

12-BIT WORLD

~~48~~ ~~DECUS~~ y

617-435 9064
435 9061

DECUS 12 BIT NEWS
=====

NUMBER 49

MAR 1985

Contributions and correspondence should be sent to:

Robert Hassinger, 12 Bit Coordinator
c/o DECUS ..or.. Liberty Mutual Research Center
249 Northboro Rd (BPO2) 71 Frankland Road
Marlboro, MA 01752 Hopkinton, MA 01748

(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. Based on the current schedule, material must be received by the first day of February to be included in the next issue.

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 Bob Hassinger for further details on acceptable media and formats if you plan to make a submission in machine readable form.

UPDATE ON GETTING OS/278 TO WORK ON THE DECMATE III

Jeff Monasch called. Many of you will recall he has given a number of interesting presentations at Symposia on OS/278 based games and other DECmates topics. After reading the item in the last newsletter about getting OS/278 to work on a DECmate III, he wanted to pass on an easier way to do it that does not require access to a DECmate II.

Jeff says the DECmate III comes with a System Test diskette. When you boot it, one of the functions is Update Internal Data on System Diskette. You put a copy of your OS/278 system diskette in drive 1 and run this option. It updates the "slushware" tracks on the OS/278 floppy floppy so it will work on a DECmate III.

Jeff is with DEC in New York at (212)-971-3947.

12 BIT COMMITTEE

Robert Hassinger
- address above -
(617) 435-9061

COS/DIBOL and WPS liason
Lawrence H. Eisenberg
17141 Nance Street
Encino, California 91316
(818) 788-0354

Education, Multiuser systems, PASCAL
Father Geoffrey Chase O.S.B.
Portsmouth Abbey School
Portsmouth, RI 02871
(401) 683-2000

Representitive to DECUS Product Planning Committee
Jim van Zee
Lab Data Systems
10320 Ravenna Ave NE
Seattle, Washington 98125
(206) 522-6950

12 Bit Software Project
Wally Kalinowski
The Aerospace Corporation
PO Box 92957
Los Angeles, CA 90009
(213) 648-6940

NEW SOFTWARE FROM JIM VAN ZEE

I recently was pleasantly surprised to receive a box from Jim van Zee. It contained a DECUS submission for me to review and pass on to the Library. Whenever Jim makes a submission it is worth looking at and this one is no exception.

Jim wrote that he has been following the evolution of the 12-bit product line in the Newsletter (which he now gets second-hand from Wally Kalinowski - thanks Wally) and he saw that I had submitted a copy of his DIRECT program (DECUS 8-842) on RX50 floppy for the DECmate II and III. He says he was pleased that the many man-months of work that went into that project are now available to a new set of users.

Jim says that coincidentally he received a copy of some OS/278 stuff, particularly the new screen editor, and he was intrigued by the version of DIRECT in the OS/278 package. He thought it looked like a modification of his DECUS version. He tried the non-DEC options he had added to DIRECT and found they were all there except for a bug in the /H option which caused garbage to appear before a header. The modification appeared to involve principally lower-case text strings and the elimination of some error messages in order to have the room for the code required for this feature.

This interesting discovery provided the incentive Jim needed to add a "few more features" to his V7B version that have occurred to him over the last six years, including, of course, the use of lower case text strings. The principal features Jim wanted were: (a) to have /B (starting block numbers) reduce the column count by one so that he could use this switch without having to specify '=2', (b) to have a null extension be equivalent to '*', and (c) to improve the looks of a single column listing. After he implemented these and the lower case feature he had another idea: why not make '*' work more like it does on the VAX. So he rewrote some more routines, and behold: .DIR AB* is now equivalent to .DIR AB????.* - but MUCH easier to type!

Jim notes that unlike the DEC version which he thinks may have been done by Dick Murphy, this version is complete with all the original error messages, including the famous "THERE IS NO HOPE - THERE IS NO TTY HANDLER IN YOUR SYSTEM" message. Of course, the program is still exactly the same size, seven blocks as DEC's original OS/8 program. I recall talking to Jim at the time he submitted V7B. He felt it had been quite a trick adding all his features without having the program grow and it would be very hard to fit in some of the features that would be nice. Now he has managed to go and add all of them and more! It just goes to prove what the original author of OS/8 once said. On the PDP-8 you can always optimize out one more instruction. Applied iteratively this means you can always find the space to make the code fit if you try long and hard enough.

Jim has been working with Louis Tribble on another program called CCLX. He thinks it is really neat and will be submitting it to DECUS when it is ready. This submission includes a preliminary version of CCLX (no source) as well as versions of a number of other programs. The writeups for the ones that have been changed recently follow. Because of the way they are integrated with this material, Jim also included the writeup files for his Lab Data Focal (LDF) and TERMIN on the disk and they are included here even though the programs themselves are not on this disk.

