TEK MICROCOMPUTER DEVELOPMENT PRODUCTS

USER NEWSLETTER
VOLUME II, ISSUE NO. 3, DECEMBER, 1983

USER GROUP NEWS

Tektronix®
COMMITTED TO EXCELLENCE

PRODUCT INFORMATION SECTION

TEK INTRODUCES THE 6140 DEVELOPMENT SYSTEM

The 6140 from Tektronix is a complete software and hardware Microprocessor Development System for one or two users. The 6140 consists of the 8561 (two-user s/w development station), the 8540 Integration Unit, the new Tek 4105M Color Graphics Terminal and a complete support package for the microprocessor of your choice. Also, this quality system comes standard with Tek's ColorKey+ User Interface which makes designing with the 6140 an enjoyable and productive experience.

The 6140 is offered for the Z80, 8085, 6800, 6802, 6809 and NSC800 microprocessors.

NOT JUST AN 8 BIT SYSTEM

The 6140 can support all of Tek's s/w and emulation packages. These support packages include the 8086/88, 80186/188, 68000/008/010, Z8001/2, 9989/9900 and more.

COLORKEY+ USER INTERFACE

ColorKey+ is a special color user interface for the 8560 and 8561 Multi-User Microcomputer Development Systems. Its advanced use of color coding and other graphics features provide you with an exceptionally fast learning curve and a valuable reference tool. ColorKey+ is a standard part of the 6140 TNIX V2.0 operating system, and is designed primarily for use with the Tektronix 4105M Color Terminal, although many of its features are operable on other terminals.

COLORKEY+

Color has a strong and proven track record for bringing increased productivity to computer-based user interfaces. The benefits of coding displayed information in color fall into three main groups. First, color allows quick discrimination between different types of displayed data. Second, color reduces the chances of error due to misinterpretation. Third, color reduces user fatigue.

ColorKey+ takes advantage of the intensive ergonomic research conducted at Tektronix and other research sites to determine the optimum application of color to man/machine interfaces. The overall ColorKey+ display is subdivided into specific functional areas, each with color coding geared to optimize user interaction. For instance, the user's work area is coded with white characters on a blue background to clarify the text, minimize glare problems and reduce eyestrain. The soft key labels use black characters on a yellow background for quick reference and easy readability. TNIX messages use red since this color demands immediate attention.

COLOR**KEY**+

ColorKey+ uses a "programmed key" format that simplifies command entry and keeps the number of required keystrokes to an absolute minimum. The user is presented with a set of "current key labels" across the bottom of the display that are associated with eight "soft" keys on the keyboard. Further, the soft key system is arranged in a hierarchical manner for simple and systematic access to all the 856X's software development and debug tools. For instance, the top layer of the hierarchy presents current key labels to select the major tools within the system. Once a tool has been selected, the next layer presents key labels naming the major functions of the tool. The next layer might define specific operations within a selected function. In this manner, each user has a simple, self-explanatory path through the entire system.

Also, ColorKey+ adds an extra level of intelligence to the soft keys which makes user interaction faster than ever. Whenever you are prompted for a command parameter, ColorKey+ retains your input. The next time you use the command, your last parameter entry will appear as one of the function keys. You now have the option of using the function key or typing in a new parameter, if needed.

COLORKEY+

A good user interface lets you start work on a design project with no advance knowledge of the microcomputer design system. A better interface teaches you the specifics of the system while you interact with it. The best interface allows you to freely intermix interface commands with actual system commands. This way, you benefit from the tutorial aspects of the user interface, but can opt for the efficiency of system commands at any time you wish.

ColorKey+ gives you all these capabilities. For instance, if you are entering commands via the user interface keys, ColorKey+ will display the actual TNIX commands that would perform the same operations. As you become more familiar with the command set, you can enter TNIX commands at any time to interact directly with the system, and also use ColorKey+ when you need to.

8561 Software Development Station

The Tektronix 8561 Software Development Unit provides a powerful and complete set of microcomputer design tools to the smaller design team while accommodating future expansion through a simple, cost-effective upgrade path. The basic 8561 fully supports two workstations, which may be either standard CRT terminals or Tektronix 8540 Integration Units designed specifically to handle hardware/software integration tasks through real-time emulation. Through a series of upgrade options, this basic package can be expanded to accommodate up to eight workstations.

The basic version of the 8561 includes an LSI 11/23 16-bit processor, 256Kb of RAM, 13.6 Mb hard disk storage, 1 Mb of flexible disk storage, two user ports and two line printer ports. This basic system can be easily upgraded within the same mainframe to up to eight user ports and 35.6 Mb of hard disk storage and 1 Megabyte of main memory.

All versions of the 8561 run under TNIX*, an exceptionally powerful operating system derived by Tektronix from Bell Laboratories' UNIX**. TNIX is dedicated specifically to supporting the microcomputer design process.

8540 INTEGRATION UNIT

The 8561 is designed for easy interfacing with the Tektronix 8540 Integration Unit, which provides real-time emulation for the entire range of Tektronix 8-bit and 16-bit chip support. The 8540 comes standard with 64K of user memory when ordered as part of the 6140 system. Code developed on the 8561 is downloaded to the 8540's program memory, up to 256K, for execution on the emulator processor. Execution takes place under control of powerful debug software, and the resulting data can be passed to other TNIX commands for additional processing. For in-depth analysis of real-time code execution, an optional Trigger Trace Analyzer includes sophisticated triggering to capture program flow in a high-speed memory buffer.

* TNIX is a registered trademark of Tektronix

** UNIX is a registered trademark of Bell Laboratories

Bob Ferguson, Product Marketing Manager

HIGH-LEVEL LANGUAGE FOR 6140

To further support the 6140 Development System, MDL/u, Modular Development Language for Micros, is now available on the 8560 for 8-bit microprocessors, 8080, 8085, and Z80/NSC800. MDL/u, based on the BASIC language, has been expanded specifically to support microprocessor-based product development.

The extensions include:

- Direct access to I/O ports and addresses
- Interrupt handling
- Modularity Features - Separate compilation of modules and combination with assembly language modules
- Procedure definition to execute an algorithm as well as function definition to return a value
- Extensive support for manipulation of strings
- Logical operations on integers as well as shift and rotate operations for bit manipulation
- Descriptive variable names

We are offering two products on the 8560/61 development systems: MDL8085 for 8080/8085 and MDLZ80 for Z80/NSC800.

Included with the MDL/u product is a Class C program, Rational, which provides program execution constructs for MDL/u. It runs as a preprocessor to the MDL/u compiler. See the following article, RATIONAL PROGRAMMING LANGUAGE.

Marilyn Hanson, MDP Product Marketing

RATIONAL PROGRAMMING LANGUAGE

A NEW 8 BIT DEVELOPMENT LANGUAGE FOR THE 8560(1)

Tektronix is announcing the availability of the **Rational** preprocessor for the **Modular Development Language** compiler (**MDL/u**). The preprocessor and the compiler are available for the 8560 and 8561 development systems. The source for the libraries will be provided to allow the user to tailor the language implementation to the needs of the design. While floating point capability is not provided, extensive support is provided for:

- the manipulation of strings (up to 255 characters long) including string arrays,
- 8 bit and 16 bit integers and integer arrays including boolean operations,
- interrupt handling,
- and direct access to I/O and memory.

Separation of code, constants, and variables provides prototype configuration capability. Code generated may be easily combined with code generated from assembly source.

Since the control constructs of **MDL/u** are somewhat limited, we have developed a preprocessor called **Rational**. **Rational** provides complete set of execution control constructs as well as other enhancements that will be described later. **Rational** produces **MDL/u** source which subsequently can be used with the 8080, 8085, Z80, and future versions of **MDL/u**.

Why another High Level Language?

What makes a high level language suitable or desirable as a code generating tool for a microprocessor design? The following is an outline of some of the more important features that make a high level language useful as a code generation tool.

FITS TEAM DESIGN METHODOLOGY

1. Modularity
2. Self Documenting
    - o Obvious language - Not Convoluted
3. Support Structured methodology

ADAPTS TO A WIDE RANGE OF APPLICATION ENVIRONMENTS

1. Robust Environment Support
2. Prototype I/O Does Not Require Extensive Coding Support
3. Development System I/O supported
4. Easy to configure to prototype memory configuration
5. Easy to configure to prototype i/o configuration
6. Interrupt structure support

ADAPTS TO MIXED METHODS OF CODE GENERATION

1. Easy to combine with assembly language generated code
2. Easy to combine with custom libraries
3. High level language generated code can call asm. level generated code
4. Assembler generated code can call high level language generated code

EASY LANGUAGE IMPLEMENTATION

1. Fundamentals of the language are well understood
2. Extensions are easy to learn
3. Dual Language programming interface
4. Recursion supported
5. Function and procedure calls
6. Effective code execution control
    - o For
    - o While do and do while
    - o repeat until
    - o case
    - o If then else

EFFECTIVE USE OF CHIP RESOURCES

1. Optimization
2. Only required library elements loaded
3. Library Source supplied for implementation optimization

BROAD BUT NOT A LIMITING LIBRARY SET

LIBRARY ELEMENTS MAY BE EASILY REPLACED WITH ELEMENTS MORE FOCUSED ON THE NEEDS OF THE PROJECT.

The following text describes the **Rational** language interface to the **Modular Development Language.**

RATIONAL

LANGUAGE DESCRIPTION

Rational is **MDL/u** except for one aspect. Since control flow is central to any program, regardless of the specific application, the primary task for **Rational** is to conceal this part of **MDL/u** from the user, by producing decent control flow structure.

THE IF STATEMENT

The standard construct **if a condition is true, do a group of things** for example

```
if (x > 100)
{print "error: x > 100" ; err=1}
```

can't be implemented directly in **MDL/u.** Instead a programmer must translate this relatively clean thought into **MDL/u** as follows:

```
10      IF (NOT (x > 100))  THEN 40
20      PRINT "ERROR: x > 100"
30      err = 1
40      ...
```

When a program doesn't work, or when it must be modified, this must be translated back into a clearer form before one can be sure what it does.

RATIONAL eliminates this error-prone back-and-forth translation. A group of statements can be treated as a unit by enclosing them in braces "{" and "}" in **Rational.** This is true throughout **Rational:** wherever a single statement may be used, there can be several enclosed braces.

Rational is a free format language: statements may appear anywhere on a line, and several may appear on one line if they are separated by a semicolon.

THE ELSE CLAUSE

Rational provides an **else** statement to handle the construct **if a condition is true, do this thing, otherwise do that thing.**

```
if (a<b)
  { sw=0;  print a,b}
else
  { sw=1;  print b,a}
```

This writes out the smaller of a and b, then the larger, and sets sw appropriately. The syntax of the if statement is

```
if (legal MDL/u condition)
      Rational statement
else
      Rational statement
```

where the else part is optional.

NESTED IFS

Since the statement following an **if** or and **else** can be any **Rational** statement, this leads immediately to the possibility of another **if** or **else** As a useful example, consider this problem: The variable "f" is to be set to -1 if "x" is less than zero; to + 1 if "x" is greater than 100, and to 0 otherwise. Then in **Rational**, we write:

```
if (x < 0)
  f=-1
else if (x > 100)
  f=+1
else
  f=0
```

In general, the structure

```
if (. . . )
else if ( . . . )
else if ( . . . )

else
```

provides a way to specify the choice of exactly one of several alternatives.

THE SWITCH STATEMENT

The **switch** statement provides a clean way to express multi-way branches which branch on the value of some integer-valued expression. The syntax is:

```
switch (expression)  {

  case expr1:
     statements
  case expr2, expr3:
     statements

       . . .

  default
     statements
```

Each case is followed by a list of comma-separated integer expressions. The expression inside **switch** is compared against the case expression expr1, expr2, and so on until one matches, at which time the statements following the case are executed. If no cases match the expression, and there is a **default** section, the statements with it are done; if there is no **default** nothing is done. (Those familiar with C should beware since the behavior is not the same as the C **switch**).

THE WHILE STATEMENT

RATFOR provides a **while** statement; which is simply a loop: **while some condition is true, repeat this group of statements.** For example :

```
i = 1
while (i <= 100)  {

. . .

i=i+1

}
```

This routine will function similarly to the **MDL/u** statements

```
FOR I = 1 TO 100
. . .

NEXT I
```

The syntax of the **while** statement is

```
while (legal MDL/u conditional)
  Rational statement
```

THE FOR STATEMENT

The **for** statement is another **Rational** loop, which attempts to carry the separation of loop - body from reason for looping a step further the the **while. A for** statement allows explicit initialization, extent, and increment steps as part of the statement. For example:

```
for (I=1;  I<=100;  I=I+2) . . .
```

Is equivalent to

```
i=0
while (i<=100) {. . .
. . .
   i=i+2
}
. . .
```

or the **MDL/u** source

```
FOR I= 0 to 100 STEP 2
. . .

NEXT I
```

However the **Rational** for statement is not a **MDL/u FOR** statement. The **MDL/u FOR** is not supported in **Rational.**

The syntax of the **for** statement is

```
for (init;  condition ; increment)
  Rational statement
```

The init, condition, and increment parameter may be omitted but the semicolons";" must be present. A non-existent conditional is always true, so for (;;) is a loop forever.

THE REPEAT - UNTIL STATEMENT

There are times when it is necessary to have a loop that tests at the bottom after one pass through. This is done by the **repeat - until** statement.

```
repeat {
  RATIONAL statement
} until (conditional)
```

THE RETURN STATEMENT

The standard MDL/u mechanism for returning from a function or procedure is to branch to either **prend** or **frend**. If a value is returnable by a function it must first be assigned to the function name and the branch to **frend**. The **Rational return** statement provides these functions. For example

```
deffn fact (arg)
 if (arg<=1)
   return (1)
 else
   return (fact(arg-1)*arg)
fnend
```

produces the MDL/u source

```
10      DEFFN FACT (ARG)
20        IF (NOT (ARG<=1)) THEN 50
30           FACT=1
40           GOTO 70
50        FACT = FACT(ARG-1)*ARG
60        GOTO 70
70      FNEND
```

In the case of a procedure a return does the following:

```
defpr proc(arg)

...
 return
pread
```

generates this MDL/u source

```
10 DEFPR PROC(ARG)
20 ...

xx GO TO xx+1
xx+1 PREND
```

The last statement in either a function or procedure must be a **return**.

DEFINE STATEMENTS

Any string of alphanumerics can be defined as a name; thereafter, whenever that name occurs in the input it is replaced with rest of the line. **define** is usually used to create symbolic parameters:

```
define MAXLEN 256

dim    A$*MAXLEN
```

INCLUDE STATEMENT

The Statement

```
include test.r
```

inserts the contents of the file test.r into the **Rational** input in place of the include statement.

BREAK AND NEXT STATEMENT

RATIONAL provides a statement for leaving a loop early, and one for beginning the next iteration. **break** causes an immediate exit from the loop; in effect it is a branch to the statement after loop. **next** is a branch to the bottom of the loop, so it causes the next iteration to be done. For example this code will exit the loop if "j" is equal to 111.

```
define TRUE 1==1

        while (TRUE) {
        . . .
      if (j==111)
        break;
        . . .
    }
```

break and **next** also work in other **Rational** looping constructions.

break and **next** can be followed by an integer to indicate breaking or iterating that level of enclosing loop; thus

```
        break 2
```

exits from two levels of enclosing loops, and **break 1** is equivalent to break. **Next 2** iterates the second enclosing loop. (Realistically, multi-level **break's** and **next's** are not likely to be much used because they lead to code that is hard to understand and somewhat risky to change.)

TRANSLATION SERVICES

Any line beginning with "%" is passed through **Rational** unaltered. Otherwise the following translation occurs

```
"=="  -> "="
"&&"  -> "and"
"||"  -> "or"
"!="  -> "<>"
```

Anything preceded by a "@" is a comment and is ignored by **Rational.**

THOU SHALT NOTS

1. Imbed a ";" in a **print** statement and expect it to act like a **MDL/u PRINT** statement unless it is preceded with a "%".
2. Use labels of any sort (especially line numbers).
3. Pass a character or string in a switch expression.

