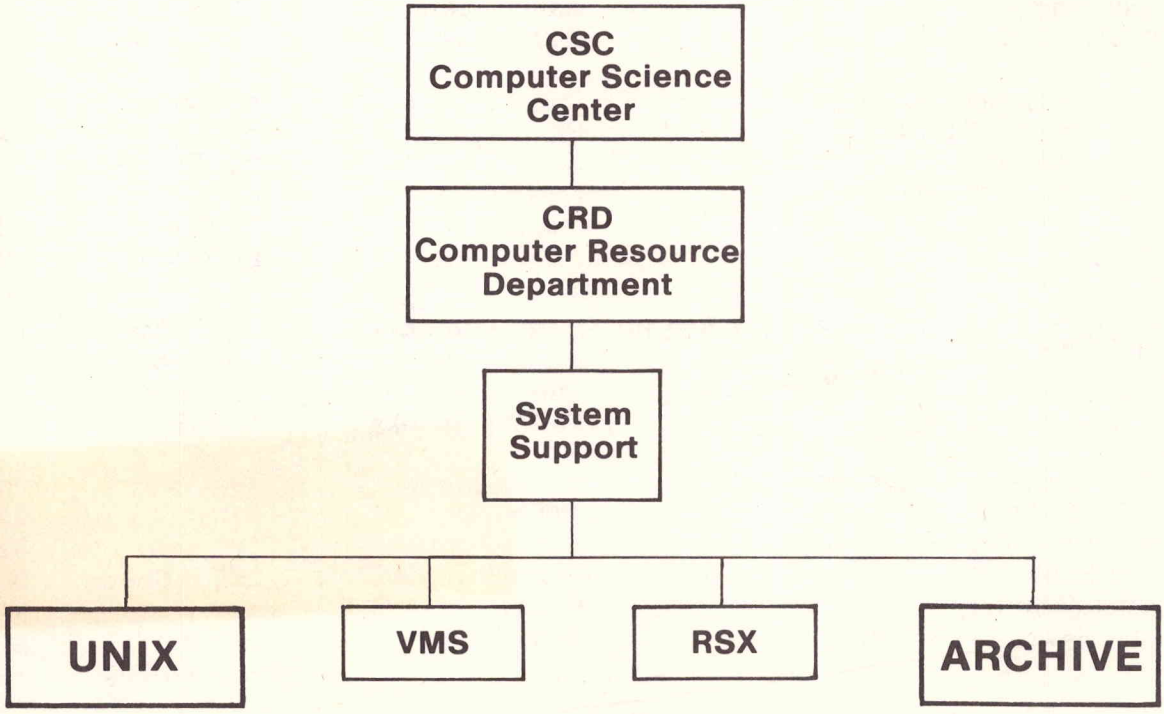
# CSC INTERFACE

## COMPUTER SCIENCE CENTER NEWS

VOLUME 4 NUMBER 1 DECEMBER–JANUARY 1981

Contents:






