1961-1986

Leading
Toward
The
Future

DECUS U.S. CHAPTER SIGs NEWSLETTERS

APL SIG .... APL
ARTIFICIAL INTELLIGENCE SIC .... AI
BUSINESS APPLICATIONS SIC .... BA
COMMERCIAL LANGUAGES SIG CL
DAARC SIG .... DAR
DATA MANAGEMENT SIG .... DMS
DATATRIEVE SIG .... DTR
EDUSIG .... EDU
GRAPHICS SIG .... GRA
HARDWARE MICRO SIG .... HMS
IAS SIG .... IAS
LANGUAGES AND TOOLS SIG .... L&T
LARGE SYSTEMS SIG .... LS
MUMPS SIG .... MMP
NETWORKS SIG .... NTW
OFFICE AUTOMATION SIG .... OA
PERSONAL COMPUTER SIG .... PC
RSTS SIG .... RST
RSX SIG .... RSX
RT SIG .... RT
SITE MANAGEMENT & TRAINING SIG .... SIT
UNISIG .... UNI
VAX SYSTEMS SIG .... VAX
LIBRARY INFORMATION SECTION .... LIB
"HOW TO" SUBMIT AN ARTICLE... GROUP FORMS .... HOW
QUESTIONNAIRE SECTION .... QU

JUNE 1986 Volume 1, Number 10

GENERAL TABLE OF CONTENTS

SECTIONS **PAGE NO.**

APL SIG
. Steering Committee Listing .... APL-i

ARTIFICIAL INTELLIGENCE SIG
. Steering Committee Listing .... AI-i

BUSINESS APPLICATIONS SIG
. Steering Committee Listing .... BA-i

COMMERCIAL LANGUAGES SIG
. Steering Committee Listing .... CL-i
. DIBOL Update CL-1

DATA ACQUISITION, ANALYSIS, RESEARCH AND CONTROL SIG
. Steering Committee Listing .... DAR-i

DATA MANAGEMENT SIG
. Steering Committee Listing .... DMS-i
. One Man's Experience with RDB and SMARTSTAR DMS-1
. The relationship of OPAL/REQUEST/RDB DMS-3
. Querying the Database not using the DAR DMS-7
. Background on OPAL DMS-8
. Testing the status of request calls DMS-9
. DAR and Data action failure DMS-13
. Multiple transaction rollback DMS-21

DATATRIEVE/4GL SIG
. Steering Committee Listing .... DTR-i
. Chairman's Corner DTR-1
. From the Editor's Pen DTR-2
. Ask the Wombat Wizard DTR-3
. Introducing the Product Improvement Request System DTR-7
. VAX-11 Datatrieve DTR-8
. CDD DTR-16
. Index of WOMBAT Examiner Articles DTR-18
. About the Cover DTR-31
. Product Improvement Request Submission Form QU-11
. DTR/4GL SIG Spring 1986 PIR Ballot QU-13
. Ask the Wombat Wizard Submission Form QU-15

EDUSIG
. Steering Committee Listing .... EDU-i

GRAPHICS APPLICATIONS SIG
. Steering Committee Listing .... GRA-i

HARDWARE MICRO SIG
. Steering Committee Listing .... HMS-i

IAS SIG
. Steering Committee Listing .... IAS-i
. Dear IAS Enthusiast IAS-1
. Trivia for March and April IAS-3
. Software Performance Report (SPR's) IAS-5
. Writing MACRO Subroutines for FORTRAN Programs IAS-23
. Reese Basic (The Other Basic) IAS-32

LANGUAGES AND TOOLS SIG

. Steering Committee Listing L&T-i

LARGE SYSTEMS SIG

. Steering Committee Listing LS-i

MUMPS SIG

. Steering Committee Listing MMP-i

NETWORKS SIG

. Steering Committee Listing NTW-2
. In This Issue NTW-3
. The Editor's Cobwebb NTW-4
. New Ways of Communicating With IBM: A User View NTW-8
. New Alternatives In Local Network Technologies NTW-12
. DATAGRAM NTW-19
. DATAGRAM QU-7

OFFICE AUTOMATION SIG

. Steering Committee Listing OA-i
. In This Issue OA-1-1
. From The Editor OA-1-2
. Why Reordering A Forms Library Doesn't Work With ALL-IN-1 OA-2-1
. ALL-IN-1 Hints and Kinks OA-2-2
. Patches For ALL-IN-1 Version 2.0 OA-6-2
. OA SIG System Improvement Request Submission form QU-9

PERSONAL COMPUTER SIG

. Steering Committee Listing PC-i
. Table of Contents PC-iii
. Submitting Articles to the PC SIG Newsletter PC-1
. From The Chairman PC-2
. Contact List PC-4
. Hanging In There PC-6
. Orphan Computers PC-6
. Dallas DECUS Symposia Trivia Questions PC-7
. RAINBOW PC-10
. Winchester Specifications PC-41
. Word Perfect and the Rainbow PC-47
. DECmate PC-53
. PRO PC-60
. 1986 DEC Professional 300 Wish List Voter Results PC-63
. Memory Expansion for the PRO-350 PC-66
. PRO 350 Memory Modification PC-69
. PC Postscript QU-17
. Personal Computing Special Interest group volunteer Form QU-19
. Information Resource Sign Up Sheet QU-21
. DECUS Personal Computer SIG Questionnaire QU-23

RSTS SIG

. Steering Committee Listing RST-i
. SIG Activities Report RST-1

RSX SIG

. Steering Committee Listing RSX-i
. Table of Contents RSX-1
. From the Editor RSX-2
. How to Contribute to the Multitasker RSX-2
. An RSX-11M/M-Plus Application Note RSX-10
. Full-Duplex Terminal Driver Hooks for ACD Processing RSX-13
. Chris Isbell/Kenda Electronic Systems RSX-20

RT SIG

. Steering Committee Listing .... RT-i
. From The Editor RT-1
. Completion and Interrupt Routines in Oregon Software Pascal-2 .... RT-3
. Bad Block Tech Tip RT-5
. RT-11 (TSX+)—VMS File Transfer RT-6
. FORTRAN-77/RT-11 Differences Between Versions V5.0 and V5.0A RT-7

SITE MANAGEMENT & TRAINING SIG

. Steering Committee Listing .... SIT-i

UNISIG

. Steering Committee Listing .... UNI-i

VAX SYSTEMS SIG

. Steering Committee Listing VAX-2
. Editor's Workfile VAX-3
. Just a Modest VMS Proposal VAX-4
. Spring 1986 SIR Ballot Results VAX-7
. VAX System SIG Committee List VAX-18
. INPUT/OUTPUT VAX-21
. Input/Output Submission Form QU-1
. System Improvement Request Submission Form QU-3

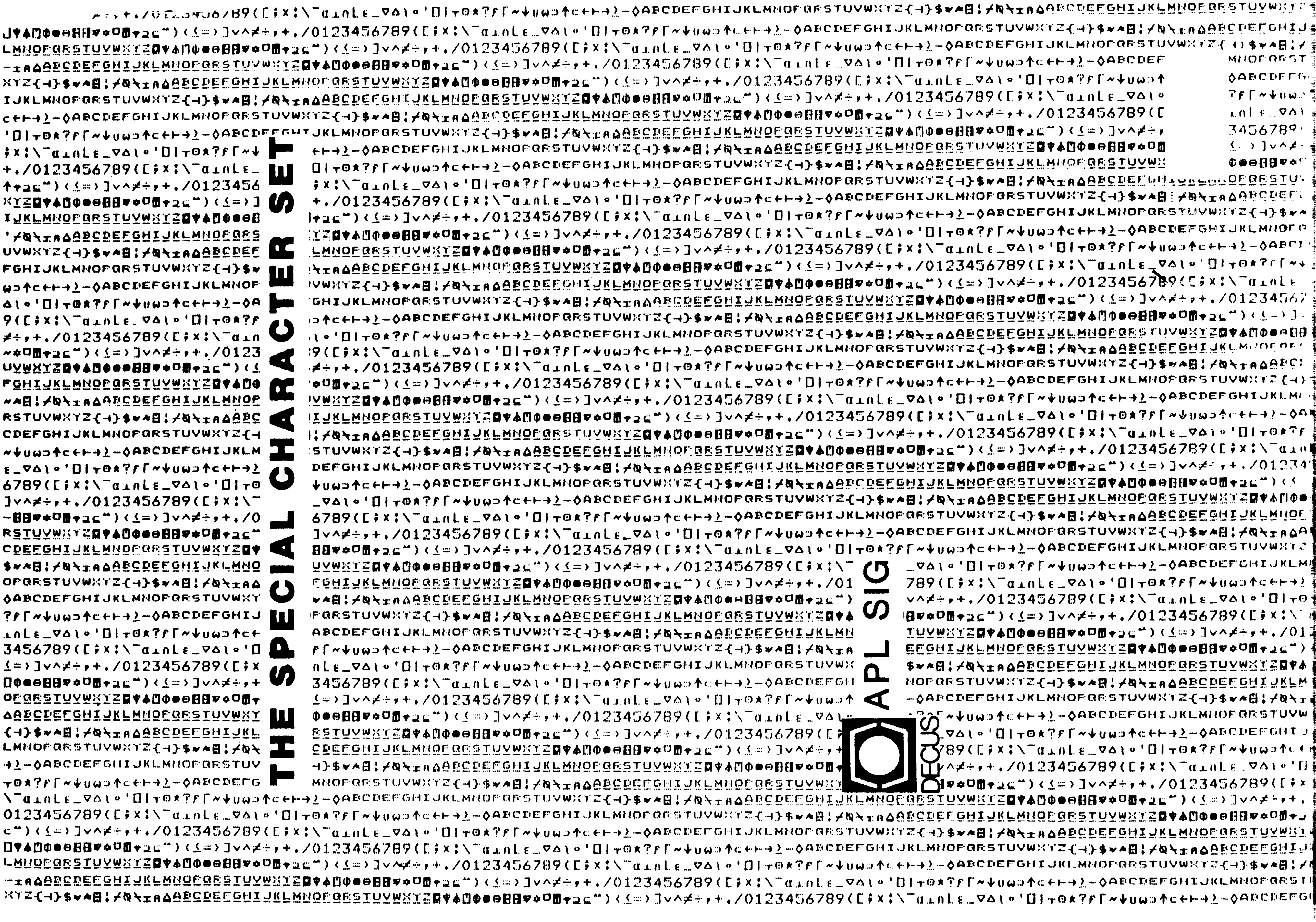
LIBRARY LIB-1

HOW TO SUBMIT AN ARTICLE . . ORDER FORMS SECTION

. How to Submit An Article .... HOW-i
. Article Submission for HMS HOW-1
. Subscription Service Order Form HOW-3
. Membership Application Form HOW-5

QUESTIONNAIRE SECTION

. INPUT/OUTPUT Submission Form QU-1
. System Improvement Request Submission Form QU-3
. VAX System SIG Spring 1986 SIR Ballot QU-5
. DATAGRAM QU-7
. Office Automation SIG System Improvement Request Submission Form QU-9
. Product Improvement Request Submission Form QU-11
. DTR/4GL SIG Spring 1986 PIR Ballot QU-13
. Ask the WOMBAT Wizard Submission Form QU-15
. PC Postscript QU-17
. Personal Computing Special Interest Group Volunteer Form QU-19
. Information Resource Sign Up Sheet QU-21
. DECUS Personal Computer SIG Questionnaire QU-23



THE SPECIAL CHARACTER SET

APL SIG

DECUS

APL SIG

Chairman (RSTS)
Larry LeBlanc
Teletype Corp.
Elk Grove, IL 60007

Library Coordinator
Susan Abercrombie
Ventrex Laboratories
Portland, ME 04103

Newsletter Editor (RT, TSX)
Douglas Bohrer
Bohrer and Company
Wilmette, IL 60091

Symposia Coordinator (RSX)
Bob Awde Jr.
General Mills Inc.
Minneapolis, MN 55427

Symposia Assistant
Bob Van Keuren
UserWare International, Inc.
Escondido, CA 92025

Standards Representative
Dan Esbensen
Touch Technologies
Escondido, CA 92025

VAX APL (Contact:)
OPEN

European Contact (TOPS)
Jean-Pierre Barasz
BARTS
75008 Paris, France

Digital Counterpart
Dave Quigley
Digital Equipment Corporation
Nashua, NH 03062

ARTIFICIAL INTELLIGENCE

DECUS

AI

ARTIFICIAL INTELLIGENCE SIC

Chairperson
Cheryl Jalbert
JCC
Granville, OH

Ass't Chairman
Don Rosenthal
Space Telescope Science Inst.
Baltimore, MD

Symposium Coordinator
David Slater
Institute for Defense Analysis
Alexandria, VA

Ass't Symposium Coordinator
Session Note Editor
Greg Parkinson
Cognitive Systems Inc.
New Haven, CT

Newsletter Editor
Terry Shannon
Digital Review
Boston, MA 02109

Newsletter Publisher
Bob Zeek
Pfizer Inc.
Groton, CT

Membership Coordinator
Pamela Vavra
KMS Fusion Inc.
Ann Arbor, MI

PSS Scheduling
Tom Viana

Store & Buttons
Sally Townsend
Inst. Defense Analysis
Alexandria, VA

Quality Control Chair
Dick Ciero
Harris Corp.
Palm Bay, FL

Quality Control
Carol Guyermelli

Site Coordinator, Anaheim
Chris Goodard

Volunteer Coordinator, Anaheim
Peter MacDonough
Tractor Inc.
California, MD

Members-at-Large
George Humfeld

Matt Matthews IV

Evaluation Research Corp.
King George, VA

DEC Counterpart
Art Beane
Digital Equipment Corporation
Hudson, MA

DECUS

BUSINESS APPLICATIONS

BA

BUSINESS APPLICATIONS SIC STEERING COMMITTEE

Chairman
Stuart Lewis
Douglas Furniture
Bedford Park, IL

Symposium Coordinator
Steve Simek
IRT Corporation
San Diego, CA

Asst. Symposium Coordinator
Bobbie Wiley
CEI Perry Nuclear Power Plant
Euclid, OH

LRP and Marketing Coordinator
Arnold I. Epstein
D-M Computer Consultants
Rolling Meadows, IL

Marketing Asst.
George Dyer
Gallaudet College
Washington, DC

Communications Representative
OPEN

Newsletter Editor
Thomas Byrne
L. Karp and Sons
Elk Grove Village, IL

Session Notes Editor
Raymond Swartz
Goodyear Tire and Rubber Co.
Akron, OH

Library Representative
David Hittner
Projects Unlimited
Dayton, OH

CL SIG Liaison
Becky Burkes
Financial Insurance Consultants
Covington, LA

DMS SIG Liaison
Joe Sciuto
Army Research Institute
Alexandria, VA

Members-at-Large
Robert D. Lazenby
Dixie Beer Dist., Inc.
Louisville, KY

Robert Kayne
Gallaudet College
Washington, DC

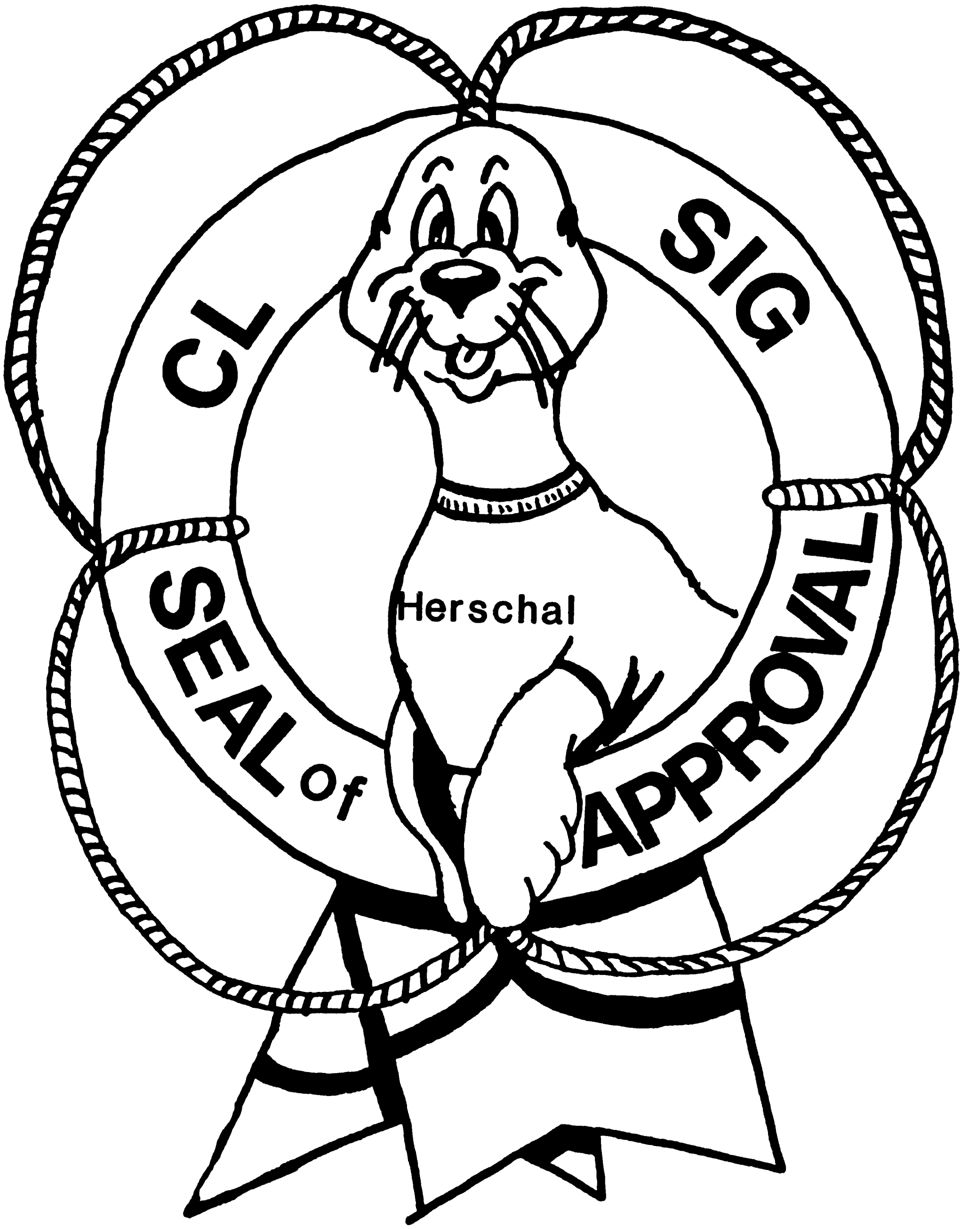
Ray Evanson
Paragon Data Systems
Winona, MN

Digital Counterparts
Sue Yarger
Digital Equipment Corporation
Merrimack, NH

Ray Arsenault
Digital Equipment Corporation
Merrimack, NH

SIC Mentor
Bill Brindley
Networks SIG Chair

SIC Review Committee
Larry Jasmann
Leslie Maltz
Ted Bear
Jeff Killeen


CL
SIG
Herschal
SEAL of APPROVAL

COBOL BASIC DIBOL RPG

COMMERCIAL LANGUAGES SIG

Chairman
Jim Wilson
Pfizer Inc. QC Div.
Terre Haute, IN

Symposium Coordinator
Ray Strackbein
Palm Desert, CA

Library Coordinator
Philip Hunt
System Industries
Milpitas, CA

Session Note Editor
Bob Van Keuren
Userware International, Inc.
Escondido, CA

Newsletter Editor
Ted A. Bear
Ramtek
Santa Clara, CA

Ass't Newsletter Editors
Beverly Welborne
LaPorte, IN

Kevin Cullen
VITA-Mix Corp.
Holmstead Falls, OH

Daniel Cook
Userware International, Inc.
Escondido, CA

Basic Working Group Members
Mark Hartman
Jadtec Computer Group
Orange, CA

Rocky Hayden
UserWare International Inc.
Escondido, CA

Bill Tabor
Computer Productss
Pompano Beach, FL
Ted A. Bear
Ramtek
Santa Clara, CA

Cobol Working Group Members
Keith Batzel
Crowe, Chizek & Co.
South Bend, IN

Mary Anne Feerick
RDBS Inc.
Kernersville, NC

Bill Leroy
The Software House, Inc.
Atlanta, GA

Herbert J. Matthews IV
ManTech International Corp.
Alexandria, VA

Jim Welborne
Crowe Chizek & Co.
South Bend, IN

DIBOL Working Group Members
Neil Baldridge
CompuShare
Lubbock, TX

Becky Burkes Ham
Financial Insurance Consultant
Covington, LA

Colin Chambers
Software Ireland Rep. Inc.
Portola Valley, CA

Mark Derrick
WAAY-TV
Huntsville, AL

Gary A.P. Kohls
Milwaukee, WI

Ken Lidster
Disc
Sacramento, CA

Kenneth M. Schilling
MCBA
Montrose, CA

Marty Schultz
Omtool Inc.
Tewksbury, MA

David L. Wyse
Professional Business Software
Dayton, OH

Marty Zergiebel
The Software Gallery
Brookfield, CT

RPG Working Group Members
Keith Batzel
Crowe Chizek & Co.
South Bend, IN

Ted Bear
Ramtek
Santa Clara, CA

Digital Counterparts
Tom Harris
Nashua, NH

Jim Totten
Nashua, NH

Joe Mulvey
Nashua, NH

Shirley Ann Stern
Nashua, NH

Standards Representatives
BASIC
Dan Esbensen
Touch Technologies, Inc.
Escondido, CA

COBOL
Bruce Gaarder
Macalester College
St. Paul, MN

DIBOL
Eli Szklanka
TEC
Newton, MA

DIBOL UPDATE

by

Kevin Cullen
VITA-MIX CORPORATION
8615 Usher Road
Cleveland, Ohio 44138

Many months ago there were rumors of a new DIBOL release and then for a long time we saw and heard nothing. In this case no news was not good news. Apparently delayed by many problems DIBOL version 3.0 finally reached field test status January 28, 1986. Most features promised were included although in most cases not fully implemented. Since that time it is obvious that much work has been occurring, both by the developers and those involved in the field test.

The second release of field test DIBOL V3.0 was released on March 26, 1986. Almost all known bugs appear to have been fixed and almost all features have been fully implemented. I can't go into specifics here and now, although I do hope to be able to do that in the future. There is, of course, no projected date as to when V3.0 will reach the Software Distribution Center. Judging from the length of time from the first release to the second it would be optimistic to expect anything before July. Additionally, there appears to be some substantial user-transparent changes still to be made. Perhaps these will address performance issues.

As a side point it is interesting to note that references to V3.0 as DIBOL-85 are still being made. Perhaps they should 86 that name!

Seriously, those of you who have longed for enhanced structure and greater compatibility with the rest of the VAX world will find this new version worth most of the wait.

DECUS

D A A R C

DAARC

Chairman
James Deck
Inland Steel Research Lab
East Chicago, IN

Symposium Coordinator
Mack Overton
FDA
Chicago, IL

Newsletter Editor
Ellen Reilly
William H. Rorer
Ft. Washington, PA

DEC Counterpart
Nancy Kilty
Digital Equipment Corporation
Marlboro, MA

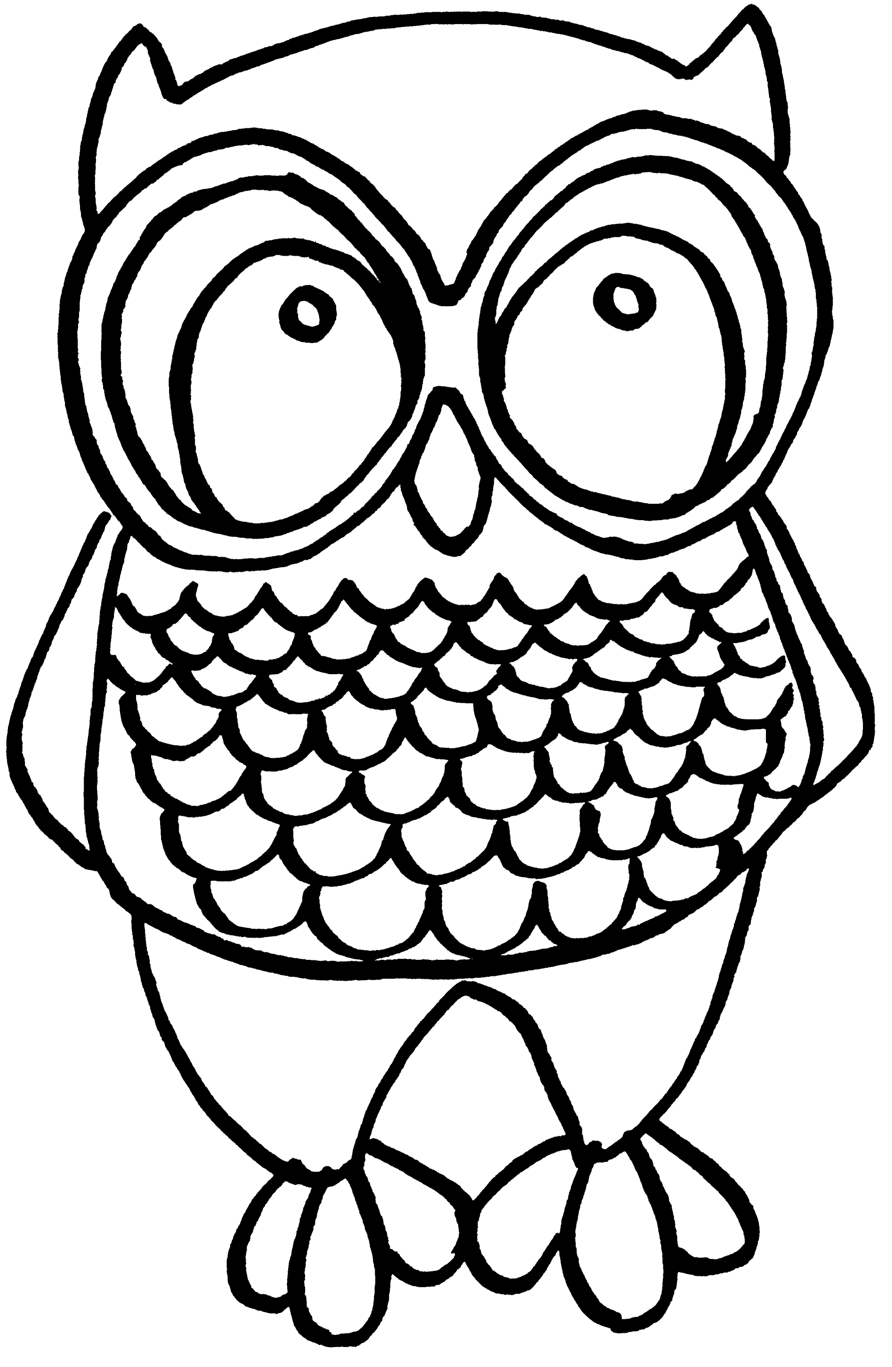
Hardware & Interfacing
Peter Clout
Los Alamos National Lab
Los Alamos, NM

Math Statistics & Analysis
Herbert J. Gould
C.C.F.A. University of Illinois Med Center
Chicago, IL

Process Control - Industrial Automation
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

RS-1
George Winkler
CPC International
Argo IL

DMS SIG



JOIN THE WISE

Working Group Coordinator/
Database Working Group:
Jim Perkins
PSC, Inc.
Shelburne, VT
(802) 863-8825

Forms Working Group:
Debbie Kennedy
Land Bank National DP Center
Omaha, NE
(402) 397-5040

Non-Digital Working Group:
Doug Dickey
GTE Government Systems
Rockville, MD
(301) 294-8400

RMS Working Group Coordinator:
Allen Jay Bennett
Lear Siegler Apistan
Grand Rapids, MI
(616) 451-6429

Pre-Symposium Seminar Coordinator/
Black Book:
David B. Turner
Korn/Kerry International
Los Angeles, CA
(408) 945-9600

ANSI Standards Coordinator:
Herman "Spence" Spencer
Army Research Institute
Alexandria, VA
(202) 294-9420

Member-At-Large:
Larry W. Hicks
Relational Database Services
Kernersville, NC
(919) 996-4882

Member-At-Large:
Richard Arndt
Cognos Corporation
Houston, TX
(713) 690-1105

AI SIC Liaison:
David Slater
Institute for Defense Analysis
Alexandria, VA
(703) 845-2200

SIG Chairman:
Joseph F. Sciuto
Army Research Institute
Alexandria, VA
(202) 274-9420

Comptroller:
Alan Schultz
Land Bank National DP Center
Omaha, NE
(402) 397-5040

Symposium Coordinator:
Keith Hare
JCC
Granville, OH
(614) 587-0157

Symposium Coordinator:
Barbara Mann
TRW
Redondo Beach, CA
(213) 532-2211

Communications Committee Representative/
Newsletter Editor:
J. G. Russell Poisson
SEED Software Corporation
Alexandria, VA
(800) 428-9400

Session Note Editor:
Mark Morgan
Farm Credit Banks
Springfield, MA
(413) 732-9721

Membership Coordinator:
VACANT

MIS Working Group Coordinator
Steve Pacheco
Ship Analytics (Past SIG Chairman):
North Stonington, CT
(203) 535-3092

MIS Working Group (Past SIG Chairman):
Sandy Krueger
Key Financial Systems, Inc.
Pine Brook, NJ
(201) 299-6600

Datatrieve Liaison:
John Schutt
J. R. Simplot Company
Boise, ID
(208) 336-2110

DEC Counterpart:
Wendy Herman
Digital Equipment Corporation
Nashua, NH
(603) 881-2494

Relational Database Services
Kernersville, NC
(919) 996-4882

Member-At-Large:
Richard Arndt
Cognos Corporation
Houston, TX
(713) 690-1105

AI SIC Liaison:
David Slater
Institute for Defense Analysis
Alexandria, VA
(703) 845-2200

Datatrieve Liaison:
John Schutt
J. R. Simplot Company
Boise, ID
(208) 336-2110

DEC Counterpart:
Wendy Herman
Digital Equipment Corporation
Nashua, NH
(603) 881-2494

Contributions

Contributions to the newsletter can be sent to either of the following addresses:

Editor, DMS SIG Newsletter
c/o DECUS U.S. Chapter
219 Boston Post Road, BP02
Marlboro, MA 01752

Russ Poisson
DMS SIG Newsletter Editor
SEED Software Corporation
2121 Eisenhower Avenue
Alexandria, VA 22314

Letters and articles for publication are requested from members of the SIG. They may include helpful hints, inquiries to other users, reports on SIG business, summaries of SPR's submitted to DEC, etc. Machine readable input is highly desirable. Submitters should keep in mind the DECUS policy on commercialism.