John Owens, Systems Applications Manager

PASCAL LANDS NOW SUPPORTS 68010

TEKTRONIX PASCAL LANDS now supports the 68010 microprocessor. Version 2 of the 8560 Pascal Compiler for the 68000 (PAS68K opt 1A) and Version 2 of Pascal Debug for the 68000 (PDB68K opt 1A) will be available to interface to the 68XXX Emulator which supports 68000, 68008 and 68010. The VAX versions of Pascal 68000 and PDB 68000 will include this support at the time of release.

The Integration Control System, included with the compiler, now accepts 68010 parameters as well as 68000 and 68008. ICS supports the relocatable vector table of the 68010, allowing you to choose the location for the 68010's vector table.

Version 2 of PDB68K will interface to the new 68XXX emulator, (8300E36), with the 68000, 68008, or 68010 probe, as well as the 68000 emulator (8300E26), which was previously available. Version 1 of PDB68K does not work with the 68XXX emulator, only with the 68000 emulator. Customers purchasing the 68XXX emulator or the 68000 to 68XXX conversion package should be sure to use Version 2 of PDB68K.

Version 2 of the Compiler also has improvements in the floating point run-time libraries to dramatically increase the execution speed of the floating point operations.

CUSTOMER UPDATES

All U.S. and International customers under warranty (purchased since September 12, 1983) will receive the new version if they returned the Warranty Registration card, which identifies who is to receive it.

U.S. customers currently under Software Subscription Service for 8560 PAS68K, PDB68K or PLAN68K will receive the new versions automatically. U.S. customers not covered by warranty or SSS may obtain the new versions by ordering the Software Subscription Service. International customers should contact the local distributor or subsidiary for information on software updates.

Marilyn Hanson, MDP Product Marketing

- -

68000/68010 C-LANDS PACKAGE IS AVAILABLE FOR THE VAX

The Tektronix 68000/68010 C Language Development System (C-LANDS) is a software package which supports you throughout the entire software design cycle. 68000/68010 C-LANDS basically consists of four products:

- C Language Directed Editor (LDE)
- 68000/68010 C Compiler and the Integrated Control System (ICS)
- 68000/68010 C Debug
- 68000/68010 Assembler

A linker, library generator, lister, and cross reference lister are also included in the 68000/68010 C-LANDS package.

Language Directed Editor

LDE combines the text manipulation functions of a general purpose editor with the syntax checking function of a compiler. LDE can check the syntax of your C program, thus saving you repeated compiler passes to remove syntax errors.

68000/68010 C Compiler and ICS Support

The Tektronix 68000/68010 C Compiler translates statements written in the C Programming Language into object code for the 68000, 68008 and 68010 processors. The Tektronix C Compiler implements the C Programming Language as described by Kernighan and Ritchie*. The C Compiler package also includes a runtime library, standard I/O library, IEEE floating point support, 68000 assembler subset, linker, library generator, and ICS the Integration Control System. The ICS system is specifically designed to automate the integration of the software written in C to the hardware configuration of the prototype. The ICS program uses a list of hardware and software configuration parameters contained in a prototype description (ICS source file) which is provided by the user. ICS generates the interface code and linker command file to provide memory configuration, interrupt handling, and interrupt and program initialization. An interactive ICSPrompter is also included to create the ICS source file.

68000/68010 C Debug

Tektronix C Debug is a real-time symbolic debugging tool for programs written in C. C Debug (CDB) allows the programmer to use C constructs to:

- Display and modify variables
- Control program execution
- Record and display debug information

The 68000/68010 C-LANDS package is available for VAX** 730/750/780/782 mainframes running VMS** 3.1+ or UNIX*** 4.1bsd. Each component of the C-LANDS package is also available separately.

Diane Wortsmann, MDP Product Marketing

NEW MANUALS ORDERABLE

8086/80186 Assembler Specifics Users B Series	070-3853-01
C 68000 Compiler Users for VAX/VMS Host (PRELIM!)	061-2892-00
C 68000 Compiler Users for VAX/UNIX Host (PRELIM!)	061-2866-00
8560 Series System Ref Manual TNIX Version 2	070-4729-00
Pascal LDE Users Manual for VAX/VMS	070-4854-00
Pascal LDE Users Manual for VAX/UNIX Host	070-4855-00
8500 Series 68XXX Emulator Processor with 68000-A/68008/68010 Prototype Control Probes Users & Installation Instruction Manual	070-4691-00
8560 GPIB Interface Service Manual	070-4475-00

Charlene Eason, MDP Customer Support

---

* The C Programming Language by Brian W. Kernighan and Dennis M. Ritchie

** VAX and VMS are trademarks of Digital Equipment Corp.

*** UNIX is a trademark of Bell Laboratories

KITS FOR VAX MANUALS

Following is a list of kits that are set up for VAX software manuals. The difference between these kits and just ordering the manuals separately is that the kits include all the manuals for the product. For example, the kit for VAX VMS Pascal 68000/08 compiler will get you the Pascal Language Reference Manual, the VAX VMS 68000/08 Users Manual and the Pascal Users Manual and Report (Jensen and Wirth).

ICOM40	UNIX	020-1152-00
	VMS	020-1186-00
PAS68K	UNIX	020-1187-00
	VMS	020-1188-00
PDB68K	UNIX	020-1189-00
	VMS	020-1190-00
PLDEDIT	UNIX	020-1191-00
	VMS	020-1192-00
CCC68K	UNIX	020-1157-00
	VMS	020-1185-00
ASM68K	UNIX	020-1150-00
	VMS	020-1183-00
ASMZ80	UNIX	020-1147-00
	VMS	020-1180-00
ASM8086	UNIX	020-1149-00
	VMS	020-1182-00
ASM8085	UNIX	020-1148-00
	VMS	020-1181-00
ASM6809	UNIX	020-1151-00
	VMS	020-1184-00

Diane Wortsmann, MDP Product Marketing

Z8000 ASSEMBLER VERSION 2.0

Version 2.03-09 of the Z8000 Assembler for the 8560 (ASMZ8K opt 1A) was released August 17, 1983. Version 2.03-09 includes a virtual symbol table for user-defined symbols. To assemble code with large number of symbols, add a -b to the invocation line.

Customers under warranty (purchased the product 90 days prior to revision release date) or who are subscribed to the Software Subscription Service will receive the revisions automatically.

Marilyn Hanson, MDP Product Marketing

~ ~

8086 PASCAL/PDB MOD

The Pascal Compiler (PAS8086 opt 1A) and PDB (PDB8086 opt 1A) for 8086/8088 have been revised to correct reported problems. These mods do not reflect any enhancements to support the full addressing of the processor. Those enhancements will be included in the Version 2 release of the product scheduled for second quarter of 1984.

Customers under warranty (purchased product 90 days prior to revision release date) or subscribed to the Software Subscription Service, will receive the revisions automatically.

The current version numbers are:

Product	Version	Release Date
PAS8086	1.10-05	6-7-83
PDB8086	1.11-00	8-16-83

Marilyn Hanson, MDP Product Marketing

~ ~

MUGL - VOLUME II DISK RELEASED!

The second MDP User Group Library (MUGL) disk has just been released and copies may be obtained at no charge from your local sales office. This volume contains over 40 new application programs to run on your 8560/61. There are lots of graphics pictures and utilities for the new 4105 color terminal, including an "interactive graphics editor" which allows you to easily create nice "color slides" on your terminal. A new disk reader utility allows porting IBM disk files to your 8560. A handy calculator simulator implements a powerful calculator for complex mathematic and business oriented problems. There is a complete mailing list database manager package which we use in MDP for handling the User Group News mailings. For additional information on these and many other new submissions, see the abstracts section of this issue.

MUGL is provided as a service to MDP users for collecting and distributing user contributed software for all Tektronix Microprocessor Development Products. The program works like this:

- All users are encouraged to submit their creations to MUGL, MDP Marketing, PO Box 4600 MS 92-635, Beaverton, OR 97075. All submissions will be considered and are made with the understanding that the software may be placed in the public domain. Please don't send your only copy, as we are unable to return any submissions, whether accepted or not. For your convenience, a software submission form is included in this issue and on each MUGL disk volume. We must have the author's name to consider a submission, but we will withhold it if you prefer not to be contacted by anyone.
- We will generally check out the programs, but no guarantees of any kind will be made. We prefer to have the object, source, documentation, and manual page (as applicable) submitted on a floppy disk, but we'll take whatever you have. If the program warrants it, we can add the documentation.

- All accepted programs will be archived in MDP Marketing and as soon as we have enough to reasonably fill a disk, a new volume will be released. Copies of these disks will be available from your local field offices.
- Annually, we will provide a master listing and index of all MUGL software. Each volume will also include a catalog listing and summary of all software included on that disk.

Here's your chance to obtain lots of neat applications software, for free! However, we need your contributions to keep the program rolling, so send your programs in!

Greg Saville, Software Applications Manager

APPLICATIONS SECTION

4105 TERMINAL RS-232 PINOUTS

The RS-232 pinout for the 4105 Terminal is not in the user manuals, however it is in the service manuals (in very short supply right now). In the interim, here is the pinout:

1 - GND
2 - TX DATA
3 - RX DATA
4 - rts
5 - cts
6 - dsr
7 - GND
8 - DCD
11 - jumper selectable for secondary rts (rs232a vs -c)
12 - secondary carrier detect
15 - t clk
17 - r clk
19 - secondary rts
20 - DTR

Only the pins listed in ALL CAPS are of concern for connecting a 4105 directly to an 8560. For 4105 to 8540, use pins 1 through 8, and 20 bussed straight across.

Greg Saville, Software Applications Manager

8086 PASCAL LIMITATIONS

The current version of 8086 Pascal, V 1.1, is being shipped with the 8500 Pascal Language Reference Manual rather than the 8086 Pascal Language Reference Manual that was shipped with V 1.0. A Technical Note was omitted from this manual stating the limitations of V 1 of 8086 as compared to 68000 Pascal and Z8000 Pascal. The note should read:

Version 1.X of the 8086 Pascal has the following limitations:

- Constants of type ARRAY and RECORD are not available.
- Procedural and functional parameters are not available.
- The standard procedures PACK and UNPACK are not available.

Version 1.X of 8086 Pascal also only supports 64K of code and 64K of data. Version 2, available second quarter of 1984, will support all the features as stated in the 8500 Pascal Language Reference Manual, as well as full memory support.

The 8086 Pascal Data Sheet is correct and a change notice sheet has been added to the manual.

Marilyn Hanson, MDP Product Marketing

~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

8087 MACRO CORRECTION

In the current 8087 macro package, the opcode values for FDIVP and FDIVRP and for FSUBP and FSUBRP are reversed. To correct these bugs, the following lines must be changed:

Line Number	New Line	
124	F._STACK	'ST(1)',11111
145	F._STACK	'"1"',11111
165	F._STACK	'ST(1)',11110
186	F._STACK	'"1"',11110
279	F._STACK	'ST(1)',11101
300	F._STACK	'"1"',11101
320	F._STACK	'"1"',11100
341	F._STACK	'"1"',11100

Greg Saville, Software Applications Manager

~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~

INCLUDING MATH LIBRARIES IN NATIVE C PROGRAMS

Including math libraries in C programs with the optional native C package is not automatic with the normal cc invocation. To gain access to the math functions, use the -lm option on the command line. The following example shows how to compile and run a short example using a square root function.

C program 'sqrt.c'

```
#include <math.h>
main() {
  float i=2.0;
    printf(" sqrt(2.0) = %f\n",sqrt(i));
}
```

Compilation:

```
cc sqrt.c -lm
```

Execution:

```
$ sqrt
 sqrt(2.0) = 1.414214 $
```

The -lm option is not documented under "cc" because it is a loader flag. Use "man ld" or see "ld" in section 6 of the TNIX System Reference Manual. When used with "cc" the -lm option must be at the end of the command line.

Greg Saville, Software Applications Manager

~ ~

TNIX COMMENT LINES ARE NOT IGNORED

In a TNIX command file, a line may become a comment by preceding the text on the line with a ": " (a colon and a space). The content of a commented line however is still evaluated by the shell and then ignored. If the evaluation uncovers a problem, the execution of the command file will be terminated.

```
:  I don't want this line executed.
```

The above line in a command file will fail due to the "'" character in the line.

This feature is quite useful as the following example shows.

```
:  A command file that requires three parameters be passed to it
:
:  ${1?} ${2?} ${3?}
:
MDL $1 $2 $3
```

If the command is invoked without all parameters being defined, the command file will fail and exit prior to executing the "mdl" command and will state which parameter was missing.

The TNIX manual section "Shell Programming" has a good description of other uses of the "${variable - function - variable} capability.

The comment line "I don't want this line executed" can be used as a comment if it is enclosed in quotes as shown here. Thus;

```
: "I don't want this line executed"
```

will not create a problem in a command file.

John Owens, Systems Applications Manager

~ ~

ED AND ACE AND CHARACTERS > 177 OCTAL

Characters in a text file with the eighth bit set will confuse both the "ACE" and "ED" editors. Text transferred from other equipment sometimes has the eighth bit set.

To correct the file so that it may be edited use the following filter.

tr ['\201'-'\377'] ['\001'-'\177'] <source >destination

All occurrences of characters with the eighth bit set will be replaced with characters in the normal ascii range.

John Owens, Systems Applications Manager

~ ~

ICOM40 SHELL SCRIPTS FOR VAX

The following shell scripts may be of help to users running ICOM40 with VAX under UNIX with the Bourne or Cshell.

*** CSH scripts ***

enter:

```
set prompt = EM:
set tmppath = $PATH
setenv ICOMSPEED 9600
setenv ICOMPORT /dev/ttyib
setenv ICOMRETRY 4
setenv PATH /e/cbase/tek/bin:/bin/icom40:$PATH
```

exit:

```
portcu
portcu
setenv PATH  $tmppath
unsetenv ICOMPORT
unsetenv ICOMRETRY
unsetenv ICOMSPEED
unset tmppath
set prompt = :
```

alias:

```
alias enter 'source ~ /lib/enter'
alias exit 'source ~ /lib/exit'
```

*** SH scripts ***

enter:

```
tmppath=$PATH
ICOMSPEED=9600
ICOMPORT=/dev/ttyib
ICOMRETRY=4
PATH=/e/cbase/tek/bin:/bin/icom40:$PATH
PS1=EM:
export ICOMSPEED ICOMRETRY ICOMPORT PATH
```

exit:

```
portcu
portcu
```

```
PATH=$tmppath
ICOMPORT=
ICOMRETRY=
ICOMSPEED=
tmppath=
PS1=$
export ICOMPORT ICOMRETRY ICOMSPEED PATH
```

Only one set would be needed for any one user. The only side effect that the sh script does is that once an environment variable is set, there is no way to unset it except to set it null, so ICOMPORT, ICOMSPEED, ICOMRETRY are all set but set null.

As you should know, the speed for the ICOMSPEED, the port for the ICOMPORT, the retry value for ICOMRETRY, and the path names for the icom40 package and the tek bin package are dependent on the specific users setup. The prompt characters may be anything the user wishes, this just lets them know they are in "emulator mode".

A similar setup could also be made for the setup you are running on your system.

Victor Riley, MDP Engineering

~ ~

8560 INITIALIZATION PROCESS EXPLAINED

This explains how to startup the 8560 and what happens during the procedure.

1) With the HALT/RUN switch in the HALT position, turn both the AC power switch on the rear of the 8560 and the DC switch on the front of the 8560 "on". The AC switch causes the AC ON light to be illuminated. The DC switch causes the green ON light to be illuminated. In a short while, the red led on the hard disk will be lit. This means the hard disk is up to operational speed.

2) Now move the HALT/RUN switch to the RUN position and push the RESTART switch up then release. This action causes the LSI 11/23 processor to "jump" to address 773000 and start executing a program there. This ROM does the Initial Program Load (IPL). The ROM checks the system. If there are problems discovered, an interactive portion of the ROM takes over and allows one to check the system further. See the 8560/61 Installation manual for more information about this process.

3) When the 8560 ROM gets through the system checkout, it then looks for a floppy diskette in the drive to see if it should load from it. If there is no floppy disk, the hard disk is checked. What the ROM is looking for is a block zero "boot" block. This 512 byte program on either the floppy disk or hard disk is loaded into 8560/61 memory at address 0. At which point the ROM "jumps" to address 0 and that program is executed.