PAGE 3 PAGE 5

# People and Telephone Numbers to Know

| Department/Service | Name | Ext. | D/S |
|---|---|---|---|
| **COMPUTER RESOURCE DEPARTMENT** | **Bob Mainero** | **BGTE 5357** | **50–454** |
| **Operations Support** | **Dawn Vance** | **BGTE 4600** | **50–454** |
| Hardware Coordinator | Jeff Mulick | BGTE 4600 | 50–454 |
| Data Communications Support** | Mary Ann Feeback | BGTE 4600 | 50–454 |
| Public Support (user areas) | Jeff Mulick | BGTE 4600 | 50–454 |
| Computer Operations Manager | Gayle Monroe | BGTE 5104 | 50–454 |
| Computer Operators | staff | BGTE 5104 | 50–454 |
| Remote Operator (WRIP) | Mary Ann Feeback | WR 1917 | 92–675 |
| Remote Operator (Wilsonville) | Craig Willcox | WI 3037 | 63–397 |
| **Systems and Data Base Management** | **Andy Davidson** | **BGTE 7668** | **50–454** |
| User Numbers/CSC Manuals | Georgene Kayfes | BGTE 6870 | 50–454 |
| Systems Project Leader | Ken Hadfield | BGTE 7668 | 50–454 |
| Systems Programmers/Analysts | staff | BGTE 7668 | 50–454 |
| Data Base & Utility Project Leader | Glen Fullmer | BGTE 5833 | 50–454 |
| Data Base Programmers | staff | BGTE 5833 | 50–454 |
| **Network Development** | **Bob Mainero** | **BGTE 5357** | **50–454** |
| Network Project Leader | Tim Fallon | BGTE 5714 | 50–454 |
| UNIX Support Project Leader | Greg Harris | BGTE 5102 | 50–454 |
| **Technical Communications Manager** | **Carolyn Schloetel** | **BGTE 4855** | **58–122** |
| CSC INTERFACE Editor | Rich Amber | BGTE 4945 | 58–122 |
| Component News Editor | Jacquie Calame | BGTE 6867 | 58–122 |
| ManuFACTuring Editor | Mike Quigley | BGTE 6867 | 58–122 |
| SEMINARS Editor | Lola Janes | BGTE 6867 | 58–122 |
| Tektronix Sourcebook Editor | Jacquie Calame | BGTE 6867 | 58–122 |
| HELP file maintenance | Nancy Peate | BGTE 6244 | 58–122 |
| WRITEUP file maintenance | Cliff Morgan | BGTE 6244 | 58–122 |
| **Applications Support Manager** | **Imants Golts** | **BGTE 4675** | **50–454** |
| User Assistance, LOBARB | Paul Hoefling | BGTE 4004 | 50–454 |
| Statistical Programs | Carol Golding | BGTE 5976 | 50–454 |
| Graphics, IMSL, VARIAN | Kurt Krueger | BGTE 5976 | 50–454 |
| Process Simulation | Imants Golts | BGTE 4675 | 50–454 |
| N5500, PERT | John Burley | BGTE 4675 | 50–454 |
| **MICROPROCESSOR SUPPORT MANAGER** | **Lynn Saunders** | **WR 1910** | **92–134** |
| Portable High Level Languages | Lynn Carter | WR 1181 | 92–134 |
| Microprocessor Software | Sue Anne Smith | WR 1890 | 92–134 |
| Microprocessor Hardware | Ferrous Steinka | WR 1920 | 92–134 |
| **CAD/CAM DEVELOPMENT MANAGER** | **Ron Bohlman** | **WR 1141** | **92–112** |
| 4081 CAD Stations | Jim Murphy | WR 1146 | 92–112 |
| Automated Routing (ECB Design) | Roger Bonzer | WR 1152 | 92–112 |
| Circuit Simulation | Graeme Boyle | BGTE 5866 | 50–454 |
| Component Library | Phil White | WR 1145 | 92–112 |
| Mechanical CAD/CAM Support | Barry Ratihn | BGTE 7789 or WI 3791 | 50–454 |

**For Telecommunications repair, call Beaverton DanRay, ext.5040

# UNIX OPERATING SYSTEM

"UNIX" is the name of a time–sharing system which runs on most models of Digital Equipment Corporation's PDP11 and VAX computer systems. Written at Bell Labs, it was first announced to the world in 1974.

UNIX is here for a number of reasons, not the least of which include that it runs on a system which is already available, it is compact and accessable, it provides an extensive set of very usable facilities, and not only is intrinsically interesting, but actually breaks new ground in a number of areas.

Not the least amongst the charms and virtues of the UNIX time–sharing system is the compactness of its source code, which, for the permanently resident "'nucleus" of the system when only a small number of peripheral devices is represented, is comfortably less than 9000 lines of code.

It has often been suggested that 10,000 lines of code represent the practical limit in size for a program which is to be understood and maintained by a single individual.

Most operating systems either exceed this limit by one or two orders of magnitude, or else offer the user a very limited set of facilities; i.e., either the details of the system are inaccessible to all but the most determined individuals, or the system is rather specialized.

Not only is UNIX effective, but it is accessible in a way that most other systems are not: the amount of material which must be mastered in order to gain a reasonably deep understanding of the system is not impossibly large.

Much of the effectiveness of UNIX derives from simple and direct implementation, using an appropriate high level language called "C." There is very little assembly code in the operating system and very seldom is it ever modified.

## UNIX Operating System

The UNIX Operating System, namely the code which is permanently resident in the main memory during the operation of UNIX, has the following major functions:

Initialization
Process management
System calls
Interrupt handling
Input/Output operations
File management

## Utilities

The remaining part of UNIX is composed of a set of suitably tailored programs which run as "user programs" and may be termed "utilities."