ONE MAN'S EXPERIENCE WITH RDB AND SMARTSTAR continued....

by

Tom Colati
System Analyst
Mobil Oil - Toxicology Div.
Princeton NJ 08540
(609) 737-5679

This is the second in a series of articles that I plan to submit to the newsletter outlining various things about the SMARTSTAR and RDB interface. These articles are NOT, repeat NOT, intended to be a sales pitch for SMARTSTAR.

It is intended as a means to pass on information that I have discovered and think is helpful about the SMARTSTAR/Rdb interface . This information was obtained (mostly through long hours and a lot of sweat) from reading the documentation, experimentation and conversations with DEC and STI.
I would very much like to be informed if anything I say is off base.
This article is geared for those who have SMARTSTAR and are familiar with its components.

In the last issue of this newsletter, I discussed the background of Smartstar, the backgound of STI, the vendor, an overview of the software, and the 4 different ways you can control a SMARTSTAR application, and lastly, transaction management. In this issue, I will share some general thoughts about the product and provide some examples.

Here are some general ideas and examples that I have come across. They are presented in no particular order and are not necessarily related.

All examples will be presented in BASIC (sorry boss) though they can be easily converted to COBOL or FORTRAN.

Read Only Queries and Updates

If you are doing read only queries to the database using the same ADF, I would suggest letting SMARTSTAR do all the work either by calling SQUERY from DCL or calling SQUERY from your program.

```
EX.
                 .
                 .
        call sc_appl('application')
        call sc_query(,)
                 .
                 .
```

If you are updating several relations at once you may want to write your own DAR to access REQUEST via OPAL. If you use REQUEST commands via OPAL you MUST make a call to OB_SWITCH to switch the database type from ADF to RDB.

```
   call ob_switch(1%,3%,)
```

This means that we will be accessing the database directly (via request) and not through the ADF

Views

There are 2 ways to join relations together. The first is to create a view in Rdb and map your ADF over the view. The second is to create the join in ADF itself. To do this you map table 1 of your ADF over the first relation and table 2 over the second relation. To join the 2 relations you must define a field that is common to both relations. This common field creates the link between the 2 relations. When joining 2 relations, I didn't find any difference between defining a view in Rdb and creating the join in SMARTSTAR. For joins of 3 or more relations , I would suggest defining a view in Rdb and mapping over the view in SMARTSTAR. I believe SMARTSTAR gets confused as to how to join the three tables together and sometimes attempts to join the records in all possible combinations. I have not looked into this situation enough other than to say that this is something to be aware of (by the next newsletter I hopefully will figure out what is going on).

SMARTSTAR issues a start_trans with a wait

When smartstar sends a command to the DSRI it is sent with a wait. Under normal circumstances this is ok, the query just waits for the relation or record to be free before executing the query. Well if a user or the DBA readies a relation exclusive and a user issues a query via SMARTSTAR , SMARTSTAR will sit and wait until doomsday for the relation to be free. It does not send a message back to the user (or to your program) that it is waiting for a relation to be free. The user just sees his cursor blinking on the screen. He does not know if the system crashed or he forgot to hit the append key (or whatever key he hit).

Indices

Smartstar has a slight problem dealing with indexes. For ADF's that are read only you do not have to define the index (via mode 7) in your ADF. For all other other data actions YOU MUST DEFINE THE INDEX IN THE ADF. If you do not, you will get the error message of

INDEX IS NOT UNIQUE

I feel that this is a bug, but STI keeps telling me that is the way things are suppose to work. Yes this definitly defeats one of the nice things about Rdb - to be able to add and delete indexes quickly. Well if you change an index or add an index you must make the changes in all the ADF's that access the relation. When defining your index in your ADF , Smartstar does not ask you what the name of the index is, it makes it's own name. The form of the index name is

```
relationname_keynumber
```

If the relation is named TEST and this is the first index for test defined in the ADF Smartstar will give the index a name of

```
test_0
```

The second index defined in your ADF will be

```
test_1
```

You are now restricted to using these index names when defining the index in Rdb.

Ob_switch and sc_open

Sc_open or sc_load automatically switches the database access type to ADF If you want to access the rdb database directly you must make you call to ob_switch AFTER you have opend all your ADF's

The relationship of OPAL/REQUEST/RDB

What actually happens after you make a request to the DAR or call REQUEST via OPAL?

Once OPAL has the request it then goes off and does its thing. Though you no longer have control over the query, it might be helpful to know what is happening. SMARTSTAR (actually OPAL) does not do any of the database operations, or management , that is completely done in Rdb. All that SMARTSTAR does is transform the query from the frontend syntax (whether it is from the ADF or your host program) to a DSRI command and than passes it to the DSRI. (I am a bit fuzzy on how and where the DSRI and RDB fit together so to the best of my knowledge what I am saying is true but it may not be.I may have to make amendments to this discussion after the spring decus - there is a DSRI seminar and I get a hold of a DSRI manual). I believe that the DSRI is a box that contains a set of rules and regulations that must be followed. Commands are passed to the DSRI and are checked to make sure that they conform to the rules, if so , the query is then passed on to the appropriate software. At this time, Rdb/vms and Rdb/elan is the only software that uses the DSRI, but I am told that other software will eventually hang of the DSRI as well (what that software is, I do not know). SMARTSTAR then waits for the results to be returned from Rdb.

From a performance statement, the time of a query is the same going from OPAL to DSRI as is it going from DTR to DSRI and RDO to DSRI (how the query gets to OPAL may affect the total time of the query). I have not run any benchmarks to support this statement (I plan to very soon though) but this is what I gathered from my talks with the Rdb people.

If you use a precomplier, the query will be slightly faster due to the fact that the DSRI query is complied once and left in memory in case you do the exact same query later. This does not happen in DTR,RDB or SMARTSTAR. I am told that the difference is very slight .

Tracing the call to DSRI

You can trap the actual DSRI query (or actually the calls to the DSRI) that SMARTSTAR passes to the DSRI by turning on the smartstar trace option.

You can turn on tracing either from your program or while directly in the ADF. To inact tracing from your program, make a call to ob_switch

```
  (in basic)

  call ob_switch(16%,1%,)
```

This will put a copy of the DSRI query in to a file. By default, the file is called OPALTRACE.LIS (you can redirect the output to other files using some of the other OPAL input/output calls).

From within the ADF you can turn on tracing by simultaneously hitting <PF1>~

Below is a portion of a trace of an append transaction

The first 3 calls are OPAL routine calls that indicate the type of query, the relation name and the field names

```
call ob_query (
     "APPEND "
)
call ob_query (
     "CRU_INFO
)
call ob_query (
     "("
)
call ob_query (
     "CRU_NO=CRU_INFO$CRU_NO,TRN_NUMBER=CRU_INFO$TRN_NUMBER,LOT_NO=
      CRU_INFO$LOT_NO,SAMPLE_ID=CRU_INFO$SAMPLE_ID,AMT_AVAILABLE=
         ( I cut out some of this information)
      CRU_INFO$EXP_DATE,DATE_MADE=CRU_INFO$DATE_MADE,HANDLING=
      CRU_INFO$HANDLING,TEST_ARTICLE=CRU_INFO$TEST_ARTICLE"
)
call ob_query (
     ")"
)
call ob_exec (
     <NULL>
)
```

A call to DSRI - issues a start trans - Since I don't have a DSRI manual I do not know what the parameters of the call are exactly. I think one specifies access mode and another whether wait or nowait

```
call rdb$start_transaction (
     0
     1 by value
     2386004
     4 by value
          01 09 02 06
)
exiting  rdb$start_transaction:
     2386073
```

The query is compiled

```
call rdb$compile_request (
     2386004
     output argument at 000a2cec
     522 by value
          04 0f 4a 08 43 52 55 5f 49 4e 46 4f         J CRU_INFO
          1a 02 01 15 0e 03 00 31 32 33 17 1a             123
          06 43 52 55 5f 4e 4f 01 15 0e 03 00        CRU_NO
          33 33 33 17 1a 0a 54 52 4e 5f 4e 55       333   TRN_NU

          ( I cut out some of the dump info)

          41 4e 44 4c 49 4e 47 01 15 0e 01 00       ANDLING
          20 17 1a 0c 54 45 53 54 5f 41 52 54           TEST_ART
          49 43 4c 45 fff4c                         ICLE L
)
exiting  rdb$compile_request:
     2386004
     2386093
```

Here I think the compiled request is passed to the DSRI

```
call rdb$start_request (
     2386093
     2386073
     0 by value
)
call rdb$release_request (
     2386093
)
exiting  rdb$release_request:
     0
```

! A commit is issued

```
call rdb$commit_transaction (
     2386073
)
exiting  rdb$commit_transaction:
     0
)
```

The detach to the database is made because I exited the ADF

```
call rdb$detach_database (
     2386004
)
exiting  rdb$detach_database:
     0
```

Here is a portion of a trace of an operation that failed a constraint

```
call ob_query (
     "APPEND "
)
call ob_query (
     "CRU_TRANS                            "
)
call ob_query (
     "("
)
call ob_query (
     "TRANSACTION=CRU_TRANS$TRANSACTION,cru_no=CRU_TRANS$cru_no,DISPENSE_NO=
      CRU_TRANS$DISPENSE_NO,TRANS_DATE=CRU_TRANS$TRANS_DATE,AMOUNT=
      CRU_TRANS$AMOUNT,TRANS_STORE_DATE=CRU_TRANS$TRANS_STORE_DATE,
      AUDIT_TRAIL_TRANS=CRU_TRANS$AUDIT_TRAIL_TRANS"
)
call ob_query (
     ")"
)
call ob_exec (
     <NULL>
)
call rdb$start_transaction (
     0
     1 by value
     2375212
     4 by value
          01 09 02 06
)
exiting  rdb$start_transaction:
     2375281
call rdb$compile_request (
     2375212
     output argument at 000a2cec
     175 by value
          04 0f 4a 09 43 52 55 5f 54 52 41 4e          J CRU_TRAN
          53 1a 02 01 15 0e 02 00 52 43 17 1a       S        RC
          41 55 44 49 54 5f 54 52 41 49 4c 5f       AUDIT_TRAIL_
          54 52 41 4e 53 fff4c                      TRANS L
)
exiting  rdb$compile_request:
     2375212
     2375301
call rdb$start_request (
     2375301
     2375281
     0 by value
)
```

```
RDB message vector:
013880b2 00000001 00261470
constraint CRU_INFO_NO_CHECK failed

call rdb$release_request (
     2375301
)
exiting  rdb$release_request:
     0
call rdb$rollback_transaction (
     2375281
)
exiting  rdb$rollback_transaction:
     0
exiting  ob_exec:
DSRI start request fails
constraint CRU_INFO_NO_CHECK failed

call rdb$detach_database (
     2375212
)
exiting  rdb$detach_database:
     0
```

In this example the transaction was rolledback. OPAL then analyzes and formats Rdb message vector for screen display. I will show how to further analyze this message vector and trap for specific Rdb errors. This trace file is a nice way to see exactly what is going on in the backend.

Querying the Database not using the DAR

Here is an example of doing a query from your program and not using the DAR. Some fields in our database are abbreviations (we abbreviate our names of divisions - tox for toxicology) and at times our users might want know the full name of the abbreviation. I define a KAR that calls this program. (Note adf-1 and adf-2 are loaded in the main program)

```
10   function integer full_name(in%,out%)

     external integer function sc_obact2
     declare long rev
     declare long constant sc__data_enter

    ! The abbreviation is retrieved from the ADF that is on the screen

     call sc_retfld('abbrev',abbrev$)

   ! adf-2 is called with nodisplay . adf-2 should map over the
   ! relation or file that contains the abbreviation - full name data

    call sc_appl('adf-2', 5%)

   ! the abbreviation to be retrieved is put into the proper field

     call sc_putfld('abbrev',abbrev$)

   ! A call to the function to do the required operation - in this
   ! case a retrieve - the proper dataaction is passed
```

```
     call sc_obact2(rev,sc__data_find)

   !  the full name of the abbreviation is pulled from adf-2

     call sc_retfld('full-name',variable$)

   ! the original application is called (to retain current context)

     call sc_appl('adf-1', 5%)

  ! the full name is displayed to the user

     call sc_msg(" the full name is " + variable$)
     out%=0%
    end
```

The function sc_obact2 is nothing more than a copy of the function sc_obact (which is often used as the DAR) and renamed. This means that you are using the same function that is enacted by the DAR except that you are calling it from your program. This function is expecting 2 arguments - the first is an unused long word. The second is the type of data action you want performed. A list of all data action codes can be found in the SMARTQUERY manual A.14 You must define the data action codes as external long constants

Background on OPAL

OPAL is the backend interface to REQUEST. It is a means to execute REQUEST commands from your program

OPAL maintains a query buffer. This query buffer holds the request text commands for execution

There are 3 basic commands that you need to be familiar with

OB_QUERY - appends a character string to the query buffer

```
call ob_query('select table' )
```

This puts the request command SELECT TABLE in the query buffer - note the command is NOT execute by merely putting the command in the query buffer. You may break up your REQUEST command into multiple calls to ob_query

```
call ob_query('select')
call ob_query('table')
```

This will produce the same results as performing in one command

OB_GO   - causes all the commands in the query buffer to be executed

```
call ob_query('select table')
call ob_query('fields_of table')
call ob_go()
```

In this example ob_go will cause both commands in the query buffer to be executed

OB_EXEC - Causes the next command in the query buffer to be executed

```
call ob_query(’select table’)
call ob_query(’fields_of table’)
call ob_exec()
```

the call to ob_exec will cause the command SELECT TABLE to be executed. the request command FIELDS_OF TABLE is not executed and remains in the query buffer
until the next call to ob_exec or ob_go

Testing the status of request calls

You can test the status of your request commands by evaluating the status code
returned from ob_exec

```
call ob_query(’select table’)
call ob_query(’fields_of table’)
st=ob_exec()
if st <>ob_normal
   then
      call sc_msg("error")
end
```

The call to ob_exec will cause the SELECT TABLE command to execute - if the REQUEST command executes successfully than ob_normal is returned from ob_exec. The variable you declare to receive the status code should be define as a longword and ob_normal should be defined as an external long constant.

By calling sc_obmsg you can see why the REQUEST command fails.

```
call ob_query(’select table’)
call ob_query(’fields_of table’)
st=ob_exec()
if st <>ob_normal
   then
      call sc_obmsg(ST)
end
```

Note, this checking of the status of the request command just checks the status of the command, it does not trap for syntax errors. Here we misspell the word select. This will cause a syntax error and will cause a fatal error back to your program. (hopefully your program will handle these types of errors). The if statement will not be evaluated since your program will exit (or goto to your error handling routine)

```
call ob_query(’selct table’)
call ob_query(’fields_of table’)
st=ob_exec()
if st <>ob_normal
   then
      call sc_obmsg(ST)
end
```

Difference between ob_go and ob_exec.

You should be aware of the differences between ob_go and ob_exec.

```
call ob_query (’select table’)
call ob_query(’fields_of table’)
st=ob_go()
if st <>ob_normal
   then
      call sc_obmsg(ST)
end
```

This may not necessarily produce the results you think. Since ob_go executes all commands in the query buffer, the status returned to ST will be for the status of the last command in the query buffer (in effect issues several ob_exec commands) . All commands in the query buffer will be executed regardless of any errors that occur. For instance, if a nonexistent table is specified in the select statement ob_go will return a status of ob_normal, since the command FIELDS_OF TABLE will be executed successfully (it just returns no fields found). This is probably not the desired result. You should be careful when using ob_go and checking the status of commands.

Another difference between ob_go and ob_exec is the results of the command being executed. ob_go will display the results of the command (if there are any) to the screen while ob_exec will not.

```
call ob_query(’select test’)
call ob_query(’fields_of test’)
call ob_go()
```

This code will produce the following output to the screen

```
|column|field                          |type           |size  |scale |
|--------------------------------------------------------------------|
|     1|TEST                           |text           |    10|     0|
|--------------------------------------------------------------------|
1 fields found
```

The following code will not display output to the screen (note 2 calls to ob_exec are needed to execute both commands) The command is executed, it doesn’t return the information of the command to the screen. This is useful when you are retrieving a record and you want information from the record to go directly into your host variables.

```
call ob_query(’select test’)
call ob_query(’fields_of test’)
call ob_exec()
call ob_exec()
```

I stated that ob_exec will execute the next command in the query buffer. When ob_exec executes a retrieve operation ob_exec only initiates the retrieve, it DOES NOT get the record. You must make a call to ob_get.

```
              call ob_query('select table')
              call ob_go()
              call ob_query('retrieve')
              call ob_exec()
```

This will not retrieve any records in the table The following code will retrieve the first record in the table.

```
              call ob_query('select table')
              call ob_go()
              call ob_query('retrieve')
              call ob_exec()
              call ob_get()
```

In the case of a retrieve, if you want to test the status of the retrieve command you must check the status code being returned from ob_get not ob_exec since ob_get is doing the actual fetch. Here we are testing to see if there are any records in the table. We are testing the status of ob_get (one of the possible return status codes is ob_endtab.)

```
              call ob_query('select table')
              call ob_go()
              call ob_query('retrieve')
              call ob_exec()
              st=ob_get()
              if st=ob_endtab
                    then
                      call sc_msg( 'record not found')
              end if
```

Trapping status codes and error messages -

All OPAL routines return a status code. I believe it is a 4 byte status code with the low order bit determining success or failure . Generally, to test for success or failure you want to test if ob_normal is returned. As I have shown previously you trap the return status code in a local variable then evaluate the variable.

```
             st=ob_get
               if st <> ob_normal
                    then
                     ......
```

ob_normal - should be declared as an external long constant. The variable trapping the status code (in this case ST) should be declared as a longword. The function should be declared (OB_GET) as external integer function

This is an acceptable technique if all you want to do is test for success or failure . Sometimes, if a command does not return ob_normal, I need to further analyze the status code to determine what the failure is and then take appropriate action. For example, if I issued an append command and it failed, I want to check if it failed due to a constraint fail or a duplicate index fail, in which case I would give the user the opportunity to change the record being stored. I would consider all other error's ,fatal errors and stop further processing.

When you are in SQUERY and you issue DAR request (like append) and it fails for some reason, the reason is displayed on the bottom of the screen

```
           Ex
              DSRI start fails
              append would cause duplicates on unique index name
```

Well I needed to get at that information somehow and analyze it. Smartstar does not do anything magical to obtain this information, all it is doing is interpreting the RDB MESSAGE VECTOR. The Rdb message vector is a series of 4 longwords that you can interpret directly if you wish. I tried this and found it to be a pain in the neck. There is a better way to analyze the Rdb message vector. You can use the OPAL call OPAL_GETMSG to format the Rdb message vector into a character buffer (or host program variable) , then you search the character buffer for the desired error code.

Here is an example of what is in the Rdb message vector. These are hex codes for RDB mnemonic error codes. You can find the corresponding numeric value than look up the numeric value to find the Rdb mnemonic code

```
                013880b2 00000001 00261470
```

That is far to much work to be doing when there is an easier way. The call OPAL_GETMSG does exactly this.

```
                call ob_query('select table')
                call ob_go()
                call ob_query( 'append (NAME="John Doe")')
                st=ob_exec()
                call opal_getmsg(st,,,,result,dummy)
```

This code will take the Rdb message vector and format it into a character string and store it in the variable RESULT. You can now search RESULT for the Rdb mnemonic desired. You do not have to specify the 2nd, 3rd or 4th parameters when calling opal_getmsg. You must declare the variable dummy (or whatever you call it) as a longword and you must define the variable that receives the text, in this case, result a length of 300 (in basic you must use a map statement or an error will occur).

For example. If an append fails I want to determine if it failed due to a constraint fail or an index fail

```
declare long  st,dummy
external long constant ob_normal
map (a)  string result=300
.           .              .
.           .              .
.           .              .
call ob_query('select table')
call ob_go()
call ob_query( 'append (NAME="John Doe")')
st=ob_exec()
         if st <> ob_normal
           then
              call opal_getmsg(St,,,,RESULT,dummy)

              st%=pos(RESULT,"no_dup",0%)

                 if st% = 0%
                    then
    call sc_msg(" you can not store a duplicate record ")
                    else
                     goto 10
                   end
```

First a check is made to determine if the append is successful. If the append is not successful than we want to convert the status code into a character string that can be searched. I call opal_getmsg to format the error code into the variable RESULT. I then search RESULT for the RDB mnemonic NO_DUP (NO_DUP is the mnemonic for trying to store a duplicate record) If I find NO_DUP in the string, I inform the user and give them a chance to change some information. If some other error occurs I display the error message and take appropriate action. A partial list of RDB mnemonic can be found in the VAX Rdb/VMS Guide to Programming manual table 6-1. The mnemonic formatted by opal_getmsg will not be in the exact form as listed in the table. For instances, under the statement modify a possible error mnemonic might be RDB$_INTEG_FAIL

The format of the string from opal_getmsg might be

```
%RDB-E-NO_DUP, Duplicate record encountered
```

so you want to search for part of the mnemonic found after the RDB$_ portion (in this case INTEG_FAIL)

DAR and Data actiion failure

When you are in SQUERY and you attempt an append and for some reason it fails (dups or constraint), an error message will be displayed and you will be returned to squery. Well being the knowledgable user that you are, you then go and change the field that is causing the duplicate record and then hit the append key again ,well, SMARTSTAR will just sit there. The fail of the append confuses SQUERY (it seems to think that you are trying to modify the record and keeps returning a "no fields were changed" message). The only way to correct this situation is to hit the CLRFRM key. If you have an ADF with of 20 or 30 fields, this is a pain to have to reenter all those fields . For some reason, when an append fails some internal flags (which I will not go into) are not being reset. These flags are reset when you hit the CLRFRM key. There is a fix, you must modify the append portion of the DAR SC_OBACT which is found in SMARTSTAR:SCOBACT.BAS (also a cobol and fortran version)

Replace..........

```
      case sc__data_enter
          call sc_rettbl (tableno, bufer (tableno))
          result = sc_obapp (tableno)
          if (result and 1%) then
              call sc_clrfrm
              call sc_obflg (1,,0)
              call sc_msg (' rECORD aPPENDED ')
          end if
```

with...........

```
      case sc__data_enter
          call sc_rettbl (tableno, bufer (tableno))
          result = sc_obapp (tableno)
          if (result and 1%) then
              call sc_clrfrm
              call sc_obflg (1,,0)
              call sc_msg (' record apPENDED ')
          else
             sc_obact=sc__app_err
          END IF
```

(note - sc__app_err must declared as an external long constant)

Once you make this change you must recompile the function and replace it in the SMARTSTAR library of SHARE.OPT.

Remember you must make this alteration after each smartstar software update since a new share.opt will be included .

I mentioned earlier in this article that I use REQUEST calls to append data into the datafile without having to allocate buffer space in my program. Here is an example how I do it. The code is generic and will work on any ADF that has RDB as a backend. The syntax for an append statement in REQUEST is

```
select table
append (target_list)
```

where the targetlist is in the form

```
(field1="data1", field2="data2" , .... fieldN="dataN")
```

The problem is how do I find all the field names, then find the data in those fields, then pipe it into my append statement. Well one way is to issue sc_retfld statements to put the data into a variable in your program .

```
sc_retfld(1%,var1$)
sc_retfld(2%,var2$)
sc_retfld(3%,var3$)

append (field1=var1$, field2=var2$ , field3=var3$)
```

The problem with this is if you happen to add or delete a field in your ADF you now have to make a hard code change in your program. Well in this era of 4GL we are suppose to be able to quickly change our ADF and not have to change our program. There is an OPAL call that does exactly this, it is called SC_OBWHRET. SC_OBWHRET creates a string based on all of the current qualifications that the user has entered into the ADF via the qualify key. This is close to what we want except we want all the information the user enters into the screen not just the qualiifers. If we call SC_SELECT prior to SC_OBWHRET we will accomplish this. SC_SELECT will make qualifiers out of all fields that have data in them regardless of how they are entered.

```
   call sc_select
   call sc_tblnam(tableno%,tablename%)
   call sc_obwhret(tableno%,where)
   call ob_query('select')
   call ob_query(tablename$)
   call ob_query('append')
   call ob_query(where)
   call ob_go()
```

So it looks like we have accomplished what we want. Well not quite SC_OBWHRET returns the information in the form of

```
   tablename.fieldname="data"
```

the syntax of the target list is

```
   fieldname="data"
```

so we must somehow search this returned string and strip out the tablename. I have written a little basic program that will strip out the table name from the string.

```
10         function string extract (table%)

           declare double clause

            call sc_select
            call sc_obwhret(table%,clause)
            call str$copy_dx(clause$,clause)
            call sc_tblnam(table%,table$)
            ! Construct the search string

                table$= "(("+table$+"."

             ! find length of search string

               length%=len(table$)

               ! Find the length of the clause to be searched

20                     clen%=len(clause$)

           ! find the starting position of )) in the clause

             start1%=pos(clause$,"))",0%)
              ! the position of the next occurence of the (( after the
              ! )) is found
```

```
               end1%=pos(clause$, "((",start1%)

               ! A check to determine if "))" ere found at the end of the
string
               !

             if end1%=0
                 then
                    goto 101
               end if

            ! the character string from  the (( to the )) is extracted

             string1$= seg$(clause$,start1%,end1% + length% - 1)

             ! all blanks are removed

             string2$= edit$(string1$,2 + 32)

             !

             !  A check to see if the required string exits

             if (string2$=edit$("))AND" + table$ ,32))
                then

             ! the word 'And' is replaced with a , Also the leading ((
             ! and trailing )) and table name is deleted

             clause$=seg$(clause$,0%,start1% - 1) + " , " + seg$(clause$,end1%
+length%,CLEN%)
               goto 20
                 end if

101

           ! find length of clause

             clen%=len(clause$)

             ! find the position of the )) at the end of the clause

             start1%=pos(clause$,"))",0%)

             start2%=pos(clause$,table$,0%)

             ! The trailing )) and leading (( and table name are extracted

           clause$=seg$(clause$,start2% + length%,start1% - 1)
            extract=clause$

200        end function
```

Now if we use the following code we can get the information from the screen and pipe it right into our append statement and our program does not have to allocate any space for the fields nor does it have to know how many fields there are. If we add or change the ADF, no alterations to our program are needed.

```
   call sc_select
   call sc_tblnam(tableno%,tablename%)
   where$= extract(tableno%)
   call ob_query('select')
   call ob_query(tablename$)
   call ob_query('append')
   call ob_query(where$)
   call ob_go()
```

There is one more slight problem with SC_OBWHRET and that is it does not work on ADF's with multiple tables. For all tables other than the first table SC_OBWHRET will not include those fields that appear in multiple tables.

This certainly presents a problem if I want to use this technique on ADF's with multiple tables .

I have written the following function that is in reality my own version of SC_OBWHRET. It corrects the problem with the multiple tables and it returns a string in the form of fieldname="data" so there is no need to call EXTRACT to strip out the tablename. This function works ONLY on ADF's using RDB. I am pretty sure it won't work on RMS files because RDB and RMS handle dates a bit differently. An additional restriction is that the ADF CAN NOT have a date field that uses a format which includes hours, minutes or seconds or hundreths. All other date formats seem to work. (I ran into some unexpected error when hours and minutes are included. I hopefully will have a fix in the next submission). You must pass the field number that is to be extracted to the function . This function will create a string of only those fields that have data in them. If you wish all fields in the table , then you must modify the code to eliminate the check that checks to see if the field is empty.

```
10      function string target_list (T%)


       external long constant ss$_normal,     &
                              sc_k_ftype_a,   &
                              sc_k_ftype_n,   &
                              sc_k_ftype_d

       external integer function sc_fldnam

        declare word     XTA(9),  &
                         auxlen


        declare long type1,   &
                     result,  &
                     digits,  &
                     scale1
```

```
      ! These fields must be define dynamically

        map (c) item1$ =8,integer fldno, more

       ! sets default time to current system time

       call sys$numtim(xta,)
        ! get the name of the table

      call sc_tblnam(T%,tablename$)

      ! Counters

      A%=99
      more= 1

      ! The screen used to translate the date template and strip out the
      ! formatting characters
      ! note N and n have been replaced with M and m

      translate$=  string$(65%,0%) +  "A CDEFGHIJKLLNOPQRSTUVWXYZ      a
cdefghijkllnopqrstuvwxyz"



         fldnam$=" "
         fldno=1%

        ! The beginning of the loop to get all the fields in a the table

        for fldno=1% while more=1
          ! reset some variables to insure they are empty

         fldnam$=" "
         item$=" "
         item1$="        "
        ! get the field name for the current field #
         result=sc_fldnam(tablename$,fldno,fldnam$)

         if result=ss$_normal
            then
             ! Get the data type, screen format of the field and the
             ! the scale (if it is a numeric

               call sc_fldfmt(fldno,template$,,,,,type1,scale1,,)

              ! get the data in the field

               call sc_retfld(fldno,item$)

              select type1
                 ! a field type of alpha
                   case sc_k_ftype_a
                 ! if the field is empty we don't want it
                     if item$<>" "
                        then
```

```
                         ! add the fieldnam and data to the current string

                            if string1$=" "
                               then
                                 string1$=fldnam$ + '="' + item$ + '"'

                               else
                                  string1$= string1$ + ', ' + fldnam$ + '="' +
item$ + '"'

                              end if
                         end if

                 ! for numeric datatypes

                   case sc_k_ftype_n
                     call sc_atrfld(tablename$,fldno,,,,digits)
                     zero$=string$(digits,48% )

                  ! if the field is zero we don't want it

                     if item$<> zero$
                        then
                 !
                 if scale1 = 0%
                    then
                       item$=item$
                    else
                 ! The decimal must be put in for thos that have decimals

                       leng%=len(item$)
                       item$= seg$(item$,0%, leng% + scale1 )  + "." +
seg$(item$,leng% + scale1 + 1,leng%)
                     end if

                  ! the fieldname and data are add to the string

                            if string1$=" "
                               then
                                 string1$=fldnam$ + '="' + item$ + '"'

                               else
                                  string1$= string1$ + ', ' + fldnam$ + '="' +
item$ + '"'

                              end if

                         end if
                ! for date datatypes
                   case sc_k_ftype_d
                   if item$ <>" "
                     then
                      ! eliminate the formatting characters in the data
                      ! format
                       template$=xlate(template$,translate$)

                      ! The date must be translated into the correct format
                      ! expected by request which is yyyynndd
                      ! The input date is changed to an XTA date form
```

```
                       call  char_xta(template$,item$,XTA,aux$,auxlen)
                      ! The XTA date form is changed back to a character
string
                      ! in the correct form

                       call xta_char(xta,"YYYYNNDD",item1$)

                      ! The fieldname and date are added to the string
                            if string1$=" "
                               then
                                 string1$=fldnam$ + '="' + item1$ + '"'

                               else
                                  string1$= string1$ + ', ' + fldnam$ + '="' +
item1$ + '"'
                              end if
                       end if
                end select
           else
              more=0
        end if
       next fldno
        The string is returned

        target_list=string1$

1000       end function
```

With this function we can now construct a generic append statement that will append all the table's in the ADF to the backend database using REQUEST via OPAL (not going through the ADF to do the store). This is an example of how you might write the append portion of your DAR .

```
        !************************************************************
        !        A P P E N D   R E C O R D
        !************************************************************
        case sc__data_enter
        call ob_switch(1%,3%,)
        result=sc_tblnam(tableno%,tablename%)
        while result=ss$_normal
        where$= target_list(tableno%)
        call ob_query('select')
        call ob_query(tablename$)
        call ob_query('append')
        call ob_query(where$)
        call ob_go()
        tableno% = tableno% + 1
        result= sc_tblnam(tableno%,tablename%)
        next
```

Target_list returns a string that is suitable for input into any request statement that needs a target list (The above block of code will work for a modify by adding the where clause). The string returned by SC_OBWHRET is suitable for input into the where clause (remembering the bug with SC_OBWHRET)

```
        call sc_select
        call obwhret(1%,clause
        call ob_query('select')
        call ob_query(tablename$)
        call ob_query('retrieve where')
        call ob_query(clause)
        call ob_go()
```

Multiple transaction rollback

You MUST use REQUEST via OPAL calls in order to take advantage of multiple transaction rollback. This is NOT documented in any manuals. It is not an officially released feature in this version but it does exist in his version. I am told that it will be part of the next release (in late fall) . It should be assumed that this is a beta feature for this release and you should use it at your own risk (This is what I was told by STI). I have used this option and so far have not found any problems.

