sets which can be supplied for this product - i.e. German, French, Dutch or whatever.

One of the special features of the 6120 is that panel memory routines can be easily called from main memory via one of 4 'trap' instructions. Thus in the VT/278 one can print a character on the lineprinter from any field of main memory with a simple two-word call. Since the lineprinters available for this machine are of the 'serial' type (requiring monitoring of XON/XOFF, rather than simply waiting for a hardware flag), this feature saves both lots of memory space as well as development time. There are other routines for displaying data in the keyboard LED's, changing the screen brightness, etc.

Software-wise, the 6120 is ALMOST totally compatible with other 12-bit processors. Besides the new stack instructions, there is also a new 'R3L' instruction

which rotates the AC left 3 positions WITHOUT the link. This is extremely handy for ASCII-to-octal conversions, or for things like computed CDF's, and of course one can also do a 'IAC R3L' to get the constant '10'. The 6120 is a much faster processor than the 6100 (for one thing 'internal' IOT's, such as CDF's, are 5-6 times faster than they are in a 6100), with an overall factor of about 3.8 being found between the DECstation and DECmate.

Exact timing comparisons are somewhat difficult to make, however, since the processor is shared by the program in main memory and the code in panel memory which is running the display. As one example of this effect, a particular benchmark was observed to run about 5% slower when it was started by typing 'go' than when it was started by typing 'GO'. This effect was totally repeatable, but eluded understanding for a long time until it was realized that the keyboard scan routine took more time when the 'caps lock' function was not engaged! The keyboard is scanned, and the screen updated, on every refresh (i.e. 50/60 times per second). The manual states that no more than 1.5 milliseconds will be spent in this operation (and no doubt it is usually far less), but obviously in the worst case the terminal functions could consume about 10% of the CPU resources.

Unfortunately DEC 'messed-up' in one crucial area of the design: the terminal I/O instructions (device codes '03/04') are not exactly compatible with other 8's. Since terminal I/O is one of the most fundamental parts of almost every piece of software, this discrepancy causes great havoc. Mind you, the VT/278 versions are 'close', they aren't really close enough! The 'KCF' instruction, for example, now SETS the keyboard flag, instead of CLEARING it, and the 'KSF' instruction not only skips on the flag, but clears it too. The 'KCC' and 'TCF' instructions both clear the AC, but not their respective flags, etc. Some of these differences might have been unavoidable due to the 6121 I/O controller (also new) which was used, but others seem to have been simply the result of insufficient understanding of the role played by various instructions in existing software. For instance, every single OS/8 handler required a rewrite in order to work on the DECmate because the standard CTRL/C check expects the keyboard flag to remain set after a KSF instruction (allowing the monitor to type out 'C'). On the '278 this instruction sequence locks up the terminal since the flag disappears as soon as the 'KSF' skips, but the character is not cleared from the FIFO. Then since the flag is gone, no other routine knows about it, so the character just stays somewhere clogging up the system until one goes into 'setup mode', which seems to clear things up.

The VT/278 may have a very short lifetime. As most everyone presumably knows, DEC just recently announced the new 'DECmate II' processor - one of a trilogy of new 'personal computer' products - which also employs the 6120 processor. Not much is known about this machine at the moment, other than that it will be the first '8' from DEC to have mini-floppies and that it is again aimed totally at the office market. Supposedly it can have a Z80 processor in addition to the 6120, so it can run CP/M programs. But the rumor mill has it that the DECmate II will probably not be running OS/8 software.

#### DECUS PROGRAM LIBRARY NEWS

The library has decided that in light of the low volume, it is becoming too costly to continue to support paper tape. They have stopped accepting new submissions on paper tape. They plan to discontinue distributing paper tapes of items already in the library fairly soon. I would be interested in hearing from anyone who has a

problem with this policy.

The new library catalogs are out. Since there were only a few new PDP-8 family entries this year, an addendum to the last catalog is being distributed to save the cost of a whole new printing and mailing. Several are particularly interesting interesting to me. Their submission is the result of a several years of efforts by the SIG to get them into the Library were they would be available to every one. The sort package is a key element in any data processing task. In conjunction with the newer versions of BASIC that have been distributed with OS/78, you have good tools to do many data processing jobs. I have used the combination to build several very successful applications here on our DECstation 78. On larger configurations with bigger, faster disks, very good applications are possible.

The media management package has been a favorite of mine for years because of the many hundreds of DECTapes we have. Without the package I could never find the files I need (some of those tapes are 11 or 12 years old now).

Many of us have been waiting a long time to gain access to Dick Murphy's WPFLOP program. It does conversions between WPS-8 and OS/8 files. With it you can have the best of both the WPS and the OS/8 worlds. For example, you can use the tools mentioned above to manipulate, sort and process files that you maintain and process using the WPS editor, list processing and communications features.

Rosemary Williams gave a paper on WPFLOP at the Fall 1981 DECUS Symposium. It was published in the proceedings on pages 609 thru 613.

Incidentally, another of Dick Murphy's programs has been made available also. RTFLOP is a program to do file conversions between OS/8 files and RT-11 files on floppy disks. Again this can be a very valuable tool. It is included in the OS/78 version 4 distribution kit available from DEC.

DECUS 8-924 - WVU Media Management Package This is a package of programs to aid in the cataloging and management of the contents of large numbers of floppy disks, DECTapes, LINCtapes and other media. The programs in the package are MEDIAI, FIND and MEDIAO.

MEDIAI reads the directories of the media. Each disk or tape is identified by a volume number. A file containing the information from the directories and associated volume numbers is created. Directories of addition media and updates of existing directories may be placed in the file from time to time.

FIND will do an on-line search of the file created by MEDIAI, outputting information on each instance of selected files. Multiple files may be selected in a single pass and wild card specifications may be used.

MEDIAO will create a formatted cross-reference directory sorted by file name or by extension. A simple listing by filename and volume number can be produced. A full listing with file length and date as well as name and volume number is also available. SORT (see DECUS 8-925) must be used to order the master directory data before using MEDIAO.

The package runs on OS/8, OS/12 and OS/78. It has not been tested with OS/78 version 4 and it may require the usual small changes to work on the VT278 DECmate I.

Media (service charge codes): Write-up (AA), DECTape (HA), Floppy Diskette (KA).  
Format: OS/8.

DECUS 8-925 - Sort-Merge Utilities This is a package of programs for sorting OS/8 ASCII files. SORT is the principle utility. MERGE and XTRACT are companion programs to assist in efficient sorting of large data sets.

SORT is designed to work on OS/8 compatible ASCII files. They are sorted by "record"s of up to 512 characters. A record may be defined as consisting of "n" lines or as all the characters up to an arbitrary record mark character. The user has the option to define fields for sorting either by fixed column positions or bounded by arbitrary delimiting characters. The sorting can be either ascending or descending, character or numeric within each field. A total of 32 keys can be defined by columns or 42 by delimiter. The sorting procedure used is multi-pass sort-merge with intermediate temporary files. The devices for the files may be specified to optimize the sorting. The original intent of the design was to be able to sort effectively even on a minimum system with as little as two DECTapes.

MERGE will merge two previously sorted input files into one output file. The same field definitions as for SORT are used.

XTRACT can be used to reduce the size of a data set before it is sorted. Records are selected from the input file and passed to an output file based on whether the value of a field is inside or outside of a specified range of values.

The package runs on OS/8, OS/12 and OS/78. It has not been tested with OS/78 version 4 and it may require the usual small changes to work on the VT278 DECMate I.