DIRECT V7C

This is an improved version of the directory listing program distributed with OS/8, OS/12, OS/78 and OS/278. It is functionally equivalent to the official version, with a host of new features:

1. Multiple column listings are now printed in column order rather than in row order. This makes the listing much more 'readable' since sequential files and embedded empties appear in their natural order.
2. The listing can be sorted! The output can be arranged alphabetically (by name or extension), or viewed in forward or reverse chronological order. Ordering is first by date, then extension, then name.
3. Dates can be printed in either the 'DD-~~MM~~-YY' format of the original version or the numeric 'MM/DD/YY' style used by the monitor DATE command prior to 1978. The extended date algorithm devised for V3D is used to interpret file dates after 1977. (Good through 1999!)
4. The header block defined by the DECSYSTEM-8 program HEADER.SV can be listed at the beginning to provide a convenient identification of the media and the program contents, all in one place.

5. Format changes: spacing between columns was reduced so that three columns could be displayed on a normal terminal; likewise fewer blank lines are output to reduce loss of information on video terminals, and the formfeed at the end of each listing has been made optional. Text strings now appear in a more pleasing upper/lower case style, and the summary indicates the amount of space used by the files listed as well as the amount of 'empty' space.
6. Input changes: omitting the 'extension' is equivalent to specifying '*' and '*.'s can now also be used in names: .DIR ABC* = .DIR ABC???.*

The new features are controlled by the /A, /D, /H, /N, /P, /T, /X switches. '/A' produces an Alphabetical list, while '/X' groups files by their extensions, '/D' organizes the listing by Date and '/T' lists the most Timely (recent) files first. If '/E' is specified, 'empties' appear at the end of the sorted listing. '/P' starts a new page while '/N' generates Numeric dates; '/H' prints the Header block if there is one. Note that the date options only work if the system date has been specified (by a .DA command), and that '/B' and '/I' now automatically reduce the column count by one to compensate for the extra space used.

Some of the switches can be inverted to further enhance operation of the program. Thus '/P' can be changed to Prevent Paging, and '/N' can be changed to specify Non-Numeric dates rather than Numeric ones. Users who do not want lower-case text strings can remove this feature. The default value for the number of columns has been changed to '3' in order to agree with the .DIR command. Changing the default for '.DIR' via the '.SET TTY COL N' directive does not alter the value in DIRECT.

Optional patches (shown in ODT format: 'location/ value change')

```

12261/ 7710 7700 Change /N to mean 'print non-numeric dates'
12275/ 7640 7650 Change /H to mean 'do not print header information'
13267/ 7640 7650 Change /P to mean 'do not output a formfeed'
14037/ 0040 0000 Force all text strings to be in upper case
13177/ 0003 0004 Change spacing between columns (your choice)
14777/ 0003 0004 Change default to 4 columns (or your choice)

```

Implementation notes: the /W option now prints only the version number. This is consistent with the documentation, but differs from the way DEC's version works (it also prints a directory listing). To ensure complete compatibility, all the error messages were kept intact (even impossible ones!). The 'version release number' of the monitor system is now printed in the header. Note that this 'number' is actually printed as a letter to conform to the DEC convention - hence you should use the value '4' in HEADER if you want to have the letter 'D' printed. System ID=5 identifies 'OS/78' while ID=6 indicates OS/278. Header information may be upper/lower case, either 7- or 8-bit ASCII.

Acknowledgments: The initial work was done by Larry Fowler of the Boeing Commercial Airplane Co. in Seattle, Washington. He put in the /A and /H options and then turned it over to Dr. Thomas W. McIntyre of the West Virginia University Medical Center in Morgantown, W.V. Tom added the column-output feature and sent it back to Jim van Zee in the Chemistry Department at the University of Washington in Seattle, where the remaining options were added and everything was condensed in order to fit in the original file space. Thanks to

Rev. Geoffrey Chase of the Portsmouth Abbey school in Portsmouth R.I. for suggesting the /X switch and to Tom McIntyre for encouraging the addition of /D. Dick Murphy at DEC's Santa Clara office suggested an easy way to get lower case text from 6-bit strings, and Don Harmer at the Georgia Institute of Technology in Atlanta fixed a bug in the /T option.