4) Should block zero not exist on either the floppy disk or hard disk, the ROM takes this as an error and again goes into an interactive mode so the user can take some action (like calling service). When block zero is loaded correctly, it will take one of two actions. If a floppy disk is used (must be the STANDALONE diskette) the message "fbr filename to boot>" is printed. The user would then enter a valid STANDALONE command (see the 8560 Reference Manual, section 8, STANDALONE). If one is trying to IPL off of the hard disk, the block zero program will load the "/tnix" file. If this file cannot be found on the hard disk, the block zero program will print the message "filename to boot>". Note the lack of "fbr". The block zero program is giving the user a chance to enter a filename other than "/tnix" in which to load. If this occurs, the file "/tnix" cannot be found and this is considered a good time to restore the 8560/61. Should this be the case, see the 8560 System Users Manual, section 8.

5) Now that "/tnix" is loaded, control is passed to it. TNIX will start off the "swapper" routine as process 0 and "/etc/init" as process 1. "/etc/init" then does many housekeeping chores, one of which is to execute "/bin/sh" with the command file "/etc/rc". What is nice about this is, "/etc/rc" can be modified by the system manager to customize the system to specific needs. "/etc/rc" allows the user to run "syschk", but to NOT make changes (only STANDALONE "syschk" will permit changes). The programs "/etc/update" and if on the 8560/61, "/etc/cron" are also started at this time. That is why the user MUST not skip through "/etc/rc"; this process might not be started (see the 8560 System Reference Manual, section 8,

about "/etc/update" and "/etc/cron"). "/etc/rc" will also "mount" any additional hard disks for you (something like - "/etc/mount /dev/hd1 /usr1"). "/etc/init" will make multiple copies of itself; one "/etc/init" for every tty device. This is done by reading the "/etc/ttys". The first column contains either a 1 or 0. The 1 means the tty device is a "login" port. The 0 means the tty device is not a "login" port and therefore do not spawn an "/etc/init" for that tty device. The "/etc/init" is what controls each port on the 8560/61. The second column in "/etc/ttys" will be used by "/etc/getty" for the default terminal speed (see: 8560 System Reference Manual, section 8, on the "getty" process).

6) "/etc/getty" sets the default terminal speed used and sends the "Login: " message to the terminal. At this point, the user enters a correct user name (or incorrect if you like) and if there is one, will prompt for the password. Should there exist a Message Of The Day (MOTD), the file "/etc/motd" is sent to the terminal. MOTD is good for broadcasting messages of general interest to all users. The login process will invoke the program "/bin/sh". This is the program that allows TNIX the machine to work with YOU the human. The "shell" program can do many things for the user and I suggest one read the 8560 System User Manual, section 5 for more information. Next the file "/etc/profile" is executed. This file, should it exist, is good for issuing commands to every user who logs in to the 8560. It might, for example, print the "date" for every one who logs into the 8560. Once the "/etc/profile" has completed, the "/usr/username/.profile" is executed. Every user may have their own profile command file to customize the system to their needs. ".profile" might contain commands to program a terminal for special uses. Others might have it print all the current users on the system.

Doug Roberts, Washington, D.C. Field Office

~ ~

DELAYED EXECUTION OF COMMANDS

One of the 856x's more useful features is the ability to do tasks at some later time, usually in the middle of the night. This is generally accomplished by the use of the 'at' command (part of the Auxiliary Utilities Package). Users may find it difficult to keep track of all the different jobs. Use a command file (/usr/byronl/.later) to put all "later" tasks into a single place:

```
: this file executes every day at 5 AM under control of at
set 'date'

: the following commands are executed every day
cp $HOME/calendar $HOME/cal.backup

case $1 in
        Sat)    uucp /usr/byronl/logfile red!/public/logfile;;

        Sun)    cd /usr/spool/uucp
                cp LOGFILE .LOGFILE
                cp SYSLOG .SYSLOG
                >LOGFILE
                >SYSLOG;;

        Mon)  echo "Send Status Report Today"|mail byronl;;

        Tue);;

        Wed);;

        Thu);;

        Fri);;

esac
sleep 360
```

at 500 /usr/byronl/.later

After the file is created, just start it with the command 'at 500 .later'. Thereafter, the command file will execute each day at 5 AM, execute the commands specified for that day, and re-install itself to run again the next day.

Byron Lunz, MDP Customer Support

- - - - - - - - - - - - - - - - - - - -

LDE HELP SCREEN FIX

The lde help graphics screen neglects to display the function key f5 for the UNDO command. The following sequence will correct this omission.

Login as root:

Make a temporary copy of the file:

cp /usr/lib/lde/lde.4105.init /usr/lib/lde/temp

Run lde on this temp file:

lde /usr/lib/lde/temp

[line feed key] - (parameter entry mode)

F5 - (search string)

[f4 key] - (find it)

Now edit the string: 5param^]h to: 8param f5^]h

(Notice the space in the new string, it is important. Also, lde will respond much slower than you are used to since you are editing very long lines!)

[f2 key] - (write out the new file)

[F2 key] - (exit lde)

Test new file:

cat /usr/lib/lde/temp (press dialog key and verify help screen is correct)

Rename original file and keep for backup:

mv /usr/lib/lde/lde.4105.init /usr/lib/lde/4105.init.old

Put new version in place:

mv /usr/lib/lde/temp /usr/lib/lde/lde.4105.init

Greg Saville, Software Applications Manager
Jim Willey, Santa Clara Field Office

- - - - - - - - - - - - - - - - - - - -

NULL TERMINAL FOR REMOTE 8540/PDB USE

An earlier issue of User Group News mentioned the fact that when running PDB on a remotely accessed 8540, (ie terminal-8560-8540) a terminal or null-terminal connector was still required on the 8540 terminal port. This is required whenever the application program uses 8540 SVC calls to communicate with the console CONI and CONO devices. The following null terminal connector can take the place of the extra terminal.

- Obtain a 25-pin male RS-232 type plug and connect the following jumpers:
- pin 2 to pin 7
- pin 4 to pin 6
- pin 8 to pin 20

This setup, although not official, has been found to work in MDP Marketing. Feedback on field use would be appreciated.

Greg Saville, Software Applications Manager

~ ~

8086 PASCAL NEW/DISPOSE PATCH

There is a problem with the new/dispose command in 8086 Pascal V01.10-05. Until the next revision, the following patch will correct the problem.

- Using lstr or symbolic debug on the final load file, examine location "smalldynqq+0E4" and change it from 72H to 76H.

Greg Saville, Software Applications Manager

~ ~

USING TEK PLOTTERS WITH 8560

Tek 4662 plotters may be attached to an 8560 to draw 4010-style pictures. User Group Library Abstracts contain a list of pictures for a 4105, many of which are in 4010 format.

The plotter, in our configuration, is assigned a dedicated RS-232 line (tty7 in this case). The file */etc/ttys* is configured to disable logins by changing the first digit for the respective tty entry to a zero. The tty line should also be publicly accessible and configured for 1200 baud. The following steps illustrate the above procedure (assumes user is root):

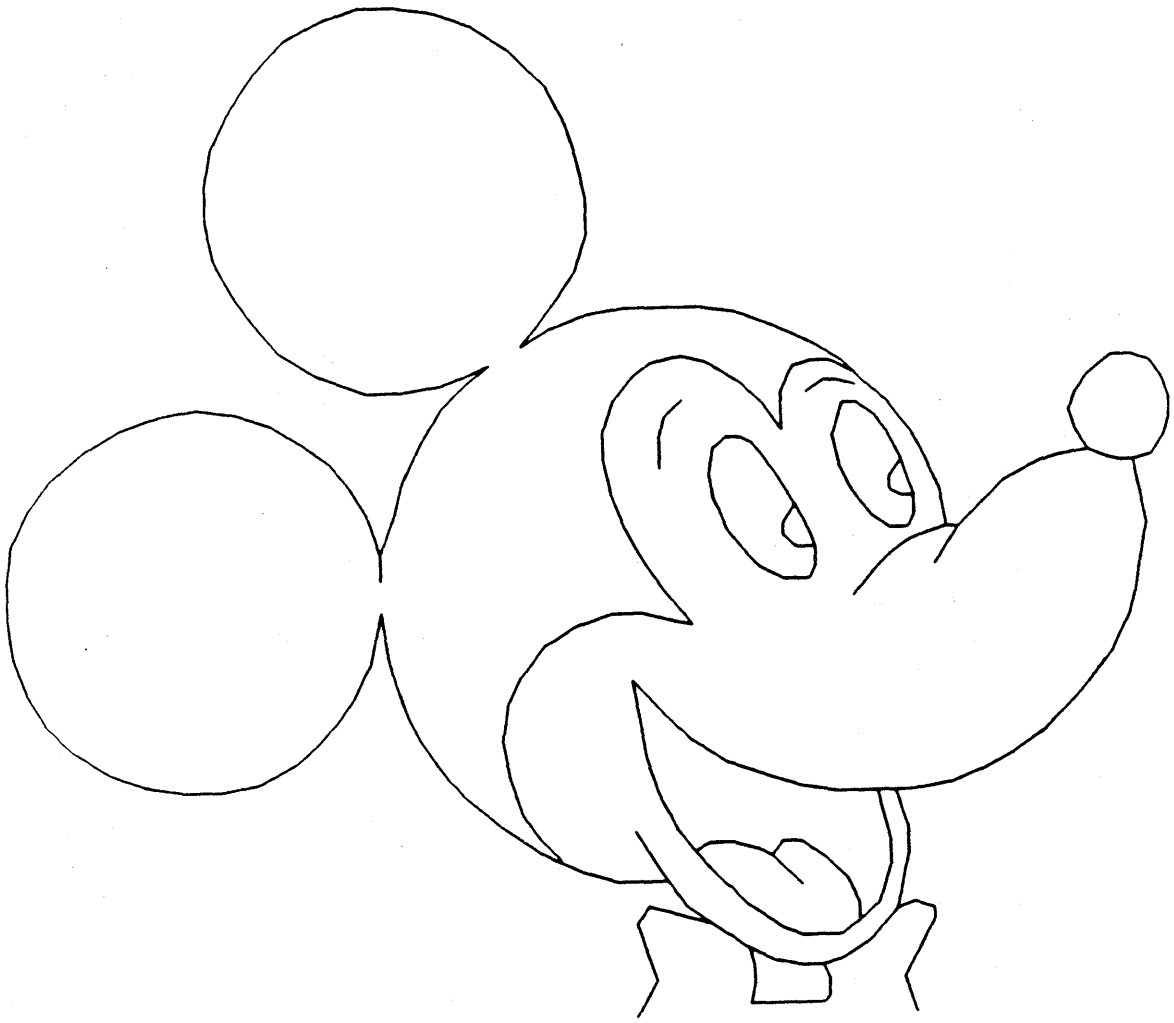
```
ed /etc/ttys
/tty7/p
s/^./0/
w
q
chmod a+rw /dev/tty7
stty 1200 >/dev/tty7
```

The plotter's switches are set to the following settings: A = 3, B = 3, C = 2, and D = 3. After the plotter has been turned on, drawing a 4010-style picture is as simple as:

```
$ cat picture >/dev/tty7
```

which results in the following display:

Example of Tektronix Plotter Display



Mon Li Tang, Rainer Wieland, MDP Engineering

SET USER ID ON EXECUTION

There has been some confusion about the file protection attributes and the use of set user id on execution. This apnote should help clear up any confusion and provides an example to work from which can be adapted for your applications.

There are times when you would like to allow users to execute an application program, but not let them read the actual file. There may be sensitive information in the program that you don't want them to see, but you still want them to be able to execute the file. Your first thought might be to chmod the file to **711** (-rwx--x--x) and chown to root. This appears to permit execution by anyone, but only the owner (root in this case) to read or write the file. If you set up a shell script in this fashion, "others" will get a -sh error message stating: cannot open. This is because they don't have read access to the file. A second attempt might be to chmod the file to **4711** (-rws--x--x) to set the user id to root privileges upon execution (documented under "man chmod"). Again, you will find this doesn't work. What is not clear in the documentation is that the set user id bit applies only on executable binary files, not shell scripts. Fortunately, there is an easy way to allow protection of a shell script application. All you need to do is create a short C program which calls the shell script for execution. Consider the following example:

Our shell script application program will be called "script" and consists of the following:

```
: script - sensitive info
: We may have some sensitive data in here that we don't
: want anyone to see, either in comments or in the actual
: commands. For this example, just a simple command string:
ls /usr/bin | wc -l
```

To try this example, create (as superuser) the "script" file in /usr/bin and "chmod **700** script". Next, "chown root script". This makes the file unreadable by anyone except the owner, root.

Now we need to create a C program which will execute this program with root privileges. Edit the following file in /usr/bin and call it "countem.c":

```
main() {countem.c}
{
system("/usr/bin/script");
}
```

Compile this program with: cc countem.c -o countem. Now set the file attributes with "chmod **4711** countem" and "chown root countem". This allows all users "execute" status for the "countem" command with root privileges upon execution. Test by entering "countem". This will be executable by anyone, however no one (except root) can look at the "script" file to see what it contains.

To summarize, you cannot simply set execute (--x) privileges on a shell script, the user must also be able to read the file in order to execute it. The setuserid (chmod 4XXX) is not meant to work on shell scripts, only executable binary files. Lastly, it is possible to setuserid on a shell script by calling the script from a C program which has the appropriate permission attributes setup as outlined above.

Greg Saville, Software Applications Manager

CREATING A BLACK 4105 SCREEN BACKGROUND

There may be some of you with 4105s and 4695 color copiers that get tired of waiting for the copier to print the blue background of the screen. What follows is a quick fix that will set the default background color to black, which will be transparent when copied. It is possible that there are programs that will reset the screen color to blue; however, this does work with ACE and LDE.

Login as root

```
cd /usr/lib/tabset
cp 4105 4105.old        Make a backup of 4105.
ed 4105                 DO NOT use the editor "p"
                        command! This file contains
                        4105 commands and will
                        reset your terminal if
                        you use "p". Use "l"
                        instead of "p" if you
                        have to display the contents.

s/TF430B3F4/TF43000/    Set dialog color 3 to black.
w
q
```

Anyone who logs in after this will get a black background instead of blue. This is a global change; that is, everyone on the system must have either a black or blue screen.

Barbara Zanzig, MDP Training

- -

UUCP PATCH FOR SYSTEM NAME

Those of you who have been using uucp to send mail to remote systems may have noticed that the sending system is always called "sneezy", rather than the system name defined in whoami.h. The name "sneezy" was hardwired in the compilation of mail. The fix is easy, simply use the patch utility program (available on TNIX MUGL Disk Volume I) and patch in your own system name using the following procedure:

Login as root and make a backup copy of mail:

cp /bin/mail /bin/mail.orig

Run the patch utility:

patch -w /bin/mail

Using the prompts in the patch program, display buffer 105 and determine the address of the string "sneezy" followed by an ascii null (hex 00). Patch your own system name on top of sneezy and be sure to end it with a null. Make sure you don't patch over anything other than "sneezy", ie. don't use a system name longer than the six bytes already allocated. If you choose a shorter name, make sure you end it with a null (hex 00) byte. Write the patch out using the write option of patch and exit.

The uucp installation makes a link from /bin/mail to /bin/rmail. Using "ls -li /bin/*mail*", verify that mail and rmail are indeed the same file with the same inode number.

Login as a regular user and test by sending some remote mail. If all is well, your system will now be known by its proper name.

Greg Saville, Software Applications Manager

- -

CU - VMS FILE TRANSFERS

The cu command as supplied with the "unicom" communications option can be used to transfer files to or from a VAX/VMS system.

Three files are required; two on the TNIX system and one on the VAX/VMS system. The first file on the 8560 is called "sendit" and it in turn calls the second file called "transmit" which uses the 8560 source file as standard input.

Command file Description

The "sendit" command file echos the command to create a file and accept input to the file. "transmit" is then called to read the source file line by line and output the lines (to the host). When the file has been transferred, the command file sends a control D to complete its function.