Media (service charge codes): Write-up (AA), DECTape (HA), Floppy Diskette (KB).  
Format: OS/8.

DECUS 8-926 - WPS-8 TO OS/78 File Conversion Utility WPFL0P is used to transfer documents from word processing floppy disks to OS/8 media or from OS/8 media to word processing diskettes. The WPS floppy is accessed using the COS compatible floppy handlers which are included in the package.

A WPS-8 Data List (generated by list processing to drop all field names) can be converted to OS/8 (i.e. OS/78) format to be read as input for a data processing program running under OS/8.

An OS/8 data processing output file record can be coded within the program to contain enclosed field names (<field names>) for conversion to WPS-8 for further list processing.

An OCR captured text file or telecommunicated text file received from a host computer can be converted to OS/8 for imbedding word wrap controls.

Note that complete sources are not available with this submission. All references to OS/8 refer equally to OS/78.

Media (service charge codes): Write-up (AA), DECTape (HA), Floppy Diskette (KA).  
Format: OS/8.

FOCAL NEWS FROM JIM VAN ZEE

"Jim van Zee reports that he is finally putting the finishing touches on his new LAB-FOCAL (known as 'LDF' for short), which he views as a successor to his much-used U/W-FOCAL. LDF is, however, rather different from previous versions of UWF, due to the inclusion of many new features (see the list below), and the desire to improve or expand the syntax. For more information contact Jim at Lab Data Systems, 10320 Ravenna Avenue N.E., Seattle, WA 98125.

Features included in LDF-V5

- \* Supports the full ASCII character set with U/L-case commands, etc.
- \* Uses 7-bit characters, rather than 8-bit (so 'A' = '65', not '193')
- \* Built-in error messages - no searching for the list of error codes
- \* Up to 4 simultaneous files (1 output+3 input, or 4 input/FRA files)
- \* Program (.FP) files are compatible with FOCALPLUS, but not with UWF
- \* Requires minimum of 16K - symbol table expands up to 32K (3400 var)
- \* Variable names abbreviated to FIRST and LAST char: AuntMary = 'Ay'
- \* 'Local' variables (automatic subscripting using current Group Num.)
- \* Arithmetic operators may be used as command delimiters: IF(), Type-1
- \* All four sets of 'enclosures' may be used: '()', '[ ]', '<>' and '{}'
- \* IF/ON commands can specify enumerated branches: IF(expr)-,0,1,2,3,4
- \* UNTIL command for 'loop on condition', plus 'relational expressions'
- \* The 'X' command used to 'nop' (temporarily disable) another command
- \* Variables may be 'set' in ASK commands, '=' key used to retain value
- \* Negative format option (%-xx) added (left-justify + suppress spaces)
- \* Subroutine calls save (and later restore) the current format setting
- \* '.' operator performs 'expression-to-ASCII' conversion: TYPE 7=bell
- \* '\$' operator starts an 'escape sequence' (clear screen = Type \$H\$J)
- \* Escape sequences may contain variable parameters: T \$[ row+1; col H"
- \* Two-letter commands for 'L'(libr),'I'(input),'O'(output), 'U' (user)
- \* 'Load and GO', 'Load and DO' commands replace 'Lib Run', 'Lib Gosub'
- \* Directory commands now generate multiple-column output (with '='opt)
- \* File/buffer option added to the OPEN commands: Open Input/3 filespec
- \* Expanded FRA function can use any file, has automatic initialization
- \* Input File/Output File; Input Terminal/Output Terminal command pairs
- \* Output Lineprinter command available; also Output Scope, Output Plot
- \* Internal lineprinter handler for serial (LA120) or parallel printers
- \* Input/Output Buffer commands allow use of buffers as a 'scratch pad'
- \* New handler management routine optimizes memory usage - bigger stack
- \* New multi-tasking features: clock (or event)-driven subroutine calls
- \* New FQUE function handles 12-level task scheduling; 24-bit FAND fnct
- \* Two free pages left in Field 1 for user-written code (but it was 4!)"

OMNI-8 MULTIUSER OPERATING SYSTEM

Recently I received some information from Steven J. Peschke on the OMNI-8 operating system. It will allow a PDP-8 to support multiple terminals, multi-tasking for up to 32 jobs, extended memory up to 256K bytes, and the RLO1/RLO2 disk. Each user can run OS/8-OS/78 or COS-310. Steven says that a PDP-8/E can out perform a comparably equipped 11/03, 11/23, 11/34 class PDP-11 system in "most" applications. For more information, contact Steven at Network-Systems Design, Inc. 404 North Main St., Suite 808, Oshkosh, WI 54901 (414)231-3333.

MULTIS NEWS FROM JIM VAN ZEE

"The MULTIS real-time multi-user/multi-tasking operating system, which has been in use for many years in Europe, is now also available in the USA. The price is reportedly quite reasonable. The system has a very sophisticated memory-management board which does all the 'field-mapping' in hardware (unlike the RTS-8 system). This board also eliminates the 'CIF emulation' problem and permits the use of A/D converters, 'scopes and other special interfaces in the background WITHOUT having to write a foreground support task. Spooling services are provided for both a lineprinter and a plotter. Jim van Zee is fielding inquiries about the system, so for more information please get in touch with him at Lab Data Systems, 10320 Ravenna Avenue N.E., Seattle, Washington 98125."

COSMOS-8

In the last few Newsletters there has been a considerable interest in small 8 based systems. DEC is not active in this market and I can not find anyone in the company who shows any prospects of becoming interested in the future. The only 8 related market that looks big enough for DEC to take an interest is the Word Processing area. The small 8 compatible systems are viable enough however to keep a number of small companies active, and most important, that activity is helping to support continued software development work that benefits the entire 12 bit user community. The following is extracted from material Jim Butch sent recently. It is included here as an excellent example of these points. I would like to see someone from DEC recognize the potential in this area.

"The Eagle Research Corporation CosMOS-8 is a 6100 based, CMOS microcomputer system. The CosMOS-8 is targeted for industrial and scientific applications including data acquisition, data reduction, interactive plotting, and process control. The CosMOS-8 features a complete line of peripherals and software to support these applications.

"The CosMOS-8 is available in three configurations: an OEM component set, a very compact table-top system, and a NEMA-12 packaged industrial system.

"The OEM component set consists of one or more of the following system components:

- 1) SBC-6100 An all-CMOS single board computer including:
- 1) 6100 CPU
  - 2) 6102 MEDIC

- 3) 6103 PIO
- 4) 2K CMOS EPROM and 1K CMOS RAM (in either control panel or user memory space)
- 5) Two serial I/O ports
- 6) User 6101 PIE
- 7) Emulation of any 6000 or 7000 instruction in control panel routines is possible
- 8) Provisions for indirect addressing in control panel memory
- 9) OS/8 compatible (with optional boot PROMs)
- 10) Buffered expansion bus
- 11) Small size: 6" x 8"

- 2) 16KS 16K CMOS memory board with a bank switching feature for memory systems of up to 128K, and optional battery backup
- 3) IADC 12-bit, 8-channel, low level, integrating A/D converter board
- 4) 3SIO 3-port serial I/O board with software programmable baud rates
- 5) GPIOC Multi-purpose interface board with three 6101 I/O controllers; models available include up to three of the following options:
  - 1) Serial I/O port
  - 2) Multi-mode, 4-digit counter
  - 3) Parallel input and output port
  - 4) Two 10-bit D/A converters
  - 5) 10-bit, high speed A/D converter
  - 6) Prototype area
- 6) DCDC DC-DC converter board for system power
- 7) TU58-BB Tape cartridge mass storage system
- 8) MDISK Intelligent dual 5 1/4" floppy disk storage system, 200KB per drive
- 9) KDSW 32-key keyboard, 32-character LDC display, and switch register board
- 10) MODM 103/202 compatible CMOS modem board
- 11) PWIO Optically isolated power I/O modules

12) DMP-3 Intelligent 6-pen plotter

"The CPU board, memory board, and digital I/O boards are rated for operation over a -40C TO +85C temperature range. The analog I/O, KDSW, and MODEM boards are rated for operation over a -20C TO +70C temperature range (with -40C TO +85C available on special order). The mass storage systems are rated for operation over a +15C TO +40C temperature range.