There are 3 additional requests commands which are used in multiple transaction rollback

BEGIN - is used whenever multiple database commands are to be treated as a single transaction. It signals the Start of the a group of commands that are to be treated together

END - Completes a BEGIN block - Signals that a commit should be given

ABORT - Signals a rollback should be issued - All commands back to the last begin will be rolledback.

Note you can not have nested begin blocks

```
      call sc_tblnam(tableno%,tablename%)
      where$= target_list(tableno%)
      call ob_query('select')
      call ob_query(tablename$)
      call ob_query('begin')
      call ob_go()
      call ob_query('append')
      call ob_query(where$)
      result=ob_exec()
      if result=ob_normal
         then
            call ob_query('end')
            call ob_go()
          else
            call ob_query('abort')
            call ob_go()
         end if
```

This is a trivial example of how the begin- end/abort works. If the append transaction fails than we want the entire transaction to rollback (This is trivial since a single transaction will automatically rollback on a fail). The command BEGIN is put into the query buffer and ob_go is called - this is so the BEGIN command is executed. The request command is then formulated and executed by ob_exec. The status of ob_exec is checked. If the command is successful then we issue an END command in the buffer otherwise we issue an ABORT in the buffer an and execute it.

Now we can put some of these idea's together. Here is one of the functions we use as a DAR. It only allows appending records.

```
100     function integer  trans_lg (long rev,long action)


         external integer function ob_go,    &
                                  ob_exec,  &
                                  ob_get,   &
                                  ob_reset


         external string function target_list

         external long constant    ob_normal,          &
                                  sc__data_enter,     &
                                  ob_endtab,          &
                                  sc__data_clear

         declare long  result

        ! checks the action requested

        select action
           case  sc__data_clear
                    call sc_obclr
                    goto 1000
           case  sc__data_enter
           case else
                    CALL SC_MSG(" PLEASE PRESS THE ENTER KEY TO COMPLETE THIS
TRANSACTION  ")
                    goto 1000
         end select


        call sc_msg(" Transaction update in progress ")


          clause2$=target_list(2%)
          call sc_tblnam(2%,table2$)
          clause1$=target_list(1%)
          call sc_tblnam(1%,table1$)
         ! start the begin block

          call ob_query('begin')
          call ob_go()
         ! send the first request command to the buffer
```

```
          call ob_query('select')
          call ob_query(table1$)
          CALL OB_GO()
          call ob_query('append  (')
          call ob_query(clause1$)
          call ob_query(')')

           ! execute the request command and check the status
           ! if it is successful then allow the next request statement
           ! otherwise issue a rollback

                 result=ob_exec()
                     if result <> ob_normal
                       then
                         goto  800
                       end if

          ! send the next request to the buffer

          call ob_query(' select ')
          call ob_query(table2$)
          call ob_go()
          call ob_query('append (')
          call ob_query(clause2$)
          call ob_query(')')

           ! execute the statement - check the status

            result=ob_exec()
                    if result <> ob_normal
                      then
                        goto  800
                      end if


              ! if both commands are successful then issue an end
              ! to force a commit

                         call ob_query('end')
                        call ob_go()
                        call sc_clrfrm
                       call sc_msg(" Transaction complete ")
                       goto 1000

800
          ! if either request fials - display the reason

                      CALL sc_obmsg(result)
                        sleep 3%
                        call sc_obmsg(result)
          ! clear the query buffer
                       call ob_reset()
           ! issue an abort and rollback transaction

                        call ob_query('abort')
                        call sc_msg(" Transaction has been rolled
back ")

                        call ob_go()
```

```
1000                                end function
```

I have tried to pass on some things that I have come across and hope that they will be helpful to others. My intent was not to present an overview of the product but present some deeper idea's as to what is going on. I am sure there are other ways to do some of the same type of things that I have shown and I would appreciate any other idea's anybody else has. In the next submission I will talk about BINDING - what it is , how to get around it and how to make it as generic as possible in your program. I also did not deal with performance, since I have not quite gotten that far in my application yet, but hopefully by next time I will .

Feel free to give me a call if you have any questions about anything in this article or want to talk in general - I AM ALWAYS HAPPY TO TAKE A BREAK FROM BANGING ON MY TERMINALS KEYS.

TOM

The Wombat

EXAMINER & 4GL Dispatch

"Increases the Circulation of Anyone in America" *Volume 7* *Number 10*

FAMILY ALBUM
Photo Album

DTR/4GL Special Interest Group - Officers

3-26-86

SIG Chairman	Joe H. Gallagher	Research Medical Center	2316 East Meyer Blvd.	Kansas City, MO 64132	816-276-4235
Past SIG Chairmen	Larry Jasmann	U. S. Coast Guard	10067 Marshall Pond Rd.	Burke, VA 22015	202-426-2344
Communications					
Committee Rep.	Elaine McWilliams	Boeing Company	P. O. Box 24346 M/S 9c55	Seattle, WA 98124	206-773-0102
Editor	Donald Stern	Warner Lambert Company	10 Webster Road	Milford, CT 06460	203-878-9351
Production Editor	Steve Cordiviola	Kentucky Geological Survey	311 Breckinridge Hall	Lexington, KY 40506	606-257-5863
Artist	Bart Z. Lederman		2572 E. 22nd St.	Brooklyn, NY 11235	718-743-9593
Symposia					
Symposia Committee Rep.	Chris Wool	E.I. duPont Eng. Dept.	Louviers Bldg.	Wilmington, DE 19898	302-366-4610
Symposia Assitant	Diane Pinney	Computer Sciences Corp.	443 Inyokern Road	Ridgecrest, CA 93555	619-446-6585x284
Pre-Symposia Seminars	Dana Schwartz		15719 Millbrook Lane	Laurel, MD 20707	301-859-6277
Symposium Notes	Elaine McWilliams	Boeing Company	P.O. Box 24346 M/S 9c55	Seattle, WA 98124	206-773-0102
Product Development					
Development Coordinator	Bart Z. Lederman		2572 E. 22nd St.	Brooklyn, NY 11235	718-743-9593
New Product Focal Point	Diane Pinney	Computer Science Corp.	443 Inyokern Road	Ridgecrest, CA 93555	619-446-6585x284
DEC Counterparts					
Datatrieve	Andy Schneider	Digital Equipment Corp.	110 SpitBrook ZK02-2/N59	Naushua, NH 03062	
TEAMDATA, RALLY	Basil Harris	Digital Equipment Corp.	110 SpitBrook ZK02-2/N59	Naushua, NH 03062	
Library Committee Rep.	Bart Z. Lederman		2572 E. 22nd St.	Brooklyn, NY 11235	718-743-9593
Volunteer Coordinator	Larry Jasmann	U. S. Coast Guard	10067 Marshall Pond Rd.	Burke, VA 22015	202-426-2344
Campground	Bert Roseberry	U. S. Coast Guard	500 Camp Street	New Orleans, LA 70130	504-589-4934
Special Effects	Alex L. Lamb	U. S. Army	32 Pearce Ave	Eatontown, NJ 07724	201-532-3318x1843
DMS SIG Liason	Phil Naecker	J M Montgomery Engineers	250 N. Madison Ave	Pasadena, CA 91101	818-796-9141

DTR/4GL Masters List

2-26-86

RSX,P/OS,VMS	Bart Z. Lederman		2572 E. 22nd Street	Brooklyn, NY 11235	718-743-9593
PRO,VMS	Joe H. Gallagher	4GL Solutions	10308 Metcalf Suite 109	Overland Park, KS 66212	913-894-9550
VMS	Joe Kelly	WymAn Gordon Co.	Worchester Road	North Grafton, MA 01536	617-839-4411x5480
VMS	Larry Jasmann	U. S. Coast Guard	10067 Marshall Pond Rd.	Burke, VA 20015	202-426-2344
VMS	James Swanger	G. D. Searle & Co.	4901 Searle Parkway	Skokie, IL 60077	312-982-7430
VMS	Chris Wool	E. I. duPont, Engineering Dept.	Louviers Bldg.	Wilmington, DE 19898	302-366-4610
VMS	Dick Azzi	Motorola SPS	Box 2953 MS: P116	Phoenix, AZ 85062	602-244-4316
VMS	Chris Heinz	American Board of Family Practice	2228 Younge Drive	Lexington, KY 40505	606-269-5626
VMS	Bert Roseberry	U. S. Coast Guard	500 Camp Street	New Orleans, LA 70130	504-589-4934

Contributions

Contributions for the newsletter can be sent to either of the following addresses:

Editor, DATATRIEVE Newsletter
DECUS U.S. Chapter
219 Boston Post Road, BP02
Marlboro, MA 01752

Donald E. Stern, Jr.
Warner Lambert Company
10 Webster Road
Milford, CT 06460

Letters and articles for publication are requested from members of the SIG. They may include helpful hints, inquiries to other users, reports on SIG business, summaries of SPRs submitted to Digital or other information for members of the DATATRIEVE SIG. Machine readable input is highly desirable and machine-to-machine transfer of material is preferred, but most anything legible will be considered. However, this newsletter is not a forum for job and/or head hunting, nor is commercialism appropriate.

Table of Contents

DECUS U. S. Chapter
SIGs Newsletter, Volume 1, Number 10, June 1986
Wombat Examiner and 4GL Dispatch, Volume 7, Number 10

1 Chairman's Corner
2 From the Editor's Pen
3 Ask the Wombat Wizard
7 Product Improvement Request System
18 Index to Wombat Examiners
31 About the Cover

Chairman's Corner

Joe H. Gallagher, 4GL Solutions, Overland Park, KS

At this writing, it is still in the middle of a gloriously beautiful spring in Kansas City and before the national creative accounting contest which is held each year on April 15th. You will be reading your June issue of the newsletter after the 25th Anniversary activities at the 1986 Spring Symposium at Dallas and after Memorial Day, the traditional beginning of summer. The Dallas Symposium has occurred at the time of the usual deadline for the newsletter which has caused the June issue to have an extra long publication cycle.

1986 is almost half over! This has been a period of tremendous change for me, for the SIG, and for DEC. The SIG has changed from a "one product" special interest group to include Rally and Teamdata, DECReporter, ADE, the technical side of the A to Z Applications Generator, and non-DEC 4GL products in addition to the three flavors of DATATRIEVE. Several hats have been changed on the SIG Steering Committee and some new energy has been injected into the group by Phil Naecker. Because the areas of interest are expanding, the depth of the SIG leadership must also expand. Larry Jasmann, the immediate past SIG Chair, will be taking over as Volunteer Coordinator to build a broader and deeper base of leadership for the SIG. It is, regretfully, normal to loose several SIG leaders each year.

The first half of 1986 has seen Digital announce a whole series of machines and products in unprecedented rapid fire succession. They are accelerating the pace of technologic change. Clearly, Digital is the leader in processor hardware, networking, ASCII terminals, and fundamental software tools. Given a little time, their applications and end-user 4GL software and marketing will match their other strengths.

It is exciting to be a part of such an innovative process as the change we are experiencing. As I recall, somewhere in the oriental world you are greeted with a salutation and a curse which roughly translates, "May you live in interesting times." Well, we are certainly living in interesting times -- full of change.

From the Editor's Pen

Donald E. Stern, Jr., Warner Lambert Company, Milford, CT

I'm pleased to announce that the DTR/4GL Product Improvement Request system begins with this issue. Phil Naecker has accepted the challenge and agreed to become the PIR Editor for the SIG. Please support this effort by casting your vote and by submitting PIR's of your own.

Since this is the last issue of the newsletter in this fiscal year, it seemed appropriate to include a master index of Wombat Examiners; we hope you find it useful.

We continue to seek support from you, the readers. At this time, there are several ways you can help. As always, we seek high quality technical articles. In addition, short technical notes regarding how you solved a problem can be really useful to others. If you have a problem that you haven't solved, a letter to the Wombat Wizard might be the first step to a solution.

The bottom line is: "Support your society. Get involved!"

Ask the Wombat Wizard

Each month the Wombat Wizard will answer user's questions regarding the use of Datatrieve. To make this a regular feature of the Wombat Examiner, the Wizard needs your help! Send him your sick, your slow, and your unfriendly Datatrieve problems, and the Wizard will provide wisdom and guidance in this column every month. No question is too silly, no problem too hard, no subject too sensitive! In the tradition of Dear Abby and the Playboy Advisor, no reasonable question will go unanswered. Each month, the most interesting and instructive questions and problems will be published in this column.

To send in questions or problems to the Wombat Wizard, use the form at the back of the Newsletter or just send a letter to me c/o the Newsletter Editor.

*****

Dear WW:

I have a domain that is used to keep track of documentation being checked out of the library. We read our documentation a lot around here, so there is a lot of activity in this domain. The domain is structured with just a pointer to the document part number and description, another pointer to the employee (by badge number) and the date the document was checked out. Since we want to be able to do a quick retrieval by all three fields, I used the DTR DEFINE FILE command to put a key on each field. When a book is returned, we simply erase the record.

Everything works great, but I have two problems:

1. Sometimes it takes a very long time to store a record. In fact, it seems like it takes longer every day!
2. The file keeps growing even though the records are erased. What can I do about these problems? The documentation librarian won't talk to me anymore!

Signed,
Slower Stores Every Day

Dear Slower:

See if you can talk your librarian out of the VMS Guide to File Applications and the Convert Utility Manual. (You want to read the first of these even if you aren't using a VAX! If you're on a PDP-11, try and find the documentation IFL (Indexed File Loader.))

The basic problem is as follows: In it's default mode of operation, RMS does not actually delete a record from an indexed (ISAM) file. Instead, it just deletes the record's entries in the indexes. (In VMS's fast-delete mode, a connect option, it does even less - it just marks the record as deleted.) Thus, a record still occupies space in the file. Moreover, the index space is not reclaimed either. Thus, an RMS ISAM file cannot shrink. Further, all those deleted pointers (called RRV's - Record Reference Vectors) tend to clutter up the index buckets. This further aggravates the problem by causing more bucket splits.

Bucket splits are the way that RMS creates additional index space for itself. If an index bucket gets full, the contents of the bucket are "split" in half - one half for the existing bucket and one half to begin a new bucket. The problem is that, like records, index buckets are not reusable. So you have more and more index buckets being created, each first being filled and then being emptied as the records are deleted.

The solution to this mess is file maintenance. You maintain files by the use of the CONVERT utility (or IFL on 11's.) Simply:

```
$ CONVERT filename filename
```

(while no one is using the file) is enough to clean up the mess and reduce the size of the file. It's something you can put in a daily, weekly, or monthly batch command file if you want.

Another way to improve things is by judicious file tuning, but take one step and regularly CONVERT your files. We'll talk about file tuning in a later WW column.

******

Dear WW:

I have developed a very nice DTR application that collects a great deal of data about laboratory tests. I generate a number of statistical reports on the data, but the scientists want to use the data for simulations as well. At their instruction, I've written a procedure which uses the PRINT command to put the data into a format they can read from FORTRAN. However, every time I add data to my database, I have to run the procedure to convert the data for their programs. Is there a better way?

Signed,
Flustered by FORTRAN

Dear Flustered:

There are two solutions that are better than the one you describe, and which one you should use depends on your operating system and the nature of your database.

First, let's assume you are using a VAX and that the data of interest to the scientists is stored in a single domain. In this case, the best solution is to do nothing at all to convert the data. You don't HAVE to do anything with it - everything stored in an RMS file by DTR is completely language-independent and readable by a FORTRAN program. What's more, since you've created a record in the CDD (you had to to create the domain), the FORTRAN programmers don't even have to define the layout of the file. They can access it even if it has keys. About the only thing they can't get from the file are COMPUTED BY fields, which, of course, aren't stored in the file anyway.

Tell your FORTRAN programmers to read about the CDD in the VAX Guide to Programming on VAX/VMS - FORTRAN Edition. They will want to do the following steps:

1. Use a DICTIONARY statement to define the record layout of the file. It doesn't even require knowledge of the CDD - the record is automatically translated into FORTRAN statements, even though you wrote it using DTR record definition statements!

2. Use a RECORD statement to declare a particular instance of the record.

3. Use an OPEN statement to open your file, UNFORMATTED, with everything else left to the defaults. RMS will automatically fill in the defaults with the correct information, even if the file is indexed.

4. Simply READ the file as you would process any other file in FORTRAN.

The situation is slightly complicated if they want to take advantage of the keys, or if you have dates in the file. But the dates in the file are in standard VMS Date Format, and can be processed using calls to SYS$NUMTIM and SYS$ASCTIM. Again, refer to the Programming Under VAX/VMS Guide for information on processing dates using system services. But the real numbers you stored in the file (representing, say, the result of a lab test) or the integers you used to indicate test numbers or sample numbers, can all be accessed directly from FORTRAN.

If you are using an operating system other than VMS, or if you can't convince the FORTRAN programmers to get with the program and start accessing the CORRECT file, you can still make things a little easier. Instead of using PRINT to convert the format of the data, use the RESTRUCTURE statement or the STORE statement to convert the data from one format to another. In this case, you will have two record definitions:

```
DEFINE RECORD ORIGINAL_REC USING 01 DATA.
       03 SAMPLE_IDENTIFICATION.
           05  SAMPLE_NUMBER USAGE LONG.
           05  COLLECTION_DATE USAGE DATE.
               06 FILLER.
                  07 COLLECTION_MONTH COMPUTED BY
                      FN$MONTH(COLLECTION_DATE).
                  07 COLLECTION_DAY COMPUTED BY
                      FN$MONTH(COLLECTION_DATE).
                  07 COLLECTION_YEAR COMPUTED BY
                      FN$YEAR(COLLECTION_DATE).
           05 more stuff... ...
```

```
DEFINE RECORD CONVERTED_REC USING 01 DATA
       03 SAMPLE_IDENTIFICATION.
           05  SAMPLE_NUMBER PIC 9(9).
           05  COLLECTION_DATE.
               07 COLLECTION_DAY PIC 99.
               07 COLLECTION_MONTH PIC 99.
               07 COLECTION_YEAR PIC 9999. ...
```

Then, use the RESTRUCTURE or STORE to build the new domain from the old domain. This is must better than using the PRINT for the following reasons:

1. You have more control over the results, especially the spacing.
2. You don't have to deal with print headers.
3. You can now write a procedure to just update the domain that uses CONVERTED_REC, using the sample collection date or some other information in the original domain to only write out those records that have been updated.
4. In theory, the STORE or RESTRUCTURE should be faster than the PRINT statement because DTR doesn't have to figure out spacing and you should be able to avoid complicated formatting statements. (Actually, at one time there was a bug in DTR that made it SLOWER to STORE than to PRINT - does anyone know if that bug is still there?)

Exercise for the reader: Why did WW put a FILLER clause in the original record definition? Answer next month!

*****

Dear WW:

I have a nifty report that fits on the terminal just right when there are less than 20 records, but if there are more records the heading scrolls off the top before I can catch it. Can you help?

Signed,
Scrolled Away

Dear Scrolled:

Ah, sometimes the simplest things are such a source of frustration! But this time Wombat Magic at DECUS two years ago will come to your rescue. You're certainly not the first person to have this problem, and here is one solution you can try:

```
       REPORT report-RSE
           SET LINES_PAGE = 20         !Make sure the heading fits
           SET COLUMNS_PAGE = 80         !This is a narrow report.
           PRINT (put the detail lines here)
           (You may also want some summary lines....)
           AT BOTTOM OF PAGE -     !This is the magic part here...
               PRINT " "|*."any character and RETURN when ready",
           -
```

```
                   "or ^Z to exit."USING X
       END_REPORT
```

The prompt string (*.any character...) is printed at the bottom of the page. But we prepend the user's input with a blank and then print the resulting concatenated string with format that suppresses the string the user enters. After all, we wouldn't want to put a yucky character on the bottom of our pretty report, would we? The result is a prompt at the bottom of each page that causes DTR to wait for input from the user before continuing. It even gives us a graceful way to exit the procedure (with ^Z or the EXIT key on a VT200-series terminal.)

*****

Dear Readers:

Here is an exercise that should keep you thinking for a bit: Consider a list of proper names. Because no one told the temporary data entry person where the shift-lock key was found, all of the names were entered in upper case. But now we want to convert them to upper-and-lower case, as most proper names are done, e.g. "Smith" instead of "SMITH." (Ignore for the moment special cases such as McDonald and duBoise.)

How would you use DTR to convert the list of all upper-case names to upper-and-lower? There are at least two solutions, one relatively simple and one relatively difficult but with better performance. Can you think of them?

Introducing the Product Improvement Request System

Phil Naecker, PIR Editor - Consulting Software Engineer

Borrowing heavily from the VAX System Improvement Request system (thanks, Gary!) the DTR/4GL SIG has decided to implement a Product Improvement Request System. This is your opportunity to make known to the DTR and 4GL developers the ideas you have for improving the software product. The range of PIRs is broad - from specific "nit-picking" about syntax to long range goals and directions. You can influence the development of the products only to the extent that you take the time to fill out a PIR Submission Form and/or vote on the PIR Ballot.

The following PIRs are a just to get things rolling, generated by the PIR Editor and the DTR/4GL Steering Committee. We hope you will vote on these and then use the form in the back of this issue to submit your own PIRs. Although most of the attached PIRs are regarding DEC products, note that the PIR process is not limited to DEC 4GL's - we welcome submission on ANY 4GL product.

Please return your ballot as soon as possible. We will collect and tabulate your ballots and then forward the PIRs and vote counts to the software developers for their response. BALLOTS RECEIVED AFTER July 31, 1986 CANNOT BE COUNTED. Responses will then be presented in future editions of this Newsletter as well as the next DECUS Symposia in San Francisco. As we receive additional PIR submissions from readers (you WILL send one in, won't you?) we will publish those and begin the process again. We hope to complete the entire cycle twice a year, and will try to time it to allow the presentation of the results at the Symposia.

************** Instructions **************

You have a total of 50 points with which to vote. You may allocate points either positive (in favor of a proposal) or negative (against a proposal.) The number of points you allocate to a particular PIR indicates how strongly you feel about it. In order to assure a wide range of choices, however, you may not allocate more than 10 points (positive or negative) to any one PIR and the absolute value of the total may not exceed 50 points. Note that you must be a DECUS member to vote, (we require your membership number on the ballot) but that you do not have to be active in the DTR/4GL SIG to vote - anyone using any 4GL product is encouraged to participate in the PIR process.

Vote early! Get your friends to vote! And watch future editions of the Wombat Examiner for the results.

VAX-11 Datatrieve

PIR: S86-#001

Abstract: Provide a case-insensitive equals Boolean operator.

Description: In retrieval of true text data, especially names, it is often important to have a case-insensitive retrieval. Currently, this is possible using the FN$UPCASE operator on the record side of the Boolean expression, but this has several disadvantages.

(1) It is not clear to users how to do this.
(2) The use of the function does not use keys in the retrieval.

However, DTR could potentially do a "smart" case-insensitive retrieval, by implementing a search that would use keys (where applicable) in a recursive use of the RMS GE key retrieval. Similar functions are possible for Rdb.

PIR: S86-#002

Abstract: Use of EDIT_STRINGs on forms.

Description: Provide a means to specify that edit-strings are to be used with forms. DTR currently uses some default format for displaying data on a form. Also, a DISPLAY_FORM statement can be used to format a field in a particular manner. However, if there are many DISPLAY_FORM statements they add substantially to the complexity and compile time of the procedure and make maintenance more difficult. Either a SET option, DISPLAY_FORM keyword (without specifying each field), or record definition clause would seem to eliminate these problems.

PIR: S86-#003

Abstract: Evaluate more than one level of logical name for DTR$STARTUP.

Description: Many sites have DTR$STARTUP and DTR$SYNONYM defined system-wide. However, if a user wants to have their own DTR$STARTUP or DTR$SYNONYM, they have to define the contents of the system procedures as well. This makes maintenance difficult. Provide a means of having multiple logicals defined and used by DTR. One method might be to use search lists, or to check all the tables in LNM$-PROCESS_DIRECTORY.

PIR: S86-#004

Abstract: Provide a device independent interface to VAX-11 DTR graphics.

Description: DTR graphics is currently an un-supported "language" that works conveniently only with ReGIS devices. Users have modified it to drive non-ReGIS devices, but this is difficult and limited in functionality. However, with an apparent reduction in support of ReGIS from DEC (not available for LN03's or VAXstations, for instance) it is becoming important that DTR interface to other graphics devices. Rather than providing device-independent graphics in DTR, it would be best to provide an interface to any of the many DI packages out there, including DEC's CORE and GKS packages. The interface could be embodied in the current plotting language (or something similar) which would continue to understand DTR syntax and be usable from DTR PLOT statements.

A suggested implementation would use the new V4 VMS FIND_IMAGE_SYMBOL RTL call to invoke a sharable image "on the fly" using a logical name. Interfaces to the sharable image would involve DTR-specified calling mechanisms and arguments, and the sharable image would provide certain graphics primitives such as draw line, draw circle, plot point, etc. An even more general interface would use a mechanism similar to the DTRFND definitions, but that is not necessarily required.

Any of these techniques would accomplish the goal of letting users access DI graphics via DTR.

PIR: S86-#005

Abstract: Provide full functionality for remote domain access.

Description: There are a number of limitations in remote domain access. Lift these restrictions to allow DTR to perform CROSSes with remote sources, etc.

PIR: S86-#006

Abstract: Provide an artificial-intelligence training and guidance tool.

Description: DTR is sufficiently powerful that it serves as an excellent development environment and prototyping tool - not just as a data management system for novice users. However, the complexity of developing an application is great enough that some developers need some assistance in the definition of data formats, keys, order of CROSS operations, and so on. These concepts could be embodied in a separate AI product that would analyze DTR procedures and other metadata, working in a rule-based fashion. A substantial product could be developed with perhaps as few as 100 rules.

PIR: S86-#007

Abstract: Provide access to the TDMS Request and validation functions from DTR.

Description: Currently DTR uses TDMS with essentially the same functionality as FMS. But TDMS has several features that would be useful from DTR, including Requests and Field Validators. Field Validators have the advantage of being instantly validated, at the time the user is typing rather than upon completion of the entire form. Requests allow the use of multiple forms and complicated procedural input/output activity, without coding this information inside of DTR. This is particular useful since applications using TDMS from other languages (COBOL, etc.) already have defined requests, and since this would allow the procedural activity associated with a form to be defined only once instead of in each access to the form by DTR using DISPLAY_FORM, PUT_FORM, and GET_FORM.

PIR: S86-#008

Abstract: Value User Defined Keyword

Description: User Defined Keywords (not UDF's) currently must have the type SET, SET_NO, SHOW, STATEMENT or COMMAND. It would be useful to have a VALUE type UDK. This would allow users to define UDK's that appear in places where DTR expects value expressions. Consider, for example, the case of a DTR procedure attempting to read the VMS Accounting File. If the program uses UDFs to unpackage the accounting records, it must be given record context in the UDF arguments, e.g., PRINT FN$ACR_CPU_TIME(record-name). But if a value UDK were implemented, the unpackaging operation could retrieve record context in the same manner that a statement UDK does, e.g., PRINT ACR_CPU_TIME OF rse.

PIR: S86-#009

Abstract: Implement features supported in the Screen Management RTL.

Description: VMS RTL now contains Screen Management (SMG) support for many functions useful to DTR users. In particular, SMG allows users to define keys and windows. A keypad interface to DTR would be very helpful to many users, and a mechanism for defining keys at the site and user level would be best. An indirect command file similar to DTR$SYNONYMS seems appropriate. Also, the interface should allow for the definition of different window areas in a manner similar to that used by the VMS debugger. This would allow users to put input in one window, output in another, messages in another, and so on.

PIR: S86-#010

Abstract: Provide command recall similar to that in DCL.

Description: DTR was the first piece of software to implement editing of previous commands. But the EDIT command is limited to the previous command. Without changing the format of the EDIT command, it would be useful to have command line editing similar to that in DCL. DCL implements command line editing by processing all the input from the terminal, and it is recognized that such an implementation has difficulties. However, since DTR already keeps the previous command, a stack of these "previous" commands could be built and accessed via the up-arrow key using the SMG interface.

PIR: S86-#011

Abstract: Provide SET and SHOW commands for all debugging features of DTR.

Description: Currently, one can retrieve certain debugging information only by running DTR with the debugger. SET and SHOW commands should be provided for this functionality. E.g., SET DISPLAY_OPTIMIZATIONS and SET DISPLAY_DDMF seem like minimal features.

PIR: S86-#012

Abstract: Support bit datatypes and Boolean fields.

Description: Although CDDL can describe bit fields, DTR does not recognize them except in aggregate form (a group of bits treated as a character string.) Provide a means to reference and operate on bit fields. Bit fields are a very compact means of storing flags and options in records, and can be referenced by other languages.