Under this heading come a number of programs with a very strong symbiotic relationship with the operating system, such as the "shell" (command language interpreter), "/etc/init" (terminal configuration controller) and a number of file system management programs.

It should be pointed out that many of the functions carried out by the utility programs are regarded as operating system functions in other computer systems, and that this certainly does contribute significantly to the bulk of these other systems as compared with the UNIX Operating System.

In addition to the utilities which one normally expects to find included with an operating system, there are several hundred additional software tools which make everyone's task easier. They range from tools to assist compiler developers to write compilers, to tools to check spelling and grammar in memos. New tools are constantly being developed, and because over 1000 UNIX systems are in use in universities, most software is in the public domain.

Descriptions of the functions and use of the UNIX utilities may be found in the "UNIX Programmer's Manual" (UPM). On the whole, you will find that the authors of UNIX have created a program of great strength, integrity, and effectiveness.

(Continued next page)

## UNIX Support

The Computer Resource Department has been chartered to support inhouse use of UNIX. This includes:

License Support –

| | |
|---|---|
| Version 32 | VAX |
| Version 7 | PDP11/44, 11/70 |
| Version 7 | PDP11/23, 11/34 |

The Computer Resource Department will obtain the software license and handle all of the required accounting to Western Electric.

Tek Standard System Software –
(UNIX Version 7 as modified by U.C.Berkley)

Distribution –
System updates (semi–annual)
Software tools and updates (semi–annual)
Newsletter.

Standardization –
Collection and distribution of enhancements
Documentation (on line documentation and manuals)

Operations –
Provide start–up support
Provide UNIX trained operators
File backups during offshifts
and Offsite tape storage (see Archive article).

The System Support group within CRD also acts as a "Software Warehouse" for specialized tools and device drivers. For further information regarding UNIX, contact Greg Harris, ext. BGTE 5102.

# CYBER USER AREAS

The Computer Resource Department supports public user areas in the following locations:

| | | | |
|---|---|---|---|
| 19–092 | 39–007 | 48–268 | 50–464 |
| 53–063 | 58–178 | 60–725 | 63–397 |
| 73–316 | 78–537 | 92–567 | 92–675 |

For a complete list of equipment located in these buildings, type: **HELP,USEAREA.**

Any problems with this equipment should be reported to us immediately at ext. BGTE 4600 during the day, or ext. BGTE 5104 after hours. This includes data line problems, line printer problems and terminal problems. If you should experience any problems, please do not move, repair, or disconnect any user area equipment. Also, please help us keep the areas clean by recycling old listings and being responsible for cleaning up after your work session is through.

Here are some numbers to call if you have problems with *your own equipment:*

1– For communications problems, call Telecommunications, ext. BDR 5040
2– For hardware/terminal problems, call Factory Service, ext. MR 8600.

# NETWORK UPDATE

Two issues ago, we told you about our local network project. Well, folks, Tim Fallon says that the Hyperchannel network hardware to link the CYBER 175 to the Tek Labs PDP 11/70 has arrived and was physically installed on January 9th. That doesn't mean you can run right out and transfer data yet. That part is still a stretch down the road, but we are a few steps closer.

Another forward step: UNET has been installed on the 11/70 UNIX system. UNET, which was purchased from 3COM Corporation, is an implementation of the Department of Defense's Standard Internet Protocol (IP) and Transmission Control Protocol (TCP). We are currently implementing these same high level protocols on the CYBER 175. Stay tuned for further updates.

# CORPORATE ARCHIVING SYSTEM

## The Archive Problem

Tektronix has a policy of maintaining products in the field for nine years after their manufacture has stopped. Because electrical and mechanical parts will perform specific functions, they can be tested to see if they do indeed perform that function, and replaced in the field if they do not function properly. This is not the case for software and firmware. Maintenance of those products containing software and/or firmware requires a different approach.

There are basically three levels of software maintenance (which includes firmware; as firmware is a program stored in a ROM):

1– Only replacement of original code
2– Fix program errors in the original code
3– Introduce new options or improve original code

Each level includes the lower levels. The lowest level of maintenance (1) would be the easiest to provide. One need only have available the original object code; it could be saved on any number of media: papertape, microfiche, magnetic tape, etc. A very simple archiving scheme would easily provide the capability for this level of maintenance.