"The table-top system is targeted for program development and laboratory applications. This system is very compact with the standard five card CosMOS-8, the TU58, and the Disk each packaged in attractive enclosures measuring 11" w X 11" d X 4-3/4" h. An optional eight card CosMOS-8 is available and is packaged in an enclosure measuring 11" w X 11" d X 6-1/2" h. A limited control panel and a "control panel memory" resident ODT is provided for program debugging.

"The NEMA-12 packaged system is targeted for industrial data acquisition and process control applications in harsh environments. This CosMOS-8 System includes an uninterruptable power supply which is capable of powering the CosMOS-8, the Mini-disk, the Modem, and up to eight analog transducers (4-20ma.) for up to five hours (with one 9 amp-hour battery pack) during periods of power outages. The NEMA-12 packages are intended for wall mounting and measure 12" w X 6" d X 16" h each. The operator keyboard/LCD display board is provided with this model as a standard feature.

"The CosMOS-8 System supports a wide variety of software compatible with Digital Equipment Corporation's PDP-8 minicomputer. This software includes Digital Equipment Corporation's OS/8 operating system and Lab Data Systems' LAB-FOCAL Version-5B. System and non-system OS/8 handlers are available for the TU58 and the Mini-disk. A CosMOS-8 real-time I/O overlay for LAB-FOCAL is also available. This overlay provides commands, functions, and internal handlers to support the A/D converter, the D/A converter, the digital counters, the digital I/O ports, the serial I/O ports, and more. The overlay also provides support for LAB-FOCAL's new multitasking feature for these peripherals.

"For more information on the CosMOS-8 microcomputer system, contact Jim Butch, Eagle Research Corporation, 402 Fifth St., St. Albans, WV 25177. Phone 304-722-4257."

NOTE FROM WALLY KALINOWSKI

Along with his field test report on OS/8 PASCAL, Wally sent the following observations: "Lately there have been some super deals in PDP/8 hardware. FPP's for \$600, EAE's for \$100, I/O boards for \$20 etc. I think one could move the fans in a PDP8-M or F to make room for the FPP. Even though it is a hex size board, it only requires quad type connections.

"There are now a few Tektronix 4010 graphics terminal emulators available for approximately \$2000. Along with E. Lynch's program TKPLOT (available thru DECUS) some good graphics can be done on the '8'."

Actually, the Matrox 4010 emulator board for the VT-100 is now going for less than \$1000. Back when I last looked at TKPLOT there was no chance to get a 4010, but as Wally notes, that has all changed now. I took another look at TKPLOT recently and it really does look good. With some updating it could be very useful. I understand that Wally has located Dr. Lynch (who has moved since his DECUS submission was made)

and that there may be some new work going on to improve and update the package. Wally is at Aerospace Corp., P.O. Box 92957, L.A. Calif. 90009 (213) 648-6941.

TU58 TOPICS

I noted in Micronote number 97 (published in the LSI-11 SIG Newsletter Vol. 4, No. 1, January 1981) the following previously unpublished information on the "special addressing" mode of the TU58:

"Special Addressing Mode" - Setting the most significant bit of the modifier byte (byte 3 of command packet) to a 1 selects special address mode. In this mode all tape positioning operations are addressed by 128-byte blocks (0-2047) instead of 512-byte blocks (0-511). Zero-fill in a write operation only fills out to a 128-byte boundary in this mode. Applications that have less than 128-bytes of data per block could use special addressing mode to put more data on a cartridge."

Note that this mode is of particular interest to 12-bit systems because our blocks only need 384 bytes. In special address mode, a device handler can avoid wasting 1/4 of the available storage. It is also possible, using such a device handler, to access all the bytes written on a tape by a system like RT-11 that uses the entire 512 bytes of a normal block (which is not to say it is easy, you still have to unscramble the block boundaries and so on and cope with any differences in packing, file formatting and/or file structures).

Another point of interest to people programming the TU58: In the last few months I have come across a couple of indications that DEC has modified the protocol they use for communication with the TU58. The story goes that systems like the VAX and the 11/44 have trouble keeping up with input on a 19 Kbaud serial link from a TU58, so a change has been made that can slow down the transfer rate. I have not been able to get details but it sounded like the TU58 waits after sending each byte until it knows the byte has been received. If anyone has more information, let me know.

INPUT FROM JIM VAN ZEE

Along with the other material in this Newsletter, Jim sent an updated list of OS/8 - OS/78 software that he has available. Since many of these items can be very useful and they are not otherwise available, I will briefly list some of them:

RTPIP - An OS/8 program which allows one to transfer files in either direction to (or from) a single- or double-density RT-11 floppy disk. The usual sort of 'PIP' type options are available (directories, etc.). This program is similar in many respects to RTFLOP that DEC included in the Version 4 release of OS/78 but RTPIP corrects a number of deficiencies and bugs and adds functionality. Also, it is available to non-OS/78 V4 users.

TU58NS - Non-system handler for the TU58 serial line "DECTape II" cartridge tape. Requires a KL8E or M707 interface, or a VT78 SLU port.

VAX and TERMIN - ASCII file transfer handler (read/write) for use with other DEC operating systems; includes terminal emulator program.

RX2SYS - RX01/RX02 word-mode handler supporting both drives with interchangeable media on any DEC-compatible drive; needs 12K.

RX11/RX61 - Byte-mode floppy handlers for use on all DEC-compatible drives; standard bootstrap used for 12K system handler. 667 blocks per diskette; faster data transfer; 'RX61' version is for VT78s.

LPTS - Serial line handler for devices such as LA120.

LQPR - Improved LQP (i.e. Diablo printer on VT78) handler; left margin, backspace, variable pitch, single-page support.

DSKN - pseudo-device file handler: treats certain OS/8 files as devices. (Can be very useful, particular with larger devices - gives an effect something like having separate directories for each user or project on systems like RSX or RSTS - RH)

LPTT - Line printer emulator handler for any terminal; supports standard SET options.

PT - Paper tape/SLU handler (both read and write); changeable device code.

RK8ESY - Bootstrap for RK8E system handler modified to preserve the system date and to allow more convenient switch settings; no changes to standard handler.

VC8E - Handler for displaying uppercase ASCII characters on a point-plot CRT (VC8E interface).

LABL - Handler for punching readable labels on paper tape; uses either HS or LS punch.

#### NOTE FROM GEOFFREY B. INGLIS

Geoffrey wrote a while back to point out an article he noticed in Computerworld. Published two years ago, it was one of the series that reported Association of Computer Users (ACU) benchmark measurements on less-than-\$15000 systems. In this case, a DECstation 78 system, then priced at \$11,570, was reported to have done well. The article discussed a "unique aspect" of the machine - the twelve bit word length - and told how it "enables arithmetic calculations of 16-digit accuracy, as opposed to the more conventional six digit accuracy of many Basic systems". Also noted was that the 12-bit word length "allows two-instruction fetches from memory, which speeds up execution". Geoffrey thought it seemed funny to see the PDP-8 called advanced and unusual.