Complementary to bit fields would be the concept of Boolean or Logical fields. This would be familiar to FORTRAN users and would allow operations such as the following statement:

```
IF ENTRY_COMPLETE AND NOT ENTRY_HAS_BEEN_-
REPORTED THEN
   PRINT stuff
ELSE ...
```

as opposed to

```
IF ENTRY_COMPLETE  =  "Y"  AND ENTRY_HAS_-
BEEN_REPORTED = "N" THEN
```

Note that the bit field implementation is more general, clearer, and takes one-eighth the record space.

PIR: S86-#013

Abstract: EXTRACT ALL [object-type]

Description: Provide an extension to the EXTRACT ALL command allowing the keywords DOMAINS, RECORDS, PROCEDURES, etc. as optional arguments. This would allow users to extract just portions of a dictionary without using the DMU utility.

PIR: S86-#014

Abstract: Support for other editors.

Description: Provide a means of both LSE and TPU from Datatrieve. A DTR$EDIT logical is one method of giving users a means of choosing their favorite editor. A SET EDITOR command would also allow users to easily change editors in the middle of a DTR session, and more cleanly than by using FN$CREATE_LOG to change the logical name.

PIR: S86-#015

Abstract: Extend the DEFINE FILE command to use FDL files.

Description: The proper definition of RMS indexed files is important to good performance, but the DTR DEFINE FILE command only uses some simple defaults for key parameters such as bucket size, prologue, etc. Provide a means of using FDL files directly from the DEFINE FILE command. A suggested syntax is DEFINE FILE FOR domain-name USING fdl-filename.

PIR: S86-#016

Abstract: Provide access to VMS CONVERT utility from DTR.

Description: Storing many records in RMS files using $PUTs is inefficient. There are cases when the user is aware that CONVERT should be used, but leaving DTR to use CONVERT is itself inefficient and may be difficult and time-consuming to build into an application. Calling the CONVERT command directly from DTR would improve the situation. A suggested syntax would be essentially an extension of the restructure command:

```
DTR> CONVERT rse TO  destination-domain.
```

PIR: S86-#017

Abstract: Utilize the RMS connect-time file options whenever possible.

Description: Extensions to RMS in V4 of VMS provide for connect-time performance options to be defined for a file. These connect-time options may be readily defined by users using the FDL utility, but DTR does not utilize all of them. If connect-time options (such as read-ahead and write-behind) are defined for a file, DTR should defer to those options whenever possible.

PIR: S86-#018

Abstract: Provide a means to edit a field-value directly.

Description: Provide an EDIT_VALUE or similar value function. The EDIT_VALUE would take as an argument a list of fields to be placed into a buffer before calling the user's predefined editor (EDT or TPU). The result of the EDIT_VALUE value expression could then be stored in a field or otherwise processed like any DTR value expression. This would be particularly useful for access Rdb segmented strings or long text fields in RMS or DBMS domains.

PIR: S86-#019

Abstract: Provide a sort-merge join optimization.

Description: Although DTR originated as a moderate-performance tool for novice users, it is also an excellent software development environment for experienced users. Experienced users often know enough about the application to be able to give DTR information about certain optimizations. In the case of a join or CROSS between two record streams, the user may know enough about the size and access method of the two sources to be able to specify a special kind of join called a sort-merge join. In this case, two (or more) record sources would first be sorted and then merged in a key-matching fashion rather than the traditional method of using the first record stream as a source and then finding matches in a second stream. The sort-merge join, when used, may be orders of magnitude faster than the traditional join.

PIR: S86-#020

Abstract: Delete unnecessary SORTED BY processing.

Description: There are times when a record source is already naturally sorted. This may be because it is a collection that has been sorted, or because of the keys in the RMS file or Rdb relation. Provide an optimization in DTR to avoid additional sorts on these record streams. This would improve performance in those cases where users have general-purpose procedures that are developed without knowledge of the specific sort state of the input streams, such as a standard report invoked on arbitrary collections.

PIR: S86-#021

Abstract: Provide an outer join.

Description: An outer join would allow users to deal with missing values in relational joins. Basically, an outer join provides a user with a means of getting all the records in two joined record streams, even if they don't have a match. Consider, for example YACHTS CROSS OWNER UNDER TYPE. It might return:

```
STEVE    ALBIN    VEGA
HUGH     ALBIN    VEGA
JIM      C&C      CORVETTE
         PEARSON  12M
```

Note that no owners have indicated an interest the Pearson 12M yacht, but it appears anyway because this is an outer join. In this manner, all the yachts would appear, with null values (or missing values if defined) for the owner field. The syntax is arbitrary and might be improved upon.

PIR: S86-#022

Abstract: Provide a merge operator.

Description: Provide a means to merge two record streams together to produce a single record stream. The input streams might be collections, separate RSEs, or even separate domains. An incomplete implementation, such as one that requires that the two sources be from the same domain or one that only produces a collection, would be acceptable. Note that this PIR is related to, but not the same as, the previous PIR.

PIR: S86-#023

Abstract: Provide a SET (NO) VALIDATE command

Description: Provide a means for toggling field validation on and off. There are instances when it would be useful to ignore VALID IF clauses on field definitions.

PIR: S86-#024

Abstract: Provide support for forms scrolling regions

Description: Both FMS and TDMS provide mechanisms for establishing scrolling regions. Datatrieve should be able to support this feature.

CDD

PIR: S86-#025

Abstract: Interface the CDD ACL system with the VMS security system.

Description: CDD access control should be interfaced with VMS ACL's.

Alarm ACL's should be supported, especially where privileges are used to access a CDD object. CDD access control should be a class of VMS ACL for which alarms can be separately enabled or disabled.

PIR: S86-#026

Abstract: Provide a call interface to the CDD.

Description: Provide a means for users to interface with the CDD in the same fashion that the DEC layered products can. There should be a means of creating user-defined classes of objects, doing directory operations, and so forth.

PIR: S86-#027

Abstract: Support CDDL in LSE.

Description: Provide an LSE Environment File (or definitions in text form) to support CDDL from the Language Sensitive Editor. CDDL is useful and powerful, but can also be verbose and is relatively unforgiving in syntax, making it somewhat difficult to use. Providing LSE support would make it easier for users to develop CDDL record definitions.

PIR: S86-#028

Abstract: Improve CDDL support from languages.

Description: The degree of support given to CDDL from the different layered products and languages varies greatly. Although it is recognized that the lack of CDD support in the languages rightly belongs in the lap of each of the language development teams, they are many and the CDD team is one. Therefore, the users would like the CDD team to concentrate effort on standardizing the use of CDDL from the languages. Examples of non-standard support include: COBOL allows records that don't contain structures, FORTRAN does not; most languages do not support the INITIAL_VALUE clause, although all have it in their native language constructs; only COBOL supports the EXTERNAL qualifier.

PIR: S86-#029

Abstract: Improve DELETE/HISTORY command and support for history lists.

Description: The audit history is a potentially powerful tool for managing change in an organization, but it's functionality is currently limited by inadequate tools for manipulating the CDD based upon the contents of the history. In particular, there is no way to perform a relational project to reduce the history to a unique qualifier, especially the program name. Without this capability, the list either (1) grows indefinitely, or (2) must be manually purged, keeping track of the programs that were compiled against the record. If this capability were added, then CDD managers could simply determine the logical dependencies on a record and know what systems and programs must be changed if the record changes. Further, a tool should be provided to automatically generate a list of the CDDL records that are dependent upon a record, and optionally generate a list(s) of other language source files that depend upon this record.

PIR: S86-#030

Abstract: CDD needs to have version limits and/or SET NO VERSIONS

Description: There should be version limits on a per-dictionary basis. These would be analogous to the version limits in RMS directories - creating a new version of an object would automatically purge the oldest version if the number of versions exceeded the version limit.

There are many times when version numbers are more of a hindrance than a help. There should be a means of suppressing version numbers if they are not needed.

PIR: S86-#031

Abstract: Enhance use of privileges or ACLs.

Description: There is an implicit assumption in the structure of CDD access control that the system manager or someone working for him is the manager of the CDD as well. In many shops, CDD management is undertaken by a completely different group of people. Thus, a system manager may be reluctant to give certain privileges (such as BYPASS) to a CDD manager.

A VMS privilege other than BYPASS should be required to bypass CDD protection. Alternately, provide an ACL mechanism to FORCE access to a CDD object, even if the owner of that object has protected it from access. This is the converse of the BANISH operation.

INDEX OF WOMBAT EXAMINER ARTICLES

David R. Hornback, compiler, Kentucky Geological Survey
Lexington, Kentucky

The Index of Wombat Examiner Articles was compiled from Newsletters and Volume 1 through Volume 7 of Wombat Examiner editions kept by the Datatrieve SIG. Efforts were made to include all the articles, notes, and other items of interest, however, some items defy proper indexing. Wherever possible, author names are included. No effort was made at this time to index articles by subject or author. Future indexes will be expanded along these lines.

*****

Newsletter - December 7, 1978

Watson	Use of Datatrieve With Two or More Files

*****

VOLUME 1, Number 1, July 1979

PAGE	AUTHOR	TITLE
3	Watson	Message From the Chairman-Intro to Wombat Newsletter
3		Version 1.0
4		Version 1.1 Multiple File Usage - Revised from New Orleans Notes and Newsletter Noted Previously
10	Nordby, Hilton	Version 1.1 Notes from Users, Examples of Working with More Than One Domain and/or Collection
15	Watson	One Way of Dealing With Variable Length Records
17		Version 1.1, SPR Summary

17		Version 1.1, Undocumented Features
17	Watson	Preview of Datatrieve V2
19		Datatrieve Wish List
19	Watson	Interview with Jim Starkey
23	McIntyre	An Approach to Datatrieve Management - Functions of a Datatrieve Manager and Manager Concepts
28		Questions and Answers
28		San Diego SIG Notice

*****

VOLUME 2, Number 1, May 1980

PAGE	AUTHOR	TITLE
4		Alternate Record Description May Save Space-Suggestions on Avoiding "Storage Pool Exhausted" Message Using Report Writer
5	Version 2 & 1	Commands vs. Statements - Definition of Sentence, Command and Alphabetical Summary of Commands and Statements
7	Ingram	Datatrieve Support for DEMS-11: A DECUS Report -Question and Answer Session on DEMS-11
10	Carter	Summary of Datatrieve V2 Preview Session Given by Jim Starkey at the New Orleans SIG
14		Datatrieve Wish List
18	Starkey/editor	Fall 1979 DECUS Wombat Magic
24	Bowden	DECUS UC letter
25	Booman	Solution to Problem From VOLUME 1
27	Guidi	Using Containing Clause to Emulate Variable Length Fields
28	Helton & Elia	Version 1.1 Notes Extracted From an Internal Newsletter
30	McCormick	Datatrieve Think Response = DTR-Datatrieve Problem Challenge
31	Thomas	Using Datatrieve on RSX-11M
31	Thomas	RSM-11M V3-1-Problems and Solutions
36	Gey	Fortran Support for the DEC Record Manager

*****

VOLUME 2, Number 2, Fall (?) 1980

PAGE	AUTHOR	TITLE
3	Watson	Storage Pool Exhausted? (Revisited)
7	Meirowitz	Letter and News Release - Digital Announces Enhanced Datatrieve Software
10	Rock	Letter Concerning Formation of Site Management SIG
11	Watson	Letter to Volunteers at DECUS Symposium
14	Hazelwood	Datatrieve Field Test Panel Notes
16	Shafer	Introduction to Datatrieve as Presented at DECUS 4/22/80 - Notes on Ms. Anne Duncan's Tutorial Presentation
39	Watson	Latest News About Version 2 Including a Major Bug
40		News Clipping

*****

VOLUME 3, Number 1, February 1981

PAGE	AUTHOR	TITLE
1		From Digital Equipment, Australia
2	Arsenault	Potential Problem - Phoney Data Requests From "Digital" Employees
3		Pre-symposium Workshop
5	Anderson	Letter Regarding Hardware at Symposium
6		How to Sign Up
7		DECUS Spring 81

8		Datatrieve-11 V2.0 Patch
9	Corr	Datatrieve Notes
13	Duncan	What's New in Datatrieve V2.0
23	Watson	Datatrieve V2 Dates - Date Storage in Datatrieve
29	Duncan	Conversion to Datatrieve-11 Date Datatype
30	Snyder	Screen Formatting Using Datatrieve
36	McIntyre	Processing Fortran Sequential (FCS) Files Using Datatrieve
44		Letters Requesting Help
46	Palme	The Hassle DTRFOR Package Version 1, June 1986

*****

VOLUME 4, Number 1, February 1982

PAGE	AUTHOR	TITLE
1	Curry, Sventek	Message From The Editors
3		Survey of Datatrieve Sig Members at Miami Symposium, May 21, 1981
5		Datatrieve Workshop - Notice
6	Watson	VAX Data Management Day at Atlanta DECUS
7		Review of Los Angeles DECUS
31	Duncan	Datatrieve-11 Optomization Commands and Statements
36		Letters and Papers
36	Naecker	User Defined Functions in VAX 11 Datatrieve
44	Watson	Datatrieve Training
45		Memorandum
46	Wilson	Wish List Request
50	Horn	Request For Help in Optomization of DTR
51		Letters Requesting Help
55	Boykin, Morton, Runyon	Results of Pilot Phase of PDP 11 DECUS Library Program Evaluation Project
61	Duff	DTR Dictionary Use in Processing VAX Accounting Log File
67	Kessler	FMS Working Group Summary
68	Goldfield, Lambert	A Structured Analysis of a Blind Rehabilitation Evaluation Database Using Datatrieve
81		Datatrieve Library
82	Lederman	Command File to Create a New Query DIC
84	Lederman	Prompting for a Domain Using a Command File
86	Hilton, Lemmon	Storing Records into Domains With Variable Occurs
93		Wombat Magic
94	Merkle	Printing Via a Switch
95	Morgan	Customizing the Data Dictionary Via the Start-up File
96	Landau	Cross File Computed Field
97	Hilton	VAX User Count
98	Lederman	Sorting Records with Variable Length Occurs
99	Jasmann	Procedure to Modify Variable Length Occurs
100	Schweer	Nested Procedure to Write Report from More Than One Collection
101	Landau	Procedure for Find...Beginning With "..."
102	Jones	Using Start-Up File to Verify if User is Allowed to Run DTR
103	Lederman	Procedure to Solve Complex Equations

*****

VOLUME 4, Number 2, June 1982

PAGE	AUTHOR	TITLE
2	Hilton, Et Al	Hints and Kinks Sorting, ACL, DTR Start-Up, While, Control Breaks, Totaling Lists
8		Datatrieve Wish List
15		Ballot
16		Datatrieve Masters List

*****

VOLUME 4, Number 3, November 1982

PAGE	AUTHOR	TITLE
2		Instructions for Contributors
3	Jasmann	Letter From the Chairman
4	Stephenson	Roadmap DECUS US Symposium, Fall '82 Anaheim
7	Lederman	Solving Equations in Datatrieve
16	Dickerson	Letter
17	Dickerson	Wombat Magic
31	Saxer	Datatrieve Hints and Kinks
44	Merkle	List Processing Data Example as Used With Word Processing Software
52		Datatrieve Masters

*****

VOLUME 5, Number 1, March 1983

PAGE	AUTHOR	TITLE
7	Dietterich	VAX-11 Datatrieve Technical Tutorial
10	Dietterich	VAX-11 Datatrieve Future Directions
12	Dakuzaku	A Reference Tool for Datatrieve
22	Morris	Writing Reports with VAX-11 Datatrieve
45	Jones	Drawing Pictures With Datatrieve
51	Dickerson	Answers to DTR Challenge in Previous Issue and Datatrieve and the Operating System
57	Annala	Conditional Printing Using Print All Statements
60		Wombat Magic
72		Datatrieve Sig Operating Procedures, November 30, 1982
79		Datatrieve Masters

*****

VOLUME 5, Number 2, Fall (?) 1983

PAGE	AUTHOR	TITLE
2	Annala	A VAX-11 Datatrieve Performance Monitoring Facility
13	Dickerson	On Using Computers As Computers Should Be Used
19	Swanger	Character String Manipulation Using VAX-11 Datatrieve
32	Lederman	Bar Graphs in Datatrieve
45	Racel	Sample of a Datatrieve/FMS Update Procedure Using: 1-Signed Numerics, 2-Repeating Item Lines, 3-Prompts Without Repainting
57	Azzi/editor	Wombat Magic Night, St. Louis, Spring, 1983
72	Lederman	Wish List Items and Ballot Results
77		Datatrieve Masters
79		Miscellany

*****

VOLUME 5, Number 3, January 1984

PAGE	AUTHOR	TITLE
1	Glaser	Improving Performance of RMS ISAM Files
8	Pitluck	Some Examples Using a Single Hierarchical Domain
20	Becker	How to Write Plots in DTR
32	Azzi	Time in Datatrieve
33	Toyne	Four-Column Mailing Labels in Datatrieve
36	Cassidy	Distributed Datatrieve Access Facility (DDAF)
43	Cassidy	VAX-11 Datatrieve Technical Tutorial
49		Query and Reporting Facility on the Professional 350 - Digital Equipment Corporation
52	Roseberry/editor	Fall DECUS 1983 Magic Session Las Vegas, Nevada
67		Datatrieve Masters
69		Miscellany
70		DECUS Subscription Service Order Form

*****

VOLUME 5, Number 4, Fall 1984

PAGE	AUTHOR	TITLE
1	Jasmann	Chairman's Corner
3	Lederman	Datatrieve Wish List
8	Gallagher	Wombat Magic Session
55		Masters List

*****

VOLUME 6, Number 1, Winter 1985

PAGE	AUTHOR	TITLE
2	Lederman	Datatrieve Wish List
5	Gallagher	From the Editor's Pen
6	Gallagher	Wombat Magic Session - 1984 Fall Symposium

*****

VOLUME 6, Number 2, Spring 1985

PAGE	AUTHOR	TITLE
2	Jasmann	Chairman's Corner
2		Datatrieve and Fourth Generation Language SIG Proposal
6	Wool	Datatrieve SIG Sponsored Presentation at New Orleans Symposium
16		Hints and Kinks
17		Datatrieve Application Development Standards - Williams College Administrative Computing
29		Hints and Kinks
30	Begelman	An Open, Menu-Driven, Graphic Report Generator Using DTR and VAX - Basic
36	Briggs	A Useful Datatrieve Date Function
37		Datatrieve Masters
38		Datatrieve SIG Officers and Steering Committee
40	Gallagher	From the Editor's Pen
41		News Briefs
42	Gallagher	Datatrieve User Command File

*****

VOLUME 6, Number 3, **June 1985**

PAGE	AUTHOR	TITLE
2	Jasmann	Chairman's Corner
2	Gallagher	From the Editor's Pen
3	Lederman	Datatrieve Wish List
8	Roseberry	Views From a Datatrieve Field Test Site
11		Library Notes
13		Hints and Kinks
14	Gallagher	Wombat Magic Session - 1985 Spring Symposium - Part 1

*****

VOLUME 7, Number 1, **September 1985**

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
3	Gallagher	From Editor's Pen
4	Lederman	Horizontal Bargraphs without Graphics Devices
10	Gallagher	VAX-DATATRIEVE Graphics Internals
30	Lederman	Modifying VAX-DTR Plots
35	Stern	Additional DATATRIEVE Plots

*****

VOLUME 7, Number 2, **October 1985**

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
3	Lederman	Converting Quad Word Dates to/from ASCII Under VMS
11	Gallagher	Converting Quad Word Dates to/from ASCII Under VMS
12	Stern	Accessing SYSUAF.DAT and QUOTA.SYS with VAX DATATRIEVE
15	Zuhr	A Solution to the Wishlist Request for a Prompted READY
16	Gallagher	From the Editor's Pen

*****

VOLUME 7, Number 3, **November 1985**

PAGE	AUTHOR	TITLE
2	Wool	DATATRIEVE Sponsored Sessions 1985 Fall DECUS Symposium
3	Jasman	Chairman's Corner
3	Gallagher	From the Editor's Pen
4	Bowden	Editing a File From Inside of VAX DATATRIEVE
5	Mazzoni	Supervisor Mode Libraries for DATATRIEVE-11
6	Gallagher	Loss of DECUS Leadership, Editorial
8	Gallagher/editor	Wombat Magic Session, 1985 Spring Symposium - Part 2

*****

VOLUME 7, Number 4, **December 1985**

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
2	Gallagher	From the Editor's Pen
3		'Twas the Night Before Implementation - A Seasonal Poem
4	Cordiviola/editor	Fourth Generation Language Panel - 1985 Spring DECUS
17		Errata

*****

VOLUME 7, Number 5, January 1986

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
2	Gallagher	From the Editor's Pen
3	Lederman	Disappearing Page Quota When Using Callable DATATRIEVE
4	Gallagher/editor	Wombat Magic Session - 1985 Spring Symposium, Part 3

*****

VOLUME 7, Number 6, February 1986

PAGE	AUTHOR	TITLE
2	Stern	DTR/4GL Product List
3	Gallagher	Chairman's Corner
4	Stern	From the Editor's Pen
4	McWilliams	Message From DTR/4GL Communications Committee Representative
5	Gallagher	News of the SIG
6	Roseberry	Converting Decimal UIC's to Octal UIC's
8	Stern/editor	Wombat Magic - Part 1 1985 Fall DECUS Symposium
13	Lederman	DATATRIEVE Wishlist, Fall 1985 Symposium
20		Nostalgia Bit - Original Wombat Examine Cover - Volume 1, Number 1

*****

VOLUME 7, Number 7, March 1986

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
3	Stern	From the Editor's Pen
3		DATATRIEVE/Fourth Generation Languages SIG Operating Procedures
9	Stern/editor	Wombat Magic - Part 2 1985 Fall DECUS Symposium
22	Stern	Adding Titles to DATATRIEVE Plots

*****

VOLUME 7, Number 8, April 1986

PAGE	AUTHOR	TITLE
2	Gallagher	Chairman's Corner
2	Stern	From the Editor's Pen
3	Wool	DTR/4GL SIG Sessions at U. S. DECUS Spring 1986 Symposium
5	Harris	VAX DATATRIEVE Internals
12	Lederman/editor	DATATRIEVE-11 to VAX-DATATRIEVE Conversion Panel
20	Stern/editor	Wombat Magic - Part 3 1985 Fall DECUS Symposium

*****

VOLUME 7, Number 9, May 1986

PAGE	AUTHOR	TITLE
2	Jasman	Chairman's Corner
2	Stern	From the Editor's Pen
3		Ask the Wombat Wizard
4	Harris	Introducing VAX TEAM DATA and VAX RALLY - New 4th Generation Language Solutions From Digital
23	Stern	VAX/VMS ReGIS to SIXEL Conversion

*****

VOLUME 7, Number 10, June 1986

PAGE	AUTHOR	TITLE
1	Gallagher	Chairman's Corner
2	Stern	From the Editor's Pen
3		Ask the Wombat Wizard
7	Naecker	Product Improvement Request System
18	Hornback	Index to Wombat Examiners

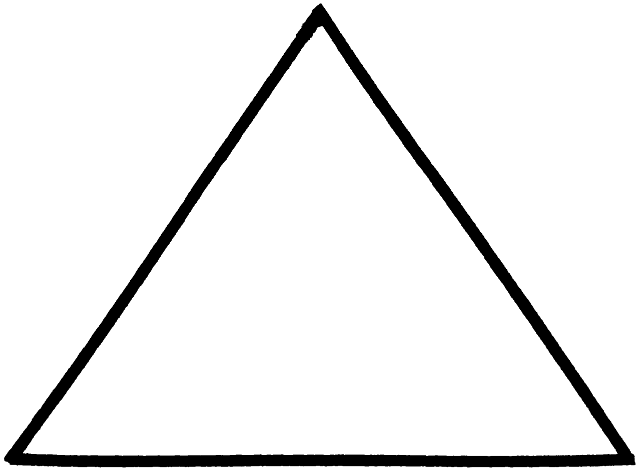
About the Cover

The Wombat reflects over his family history and growth; looking back to see how DATATRIEVE and the SIG have grown together since its introduction in the late 1970's.

This issue has some of those memories, recalled in the form of an index of articles that have appeared since the first newsletter in 1979.

DECUS

edusig



EDUSIG

Chairman
Robert A. Shive, Jr.
Millsaps College
Jackson, MS

Symposium Coordinator
Sue Bates
Northwestern Michigan College
Traverse City, MI

Communications Committee Representative
Robert W. McCarley
Millsaps College
Jackson, MS

Newsletter Editor
Fred Bell
Taft College
Taft, CA

PSS Coordinator VAX Systems SIG Liaison
Donald C. Fuhr
Tuskegee Institute
Tuskegee Institute, AL

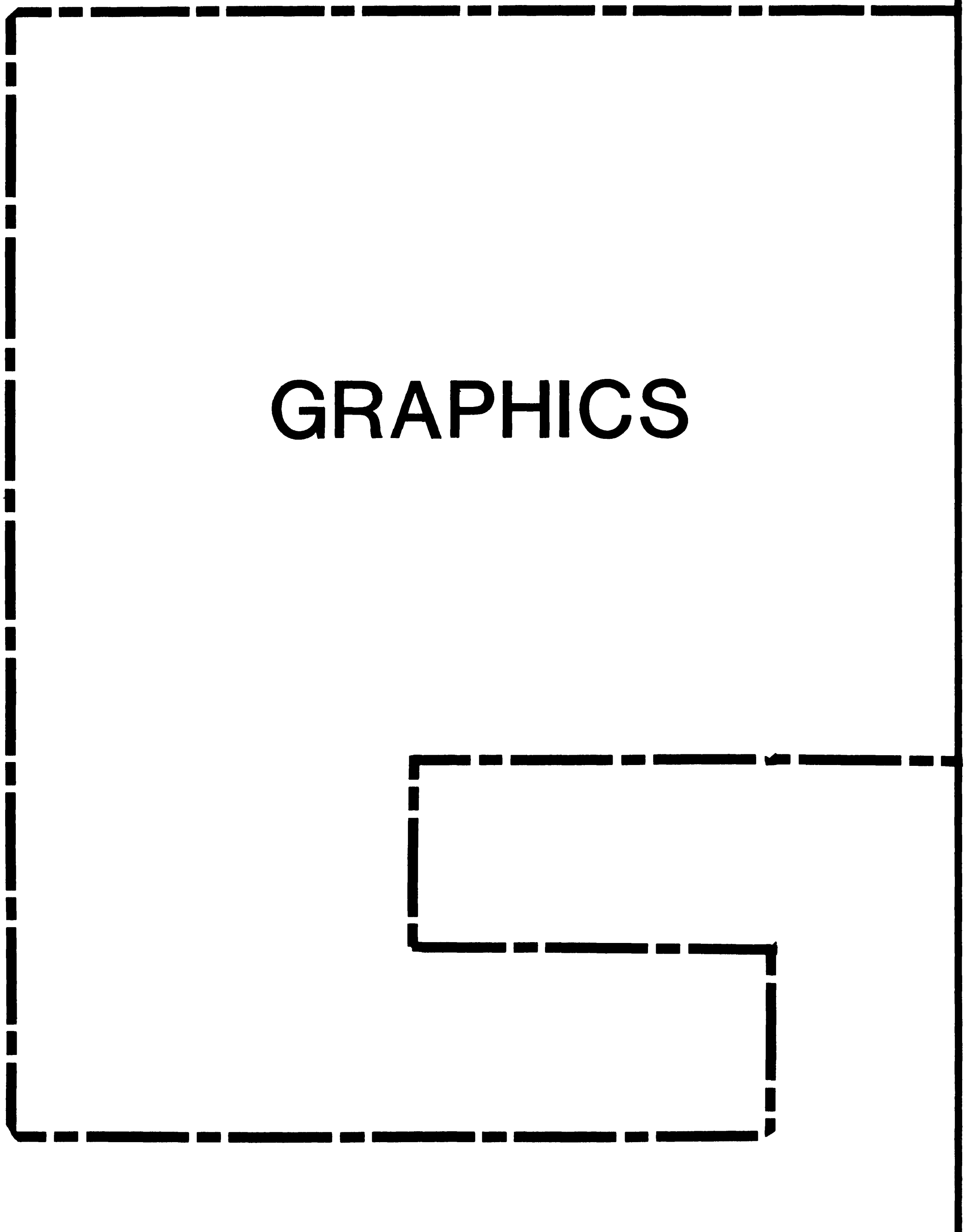
Administrative Applications Coordinator
Dave Cothrun
Taft College
Taft, CA

Courseware Coordinator
Mary Jac Reed
Off Comp Based Instruction
Newark, DE

DEC Counterpart
Gary Finerty
Digital Equipment Corporation
Marlboro, MA

DECUS

GRAPHICS



GRAPHICS

Chairman
William Kramer
University of Delaware
Newark, DE

Symposium Coordinator
Bijoy Misra
Smithsonian Institution
Cambridge, MA

Newsletter Editor
Michael P. Anton
Houston, TX

Session Note Editor
Newsletter Editor
Mike McPherson
Michigan State University
East Lansing, MI

Standards Coordinator
Jim Flatten
Ames Lab
Ames, IA

Volunteer Coordinator
Dick McCurdy
University of Florida
Gainsville, FL

Library Committee
James M. Turner
Saber Technology
San Jose, CA

DEC Counterpart
Rick Berzle
Digital Equipment Corporation
Spit Brook, NH

Information Officer
Mike York
Boeing Computer Services
Seattle, WA

Human Interface Working Group Coordinator
Dottie Elliott
Research Triangle PK, NC

Engineering Working Group Coordinator
Dana Smith
Wilmington, DE

Data Display Working Group Coord.
Joy Williams
Eaton Corp.
Southfield, MI

DECUS

H M S

HMS

HMS

Chairman VAX SIG Liaison
Thomas J. Provost
MIT/LNS Bates Linac Facility
Middletown, MA

Product Planning Coordinator
George Hamma
Synergistic Technology
Cupertino, CA

Symposium Seminar Coordinator
Mike Allen
Lawrence Livermore National Labs
Livermore, CA

Communications Coordinator
John G. Hayes
Information Systems - S. Central Bell
Birmingham, AL

Publications Coordinator (Editor)
Bill K. Walker
Monsanto Research Corp.
Miamisburg, OH

Session Notes
DAARC SIG Liaison
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

Standards Coordinator
CAMAC Working Group Coordinator
Peter Clout
Los Alamos National Lab
Los Alamos, NM

LUG Coordinator
Gregg Giesler
Los Alamos Science Lab
Los Alamos, NM

Pre-Symposium Seminar Coordinator
Mike Allen
Lawrence Livermore National Labs
Livermore, CA

TOEM (Chips % Boards)
Jack J. Peterson
Horizon Data Systems
Richmond, VA

HHK (Hardware Hints & Kinks)
Wayne Kesling
Monsanto Research Corp.
Miamisburg, OH

UNIBUS Hardware
Ron Bogue
LIV Aerospace & Defense Co.
Dallas, TX

Performance Measurement Coordinator
William Wallace
600 W. Washington St.
Peoria, IL

CAMAC Coordinator
Peter Clout
Los Alamos National Lab
Los Alamos, NM

CSS Coordinator
Pratap Gohel
E.I. Dupont
Ingleside, TX

Networks SIG Liaison
Sandra Traylor
Target Systems
Yorba Linda, CA

VAX SIG Liaison
Dave Schmidt
5100 Centre Avenue
Pittsburgh, PA

DAARC SIG Liaison
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

UNISIG SIG Liaison
Jim Livingston
1 Results Way
Cupertino, CA

SITE SIG Liaison
Emily Kitchen
A.H. Robbins Co.
Richmond, VA

RT-11 SIG Liaison
Gary Sallee
Sallee Software Consulting
Yorba Linda, CA

RSX SIG Liaison
Hans Jung
Associated Press
New York, NY

Members-At-Large
Mike Rembis
American Dade
Costa Mesa, CA

Hans Dahlke
Richland, WA

Jim Cutler
EDS Tower, 26533 Evergreen
Southfield, MI

DEC Counterparts
Terminals
Nina Abramson
Digital Equipment Corporation
Maynard, MA

TOEM (Chips & Boards)
Art Bigler
Digital Equipment Corporation
Marlboro, MA

Diagnostic
George D. Cooke
Digital Equipment Corporation
Maynard, MA

Storage
Marilyn Fedele
Digital Equipment Corporation
Maynard, MA

MSD (Micro Systems Development)
Roy Rodgers
Digital Equipment Corporation
Maynard, MA

Printer Products
Frank Orlando
Digital Equipment Corporation
Maynard, MA

DECUS Europe Liaison
Hans Zoller

THE DeVIAS LETTER

IAS SIG Steering Committee

Chairman
Bob Curley
Division of Medical Physics
University of Pennsylvania
Philadelphia, PA

WHIMS Commissioner
Kathleen Anderson
Eaton Information Management
Systems Division
Hampton, VA

Library Coordinator
Bob Schuldt
INCO Inc.
McLean, VA

RSX Liaison
Ray French
Boeing Computer Services
Seattle, WA

Member-at-Large
Doug Reno
Abbott Laboratories
North Chicago, IL

DEC Counterpart
Mike Reilly
Digital Equipment Corp.
Maynard, MA

Symposium Coordinator
Skip Stanfield
USAF
Washington, DC

Librarian
Mike Robitaille
Grumman - CTEC, Inc.
McLean, VA

DeVIAS Letter Editor
John Roman
McDonnell Douglas - Dept. N436
600 McDonnell Blvd.
Hazelwood, MO 63042

Member-at-Large
Kerry Wyckoff
LDS Church
Salt Lake City, UT

DEC Counterpart
Tim Leisman
Digital Equipment Corp.
Stow, MA

DEC Counterpart
Bob Mack
Digital Equipment Corp.
Landover, MD

Division of Medical Physics
Department of Radiation Therapy
University of Pennsylvania
Room 410
133 South 36th Street
Philadelphia, Pennsylvania 19104
15 April 1986

Dear IAS Enthusiast,

Today is the day to pay taxes. All of you (probably) do it and all of you (probably) grump a bit at it. Me too. It focuses all of our attention on the painful magnitude of the payment we must make. Some of us are luckier than others, we are allowed to pay the more local governments a piece too. Here in Philadelphia we pay the City of Philadelphia and the Commonwealth of Pennsylvania as well at the federal government. I really feel that we must support our government but find it painful to be brought to the realization of exactly how much I support our government! Somehow the rest of the year it is less painful. For all of that, today is a good day - my 11/70 that runs IAS is running! This is the first full day (and it's not over yet) that that machine has been up in three weeks.