An archive is a place in which materials are preserved. An archiving scheme is a determination of which materials are to be preserved and the methods to be used in their preservation. Historically, each product group has maintained its own archives for its own products. However, with the increased complexity of today's systems and growth of groups and the company, to be truly effective in providing the capabilities needed to support the corporate policy of nine years of maintenance after production has stopped, there needs to be an archive scheme which is also corporate wide.

## Archive History at Tek

It has traditionally been the responsibility of individual product groups to support and maintain the products they develop. The introduction of software into Tek did not immediately alter that tradition. However, approximately four years ago, a committee was formed to establish guidelines for archiving software.

The Computer Science Center has been involved with archiving software for a number of years. In 1978, Andy Davidson wrote a program that allowed software designers using the CYBER in bldg. 50 to transfer files to microfiche, which could then be stored by Reprographics.

New Product Introduction recognized in 1978 that it needed a method of ensuring that software had been archived. This gave rise to the establishment of a software archiving policy that product groups were supposed to follow before PSR. This policy established a minimum of what should be archived as follows:

1– An Index of material to be archived
2– All information needed to reproduce the product in shippable form from the original source code, *assuming the original development compiler, operating system, hardware, etc. is available.*
3– Uncompiled source code
4– Compiled source listings
5– Object code for shipped product
6– All final internal and external specifications of the product
7– Product code of the actual medium for the software as shipped
8– Other applicable material

Product groups could archive as much material as they wanted over and above the listed minimum. The presumption is clear that, in the case of hardware or software tool changes, enough material would be archived to allow programmers to quickly modify the source so it could run on the changed tools.

(Continued next page)

Then Tom Hamilton, of Al Zimmerman's Digital Products Coordination group in Tek Labs, proposed a revision to Gary Hamrick's previous proposal, which added a tape format standard and the use of a database for cross–reference information. It also expanded the amount of material to be archived to a maximal set – all software and hardware in every computer system used for software development. Hamilton's presumption is that no modification of the archived source should be necessary to regenerate the original object code for the shipped product. Kurt Krueger, CRD, recently wrote a program which creates tapes on the CYBER using Hamilton's tape standard; and Greg Harris, CRD, is creating the cross–reference database.

## Archiving Plans

The corporate wide archiving system, as of this date, calls for the installation of a tape vault for storage in Reprographics, bldg. 58 (installation to be complete this month), and another in bldg.71 (installed by July 1981). Also, a DEC computer system will be installed to perform tape copies and maintain the library (installed by Fall 1981). A software engineer will be hired, whose initial duties will be to work on the system design. It is expected that the complete archive implementation will be accomplished by January 1982.

## How It Will Work

During product development, backup tapes of work in progress (WIP) will be sent to archive (see Fig.1). Before "PSR OK" is received, Corporate NPI will independently verify that the product's software and/or firmware is archived. Manufacturing will then get the approval to retrieve a copy of the software/firmware from the archive.

Corporate Engineering change control procedures will be modified to insure that all software/firmware to be changed comes from the archive, and that before the change is released, the modifications are placed in the archive.

Field Service will request all updates from the archive and Instrument Control will track hardware modifications to development systems, and when the systems are