If the DECstation 78 was a good system (running OS/78 BASIC) at that price, the newer DECmates, at much lower prices must be very interesting. It is a shame that no one has done anything about OS/78 and BASIC on the DECmate II yet!

#### HELP - FORTH

James N. Butch wrote to ask if anyone knows of a PDP-8 version of FORTH. He is Vice-president of Eagle Research Corporation, 402 Fifth Street, St. Albans, WV 25177.

#### HELP - ST. LEWIS LOCAL USERS GROUP

Kent Glossop wrote to say he is interested in starting a local user's group in the St. Lewis area. If any one is interested in forming a LUG or just in getting in contact with one another, they can contact Kent at 8894 Berkay Avenue, Jennings, Mo. 63136.

#### PDP-12 NEWS FROM JIM VAN ZEE

"Jim van Zee reports that he has just finished writing a LINCtape system handler that supports both Drive 0 and Drive 1. With this handler most PDP-12 installations no longer need to use the 'non-system' handler, which (a) makes available another hand 'slot' and (b) expedites all operations on Drive 1, since the handler for that drive is now always in memory.

"PDP12 systems seem to run out of handler space more often than most other small OS/8 systems because of the versatility of the hardware; the desire to make room for an additional handler was thus the primary motivation. The machine on which this development took place now has handlers for LTAO: and LTA1: (the new system handler), DTA1: (DECTapes), DL: (DIAL tapes), TV: (an 85 char/line 'scope handler), TTY:, LPTT: (a lineprinter simulator for the terminal), LBL: (a handler for making 'labels' on paper tape), PTP: and PTR:, VAX: (a bi-directional communications handler for the VAX), DTUO: and DTU1: (a 12-bit file-structured handler for the TU58 tape drive that uses a 'TU58-simulator' program running on the VAX to provide large amounts of medium-speed, random-access storage that is easily archived on 9-track tape.), and DSK1: and DSK2: (which treat files on either drive with the names 'DSK1' and 'DSK2' as though they were independent devices; these files have their own internal OS/8 directory and thus serve as convenient storage areas for backup copies of important programs, etc.).

"As is evident from this list, the 8 handler 'slots' available in the system are clearly insufficient for even a 'plain jane' PDP12 with two tape drives and a teletype! The papertape handlers are usually omitted, but the rest are all essential; combining support for both LINCtape drives into one handler was thus the only way to fit everything in. The new handler has an additional nice feature: it preserves the system date when it is booted up, thus eliminating the vexation of having un-dated files!

"Anyone interested in obtaining a copy of this system handler may send a LINCtape plus return postage (\$2 US, \$5 foreign) to Jim van Zee, Lab Data Systems, 10320 Ravenna Avenue N.E., Seattle, WA 98125."

#### COS/DIBOL AND WPS NEWS FROM LAWRENCE H. EISENBERG

Larry Eisenberg sent a copy of the material from his last DIBOL BUSINESS SIG Newsletter. I have tried to edit it to fit the available space so our 12 BIT SIG members will have access to current information about COS and WPS on 12 bit systems. Larry's newsletter included the 30 mandatory patches for COS-310, version 9 that were released as of his publication date. Since there are no doubt even more by now and the DECUS publication budget is in big trouble, I have decided to omit the patches here, but be aware of the need for them if you use version 9. All of the following is edited from Larry's newsletter material.

DIBOL -- COS-310, VERSION 9.00

At last, VERSION 9.00 of COS-310 has been released. This DIBOL-8/11 compatible version of COS-310 is simply remarkable. Not only that, it works!

For now, let's discuss some of the highlights and mention that VERSION 9 includes both a DIBOL-8 compiler and a DIBOL-11 compiler. Both of these compilers work on the PDP-8 and most DIBOL-11 programs, presently running on PDP-11s, can be compiled with little, if any, change. The COS-310 editor is used for creating, editing and modifying

source files. The editor, by the way, has been improved to allow for a search and substitute technique. While not quite as fancy as TECO, it sure does make life a lot easier. For example, to convert a DIBOL-8 source program to a DIBOL-11 source program, using the statement SUB 'INIT,INIT', will find every line where an INIT statement (which will have to be changed to an OPEN statement) has been used. (If we used SUB 'INIT'OPEN', we could change all INITs to OPENs, but there is a little more work required.)

There are a few subtle changes which must be made if DIBOL-8 sources are going to be changed to DIBOL-11. The most important concerns the file handling structures. If your DIBOL-8 programs contemplated appending records to the end of files as needed, you're going to have to change your technique. DIBOL-11 requires all file lengths to be predetermined. There is no way to append a new record to the end of a file. This means that every record which can be written in a logical unit assignment must be written and defined. We use special characters at the beginning of the record to define each unused record. To "append" a record, we merely search for the first record defined with the special characters and write over in order to "append". [Now, I have said there is "no way" to append. Actually there are a couple ways that records can be appended under DIBOL-11. One is to simply write over the end of file marker with a WRITE, RECORD, NUMBER instruction. So long as the logical end of file has not been overwritten this will work. However, it does not result in an end of file marker. However, a simple DIBOL-8 program to read to the end of the file, XMIT a NULL, and then close the file will "fix" the problem. If the file is opened in output mode and a WRITES command is given, then an end of file marker will be appended, which is the second method. However, both of these methods, while they will work, and will work quite well, cannot be used on any PDP-11 operating system.]

Once the new technique is used, it is far easier than the DIBOL-8 append method, with all of the extra steps for closing and opening the files to "fix" a newly appended record. Of course SORTing does require another technique; generally the use of an index or pointer file for sorting results in faster and more efficient utilization of space and time.

For those who have been using COS-310, VERSION 9 will be a real pleasure. DIBOL-11 programs may be intermixed with DIBOL-8 programs under total BATCH control, although they cannot use the STOP prgram instruction from a DIBOL-11 to a DIBOL-8 program. Further (NOW HEAR THIS HERE, AS IT ISN'T DOCUMENTED ANYWHERE ELSE), the instruction STOP '@prnam.ex' will act as a separate BATCH command, if prnam is a BATCH file. The instruction STOP in a DIBOL-11 program, when followed by the name of a program, means stop execution of the present program and commence execution of the program named as prnam. If the prnam is preceded by the "@" symbol, then the instruction acts as a BATCH command.

#### VERSION 9's REAL FEATURES

O.K. So we can run DIBOL-11 programs on PDP-8s; is that all? Of course not. Version 9 allows you to mix RXO1 and RXO2 devices at the same time. I.e., if you identify drive 0 as an RXO1 and you tell the computer that drive 1 is an RXO2, then you can use drive 1 with an RXO2 diskette, although there is an RXO1 diskette in drive 0. Further, change the designation of the RXO2 from DY1: to RX1:, and, what do you know, you now can use RXO1 diskettes in that drive. What all of this means, of course, is that we now can transfer data to and from RXO1s and RXO2s at the same time. This includes any and all program and data material and you can SYSGEN from an RXO1 to an RXO2 or vice versa. (Keep in mind, however, that RXO, which always is the SYSTEM device, must retain its integrity

as an RXO1 or an RXO2, depending upon which format was used to boot the system.)

More fun. You can run programs residing on any device sysgened into the system. Thus, if a program resides on DY1: you can run it without having to remove the diskette and placing it into the SYSTEM drive. This is, of course, especially helpful where the SYSTEM device is a different type of media.