It was necessary to move the 11/70. Not far, just down the elevator one floor and along an inside corridor to the next building. DEC Field Service took it apart ("de-installed it"), then Ted and I pushed it down the hall. It was running by late the next day, but it didn't run IAS for more than 12 hours or so before it stopped. It's all sort of a blur - The RP04, the RM03, the CPU boards, the Cache system, the floating point processor, the memory boards and the memory backplane and the two replacements of the UDA50. Everyone was amazed. I always have a feeling of paranoia when I get into the grocery store line that changes clerks and has a defective cash register, but this was really bad.

At one point all the "diagnostics" ran faultlessly and Ted found that under IAS using VFY he could hang the system by VFY DU0: with the DR0: as the system disk. If that didn't do it, then an immediate VFY DU0:/L0 would. The big question was, "What is VFY doing that DEC-X doesn't do?" No one knew. It was observed in several places that IAS was a more severe test of the hardware than DEC-X. Very interesting - the rumor is that the Field Service organization now "owns" IAS, maybe they should use it! The VFY problem was traced to an accelerometer in the RM03 that was loose.

I wish I knew what "caused" the problems. But I do know that Field Service really did a great job. And IAS support contributed too. Everyone helped! But it does feel very good that it is over. The only thing that can make today better is for the "Flyers" to win the hockey game tonight!

The bad news is the Alison Nylander has left the IAS Development Group. We shall miss her. More news along that line after the Dallas news and rumor gathering trip.

In my letter following DECworld I told you about the "Corporate Software Retirement Policy" that Mr. Andrus said he would send me. I have received nothing. Maybe next time...

Thanks for the votes. I appreciate the confidence that you've shown by voting for me. Should I win, I shall try to justify it.

Happy Tax Day,

Bob Curley
IAS SIG Chairman

Trivia for March
Tim Moffitt
Digital Equipment Corporation

I heard from Bob Mack that I was introduced to the IAS SIG last month and then turned around and didn't get a submission in by deadline.

Sorry 'bout that. Spent a lot of time in February getting ready for DECworld and let a lot of other things slide. I'll try to do better in the future but you know how VAX people are...virtual.

So now the question(s) for the day:

What PDP-11 came standard with two Unibusses, why did it have two, what were the reasons that the concept never became popular, and what controllers were designed to take advantage of this unique system?

If you knew that the PDP11/45 /50 /55 systems had two Unibusses, congratulations. The use of the second bus showed that as early as 1972 DEC realized that some applications could generate enough traffic to begin to saturate a single Unibus. The solution was to build a second path into the processor where a block of high speed memory could reside and hang the second Unibus off that path. There was no priority arbitration on the second bus; its sole purpose was to provide for CPU to memory and peripheral IO to memory transfers.

Several issues worked to kill the advantages of this system. First and probably foremost was a limitation of 32K words of memory on the second bus with no easy way of expansion. Second was the lack of controllers that could be set up for a transfer on the primary bus, do the actual transfer on the second bus, and then interrupt across the primary bus upon completion or error. Third was the difficulty of the software to drive such an interface. Since NOBODY was happy with 32Kw of memory, everybody added memory to the 11/45 and the only place to put it was on the primary bus. Once that was done, it was up to the software to figure out if the next transfer would be crossing a memory boundry from the primary to the secondary bus or visa-versa. If so the transfer would have to be broken into segments and directed to take place across the proper bus. A major pain to be sure.

The simple fix to the problem was to tie the two Unibusses together and treat them as one. DEC sold a lot of M9200 Unibus jumpers that did just that and if you open an 11/45 today my guess is that you'll find one lurking in the CPU backplane just after the MOS memory slots.

The only controller that I know of that was ever designed to use both Unibusses was the (infamous) RH11. Look in the prints and you can see the B bus logic. also check out the register definitions for RH Control and Status 1 and you'll see bit 10 used to steer the transfer to the proper bus.

Lest you think this effort was a total waste, keep in mind that the fastbus (secondary Unibus) was modified slightly to support much more memory, a bipolar cache was added and the ugly duckling 11/45 turned into the 11/70; until recently the king of the PDP-11 hill. Who says you can't teach an old dog new tricks?

tim moffit

Trivia for April

So I hope you're all still with me and not too bored.

We have a couple of questions this month that deal with terminal communications hardware.

Digital has used asynchronous interfaces, both single line and multiplexed to connect terminals to PDP-11's a tradition that continues into the VAX line although terminal servers are coming on strong. Let's revisit some of these devices.

What was the first single line asynchronous comm device on the PDP-11 and how did it differ from its successor, the DL11. Next...how many different types of DL11's are there really? Hint: don't believe what you read in that old systems and options summary you've kept around for the past seven years. Don't include the DL11-W, that's a totally different animal.

What were the first two async. communications muxes that DEC introduced, how did they differ, and what replaced them?

Finally, and here is the tough one, in current loop operation (as opposed to RS232) why was the value of 20 milliamps chosen as a standard?

So, how many of you remembered the KL11, a set of three boards (M780, M105, M7820) that was the predecessor to the DL11? Timing was set with an RC network instead of a crystal and PM's involved checking and adjusting the clock each month. The big innovation was the use of an LSI chip (the UART) to handle the serializing and deserializing of data. True DEC diehards will remember the DC11, an interface that did essentially the same thing without the aid of a UART. Needless to say itwas larger and tougher to work on. You could only get two DC11's in one four slot backplane. At least it had modem control.

There really are only two types of DL11's. The DL11-B/D/E variants were identical with the exception of jumper configurations that could be changed in the field. The DL11-A/C variants were similar to the others but came without the EIA interface chips and support circuitry.

It became obvious that an async. mux would be a better way of connecting terminals than stringing 16 or 20 DL11's onto a system so the DJ11 and DH11 were introduced. Both devices multiplexed 16 lines and had a 64 character silo. The DH11 was more sophisticated with features such as DMA on output, split speeds, the ability to support full modem control, and could be configured to support various electrical interfaces on a per line basis. The DJ11 was replaced by the DZ11 and the DH11 by the DHU11.

The final question...why 20 milliamp? Well in the old days (when I was new at DEC) there were these printing terminals called teletypes. Teletypes used mechanical parts to encode and decode characters. It was found that 20 milliamps of current was sufficient to keep the copper commutator clean without burning it or destroying the carbon brushes that rubbed against it. It's another instance of a requirement for one vendor becoming a standard for the industry. Now there's a topic for another column...

tim moffit

digital **SOFTWARE PERFORMANCE REPORT**

FIELD NO.:

CORPORATE SPR NO.: 993121

✓ TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

PAGE ______ OF ______

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
IAS	3.2B	VFY	D0528	28-march-86

NAME: Robert O. Smith
FIRM: Dept. Physiology and Biophysics
Univ. of Mississippi Medical Center
2500 North State Street
ADDRESS: Jackson, Mississippi 39216
CUST. NO.: MSD 00533

DEC OFFICE AND CONTACT PERSON: Jackson, Ms. Ed Palmer

DO YOU HAVE SOURCES? YES ☐ NO ☒

REPORT TYPE/PRIORITY

- ☒ PROBLEM/ERROR
- ☐ SUGGESTED ENHANCEMENT
- ☐ OTHER

1. ☐ HEAVY SYSTEM IMPACT
2. ☒ MODERATE SYSTEM IMPACT
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTION

SUBMITTED BY: Robert O. Smith
PHONE: (601) 984-1808

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
MAG TAPE ☐ FLOPPY DISKS ☐ LISTING ☒ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? PLEASE EXPLAIN IN PROVIDED SPACE BELOW. YES ☐ NO ☒

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
11/70	5022	512 Kb	Mag. tape	RP04	☐

The file structure verification utility (VFY) does not delete bad file headers on a volume when that switch is used. Apparently, it does not even detect that they exist although PIP finds them and can do nothing with them. The disk pack in question was initialized, Etc. on an IAS version3.1 system. However, if there is a way to delete files with bad headers on that version, we have not found it. That was the reason for using the VFY on the IAS Version3.2B system and it doesn't seem to work either.

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

SHORT NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL	PRB. TYPE
DATE RECEIVED (MAIL)		DATE TO MAINTAINER	XFER DATE	LOGGED ON	
DATE RECEIVED (ASG)		DATE RECEIVED FROM MAINTAINER	ANSWERED	LOGGED OFF	

EN-01044-07-REV J (35C)

ADMINISTRATIVE SERVICES

Geleen, 4-3-'86

Mr.John Roman
Mc Donnel Douglas Corp. Dept. N436
600 Mc Donnell Blvd
Hazelwood, Missouri 63042

Dear John,

These are the SPR's which I sent to DEC in the past months. Some of them are about problems I encountered with IAS V3.2 update B.

I got a quick but very dissatisfying answer on the 11/23+ generation problem. I have enclosed a patch on module EM03 of the EXEC to overcome any generation problems on our 11/44 with IAS V3.2.

I always had assumed that the code in EM03 was a nice implementation of the feature to generate a different type of CPU system on our 11/44.
It was a surprise to hear that it does not work. Up to now I have been able to generate systems with every and any type of CPU on our 11/44 and it always worked well. Those of you who are using an 11/44 and who need to use the DB or DR-handler, should take notice of the SPR's concerning these handlers. The ECC-correction in the handlers does corrupt your memory.

I have asked DEC to correct these handlers in update C. You might be able to correct the handlers yourself. If you have troubles, phone me (take note of the time-difference though).

Yours sincerely

H. Plasman
Informatie Services
DSM Limburg BV
Postbus 600
6160 MJ Geleen.

---

digital SOFTWARE PERFORMANCE REPORT

98760

PAGE 1 OF

✓ TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
IAS	V3.2	TS11-handler	3.205	5-11-1

NAME: J.L.C. Plasman
FIRM: DSM Limburg BV
Afd. Systeem Technieken
Postbus 600
ADDRESS: 6160 MJ GELEEN
CUST. NO.:

DEC OFFICE AND CONTACT PERSON

DO YOU HAVE SOURCES YES ☒ NO ☐

REPORT TYPE/PRIORITY

☒ PROBLEM/ERROR
☐ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☐ HEAVY SYSTEM IMPACT
2. ☒ MODERATE SYSTEM IMPA
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGEST

SUBMITTED BY:

PHONE: 04494-66755

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
MAG TAPE ☐ FLOPPY DISKS ☐ LISTING ☒ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? PLEASE EXPLAIN IN PROVIDED SPACE BELOW. YES ☐ NO ☒

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
PDP11/44	19665	256K	Magn. tape	RK07	☐

TS11-HANDLER

1. Unexpected interrupt crashes system
   Do not do a RTS PC because it will corrupt the stack. It is better to jump to CALLR[illegible]..INTX.

2. BRU copie on more than one tape fails
   When the VCK bit in XSTAT0 is set and the unit is online, the command will fail if the CM.CVC bit in the command header is not set.

   A check in EXECOM on the VCK bit and conditionally setting CM.CVC will prevent problems in command execution.

3. Mount operation
   The mount operation will always be succesfull unless the unit is offline. However the operation may be unsuccesfull and therefore it would be better not to overwrite RTRNCD in SETCHR.

4. SETCHR operation
   The SETCHR operation will always [illegible] fail, when SR.OFL is up. TSSR however is only updated when a command is executed and it may not represent the actual status of the unit. Therefore one first should perform a CC.GST before testing TSSR.

General remark

When MOU MS:TEST fails (e.g. wrong lable), the tape is rewinded and unloaded. The latter operation is a bit nasty.

Can I prevent this and if so how?

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

SHORT NAME	[illegible]	[illegible]	XFER SPR	PL	PRB. TYP
DATE RECEIVED [illegible]		DATE TO MAINTAINER	XFER DATE	LOGGED ON	
DATE RECEIVED [illegible]		DATE RECEIVED FROM MAINTAI	ANSWERED	LOGGED OFF	

EN-01044-07-REV1 (35C)

SOFTWARE PERFORMANCE REPORT

FIELD NO.:	CORPORATE SPR NO.:

987605

PAGE ______ OF ______

TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

ERATING SYSTEM IAS	VERSION V3.2	SYSTEM PROGRAM OR DOCUMENT TITLE DR-handler	VERSION OR DOCUMENT PART NO. 43	DATE 3.10.1985

NAME: H.L.C. Plasman
IRM: DSM Limburg BV
Afd. Systeem Technieken Ms1
Postbus 600
DRESS: 6160 MJ GELEEN
ST. NO.:

DEC OFFICE AND CONTACT PERSON

DO YOU HAVE SOURCES? YES ☒ NO ☐

REPORT TYPE/PRIORITY

☒ PROBLEM/ERROR
☐ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☒ HEAVY SYSTEM IMPACT
2. ☐ MODERATE SYSTEM IMPACT
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTION

MITTED BY: PHONE:

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
G TAPE ☐ FLOPPY DISKS ☐ LISTING ☐ DECTAPE ☐
HER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? YES ☐ NO ☒
PLEASE EXPLAIN IN PROVIDED SPACE BELOW.

TYPE DP 11/44	SERIAL NO. 19665	MEMORY SIZE 256K	DISTRIBUTION MEDIUM Magn. tape	SYSTEM DEVICE RK07	DO NOT PUBLISH ☐

DR-HANDLER

Problem description

1. Error in ECC corrections on 11/44.

   On an 11/44 CPU UMR's are used for transfers. In routine ..ALMR R.PB and R.PB+2 are changed. However the ECC correction uses R.PB+2 and R.PB+12 to form the real address of the buffer. Therefore the ECC correction on an 11/44 CPU will not take place on the right address, but somewhere else in memory, which can lead to weird problems.

2. Routine SAVREG clears the contents of R.PB+12 and does not reset it on an 11/44 CPU. This will lead to errors when more than one ECC correction takes place for a transfer. This routine is extensively improved for the DB-handler and we suggest to implement this code in the DR-handler

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

T NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL. / PRB. TYPE
RECEIVED (MAIL)		DATE TO MAINTAINER	XFER DATE	LOGGED ON
RECEIVED (ASG)		DATE RECEIVED FROM MAINTAINER	DATE ANSWERED	LOGGED OFF

1044-07-REVI (35C)

SOFTWARE PERFORMANCE REPORT

FIELD NO.:	CORPORATE SPR NO.:

987601

PAGE ______ OF ______

TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

OPERATING SYSTEM IAS	VERSION V3.2	SYSTEM PROGRAM OR DOCUMENT TITLE [illegible] ndler	VERSION OR DOCUMENT PART NO. 46	DATE 2-10-1985

NAME: H.L.C. Plasman
FIRM: DSM Limburg BV
Afd. Systeem Technieken Ms1
Postb[illegible]
ADDRESS: 6160 MJ GELEEN
CUST. NO.:

DEC OFFICE AND CONTACT PERSON

DO YOU HAVE SOURCES? YES ☒ NO ☐

REPORT TYPE/PRIORITY

☒ PROBLEM/ERROR
☐ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☒ HEAVY SYSTEM IMPACT
2. ☐ MODERATE SYSTEM IMPACT
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTION

SUBMITTED BY: PHONE:

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
MAG TAPE ☐ FLOPPY DISKS ☐ LISTING ☐ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? YES ☐ NO ☒
PLEASE EXPLAIN IN PROVIDED SPACE BELOW.

CPU TYPE PDP 11/44	SERIAL NO. 19665	MEMORY SIZE 256K	DISTRIBUTION MEDIUM Magn. tape	SYSTEM DEVICE RK 07	DO NOT PUBLISH ☐

DB-handler

Problem Description

Error in ECC-correction on 11/44.

On an 11/44 CPU UMR's are used for transfers. In routine ..ALMR R.PB and R.PB+2 are changed. However the ECC correction uses R.PB+2 and R.PB+12 to form the real address of the buffer. Therefor the ECC correction on an 11/44 CPU will not take place on the right address, but somewhere else in memory, which can lead to weird problems.

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

SHORT NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL. / PRB. TYPE
DATE RECEIVED (MAIL)		DATE TO MAINTAINER	XFER DATE	LOGGED ON
DATE RECEIVED (ASG)		DATE RECEIVED FROM MAINTAINER	DATE ANSWERED	LOGGED OFF

N 01044-07 REVI (35C)

digital **SOFTWARE PERFORMANCE REPORT**

FIELD NO.: | CORPORATE SPR NO.: 060419

TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

PAGE ____ OF ____

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
IAS	3.1	DECNET	v3.0	15.03.85

NAME: Plasman J.L.C.
FIRM: DSM Limburg BV
Afd. Informatie Services Ms1
Postbus 600
ADDRESS: 6160 MJ Geleen

CUST. NO.:

DEC OFFICE AND CONTACT PERSON:

DO YOU HAVE SOURCES? YES ☐ NO ☒

REPORT TYPE/PRIORITY

☒ PROBLEM/ERROR
☐ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☒ HEAVY SYSTEM IMPACT
2. ☐ MODERATE SYSTEM IMPACT
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTION

SUBMITTED BY: H. Plasman
PHONE: 04494-66755

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☐ NO ☒

ATTACHMENTS

MAG TAPE ☐ FLOPPY DISKS ☐ LISTING ☒ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? PLEASE EXPLAIN IN PROVIDED SPACE BELOW. YES ☐ NO ☒

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
PDP11/44	19665	256K	Magn. tape	RK07	☐

Problem description:

Bug in NT.DV module DV.

In the routine starting at C02160 (see listing) a false usage is made of register R4 in lines 415 and 416. R4 is throughout this routine an offset in a CCB. The MOVB and TSTB instructions are therefore executed on the statusword in the CCB. When the software executes this part of the coding it will loop between line 416 and 417, because bit 15 will not be cleared.

On lines 415 and 416 R4 should be changed into R1, which contains at the moment the SAR addres of the associated DV. When this part of coding is executed with the change made, the MOVB and TSTB instructions work on the LCR of the DV and then the coding makes sense.

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

SHORT NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL	PRB. TYPE
DATE RECEIVED (MAIL)		DATE TO MAINTAINER	XFER DATE	LOGGED ON	
DATE RECEIVED (ASG)		DATE RECEIVED FROM MAINTAINEI	›SWERED	LOGGED OFF	

EN-01044-07-REVI (35C)

ADMINISTRATIVE SERVICES GROUP, SWS

```
400 002156  000207                             RETURN
401 002160  162764  000000G 000016  C02160:  SUB    #DLC$OV,16(R4)
402 002166  062764  000000C 000020           ADD    #DLC$OV+CRC$OV,20(R4)
403 002174  062764  000000C 000030           ADD    #DLC$OV+CRC$OV,30(R4)
404 002202  010503                           MOV    R5,R3
405 002204  062703  000002                   ADD    #2,R3
406 002210  004767  000000G                  CALL   $CMQIN
407 002214  021363  000002                   CMP    (R3),2(R3)
408 002220  001022                           BNE    C02266
409 002222  004067  002136                   JSR    R0,C04364
410 002226  103002                           BCC    C02234
411 002230  000167  000000G                  JMP    $RDBRT
412 002234  004767  001744          C02234:  CALL   C04204
413 002240  004767  176522                   CALL   C00766
414 002244  152765  000040  000035           BISB   #40,35(R5)
415 002252  116564  000035  177775           MOVB   35(R5),177775(R4) }
416 002260  105764  177775          C02260:  TSTB   177775(R4)        should be R1
417 002264  100775                           BMI    C02260
418 002266  000207                  C02266:  RETURN
419 002270  004067  002070          C02270:  JSR    R0,C04364
420 002274  103474                           BCS    C02466
421 002276  013746  000000G                  MOV    @#PS,-(SP)
422 002302  112737  000000G 000000G          MOVB   #PR7,@#PS
423 002310  010503                           MOV    R5,R3
424 002312  062703  000022                   ADD    #22,R3
425 002316  011446                  C02316:  MOV    (R4),-(SP)
426 002320  005014                           CLR    (R4)
427 002322  004767  000000G                  CALL   $CMQIN
428 002326  012604                           MOV    (SP)+,R4
429 002330  001372                           BNE    C02316
430 002332  105765  000036                   TSTB   36(R5)
431 002336  001403                           BEQ    C02346
432 002340  112637  000000G                  MOVB   (SP)+,@#PS
433 002344  000207                           RETURN
434 002346  004767  000000G         C02346:  CALL   $CMQRM
435 002352  112637  000000G                  MOVB   (SP)+,@#PS
436 002356  004767  002110                   CALL   C04472
437 002362  010465  000014                   MOV    R4,14(R5)
438 002366  062704  000014                   ADD    #14,R4
439 002372  010465  000016                   MOV    R4,16(R5)
440 002376  000302                           SWAB   R2
441 002400  004767  002242                   CALL   C04646
442 002404  013746  000000G                  MOV    @#PS,-(SP)
443 002410  112737  000000G 000000G          MOVB   #PR7,@#PS
444 002416  004767  002510                   CALL   C05132
445 002422  052713  000004                   BIS    #4,(R3)
446 002426  032713  000040                   BIT    #40,(R3)
447 002432  001004                           BNE    C02444
448 002434  152765  000020  000033           BISB   #20,33(R5)
449 002442  000404                           BR     C02454
450 002444  112710  000013          C02444:  MOVB   #13,(R0)
451 002450  052711  000004                   BIS    #4,(R1)
452 002454  116515  000001          C02454:  MOVB   1(R5),(R5)
453 002460  012643                           MOV    (SP)+,-(R3)
454 002462  112637  000000G                  MOVB   (SP)+,@#PS
455 002466  000207                  C02466:  RETURN
456 002470  004067  001670          C02470:  JSR    R0,C04364
```

digital SOFTWARE PERFORMANCE REPORT

FIELD NO.:
CORPORATE SPR NO.:

987609

TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

PAGE 1 OF 1

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
IAS	V3.2	DECNET	V3.0	5-11-1985

NAME: J.L.C. Plasman
FIRM: DSM Limburg BV
Afd. Systeem Technieken
ADDRESS: Postbus 600
CUST. NO.: 6160 MJ GELEEN

DEC OFFICE AND CONTACT PERSON

DO YOU HAVE SOURCES? YES ☐ NO ☒

REPORT TYPE/PRIORITY

☒ PROBLEM/ERROR
☐ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☐ HEAVY SYSTEM IMPACT
2. ☒ MODERATE SYSTEM IMPAC
3. ☐ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTIO

SUBMITTED BY:
PHONE: 04494-66755

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
MAG TAPE ☐ FLOPPY DISKS ☐ LISTING ☒ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? PLEASE EXPLAIN IN PROVIDED SPACE BELOW. YES ☐ NO ☒

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
PDP11/44	19665	256K	Magn. tape	RK07	☐

PROBLEM DESCRIPTION:

When you have brought up DECNET on a node, the following command sequence (commands are abbreviated)

```
NCP > SET EXE STA OFF
NCP > CLE SYS
NCP > SET SYS
NCP > SET EXE STA ON
```

causes diminishing of free pool space of about 18 to 20 nodes. When your system is up for a long time and you frequently have to load and unload DECNET for whatever reason, you eventually have to reboot the system, because of nodepool shortage.

Analyzing the crash-dump reveals that after DECNET is unloaded, the physical link database remains present in SCOM. After loading DECNET the database is not reused, but a new one is created.

SOLUTION:

The unload procedure should really free all DECNET resources picked in SCOM.

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

SHORT NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL	PRB. TYPE

DATE RECEIVED (MAIL)	DATE TO MAINTAINER	XFER DATE	LOGGED ON

DATE RECEIVED (ASG)	DATE RECEIVED FROM MAINTA	ANSWERED	LOGGED OFF

EN-01044-07-REVI (35C)

---

digital SOFTWARE PERFORMANCE REPORT

FIELD NO.:
CORPORATE SPR NO.:

060397

TO SET UP FOR PROPER ALIGNMENT, START AT MARK BELOW.

PAGE ____ OF ____

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
IAS	v3.1.	DECNET	v3.0	12.03.'85

NAME: J.L.C. Plasman
FIRM: DSM LIMBURG BV
Afd. Systeem Technieken
Postbus 600
ADDRESS: 6160 MJ GELEEN
CUST. NO.:

DEC OFFICE AND CONTACT PERSON

DO YOU HAVE SOURCES? YES ☐ NO ☒

REPORT TYPE/PRIORITY

☐ PROBLEM/ERROR
☒ SUGGESTED ENHANCEMENT
☐ OTHER

1. ☐ HEAVY SYSTEM IMPACT
2. ☐ MODERATE SYSTEM IMPACT
3. ☒ MINOR SYSTEM IMPACT
4. ☐ NO SIGNIFICANT IMPACT
5. ☐ DOCUMENTATION/SUGGESTION

SUBMITTED BY: H. Plasman
PHONE: 04494-66755

CAN THE PROBLEM BE REPRODUCED AT WILL? YES ☒ NO ☐

ATTACHMENTS
MAG TAPE ☒ FLOPPY DISKS ☐ LISTING ☐ DECTAPE ☐
OTHER:

COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? PLEASE EXPLAIN IN PROVIDED SPACE BELOW. YES ☐ NO ☐

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
P 11/44	19665	256K	Magn. tape	RK07	☐

Follow-up on our SPR nr. 917522, DEC reference nr. H11-2927

Problem description

As we stated in our previous SPR we had a problem with the down line load of RSX11S task ima

Further investigations have revealed, that:

1. The problem is caused by secundary and tertiary loader task. Both loaders are built for our systems with module DLDRV from BOOTS.OLB. This module contains a test which is perfect for DL11-E interfaces. For other DL11 types however it is a loop of 64K times, which causes the delay which we discovered (see listing 1).

2. Module DLVDRV does not contain this loop and is therefor suited to serve non DL11-E type DL11 interfaces.

2. NETGEN does only fake the support of a DLV-type interface. The NETGEN procedure mutates a DLV remote bootstrap device name immediately into DL. So NETGEN will not allow module DLVDRV to be built into the loader tasks.

3. The hardware bootstrap ROM has a special test procedure which discriminates the DL11-types in a very simple way. This procedure could be used in module DLDRV and make it suitable for all DL11 types (listing no. 2).

We have the following suggestion to overcome the problem.

a. Either improve the NETGEN procedure such that it can discriminate between DL11-E and non DL11-E interfaces.

b. Or implement the hardware test procedure in module DLDRV.