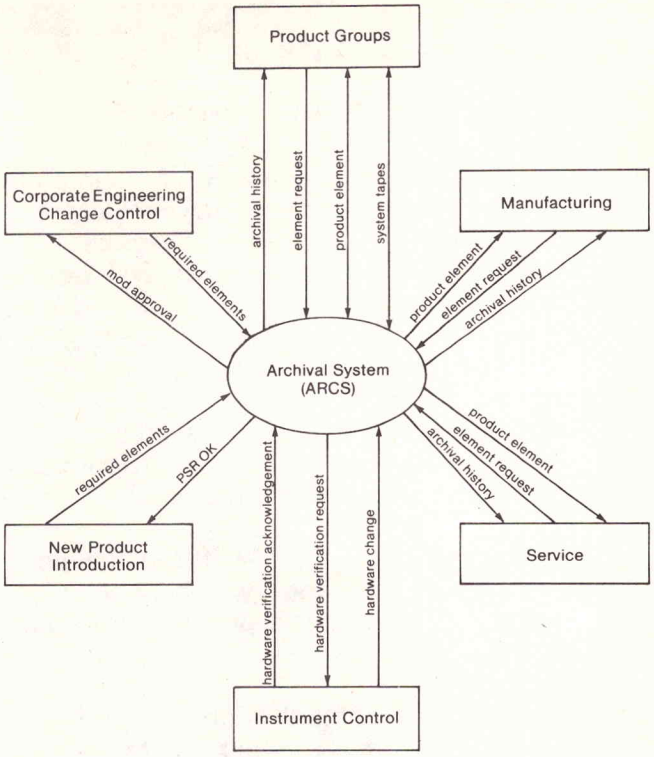



**Figure 1 — Data Flow Diagram**

replaced with non–compatible equipment, Instrument Control will store and maintain the hardware for the product's life.

The keys to the success of the archive to protect Tektronix from great amounts of lost time and inability to support products for their planned life are: 1– All software/firmware which goes to a customer comes from an archive copy, and 2– complete systems of hardware and software are maintained that allow for modification and regeneration of the software from the source.

## Where Are We Now?

As stated previously, the vault in Reprographics will be finished this month. We are currently accepting tapes into archive (including WIP tapes), but as yet do not have the capability to copy them.

This system is available for use by anyone who "wants" to use it, but take note that tapes which go into archive *must* conform to Hamilton's standard. TF tapes are *not* acceptable.

# PROGRAMMING EFFICIENCY

## Reducing Memory

Installment Three

by

Kurt Krueger


*O.K., you've done everything possible to reduce the size of your program. All of the earlier articles had useful hints, but the program is still too big. The boss says it costs too much to run and unless you can shrink it some more, you stand to lose six months work! Never fear, I still have some tricks up my sleeve.*

The previous articles have described methods that don't require too much knowledge of the individual computer or require the use of non–standard extensions to FORTRAN. Sometimes that is not enough. When it's make it or break it time, anything is worth trying.

The CYBER word size is 60 bits, but due to hardware design limitations, only 48 bits are available for integer computations. This is really not all that bad because 48 bits can hold a 14–digit base 10 number (max value is $2.814...X\ 10^{14}$).

Fourteen digits is an awful lot. Most integer numbers commonly encountered are at most, five or six digits in length. There is actually room for two or more values in each CYBER word. What remains to be seen is how to manipulate them... Example:

An integer array consists of values that are, at most, seven digits long. Two of these numbers could be placed in one variable as follows (using 1242 and 7621 as examples):

**00012420007621**

A FORTRAN statement to accomplish this is:

```
I=1242  J=7621
IPACK=I*10000000+J
```

And statements to undo it could be:

```
I=IPACK/10000000
```

Truncation leaves the 1242

```
J=MOD(IPACK,10000000)
```

Remainder is 7621

This process is commonly referred to as PACKING and UNPACKING. It should be noted that the number used to pack and unpack does not have to be a power of 10. The only thing that is necessary so that we can undo the packing is that our number be greater than the maximum number we are packing.

This method works fine, but because a computer works in base 2 instead of base 10, it is much more efficient to allocate a certain number of base 2 digits (bits) instead of base 10 digits. We make the observation that a seven digit decimal number will require 24 bits ($2^{24}$ =16,777,216 while $2^{23}$=8,388,608). We can pack and unpack the same way, but use our power of 2 number instead of the power of 10 number:

```
I=1242  J=7621
IPACK=I*16777216+J
```

To undo:

```
I=IPACK/16777216
J=MOD(IPACK,16777216)
```

(Continued next page)

The whole problem *seems* to have become more complicated. It has for humans, but has become much more simple for the computer. Why? Because in base 2, multiplication or division by a power of 2 number involves adding zeros or removing digits, the same way that multiplication and division by powers of 10 work on base 10 numbers. As long as the base 2 number is written as a *constant* in the expressions, the CYBER will recognize this and use fast instructions to multiply and divide. It should be noted that not all computers will recognize the base 2 number, and thus, there may be no gain over the earlier base 10 example. On the CYBER, the gain is very significant. Also, due to features of the machine, all 60 bits can be used when power of 2 constants are used to pack and unpack.