Summary of the command decoder options:

```
/A      ALPHABETIZE DIRECTORY BEFORE PROCESSING
/B      PRINT STARTING BLOCK NUMBERS (OCTAL)
/C      LIST ONLY FILES WITH THE CURRENT DATE
/D      ORDER FILES BY DATE, THEN ALPHABETICALLY
/E      INCLUDE EMPTIES IN THE LISTING
/F      FAST MODE (NO LENGTHS OR DATES)
/H      PRINT CONTENTS OF THE HEADER BLOCK
/I      PRINT ADDITIONAL INFORMATION WORDS (OCTAL)
/M      LIST ONLY THE EMPTIES AND THE FILE SUMMARY
/N      PRINT DATES IN NUMERIC FORM (MM/DD/YY)
/O      LIST ONLY FILES WITH OTHER THAN TODAY_'S DATE
/P      PAGE AFTER EACH LISTING
/R      LIST REMAINDER OF FILES AFTER THE FIRST ONE
/T      LIST THE MOST TIMELY (RECENT) FILES FIRST
/U      TREAT EACH INPUT SPECIFICATION SEPARATELY
/V      LIST ALL FILES NOT-OF-THE-FORM SPECIFIED
/W      PRINT THE VERSION NUMBER ON THE TERMINAL
/X      ALPHABETIZE BY EXTENSION, THEN BY NAME
=N      USE N COLUMNS - OUTPUT IS ORDERED BY COLUMN
        (NOTE: /B AND /I REDUCE NO. COLUMNS BY ONE)

*       WILD NAME OR EXTENSION (NULL EXTENSION = *)
?       WILD CHARACTER ( '_?X*' IS EQUAL TO '_?X????_' )

DEFAULT INPUT SPECIFICATION:  *.*
DEFAULT OUTPUT DEVICE:       TTY:
DEFAULT OUTPUT FILE EXTENSION: .DI
```

CCLX

CCLX is a user-version of OS/8 CCL developed by John Easton and John van Essen (of PASCAL fame) at the Speech and Hearing Research Lab of the University of Minnesota (with some extra features added by Jim van Zee).

CCLX implements all the OS/8-V3D CCL commands -- and more -- yet occupies significantly less disk space. The command summary shown below is output whenever an unrecognized command is encountered (just type a SPACE plus a RETURN). Users can .DEFine additional commands at any time without modifying the source code: e.g. .DEF HEADER adds the .HEADER command!

The .DEscribe command provides easily extensible documentation: .DES will list all .HL, .WU and .DC files on any device, or will print out a specific one. The .REcall command is equally useful for listing any or all of the command strings CCLX saves in its 'remembrance' area. The .DATE command also automatically lists the contents of the file DATE.TX to provide notes on daily system operation, while the .YESTerday and .TOMmorrow commands help the

forgetful user. There is even an automatic prompt for the current date whenever you 'bootup' the system.

Other features include: semicolons ';' are converted to ':'s, in case you forget to shift when spelling a device name; backslash '\' options un-set CCLX switches, e.g. 'DEL *.*q' disables the '/Query' option on a global delete, and .COPY DTAO:<RKBO:l omits the tedious 'files copied' message. All system messages are output in upper/lower case. Two of the commands were changed to agree with other DEC operating systems: .DIFFer replaces .COMPare, and .XREF replaces .CREf (so .CReate could be shortened to two letters). The .PRint command in THIS version has been patched to be the same as .LIst by changing the filename reference from LPTSPL to FOTP.

The .VER command has a /E option to list the version numbers of all CUSPS (using the VERSN3 program), and the .VieW command joins .TYPE, .PRINT and .LIST as a way to send output to a specific device, in this case the TV:. Many bugs in CCL have been fixed, for example .MAP never calls ABSLDR, as DEC's version often did! The .LGo (Load 'n Go) command provides a useful option, while .TIDY helps keep a clean disk by deleting all *.TM and *.WS files. .Eject starts lineprinter output on a fresh page.

The .FUtil, .TERmin, .RTpip and .LDF commands have been included for the convenience of those who regularly use these CUSPS - use .DES to get more information. The two OS/78 symbiont commands (.REQuest and .Queue) were omitted because they limit memory to 16K and only work on floppy systems.

To install CCLX: (1) copy it to your system device; (2) use the .CCL command to disable CCL and (3) .R CCLX to enable it. Reverse this procedure (.CCLX,.R CCL) to go back to DEC's version. The .CCLX command un-defines all .DEFined commands. The 'native' command list is shown below:

BACKsp	DEassign	EJect	LIst	PRint	SKip	UA	ZERO
BASic	DEFine	EOf	LOad	PUnch	SQuish	UB	
BOot	DELete	EXecute	MACrel	RECall	SUBmit	UC	
CCLx	DEscribe	FUtil	MAKe	REName	TEco	UNload	
COMpile	DIFFer	HElp	MAP	RESource	TERmin	VERsion	
COPy	DIRect	LDF	MEMory	REWind	TIDY	VieW	
CReate	DUPLICat	LGo	MUNg	RTpip	TOMorrow	XREF	
DATE	EDit	LINK	PA1	SET	TYpe	YESTer	

HEADER

HEADER is a program for creating DECSYSTEM-8 header blocks, which are used for machine-readable identification of each media 'volume'. The header information is stored in block 6, which is the last block used by the directory. On small devices, such as DECTapes or floppy disks, this block is never used; on large disks, however, the header block may ultimately be overwritten, in which case the header information will be lost, but no other side-effects will occur.