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION

SHORT NAME	MNT. CAT.	MNT. GRP.	XFER GRP.	PL	PRB. TYPE

DATE RECEIVED (MAIL)	DATE TO MAINTAINER	XFER DATE	LOGGED ON

DATE RECEIVED (ASG)	DATE RECEIVED FROM MAINTAINER	ERED	LOGGED OFF

EN-01044-07-REVI (35C)

ADMINISTRATIVE SERVICES GROUP. SWS

listing 1

```
        .TITLE  DLDRV
        .IDENT  /V03.00/
        .ASECT
        .GLOBL  DEVTYP
        .GLOBL  REBOOT
        .PSECT
        .GLOBL  SNDDRV
        .GLOBL  INIDRV
        .GLOBL  TMR
        .GLOBL  RCVDRV
        .PSECT
C00000: HALT
TMR:    HALT
SNDDRV: MOV     2(SP),R5
        SUB     #14,R5
        MOV     #177777,(R5)+
        MOV     #177777,(R5)+
        MOV     R5,R4
        MOVB    #220,(R5)+
        MOVB    R2,(R5)+
        SWAB    R2
        MOVB    R2,(R5)
        SWAB    R2
        MOV     R2,R3
        BISB    #300,(R5)+
        CLRB    (R5)+
        MOV     #400,(R5)+
        MOV     #6,R2
        CALL    C00456
        MOV     R5,(R4)+
        MOV     R3,R2
        CALL    C00456
        MOVB    R5,(R4)+
        SWAB    R5
        MOVB    R5,(R4)+
        MOV     #6,(R1)
        CLR     -(SP)
C00112: BIT     #20000,(R1)
        BNE     C00124
        DEC     (SP)
        BNE     C00112
C00124: TST     (SP)+
        CMP     (R1)+,(R1)+
        TST     (R1)+
        MOV     #377,(R1)
        MOV     R3,R2
        ADD     #21,R2
        MOV     2(SP),R4
        SUB     #14,R4
        BR      C00160
C00156: MOVB    (R4)+,(R1)
C00160: TSTB    177776(R1)
        BPL     C00160
        DEC     R2
        BNE     C00156
        CMP     -(R1),-(R1)
        BIC     #4,-(R1)
```

not in DLVDRV.

wait loop for non DL11-E interface

listing 2.

```
        [illegible]
        .IDENT  /20NO84/
;
 ;;;    .ASECT
;
;************************* FIRST ROM ******************************************
        NOP
        BR      STARTO
STARTO: SEC                     ;
        MOV     #0,R0           ;
S0:     MOV     #176540,R1      ; CSR-ADDRESS (ALTERNATE 177560)
        MOV     PC,R4           ;
        BMI     S1              ;
        BR      S2              ;
ODD:    .WORD   26
        .WORD   0               ;
START1: SEC                     ;
        MOV     #1,R0           ;
        BR      S0              ;
S1:     JMP     @#165564        ;
S2:     BIC     #177740,R0      ;
        MOV     R0,@#17776      ; SAVE CSR-OFFSET
        ASL     R0              ;
        ASL     R0              ;
        ASL     R0              ;
        ADD     R0,R1           ;
        MOV     @#ODD,@#4       ; SET ODD-ADDRESS TRAP VECTOR
        MOV     @#ODD+2,@#6     ;
        CMP     #177422,BOOTME  ;
        BNE     HALTX           ;
        CMP     #12746,RECEIVE  ;
        BNE     HALTX           ;
        MOV     PC,R2           ;
        CMPB    #<.&377>,R2     ;
        BNE     CONTX           ;
        CMP     #12711,M173624  ;
        BNE     CONTX           ;
        BR      XYZ             ;
HALTX:  HALT                    ;
        BR      HALTX           ;
CONTX:  MOV     #17440,SP       ;
CONTX1: MOV     PC,R4           ;
        ADD     #<BOOTME-.>,R4  ;
        MOVB    (R4)+,R3        ;
        RESET                   ;
        MOV     #6,(R1)         ;
        BIT     #4,(R1)         ;
        BR      3$              ;
        .WORD   125357          ; CHECKSUM (ALTERNATE 107616)
;************************* SECOND ROM *****************************************
        .WORD   177776          ;
3$:     BEQ     5$              ;
4$:     BIT     #20000,(R1)     ;
        BEQ     4$              ;
5$:     BR      7$              ;
6$:     MOVB    (R4)+,6(R1)     ;
7$:     TSTB    4(R1)           ;
```

procedure to discriminate between DL11 interface types.

416
541
310.

SPR RESPONSE

:   SPR Number 11-79646

	System	System Version	Component
oftware:	IAS	3.2	EXECUTIVE

Problem
Statement

we tried to generate an 11/23+ on our 11/44. SYSGEN Phase One caused no problem. Booting for SYSGEN Phase Two immediately caused our first system crash. Crash analysis revealed that the DL Handler caused the crash while loading the Terminal Handler. We discovered that in .UMR22+1 the Q-bus ON.QB bit was set. The DL Handler tests on this bit and assumes when it is set that it talks to the Q-bus controller with the extra register. On a non Q-bus system this causes a crash.

The next thing we tried was starting SYSGEN Phase Two on the 11/23+ itself. This caused our second crash. Analysis revealed that now .UMR22+1 was cleared. Therefore, bit 4 in MMR3 is not set, so Memory Management is not enabled, which caused the crash. We conclude that IAS V3.2 does not support 11/23+ CPU types.

Response

ank you very much for your SPR. IAS V3.2 does indeed support the 11.23+ CPU type. There has been a restriction since RSX-11D such that IAS does not allow a SYSGEN to be successfully done on one CPU type to run on another CPU type, as you have done for an 11/23+ on an 11/44. You can try to do a SYSGEN for the 11/23+ on the 11/44 only if you specify memory size of less than 124K. This may work, but it is not supported under IAS. In addition, EXEC module EM03 does not know anything about CPU types, it is the module SAVE that gathers all the CPU information and decides what to do with the CPU after SYSGEN.

```
;
;       LAST UPDATE: H.PLASMAN 29-JUL-85        HP001 SUPPORT Q-BUS SYSTE
;
;       PREVIOUS    : .IDENT /3.201/
;
        .TITLE  EM03
        .IDENT  /3201HP/
        .ASECT
        .GLOBL  AF.PF
        .GLOBL  A.TD
        .GLOBL  A.TF
        .GLOBL  BOTKSP
        .GLOBL  CRTNX1
        .GLOBL  CRTNX2
        .GLOBL  CURMOD
        .GLOBL  EV.PR
        .GLOBL  EV.SE
        .GLOBL  EXINTX
        .GLOBL  ISVFEC
        .GLOBL  ISVSR0
        .GLOBL  ISVSR1
        .GLOBL  ISVSR2
        .GLOBL  KP.AR0
        .GLOBL  KP.AR7
        .GLOBL  KP.DR0
        .GLOBL  KP.DR7
        .GLOBL  ON.CSM
        .GLOBL  ON.QB
        .GLOBL  ON.SD
        .GLOBL  ON.UM
        .GLOBL  ON.22
        .GLOBL  ON.44
        .GLOBL  ON.70
        .GLOBL  PARRG0
        .GLOBL  PER.70
        .GLOBL  PRVMOD
        .GLOBL  PRVSUP
        .GLOBL  PS.EXP
        .GLOBL  SLEXP
        .GLOBL  SPAR0
        .GLOBL  SPDR0
        .GLOBL  SR0
        .GLOBL  SR1
        .GLOBL  SR2
        .GLOBL  SR3
        .GLOBL  SYF.SD
        .GLOBL  SY$FLG
        .GLOBL  TOPKST
        .GLOBL  UPAR0
        .GLOBL  UPAR7
        .GLOBL  UPDR0
        .GLOBL  UPDR7
        .GLOBL  .ATLLH
        .GLOBL  .CLKAD
        .GLOBL  .CLKBR
        .GLOBL  .CLKCR
        .GLOBL  .EMINT
        .GLOBL  .FP45X
        .GLOBL  .O.ODT
        .GLOBL  .SERFG
        .GLOBL  .SG.S0
        .GLOBL  .UMASK
        .GLOBL  .UMRAD
        .GLOBL  .UMR22
        .PSECT
        .PSECT  EXEC
        .PSECT  AAPWUP
        .GLOBL  CC.UMR
        .GLOBL  PRXREC
        .GLOBL  PWRXDN
        .GLOBL  PWR.UP
        .GLOBL  .PFIAD
        .GLOBL  .SG.SP
        .PSECT  AAPWUP
.SG.SP: .WORD   .SG.S0
G00002: .WORD   BOTKSP+177400
G00004: HALT
.PFIAD: HALT
PWRXDN: TST     .PFIAD
        BEQ     G00022
        MOV     .PFIAD,(SP)
G00022: MOV     R0,-(SP)
        MOV     R1,-(SP)
        MOV     R2,-(SP)
        MOV     R3,-(SP)
        MOV     R4,-(SP)
        MOV     R5,-(SP)
        BIS     #PRVMOD,PS.EXP
        TST     TOPKST+177776
        BMI     G00060
        BIC     #^RPRVSUP&PRVMOD,PS.EXP
G00060: MFPI    SP
        BIT     #ON.UM,.UMR22
        BEQ     G00134
        BITB    #ON.UM,.UMR22+1
        BNE     G00120
        MOV     #37,R4
G00106: CLR     -(SP)
        MOV     #.UMASK,-(SP)
        SOB     R4,G00106
        BR      G00134
G00120: MOV     #.UMRAD,R3
        MOV     #76,R4
G00130: MOV     (R3)+,-(SP)
        SOB     R4,G00130
G00134: MOV     #UPDR7+2,R0
        MOV     #UPAR7+2,R1
        MOV     #KP.DR7+2,R2
        MOV     #KP.AR7+2,R3
        MOV     #10,R4
G00160: MOV     -(R0),-(SP)
        MOV     -(R1),-(SP)
        MOV     -(R2),-(SP)
        MOV     -(R3),-(SP)
```

```
        SOB     R4,G00160
        MOV     .FP45X,G00004
        BEQ     G00232
        STFPS   -(SP)
        SETD
        STF     R0,-(SP)
        STF     R1,-(SP)
        STF     R2,-(SP)
        STF     R3,-(SP)
        .WORD   172404
        STF     R0,-(SP)
        .WORD   172505
        STF     R1,-(SP)
        STST    ISVFEC
G00232: MOV     SR0,ISVSR0
        MOV     SR1,ISVSR1
        MOV     SR2,ISVSR2
        MOV     @#4,-(SP)
        MOV     #.EMINT,@#4
        MOV     SLEXP,G00002
        MOV     (SP)+,@#4
        MOV     SP,.SG.SP
        HALT
PWR.UP: MOV     #4100,@#PS.EXP
        MOV     TOPKST+177776,R0
        BMI     G00336
        BIT     #CURMOD,R0
        BNE     G00344
        BIC     #^RPRVMOD,R0
        BR      G00350
G00336: MOV     #PRVMOD,R0
        BR      G00350
G00344: MOV     #PRVSUP,R0
G00350: BIS     R0,PS.EXP
        MOV     .SG.SP,SP
        MOV     #4,R5
        MOV     (R5),-(SP)
        MOV     #.EMINT,(R5)
        MOV     G00002,@#SLEXP
        MOV     #20,R0
        MOV     #PARRG0,R1
G00410: MOV     #1,(R1)+
        SOB     R0,G00410
        MOV     #177777,@#PER.70
        MOV     (SP)+,(R5)
        TST     G00004
        BEQ     G00460
        SETD
        LDF     (SP)+,R1
        .WORD   174105
        LDF     (SP)+,R0
        .WORD   174004
        LDF     (SP)+,R3
        LDF     (SP)+,R2
        LDF     (SP)+,R1
        LDF     (SP)+,R0
        LDFPS   (SP)+
G00460: MOV     #10,R4
        MOV     #KP.ARO,R3
        MOV     #KP.DRO,R2
        MOV     #UPARO,R1
        MOV     #UPDRO,R0
G00504: MOV     (SP)+,(R3)+
        MOV     (SP)+,(R2)+
        MOV     (SP)+,(R1)+
        MOV     (SP)+,(R0)+
        SOB     R4,G00504
        BITB    #SYF.SD,SYSFLG
        BEQ     G00574
        BIS     #ON.CSM!ON.SD,SR3
        MOV     #UPARO,R0
        MOV     #SPARO,R1
        MOV     #10,R2
G00550: MOV     (R0)+,(R1)+
        SOB     R2,G00550
        MOV     #UPDRO,R0
        MOV     #SPDRO,R1
        MOV     #10,R2
G00570: MOV     (R0)+,(R1)+
        SOB     R2,G00570
G00574: BISB    .UMR22,.UMR22+1
        MOV     (R5),@#BOTK[illegible]
        MOV     #G01070,(R5)
        TST     @#.UMRAD
        BICB    #ON.QB,.UMR22+1                 ; HP001 MAKE SURE NOT ON Q-BUS
        MOV     #G01124,(R5)
        BIT     #ON.UM,.UMR22
        BEQ     G00662
        MOV     #76,R4
        MOV     #.UMRAD+174,R3
G00650: MOV     (SP)+,-(R3)
        SOB     R4,G00650
        BISB    #ON.UM!ON.22,.UMR22+1
G00662: MOV     #G01134,(R5)
        MOVB    .UMR22+1,R0
        BIC     #^C<ON.UM!ON.22>,R0
        BISB    R0,SR3
G00702: MOV     @#BOTKSP,(R5)
        BIS     #1,SR0
        MTPI    SP
        MOV     (SP)+,R5
        MOV     (SP)+,R4
        MOV     (SP)+,R3
        MOV     (SP)+,R2
        MOV     (SP)+,R1
        MOV     (SP)+,R0
        BISB    #EV.PR,.SERFG
        BIT     #140000,2(SP)
        BNE     G00770
        CMP     (SP),#CRTNX1
        BCS     G00770
        CMP     (SP),#CRTNX2
        BHI     G00770
        MOV     #CRTNX1,(SP)
```

```
G00770: CMP     #100,.CLKCR
        BEQ     G01056
        MOV     R0,-(SP)
        MOV     .CLKAD,R0
        MOV     .CLKBR,2(R0)
        MOV     .CLKBR,4(R0)
        MOV     (SP)+,R0
        CMP     104,#.EMINT
        BNE     G01056
        MOV     100,104
        MOV     102,106
        MOV     #.EMINT,100
G01056: MOV     .CLKCR,@.CLKAD
        JMP     EXINTX
G01070:                                         ; HP001 ***-1
        BICB    #ON.UM!ON.44!ON.70,.UMR22+1     ; HP001 IT IS NOT AN UMR TYPE.
        BITB    #ON.UM,.UMR22
        BEQ     CC.UMR
        MOV     [illegible]P)+,172(SP)
        MOV     [illegible]P)+,172([illegible]
        ADD     [illegible]70,SP
        BR      [illegible].UMR
[illegible]124: IOT
[illegible].UMR: MOV    #G00662,(SP)
        RTI     [illegible]
[illegible]134:         [illegible]
        CLRB    [illegible]MR22+1                ; HP001 NOW SAY IT IS ON 11/40.
        MOV     #G00702,(SP)
        RTI
PRXREC: MOV     #.ATLLH,R0
G01152: MOV     (R0),R0
        CMP     R0,.ATLLH+2
        BEQ     G01200
        TST     A.TD(R0)
        BEQ     G01152
        BIS     #AF.PF,A.TF(R0)
        BR      G01152
G01200: BICB    #EV.PR,.SERFG
        BISB    #EV.SE,.SERFG
        MOV     (SP)+,R1
        MOV     (SP)+,R0
        JMP     .O.ODT
        .END
```

Writing MACRO Subroutines for FORTRAN Programs

Frank R. Borger
Michael Reese Hospital and Medical Center
Chicago, Illinois

There comes a time in every FORTRAN programmer's life when they wish they could use a few machine language instructions now and then. MACRO subroutines can be often used to advantage for various reasons:

1. Significant speed increases can be obtained by re-coding often executed routines in machine language.
2. Special functions, (byte swapping of data from foreign magnetic tapes, code conversions, parity checking, etc.) often can easily be coded in machine.
3. Accessing external page hardware registers is easier in machine.
4. Some integers (such as UIC's,) need unsigned arithmetic, a function quite hard to do in FORTRAN.

Incorporating machine language subroutines and functions into FORTRAN programs isn't hard. The information you need to know about the interface between FORTRAN and MACRO is available, just not in one place. This article is intended to be an introduction and guide to programmers who

1. Have a subroutine or function they want to code in MACRO.
2. Know enough MACRO programing to write the function in MACRO, (or want to learn enough.)
3. Need to know how to communicate between FORTRAN and MACRO.

There are four things one needs to know About the internals of FORTRAN.

1. How FORTRAN calls the Subroutine or Function.
2. How FORTRAN passes data to the MACRO Subroutine.
3. How subroutines return data to the main program.
4. How to use the traceback routines for error reporting.

The most important source of information is not in any manual. (It's in the FORTRAN compiler itself.) The "/LI:3" switch on the listing file causes the listing to include the MACRO code generated by the compiler. Much of the information you need to understand flow control and data passing can be obtained from this programmer's "Rosetta Stone." (Its also a very good tutorial for the neophyte MACRO programmer.) Let us apply the "/LI:3" switch to the following simple program:

```
PDP-11 FORTRAN-77 V4.0-1        10:44:16     3-Apr-84     Page 1
CALL.FOR;1                /F77/TR:BLOCKS/WR

          C
          C    SAMPLE OF A SUBROUTINE CALL
          C
0001              CALL SUB(A,B,C,D)
0002              END
.
.
.
PDP-11 FORTRAN-77 V4.0-1        10:44:16     3-Apr-84     Page 2
CALL.FOR;1                /F77/TR:BLOCKS/WR

                  .TITLE  .MAIN.
000000                    .PSECT  $PDATA
000000                    .WORD   4              Note 1,2,3
000002                    .WORD   A              Note 4,5
000004                    .WORD   B
000006                    .WORD   C
000010                    .WORD   D
000000                    .PSECT  $CODE1
000000                    JSR     PC,OTI$        Start of Main program
000004                    MOV     #35214,-(SP)
000010                    MOV     #-47367,R4
000014                    JSR     R4,NAM$        Note 6
                                                 ; 0001
000020                    MOV     #-1,$SEQC              Note 7
000026                    MOV     #$PDATA,R5             Note 1
000032                    JSR     PC,SUB
                                                 ; 0002
000036                    JSR     PC,EXIT$
                  .GLOBL  $OTSVA
                  .GLOBL  ORGSQ$
                  .END
```

In the above code, I have added comments (without the semicolon that MACRO expects as a separator,) to the listing. The Compiler has added the comments "; 0001", (which correspond to the line numbers from the FORTRAN listing.) When the FORTRAN program does a "JSR PC,SUB" to enter the subroutine "SUB":

1. R5 points to the argument list, in a DATA P-SECT, $PDATA.
2. The lower byte of the first word of the argument list contains the number of variables passed by the FORTRAN CALL.
3. The upper byte is undefined. (Actually, the first RSX11D compiler, old MOP (Midnight Optimized Polish) FORTRAN generated code with the Addresses of the passed

variables directly following the JSR subroutine call. Directly after the JSR was a Branch instruction to avoid executing the variable addresses, and the lower byte of the Branch instruction happened to conveniently contain the number of variables.)

4. Successive words contain the ADDRESS of the variable being passed.

5. For arrays, the address is that of the first element of the array.

6. The NAM$ subroutine puts the program name, or successive called subroutine names, into the traceback linkage for possible error reporting.

7. The variable $SEQC contains the current line number for traceback error reporting.

8. A FORTRAN 2-word floating point variable "A" results in

```
A:    .BLBK   4
```

Now lets look at a very simple subroutine:

```
PDP-11 FORTRAN=
PDP-11 FORTRAN-77 V4.0-1        16:58:16     3-Apr-84    Page 1
SUB.FOR;2               /F77/TR:BLOCKS/WR

          C
          C     SAMPLE SUBROUTINE
          C
0001            SUBROUTINE SUB(A,B,C,D)
0002            A=B+1
0003            RETURN
0004            END
.
.
.
PDP-11 FORTRAN-77 V4.0-1        16:58:16     3-Apr-84    Page 2
SUB.FOR;2               /F77/TR:BLOCKS/WR

                .TITLE  SUB
                .IDENT  03APR
000000          SUB:
000000                  CLR     -(SP)
000002                  MOV     #75012,R4
000006                  JSR     R4,NAM$
                                                ; 0002
000020                  SETF
000022                  LDF     @4(R5),F0
000026                  ADDF    #40200,F0
000032                  STF     F0,@2(R5)
000036                  RTS     PC
                .END
```

As a result of the calling routine's "JSR PC,SUB", the machine language code starts executing at the label "SUB". Although not apparent here, FORTRAN has made SUB a global variable by using the .GLOBL command. The routine:

1. Identifies itself via the NAM$ subroutine.

2. Sets the line number in $SEQC for traceback.

3. Sets the floating point mode to single precision.

4. Loads variable B into floating point register 0.

5. Adds a floating point 1 to floating point register 0.

6. Stores floating point register 0 in A

7. returns to the calling routine.

For some cases, you might want to create a FUNCTION rather than a SUBROUTINE.

```
PDP-11 FORTRAN-77 V4.0-1        16:39:54     4-May-84     Page 1
FUN.FOR;4               /F77/TR:BLOCKS/WR

          C
          C     SAMPLE FUNCTION
          C
0001            FUNCTION FUN(A,B)
0002            FUN=A+B
0003            RETURN
0004            END
.
.
.
PDP-11 FORTRAN-77 V4.0-1        16:39:54     4-May-84     Page 2
FUN.FOR;4               /F77/TR:BLOCKS/WR

                .TITLE  FUN
                .IDENT  04MAY
000000          .VREG:
000000                  .WORD   0,0
000000                  .PSECT  $CODE1
000000          FUN:
000000                  CLR     -(SP)
000002                  MOV     #24326,R4
000006                  JSR     R4,NAM$
                                                ; 0002
000012                  MOV     #-2,$SEQC
000020                  SETF
000022                  LDF     @2(R5),F0
000026                  ADDF    @4(R5),F0
000032                  STF     F0,-(SP)
000034                  MOV     (SP)+,R0
000036                  MOV     (SP)+,R1
000040                  RTS     PC
                .END
```

1. Variables are passed TO the FUNCTION routine the same way as for a regular SUBROUTINE.

2. FUNCTIONS are restricted to returning a single argument. They should be returned in the following manner.

TYPE	RESULT RETURNED IN
INTEGER*2 LOGICAL*1 LOGICAL*2	R0
INTEGER*4 LOGICAL*4	R0 -- Low order result R1 -- High order result
REAL	R0 -- High order result R1 -- Low order result
DOUBLE PRECISION	R0 -- High order result R1 -- * R2 -- * R3 -- Lowest order result
COMPLEX	R0 -- High order real result R1 -- Low order real result R2 -- High order imaginary result R3 -- Low order imaginary result

Finally, another (faster) way to access variables is to put them into a FORTRAN COMMON. Consider the following:

```
PDP-11 FORTRAN-77 V4.0-1        16:22:19    11-Apr-84   Page 1
COMMON.FOR;2          /F77/TR:BLOCKS/WR

        C
        C    SAMPLE OF A SUBROUTINE CALL
        C
0001          COMMON/BUFFER/A(10),B(10)
0002          CALL SUB
0003          END
.
.
.
PDP-11 FORTRAN-77 V4.0-1        16:22:19    11-Apr-84   Page 2
COMMON.FOR;2          /F77/TR:BLOCKS/WR

              .TITLE  .MAIN.
000000                .PSECT  BUFFER
000000        A:
000050        B:
000000                .PSECT  $PDATA
000000                .WORD   0
000000                .PSECT  $CODE1
000000                JSR     PC,OTI$
000004                MOV     #35214,-(SP)
000010                MOV     #-47367,R4
000014                JSR     R4,NAM$
                                                ; 0002
000020                MOV     #-2,$SEQC
000026                MOV     #$PDATA,R5
000032                JSR     PC,SUB
                                                ; 0003
000036                JSR     PC,EXIT$
              .GLOBL  $OTSVA
              .GLOBL  ORGSQ$
              .END
.
.
.
PDP-11 FORTRAN-77 V4.0-1        16:22:19    11-Apr-84   Page 3
COMMON.FOR;2          /F77/TR:BLOCKS/WR

PROGRAM SECTIONS
Number         Name       Size                 Attributes
  1           $CODE1  000042    17            RW,I,CON,LCL
  2           $PDATA  000002     1            RW,D,CON,LCL
  8           BUFFER  000120    40            RW,D,OVR,GBL
.
.
.
```

In the above case, the parameter list is only one word, the number of variables passed, (0.) The variables you wish to work on, (A and B) are in a separate PSECT. If you reference that same PSECT in your MACRO subroutine:

```
      .PSECT  BUFFER,RW,D,GBL,REL,OVR
A:    .BLKB   4*10.
B:    .BLKB   4*10.
      .PSECT  MYCODE
SUB::                          ;(Double colon makes SUB a GLOBAL)
.
.
.
      RTS     PC
```

The PSECT "BUFFER" in your MACRO subroutine will be mapped by the Task Builder to the FORTRAN COMMON "BUFFER". Passing addresses this way has the following traits:

1. Execution is the fastest, due to the simpler addressing.

2. Your SUBROUTINE or FUNCTION will always operate on the same Data, it doesn't have the calling flexibility of regular calls.

The following points must be remembered about Register usage across subroutine calls.

1. The Calling code must assume that all registers (in-

cluding the R5 pointer to the argument list,) will be used by the called subroutine. No registers are saved by called SUBROUTINES.

2. The Called code may use any of the registers, including R5.

3. The Calling code must assume that floating point register contents, (including status, calculation mode, etc.) will be changed.

4. The Called code may use any FPP registers, and must set calculation mode, etc. each time it is called.

5. The Stack may be used for temporary storage, but must be returned to its starting point before doing an RTS PC.

As a final point, a little more explanation may be necessary concerning the NAM$ subroutine. A programmer used to FORTRAN might ask, "If I tell the FORTRAN OTS I entered my subroutine, shouldn't I have to do a similar call to before I exit, to tell the OTS I'm leaving?" In actuality, NAM$ handles both functions, by a MACRO technique called CO-CALLING. When your subroutine ("MYSUB") identifies itself,

```
MYSUB:: MOV     #RUB,-(SP)      ;Second half of name on stack
        MOV     #RMYS,R4        ;Push first half of name
        JSR     R4$,NAM$         ;call the naming routine
```

You are now in the subroutine NAM$, with the name of your subroutine on the stack in RAD50, and the return address in R4. If one disassembles NAM$ one finds the following:

```
NAM$::
        MOV     @#$OTSV,RE       ;address of OTS temp. area to R3
        MOV     (R3)+,-(SP)      ;save a couple of things
        MOV     (R3),-(SP)
        MOV     SP,(R3)          ;remember new Stack pointer
        CLR     -(R3)
        JSR     PC,(R4)          ;co-call the caller
        MOV     @#$OTSV,R4       ;get OTS temp area address again
        MOV     (SP)+,2(R4)      ;restore a couple of things
        MOV     (SP)+,(R4)
        CMP     (SP)+,(SP)+      ;kick sub's name off the stack
        RTS     PC               ;return to caller of MYSUB
        .END
```

The JSR PC,(R4) is called a CO-CALL. The net effect of the JSR is that your subroutine is returned to after your subroutine call, but when your subroutine does a RTS PC, it returns to NAM$, not to the routine that called your subroutine. On return to NAM$, (after the JSR PC,(R4)) NAM$ gets rid of your subroutine name from the stack, does other cleaning up after itself, and then does a RTS PC which gets back to the routine that called your routine.

You should be ready for a real example by now. The following routine (from Aeronautical Research Associates of Princeton,) is a nice example of a routine to open a file by File ID. It also shows you how to find the elusive FDB for a file, given only the FORTRAN logical unit number. (Note that this routine opens the file in block mode for very fast access.) Opening by File ID is much faster since F11ACP does not have to search directories, and is a good example of how machine language can speed up FORTRAN programs.

```
        .TITLE  OPFID
        .SBTTL  OPEN ON FILE ID
;
        .MCALL  FDOF$L,OFID$R,NBOF$L,FDOP$R,FDRC$R
        FDOF$                   ;DEFINE THE FDB OFFSETS
        NBOF$                   ;DEFINE THE NAME BLOCK OFFSETS
; CALL:
;       CALL OPFID (LUN, ID, NERR)
; WHERE:
;       LUN  = FILE'S LOGICAL UNIT
;       ID   = 3 WORD FILE ID (TWO WORDS FROM FULL DIRECTORY)
;                              (3RD WORD IS 0)
;       NERR = FCS ERROR CODE IF OPEN FAILS
;
D.FDB   = 14                     ;OFFSET TO FDB FROM FFDB
FBY     = 0                      ;FIRST FREE BYTE IN LAST BLOCK
;
OPFID::
        MOV     #OPFIDN,R0       ;NAME TO TRACE BACK
        MOV     (R0)+,R4         ;FIRST HALF TO R4
        MOV     (R0)+,-(SP)      ;2ND HALF TO STACK
        JSR     R4,NAM$          ;TRACEBACK CHAIN
        TST     (R5)+            ;DITCH NUMBER OF ARGUMENTS
        MOV     @(R5),R2         ;GET LUN
        MOV     @#$OSTV,R3       ;POINT R3 AT OTS IMPURE AREA
        JSR     PC,$FCHNL        ;GET DEVICE TABLE ENTRY ADDR
        ADD     #D.FDB,R0        ;LUN'S FDB ADDR
        FDOP$R  R0,@(R5)         ;DO RUN-TIME OPEN INITIALIZATION
        FDRC$   R0,#FD.RWM       ;DO RUN-TIME RECORD-ACCESS SET
        MOV     R0,R1            ;SET UP FNB WITH FILE ID
        ADD     #F.FNB,R1        ;POINT TO FNB
        MOV     2(R5),R2         ;POINT TO FILE ID IN PARM LIST
        MOV     (R2)+,(R1)+      ;COPY 3-WORD FILE ID
        MOV     (R2)+,(R1)+
        MOV     (R2)+,(R1)+
        OFID$R  R0               ;OPEN FILE (READONLY)
        BCS     OPERR            ;BRANCH IF ERROR OCCURS
        MOV     #0,@4(R5)        ;RETURN 0 FOR SUCCESS CODE
        RTS     PC               ;RETURN TO CALLER
OPERR:  MOVB    F.ERR(R0),R2     ;ERROR TO R2 (SIGN IS EXTENDED)
        MOV     R2,@4(R5)        ;RETURN ERROR TO CALLER
        RTS     PC               ;AND RETURN
OPFIDN: .RAD50  /OPFID /         ;TRACE NAME
```

```
        .END
```

If you can understand the last example, (with the help of the IAS/RSX-11 I/O Operations Reference manual,) you pass the test with flying colors.