The fact that these power of 2 numbers (e.g. 16777216) *must* always appear as constants can be annoying because the numbers are always rather strange looking. This can be remedied by converting the base 2 number to a base 8 number. In our example, $2^{24}=(2^3)^8=1X8^8$ or $100{,}000{,}000_8$. Taking advantage of the fact that blanks are ignored in FORTRAN, we can rewrite our expressions as:

```
I=1242
J=7621
IPACK= I*100 000 000B
```

where the **B** after the constant tells CYBER it is base 8, not base 10.

## Another Approach

This problem can also be solved with SHIFT and Boolean functions, AND and OR. These functions are available to FTN. Their usage is beyond the scope of this article. Details are in the FORTRAN IV Reference Manual.

A couple of processors are available to automatically (and efficiently) pack and unpack variables without any major modifications to FTN source code. These are experimental programs. Usage can be arranged by contacting Kurt Krueger, ext. BGTE 5976.

# RJE CLASSES

Classes are being scheduled for all users interested in the basic operation of RJE printers. These classes will be helpful to *all* users, and particularly those of you who will be using the RJEs during the budget cycle. Classes will be held at the RJE locations in the following buildings:

| | | | |
|---|---|---|---|
| 39–007 | Tues | March 10 | 10:00 AM |
| 19–092 | Tues | March 10 | 1:00 PM |
| 63–397 | Wed | March 11 | 10:00 AM |
| 92–675 | Wed | March 11 | 1:00 PM |

A class can be scheduled in any additional buildings if there are a number of people or groups who request it. If you require an additional class, or have any questions, please call Mary Anne, ext. BGTE 4600.

## February Shutdown

During the shutdown week of Feb 9–13, the CYBER will operate as normal. However, the weekend before, CYBER will go down at 0500, Saturday, Feb 7 and be down for the weekend. Also, the weekend following, CYBER will go down at 0500, Saturday, Feb 14, and stay down the remainder of the weekend. At 0000, Monday, Feb 16, CYBER will once again be awakened to resume normal schedules.

## DEC Users Group Meeting

There will be a meeting of all interested DEC users on February 5th, 11:00 AM – noon in Walker Road conference room 2N, rooms 11–13. On the agenda are:

Non–DEC Peripherals
Computer Supplies
UNIX Support

# MINPLOT

## Memory Saving Plot Package

by Kurt Krueger

Here is good news for programmers who want to take advantage of the feature of the Advanced Graphics package in PLOTLIB, but are a little tight on CPU memory. Advanced Graphics (AGII) is the portion of PLOTLIB that corresponds to calls of BINITT, CHECK, DISPLAY, etc.

The new library (called MINPLOT) contains the basic features of AGII, but is 10K (octal) words smaller. The function of the routines in MINPLOT is similar to those in PLOTLIB. No attempt has been made, however, to make the graphs produced by MINPLOT identical to those produced by PLOTLIB, nor to include all the features of PLOTLIB (it wouldn't be a mini package anymore). What has been created is a package that for most programs will produce a high quality graph (see Fig.1) using the same calls as PLOTLIB. Existing programs that use AGII should be able to use MINPLOT and benefit from the memory savings. If you're not sure if MINPLOT will work, the best thing to do is try it. If it does work for you, you've just saved 10K of memory (and some bucks).

The following sequence of commands can be entered to use MINPLOT:

```
ATTACHL,MINPLOT
FTN,I=pgm-name
LINK,P=MINPLOT,X
```

Subroutines available in MINPLOT:

| | | |
|---|---|---|
| BINITT | CHECK | CPLOT |
| DSPLAY | DLIMX | DLIMY |
| LINE | NPTS | SLIMX |
| SLIMY | XTYPE | YTYPE |

(Continued next page)