What other goodies? Well, we can disable CTRL C (temporarily or permanently); we can set and disable autostart with the SET command; we can examine the currently SYSGENED devices and the terminal settings with the SHOW command.

Moreover, with the ability to write files to the DIRECTORY it now is possible to create BATCH command files on the fly and depending upon operations which were performed in the running program. E.g., suppose a program is run where a SORT may be required, but only if records were deleted or appended or file designations were changed. With the ability to write a DIRECTORY file you can tell the file to SORT, or not, depending upon whether that operation actually is necessary. [Remember, the first two characters to be written in each record to a source file should be an empty A2 or D2 field.]

Have the need to address the clock? If your PDP-8 is equipped with a clock, as all DEC word processing systems are, it can be addressed with external subroutines provided for DIBOL-11 programs. Upper and lower case handling, while quite clumsy, can be output to the printer, as well as handled by the terminal. (Upper and lower case recognition to the terminal is quite simple, but if you want to store the information you either will have to utilize some sort of shift routine, as used by Word Processing, or store your characters as three character decimals, which then can be converted to, and output as, their ASCII equivalents.) However, it is not necessary to lock the CAPS LOCK when using VERSION 9. If lower case characters are input they will be converted to upper case by the operating system, unless you set the flag to reproduce them as received.

Execution time of DIBOL-11 vs. DIBOL-8 on the Eight? We haven't been able to detect any difference. However, program loading does seem somewhat longer at the beginning, but where appropriate overlays are used, program loading is far faster than chaining under DIBOL-8.

Anything else? Well, for starters, the new DECmate (the "278") requires the VERSION 9 operating system. NO DIBOL PROGRAMS WILL RUN ON THE DECMATE FROM VERSIONS EARLIER THAN VERSION 9. This does not mean, however, that you have to recompile all of your programs. Your programs will run on the DECMate; you just have to have a VERSION 9 operating system installed. There can be a few exceptions, however, depending upon what you have done with special escape sequences and graphics instructions and whether you use the terminal identifier sequence. Although the DECMate looks like a VT-100, it isn't and there are several subtle differences. (E.g., it doesn't have a VT-52 mode!)

#### DIBOL-8 PROGRAM PROBLEM -- NOTED HERE -- SOLVED HERE!

If you have any programs which require that a portion of a record be cleared and you use the name of the record to do so, the last two characters of the defined area presently won't clear. E.g., suppose you have a record defined as follows:

```
RECORD DEMO
DATA1          ,D5
DATA2          ,D10
```

```
DATA3      ,D10
DATA4      ,D6
```

If you want to clear the entire record, you could say:

DEMO=

The contents of each of the DATA fields will be cleared to spaces. But, if you only want to clear the fields identified as DATA2 and DATA3 (a common experience where there are a large number of such fields within a record) the usual method is the following instruction:

DEMO(6,26)=

At the present time, due to an oversight in the operating system (and this goes for all versions of COS-310), the foregoing instruction will clear the eighteen characters represented by DEMO(6,24), but will leave the last two characters (DEMO(23,24)) unchanged. There are two current "fixes" which can be used. The first is to clear the data field, DATA3, as well as the instruction for RECORD(n1,n2) and the other is to create a RECORD OVERLAY and create a field name for the area to be cleared and then clear that field.

Remember, this problem exists regardless of the size of the record and of the size of the portion of the record that is being cleared. DO NOT TRY TO CLEAR AN EXTRA TWO CHARACTERS (e.g. DEMO(6,28) in the above illustration) AS YOU MAY GO INTO THE NEXT RECORD OR, WHEN THE PATCH IS MADE, YOU WILL CLEAR TWO EXTRA CHARACTERS FROM THE FOLLOWING FIELD.

In the meantime we understand that a PATCH will be prepared for the problem and it should not be necessary to recompile any existing programs which are experiencing this particular problem.

#### INSTALLATION OF VERSION 9.00 ON EXISTING SYSTEMS

The following is a discussion of how to install Version 9.00 on existing systems so as to save your DIRECTORY and DATA files. Those of you who have experience in these matters can pass onto other and better things, but there are a large number of "novices", like this author, who might be interested.

#### CONVERTING A COS-310 SYSTEM TO VERSION 9.00

#### INTRODUCTION

Due to the fact that several differences exist in the protocol used by Version 9.00 from that used in earlier versions, as well as a difference in PIP (as described below), it is hoped that this article will be of assistance, especially to those of you who want to transfer your files from prior systems to Version 9.00.

#### DESCRIPTION OF DEVICE EXAMPLES

This article assumes a two-drive RX01 system, which represents many existing systems. If RX02 drives are to be used, the expression Dn should be substituted for RXn: ("n" referring to the drive number). Similar substitutions may be made for hard disk systems. (In all cases, RX indicates RX01; DY indicates RX02; DK indicates RK05; and DL indicates RLO1 and/or RLO2.)

Identification of a device under Version 9.00 requires that the device name be followed with a colon (except for DPU where the colon is NOT used). We will follow the conventions used in the manuals.

Under Version 9.00, it is possible to use RX01 and RX02 devices at the same time. Thus, drive 1 can be identified as an RX01 and drive 0 can be identified (or used) as an RX02, etc., at the same time. (This can be especially useful when using data files stored on one type of device format and being used by another, or with word processing files which are stored on RX01 formatted diskettes.) Of course an RX01 physical device cannot be used as an RX02 and, if RX02 is designated, its diskettes will have to be formatted for RX02 use.

#### INITIAL SET-UP CONSIDERATIONS

The Version 9.00 monitor requires 18 more blocks of disk space than versions in the 8.0n series. (The Version 9.00 monitor is 114 blocks long, whereas the Version 8.02 monitor used 96 blocks. The new monitor permits operation of DIBOL-11 programs compiled and linked under COS-310, Version 9.00.) When using the examples provided in this handout, consideration must be given to the additional space which will be used by the Monitor. (Some of the space may be recaptured by reason of some system programs being smaller under Version 9. See COS-310 Version 9.00 Release Notes and Installation Guide (June 1981) Order No. AA-J215B-TA, page 11, for a complete program size comparison.)

All DIBOL-8 compiled programs will run under Version 9, regardless of which version was used to compile the program. [However, the DECmate does not utilize VT-52 mode and certain characteristics may have to be modified in earlier programs if graphics, special ESCape sequences or the alternate mode characteristics of the keypad have been used. This is discussed below.]

#### CREATING THE NEW SYSTEM

Take a NEW (or scratch) diskette and place it in the right drive. Place your Version 9.00 SYSTEM diskette in the left drive. Boot the system (which will take longer than on earlier versions; if an RL handler has been SYSGENed but does not exist, the boot can take several minutes) and enter the date [DATE dd-mm-yy]. It is not necessary for the CAPS LOCK to be locked in upper case as lower case characters will be converted to upper case with Version 9. [Version 9.00 does not support lower case characters, although it is possible to receive, store (indirectly) and generate lower case characters using DIBOL-11 routines available under Version 9.00.]

New SYSTEM devices are created in the same manner as under Version 8. Run SYSGEN/B and follow the screen prompts. The difference, now, is that you may elect a secondary device, i.e., the ability to interchange the use of RX01 and RX02 or RLO1 and RLO2 at the same time. You now may have any two handlers "SYSGENED" into the system at the same time.

Upon entry of the "Y" response to IS EVERYTHING CORRECT?, the monitor will be transferred to the new device. REMEMBER: ALL PREVIOUSLY EXISTING DATA ON THE NEWLY CREATED DISKETTE WILL BE DESTROYED.