As a final suggestion before we close, remember that you can go the other way also, (a MACRO routine calling a FORTRAN subroutine.) One of MACRO's shortcomings is the cumbersome I/O and text formatting. (Between QIO's and the $EDMSG routine, there is a lot to learn.) Why not take advantage of FORTRAN's good I/O. Here a MACRO program calls a FORTRAN sub to print a variable.

```
        .TITLE  MACF
        .MCALL  EXIT$S
START:  MOV     #PLIST,R5       ;POINT TO LIST
        JSR     PC,PRINT        ;PRINT A VARIABLE
        EXIT$S
PLIST:  .WORD   1
        .WORD   I               ;ADDRESS OF VARIABLE
I:      .WORD   33              ;VARIABLE
        .END    START
```

And here is the FORTRAN subroutine to print the variable.

```
      SUBROUTINE PRINT(I)
      WRITE (5,100) I
  100 FORMAT (' I = ',I5)
      RETURN
      END
```

The above example can be expanded to include File I/O, Opens, Closes, etc. One major point is that if you do File I/O with a MACRO main program, the program must do a

```
JSR     PC,OTI$
```

to initialize the OTS and file management data areas. (More information is available in the Object Time System manual.)

To summarize, FORTRAN and MACRO both have their strong points and weak points. A little knowledge of how the two interact can let a programmer use the best of both. I hope this article has led you enough of the way to code your own simpler examples. Keep at it. Even the I/O Operations Reference Manual starts to make sense after a while.

REESE BASIC (The Other Basic)

Everyone knows about BASIC+2 for RSX11 and IAS, but there is another BASIC available for 11's which has some features that make it a valuable addition to any RSX/IAS operation. It's Reese Basic. It's available from DECUS or from various SIG tapes. It runs on IAS, RSX11M and VMS in 11M compatibility mode.

It's also an Interpreter rather than a compiler. (For those who may not understand the difference between an interpreter and a compiler, the basic differences are:)

1. Instead of assembling and then task building your source program, you run the BASIC operating system and tell the system to read in your program.

2. The program always stays in memory in text form.

3. The BASIC operating system effectively does a continuous compilation of your source text into actual program operations.

Although interpreters can not be as efficient as compilers, they offer unique advantages, mainly in program development and debugging:

1. Any program statement typed in from the terminal without a line number is executed immediately. With the program stopped you can examine variables, reset loop counters, etcetera.

2. Any incorrect code can be replaced by just re-typing the offending line. The program can be re-run immediately.

3. Diagnostic halts and/or variable print-out statements may be entered easily.

4. Program execution may be continued at any line, single stepped, etc.

5. A trace mode is available, whereby any goto, computed goto statements are reported.

6. Programs that have been corrected in memory can then be saved back to disk.

Program development and debugging is incredibly faster for an interpreter. As an example, assume your basic program just bombed because you made a simple error and used A(N) instead of A(M). You wish to correct this error and test the new version. Modification time for REESE BASIC compared to BASIC PLUS 2 are:

REESE BASIC	1 Type new line	
	2 RUN or CON nnn	5-10 seconds
BASIC PLUS 2	1 Edit source	10-30 seconds
	2 Compile with basic	20-40 seconds
	3 Link with TKB	3-5 minutes

The above compile and link times are for an 11/44 running IAS, with the large TKB, separate work disk, etc. With a smaller CPU, smaller TKB, one disk, the difference would be even more dramatic. If you don't think you can debug programs faster with this difference in time, your programs always work perfectly the first time. (Mine don't.) But you never get something for free, the cost is that interpreters are slow. If we compare execution for a simple tight loop we find the following:

```
 10 FOR I=1 TO 10000
 20 A=LOG(I)
 30 NEXT I
 40 EXIT
```

Reese Basic takes 15 seconds
Basic Plus 2 takes 4 seconds

A factor of 4 in speed seems like a very high price to pay. But nobody I know writes programs like the above simple test. If we compare the two basics using more day to day examples things even out. The following is an example of what we use Basic for a lot. We are searching a fairly large data base (1400 70-character fixed length records.) and wish to print any records containing a given 6-character string.

```
 10 dim a$[70]
 20 open #3, "POLOG.F84"
 30 open #4, "POLOG.SEL/WR"
 40 if end #3 then 100
 50 input line #3, a$
 60 if pos(a$,"NEWARK") < 1 then 50
 70 print #4, a$
 80 goto 50
100 close
110 exit
```

REESE BASIC	14 seconds 60% CPU usage
BASIC PLUS 2	12 seconds 52% CPU usage

We initially bought BASIC PLUS 2 because we thought we would get a large speed increase over our "home made" basic. A 10% faster operation didn't really buy us much.

As a minor digression, the above example illustrates the real strength of of a basic interpreter. It is so easy to write a quick 5 to 10 line program like the above that we do it all the time. That program was entered, saved to disk and completed execution before DEC's basic compiler finished compiling, to say nothing of TKB time.

Another thing they didn't tell us was that BASIC PLUS 2 programs are large. Incredibly large. That small program above only takes one block of storage, but the minimum BASIC PLUS 2 program task image takes up 186 blocks. I did a wild card search of our main user's disk and found just under 1700 basic programs. (That surprised me. I had no idea we had that many basic programs around.) A comparison of file storage requirements for the two basics yields:

REESE BASIC (Source text only)
Currently 1676 programs occupy 11K blocks

BASIC PLUS 2 (Minimum task image size 186 blocks)
The same 1676 programs would use 312K blocks

We have the equivalent of 5 RM05's on our system, and those disks would be pretty full if we kept every basic program as a B+2 task image. I guess DEC likes to sell RA80's.

While we are looking at relative sizes, lets compare memory requirements. At first glance, the 11K words of interperter code would at first seem to be a significant restriction on program size and/or variable storage. In fact, DEC's basic hauls in scads of RMS11 file access routines. This combined with the fact that (for IAS,) the interpreter code is a shared library, results in both Basics taking up about the same memory. DEC basic wins if only one basic is in memory, Reese basic wins if two or more are present:

REESE BASIC	Pure code library (shared)	53500	
	Minimum Program	16300	
Total for ONE program			72000
Total for TWO programs			110300
BASIC PLUS 2	Minimum task size	50200	
Total for ONE program			50200
Total for TWO programs			120400

Likewise a comparison of maximum variable storage yields about equal variable storage capabilities, (with small program sizes.) The following program was executed by incrementing the size of the dimensioned variable until an error occurred. Results for Reese Basic versus DEC basic follow:

```
10 dim a(64,102)
20 print "Max =";64*102;". floating point variables"
30 print "    =";64*102*4;". bytes"
40 exit
```

```
MCR>bas sizetest/rn         (Reese Basic)
Max =  6528 . floating point variables
    = 26112 . bytes
```

```
MCR>run sizetest$          (DEC B+2)
Max =  6720 . floating point variables
    = 26880 . bytes
```

Lets look at some more of the differences between the two basics. The first thing that is totally different is the file open command. Early in the development of Reese basic it was decided to not use the standard basic "OPEN" command syntax, but to use a format compatible with the standard DEC file name parser, including various access mode switches. Comparing the two types of open statements we see:

```
REESE BASIC    20 OPEN #3, "POLOG.F84"
               25 OPEN #4, "POLOG.SEL/WR"

BASIC PLUS 2   20 OPEN 'POLOG.F84' FOR INPUT AS FILE #3
               25 OPEN 'POLOG.SEL' FOR OUTPUT AS FILE #4
```

Reese Basic file specifications are essentially just String variables, but with several 2-character switches that provide the full range of files11 access modes:

/FX	Fixed Length File	/RO	Read Only
/RN	Random Access	/WR	Write Access (New File)
/LN:n	n = Record length (bytes)	/UP	Update Access (R,W,A)
/EN:n	goto line n on EOF	/MO	Modify Access (R,W)
/BN	Binary file	/AP	Append to existing file
/BL	Virtual Array	/SH	Shared Access

Note that since file names are string variables, one has great capability for name parsing. For example, it is very easy to check for a file extension and supply a default one if one is not there:

```
10 dim na$(25)v
20 input "Name of Input File ";na$
30 if pos(na$,".") < 1 then let na$=na$+".DAT"
40 open  3, na$+"/RO/SH"
```

All calculations are done using 2-word floating point format, (and in fact require a Floating Point unit, or use of a floating point emulator.) Variables are stored however in several different formats, both to save space and to provide for better mapping to virtual files. The following types are supported:

	A&	8-bit Integer
	B1%	16-bit Integer
	CC	32-bit floating point
DIM	A[n,n]	Arrays may be Up to 2 dimensions
DIM	A$[n]	Fixed Length String
DIM	A$[n]V	Variable Length String
DIM	A$[n]V(x,y)	Dimensioned String

Reese basic also supports virtual array storage. Special forms of the DIM statement and the OPEN statement are used. These allow one to directly access any file. The following example shows how to access any word of a 400 block file. (Note that for virtual arrays, the first unit of any array is N(0)).

```
120 dim #4,DD%(399,255)
.
204 open #4,"[11,17]TEST.DAT/BL/MO"
.
300  ! Print octal value
305 print oct$(dd%(bl,by))
```

The above example prints the contents of offset by into block bl of file test.dat.

Reese Basic has a full set of standard string functions, (one of the strong points of any basic,) with some added ones that make a programmers life easier.

INX/POS	Position of substring in main string
LEN	Length of a variable length string
SEG$	Substring from main string
SBS$	Substring from main string
RIGHT	Right-most N characters (not N thru end)
LEFT	Left-most N characters
MID	Same as SBS$
PIECE$	Substring between ascii dividers
LTR$	Leading blank trim
TRM$	Trailing blank trim
RJS$	Right justify string
LJS$	Left justify string
SPACE$	String of n spaces
STRING$	String of n characters

It has a large set of internal and I/O conversion routines:

VAL	String to numeric conversion
OCT	String to octal conversion
AR5	Ascii to RAD50 conversion
ASC	Ascii character to numeric conversion
CHR$	Numeric to ascii character conversion
STR$	Numeric to string conversion
OCT$	Octal to Ascii conversion (unsigned)
OCS$	Octal to Ascii conversion (signed)
R5A$	Rad50 to Ascii conversion.
FRMT$	Print using, FORTRAN FORMAT style
TAB	Tabulate to position n

Some very nice DEC system based functions

NRC	Number of records in fixed length file
ERR	Number of last error
ERL	Line number at last error
FCS	Last FCS related error number
COR	Free space available in bytes
DAT$	Date in format MM/DD/YY from day-of-century
DDAT$	Date in format DD-MMM-YY from day-of-century
TIM$	Time-of-day from seconds
DCEN	Returns day-of-century from date string
SEC	Returns seconds from time-of-day

And some non standard commands also tailored to DEC system:

STEP [Line number]	Single step program
CON [line number]	Continue after stop or error
ON ERROR GOTO	Transfer if error occurs
BREAK	Print using write-pass-all
SET PROMPT on/off	Turn on/off question mark
SET TRACE on/off	Turn trace on/off
SET UPPER on/off	Turn case conversion on/off
SET READ-PASS-ALL	Turn read-pass-all on/off
SET WRITE-PASS-ALL	Turn write-pass-all on/off
SLEEP n units	Do mark-time
WAIT n units	Terminal read with time-out
TRACE on/off	Turn trace on/off

Reese basic contains a simple but effective method for doing various types of question processing. Options include single and multiple line questions, single and multiple choice answers, and automatic linking to further questions based upon question answers. The text of the questions themselves is contained in a special macro source file. The command:

```
OPEN LIBRARY #N, FILENAME
```

Opens a macro question source file containing question text. Questions are of three basic types:

```
.MACRO   MULTXT  0                 Multiple line question
This is an example of a
multiple line question.
.ENDM

.MACRO  SINGLE  1                  Single line question
Type your name, (last,first)
.ENDM

.MACRO  MULCHO  2                  Multiple-choice question
Select your state of mind
GOOD\MULCH1
FAIR\MULCH2
TGIF\MULCH3
.ENDM
```

Various routines are available to automatically output these questions and return ascii (or numeric for multiple choice questions) answers. The basic question display routines are as follows:

Three basic forms of question processing calls exist:

x=M Reference question by NAME
x=C Reference by link to previous question
x=R Reference by last question accessed

CALL "xDIS"(LUN,STAT,NAME)	Display Question
CALL "xPOS"(LUN,STAT,NAME)	Point to question
CALL "xQTXT"(LUN,STAT,NAME,STRING)	Returns question text
CALL "xATXT"(LUN,STAT,NAME,STRING)	Returns answer text
CALL "xQNAM"(LUN,STAT,NAME,STRING)	Returns question name

The above question handling routines are just one of a series of machine language subroutines that can be loaded on command into the program area of the basic interpreter. Although the mechanism for passing arguments from basic to the machine language routine is somewhat complicated, (since these routines must do what TKB does when it links a program,) machine language routines are just as flexible as they are with DEC basic, Fortran, etc. Machine routines are handled by three special commands:

LOAD "PROGNA"	Loads macro routine
UNLOAD	Unloads all routines
CALL "PROGNA"(arg,arg,arg)	Calls macro routine

Some of the more noteworthy loadable routines include:

SPAWN	spawns an MCR command
BINSRC	Fast binary search
DIRECT	Issue any system directive
EXITST	Exit with status
INSTAL	Install a task
LOWCAS	Convert ascii to lower case
REVSTR	gnirts txet a esrever
UPPCAS	Convert ascii to upper case

The spawn directive has proved to be a very powerful one. With it, BASIC has the capability of acting as a alternative to MCR. Many of our less sophisticated users interact with a menu program written in BASIC, and never see or need to use MCR or DCL. The combination of math capabilities and string manipulation with spawning capability provides a much greater capability than batch, indirect MCR or DCL.

That's about it for an intro to Reese Basic. We use it a lot, and it seems (at least to us,) to be really what BASIC should be.

1. An easy to use programming language.

2. A good system for learning on, letting new users correct their errors quickly.

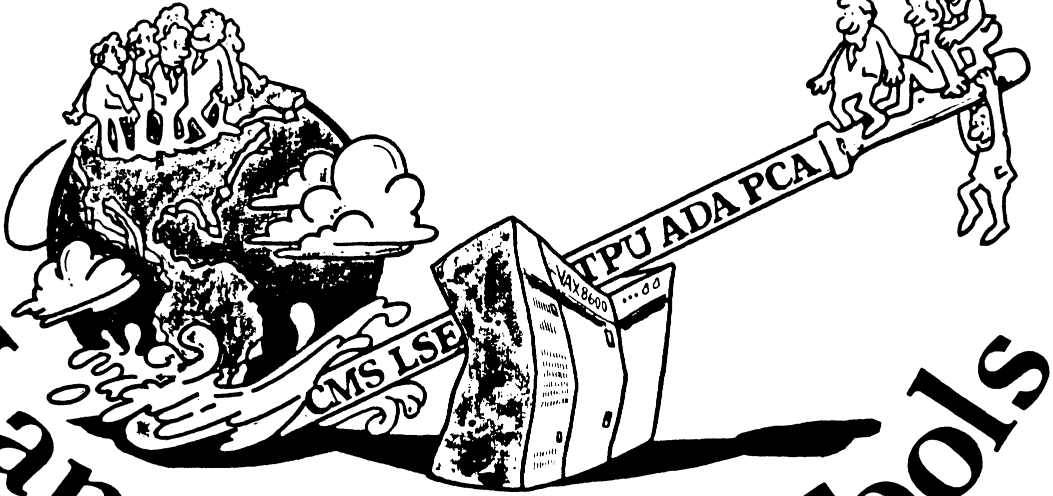
3. A language with strong text manipulation commands.

4. A simple way to quickly write and debug small, simple programs.

If you are interested, the next question is "Where do I get it." Its in the DECUS library as 11-SP-72. And is on several of the Sig tapes, (the latest version is the Spring 84 RSX tape.)

What about the future? There is a new release coming out at the next DECUS. It has a few minor enhancements, and corrects one major shortcoming. Lower case text could only be read with the "INPUT LINE" statement. It also could not be included in statements of the form A$="Test" or INPUT "Enter your name ";NA$. The new version corrects this significant shortcoming.

Leverage


TPU ADA PCA
CMS LSE
VAX8600
...00

Languages & Tools

L&T

STEERING COMMITTEE MEMBERS
LANGUAGES AND TOOLS SIG

Susan Abercrombie Ventrex Laboratories 217 Read Street Portland, ME 04103	**Library Rep** (207) 773-7231
Bob Awde General Mills 9000 Plymouth Ave N Minneapolis, MN 55427	**Steering Committee** (612) 540-4432
Mark Bartelt HSC - Research Development Ctr 555 University Avenue Toronto, Ontario, Canada M5G 1X8	**UNISIG Interface** (416)598-5955
Doug Bohrer Bohrer & Co. 903 Ridge Rd. Suite 3 Wilmette, IL 60091	**Publications Committee** (312) 251-9449
Gordon Brimble Bldg 180 Labs Area Defence Research Centre Box 2151 GPO Adelaide, S.A. Australia 5001	**Australian L&T Interface** (61)(8)259-6119 (office)
Barb Chase Hughes Aircraft P O Box 92426 Bldg R1 MSC327 Los Angeles, CA 90009	**Human Interfaces Coordinator** (213) 606-1601 (work)
Earl Cory Cory Computer Systems 366 North Nueve Court Camarillo, CA 93010	**Symposium Coordinator** (818) 706-5385 (work)
Jack Davis Philips Home Interactive Systems 1111 North Shore Drive Knoxville, TN 37919	**Modula II Coordinator** (615) 558-5206 (work) (615) 588-5800 (switchbd)

Celeste LaRock 110 Spit Brook Rd ZK02-3/Q08 Nashua, NH 03062	**DEC Counterpart**
Jim Livingston Measurex Corporation 1 Results Way Cupertino, CA 95014	**Past Chair** (408) 255-1500 X4468
Dave Martin Hughes Aircraft Company PO Box 92426 Bldg R1, MS C320 Los Angeles, CA 90009	**Tape Librarian** **STUG Interface** (213) 648-9927
Shava Nerad Systems Alternatives 43 State St Montpelier, VT 05602	**Steering Committee** (802) 229-0823
Al Rizzuto EMC Control, Inc PO Box 242 Cockeysville, MD 21030	**Wishlist Coordinator** (301) 628-8167 (work) (717) 456-5014 (recorder)
Don Rosenthal Space Telescope Science Institute Homewood Campus Baltimore, MD 21218	**LISP/AI Coordinator** (301) 338-4844 (work)
Tony Scandora Argonne National Laboratory CMT 205 Argonne, Illinois 60439	**RSX Interface** (312) 972-7541
Bill Segal Digital Equipment Corp. 110 Spit Brook Rd. ZK01 Nashua, NH 03062	**Counterpart Emeritus** (603) 881-1263

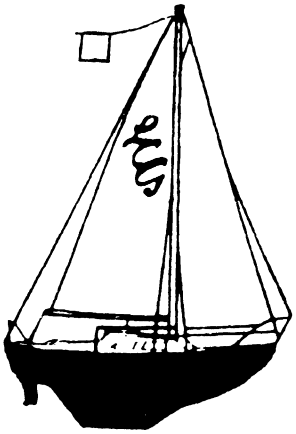
Jim Flatten Ames Lab 304 Metallurgy Ames, IA 50001	**GAPSIG Interface** (515) 294-4823 (work)
Alan Folsom, Jr. Fischer & Porter Co. E. County Line Rd. Warminster, PA 18974	**Newsletter Editor** (215) 674-7154 (work)
Bob Gable Lear Siegler, Instrument Division 4141 Eastern SE MS 121 Grand Rapids, MI 49508	**Ada Coordinator** (616) 241-8273
Dorothy Geiger Wollongong Logistics Group 49 Showers Drive # 451 Mountain View, CA 94040	**Intersig Coordinator** (415) 962-7160
Bernd Gliss Max-Planck-Institute Heisenbergstraße 1 7000 Stuttgart 80, W. Germany	**European Methods, L&T Interface** (711) 686-0251 (work)
Keith Hare JCC PO Box 381 128 West Broadway Granville, Ohio 43023	**DMS & DTR Liaison** (614) 587-0157 (work)
Howard Holcombe RCA Front & Cooper St. Camden, NJ 08055	**DEC Personnel Coordinator** (609) 338-4946 (work)
Kathy Hornbach Lear Siegler/Instrument Division 4141 Eastern SE MS 121 Grand Rapids, MI 49508	**Chair** **Productivity Tools Coordinator** **Pre-Symposium Seminar Coordinator** (616) 241-8800
Mark Katz GTE Govt Systems 100 First Ave. Waltham, MA 02154	**Session Notes Editor** (617) 466-3437

Kathy Tamer Rockwell International 1840 Nasa Rd./MS ZC01 Houston TX 77058	**Ada Packages Project** (713) 333-0827 (work)
Pat VanMunn Measurex Inc. One Results Way Cupertino, CA 95014	**Methods Coordinator** **PSS Committee Representative** (408) 255-1500
Jay Wiley Bechtel Power Corp 12400 East Imperial Highway Norwalk, CA 90650	**Standards Coordinator** **Fortran Coordinator** (213) 807-4016 (work)
JR Westmoreland Custom Software Products 6748 Acoma Rd Midvale, UT 84047	**Assistant to the Chair** **C Coordinator** **Commercialism Task Force** **TeX/LaTeXCoordinator** (801) 535-4784 (work)
Melodee Westmoreland Custom Software Products 6748 Acoma Rd Midvale, UT 84047	**Recording Secretary** (801) 533-2350 (work)
Sam Whidden American Mathematical Society 201 Charles St PO Box 6248 Providence, RI 02940	**36 bit Coordinator** **Store Liaison** (401) 272-9500 (work)
Ed Whipple Lawrence Berkeley Labs University of California Berkeley, CA 94720	**CMS/MMS Coordinator** **Session Chair Coordinator** (415) 486-7167 (work)
Louise Wholey Measurex Corp One Results Way Cupertino, CA 95014	**VMS Interface** (408) 255-1500 X4452 (work)
Jim Wilson QZ Division PO Box 88 Terre Haute, IN 47808	**Commercial Languages Interface** (812) 299-2121 X271 (work)

DECUS

LARGE SYSTEMS SIG
DECUS
U.S. CHAPTER

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY



AT LARGE

The Newsletter of the Large Systems SIG

LS

SIG STEERING COMMITTEE

SIG Chairperson
Leslie Maltz
Stevens Institute of Tech.
Computer Center
Hoboken, NJ 07030
(201)420-5478; BITNET: LMALTZ@SITVXB;
ARPANET: SIT.MALTZ@CU20B

Symposium Coordinator
Robert C. McQueen
Stevens Institute of Technology
Computer Center
Hoboken, NJ
(201)420-5454; BITNET: RMCQUEEN@SITVXB;
ARPANET: SIT.MCQUEEN@CU20B

Newsletter Editor
Michael Joy
First Church of Christ, Scientist
Christian Science Center A30
Boston, MA 02115
(617)262-2300 x3903

Menu Coordinator
Charles R.T. Bacon
National Institutes of Health
Building 12 B Room 2N207
Bethesda, MD 20205
(301)496-4824

Hardware Coordinator
Clive Dawson
Microelectronics & Computer Tech Corp.
9430 Research Blvd.; Echelon Bldg. #1, Suite 200
Austin, TX 78759
(512)343-0860; ARPANET/CSNET: CLIVE@MCC

Languages Coordinator
David Edwards
DPEX, Inc.
240 Pamela Drive, Suite #1
Mountain View, CA 94040
(415)965-3739

Systems Software Coordinator
Betsy Ramsey
American Mathematical Society
P.O. Box 6248
Providence, RI 02940
(410)272-9500 x295

Special Projects Coordinator
E.F. Berkley Shands
Emerson Electric Co. m/s 4451
8100 W. Florissant
St. Louis, MO 63136
(314)553-4451

DEC Counterparts
Dave Braithwaite
Digital Equipment Corporation
Marlboro, MA

Jack Buckley
Digital Equipment Corporation
Marlboro, MA

Reed Powell
Digital Equipment Corporation
Marlboro, MA

MUMPS

MMP

MUMPS SIG STEERING COMMITTEE

Chairman
Mark Berryman
Plessey Peripheral Systems
Irvine, CA

Symposium Coordinator
Chris Richardson
Computer Sciences Corp.
Ridgecrest, CA

Communications Rep.
Mark Hyde
Advanced Computing Services
DeWitt, NY

Newsletter Editor
Janet Berryman
Plessey Systems, Inc.
Irvine, Ca

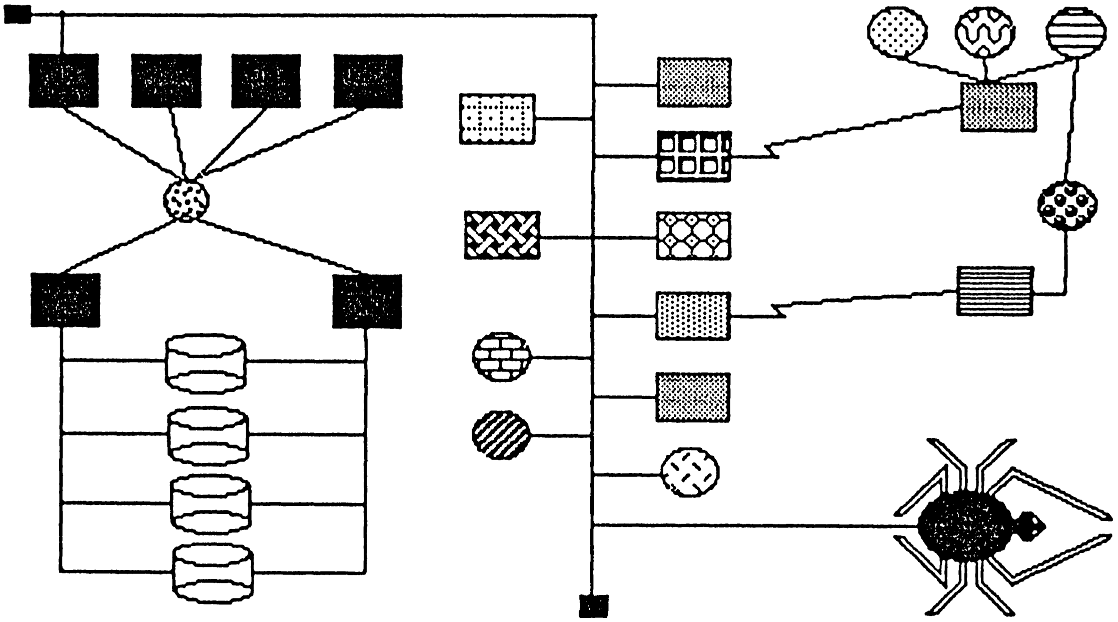
VAX Liaison
Coyett A.J. Dese
VA DM&S Verification & Dev. Ctr.
San Francisco, CA

Digital Counterparts
Beatrice Walther
Digital Equipment Corp.
Marlboro, MA

Diane Brown
Digital Equipment Corporation
Marlboro, MA

NETwords

The NETWORKS SIG Newsletter



NTW

Networks SIG Steering Committee

Bill Brindley
Chairman
Naval Security Group Command
(202) 282-0527

Jim Ebright
Communications Coordinator
Software Results Corporation
(614) 421-2094

Vickie Hancock
Newsletter Editor
(214) 495-7353

Sandy Traylor
Symposia Coordinator
Target Systems, Inc.
(714) 921-0112

Bill Hancock
Technology/
Standards Coordinator
(214) 495-7353

Carole Greenfield
DEC Counterpart
Digital Equipment Corporation

The Networks Special Interest Group (SIG) is one of 25 SIG's within in Digital Equipment Computer User's Society (DECUS). The main purpose of the Networks SIG is to promulgate information concerning the use, development, and standardization of network products that function or involve Digital Equipment Corporation systems. Additional functions of the SIG include the coordination and scheduling of symposia sessions, providing methods for free-flow communications, publication of the Networks SIG newsletter NETWords, participation in domestic and international standards committees, input to Digital for new products and corrections to existing products, promotion of working groups for special network needs and topics, and many, many other functions.

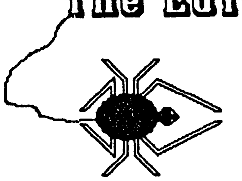
The Networks SIG Steering Committee invites you to participate in the Networks SIG. There are many ways that you can help the Networks SIG. Some of those include chairing sessions at symposium, participation in the various Networks SIG working groups, participation in special research projects, and others. If you are interested in devoting your time and expertise, contact any of the steering committee members.

DECUS is run entirely by volunteer leadership. Help us make DECUS and the Networks SIG better - take an active part in **your** SIG!

In This Issue:

Networks SIG Steering Committee NTW-2
Index NTW-3
The Editor's Cobwebb NTW-4
  Vickie Hancock, NETWords Editor
ANSI X3S3.7/4 Meeting Report (*71) NTW-5
  Joel Snyder, X3S3.7 Representative
ANSI X3S3.7 Meeting Report (*72) NTW-7
  Joel Snyder, X3S3.7 Representative
New Ways of Communicating with IBM NTW-8
  David Korf, Digital Equipment Corporation
New Alternatives in Local Network Technologies NTW-12
  Rod Flakes, Digital Equipment Corporation
Datagrams NTW-19

The Editor's Cobwebb



By Vickie Hancock

It's hard to believe this is the last issue of the newsletter for the 1985-86 year. Yep, folks, a whole year has gone by! It's amazing how time flies when you are having fun!

In this issue, we have a couple of ANSI reports from our X3S3 standards representative, Joel Snyder. Thanks, Joel! We appreciate the time and effort you spend in attending these meetings and keeping us informed. Keep up the good work!

In addition to the standards reports, we have a user's viewpoint on the new ways of communicating with IBM. Last, but certainly not least, Roderick Flakes shows us some new alternatives in local network technologies. I believe you will find both of these articles quite interesting and informative. Thanks, David and Roderick!

Speaking of thank yous, I would like to thank all of you who sent in submissions for the newsletter for the past year. I appreciate your efforts and the time spent on these articles.

See ya next issue!!

VH

ANSI X3S3.7 (PDNs) 71st Meeting
November 18-22, 1985
Westboro, MA

Joel Snyder
Networks SIG X3S3.7 Representative

I attended the 71st meeting of ANSI task group X3S3.7 on Public Data Networks in Westboro, MA from 18-November to 22-November.

This was a somewhat unusual meeting in that almost no output documents were created. Almost the entire time of the meeting was spent on reports and analyses of other standards groups and their evolving standards. Many results of meetings we had prepared input for were available, and a lot of nearly-finished work was handed out for final review.