The TNIX file "sendit" contains:

```
echo "CREATE $2"
transmit <$1
echo ''
```

The TNIX file "transmit" contains:

```
while read AA || exit
do
echo $AA
done
```

The VMS file "SEND.EXE" contains:

```
$ ! send a file to the 8560
$ ! [@SEND.EXE "VMSsourcemame" "8560destname" ]
$ WRITE SYS$OUTPUT ""
$ WRITE SYS$OUTPUT "~ >:"'P2'
$ TYPE 'P1'
$ WRITE SYS$OUTPUT "~ >"
```

File Transfer Methods

When the above files are created, files can be uploaded or downloaded as in the following commented examples:

- VMS to TNIX while in cu to a VMS host

```
                    **      The VMS prompt is a "$"
                    **      Show the current VMS directory contents:
$ DIR
  SRCFILE.TXT
$


                    **      The current TNIX directory contains:
$ ~!ls              **      This command goes to the 8560
  uploadtest
$


$ @SEND.EXE SRCFILE.TXT DOWNLOADTEST
```

```
~>:DOWNLOADTEST  **      Sent by VMS
~>               **      Sent by VMS
                 **      Text transfer is not visible
               **      Note, the file 8560 name created will be upper case.
$
```

- TNIX to VMS while in cu to a VMS host

```
$
$ ~$sendit uploadtest dest.txt
CREATE DEST.TXT  **      Sent by TNIX to VMS
                 **      The text in the file will appear here
                 **      The VMS file name will be uppercase
$ DIR            **      The VMS directory now contains:
  DEST.TXT     SRCFILE.TXT


$~!ls            **      The TNIX directory now contains:
  DOWNLOADTEST  uploadtest
$
```

NOTE, ** <text> is a descriptive comment that does not appear in use.

The file downloaded to the TNIX system will contain a blank line between each text line sent.

John Owens, Systems Applications Manager

~ ~

ACE AND LDE SUPPORT FOR VT100 TERMINALS

Following are copies of configuration files for ACE and LDE. A keypad layout is attached for LDE on the DEC VT100 family of terminals connected to the 8560. Keep in mind that one of the requirements for ACE is the insert/delete character/line capability on a terminal and that is not supported on all DEC terminals. The following terminals are in the VT100 family:

- VT100 -- Works with LDE. Requires advanced video option to give the above insert/delete edit capability for ACE. If the terminal has the printer port option then it also has the advanced video option. The printer port is an extra RS-232 connector located in the middle of the back access cover.
- VT101 -- Works with LDE. The vt101 has no functional upgrade options. Since you cannot add the advanced video option, ACE will not work.
- VT102 -- Works with LDE. The advanced video features are built in so ACE will run with no other options required.

There are many VT100 'look-alikes' that may or may not run with KSH and these editor configuration files. Consult the ACE and LDE manuals for a list of the minimum terminal capabilities required to run these editors.

A disk is now available from your Tektronix Application Engineer containing ACE and LDE configuration files for other terminals ("ACE & LDE Configuration Library"). The following terminals are currently on the disk:

Ann Arbor Ambassador	ACE
Cybernex XL87	ACE LDE
Dec VT100	ACE LDE
General SW10	ACE
HP 2465	ACE

Lear ADM42	ACE
Tab 132	ACE
Televideo 920,950,970	ACE
Zenith Z19	ACE

Dave Heiss, Irvine Field Office.

~ ~

The following is an ACE Configuration for a VT100 terminal:

```
Viewing file vt100k.cfg
Terminal name:vt100:
Cursor up key:^[OA;^[Ox:
Cursor down key:^[OB;^[Os:
Cursor right key:^[OC;^[Ov:
Cursor left key:^[OD;^[Ot:
Cursor home key:^[Ou:
Scroll up key:^[Ow:
Scroll down key:^[Oq:
Scroll right key:^[On:
Scroll left key:^[Op:
Page up key:^[Oy:
Page down key:^[Os:
Revise mode key:^[OM:
Insert character key:^[OP:
Insert line key:^[OQ:
Insert termination key:^[OP:
Delete character key:^[OR:
Delete line key:^[OS:
Command escape key:^[Ol:
Mark cursor key:^[Om:
Erase screen sequence:50^[[;H^[[2J:
Erase to end of line sequence:3^[[K:
Erase to end of screen sequence:50^[[J:
Insert character mode start sequence:OMITTED
Insert character mode stop sequence:OMITTED
Insert character sequence:^[1:
Delete character sequence:^DC:
Insert line sequence:^[[20h:
Delete line sequence:^[[20l:
Cursor x-y positioning sequence:5^[[%i%2;%2H:
File with initialization sequence:OMITTED
Terminal initialization sequence:
^[[0;19r:
Terminal termination sequence:
^[[0;24r:
Cursor up sequence:2^[[A:
Turn cursor on sequence:OMITTED
Turn cursor off sequence:OMITTED
Turn on primary stand out mode sequence:2^[[7m:
Turn off primary stand out sequence:2^[[m:
Turn on secondary stand out mode sequence:OMITTED
Turn off secondary stand out sequence:OMITTED
Turn on monitor mode sequence:OMITTED
Turn off monitor mode sequence:OMITTED
Mark representation sequence:@:
End of line representation character:~:
Number of usable lines on screen:24:
```

```
Number of physical columns on line:80:
Backspace if not CTL-H:OMITTED
Pad character if not NUL:OMITTED

LDE configuration for VT100 terminal

Terminal Name Comment: vt100
BACKSPACE Key: ^H
CANCEL MARK Key: ^[Os
CANCEL INPUT Key: ^U
CURSOR DOWN Key: ^[OB
CURSOR LEFT Key: ^[OD
CURSOR RIGHT Key: ^[OC
CURSOR UP Key: ^[OA
DELETE Key: DEL
EXECUTE SYSTEM COMMAND Key: ^X
FIND STRING Key: ^[OR
FIND TOKEN Key: ^[Oy
FORMAT Key: ^F
GENERIC Key: ^G
LITERAL Key: LOCATE Key: ^[Ox
MARK/CUT Key: ^[Oq
MORE Key: ^[Op
PAGE DOWN Key: ^[Ow
PAGE UP Key: ^[OP
PARAMETER KEY: ^J
PARSE Key: ^P
PASTE Key: ^[Or
QUIT Key: ^[Ov
REDO Key: ^[Ou
REPAINT Key: ^[On
REPLACE STRING Key: ^[OS
REPLACE TOKEN Key: ^[Om
STATUS Key: ^[OQ
UNDO Key: ^[Ot
UPDATE Key: ^[Ol
INFINITE REPETITION Key: .
SEPARATOR Key: ,
Terminal Initialization File:
Terminal Initialization String: ^[[?1h^[=
Terminal Termination File:
Terminal Termination String: ^[[?1l^[>
```

KSH KEYPAD CONFIGURATION FOR VT100

f1	f2	f3	f4
f5	f6	f7	f8
HELP	WHERE AM I	EXPAND KEY LAB	EXPLAIN KEY LAB
REDRAW SCREEN	HISTORY FORWARD	HISTORY BACK	EXECUTE COMMAND

ACE KEYPAD CONFIGURATION FOR VT100

INSERT CHAR	INSERT LINE	DELETE CHAR	DELETE LINE
SCROLL UP		PAGE UP	MARK
	CURSOR HOME		COMMAND ESCAPE
SCROLL DOWN		PAGE DOWN	
SCROLL LEFT		SCROLL RIGHT	REVISE

MDP USER GROUP SOFTWARE LIBRARY/ARTICLE SUBMITTAL FORM

The following form may be used to submit software which you feel might be of interest to other MDP users.

MDP USER'S GROUP SOFTWARE LIBRARY/ARTICLE SUBMITTAL FORM

1. ABSTRACT. _____

_____

_____

2. Execution CPU _____ Primary Language _____

Hardware configuration required _____

_____

Software configuration required (include source if non-Tek) _____

_____

_____

3. Do you want the following to appear in U.G.N.

Author's name _____ O yes _____ O no

Company Name _____ O yes _____ O no

Area code _____ Tel. No. _____ O yes _____ O no

Company address _____

_____

_____

4. Program Title _____

Program Function _____

5. Source. If insufficient room is provided, please submit a disk (containing the information requested) attached to this form.

6. I am submitting the program/article described above for possible placement in the MDP User's Group Library. I understand there is no further compensation due to me, other than a free one year renewal to USER GROUP NEWS for an accepted program/article. This program/article is of my own design, the data contained in this submittal is not copyrighted and does not break any obligation to another person or organization relating to proprietary or confidential information. Tektronix, Inc. is authorized to distribute (free of charge on customer supplied media) or publish copies of this program to Tektronix MDP users.

Signature _____ Date _____

THIRD PARTY SOFTWARE

INTEL COMPATIBLE PL/M 8085 DEVELOPMENT SYSTEM

FROM TEKTRONIX

Through a cooperative marketing arangement with Caine, Farber, & Gordon Inc. (CFG), Tektronix now offers an 8085 development system that supports full Intel compatible PLM and assembler software development. With this system from Tektronix and CFG you and your design team can at last step up to the productivity and capacity of Tektronix development systems, gain the support of other vendor's micros, and most importantly retain your PLM-based software investment.

The PLM 8085 Development System consists of (1) Tektronix' 6140 8-bit Color Microprocessor Development System, (2) Tektronix' Intel COMM Package and (3) CFG's PL/M Software Tools. The 6140, Tektronix's multiuser development system, is described elsewhere in this issue. The Intel COMM Package provides facilities for transferring your project to the 6140 from your MDS design environment. CFG's PLM Software provides tools for you to develop 8085 PLM and assembly modules, link them, and download the object to the 8540 for emulation and symbolic debugging. CFG prior to developing this PLM package, developed Intel's PLM compilers for both the 8085 and 8086. Taking this experience they have developed this new PLM to be 100% compatible and at least 10% MORE efficient in code size. The PLM 8085 Development System can provide many benefits to you and your design team, some of which are:

Preserved Software Investment. Because CFG's PLM is 100% compatible with Intel's, you can continue to use the PLM, assembly, and object libraries developed on your Intel MDS. Also your PLM-trained designers can continue using a familiar language. Because CFG's PLM generates code at least 10% more compact than Intel's, you can add new software features to products that had used up available memory. With CFG's PLM on the 6140 you can enhance existing products developed with Intel PLM or develop new products in your line of 8085-based designs. Your investment in 8085 PLM software will be preserved and continue to provide value for future designs.

Increased Design Productivity. You can move to Tek's highly-productive, team-oriented design environment from a single-tasking, single-user system. For example, consider the task of implementing a change to your software system. Simply edit the desired changes and, with the build control tools of the 6140, issue a single command to rebuild your entire system. Contrast that with the process of compiling and assembling one at a time on your MDS with no use of the system for other tasks. The PLM Software also offers features to increase your productivity: language extensions that simplify your programming of new modules, compatibility with the Tektronix Linker for use with Tek software, and support of key z80 instructions so you can develop z80 applications.

Flexibility in Support. By choosing Tektronix for Microcomputer Development Products you are no longer locked into the support features of a micro vendor like Intel. With the extensive emulation capability of the 6140 you easily and economically add support for any of the other 25 micros supported by Tektronix.

AVAILABILITY

The PLM 8085 Development System is available now.

Ordering Info.	Description	Vendor to Order From
856140B	6140 for 8085	Tektronix
8560U04	Intel Comm SW	Tektronix
80/DS	PLM 8085 SW	Caine, Farber, and Gordon

A reprint of CFG's Licensed Program Description for their PL/M 8085 Software is at the end of this issue.

Rodney Bell, MDP Product Marketing Manager

- -

REFERRAL SERVICE HIGHLIGHTS

INTEL-COMPATIBLE SW.

8560/40 COMPATIBLE. PRODUCTS and TEST SITES.

8560 COMM PACKAGE.

CONFIGURE 8560 WITH VMS, IBM, AND OTHERS, FROM HOLOS.

JOVIAL 1750 SUPPORT.

PSS OFFERS JOVIAL 1750. 8540 COMPATIBLE. JOVIAL DEBUG UNDER DEVELOPMENT.

8-BIT HLL ON 8560.

C & PASCAL. Z80, 8085, & 6809. SOME TEK ASM COMPATIBILE. DEMO DISCS.

REAL-TIME OPERATING SYSTEM KERNEL.

VRTX FROM HUNTER & READY. RMX-86 ALTERNATIVE

THIRD PARTY SW DEMO DISCS AVAILABLE FOR SOME PRODUCTS.

PRODUCT UPDATE

INTEL-COMPATIBLE SW

PL/M from Caine, Farber, Gordon

Caine, Farber, and Gordon (Kent Gordon 213-449-3070) is offering PLM compilers for the 8085 and 8086 for the 8560. The compilers are 100% Intel compatible, produce less code than the Intel compilers, and support assembly-level symbolic debugging. CFG is now selling for the 8560 their 8085 PLM with an Intel-compatible assembler/linker, or alone to work with Tek assembler/linker. Their 80/PL supports the 8080, 8085, and z80. The 8086 PLM is available for test sites or trial use. It interfaces to the Tek linker. CFG's 86/PL supports the 8086, 8087, 8088, and 186. Evaluation licenses are available from the vendor.

Caine, Farber, and Gordon also offers these products on UNIX systems and VAX-VMS (also through First Systems; see below). The 86/PL uses either Tek linker or an Intel-compatible linker such as from SSI or First Systems. A complete Intel-compatible 8086 PLM language system that interfaces with the 8540 consists of CFG's 86/PL and an Intel-compatible assembler/linker/locator from SSI or First Systems. The VAX-VMS version of CFG's 86/PL also interfaces with Tek 8086/186 Assembler/Linker. Both of these systems are available for field test. Contact your local Sales Engineer for arrangements. See **VENDOR UPDATE & INTERNATIONAL DISTRIBUTORS.**

8086 Asm/Link from SSI

Systems and Software Inc. (Paul Chien 714-241-8650) announced the availability of REX-SMA-186, an Intel-compatible 80186/87/88 assembler package for VAX-VMS systems. It is 100% compatible with the Intel assembler/linker/locator/librarian for 8086, 8087, 8088, 186. It interfaces to either Intel ICE or Tek's 8540. The 8540-compatible version is available for field tests with CFG's 86/PL compiler (see previous article). Contact the vendor for references, prices, and terms. See **VENDOR CONTACTS, VENDOR UPDATE.**

8086 Asm/Link from First Systems

First Systems (Mary Avera 213-546-5581) offers MicroSET-86, an Intel-compatible 80186/8086 assembler package for VAX/VMS. There are numerous installations of the product which has been in the market for over a year. Available with MicroSET are the following, all for 8086:PLM, Pascal (also on IBM 370), C, and FORTRAN. The vendor also offers MicroSET-80, PLM/assembler/linker support for 8085, Z80, and NSC800 on the VAX/VMS. The vendor is currently implementing compatibility for these products with the 8540. See **VENDOR UPDATE.**

16-BIT C FOR 8560

C 8086 from Lantech

Lantech Systems Inc. (Joni Bullington 214-340-4932) offers a C 8086 compiler. The company is verifying compatibiliy with the 8560/8540 in cooperation with the Tek Dallas Field Office. This high-quality compiler has numerous installations in over two years of sales. Contact Lantech for references, prices, terms, and possible demonstration. See **VENDOR CONTACTS.**

C 68000 from Alcyon

Alcyon (Bill Allen 619-578-0860) offers a C 68000 compiler that optionally emits Tek assembly code. It has been available on the 8560 for nearly 2 years. Contact Alcyon for references, prices, and demonstration disc.

C 8085, 68000 and 8086 from Whitesmith

Whitesmith's (Don Watson 617-369-8499) products, available on the 8560 for 2 years, can be interfaced to the 8540 using an "a.out" to TEKHEX converter. This converter offers symbolic debugging of C programs with the 8540. It is available from your local Tekronix Application Engineer. Order PDP-11 UNIX version of Whitesmith product to be shipped on RT-11 single-sided, single-density diskettes. Use the RT-11 disk reader, available from your local Tektronix Application Engineer, to install the software. See also **COMPATIBIITY HINTS** and **VENDOR UPDATE.**

C 8086 from Telecon

This C 8086 compiler from Telecon (408-275-1659) is available in source form for ease of use with Tektronix systems. See **VENDOR CONTACTS.**

8-BIT HLL FOR 8560

C z80 from Interactive Systems

Interactive Systems (213-450-8363) offers a C z80 compiler that is one of the best compilers in the market (see *Selecting a Programming Language,...*, **IEEE Computer**, p. 29, August 1982). There are several installations on 8560s. The product is available on 8560 media. Contact your Tektronix Sales Engineer for references. See **VENDOR CONTACTS.**

C 6809 on 8560 from Introl

Introl Corp (John Wisialowski 414-276-2937) offers a C 6809 cross-compiler on PDP-11 UNIX and 8560 systems. It sells for $1500; specify 8560 media when ordering. The compiler accepts Kernighan & Ritchie C (the Bell Labs standard) and generates among the best code of any 6809 compiler available. The product comes with a 6809 assembler and optionally produces Tek Assembly. Contact the vendor or your local Tektronix Sales Engineer for demonstration and references. See **EVALUATING 3RD PARTY SW.**

C and Pascal from Real Time Systems (Europe)

Real Time Systems (Alan Cleary, UK, 0632 732531/732639) offers C and Pascal compilers for 8080/8085, z80, 6809, and 9900. They come with an assembler and optionally generate Tek assembly language. Versions for the 6809 and 9900 and for the Tek Assembler are available for field test on 8560 systems. The software is derived from Whitesmith's products, which RTS also distributes in Europe. Contact the vendor for references, prices, availability, and demonstration possibilities. Distribution of 6809 compilers in the US is by JMI SW (215-657-5660). See **VENDOR CONTACTS, VENDOR UPDATE, & INTERNATIONAL DISTRIBUTORS.**

C z80 on 8560 from Van Data

Van Data (Dwight VandenBerghe 206-542-7611) may port this compiler, now available on DEC Minis, to the 8560 for prospective sale. Contact the vendor to order this special version.