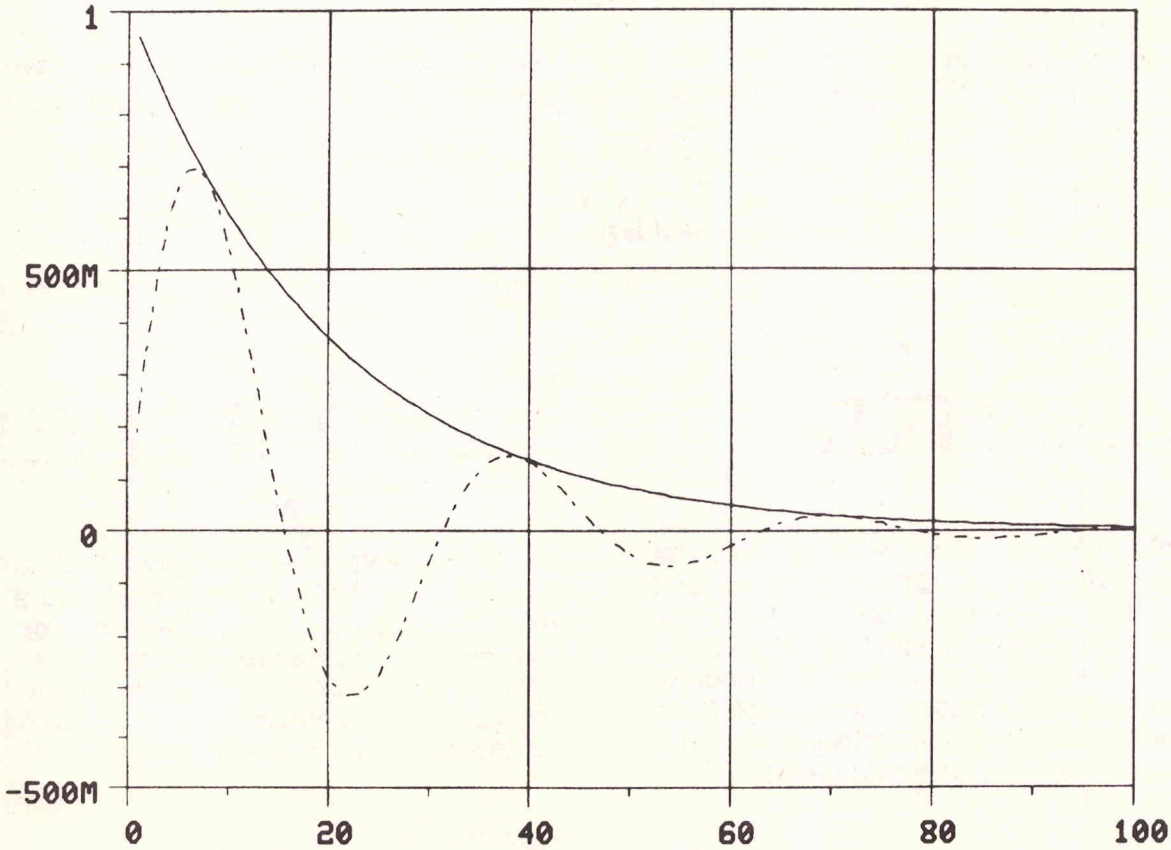



Fig.1 – Sample plot. Note engineering units on labels.

These subroutines have the same argument lists as their counterparts in AGII. Refer to the AGII manual (Tek PN 070–2244–00) for details of their use. In addition, all needed routines from TCS (such as INITT, ANMODE, etc.) are also included in MINPLOT.

*CAUTION:* MINPLOT *cannot be used with* PLOTLIB. Only one plot library can be used at a time.

### Status of the Package

MINPLOT is a fully supported operational package. A few bugs are likely to remain because this is the first release of the package. All reports of bugs should be directed to Kurt Krueger at ext. BGTE 5976. New features may be added very reluctantly (if at all); the desire being to keep this package small. MINPLOT will never totally replace AGII. Programs that depend on the fancier features of AGII will have to continue to use PLOTLIB.

Acknowledgement: I wish to thank Doug Brown (formerly CSC, now MDP) for providing several programs that form the basis for MINPLOT. –Kurt Krueger

# HELP & WRITEUP FILE STATUS

Help and Writeup files changed or added since last issue:

HELP files changed:

ALTFAM NOS13 MAGTAPE SINCMC 509 ECHO IGP

Due to policy change in review method for updating HELP files, there are many which will have a revision date without actually having been changed. These have not been listed as there was no content change.

No WRITEUPs have been changed since last issue.

# 509 NOTE

As most of you no doubt are aware, the 509 Operating System for CYBER was up and running again on January 10th, and to date is still running full steam ahead. There have been minor problems, which is to be expected, however, Andy Davidson, manager of Data Base Support says, "It looks extremely stable." With a comment like that, what more is there to say...