The descriptive information written by HEADER can be printed as part of the directory listing by the DECUS version of DIRECT (DIRECT-V7). While for most users this is the main utility of the program, HEADER is also used for the 'auto-assign' feature in DECSYSTEM-8. The program contains its own 'help' file, so just respond with a 'Y' to get help when running the program. Authors: Doug Wrege, Bob Hassinger.

COMPAF

COMPAF compares all files on two different devices, giving you a listing of those which differ. Call with `.R COMPAF`, then specify `OUTDEV:<DEV1:,DEV2:` (no defaults! Make DEV1 a tape, DEV2 a disk)

VERSN3

The VERSN3 program (Author: Phil Siemans) produces a list of the version numbers of all the CUSPS it finds on the device specified (default=SYS). Commands of the following form may be used:

```
.Version [device]      Lists version of CD, ODT, KBM and CCL
.VE/E                  Lists versions of all the CUSPS too
```

Further information is included in the program source as follows:

VERSN3 provides an expanded capability of the `.VER` command. The general structure is:

```
.VER [ODEV:[FILE.XX]][<][IDEV:][OPTS]
```

`.VER` is the default command and prints the version of OS/8 and CCL as usual.

`.VER IDEV:` prints the version of OS/8 and CCL that are present on IDEV. If one or both are missing a "NOT PRESENT" message is output, e.g.:

```
SYSTEM NOT PRESENT
CCL V1F
```

The `/E` option causes an expanded printout to occur. The version numbers of all cusps that VERSN3 knows about are printed also.

```
.VER /E
OS/8 KBM V3Q
CCL V1F
```

```
ABSLDR.SV  4A
PIP.SV     12B
```

If the file `VERNUM.SV` is present on SYS, it will be used to specify additional cusps. See comments in the source for table format.

DROP

DROP will scan an entire mass-storage device seeking clues to the nature of the files it contains. Thus if you lose the directory on a valuable disk and do not have a recent listing, DROP can be used to at least give you some idea of the files present. The program recognizes, for instance, `.BN` and `.SV` files, and

in the later case, uses the CCB to suggest a likely candidate, ie PIP or ABSLDR or some other system program. Of course it does not know about any of your own special programs, but will list the starting address and the core locations used, which is often sufficient. In the case of ASCII files, DROP will print the first 60 characters (sans all Carriage Returns and Linefeeds), which again is usually enough to identify the file. Given the starting block and the length, you can then copy it with LDF.

DROP asks at the end: 'WRITE DIRECTORY?'. Respond -only- with 'NO' (or with a carriage return); this option is not properly implemented!

```
EX:          .R DROP (or .DEFine DROP and use .DROP dev:-L)
            *output device < input device
defaults:   [none]          DSK:
```

DROP was developed by an unknown author in Prof. W.L. van der Poel's lab at the Technical University of Delft (Holland) and was made available at the 1978 European DECUS meeting in Copenhagen.

HELP

This version of HELP.SV was designed for use with the LAB-FOCAL script file. The files required for its full use are HELP.HL and LDF.HL. By default all text comes from HELP.HL, hence normal use of the .HELP command is unchanged.

The script files typically reside on SYS:, however they may also be on DSKO: (this is for MULTI8 users), or on the device called HELP: (provided it's coresident with SYS: or has a 1-page handler). A minimal HELP.HL file is provided for those who only wish to use the LDF.HL file.

```
.AS RKBO HELP - make RKBO: the 'help' device
.HELP BASIC  - get help on BASIC from HELP.HL
.HELP LDF    - get help on LDF from LDF.HL
```

If the user requests help on LDF without specifying any particular topic, a list of all commands and functions appears, and the Command Decoder (CD) is then recalled to process requests for additional help. A '*' will list all headings in the script file, while '?'s serve to indicate a range of topics. Up to 5 items can be requested at a time.

```
.HELP *      - lists all subjects in HELP.HL
.HELP LDF,*  - lists all subjects in LDF.HL
.HELP LDF,ASK - provides help on the ASK command
```

The default output device is TTY:, but any valid output device may be specified. If the device has a directory the output file will be called HELP.LS unless the user supplies another name.

```
.HELP SYS:< *,????? - lists all headings, then all material
                    found in HELP.HL. Output goes to the
                    file SYS:HELP.LS.

.HELP LDF, *, ??????-L - prints all info in LDF.HL on the LPT:
```


(Use this command to get a mini-manual)

Three Command Decoder (CD) switches are recognized by this version of HELP:

/P	-	Start a new PAGE (formfeed) for every subject
/T	-	Use TABS rather than spaces in the output file
\$ (altmode)		Recall the CD, or exit from CD to the monitor

The default is NO FORMFEED and NO TAB, which allows the TV12E or ASR33 handlers to be used for output.