REAL-TIME OPERATING SYSTEMS

VRTX z80 from Hunter & Ready

Hunter & Ready (415-326-2950) now offer VRTX for the z80. VRTX is already available for 68000, 8086, and z8000. VRTX is small, high-performance, reliable, real-time, multi-tasking operating-system kernel. It is currently provided in a loadable form that is compatible with Tektronix development tools with minor adaptation. Program Tek assembly routines to match the parameter calling conventions of 8560 Pascal with each VRTX function (about 15). Hunter & Ready also now offers TRACER, a debugging tool for VTRX-based systems, for the 68000 and 8086. See **VENDOR UPDATE & VENDOR CONTACTS.**

MTOS 68K from IPI

Industrial Programming Institute (Bernie Mushinski 516-938-6600) now offers their MTOS 68000 in Tek Assembly source for full compatibility with Tektronix' development tools for the 68000. MTOS is a multi-tasking real-time operating system with support for memory management and multi-processors. There are also options for a file system and floppy disk driver. See **VENDOR CONTACT.**

pSOS 68000 from Software Components Group

Software Components Group (Alfred Chao 408-923-2741) offers a high quality real-time multiprogramming, multitasking, operating system kernel for the 68000. pSOS is provided in loadable, object form that can be used with 8500 products. There is a companion file system FILES-68K. The company is working on additional operating system functions for the 68000. See **VENDOR CONTACT.**

UNIX APPLICATIONS

Viewcomp from UNICORP

This spreadsheet is available for the 8560 from UNICORP SW (Ed Lieb 212-307-6800). Different terminals are handled in a way compatible with termcap. The cost is $500; specify 8560 media when ordering. Contact the vendor for product info, manual, or demonstration possibilities. The product is used within Tektronix on 8560's and VAX's.

SCCS from Santa Cruz Operation and UNIQ Computer

Source Code Control System is available for $560 from Santa Cruz Operation (Doreen Hamamura 408-425-7222). SCCS is also available for 8560 from UNIQ Computer (Sam Bishop, 28 S. Water, Batavia, IL 60510). Specify 8560 media when ordering.

The Bridge on 8560 from Virtual Microsystems

Virtual Microsystems (415-841-9594 Ross Charney) offers The Bridge, a CP/M simulator for UNIX systems. There are 5-10 installations on 8560s. The Bridge allows such popular CPM programs as Wordstar, dBase II, Visicalc, Multiplan, Supercalc, Milestone (project management), and Datebook (personal calendar) to run on the 8560.

8550 SOFTWARE

ProForth from Microsystems Inc.

Microsystems Inc (Bob Hertel 213-577-1471) offers this Forth product for the 8550 and 8002.

VENDOR UPDATE

BSO Products for 8560

Boston Systems Office (Fran O'Brien 617-894-7800) now offers Pascal Z80 and 6809 on VAX/VMS and PDP-11/RSX. BSO already offers Pascal support for the 68000, 8086/186. For compatibility with Tektronix' 8540 Integration System, BSO offers download and Extended TEKHEX support. BSO also offers their assembler/linker/simulator products on the 8560.

Green Hills' 68000 FORTRAN, C, & PL/M

Green Hills Software (213-796-6543) offers FORTRAN, C and PL/M for the 68000 on a VAX/UNIX. They cost $6000; compilers for VMS are available by special request. Output is either UNIX or Motorola format.

JMI SW Offerings

JMI Software (Ed Rathje 215-657-5660) has expanded their software offering to include 16032 support (both C compiler and C Exec), a Basic to C translator, and distribution of C compilers from Real Time Systems. JMI also offers C Exec, a UNIX-like, real-time monitor, for the major micros. JMI distributes in the US the C 6809 compiler from Real Time Systems for $1550. It runs on DEC minis and the 8560. The 8560 version generates assembly compatible with Tek's 6809 assembler. Real Time Systems offers this compiler in Europe. See **PRODUCT UPDATE** for more information.

Real Time Systems Products

Real Time Systems (Alan Cleary 0632-733131, telex: 53429 PACE G, UK) both distribute software products and market their own adaptations of some of these products. They distribute for US vendors Whitesmith, JMI, and (currently?) Lantech. These include C and Pascal compilers for 8080, z80, 6502, 8086, 68000, and 6809 and the C Exec for many micros. The compilers run on DEC minicomputers and the 8560. RTS has adapted the Whitesmith products to (1) support other processors such as 6809 and 9900 and (2) interface to the Tek linker. 8560 versions for 8080, z80, and 6809 produce Tek Assembly language. RTS is seeking 8560 test sites for the 8080, z80, and 9900 versions. RTS also offers cross-assemblers for most micros on DEC minis and UNIX systems.

NEW PRODUCTS

Comm60 from Holos

Holos Corp. (Tom Ness 404-373-0110) offers a communication package for the 8560. Comm 60 is a $1000 general purpose, configurable, communications package which connects the 8560 user to a remote host such as VAX/VMS or IBM. It offers functions similar to Tektronix' 8540 COM option and 8560 UNICOM option including transparent comm downloading files. It is configurable to a variety of hosts via a series of prompts for parameters and control characters. A demo version of the product is available from the vendor.

16032 Support from JMI

JMI Software (Ed Rathje 215-657-5600) now offers their C Executive, a ROM-able real-time monitor, for the NS16032. C Executive is also available for LSI-11, MC68000, 8080/85, 8086/88, and Z80. They also offer a C 16032 compiler on VAX (VMS and UNIX) for $1500. The 16032 assembler/linker package is available separately for $500.

RTOS-80 from Microsystems, Inc.

Microsystems, Inc. (Bob Hertel 213-577-1471) now offers a real-time operating system for 8080/8085 and z80. RTOS-80 is compatible with the vendors proFORTH system which is available on the 8550 and 8002A.

BASTOC from JMI

JMI Software Consultants (215-657-5660) offers a Basic to C Translator for any system that supports C. It supports ANSI 78 BASIC with most Microsoft and some CBASIC extensions. BASTOC can be tailored to BASIC dialects, possibly including Tek's 8550 MDL/u. See **VENDOR UPDATE.**

TMS 320 Asm on 8560 from PH Assoc

PH Associates (Ronald Herold 703-281-5762) offers an assembler for the TMS 320. It runs on the 8560, PDP-11, and CP/M systems. It is very fast and supports TEKHEX (not relocatable). The price is $1000. TMS 320 from Texas Instruments is high speed single-chip microcomputer for signal processing applications.

C Compilers from Microtec

Microtec (Dave Zemel 408-733-2919) has announced a C cross-compiler for the 8080/85 and z80 running on DEC Minis or available as FORTRAN source for customer hosting. Microtec supports compatibility with 8540 through TEKHEX. No 8560 installations are known yet. Contact vendor for prices, availability, and references.

RTOS in C from Destek

Destek (Dennis Reiger 408-737-7211) offers a Real-Time Operating System kernel in C source (250 byte micro specific code) for $8000 with rights for unlimited binary copies. There is support for hierarchical file system, networking, and debugging. Destek will offer a C 68000 compiler in November with support for other micros to follow. Call vendor for references, product information, prices, etc. See **VENDOR CONTACTS.**

NEW VENDORS

Proprietary Software Systems

Proprietary Software Systems (Joel Fleiss 213-394-5233) offers Jovial compilers for 8086 and z8000 on VAX and IBM systems. Support for the 1750A, including a HLL debugger that can be used with the 8540, is under development. Assembler, linker, and simulator are also available with these products. PSS also offers a complete line of full-function assemblers for most micros and running on many hosts.

Holos

Holos Corp (Tom Ness 404-373-0110) offers C compliers for 8080, 8085, z80, and 6809. Support for others is available upon request. The compilers run on the 8560 and generate Tek assembly code. Price is $1200; specify 8560 media when ordering. Holos Corp. (404-373-0110) also offers a screen editor for the 8560. Nex is a $800 full-featured screen editor supporting user-definable commands (KEYMAP) and over 180 terminals.

C-Systems

C Systems (Jim Eakins 714-637-5362) offers a C compiler for the 8086 with a sophisticated symbolic debug capability. The product is available now on IBM PC.

Destek

Destek (Dennis Reiger 408-737-7211) offers a Real-Time Operating System kernel in C source. Destek will offer a C 68000 compiler in November with support for other micros to follow. See **VENDOR CONTACTS.**

Assisted Technology

ATI (Bob Osann 408-942-8787) offers CUPL, a high-level universal language for programmable logic, for IBM PC and VAX UNIX and VMS systems. It allows the HW designer to write general logic equations and implement them on PALs and IFLs such as those from Monolithic Memories, Signetics, AMD, Harris, TI, and National. Contact the vendor for product info, price, and availability. See **VENDOR CONTACT.**

UNIX Vendors -

The following table lists some new UNIX vendors, some of which are listed in Sept 83 issue of **BYTE**, pages 260-270. Those with products for PDP-11 UNIX v7 systems are listed here. Contact the vendor to determine if the product will run on an 8560; see **COMPATIBILITY HINTS.**

VENDOR	CONTACT	PRODUCT	DESCRIPTION
A.I. Wasserman	415-666-2951	Troll	Relational dbms, screen editor
Andyne Comput'g	613-548-4355	several	menus, profiler, on-line doc,...
Horizon SW Sys	415-543-1199	Horizon WP	Word processing
Internat'l SW	800-323-3629	MDMS	Data Base, English query, reporo
IRIS Systems	619-474-2010	ACUITY	payroll, accounts, inventory,...
PHACT Assoc	212-420-1512	PHACT-dbms	relational database mgr
Softest	201-447-3901	MCIS	Menu Creation/Interpretation
Touchstone SW	213-594-9266	MIMIX	CPM "emulator": run CPM Appl.
Urban SW Corp	212-736-4030	Leverage	mail list/info manager
UniPress SW	201-985-8000	distributor	MIMIX, LEX, menu, UNICALC, ...

VENDOR CONTACTS: NEW AND CHANGES

Assisted Technology Inc. Contact and address for this new vendor is:

Bob Osann, Pres.
Assisted Tech.
Suite 150
2381 Zanker Rd
San Jose CA 95131
408-942-8787

Destek. Contact and address for this new vendor is:

Dennis Reiger, Mktg Mgr
DESTEK
830 E. Evelyn Ave.
Sunnyvale CA 94086
408-737-7211

Holos Corp. Contact and address for this new vendor is:

Tom Ness
Holos Corp.
403 West Ponce de Leon Ave.
Decatur GA 30030
Ph: 404-373-0110

Industrial Programming Institute. IPI's correct phone number is 516-938-6600; it was previously listed incorrectly.

Interactive Systems. In the East contact ISC salesperson Cory Laws at 202-789-1155. In the West contact Ted White, Director of Sales, in Santa Monica (213-450-8363).

Introl. Contact and address for this new vendor is:

John Wisialowski
Introl Corp.
647 W. Virginia St.
Milwaukee WI 53204
414-276-2937

Irvine Computer Science Corp. Contact and address for this new vendor is:

Dan Eilers
18021 Sky Park Circle, Suite L
Irvine CA 92714
714-754-6684

Lantech. Formerly known as Advanced Digital Products, the contact and address for this vendor is:

Joni Bullington, Compiler Marketing
Lantech Systems Inc.
9365 Wendell Rd.
Dallas TX 75243
ph: 214-340-4932

Proprietary Software Systems. Contact and address for this new vendor is:

Joel Fleiss
Proprietary Software Systems
429 Santa Monica Blvd., Suite 430
Santa Monica CA 90401
213-394-5233

Systems and Software. SSI moved from Chicago effective 22 Aug 83:

Paul Chien, Pres.
Systems and Software, Inc
3303 Harbor Blvd.
Costa Mesa CA 92626
Ph: 714-241-8650

Telecon. New address for this vendor is:

Telecon
1155 Meridia
Suite 218
San Jose CA 95125

Telesoft. Address and contact for this vendor is:

Terry Baugh
10639 Roselle St.
San Diego CA 92121

NEW INTERNATIONAL DISTRIBUTORS

Caine, Farber, & Gordon. CFG has set up one international distributor

UK, Western Europe, Australia:

Graham Evans
Warren Point Computers Ltd.
Babbage Road
Stevenage, Hertfordshire
SG1 2EQ
Ph: Stevenage (0438) 316311
Tx: 826255

Hunter & Ready. H&R has set up two international distributors:

Japan: Digital Computer Ltd
No. 25 Kowa Bldg 8-7
Sanbancho, Chiwaoda-Ku
Tokyo 102, Japan
Telex: 232-2386
ph: cc (03) 264-8973

Scand: Digitailor AB
Reimersholmsgatan 8
S-117 40 Stockholm
Sweden
Tx:13887
Ph: (46)8-698803 (intl)
08-698803 (in Sweden)

Real Time Systems. For information and availability of RTS products for the 8560 please contact:

Alan Cleary
Real Time Systems Ltd
PO Box 70
Douglas
Isle of Man, England
tel: 0624-833403, 26021
tlx: 628356 TOMASS G

RTS. For availability of 6809 and 9900 C and Pascal in the US contact:

Ed Rathje
JMI SW Consultants
1422 Easton Rd
Roslyn, PA 19001
Ph: 215-657-5660

Real Time Systems distributes their products via

Austria, Germany, Switzerland:

RETIS Realtime Software AG
Bahnofstrasse 96
CH-5001 Aarau
Switzerland
tel:064-24 77 77
tlx: 981177

Denmark, Finland, Norway, Sweden:
UNISOFT AB
Heurlins Plas 1
413 01 Goeteborg
Sweden
tel: 031-13 56 32

Industrial Programming. Industrial Programming Inc. offers Multi-Tasking Operating Systems. There are several changes in their international distributors; all of them are listed here:

Country	Distributor
France	Celdis SA 53 Rue Charles Frerot 94250 Gentilly ph: (1) 546.13.13 Metrologie La Tour d'Asnieres 4 av. Laurent Cely 92606 Asnieres Cedex ph: 791.44.44
Germany	Scientific Control Systems GmbH Postfach 62 04 80 Oehleckerring 40 2000 Hamburg 62 Ph: 040 531030 Tx: 841-2174113 Celdis-Enatechnik-System Vertriebsgesellschaft GmbH Schillerstrasse 14 2085 Quickborn (Hamburg) Ph: 041 06/612-240 Tx: 841-213590
India	Saras Electronics 21 Perambur Barracks Road Madras 600012 Ph: 32497
Israel	R. N. Electronics Agencies Ltd. 15 Kineret Street Bney-Brak P. O. Box 10205 Tel-Aviv 61 101 Ph: (03) 79 69 27 Tx: 922-3472107
Japan	Tokyo Electron Limited (MTOS-86, 80, 80MP) Shinjuku Nomura Bldg 1-26-2 Nishi-Shinjuku Shinijuku-Ku Tokyo 160 Ph: (03) 344-5893 C. Itoh Data Systems Limited (MTOS-68K, 68KF, 68, 69) C. Itoh Bldg. 2-5-1 Kita-Aoyama Minato-Ku Tokyo 107 Ph: (03) 497-8096
Scandanavia	Frontec Microdatorcentrum AB Box 204

Country	Distributor
	Malmvagen 28 Sollentuna, Sweden Ph: 08-359360 Tx: 854-15130
Switzerland	Xmit AG Bellikonerstrasse 218 CH-8967 Widen Ph: 057-54656 Tx: 845-59955
U.K.	Itech Bessemer Road Stevenage Herts SG1 2DX Ph: (04380) 66561 Tx: 851-825824

COMPATIBILITY HINTS

This discussion pertains to language software.