#### TRANSFERRING OLD FILES TO THE NEW SYSTEM

THIS IS AN IMPORTANT NOTE. Unlike earlier versions, VERSION 9.00 DOES NOT PERMIT OPERATION OF PIP WITHOUT THE PROGRAM RESIDING ON AN AVAILABLE DEVICE. In other words, a

Version 9 PIP must reside upon one of the devices being used for file transfer operations. [This is not required for OPTION B (BACKUP), but a BACKUP copy would transfer the old system, as well as the files, and cannot be used for updating your current diskettes to the new version.]

The following steps should be used to update an earlier version system to the new Version 9.00:

1. Create the new SYSTEM diskette under SYSGEN/B as stated above, setting the device handlers as needed. Respond "N" to the prompt: DO YOU WANT TO COPY YOUR FILES?
2. RUN PIP. Use OPTION T (to transfer a file). Transfer PIP over to the new device with the following protocol:  

```
RUN PIP
PIP V 9.00
OPTION: T
FROM: PIP.SV
TO: RX1:
COPIED RXO:PIP.SV TO RX1:PIP.SV
OPTION: X
```
3. This will transfer Version 9's PIP to the right (#1) drive.
4. Remove the VERSION 9.00 SYSTEM diskette from drive 0 and replace it with the newly initialized diskette. [Although you now can run programs from either diskette, this method will help to prevent errors arising out of habit!]
5. Place the older version diskette, from which the file transfer is to be made, into the right drive.
6. RUN PIP. This time you will use OPTION S. Option S normally is used to consolidate files (similar to OPTION E under earlier versions of PIP) but it also transfers all directory files from one device to another. Use the following protocol:  

```
RUN PIP
PIP V 9.00
OPTION: S
FROM: RX1: ("old" diskette)
TO: RXO: ("new" V 9.00 diskette)
OPTION: X
```
7. At this time all DIRECTORY (i.e., not data) files have been transferred from RX1 to RXO. AT THIS TIME, ALSO, Version 9.00's PIP has been deleted or replaced with the older version's PIP. Now, all SYSTEM files must be replaced with new VERSION 9.00 SYSTEM files. Remove the "old" diskette from drive 1 and replace it with the original VERSION 9.00 SYSTEM DISKETTE. Ask the directory for a listing of all the system files on RXO: DIR \*.SV (the "\*" is a wildcard; "SV" is the new designation for system files, formerly "V"; the "/" no longer is required as display default now is to the screen.
8. You must exchange all of the SYSTEM (.SV) files on drive 0 with the new Version 9

SYSTEM files. You do this with OPTION T, e.g.:

```
RUN RX1:PIP ; Note: Must run PIP from RX1 at this time.
PIP V 9.00
OPTION: T
FROM: RX1:DFU.SV
TO: RXO:
REPLACE? Y
COPIED RX1:DFU.SV TO RXO:DFU.SV
OPTION: T
FROM: RX1:SORT.SV
TO: RXO:
REPLACE? Y
COPIED RX1:SORT.SV TO RXO:SORT.SV
.
.
OPTION: X
```

Note that you must use the colon after the device designation and that on Version 9.00 the device designation precedes the file identification. Also, for files to be transferred you must include the file name and its extension (.SV) or a FILE NOT FOUND error will be displayed. It is not necessary to use the file name and extension on the TO: prompt, as identifying the device is sufficient to transfer the file with the same name and extension. Specifying a file name and extension will change the old file name.ex to a new file name.ex. If a file name is designated, but no file extension is given, then the new file will be created without an extension and may result in undesirable errors. (E.g., a SV program cannot be run without the SV extension.)

Also, if RXO is the system device, you can use SY: in place of RXO:.

9. At this time you may take a Directory listing and determine whether it is necessary to squish the directory (using PIP with the S option). If free blocks are found within the directory, you may wish to do so. ALWAYS HAVE A BACKUP AVAILABLE UPON ANY DEVICE WHICH IS BEING SQUISHED AS ANY FAILURE DURING A "SQUISH" WILL RESULT IN LOST FILES. ALL FILES CAN BE LOST!
10. At this time, the diskette in drive 0 contains all of the directory files which were on the "old" diskette.

#### TRANSFERRING DATA FILES TO THE NEW VERSION 9.00 SYSTEM DISKETTE

No different rules apply for the transfer of data files from a system diskette under Version 9 than previously existed. It is not, of course, necessary to do any "conversions" on a diskette which is used only for data files (i.e., no system head exists).

An exception, however, is to keep in mind that you may have less room than on your earlier version diskettes and if you have no room left in your directory, you will have to make some adjustment either by removing one or more program or source files or reducing

your data file size to accommodate the new system requirements. Each data file segment is the equivalent of 16 blocks.

The following example is given for the transfer of data files from one RXO1 "old" system diskette to a new Version 9.00 system diskette. (An RXO1 has 41 segments available; an RXO2 has 61 segments; an RK05 has 406 segments; an RLO1 has 638 segments and an RLO2 has 1278 segments. The protocol used here should be adjusted for each such device, if not an RXO1, substituting the appropriate total storage size as required.)

Transferring data files from one system device to another system device often is confusing, as the MONITOR considers a device, other than the designated SYSTEM device, to be a data device and counts the storage blocks in a different manner on the system device than on a data device. Illustrations of the manner in which data files are stored may be found on page 4-8 of the COS-310 SYSTEM REFERENCE MANUAL (Version 9.00) and in the discussion contained in pages 3-4 through 3-6 of the New Users Guide.

The method for deciding how to designate the appropriate logical unit assignments is to first ascertain the total number of segments which previously had been assigned to data files on the "old" system diskette. We will assume three files, as follows:

```
filnam1  5 segments
filnam2  3 segments
filnam3  7 segments
```

```
Total   15 segments
```

Deduct the total segments (15 in the example) from the device size (41 for an RXO1) and we have the area which will have to be "skipped over" for the transfer procedure. In this example, 41 minus 15 will be 26. Thus, the above example would result in the following logical unit assignments under DFU (assuming that the "new" SYSTEM device is on drive 0 and the "old" SYSTEM device (with data files to be transferred) is on drive 1:

```
RUN DFU/K      (Entering the assignments from the keyboard)
DFU V 9.00

1 RX1, 26;     MONITOR AND DIRECTORY STORAGE AREA BEING "SKIPPED"
2 RX1, 5;      filnam1
3 RX1, 3;      filnam2
4 RX1, 7;      filnam3
5 RXO, 5;      filnam1 (this is to be the new filnam1)
6 RXO, 3;      filnam2 (this is to be the new filnam2)
7 RXO, 7;      filnam3 (this is to be the new filnam3)
8 END
```

In the above example, the first logical unit assignment is to the area representing the monitor and system files and available directory space which reside on the old "system" diskette in drive 1. The second through fourth assignments represent the location of the files to be transferred. Assignments 5 through 7 represent the new locations to which the files will be transferred. Note that it is not necessary to "skip" over any monitor area on the SYSTEM device, as the MONITOR automatically assumes that all area not assigned to data files on the SYSTEM device is reserved for the SYSTEM.

To transfer the files, we run PIP as follows:

```
RUN PIP

PIP V 9.00

OPTION: D
FROM: filnam1#2          (NOTE THE "#" INSTEAD OF "/" )
TO: filnam1#5           (Transfer the file)
REPLACE ZZYXZ #68? Y    (Replace message may be expected)
MORE? N
OPTION: D
FROM: filnam2#3
TO: filnam2#6
REPLACE ZZYXZ #68? Y    (Replace message may be expected)
MORE? N
OPTION: D
FROM: filnam3#4
TO: filnam3#7
REPLACE ZZYXZ #68? Y    (Replace message may be expected)
MORE? N
OPTION: X
```

The data files now are transferred from the diskette in drive 1 to the diskette in drive 2.