# CAD MOVED

CAD Development, the group that brought you ASAP, PIRATE, WIREWRAP and a host of other Computer Aided Design programs, has moved from their executive suites at Town Center to Walker Road Industrial Park. Their new delivery station is 92–112. Check page 2 of this paper for phone numbers.

## Data Communications to the CYBER and DEC10 Computers

Telephone numbers to reach the CYBER computers (VADIC & 212 MODEM compatible) as of 29 May 1980:

FROM 4000 — 4599 EXTENSIONS:
300/1200 Baud — Dial 3

FROM ANY OTHER EXTENSION:
300/1200 Baud — Dial 84 + 3

FROM AN RJE STATION:
641-1685

FROM Y-3:
300/1200 Baud — Dial 4 + 3

THROUGH BEAVERTON EXCHANGE:
300/1200 Baud — 641-1650

THROUGH TIGARD EXCHANGE:
300/1200 Baud — 620-3800

FROM WILSONVILLE:
Building 60
1200 Baud only — Dial 2025 then *N
Building 63
1200 Baud only — Dial 4225 then *N

**NOTE**

***N must be used to clear the telephone line whenever a session from Wilsonville is finished.*

Telephone numbers to reach the DEC10 computer (Wilsonville):

FROM BEAVERTON INDUSTRIAL PARK:

300 Baud — 4000 - 4599 Extensions — 7 + 2803
Other Extensions — 84 + 7 + 2803

1200 Baud — 4000 - 4599 Extensions — 7 + 2812 (VADIC Modem only for 1200 Baud)
Other Extensions — 84 + 7 + 2812

FROM OUTSIDE LINES:

| | | |
|---|---|---|
| 300 Baud — | DAY: | 682-3411, then ask for ext. 2803 |
| | NIGHT: | 682-3431, then ask for ext. 2803 |
| | | 682-3433, then ask for ext. 2803 |
| 1200 Baud — | DAY: | 682-3411, then ask for ext. 2812 |
| | NIGHT: | 682-3435, then ask for ext. 2812 |
| | | 682-3437, then ask for ext. 2812 |

The telephone number for Telecommunications data communications *repair* has been changed from ext. B 5541 to ext. 5040 DanRay. The delivery station, 55–241, remains the same.

1) Nature of the problem (constant ringing, no dial tone, etc.)

2) Which telephone number is affected.

3) When the problem first appeared.

4) Your name and telephone number.

When your trouble is reported, a dispatcher will assign someone to your problem. That person should contact you as soon as possible about the problem. If you have any problem with this service, or do not get a timely response from Telecommunications, please contact Jeff Mulick at ext. 4600.

Mary Ann Feeback, ext. BGTE 4600

# Computer Science Center Statistics

| | |
|---|---|
| **CYBER 175 System Availability** | **99.00%** |
| Downtime due to software | .18% |
| Downtime due to hardware | .14% |
| Tek downtime | .26% |
| Unresolved failure | 0% |
| Interruptions due to frontend | .42% |
| MTBF | 132.61 Hours |

| | |
|---|---|
| **CYBER 73 System Availability** | **98.44%** |
| Downtime due to software | .17% |
| Downtime due to hardware | .97% |
| Tek downtime | 0% |
| Unresolved failure | .04% |
| Interruptions due to frontend | .38% |
| MTBF | 60.17 Hours |

| | |
|---|---|
| **Frontend Availability** | **99.64%** |
| Downtime due to software | .08% |
| Downtime due to hardware | .28% |
| Tek downtime | 0% |
| Unresolved failure | 0% |
| MTBF | 77.22 Hours |

Bob Mainero, ext. BGTE-5357

The deadline for articles to be submitted for publication in the next issue of CSC INTERFACE is tentatively set for March 20, 1981. Thank you.


CSC INTERFACE

**Published by**
**Technical Communications**
**58 - 122**
Rich Amber, Editor
Cartoons, typesetting
Paste-up
Loretta Clark, graphics

To submit an article, contact Rich, ext. BGTE 4945

For mailing list additions or changes, contact Jill Miller, ext. BGTE 5502


78-557
MAIL LABEL GOES HERE
PAUL E GRAY
CSC INTERFACE

**company confidential**