Downloading to 8450

Whether on 8560, VAX, or other host, the object code must be downloaded to an 8540 for execution and debugging. Downloading involves a suitable format and a means of communication. Format can be TEKHEX (Extended), LAS Object, Intel HEX, or Motorola HEX (S records). The 8540 with the Extended Hex and Comm option can download:

FORMAT	# BITS	OBJECT	SYMBOLS	SPEED
Tekhex	8	yes	no	slow
Ext Tekhex	16	yes	yes	medium
Intel Hex	8	yes	no	slow
Intel Hex	16	yes	no	slow
Motorola Hex	8	yes	no	slow
Motorola Hex	16	yes	no	slow
LAS Obj	8,16	yes	yes	fast

Third party language software produces object in forms of which most can be downloaded to an 8540. Here are the formats, and conversions necessary, and how to download.

FORMAT	CONVERSION?	HOW?
Tek Asm	Assemble/Link to get LAS Obj	ICOM40 *, lo
LAS Obj-relocatable	Link to get LAS Obj	ICOM40, lo
LAS Obj-absolute	none	ICOM40, lo
Tekhex reg	none	ICOM40, rh
Tekhex ext	none	ICOM40, vh (Ext Hex opt)
Intel Hex 8-bit	none for object only	ICOM40, rh
Intel Hex 16-bit	none for object only	ICOM40, vh (Ext Hex opt)
Moto Hex 8-bit	none for object only	ICOM40, rh
Moto Hex 16-bit	none for object only	ICOM40, vh (Ext Hex opt)
a.out -absolute **	a.out -> Ext Tekhex (w symb)	converter fr Tek A.E.
a.out -relocatable	a.out -> LAS obj, Tek link	must program converter
Intel OMF	OMF -> LAS Obj (partial only) or OMF -> Intel Hex	must program; or get converter fr vendor

* ICOM40 is used if the product runs on a host; not needed for 8560

** a.out is the standard UNIX object format

Symbolic Debugging

Assembly-level symbolic debugging on the 8540 is often possible with third party language software. Third party language software generate object in various formats. This table shows whether symbolic debug is supported directly or whether some conversion is needed.

FORMAT	SYMB DEBUG?	HOW
Tek Asm	yes	Assemble & Link with Tek SW
LAS Obj-reloc	usually *	Link with Tek SW
LAS Obj-abslute	usually *	ready to download
Reg Tekhex	no	request vendor support Ext Tekhex
Ext Tekhex	usually *	ready to download
Intel Hex **	no	Extract symb fr Intel OMF to Ext Tekhex
Moto Hex	no	Extract symb fr Moto obj to Ext Tekhex?
a.out	yes	converter to Ext Tekhex (MDP SW Dist Cntr)
Intel OMF	no	Extract symb fr OMF to Ext Tekhex

* The format supports symbols, the product must place them there.

** Intel Hex doesn't include symbols; must get them from the OMF from which the Hex was produced.

Here are some examples of the formats supported by third party products that run on the 8560 (except VAX only is noted with *).

VENDOR	PRODUCT	FORMATS
Alcyon	C 68000	Tek Assembly
Boston Syst Off	Assemblers	Ext Tekhex, Intel Hex, Moto Hex
Caine,F.,Gordon	PLM	LAS Obj-reloc, Intel OMF, Intel Hex
Cymric	Pascal, Asm	Ext Tekhex, Intel Hex, Moto Hex
Enertec	Pascal, Asm	Ext Tekhex, Intel Hex, Moto Hex
Interactive Sys	C z80	Intel Hex
Introl	C 6809	Tek Assembly
Mark Williams	C 8086 *	Intel OMF (limited version)
Microtec	Assemblers	Ext Tekhex, Intel Hex, Moto Hex
PSS	Jovial Debug **	LAS Obj-absolute
Real Time Sys	C & Pascal	Tek Assembler, a.out, Intel Hex, Moto Hex
Santa Cruz Op	C z8000	a.out-relocatable (no linker)
Virtual Syst	FORTRAN, Pascal	Ext Tekhex, Intel Hex
Whitesmith	C compilers	a.out-absolute, Intel Hex, Moto Hex

** Product under development

Running on the 8560

PDP11-UNIX software with the following characteristics will, without any change, run on the 8560:

1. Able to run on UNIX version 7
2. Able to run in 64K bytes without overlays
3. Able to run in common instruction & data (I & D) space
   Note: PDP-11 has either separate I & D (eg 11/70) or common I & D (eg 11/23); Much UNIX Software can be generated to run in either.

Installing on the 8560

To install software on the 8560, note these media specs:

File Formats Vendors must supply 8560 SW in one of the following formats.

tar: same format as UNIX v7; Auxiliary Utilities Package is required to read tar diskettes.

fbr: TNIX only, documented in 8560 System Ref Manual; MDP offers fbr source to vendors who want to ship this format.

dsc50: TNIX only, for 8550 transfers, internally documented only; may be useful if customer has 8550.

others: RT-11, ISIS, CP/M and Motorola formats can be read using unreleased utilities available from your local Tektronix Application Engineer. Use only single-sided, single-density diskettes.

Sectors There is no interleaving and no skewing of the sectors on an 8560 diskette. Some vendors floppy controllers can be programmed for no interleave/skew. Otherwise the diskette read utilities (eg for isis, cpm) read the data as is then reorder the sectors.

Diskettes Formating the disks first on an 8560 increase probability of success. The physical format of the diskette is:

- IBM-Compatible diskettes, soft sectored
- Single or double sided, single or double density
- track 0, side 0 is 128 bytes/sector, FM-encoded, always single density
- track 0, side 1 is 256 bytes/sector, MFM-encoded
- tracks 1-76 both sides are 256 bytes each sector
- TNIX treats 2S-2D diskettes as 1995 512-byte blocks;
  1S-1D diskettes as 500 512-byte blocks

EVALUATING 3RD PARTY SW

References for Third Party Products

References for these products are available from your local Tektronix Sales Engineer.

Product	Target uC	Host	Vendor
C	z80	8560	Interactive Systems
proFORTH	8085/z80	8002	Microsystems Inc.
proFORTH	8085/z80	8550	Microsystems Inc.
C	6809	8560	Introl Corp.
C	68000	8560	Alcyon
C	68000	8560	Whitesmiths Ltd.
FORTRAN	8086	8560	Virtual Systems
Assembly	6805	8560	Virtual Systems

Demonstrating Third Party Products

Several vendors have provided demonstration discs of their 8560-based SW products to some Tektronix Sales Engineers. Others offer their products under an evaluation license. Contact your local Tektronix Sales Engineer or the vendor for possible demo.

Some of these are:

VENDOR	PRODUCT
CFG	8560 PLM 8080 and 8086
Introl	8560 C 6809
Alcyon	8560 C 68000
Cymric	8560 Pascal & Struct Asm
Syscon	8550 PLMX 8085, z80, 6809, 1802

Rodney Bell, MDP Product Marketing

PRODUCT PERFORMANCE SECTION

SOFTWARE ERRATA SHEET

MDP will begin shipping "Errata Sheets" with new versions of software products. The Errata Sheet will be updated when additional problems/workarounds are found. The Errata sheets will be shipped with the product and will be published in User Group News.

Here is the first Errata Sheet:

8560 Pascal Compiler for the 68000/68010 (PAS68K opt 1A) Version 2.

Several problems with the 68000 Version 2 release were noted. These were judged to be problems which can be easily avoided.

The first problem involves use of the **$stackch** or **$list** compiler options in the middle of a program, which may cause the optimizer to report internal errors. We recommend that the **$stackck** option be used once at the beginning of the source file for the time being. No problems are observed when optimization is suppressed.

The second problem involves the **$tagck** compiler option. It may generate unnecessary register stores. If the $stackck option is set (as by default), the register stores may precede the call to the ENTRYQQ routine at the beginning of the main program (or possibly, a procedure). This will cause the negative addresses to be referenced in the case of the main program. The problem occurs infrequently and may be made harmless by inserting $stackck- directive at the beginning of the source file. It is also possible that "with" statements will cause the same register stores. Once again, the **$stackck-** directive will ensure that the stores can only be generated after the activation record is set up by the LINK instruction.

If the number of nested "with" statements exceeds the available registers, erroneous code may be generated. This should not happen unless the "with" statements are nested more than five deep. Our test had 15.

As in the Version 1 release, type checking does not distinguish "value" versus "var" parameters to procedural parameters.

Marilyn Hanson, MDP Product Marketing

TAR COMMAND WITH GPIB REQUIRES BLOCK SIZE

The "TAR" command does not have a default block size when doing a read function. The result of omitting the blocksize on a read is that a read error is generated.

Another feature of the "TAR" command is that all the desired "dash" options are specified first and then the parameters. For example;

tar -xfb /dev/mt0 8 frodo

will extract the file or directory called "frodo" from the tape mounted on mt0 with a blocking factor of 8. Notice however; that both the "-b" and the "-f" were specified befor the parameters associated with them were provided and that they were provided in the order specified.

John Owens, Systems Applications Manager

ACE AND LDE PROBLEMS WHEN USED IN TERM MODE (8540)

Terminals manufactured by several vendors have been found to drop the "RTS" line when sending multiple ASCII characters representing a learned key. The 8540 rereads the last character when this happens. This results in the last character is repeated in the character sequence sent to the 8560 process (LDE ACE and potentially KSH).

The problem can be circumvented by tying the "RTS" line high at the 8540.

Since this does not occur with Tektronix produced terminals, and it is easily corrected, it is not considered to be a problem.

John Owens, Systems Applications Manager

4105 VERSION 3 FIRMWARE AND KEYSHELL

The new 4105 Version 3 firmware has a minor incompatibility with the current TNIX V2 Colorkey+ software. The fix is easy, simply edit the *.pix files located in /usr/lib/ksh/bin and remove the "enable gin mode" command. An easy way to do this is to invoke lde on the *.pix files and delete the "escape control-z" represented as "^[^Z" near the beginning of the file. Place the cursor on the start of the above sequence and press rubout twice, write the file out, and then exit.

Greg Saville, Software Applications Manager

MDP BUG BASE

The following product performance reports are contained in our data base. If you have encountered additional problems not listed here or in previous issues, please use the product performance report form provided at the end of this section. We will keep you informed about the progress toward the solution to the problem. We will also try to provide a "work-around" immediately.

John Owens, Systems Applications Engineer

8540 ROMPATCH FAILURE

PRODUCT

8540 executing rompatch -l

CONFIGURATION

8540 OS-40 version 1 with patch #36 and roms for TTA, COMM (V.-01), prom programmer, 8086/8088 and 68000 (without diags)

PROBLEM

With the comm rom (version -01) and all the rom sockets filled, then executing the "rompatch -l" command will cause the 8540 to halt in an endless loop.

NATIVE PROGRAMMIMG TOOLS INSTALLATION

PRODUCT

8560 and installing Class C software

CONFIGURATION

8560 TNIX version 1.4 and Class C software, Native Programming Package version 1.0

PROBLEM

If the Native Programming Package is told "no" to installation of "/usr/include" and "/bin/sed" using the "install" command, all commands or files which use "/usr/include" as a dependent file will not be installed.

DIRECTORY PERMISSIONS OF /USR/INCLUDE/SYS

PRODUCT

8560 and accessing /usr/include/sys

CONFIGURATION

8560 TNIX version 1.4

PROBLEM

Only root can execute the directory or contents of /usr/include/sys.

COMMENTS

Workaround--change attributes of /usr/include/sys from 644 (drw-r--r--) to 755 (drwxr-xr-x).

68000 MOVEM INST ERROR IN ASSEMBLER

PRODUCT

68000 assembler and the movem with long displacement

CONFIGURATION

8560 TNIX version 1.4, 68000 assembler version 2.01-08

PROBLEM

The following movem instruction (an undocumented extension of the instruction) does not produce correct object code

movem.l - value (an), #imm
movem.l - value (An,Xi),#imm
movem.1 - value (pi),#imm
movem.1 - value (pc,xi)#imm

The object code does not include the displacement value.

COMMENTS

Workaround--use register list form of instruction

NON-CODE GENERATING SECTIONS

PRODUCT

Assembler with non code generating sections

CONFIGURATION

8560 TNIX 1.3 and 6801 assembler version 01.03-18

PROBLEM

After linking, the assembler inserts the header data before the actual executable code. The load address will be incorrect if any section contains only non- code generating statements (such as block statement).

COMMENTS

Avoid the use of sections which do not generate any code; i.e. those which contain only block directives.

FIND EXAMPLE DANGEROUS

PRODUCT

8560 and the "find" command

CONFIGURATION

8560 TNIX version 1.4

PROBLEM

Section 8 of the TNIX manual (page 17) describes a procedure to remove all files owned by a user. **The example should be followed with extreme caution.** If the user omits the "-" before the "user" option, all files will be removed. Also if the user is logged-on, the command execution will result in the port assigned to the "removed user" also being removed.

6801 STATUS LINE PULSES IN MODE 1

PRODUCT

6801 Emulator and Probe in emulation mode 1

CONFIGURATION

8550 DOS-50 version 1.3 with the 6801 Emulator and Probe

PROBLEM

When the 6801 is in Emulation Mode 1 and the Processor is in Mode 7, (single chip mode) erroneous pulses are present on the output of Port 4. The pulses are only present when the DDR (Data Direction Register) is configured as outputs and the TRA A command is in effect.

8085 PASCAL AND LPT

PRODUCT

8085 Pascal compiler and LPT

CONFIGURATION

8550 DOS-50 version 2.1A, 8085 Pascal compiler version 4.02

PROBLEM

Pascal module FILUQQ installs a series of strings in memory representing various devices such as 'LPT'. In the case of LPT, a space character proceeds the L character. Thus the assign svc sees "space 'L' 'P'" which is not recognized, nor does the generated code complain at the assign failure. THUS LPT DOES NOT WORK AS A PASCAL OUTPUT CHANNEL.

COMMENTS

Workaround--after loading code in memory & the sequence " LPT" is found, the examine command can be used to change it to "LPT ".

6801 MODE STATUS IN EMULATION

PRODUCT

6801 emulator in mode 3

CONFIGURATION

8550 DOS-50 2.1A or 8540 OS-40 version 1.0 with 68XX emulator and the 6801 probe

PROBLEM

When using the 6801 probe, mode 3 (trace) will be reported as the designated mode even though another mode was selected.

COMMENTS

The mode being emulated is correct.

MAKE DEPENDENCY LIST LIMITATIONS

PRODUCT

8560 make command and more than 41 files

CONFIGURATION

8560 TNIX version 1.3

PROBLEM

If the "make" command has a dependency list of more than 41 files, then execution of "make" will cause 1) a memory fault, 2) a core dump and 3) a halt to the "make" command.

8085 PASCAL FILE WRITE

PRODUCT

8085 Pascal errors when reading, then writing a file

CONFIGURATION

8550 DOS-50 V2.1A and 8085 Pascal V4.02

PROBLEM

Errors are generated when a compiled program is executing in mode 0 and tries to open more than one file and write the lines that were read to a second file. The error message generated is "operation error <filename> error code 1003 status 5E". This message indicates that the diskette was full (even though 1600 blocks are available) and a directory was expected (even though a file name is required).

Z80 LONG DISPLAY TRACE OF REGISTERS

PRODUCT

Z80B emulator and execution of a long trace

CONFIGURATION

8540 OS-40 V1.0, 8560 TNIX V1.4 and Z80B emulator

PROBLEM

When the 8540 and the 8560 are connected via HSI, the Z80 emulator selected, and a long trace is being executed, Registers IX, IY, and SP change. IX and IY pick up the value in HL register, while SP picks up an unknown value. The 8540 must be in config term. The problem seems to be independent of whether program is in program or prototype memory.