LDF

LDF (Lab Data Focal) is an interactive language designed for experimental control and 'hands-on' problem-solving. Calculations are accurate to 10 digits (including mathematical functions). LDF has over 40 commands and 30 functions, with additional commands and functions available for common configurations such as the PDP12 or LAB8/e.

OS/8 support includes four data files with a choice of ASCII and 4 binary (random-access) modes, a program library that includes subroutine calls, directory commands, and access to any data word on any OS/8 device. File names can be 'subscripted' for automatic data collection and files can be opened, closed, and examined interactively. LDF can also run under BATCH or be used in the background under RTS/8, SKED, MULTI-8, etc. CCLX has a .LDF command that can be used to call the program and give it a series of commands at the same time: .LDF: LOAD GO FORMAT;C FORMAT DISK IN DRIVE 1.

Graphics support includes a refreshed display on the PDP12/LAB8e, plotter routines for the XY8E/I (with upper/lower case annotation, including true decenders!), and Tektronix graphics with additional support for the Z19GP terminal. For users with 'smart' graphics devices, a special 'escape sequence operator' allows fancy peripherals to do fancy things very easily; the graphics features of the VT-125, for example, are readily put to use. Another powerful feature is the ability to use the keypad on 'VT-xx' terminals to initiate subroutine calls, providing 'push-button' programming.

The Input/Output/Use Host commands allow the programmer to switch between the PDP/8 and a 'number-cruncher' at any time. The four internal buffers can also be used as I/O devices, facilitating complex operations such as 'look back' and 're-read' of input streams, editing, and data conversion. The entire 7-bit character set is supported and all programming can be in either upper or lower case.

Special real-time features include external subroutine calls initiated by one of 4 internal timers (or a Schmitt trigger event) with response times of 5 msec. (Note: these are FOCAL subroutine calls, not machine-language subroutine calls!) Arbitration is by means of a 12-level priority queue. For testing and control purposes, the FPAL function allows machine-code routines to be built into the program, providing an extremely useful combination of low-level and high-level programming.

LDF requires a minimum of 12K; additional memory (up to 32K) is used for data storage, providing over 3400 variables, with both single and double-subscripted arrays. The source files are available as well as an

'expansion package' which makes custom modifications very easy; and for dedicated applications there is a compatible non-OS/8 version that fits in 8K.

For more information, please contact Jim van Zee at Lab Data Systems, 10320 Ravenna Avenue N.E., Seattle, Washington 98125.

TERMIN

The TERMIN program connects the PDP/8/12 console device to a remote computer system through a serial-line interface (KL8-E, -JA, M707, SLU2,3). Runtime options provide selection of device codes, control-code flagging and baud rate selection (DECstations only). TERMIN is typically used to establish a connection with a remote machine, using the HOST handler for the actual data transmission. XON-XOFF handshaking is used, hence a 110 baud terminal can be used with a 9600 baud data line.

Command form: .TER /option list

Options:	/Arrow	print control codes as '_^char_'
	/Baudrate=nnnn	Set baud rate (50-19200)
	/Speed= nnnn	Same as /Baudrate
	/Input= nn	Two-digit input device code
	/Output=nn	Two-digit output device code
	/Decsta=2,3	Select SLU2 or SLU3 on VT/78
	/Normal	Restores default settings
	/Normal= nn	Same + sets device codes (nn,nn+1)
	/Permanent	Saves any changes made
	/Help	Prints option summary
	/Version	Prints version number

For more information, please contact Jim van Zee at Lab Data Systems, 10320 Ravenna Avenue N.E., Seattle, Washington 98125.

HELP - OS/8 V3D

Dr. Anthony L. Marchese wrote recently with a problem. He has inherited and resurrected a Lab-8/E that was going to the "obsolescence farm". Since the machine had been located about 50 feet from his laboratory for more than 14 years and he was unable to get his hands on it, he felt a need to restore it to its "proper lofty place".

DECTape drives and OS/8 were added in 1975. From his letter it sounds like he still has OS/8 version 3 or 3C. His problem, of course, is that the dates do not work anymore. Old timers will recall that the original date scheme in OS/8 only worked from 1970 through 1977. This was fixed in the final release of OS/8, version 3D, that came out at the beginning of 1978.

Dr. Marchese he has been having a lot of trouble trying to find a way to get a V3D update. He can not seem to find anyone at DEC who can help him but the problem is that at last report DEC still considered this software proprietary and had not committed to placing it in the public domain so there seems to be no way for him to solve his problem unless he can get together with some kind soul who can help him out. Anyone who remembers the tricks and

programs we used to work around the "date problem" before we got V3D could provide him with help on ways to make do with the version he has now too.