#### MODIFYING BATCH AND COMMAND FILES TO REFLECT VERSION 9 COMMANDS

To complete the updating of "old" system diskettes to VERSION 9.00 it may be necessary to update every source file which is used as a batch or command file under earlier versions.

The following programs may be affected and probably require modifications:

COMP, PIP, SORT, DFU, MENU and FILEX.

#### COMP

In most cases, if a COMP BATCH file exists, it will not have to be modified, as it is not necessary to identify the extension and the device with a source file used in the compilation of programs, where all of the programs reside on the SYSTEM diskette. However, under Version 9.00 source (as well as all other) files now may reside on devices other than the SYSTEM device and may be used under program control. Thus, a BATCH file for COMP now may include source files which reside both on RXO and RX1 at the same time. Where the source file resides on a different drive than the SYSTEM, then the DEVICE name must be stated (e.g., RX1:filnam.ex). The extension of .AS is assumed if not stated. E.g., RUN COMP,RX3:FIL1,RL1:FIL2,RXO:FIL3/G.

#### PIP

All PIP command files will have to be modified to reflect the new dialogue. In the case of DATA FILE PIPs, the only modification may have to involve changing the "/" to a "#" with respect to the logical unit number. If so, then the change is easily

accomplished using the SUBSTITUTE command:

```
FE pipcmdfil    Bring up PIP command file
SUB '/'# '1,    Substitute "/" with "#" on every line
LI              LIST the file and be sure it's right
WR pipcmdfil/y  WRite the file and replacè the old file.
```

All other PIP command files will require major modifications and reference to the Manual will be required. (E.g., the protocol no longer accepts filnam,dev, rather, the protocol requires dev:filnam.ex.) A tremendous help is the fact that PIP now utilizes the wildcard designations "?" and "\*". The question mark is used to replace any one or more characters (including spaces), one question mark for each such character, and the asterisk is used to replace an entire extension or name. E.g., \*.\* would include everything. \*.SV would include all .SV files and FILNAM.\* would affect all files having the name of FILNAM, regardless of extension.

#### SORT

All SORT command files will have to be modified to reflect the new dialogue. The modification is quite simple, requiring only a couple of SUBSTITUTE commands.

Logical units now are defined with a pound sign (e.g., #3) whereas previously they were defined with the slash sign (/3). But, the SORT protocol continues to use the slash sign with the SORT instruction (i.e., SORT n/lu,lu,...lu).

The following SORT command file (CLSRT) presently is defined:

```
0010                DEFINE
0020 F1,D6
0030 F2,A20
0040 F3,D7
0050 INPUT CLFIL/1;  INPUT FILE
0060 SORT 3/2,3,4;  SORT WORK UNITS
0070 KEY F1,F2;     KEY ON F1 AND F2
0080 OUTPUT CLFIL/1; OUTPUT RESULT AND OVERWRITE OLD FILE
0090 END
```

To modify the file for Version 9.00, the following steps are utilized:

```
FE CLSRT           Fetch CLSRT from the system diskette
LI                LIst it (be sure it's the right one)
SUB '/'# '1,      SUBStitute all slashes with pound signs
LI                LIst it and confirm SORT COMMAND LINE
SUB '/'# '60      Change pound sign on line 60 back to slash
LI                LIst it and check that it's right.
WR CLSRT/Y        Replacè old CLSRT with modified file.
```

(If desired, the SUB command can be used for lines 50 and 80, only, and then it won't be necessary to change back on line 50. This is just a matter of choice as the same number of key strokes will be required in both situations.)

#### MODIFYING DFU COMMAND FILES

As previously indicated above, it is probable that no changes will be required for the DFU command files. IT IS NOT NECESSARY, UNDER DFU, THAT THE DEVICE NAMES BE FOLLOWED BY A COLON and if the colon is used, a SYNTAX ERROR will be generated. Thus, with a single exception, all DFU command files generated under Version 8 may continue to be used under Version 9.

The exception is where you may have used option B to generate your logical unit assignments with a BATCH command.

Option B of DFU under Version 8 permitted using the binary scratch area for assigning logical units. A typical example could be as follows:

```
00--                BATCH COMMAND FILE
0150 ER ERASES THE BINARY SCRATCH AREA (BSA)
0160 1 RX1,5        FIRST ASSIGNMENT
0170 2 RX1,2        SECOND ASSIGNMENT
0180 3 RX0,3        THIRD ASSIGNMENT
0190 R DFU/B        ASSIGNS THE LOGICAL UNITS FROM BSA
00--                CONTINUE WITH BATCH FILE
```

If any BATCH files contain binary scratch area assignments for DFU, they will have to be changed to incorporate a DFU command file instead as DFU no longer recognizes the "/B" option, which will generate an error if used.

#### MENU

MENU command files generally will not require any changes. There are, however, a few possibilities. The first concerns the possible use of the binary scratch area for logical unit assignments as noted in the preceding section. The same changes would have to be made to MENU as would have to be made to BATCH COMMAND FILES if such BSA assignments were used.

The second involves the possible use of comments in the COMMAND section of the MENU program. At the present time, comments will be acceptable to MENU if preceded by a colon or by an exclamation mark (although this is not documented in the manuals). It may be desirable to change the comment indicators to exclamation marks, as used by BATCH commands, to be certain that future versions will be compatible.

Further, any BATCH commands which identify files or devices may have to be modified so as to change the protocol to match the new use (e.g., the pound sign in place of the slash for logical units and the device name BEFORE the filename where the file and device are designated. E.g., old: FILNAM,RX1 now is: RX1:FILNAM.EX.)

#### FILEX

FILEX command files will have to be modified to reflect the new protocol for identifying DEVICES and LOGICAL UNITS. Wherever devices are used in a FILEX command file, they will have to be modified by appending the colon to the end of the designation. However, if the device was used in connection with the location of a file, the entire line will have to be rewritten to reflect the new protocol. E.g., the old FILNAM.S,RX1 will be changed to RX1:FILNAM.AS. Similarly, data file logical unit assignments would require

modification from the slash to the pound sign. In these latter cases, the use of the SUBSTITUTE command, as illustrated in some of the examples above, would be helpful.

#### BATCH COMMAND FILES

The preceding examples provide several illustrations of how or where BATCH COMMAND FILES may have to be modified with respect to various circumstances.

The remaining alteration concerns COMMENTS. In prior versions and, indeed, with many of the existing programs, as well as DIBOL, itself, comments are preceded by a semicolon. Under Version 9.00, comments in BATCH COMMAND FILES ARE PRECEDED BY AN EXCLAMATION MARK.

It is mandatory that all BATCH COMMAND FILES be modified by changing all semicolons to exclamation marks:

|             |                                 |
|-------------|---------------------------------|
| FE BATCMD   | FETCH THE BATCH COMMAND FILE    |
| SUB '!'1,   | CHANGE ALL ";" TO "!"           |
| LI          | LIST AND CHECK                  |
| WR BATCMD/Y | OVERWRITE THE OLD COMMAND FILE. |

THIS CHANGE IS NOT MADE IN DIBOL LANGUAGE PROGRAMS, HOWEVER. That is, the comments in DIBOL-8 language programs remain separated from the program instructions by a semicolon.