Z80 EMULATION BREAKS AT NON-SELECTED ADDRESSES

PRODUCT

Z80 Emulator and random breakpoints

CONFIGURATION

8540 OS-40 V1.0 or 8550 V2.1A and Z80A or B emulator

PROBLEM

The Z80 emulator will break on random breakpoints that have not been designated by the user. This problem is very visible on Z80B emulators and particularly those emulators with the LAS memory.

COMMENTS

A mod is in evaluation.

AT COMMAND USAGE

PRODUCT

TNIX and the 'at' command

CONFIGURATION

8560 with TNIX version 2.0

PROBLEM

When the 'at' command builds its command file, it does not quote or export shell variables.

COMMENTS

This may result in the mysterious creation of 0-byte files.

MISSING FUNCTION IN LDE BACKPLANE

PRODUCT

LDE in graphics backplane

CONFIGURATION

8560 TNIX V2.0 and LDE V2.0

PROBLEM

With LDE, the graphic backplane "help" does not show that UNDO is accomplished with "f5".

COMMENTS

See "LDE HELP SCREEN FIX" in Applications section for assistance.

SEARCH PARAMETER -A ERROR

PRODUCT

8550 DOS-50 and search command

CONFIGURATION

8550 DOS-50 version 2.1A

PROBLEM

"SEARCH" (sea) command does not detect "-a" (lower case) as a legal modifier; only "-A" (upper case).

LAS LINKER AND RELINK OPTION

PRODUCT

LAS linker and relink option

CONFIGURATION

8560 TNIX V2.0, Z80 assembler V1.01-06 and linker V2.08-00

PROBLEM

Trying to link overlays using -R relink option generates error message "previously defined -link 100". The old version linker didn't.

COMMENTS

Workaround--use -D at link time to define values rather then EQU directive.

6801 ASM BANNER

PRODUCT

6800/01/02 Assembler and listing banner

CONFIGURATION

8560 TNIX V1.4 or 2.0 and 6800/01/02 assembler V1.03-18

PROBLEM

Assembler listing banner always says 6801, no matter what you set your shell variable to (6800, 6801, or 6802). This is just a cosmetic problem. Perhaps the listing could be changed to say 680X or 6800/01/02.

68000 PASCAL PUBLIC PROCEDURES

PRODUCT

68000(8560)Pascal compiler & public procedure w/variant field

CONFIGURATION

8560 TNIX V2.1 and 68000 Pascal compiler V01.01-14

PROBLEM

In the 68000 pascal compiler, declaring a procedure public and passing record with a variant field causes an abnormal termination.

6800 ASSEMBLER & BLOCK STATEMENTS

PRODUCT

6800 assembler and listing block statements

CONFIGURATION

8560 TNIX V2.x and 6800 assembler V1.03-18

PROBLEM

Macro list "MEG" option does not list BLOCK statements in asy list as described in the B Series Core Manual page 3-32. The earlier version worked correctly.

COMMENTS

Workaround--if the MACRO doesn't contain any conditional assembly statements, use a LIST ME.

MISSING MAN INFORMATION

PRODUCT

8560 TNIX and man command

CONFIGURATION

8560 TNIX version 2.0

PROBLEM

The man command will not read tta man pages (i.e. man acq will return message man can't find information for "acq").

COMMENTS Use "man acq.tta" or modify the filenames in /usr/man/cat6. i.e. acg.tta is changed to acq.6

8086/88 PASCAL POINTERS

PRODUCT

Pascal 8086/8088 compiler V01.10-05 (8560) RTS

CONFIGURATION

8560 TNIX version 2.0 and 8086/8088 Pascal compiler version 1.10-05

PROBLEM

The RTS routine dispose flags a valid pointer as bad with a runtime message "error #3."

COMMENTS Version 2 of the 8086/8088 Pascal Compiler is planned for second quarter of 1984.

68000 ASSEMBLER SHORT BRANCH

PRODUCT

68000 assembler error message 254

CONFIGURATION

8560 TNIX V1.4 and 68000 assembler V01.15-66 or V2.01-08

PROBLEM

A short branch instruction to the next instruction (displacement of zero) causes ASM error 254 (invalid short branch instruction). The code generated is ok. The manual states this error will result from this uncommon use, however the error might be better called a warning since some compilers may generate this instruction.

68000 and Z8000 PASCAL TYPE EXPRESSION

PRODUCT

Z8000 and 68000 Pascal compiler/PDB

CONFIGURATION

8560 TNIX V2, Z8000 Pascal compiler V01.09-08/PDB V1.05 and 68000 Pascal compiler V01.01-14/PDB V1.05

PROBLEM

When complex types of pointers and records are declared in the compiler source, the order those types are declared can cause PDB to abort with core dump. Any pointer types may have to be reordered.

COMMENTS

68000 Pascal Version 2 corrects this in second quarter of 1984.

PASCAL LDE AND PASTE AND CUT FUNCTION

PRODUCT

Pascal Language Directed Editor & deleting default paste files

CONFIGURATION

VAX VMS V3.1 and Pascal Language Directed Editor V1.03-00 with a vt100 terminal

PROBLEM

Even though a cut going to the default paste file is cancelled LDE will still delete all of the contents of the paste file.

PASCAL LDE AND LINEFEEDS

PRODUCT

Pascal Language Directed Editor and linefeed

CONFIGURATION

VAX VMS V3.1 and Pascal Language Directed Editor V1.03-00 with a vt100 terminal

PROBLEM

When a Line Feed is entered into the text, LDE will convert it to a carriage return with a line feed when it writes it out to a file.

8086 PASCAL & PORT ASSIGNMENTS

CONFIGURATION

8560 TNIX version 2.0 and 8086 Pascal compiler version 1.10-05

PROBLEM

With the 8086 Pascal compiler, incorrect object code is generated when 1) a variable is declared as a port and 2) reading AND writing the port in the same assignment statement.

COMMENTS

Workaround--use two statements; one to read; one to write.

8086 PASCAL & GLOBAL NAMES

PRODUCT

8086 Pascal compiler and global names

CONFIGURATION

8560 TNIX version 2.0 and 8086 Pascal compiler version 1.10-05

PROBLEM

When using declarations of the form array [lowband..highband, low2band..high2band, etc.] the compiler messes up the global names of public and external variables and public and extenrnal subroutines.

COMMENTS

Workaround--change declaration to be array [ ] of array [ ] of array [ ]...

DISP COMMAND AND THE 8540

PRODUCT

8540 in config term with the 8560

CONFIGURATION

8560 TNIX version 1.3, 8540 OS-40 version 1 and patch 36, TTA and Z80 emulator

PROBLEM

When in config term mode, if a disp command is invoked, then a "control C" will not gracefully stop the display. A large number of "control C's" will eventually halt the display, but then the system is halted with the error message "Error 12 I/O error or access violation on write."

LAS ASSEMBLER LO DIRECTIVE

PRODUCT

Any LAS assembler with lo($) & word directives in a macro exp.

CONFIGURATION

8560 TNIX version 1.3, 6801 assembler version 1.06-08 and linker version 2.08

PROBLEM

With a lo($) and word directives in a macro expansion, the object file is created with invalid code and without warning or error messages.

COMMENTS

This problem is not unique to the 6800 and 6801 assembler and can occur with any LAS assembler.

EXAM REPEAT OF THE CURRENT LOCATION

PRODUCT

8540 and 8550 EXAM command

CONFIGURATION

8540 OS-40 version 1 or 8550 DOS-50 version 2.1A

PROBLEM

While either the 8540 or 8550 is in term mode, a linefeed during the "exam" command will not repeat the current line, but instead only generate a line feed. In addition, while the 8540 or 8560 are in term mode with the 8560, error messages generated during an "exam" command will not have a carriage return. Also, when a "control-C" is used during "exam", a carriage return does not precede the system prompt.

PDB ERROR #22

PRODUCT

8086,Z8000 & 68000 Pascal & FORWARD procedure with error #22

CONFIGURATION

8560 TNIX version 1.3, 8086/88 Pascal Debug version 1.07-00A, 8086 emulator version 1.10 and 8540 OS-40 version 1 (000021)

PROBLEM

When invoking symbolic debugging (on the 8086 Pascal Debug) with the command "pdb trans.ld," the user may receive the error message "panic at proc #22 Subr range error." The system operation can not be processed without reloading. The problem is caused by the FORWARD procedures not placed in symbol table for Pascal Debug.

COMMENTS

Workaround--Avoid FORWARD procedures by using separate modules instead until the new version of 8086 Pascal compiler.

DISP OUTPUT DURING 68000 PDB

PRODUCT

68000 Pascal Debugger and ds command

CONFIGURATION

8560 TNIX Version 1.4, 8540 OS-40 version 1 and 68000 Pascal debug version 1.05-00

PROBLEM

Sometimes the output of 8540 ds command will be sent to the screen after a PDB command. The output has no impact or significance to PDB.

CRASH WHEN IU = NONEXISTENT PORT

PRODUCT

8560 or 8561 and ports 4,5,6 and 7

CONFIGURATION

4-port 8560 or standard 8561 and TNIX version 1.4

PROBLEM

If the IU shell variable is set to a value corresponding to a port which has no IOP board (4-7), the system will crash (bus error) when an 8540/50 command is executed. Removing all read, write, and execute permission (from hsi4 thru hsi7) in the /dev directory is one solution to the problem, but this results in an error in command execution rather than a system crash.

LIBRARY CALLS WITHIN LIBRARY CALLS

PRODUCT

Las linker calling 2nd library module w/many symbols in 1stmod

CONFIGURATION

All 8550s and 8560s with linker versions through 2.08 and VAX Unix linker version 2.01-00

PROBLEM

If one library in a module with many symbols calls a 2nd module in the same library, the 2nd module may not be linked.

COMMENTS

Workaround--extract the module that calls the second module and link before the library. The extracted library members may be placed in a separate library.

STRING I/O LIMITATION IN 68000 PASCAL

PRODUCT

68000 (8560) Pascal compiler and transfer of 256 bytes

CONFIGURATION

8560 TNIX V2.0 and 68000 Pascal compiler V01.01-14

PROBLEM

68000 pascal compiler routine will fail with a fatal I/O error #238 at runtime when asked to transfer over 256 bytes of data at one time.

NATIVE PROG. TOOLS AND TRAP-1

PRODUCT

8560, 8561 TNIX operating system and trap-1

CONFIGURATION

8560 TNIX V2.0 Native Language Package

PROBLEM

A trap-1 brings down the operating system when a less radical action would be better; ie. a program that accidentally writes off the end of the dimensioned array. A trap-1 is appropriate when invalid instructions are encountered, but should not crash a MULTI-user system (it can just kill the process).

68000 MOVEC INSTRUCTION ERROR

PRODUCT

68000 assembler and bad code for MOVEC instruction

CONFIGURATION

8560 TNIX version 2.0 and 68000 assembler version 2.01-08

PROBLEM

The 68000 assembler produces bad code for the following instruction MOVEC with VBR or USP as the operand. For example, MOVEC AQ, VBR generates 4E7B 8B01 and should produce 4E7B 8801. The Motorola spec sheet is wrong. This caused the assembler to produce bad code.

ACE EDIT OF READ ONLY FILES

PRODUCT

ACE editing read only files

CONFIGURATION

8560 TNIX V2.0, ACE V3.00

PROBLEM

Version 3.0 of ACE will not let a user read a file (with read only capability) into the current edit session. A brief error message (system function error #13) is generated. The workaround is to chmod to rw capability before entering ACE. The previous version of ACE did allow this.

COMMENTS

Workaround--chmod to rw capability all files before editing.

USER GROUP LIBRARY ABSTRACTS

3D & 3D1 - 4105 GRAPHICS DEMO DISPLAY

Graphics — MUGL TNIX Vol II
856X w/4105 Terminal — C

Abstract

These programs display interesting graphics patterns on a 4105 Color Terminal.

4105DEFINES.H - C DEFINE LIBRARY

C define library — MUGL TNIX Vol II
856X/4105 — C

Abstract

A .h "define" library for use in C programs which make use of the Tek 4105 Color Terminal.

BOX - DISPLAYS GRAPHICS CHECKERBOARD ON 4105

Graphics Demo — MUGL TNIX Vol II
856X with 4105 Color Terminal — C

Abstract

This program displays an interesting graphics checkerboard pattern on a 4105 Color Terminal. C source included.

COM1 - NEC TO TEK ASSEMBLER CONVERSION SCRIPT

Conversion Utility
856X

MUGL TNIX Vol II
Ed Script

Abstract

This ed script will modify an NEC assembler source file to be compatible with the Tektronix assembler.

Author: Mr. Tanaike, SONY/TEK

DEBUG.HELP - DISPLAY DEBUG HELP SCREEN

4105 Help Screen
856X/4105 Color Terminal

MUGL TNIX Vol II
4105

Abstract

This command copies a help screen to the 4105 graphics plane for emulation and debug commands.

Author: Muggsie Nixon, Boston Field Office

DNLD - TEKHEX DOWNLOADER PROGRAM

TEKHEX Downloader
Host w/Fortran

MUGL TNIX Vol II
Fortran

Abstract

This program implements a TEKHEX downloader from a host to an 8540 or 8550. The source is in Fortran and must be ported to the host machine and compiled locally.

Author: Jim E. Dunn, Simmonds Precision Products

ENCODE - 4105 PROGRAMMING UTILITY

4105 Utility
856X/4105

MUGL TNIX Vol II
C

Abstract

This utility encodes an integer to a value which can be used via a 4100 series terminal host command.

Author: Jim Willey, Santa Clara TSS

FRACTION - CONVERT FLOATING POINT TO FRACTION

Utility
856X

MUGL TNIX Vol II
C

Abstract

Converts a floating-point number to a series of fractions. Each fraction is more precise than the previous one.

Author: Bill Pfeifer, MDP Design Engineering

GCAT - CAT TO 4105 GRAPHICS SCREEN

4105 Utility
856X/4105 Terminal

MUGL TNIX Vol II
C

Abstract

This handy utility copies "standard out" to the 4105 graphics area. Useful for listing things like a directory that you will need to refer to several times. Simply "ls | gcat" and press the dialog key whenever you wish to see the directory listing.

Author: Bill Pfeifer, MPD Design Engineering

HILBERT - 4105 COLOR TERMINAL GRAPHICS

Graphics Demo
856X/4105 Terminal

MUGL TNIX Vol II
C

Abstract

This program displays the familiar "hilbert" pattern, a graphics oriented example of recursion. C source included.

Adapted by: Bill Pfeifer, MDP Design Engineering

HP - HEWLETT PACKARD CALCULATOR SIMULATOR

Utility/Simulator
856X

MUGL TNIX Vol II
C

Abstract

This program simulates a very complete HP Calculator. In addition to the reverse polish style of operations and the standard mathematical functions, other features include: numeric entry similar to the HP-1X series; full range of trigonometric functions, including hyperbolic functions; decimal and analog time conversions; statistical functions including combinations, permutations, correlation, linear regression and estimation; exponential engineering, and fix notations; integer arithmetic in decimal, octal, and hex modes including logical operations AND, OR, XOR, and NOT; 62 continuous memory registers including memory register arithmetic; register exchange functions; polar and rectangular conversions; and more. Features "cbreak" operation, so the return key acts as a true enter key and is not required for line input. Command set includes: enter, clearx, factorial, sigmaplus, sigmaminus, stats, percent, multiply, add, subtract, divide, convtime, convfrom, convto, squarex, clear, stackdisplay, exponential, fix, inverse, lastx, rotatemode, snlog, off, pushpi, recallmem, storex, squareroot, exchange, mod, power, absolute, cosine, rotatedown, fraction, pgamma, hypotenuse, integer, clog, mantissa, nlog, polar, rectangular, sine, tangent, rotateup,

xychange, and, or, xor, complement, changesign. Even simulates "continuous memory" by saving entire calculator state when terminating and restoring status upon reinvocation. Executable binary image and manual page only, source code is not available.

Author: Ed Morin, MDP Design Engineering

IBM - FORMAT DISK READER

Disk Utility
856X

MUGL TNIX Vol II
C

Abstract

This program transfers IBM format disk files to the 8560. Two versions are supplied, ibm and ibm1, which read 80 byte card image, and 128 byte logical images respectively. After transfer, the IBM end of line characters must be changed to the TNIX newline character. A utility called "reform" is supplied to make this translation. C source provided for the ibm and ibm1 programs.