If you can offer any help, please contact Dr. Marchese at Loyola University, Stritch School of Medicine, Department of Ophthalmology, 2160 South First Avenue, Maywood, Illinois 60153.

DECIMATE II/III WPS HINTS AND KINKS FROM RUDI STANGE

"Some Items about DECmate-II/III WPS which might not be known yet but may cause some confusion:

"Observing the text with "Gold View" shows just one screen of text; upon hitting the return key, the system switches back to normal. When it is desired to view more than one screen full of text, the blue "LINE" key on the small key pad should be used instead of the return key to advance through the document.

"A Ruler which is up to 79 characters long allows a max. of 18 Tabs. When more Tabs are desired in a ruler, it has to be prolonged to a length between 80 and its maximum length (250 characters with V.1.6) after this the ruler allows a maximum of 39 Tabs.

"The specification document for List Processing can have a length of 19 lines max. including the line which holds "Process Record". Each statement has to occupy its own line, i.e. no more than 1 statement per line is permitted.

"To erase a non desired ruler: for V.1.5 or later, the cursor is to be positioned directly under the ruler; Gold R (displays this ruler); and "Del Word" on the small key pad deletes the ruler. For earlier versions of WPS, it is necessary to save the ruler prior to the unwanted one by placing the cursor directly above the ruler to be deleted, press Gold R and Shift 9; this saves the ruler. Then move the cursor directly underneath the ruler to be deleted, display this ruler also with Gold R and just 9 and Return; this makes the unwanted ruler equal to the previous one and thereby gets rid of it.

"When printing address labels with List Processing it is advisable to regard every address as one page, i.e. count the number of lines from the top of one label to the top of the next label and use this number as PS (total number of lines per page) in the print menu. Also, set PM and BM (top and bottom margin to zero). Before printing all labels, just set the print menu to print only 3 or 4 pages to test the correctness of the set up."

WPS HINTS FROM RUDI STANGE

"For Readers who missed the Perspective Vol.2 No.1, I try to relay some valuable information from it produced by Patrick LaBarbera who owns "Office Computer Ssystems" Birmingham, Michigan; a firm specializing in developing office systems for DECmate users.

Creating long lists without Cut and Paste

"Since the Cut and Paste procedure to duplicate records for list processing is time consuming and tedious. When the Math Option is installed, consider the following procedure: "E.g. if you have to create some 200 records of the following format:

```
<cust name>
<cust add1>
<cust add2>
<city>
<state>
<zip>
<contact>
<>          (the record may be as long as you like)
```

"Do the following:

1. Create a document that contains 200 records (or as many as you need) named LIST. To do so type:

```
<: #>0
<><><><><><><><><><><><><><>
<><><><><><><><><><> etc. 200 times
```

2. Insert the following control Math block at the head of LIST:

```
-----START CONTROL-----
wpsmath
formula<: #>+1=<: #>
-----END CONTROL-----
```

3. Create a document named SPEC and type the following command, which tells the machine to process every record:

```
PROCESS RECORD
```

4. Create a document named FORM and type the following:

```
<<cust name>
<<cust add1>
<<cust add2>
<<city>
<<state>
<<zip>
<<contact>
<<: #><: #>          (you may use No or the like, but Math control
<<>                  block must get the same.)
<>
```

Be sure to add 2 carriage returns to space the records correctly.

5. From the Main Menu type:

```
LP
D
LIST
SPEC
FORM
give the NAME for the resulting document
GO
```

"You should get 200 records numbered consecutively from 1 to 200 in the field
<: >.

Numbering your Lists

"If you have an unnumbered list containig data and you want to number it,
do the following:

1. Create a form named REPEAT and type the following:

```
<!S>
<<:#><:#>
<!R>
```

Edit the LIST document which you want to number and add the Math
control block at the top as shown before.

2. From the Main Menu type:

```
LP
D
NAME of your LIST
SPEC
REPEAT
NAME of the resulting document
GO
```

"You should now have a numbered duplicate of your original list.

List creation using consecutive numbers

"To create check off lists of consecutive numbers do the following:

1. Create a document and name it NUMBER, which should look like this:

```
L-----T-----T-----T-----T-----T-----T-----T-----T-----R
<:#>0 <:#>1 <:#>2 <:#>3 <:#>4 <:#>5 <:#>6 <:#>7 <:#>8 <:#>9
```


"These rather simple uses of the DECmate give a clue to the untapped possibilities of List Processing, Sort and Math for solving office problems currently done manually."