#### PROGRAM CHANGES WHICH MIGHT BE REQUIRED FOR DECMATE OPERATION

Version 9.00 is designed to run on the DECMate (278). In fact, the DECMate will not operate with any earlier version of COS-310. There are, however, a few possible problems which could arise with older programs written with the VT-52 (or, even, the VT-100) in mind.

Generally, escape sequences are handled the same on the 278 (DECMate) as on the VT-100 and VT-52. (Of course if you ask the terminal to identify itself (ESCAPE Z), the response on the VT-278 is going to be different than on the VT-100 or VT-52.) There are exceptions. For the most part, the exceptions are that the VT-278 acts like the VT-100 in ASCII mode (i.e., not in VT-52 mode), which really means that the alternate key pad generates different escape sequences on the VT-278 than on the VT-52 (which is especially noticeable on the top row of the keypad).

If the top row of keys on the keypad has been used for input in earlier programs, there will be different results on the VT-278 which will require additional programming in order to use older programs. Generally, however, this will require that the extra sequence character "O" be trapped and/or identified. Similarly, if the graphics mode had been used in earlier programs, different results may be expected on the VT-278.

Just as important, if a program is written to take advantage of some of the screen handling capabilities of the VT-278, some method of trapping the extra routines will be required before the programs can be run on a VT-52 or VT-100 type device.

For the most part, DIBOL-8 programs which merely display and accept information on the screen will be treated the same on the VT-278 as on a VT-52. However there are extreme differences if you are writing a DIBOL-11 program for use on the VT-278. The VT-278 operates in ASCII mode, only, and you must use the ASCII sequences for positioning the cursor as well as for all ESCape code commands. For a complete discussion on the

ASCII escape sequences for the VT-278 see: "DECMate System Architecture Manual" (1981) EK-VT278-HR-001, Chapter 5. [NOTE: We are informed that some early models of the VT-278 will not respond to ESCape Z to identify the terminal!]

#### SUMMARY

We hope that the preceding procedures will be of some assistance to you. Once you get started on Version 9.00 you will find that it represents a significant improvement over prior versions especially if, and as, you begin to incorporate DIBOL-11 programs into your system.

We leave you with one parting and undocumented hint, which will be of invaluable assistance. If you want to commence a particular BATCH COMMAND SEQUENCE from a running program (perhaps depending upon the operator's choice), you may do so under DIBOL-11 by use of the STOP '@ dev:filnam.ex' instruction. Don't forget the "@" symbol and, especially, don't forget the single quotes and you'll be in fine shape. (The referenced filnam.ex should be a BATCH command file, which will be operated if the STOP instruction is reached in the running program. While we mentioned this routine at the beginning of this letter, it's worth repeating!)

#### DECUS SURVEY -- QUESTIONNAIRES AND RESPONSES

Both at the Los Angeles Symposium and via mail, a questionnaire was submitted to members of DIBOL BUSINESS SIG to assist COEM engineering with respect to future participation in DECUS. The questions included: (a) Why do members attend DECUS? (b) If you have attended DECUS in the past, will you do so again? (c) If you have not attended DECUS, why not? and (d) What can be done to attract customers to DECUS.

The information gathered from the returned questionnaires is summarized as follows (as plagiarized from an internal DEC memo, dated 12-24-81).

Most, even those who had not yet gone to DECUS, felt that it was a very worthwhile event. They indicated that they were very interested in attending DECUS symposium and that the information given at DECUS was very valuable.

The biggest and most frequent complaint seems to be the cost and location of the symposia sites. Many cannot attend because it is too far to travel; some did not care for the type of location (two separate hotels can be somewhat of an inconvenience -- LHE) and many also indicated that because so many sessions were held at the same time they were not able to attend all the sessions that they felt would be beneficial to them.

Those who never attended a symposium indicated that they would like to attend if the location were more suitable and the cost were less, because they felt the information they could gain would be valuable.

The following is a brief summary of the responses:

- A. Most people attended DECUS because:
  1. Interchange of ideas, problems and solutions between OEMs and DEC people.
  2. Participation in deciding the future of some products.
  3. To learn about new releases.

- B. Those who have attended a symposium in the past but have not attended the Miami Symposium were influenced by the following [NOTE: most responses were from West Coast attendees, which should have a bearing upon their feelings]:
1. Too expensive to go to both Fall and Spring Decus. Further, the general feeling is that the Fall Symposia usually are superior.
  2. Miami was too far away from residence or business.
  3. Word processing was not sufficient to make two symposia cost effective. [LHE - As the Word Processing members have been part of the DIBOL Business SIG, this observation must be weighted accordingly and also must take into consideration the formation of the Office Automation SIG (if approved).]
- C. Most people who have not yet attended a Symposium stated:
1. Can't afford the time away from business.
  2. Too expensive.
  3. Always too far away.

Of those responding to the survey, 35% had attended both Los Angeles and Miami; 44% had attended Symposia in the past, but had not attended Miami; and 23% had never attended a Symposium.

Among the most impressive general responses from those who have attended more than one Symposium are the following:

Why did you attend?

- o The exchange of ideas between users, other OEMs and DEC people. We really get a chance to participate in helping to decide the future of many products.
- o To meet others with similar problems and, hopefully, solutions. To hear lectures, forums, etc., of special interest to our applications. To see new software/hardware releases and interface with DEC on future releases.
- o Excellent forum for interchange of ideas.
- o The information disseminated at DECUS is extremely valuable and hard to come by anywhere else.
- o To keep up with changes, learn unlisted programming techniques and DEC futures.
- o To get answers.

What could be presented at Symposia to cause you to attend?

- o How to program, Hinks & Kinks, a program comparing various operating systems.
- o Applications software discussions.
- o More DIBOL sessions.
- o More user papers and more user presentations.

- o More assistance with DIBOL programming, especially with respect to unusual techniques, performance considerations, indexing, etc.

Will you attend again?

All those who had attended previously replied "Yes"!

General Comments

- o I feel the information I get at DECUS is worth every cent of the cost.
- o I felt the return on the investment was somewhere in the 100-1 ration.
- o I'll always attend ... only place to get "REAL WORLD" answers.

What could be presented at a DECUS Symposium that would cause you to attend?

- o More sessions on COS-310, CTS-300 and CTS-500.
- o Lots of detailed new information concerning Systems, Language and Products.
- o Have it in Dallas!

Finally, as to those who have attended at least one Symposium, 72% said that they would attend again and only 9% said that they would not.



DIGITAL EQUIPMENT COMPUTER USERS SOCIETY  
ONE IRON WAY, MRO2-1/C11  
MARLBORO, MASSACHUSETTS 01752

BULK RATE  
U.S. POSTAGE  
PAID  
PERMIT NO. 129  
NORTHBORO, MA  
01532

INSTALLATION - 126374

VE  
1

**MOVING OR REPLACING A DELEGATE?**

Please notify us immediately to guarantee continuing receipt of DECUS literature. Allow up to six weeks for change to take effect.

- Change of Address
- Delegate Replacement

DECUS Membership No.: \_\_\_\_\_

Name: \_\_\_\_\_

Company: \_\_\_\_\_

Address: \_\_\_\_\_  
\_\_\_\_\_

State/Country: \_\_\_\_\_

Zip/Postal Code: \_\_\_\_\_

Mail to: DECUS - ATT: Membership  
One Iron Way, MRO2-1/C11  
Marlboro, Massachusetts 01752 USA

Affix mailing label here. If label is not available, print old address here. Include name of installation, company, university, etc.