Author: Mr. Tanaike, SONY/TEK

IGE - INTERACTIVE GRAPHICS EDITOR

4105 Graphics Editor
856X/VAX/PDP11

MUGL TNIX Vol II
C

Abstract

This program implements an Interactive Graphics Editor for Tektronix 41XX graphics terminals. It can automatically generate geometrical figures, panels, text, etc. for creating "slides". Ige interactively constructs pictures in the graphics area of the terminal. A picture is composed by entering commands from the keyboard. An audit of all commands is automatically written to a file for subsequent revision. Ige can be used to format previously constructed pictures without entering the interactive mode if desired. Commands that require an xy coordinate as a parameter take the position of the cross hairs at the time the command is typed. Many of the 4105 graphics demo pictures included on this MUGL volume were created with ige. The program features a built-in help menu for easy use and includes a manual page. Complete C source and makefile included. Requires UNIX or TNIX host and "spline(1)" command for "~ " command. Will not execute through an 8540 in term mode.

Author: Rainer Wieland, Senior SW Engineer, MDP Design Engineering

---

INTELSYM - CONVERT INTEL.OBJ TO TEKHEX

Conversion Utility
856X

MUGL TNIX Vol II
C

Abstract

The intelsym program reads an Intel object file, extracts the debug symbol information, and converts it to Extended Tekhex. This allows users with Intel object files to port (intel80 disk reader or VIP com package) their files to an 8560/40 and use the symbol information for 8540 symbolic debugging. C source included.

Author: Muggsie Nixon, Boston Field Office

LINES - 4105 GRAPHICS DEMO

Graphics Demo
856X/4105 Color Terminal

MUGL TNIX Vol II
C

Abstract

Another interesting graphics demo program for the 4105 color terminal. Displays a varying color line pattern which bounces on the edges of the screen and builds upon itself. No source available.

LIST - LISTING HEADER FORMATTER

Listing Utility
856X

MUGL TNIX Vol II
C

Abstract

This utility adds various header information to listings. Some of the data included in the header will need to be modified. C source included.

LP1R - MODIFIED PRINTER SPOOLER

Printer Spooler
856X

MUGL TNIX Vol II
C

Abstract

This modified lpr spooler allows the use of "smart" printers by adding new flag options. The -s (smart) flag passes files unmodified so special printer control sequences can be used. The -d (dumb) option allows dumb printers to simulate smart actions like bold face and underlining (used to be the default). The new default (no flag specified) now strips all escapes and backspaces for standard ascii file printing with no frills. Executable binary object only, no source available.

MAINT - MAIL LIST MAINTENANCE PROGRAM

Database Maintenance
856X

MUGL TNIX Vol II
C and Shell Scripts

Abstract

This set of programs can be used to maintain a mail list database. It features encryption protection of the database, and menu driven operation. It has been used in MDP Marketing to maintain a mailing list of over 4000 names. Options are provided for creating/updating the database, selectively retrieving classes of records, and selectively printing mail labels. The programs are a combination of C programs and shell scripts.

Author: John Owens, MDP Marketing

MVUL - RENAME UPPER TO LOWER CASE FILE NAMES

Utility
856X

MUGL TNIX Vol II
Shell Script

Abstract

This script renames multiple UPPERCASE file names to their lowercase equivalent. Useful when

dsc50'ing DOS-50 files to TNIX to avoid the inconvenience of typing uppercase filenames.

Author: Greg Saville, MDP Marketing

PICTURES.DIR - DIRECTORY OF 4105 GRAPHICS PICTURES

Graphics Pictures
8560/61 and 4105 terminal

MUGL TNIX Vol II
4105 terminal

Abstract

This directory contains several pictures utilizing the graphics capabilities of the 4105 color terminal. Some are simple line-type pictures, while others are striking color examples. These files are in a format which can be "cat'ed" to the 4105's screen. Selections include: Australian flag, Christmas candle, Darth Vader, misc flags, Mt. St. Helens, kitty, lizard, Mickey Mouse, R2D2, United States flag, and others. A shell script is included which automatically displays each picture.

REFORM - CORRECT NL - CR/LF SEQUENCES

Utility
856X

MUGL TNIX Vol II
C (binary only)

Abstract

This program is used to convert the nl-cr/lf end-of-line sequence for files transferred with the ibm, ibm1, and other programs. Since this version is written in C, it executes much faster than equivalent utilities using ed scripts.

Author: Mr. Tanaike, SONY/TEK

RMD - MODIFIED REMOVE COMMAND

Utility
856X

MUGL TNIX Vol II
Shell Script

Abstract

This modified remove command protects the user from accidental file deletion by placing deleted files in a backup directory rather than actually deleting them. It saves most, but not all data, and tries to resolve specified flag options.

UNIX <-> RT-11 FILE I/O PACKAGE

File Xfer Utility
856X

MUGL TNIX Vol II
C

Abstract

This utility program allows you to read and write RT-11 disks on an 856X. Features a built-in help command and has options to list the RT-11 directory, erase an RT-11 file, upload an RT-11 file, initialize an RT-11 disk, download a TNIX file to the RT-11 disk, rename an RT-11 disk file, and type an RT-11 file. C source code and manual page included. Originally written for CP/M - RT-11 use, this version has been adapted for UNIX - RT-11. C source included.

Original author: William C. Colley, III

Adapted for UNIX by: John A. Limpert

SETCOLOR - MODIFY 4105 CHARACTER/BACKGROUND COLORS

4105 Utility — MUGL TNIX Vol II
856X/4105 Color Terminal — C

Abstract

This utility lets you easily modify the 4105's character and screen background colors. Setcolor offers two modes of operation -- interactive mode prompts you for selections via a menu, while command mode lets you specify the desired colors on the invocation command line. See "setcolor.doc" for full information. C source and definition libraries included.

Author: Patrick Glidden, MDP Engineering

SIERPINSKI - 4105 COLOR TERMINAL GRAPHICS DEMO

856X/4105 Color Terminal — MUGL TNIX Vol II
4105 Graphics Demo — C

Abstract

Another interesting graphics demo for the 4105 Color Terminal. Allows specification of the order of the first sierpinski curve and the number of overlapping sierpinski curves desired.

Adapted by: Bill Pfeifer, MDP Design Engineering

TELEX - SPECIAL CHARACTER FILTER

Utility — MUGL TNIX Vol II
856X — Shell Script

Abstract

This filter expands non-standard special characters to descriptive strings enclosed in parenthesis which are compatible with international telex systems. Useful for ensuring that no incompatible special characters are used in a document being sent by telex. Uses standard in and out. Typical usage: cat infile | telex >outfile or: cat infile | telex | nroff -ms >lp1r. This shell script uses sed to expand the special characters and may easily be modified for local use.

Author: Greg Saville, MDP Marketing

THEX - WHITESMITH'S OBJECT TO TEKHEX CONVERTER

Converter Utility — MUGL TNIX Vol II
856X — C

Abstract

This program will convert Whitesmith's object code format files to a form which allows transfer of both object code and symbol table information to an 8540. Four options are available. The default output is extended TEKHEX which is sent to standard out. The -c option outputs in 8540 command form and is used with term mode to an 8540 where rh will not load extended tekhex formated symbol information. The -d option outputs debug information about the input file to stderr. The

-s option will output table information only, in the specified form. C source and manual page included.

Author: Chris Maynard, Tek U.K. Ltd Harpenden

TREE - PRINT TREE STRUCTURE OF A DIRECTORY

Utility — MUGL TNIX Vol II
856X — C

Abstract

This neat utility prints out a tree picture of any given directory. Has many options including things like sorting with directories at top, selecting user group instead of name, including stats with file entries, setting length of printout, built-in help, etc. Submitted to the uucp news network under net.src by Dave Borman, St. Olaf College.

TTA.HELP - DISPLAY TTA HELP SCREEN

4105 Help Screen — MUGL TNIX Vol II
856X/4105 Color Terminal — 4105

Abstract

This command copies a help screen to the 4105 graphics plane for TTA commands.

Author: Muggsie Nixon, Boston Field Office

UMODEM - UNIX <-> CP/M MODEM COMMUNICATIONS

Modem Communications — MUGL TNIX Vol II
856X — C

Abstract

This program implements the HOST side of the popular CP/M MODEM7 communications package. Users with CP/M systems and the public domain MODEM7 program can use it to login to an 8560 system and transfer ASCII or BINARY files in EITHER direction. C source and manual pages included.

Authors:
Walter Reiher
Lauren Weinstein
Richard Conn
Bennett Marks
and a host of others listed in the source

XTAB - EXPAND TABS FILTER

Utility Filter
856X

MUGL TNIX Vol II
C

Abstract

This filter expands tab characters to spaces. Useful for sending output to devices which can't handle tabs. This C program is currently setup with tabs at 10 column positions, but a define statement in the source can easily be modified to any desired tabstop.

Author: Greg Saville, MDP Marketing

LPD: 5660-DS8/0

Date: 31 October 1983

LICENSED PROGRAM DESCRIPTION

PROGRAM NAME: **80/DS DEVELOPMENT SYSTEM FOR USE ON THE TEKTRONIX 8560/61 UNDER TNIX (5660-DS8)**

VERSION: **1.2**

DESCRIPTION

The 80/DS Development System is an integrated tool set for developing Intel 8080/8085 and Zilog Z80 software in both PL/M and assembly languages. It is a Licensed Program of Caine, Farber & Gordon, Inc. (CFG) and operates on the Tektronix 8560 and 8561 computers under the TNIX operating system.

The components of the 80/DS Development System are:

- the 80/PC Compiler which supports the 80/PL language (an upward compatible extension to the PL/M language);
- the 80/AS Assembler which is generally compatible with the Intel 8080/8085 Assembly Language; and
- the 80/RL Relocation and Linking Tools which combine relocatable object modules into absolute object modules for down-loading to the target machine or to an emulator (e.g., an emulator in Tektronix' 8540 Integration Unit).

THE 80/PC COMPILER

The 80/PC compiler compiles source modules written in the 80/PL language (an extension of the PL/M-86 language) into object modules for execution on the Intel 8080 and 8085 and the Zilog Z80 microprocessors. Object modules are produced in either the Tektronix LAS Object Module Format or the Intel MCS-80/85 Relocatable Object Module Format.

General Features and Capabilities

The 80/PL language is a superset of the PL/M-86 language and most PL/M-80 and PL/M-86 source modules should compile and execute without modification.

Among PL/M-86 features which are not available in PL/M-80, 80/PL includes support for the WORD, INTEGER, and POINTER data types and the full set of PL/M-86 string handling functions. The only PL/M-86 features not supported in 80/PL are the REAL data type and those, such as the LOCKSET function, which depend upon being executed on an 8086 microprocessor.

The 80/PC compiler supports the SET, RESET, and conditional compilation controls of the PL/M-86 compiler. The INCLUDE compiler control is also supported, except that the path name of a file to be included must correspond to the syntax of a TNIX path name. The other PL/M-86 compiler controls are not supported.

The 80/PL language contains a number of extensions to PL/M-86, including:

- Relaxation of restrictions on the ordering and factoring of items in DECLARE statements;
- Introduction of structures within structures;
- Introduction of explicitly based references;
- Use of the HIGH and LOW builtins as assignment targets;
- Introduction of a fully-delimited IF block construct;
- Introduction of an UNDO statement for premature loop exits; and
- Introduction of a new scope for external data and procedures so that external items declared in an included file may be redeclared within a module.

Compiler Output

Output of the 80/PC compiler consists of a relocatable object module in either the Tektronix LAS Object Module Format or the Intel MCS-80/85 Relocatable Object Module Format and, possibly, one or more error messages. The compiler also produces source listings and symbol table listings.

Run-Time Support Library

Object modules produced by the compiler may call out-of-line routines to perform word and string operations. These routines reside in a library which is distributed with the 80/PC compiler.

Available Documentation

A detailed description of the 80/PL language and the 80/PC compiler is contained in "80/PL Language and 80/PC Compiler Reference Guide", CFG Order No. 9201-1.

THE 80/AS ASSEMBLER

The 80/AS assembler assembles source modules written in the 80/AS language (generally compatible with that of the Intel 8080/8085 Assembly Language) into object modules for execution on the Intel 8080 and 8085 and the Zilog Z80 microprocessors. Object modules are produced in the Intel MCS-80/85 Relocatable Object Module Format.

General Features and Capabilities

The 80/AS language is sufficiently similar to Intel's 8080/8085 Assembly Language that most programs written for that assembler should assemble correctly under 80/AS with little or no modification to the source.

The most significant area of change is in the handling of assembler controls. Instead of using special control lines beginning with a '$', 80/AS uses assembler directives that appear in the operation field of statements.

The 80/AS extensions include:

- Optional use of names up to 31 characters in length;
- Support of both blank and named COMMON segments;
- Support of symbols with limited scope (local symbols); and
- Recognition of a limited set of Z80 instructions.

Assembler Output

The output of an 80/AS assembly consists of:

- Possible error messages;
- A relocatable object module in the Intel MCS-80/85 Relocatable Object Module Format;
- An optional assembly listing; and
- An optional symbol table and cross-reference listing.

Available Documentation

A detailed description of the 80/AS language and assembler is contained in "80/AS Language and Assembler Reference Guide", CFG Order No. 9201-2.

THE 80/RL RELOCATION AND LINKING TOOLS

The 80/RL relocation and linking tools provide an integrated set of utilities for combining and manipulating relocatable object modules (using the Intel MCS-80/85 Relocatable Object Module Format) to produce absolute object modules suitable for loading and running on Intel 8080/8085 and Zilog Z80 microprocessors.

General Features and Capabilities

The 80/RL tools are:

80/LINK	Combines multiple object modules and libraries into a single relocatable object module.
80/LOC	Converts a single relocatable object module into an absolute object module.
80/MAP	Produces an address map of one or more object modules.
80/STRIP	Deletes public and debugging dictionary information from one or more object modules.
80/THEX	Converts an object module from relocatable Intel MCS-80/85 Relocatable Object Module Format to Tektronix absolute hexadecimal form with symbols and assigns absolute addresses.
80/HEX	Converts an object module from Intel MCS-80/85 Relocatable Object Module Format to Intel absolute hexadecimal form.
80/DSOBJ	Converts an object module from Intel MCS-80/85 Relocatable Object Module Format to a convenient, human-readable form.
80/CROBJ	Converts the display form of an object module, as produced by 80/DSOBJ, to Intel MCS-80/85 Relocatable Object Module Format.
80/LIBCR	Creates a library of object modules in a form to be searched by 80/LINK.
80/LIBLS	Provides a listing of information about a library created by 80/LIBCR.

Available Documentation

A detailed description of the 80/RL tools is contained in "80/RL Relocation and Linking Tools Reference Guide", CFG Order No. 9201-3.

SUPPORTED OPERATING ENVIRONMENT

The 80/DS Development System will operate on the Tektronix 8560 and 8561 computers under the TNIX operating system.

Approximately 1200 disk blocks are required for the installed software.

INSTALLATION

Installation is the responsibility of the customer and requires only general knowledge of the Tektronix 8560/61 and of the TNIX operating system.

ORDERING INFORMATION

The 80/DS Development System is furnished only under the terms and conditions of CFG's "Agreement for CFG Licensed Programs" which provides, in part, that the programs may only be used on the single CPU for which they are licensed. Copies of this agreement, along with current prices and availability, may be obtained by requesting a quotation from CFG at:

Caine, Farber & Gordon, Inc.
750 East Green Street
Pasadena, California 91101 USA

Telephone : (213) 449-3070
Telex : 295316 CFG UR

Locations outside the U. S. contact Caine, Farber, and Gordon for possible local distributor.

The complete 80/DS Development System may be ordered as a single item, or the individual components may be ordered separately. Orders should specify Part Number and Description as:

5660-DS8	Complete 80/DS Development System for Tektronix 8560/61 under TNIX
5660-AS8	80/AS Assembler for Tektronix 8560/61 under TNIX
5660-PC8	80/PC Compiler for Tektronix 8560/61 under TNIX
5660-RL8	80/RL Relocation and Linking Tools for Tektronix 8560/61 under TNIX

DISTRIBUTION PACKAGE

The distribution package consists of:

- One or more flexible diskettes in Tektronix 8560/61 format containing the ordered software;
- One copy of each relevant Reference Guide; and
- One copy of the installation instructions.

NOTICES

RESTRICTED RIGHTS LEGEND

INDEX