C ON THE DECMATE II - FROM RUDI STANGE

"I do not work much with OS/8 anymore, except when something interesting is coming up. This happened a couple of month ago, when started reading Brian Kernighan and Dennis Ritchie's book on "programming the C Language". I write fairly regularly in the DECUS Munchen Bulletin on experiences with computers, and with the help of some readers and various correspondences with US distributors I got the opportunity and tried four C compilers on my DECmate-II.

"One of these is the PAL8 assembly of the public domain which appeared in Dr. Dobbs Journal No.45 (donated by Ron Cain) some years ago. It was intended for the 8080 and was only a subset of C. This was assembled with PAL8 and it runs under OS/8. I managed to get it running under OS/278. It is by no means a complete version by the standards of Kernigan & Ritchie but serves very well for learning purposes.

"Three other C compilers run under DECmate's CP/M operating system. Not exactly, however, I had to do some adaptation to the DECmate hardware which forced me to learn more about CP/M, which I have documented in German language, and which I present here, not too difficult to understand, when you accept a short German ex-course:

German	English	German	English
Schreiben	write	lesen	read
Zeichen	character	Konsole	console
keiner	none	Drucker	printer
von	from	zu(zur)	to
Puffer	buffer	Wag.Rucklf.	Carriage Return
abheben	lift	Kopf	head (read/write head)
suchen	search	Eintrag	entry
loschen	erase	erstellen	create
Abfrage	request	angeben	submit

"Calling CP/M routines is relatively easy. I/O routines are defined as functions, and each function has its own number attached. When calling, this number has to go into the C register. Entry parameters go into the E register as long as one byte is concerned. For 2 bytes the D register is added. The system returns data in the A register or in a buffer area in memory. When the registers are loaded, a call is made to CP/M address 0005H (hexadecimal). CP/M does not preserve the register contents, this must be done by the calling program (if necessary). A table of the functions follows:

Funkti- ons-Nr.	Funktions- description	Entry Para- meter	Returned Value	call	remarks
1	1 Zeich.v.d. Konsole lesen	keiner	ASCII- Zeichen	MVI C,1 Call NTRY	Read-Funktion-1
2	1 Zeich. zur	ASCII-Zeich.	keiner	MVI E,CHAR	Zeichen in Reg.E

	Konsole senden			MVI C,WRITE WRITE=2 Call NTRY	
3	1 Zeich.vom PTR-Gerat	keiner	ASCII- Zeichen	MVI C,RDR RDR=3 Call NTRY	
4	1 Zeichen zum Stanzer	ASCII- Zeichen	keiner	MVI E,CHAR Zeichen in Reg.E MVI C,PNCH PNCH=4 Call NTRY	
5	1 Zeichen zum Drucker senden	ASCII- Zeichen	KEINER MVI C,PRNT	MVI E,CHAR Zeich. in Reg.E PRNT=5 Call NTRY	
7	get E/A-Status	keiner	Status-Byte		
8	set E/A-Status	E/A-Status-Byte			
9	Druckerpuffer zur Konsole	Adr.des Druk- kerpuffers	keiner	LXI D,PBUF Pufferadresse MVI C,BUFO BUFO=9 Call NTRY	
10	String von Konsole lesen	Adr.Einlese- Puffers	ges. Puf- fer od.bis Wag.Rucklf.	LXI D,RBUF Pufferadresse MVI C,BUFI BUFI=10 Call NTRY	
11	Konsolen-READY- Flag testen	keiner	01 wenn Zeich.da	MVI C,ASK ASK=11 Call NTRY	
12	Kopf abheben von zugeordne- neter Disk	keiner abgehoben	Kopf wird	MVI C,12 12=Lift-Funktion Call NTRY	
13	Init Disk Access	keiner (DMA-Adr.auf	Disk-A wird zugeordnet; 0080Hgesetzt)	MVI C,13 13=Initialisieren Call NTRY	
14	Select & Login Disk	Disk-A=0, B=1 usw.	keiner	MVI E,DISKNR Disk-Nr.#in Reg.E MVI C,SELDSK SELDSK=14	
15	Open File (oder in 255H wenn File nicht CP/M gefullt)	FCB-Adr.der zu offnend. File	Byte-Adr. von FCB im Disk Direct.	LXI D,FCB (File Contr.Buff.) MVI C,OPEN Adr.in Reg. D&E Call NTRY OPEN=15	
16	Close File	FCB-Adr.der z. schliess. File	wie bei 15	LXI D,FCB MVI C,CLOSE CLOSE=16 Call NTRY	
17	File suchen	FCB-Adr.mit Name & Typ der File	wie bei 15	LXI D,FCB MVI C,SRCH SRCH=17 Call NTRY	
18	weiterer glei-	wie 17	wie bei 15	LXI D,FCB	

	cher Eintrag (darf nur nach 17 ausgeführt werden)			MVI C, SRCN Call NTRY	SRCN=18d
19	File Loschen (wird als geloscht markiert)	wie 17	keiner,FCB im Direct.	LXI D,FCB MVI C,DEL Call NTRY	DEL=19
20	Record Lesen (gelesene Daten gehen in DMA-Adr.siehe 26)	FCB-Adr.mit Disk-Map & Rec.Nr.	0=OK,1=ub. EOF geles. 2=Fehler	LXI D,FCB MVI C,READ Call NTRY JNZ ERROR	READ=20 Fehler-Routine
21	Rec.Schreiben (2=End of Disk Data;255H=Directory voll; Daten ab der DMA-Adr.)	wie bei 20	0=OK,1=Fe. ext.File	LXI D,FCB MVI C,WRT JNZ ERROR	WRT=21 Call NTRY Fehler-Routine
22	File Erstellen	FCB-Adr.d. neuen File	Dir.-Adr. (255H wenn voll)	LXI D,FCB MVI C,CREAT Call NTRY JM VOLL	CREAT=22 Directory-Voll-Rout
23	Rename File neuer Name in weiteren 16 Bits)	FCB-Adr. mit Name & Typ in 16 Bits und voll)	Dir.Adr. (255H wenn voll)	LXI D,FCB MVI C,REN Call NTRY JM NOFIL	REN=23 No-File-Routine
24	Disk Log In Abfrage	keiner	Byte mit je 1 Bit gesetzt für jede aktive Disk, LSB=A usw.		
25	Disk-Nr. Abfrage	keiner	Disk-Nr. für nächsten Zugriff		
26	DMA-Adr. angeben	Adr. des 128Byte. % Puffers	keiner	LXI D,BUFF MVI C,DMA Call NTRY	Buffer-Adr. DMA=26
27	Zuordnung abfragen (wird von STAT benutzt) wurde nicht weiter verfolgt.				

Funktion 7 und 8 must be implemented. NTRY=0005 (siehe Listing)

Typisches Listing:

```

PUSH REGISTER ;Register retten, nicht diejenigen, in denen ein
                Wert zuruckerwartet wird.
MVI C,FUNKNR. ;die Funktionsnummer muss im Reg.C sein bevor der
                "CALL" erfolgt.
MOV E,A       ;Ein Zeichen wird am besten im A-Reg. (Akkumulator)
                rausgeschickt, muss aber vor dem "CALL" in's E-Reg.
CALL NTRY     ;der CP/M Eingang muss vorher als 0005H definiert
                sein.
POP REGISTER  ;Eingangs gerettete Register restaurieren. Gleich
                viele POPs wie PUSHs verwenden!
RET          ;Ruckkehr zur Aufrufenden Routine.

```

"As far as I am concerned, C will never replace Cobol and Dibol, it might rival FORTRAN. For commercial applications it is to round about. The much praised "structuring" of C is not worth much more than with Basic, for one can define the despised "GOTO" in C. Comments in C are cumbersome to enter. The most hampering (not to say idiotic) feature is the lack of separate compilation which presents immense extra work and requires a good Editor with the capability to merge sources. The much praised "toll functions" loose much of their attraction, for any change in it requires a recompilation of all programs calling on it! (Modula-II does away with this nonsense). Of course, one can write very condensed (and incomprehensible) code in C. Error messages are a pain and need years of experience to cope with them. The portability is not much better, may be worth than with BASIC (all 4 compilers mentioned needed changes in a very short program of the sieve of erasthenes to compile). C has some advantages over other programs as far as assembler code can be used and hardware could directly be programmed (which in general is not a feature with commercial applications). Despite of the draw backs. I keep learning C until Modula-II will be available.

"Here are some tips for CP/M users: To test the printer or screen with Escape Sequences, you type

>PIP LST:=CON:

Which connects the printer with the Keyboard from where you now can issue characters until Ctrl/Z is received, e.g. This way you can use the printer as a Typewriter, and typing

ESC [2 _" Z (ESC=Escape Key)

sets the LA50 to pitch-12. The forgotten TYPE command to the Monitor is very useful to list text files on the screen, e.g. you type:

TYPE NAME.EXT

This file can also be an escape sequence."

NOTE FROM HUNGARY

A note came recently from A. Szabo asking me to put him in touch with the authors of Pascal-OS/8. He might be able to offer some help in implementing KT8A 17 bit addressing support as mentioned in Newsletter 46.

He says he enjoyed the article on the OS/8 software family because he has worked with it since the early PS/8 days so my article was like a look back into youth for him. Indeed, I think I recall his name from many years ago. If my records were not packed away I would look back to see. It is really something to know OS/8 goes back that far even in Hungary. In case there are any other readers in that part of the world, the address is H-1035 Budapest III., Vorosvari ut 3. V. 28., Hungary.