Some points which received extensive discussion are detailed below.

****
* X.32 *
****

X.32 — "Interface between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) for terminals operating in the packet-mode and accessing a packet switched public data network through a public switched telephone network or a circuit switched public data network," also known as Dial-up X.25

This CCITT Recommendation is in the very final stages of preparation before going into the CCITT as a ballot for adoption. X.32 has received a lot of attention from X3S3.7.

At the Study Group D (Department of State meeting to determine US vote at CCITT meetings) meeting in September, many of the X3S3.7 papers were forwarded to CCITT without further comment. These included documents dealing with limitations on the X.32 identification protocol, refinements to security of X.32, NUI (Network User Identification) papers, limiting unauthorized use of network resources, and dial-out procedures for X.32.

The X.32 Rapporteur meeting resolved most of the remaining issues in X.32. Many of the US contributions were accepted and incorporated in Recommendation X.32 at this meeting.

The open issues remaining in X.32 are:

1) Link level address assignment. Two standards, T.70 and X.25 provide incompatible answers to the question of link-level addresses in X.32. The question of which to use is still open. The US will propose that it be a national matter.
2) LAPX, or half-duplex, in X.32. The US seems to be adamant about getting half-duplex removed from X.32. The editor of X.32 is going to propose another X-series recommendation which would contain all the half-duplex (LAPX) text from X.32. I don't think this is a good idea at all at this stage, and may very well jeporadize the progression of X.32 at a CCITT Recommendation. The French are equally stubborn on this point (since they have an existing implementation).
3) A US contribution on even/odd numbers in identification protocols had technical problems. Resolution of signature protocols is important. Members of X3S3.7 will draft new proposals to help resolve this issue.

4) Identification protocol in general. There are some very weak spots in the security of the ID protocol. The documentation needs some firming up.

****
* X.75 *
****

A Study Group 7 Working Party 3 on Interworking meeting was held which firmed up issues in the X.75 and X.300 recommendations. X.300 was broken into three recommendations, X.300, X.301, and X.305.

In X.75, utilities were categorized as "International Mandatory," "International Optional," and "National." Work on the utility marker which separates X.75 from non-X.75 utilities was done. The question of transit delay signalling -- how long does data take to go through a network -- was debated, and a series of questions were posed. A discussion of RPOA (registered private operating authority) selection over international boundaries was brought up.

Some issues which were discussed at this meeting were:
1) X.75 use in ISDN interworking situations -- ISDN-PSPDN and ISDN-ISDN interworking. Some changes to X.75, particularly in the areas of numbering plans, may be necessary.
2) Transit delay. Although agreements were made at the SG7 meeting, many issues are unclear. How will transit delay be defined? The existing recommendations X.213 and X.135 seem to have differing definitions of transit delay. The existing X.75 proposal is to signal transit delay as a static value based on busy-hour conditions worst case across a network. This seems to be a misleading and useless implementation, and contrary to the reason for existence of transit delay.
   Additionally, an interim procedure was created to deal with networks who do not implement transit delay. The US will probably have no comment on this.
3) Hunt group. US networks wish to use hunt group-like implementations to spread international calls rather than the existing multi-link procedures. Some text was proposed for inclusion to CCITT.

****
* X.300 *
****

The X.300 recommendations (General Principles and Arrangements for Interworking) are still being studied and changed.

At the X.300 Rapporteur's Group meeting, the term "OSI Subnetwork" was examined and redefined. Also, a new recommendation X.305 was drafted on the last day. The editor of X.300 and X.301 is interested in sending these documents through the balloting procedure as quickly as possible. The X3S3.7 ad-hoc group studying X.300 found many flaws with the document and prepared a position suggesting that accelerated procedures on X.300 and X.301 would not be acceptable to the United States.

Other topics which were discussed at this meeting included ISO 8348 on Quality of Service definitions, ISO 8878 on Network Service Access points, the question of routing (joint with X3S3.3), connection-less mode within X.25, reference numbers, X.25 conformance testing, OSINET, ISDN, and X3.100 (ANSI X.25).

Members with questions on DECUS participation in ANSI networks task groups are invited to contact Joel Snyder, University of Arizona, Department of Management Information Systems, Tucson, Arizona, 85721. For questions on DECUS participation in other standards committees within ANSI, contact Bill Hancock

---

ANSI X3S3.7 (PDNs) 72nd Meeting
January 6-10, 1986
Phoenix, AZ

Joel Snyder
Networks SIG X3S3.7 Representative

I represented DECUS at the 72nd meeting of ANSI task group X3S3.7, Public Data Networks, in Phoenix, AZ, January 6-10, 1986. Also included in this report is information on the joint meeting of X3S3.7 and X3S3.3.

Some additional progress was made in the area of the US ANSI standard X3.100, the adoption of CCITT X.25. Almost all of the details have been ironed out of this document, with only a few points still under discussion. It is hoped that the March meeting of TG7 will provide a document ready to go to ANSI ballot.

The greatest difficulty in TG7 right now lies in the area of numbering and addressing. The question of how current numbering plans (such as E.163, E.164, X.121) will work together to form globally addressable entities is still open to a lot of discussion. Additional confusion because of the coming "ISDN Era" is causing even greater problems in understanding and resolving basic conceptual issues.

Some New Work Items were submitted to the ISO counterpart of TG7 for consideration. Most of these were on ISDN-related matters. Because these projects duplicate existing areas of concern within ISO, ANSI will ask that they not be brought into the program of work for TG7s parallel subcommittee.

The question of protocol identification, and the somewhat related question of higher layer compatibility elements was debated for a while. It is apparent that the membership of TG7 disagrees within itself on the fundamental questions which are raised by these issues. In addition, the concept of known addresses pointing to known, universal services received debate. These issues, related to the numbering and addressing questions mentioned above, are still very active in international standards bodies.

An ad-hoc committee on ISO DP 8880, which is a "how to use it" manual on Network layer standards, made good progress in providing a US position on this standard. Hopefully, documentation like this will make implementations of standards easier.

Work on packet mode services within ISDN is also continuing. The US position will be that the X.25-like service should be, in fact, X.25.

Some papers on routing were presented and forwarded onto appropriate CCITT committees. These dealt mostly with minor points; the routing work being done in CCITT is still far from complete. Papers on CCITT Recommendations X.121, X.32, X.25, and X.75 were also discussed at the meeting.

Much work is being done on X.25 conformance testing under the ISO umbrella. As this work matures, TG7 has suggested that a parallel project begin in CCITT. Work on principles, actual tests, and formats of implementation descriptions and test results continues both in TG7 meetings and in ad-hoc groups. This is a difficult set of standards to write; some of the basic philosophies of standard-writing need to be discussed when conformance testing to any one standard is developed.

NEW WAYS OF COMMUNICATING WITH IBM: A USER VIEW

David Korf
Networks and Communications
Digital Equipment Corporation, MK02-1/C11
Continental Boulevard
Merrimack, NH 03054
603-884-4427

ABSTRACT

Providing paths for communicating with IBM is an important challenge for major system and network vendors. Traditionally, in IBM-to-other-vendor communication, work done locally on the other-vendor systems has then been transferred in batch-equivalent mode to an IBM mainframe for data base update or inquiry.

Recent DISOSS and HCF announcements, however, have made interactive reverse communication through IBM systems to other-vendor systems more practical. As part of this session, the author will describe key methods of other-vendor communication with IBM, including new methods which permit network-wide file transfer across the heterogeneous networks and which allow IBM terminal users access to the facilities of non-IBM systems.

Although top-level executives and MIS Directors play a key role in the selection of computer equipment for an organization, it is the end-user who must, eventually, live with (and work with) the equipment decided upon.

Unfortunately, user needs are easier to state than to implement. Users want fast and easy access, from the desktop, to as much information as possible. And they want the information to be easy to work with, easy to incorporate into forms, memos, and reports, and easy to send to others in the organization. In a computing environment composed of nothing but one vendor's computer equipment, the compatibility between different elements of the system usually makes a smooth flow of information relatively easy to achieve.

However, in the real world, a computing environment based on a single vendor's equipment is extremely rare. A single computer vendor cannot meet the total computing needs within a company and is therefore both functionally and financially inefficient. It does not fit well with the real computing needs of individuals working in different parts of an organization.

A more common arrangement is a multivendor "shop;" an organization, for example, and departmental/divisional computers from Digital Equipment Corporation distributed throught the organization, with IBM mainframe computers based in an MIS Department or corporate data processing center. This arrangment optimizes the information processing capabilities of both brands of hardware and software, but it can present new hurdles to be overcome by users who need access to information residing on both the Digital equipment *and* IBM.

HELPING DIFFERENT COMPUTER SYSTEMS COMMUNICATE

For years in corporations there evolved two worlds of information systems. The first of the two worlds was that of Data Processing (DP). It is hierarchcial in structure and is often based on IBM. The second world or style of computing was traditionally used in the technical areas which needed to distribute computing power and resources to the individual. This world of distributing the computer power to where it is needed can be characterized by Digital's Style of Computing.

Initially there was no need to access and exchange information between the two worlds. As business needs evolved the need to access information in a timely fashion, regardless of where that information resided, became the key for successful business operation. Reconciling the differences that exist between different computing environments, such as IBM and Digital, is not an easy task. There are two main strategies for accomplishing an interface between IBM and non-IBM vendors. They are 1) co-existence, or 2) a complementary computing strategy. The first allows for single point interfaces; the second acknowledges that there are strengths in both and attempts to bridge the gap between the two computing worlds. For more than 25 years, Digital like other vendors, has been working towards a complementary computing strategy by introducing communications products, both hardware and software, that embody this commitment.

Depending on the level of cooperation needed, there are IBM Internet products from other vendors and Digital that span the entire range of user needs. At the first level there is protocol emulating software, which allows terminals on systems to function as IBM 3270-class video displays, and Remote Job Entry (RJE) software designed to permit Digital systems to function as general-purpose remote batch and interactive workstations to IBM hosts.

These products are useful for tasks such as database inquiries and updates, forms-based transaction processing, and program compilation. In these instances, software running on the non-IBM system provides users of that system with workstation capabilities normally provided only by IBM's Remote Batch Terminals or a HASP workstation.

Additionally Digital's VAXTM BTS (Bisynch Terminal Support) lets users of IBM-type terminals (IBM 3270s, or Teletype's 4540s) connect directly to VAX and communicate interactively with the VAX family of processors, as well as with an IBM host. The BTS software makes this possible by causing the "foreign" block-mode synchronous terminals to appear as Digital hardcopy terminals to the VMSTM operating system on a VAX.

As a result, the IBM-type terminal user on a Digital system, in addition to running IBM mainframes applications, can access Digital's VMS operating system and utilities, and layered software applications supported by VMS, including office automation software with electronic mail and calendar management capabilities.

THE DECnet/SNA GATEWAY

The preceding products, while a vital part of non-IBM computer vendors IBM interconnect strategy, are only part of an overall complementary computing solution. The previous products provide users with point-to-point protocol emulation of IBM and non-IBM terminals so that each workstation can communicate with the other host computer system. This is a satisfactory solution when the need to communicate with the "foreign" system is limited to a relatively small number of users in the organization.

However, when frequent, transparent, and bidirectional communications between IBM and non-IBM systems for large numbers of users is needed, a network-to-network solution is needed. For Digital this network-to-network solution is the DECnet™ Gateway to SNA.

The DECnet/SNA Gateway is at the heart of Digital's IBM Internet strategy. This product—a dedicated, intelligent communications processor with supporting software—links the IBM and Digital "worlds." The DECnet/SNA Gateway enables communications between the two vendors' network environments, DNA (Digital Network Architecture) and IBM's SNA (Systems Network Architecture).

In Digital's case the DECnet/SNA Gateway, individual users of either a Digital or IBM workstation can take advantage of the interactive computing, ease-of-use features, wide choice of applications, and interdepartmental communications of Digital systems, while also maintaining access to the large databases and batch/transaction processing strengths provided by IBM mainframes.

All this helps make life a lot easier for end-users, who do not have to modify their customary computer procedures. They need not be aware of any technical differences between the two computer systems—to get productive work done.

When a user on a DECnet node initiates, say, a CICS (Customer Information Control System) transaction to the IBM host, no special command language or unfamiliar procedures are necessary to connect from the Digital system to the IBM host. Similarly, when an IBM workstation user wishes to access DECnet to send a mail message to a DECmail™ user, the Digital network is accessed just as though it were another IBM application.

The key components of the Digital to IBM connection protocol software—Ethernet and DECnet hardware and software, and the DECnet/SNA Gateway—are implemented entirely on the "Digital side of the fence." IBM interconnect functions are shared between the VAX host and the DECnet/SNA Gateway server. That is, some software runs on the Digital host, another portion resides on the gateway, but no hardware or software from Digital needs to be added to the IBM system to provide the bridge between the two worlds of computing.

THE IMPORTANCE OF NETWORK STANDARDS

The strength of a computer vendor's network strategy—and the key to its ability to connect to other vendor's systems—lies in its adherence to a well-defined framework.

The open systems interconnect model (OSI), developed by the International Standards Organization (ISO) in 1981 as a standard for the computer industry, is the key to a true multivendor network. As an architecture, OSI defines the standard communications protocols, interfaces, and functions that enable computer hardware, operating systems, and applications software running on different systems in the network to share data and resources, and exchange information. DNA, Digital's Network Architecture, closely resembles the International Standards Organization's open systems interconnect model.

Because DNA conforms to the "multi-layered" ISO standards Digital can separate network applications from the communications technology on which the applications run. Thus, as the network grows, and new, more effective networking technologies are developed (packet-switched networks, microwave technology, satellite links, and the like), changes can be accommodated without any disruption of service or "relearning" by the end-user.

THE BEST GETS BETTER

Digital recently introduced three new IBM Internet products that bring the goal of effective yet effortless complementary computing between the Digital and IBM worlds even closer. These products are: DECnet/SNA VMS DISOSSTM Document Exchange Facility (DDXF), DECnet/SNA VMS Distributed Host Command Facility (DHCF), and DECnet/SNA VMS Printer Emulator (PrE).

These products represent an evolution of products to connect bidirectionally to IBM. They provide users of IBM systems with new ways to increase their productivity through the use of, and exchange of, information with Digital systems. Previous IBM Internet products from Digital and other vendors have focused on providing access to IBM mainframes. This has changed; there is now true bidirectional information exchange between the two computing worlds, including the ability to initiate transactions into Digital systems from the IBM world.

The DISOSS (Distributed Office Support System) environment is, according to IBM, a key element of its office automation strategy. With DECnet/SNA VMS DDXF, Digital system users and IBM system users can now participate in a corporate office network. The product makes its easy for users to receive, edit, delete, and distribute documents within the DISOSS network, using document processing, storage, and transmission capabilities provided by both the VAX/VMS and DISOSS/370 host systems.

These capabilities include, on the Digital side, the benefits of sophisticated word processing, electronic mail, and office automation provided by products such as WPS-PLUS™/VMS word processing, DECmail and VAXmail™ electronic mail, and the ALL-IN-1™ integrated office information system; and on the IBM side, the centralized document library and distribution services provided by the DISOSS/370 network.

DECnet/SNA VMS DHCF (Distributed Host Command Facility) allows IBM 3270-class displays connected to an IBM host in an SNA network to access VAX/VMS systems through the DECnet/SNA Gateway. More than providing IBM users with a simple window into the Digital world and its capabilities, this product allows an IBM network manager to manage and control both the IBM and Digital networks from an IBM display in an SNA network: an important feature for the centralized network control concept promoted by IBM. It also allows IBM users to access Digital's electronic mail systems, run office automation packages, perform database queries, and even do program development on a VAX/VMS system.

The third product in the trio, DECnet/SNA VMS PrE (Printer Emulator), allows data slated for printing on an IBM printer to be sent, instead, to a Digital system and then be edited, mailed, printed, or viewed on a video terminal. Also, since IBM printers are often located inconveniently for the end-user (i.e., in a central computer center rather than distributed throughout the organization), this capability can make life easier for the end-user who wants quick and easy access to information residing on the central IBM mainframe.

DIFFERENT APPROACHES TO PROBLEM-SOLVING

These latest products, along with the entire family of Internet products, represent an important distinction in Digital's computing philosophy. The key to an IBM to non-IBM network or other multivendor network is to provide customers with systems that work well together. When other brands of equipment are introduced into the picture they still work well together. At Digital, multivendor environments are viewed as an ideal way for businesses and other organizations to exchange information and make the most of their computing resources. We believe that the perceived advantages of centralized control are enhanced by the economy and flexibility of distributed communications, processing, and resource sharing. The key to making informed (and thus, successful) business decisions is having a free flow of information and communications, horizontally as well as vertically, in an organization.

That's why Digital's DNA architecture follows a multivendor, open systems interconnect model, whereby every computer system—especially our own—can access other systems to exchange information.

We also support the ongoing development and enhancement of software and hardware products that simplify the user interface in multivendor systems and networks. Digital's family of VAX processors, for example, can easily exchange documents with IBM's DISOSS system or Wang's OIS family of office systems.

Digital also provides multivendor connections to UNIVAC™ and Control Data Corporation computers, along with a large and growing family of interconnect software packages to connect other computer systems and networks with the entire spectrum of Digital operating systems.

FROM THE USER'S VIEW

Now, users are freed from the technical complexities of multivendor systems and networks, and can concentrate on the work at hand. To the user, two or more networks, even from different vendors, can appear as one. Better yet, the level of computer support assigned to each user can be determined by the actual level of need, rather than the technical requirements of a remote mainframe, the terminals, or the communications network itself.

From the end-user's point-of-view, a multivendor network should provide all of the needed compute power, and still be no more difficult to use, than a single-vendor system. For the MIS Directors and System Managers responsible for getting the systems up and running, however, it is a complex business indeed. With the multitude of host system configurations, operating systems, processors, applications subsystems, networking and communications technologies, peripherals, and options available from both IBM and Digital, a great deal of equipment planning is needed before any user can begin enjoying the advantages of the combined system.

The goal is to provide products that help users make the most of information, the management of the systems as simple and efficient as possible and to continue making the exchange of information among Digital, IBM and other vendors' systems routine and efficient.

IBM is a register trademark of International Business Machines.
DISOSS is a trademark of International Business Machines.
The following are trademarks of Digital Equipment Corporation: VAX, VMS, ALL-IN-1, WPS-PLUS, DECnet, DECmail, VAXmail.
UNIVAC is a trademark of Sperry Corporation.

Figure 1

Ethernet Integrated Into DNA

<table>
<tr><th>ISO LAYERS</th><th>DNA LAYERS</th><th colspan="3">DNA FUNCTIONS</th></tr>
<tr><td rowspan="2">APPLICATION</td><td>USER</td><td rowspan="3" colspan="3">FILE TRANSFER
REMOTE RESOURCE ACCESS
DOWN LINE SYSTEM LOAD
REMOTE COMMAND FILE SUBMISSION
VIRTUAL TERMINALS</td></tr>
<tr><td>NETWORK MANAGEMENT</td></tr>
<tr><td>PRESENTATION</td><td>NETWORK APPLICATION</td></tr>
<tr><td>SESSION</td><td>SESSION CONTROL</td><td rowspan="2" colspan="3">TASK-TO-TASK</td></tr>
<tr><td>TRANSPORT</td><td>END COMMUNICATIONS</td></tr>
<tr><td>NETWORK</td><td>ROUTING</td><td colspan="3">ADAPTIVE ROUTING</td></tr>
<tr><td>DATA LINK</td><td>DATA LINK</td><td rowspan="2">DDCMP POINT-TO-POINT MULTI-POINT</td><td rowspan="2">X.25</td><td rowspan="2">ETHERNET</td></tr>
<tr><td>PHYSICAL</td><td>PHYSICAL LINK</td></tr>
</table>

Digital Equipment Corporation's networking strategy is based on Digital Network Architecture (DNA). DNA defines the standard protocols, interfaces and functions that enable DECnet nodes to share data and resources. It closely resembles the International Standards Organization's seven-layer networking model for open system interconnection. DNA's hierarchical framework provides flexibility for adding new communications technologies, such as Ethernet, without affecting a user's applications. Ethernet protocols are added to the two lowest layers of DNA, and the structure of all the other layers remains unchanged.

NEW ALTERNATIVES IN LOCAL NETWORK TECHNOLOGIES

Roderick Flakes
Networks and Communications
Digital Equipment Corporation, TWO/B12
Andover Street
Tewksbury, MA 01876
617-858-2018

ABSTRACT

Telecommunications managers have a variety of new alternatives in planning local area networks. The three major communications technologies—PBX, baseband and broadband—are available in many combinations of network architecture, performance and capacity. The optimum alternatives are best selected on the basis of what users will be needing as their applications grow and change with time.

Telecommunications managers who are planning local networks are today facing both a special opportunity and an extra burden. On the one hand, they are in a better position than ever to create a local communications environment that best meets their companies' particular needs. On the other hand, they must make their choices and decisions among several communications technologies, many types of networking hardware and software, and numerous competing vendors.

Planning a local network is, in short, a complex problem without a clear-cut general-purpose solution. The alternatives first depend on the nature of the company's current facilities for voice, data and video communication. Then, they depend on the assortment of applications the network must be able to support, now and in the future. As a matter of fact, much of the planning must necessarily be based on projections and best guesses on what the future will bring.

The vendors in the communications and computer industries are espousing alternative network architectures and marketing proprietary hardware and software to support them. At the same time, they have been working with their telecommunications markets in the U.S. and abroad to develop industry standards on various aspects of communications technology.

Much of the progress in standardization is being accomplished through the IEEE (Institute of Electrical and Electronics Engineers) and EIA (Electronic Industries Association) in the U.S. and CCITT (Consultative Committee on International Telephony and Telegraphy) internationally. Many standards that have been promulagated by these associations originated with individual or cooperating vendors specializing in particular aspects of local networks. Proprietary specifications, too, have become de facto standards without benefits of formal recognition.

FIVE MAJOR CONCERNS

Telecommunications managers are likely to have five major concerns in planning a local area network (LAN): multivendor environment, expandability, flexibility, investment protection, and ability to take advantage of new technologies.

With a compatible multivendor environment, the telecommunications manager is free to select optimum alternatives among all potential vendors, not just one or two. It may also happen that some decisions must be made with or by others: the terminals, workstations and computers to be linked by the network are often selected by end users. As a matter of fact, many of the I/O devices and nodes-to-be may have been operating on a stand-alone basis for some time.

Standards that are explicit, broad in scope and widely accepted by communications vendors and markets are very important to a successful multivendor environment. National and international standards that have been established jointly by vendors and users, of course, offer at least the possibility of compatible products from more vendors. De facto standards, too, can have much of the weight of industry standards and, after a time, may be adopted as such anyway.

Expandability is a major concern because, almost by definition, any LAN—or wide area network as well—is constantly growing. In order to minimize the added investment at any one time, it should be possible to grow by small increments, say, a few nodes at a time. Large upgrades to a network mean that communications resources are likely to be underutilized until growth catches up.

Flexibility is similarly valuable to telecommunications managers because a LAN is probably going to change as well as grow. Nodes may be moved from one floor to another, or switched from terminal service to number-crunching.

Investment protection means that the telecommunications manager will almost always want to retain existing resources as part of a new LAN. These may include currently standalone computers that are to share data files in the network and years-old voice-grade wiring that could also be used for data communications. Terminals and desktop computers may represent a large investment over a long period of time.

The ability to take advantage of new technologies in a non-disruptive fashion (sometimes called technology migration) is far more important than simply maintaining up-to-date network service. New technologies, whether at the network, device or semiconductor level, can offer improved, expanded network service at lower cost than continued expansion with current technologies.

Note that none of these concerns directly address relative installation cost of alternatves in providing LAN service. Choices are weighed and decisions made primarily on the basis of current and future needs and long-term investment. The lowest-cost network communications technology may be the most expensive to maintain, expand and migrate. The technology offering the highest performance per dollar today may be far too restrictive in expandability to support expected growth.

APPLICATIONS AND COMMUNICATIONS

The resources of a LAN can be viewed as elements of either applications or communications technology. Application technology, which is a responsibility of the MIS manager, consists of all hardware and software functions that directly or indirectly interface users at terminals or desktop computers. Communications technology, a responsibility of the telecommunications manager, consists of all voice, data and video functions that support application technology.

The applications and communications areas can be supervised separately by the MIS and telecommunications managers, of course, or together by one person. In either case, it must be possible to improve and expand applications without change in the communications area. Conversely, efforts to upgrade data throughput and reduce communications operating cost should have no effect in the applications area. This independence of applications and communications is inherent in the layered modular structure of networking architecture as defined by the familiar ISO (International Standards Organization) Open Systems Interconnection model (right in Figure 1) and implemented in DEC's Digital Network Architecture (DNA), among others.

THREE APPLICATION AREAS

The overall purpose of a LAN is to improve productivity through better communication and sharing of information in a department, group or other company organization within the network's range.

There are three types of organizations whose applications place somewhat different demands on a LAN's resources. By far the largest of these today is the office: here, electronic mail and database access are among office network applications provided to users of terminals and desktop computers. Message length is usually relatively short, and required data transmission rate relatively low.

Secondly, the manufacturing department uses a LAN for material tracking, process monitoring and other applications that usually involve transmitting small blocks of data but often rely on the very fast response characteristic of real time computing. Finally, the laboratory and scientific environment can demand similar real time capability and high-speed transmission of very large data files between microcomputer-aided instruments and number-crunching mainframes.

THREE COMMUNICATIONS MODES

The user at a workstation in the office, on the manufacturing floor or in the laboratory may be supported by one or more of the three basic modes of communication: voice, data and video (Figure 2).

PBX-based voice facilities, of course, will have been available to most current users for some time, although they probably have been expanded greatly and the PBX upgraded several times. Video communications, although it receives a good deal of attention, has yet to be installed very widely but will surely be commonplace by the end of the eighties.

Data communications, which is the primary concern of this paper, can be implemented in a modern PBX-based circuit-switching LAN or in a circuit-switching or packet-switching LAN based on either baseband or broadband transmission media. The key factors in considering these alternative communications technologies are: a PBX-based LAN can share the PBX and wiring with voice communicaitons, a baseband LAN is dedicated to high-speed data transmission, and a broadband LAN can support data, voice and video on the same cable.

PBX-BASED LANs

Voice communications in the business environment is still based predominantly on electromechanically circuit-switching analog PBXs. Telephone wiring offers built-in physical media for data communications, both locally and with remote locations through switched and dedicated lines of public telephone systems. The data rates in analog PBXs are limited by the capabilities of the LAN's modems; the highest rate today is 9600 bps, and the required modems for this speed are extremely complex and expensive.

Digital PBXs operate much faster and more efficiently allocate voice circuits by means of semiconductor logic switching. Voice and data are switched in digital form, and time division multiplexing (TDM) assigns and switches available channels. The newer models of digital PBXs digitize voice signals at the workstation so that voice and data can be integrated and transmitted together over a single telephone wire to the PBX.

Voice and data are combined in a typical digital PBX today by means of vendor-supplied data access modules (DAMs), as shown in Figure 3(a). A DAM conditions the data from the terminal (synchronous or asynchronous) for transmission over the synchronous link from the workstation to the PBX. The synchronous links are usually rated at multiples of 64 kbps, the data rate for digitized voice. The digital signals are switched through yet another set of DAMs to the computer—and the reverse takes place between the computer and workstation. There is a separate point-to-point duplex path between each workstation and the computer.

The diagram in Figure 3(a), however, conceals a basic problem of the multivendor environment: there is no common data transmission protocol among vendors of the various communications devices that link the workstation and the computer. If the company decides to replace the PBX, the DAMs—and perhaps even the terminals—must also be replaced. (The fact that DAMs are called different names by different vendors makes things even more difficult.)

In mid-1983, Digital and Northern Telecom Inc. announced development of the Computer-to-PBX Interface (CPI) for linking computers and PBXs. Its purpose was to provide a network transmission standard that would support compatibility between PBXs and computers of different vendors. The CPI specification was also designed for easy implementation and low cost.

The diagram in Figure 3(b) shows the CPI-based equivalent of the typical current system in Figure 3(a). The DAMS at right in Figure 3(a) have been replaced by hardware/software interfaces built into the PBX and computer. The CPI standard provides for 24 independent data channels over four wires in a standard 1.544-mHz T1 carrier link. The CPI-based configuration closely matches existing digital PBX architectures: the synchronous data rate remains 56 kbps (8 of the 64 kbps is used for in-channel signaling), and CPI uses T1 TDM circuit switching just as do digital PBXs.

As of late 1984, nine vendors have committed themselves to developing CPI products. These comprise five PBX maufacturers—American Telecom, InteCom, Mitel, Northern Telecom, and Rolm—and four computer manufacturers—Data General, Digital Equipment, Prime and Wang. Since all of their CPI products will be using the same data transmission protocol, any one of these vendors' PBXs can be directly linked to any one of their computers.

Less than a year after introduciton of the CPI specification, AT&T announced an alternative specification called the Digital Multiplexed Interface (DMI). Very similar in concept to the CPI (the diagram in Figure 3b would not change), the DMI provides for 23 channels over a T1 link. Both specifications have been presented to the EIA as proposed standards.

There are a number of technical differences between CPI and DMI. The most important of these have to do with time and the CCITT's evolving Integrated Services Digital Network (ISDN), which will probably attain final form in 1988. DMI reflects many ISDN proposals but is several years behind CPI in product availability. CPI was designed to encourage early development of relatively simple, low-cost communications products, and so does not currently encompass ISDN proposals that were considered most likely to change.

Altough DMI-based products had not yet been delivered by the end of 1984, five vendors have committed to the DMI approach: Data General, Honeywell, Wang, Hewlett-Packard, and AT&T.

CIRCUIT-SWITCHING DATA LANs

Data communications in a shared PBX-based LAN is currently low-speed but 56 kbps for data is more than adequate for many common interactive applications. In theory, the 1.544-kbps bandwidth of the T1 carrier could be divided into fewer, higher-speed data channels or even into a single 1.544-kbps channel. But doing so would reduce or eliminate sharing of the existing telephone wiring plant, which was the main reason for sharing in the first place. Moreover, it's simply a matter of time before data communications between users and one or more computers begins to strain wiring and PBX capacity and degrade service.

The telecommunications manager's next step is likely to be a LAN dedicated to data communications, based on its own data switch and transmission medium, either twisted-pair wire or coaxial cable. Most or all of data service would probably be off-loaded from the PBX system, thereby immediately increasing the PBX's voice capacity. The data LAN can be either a circuit-switching, PBX-like network or a packet-switching network more closely allied to a wide area network.

Dedicated circuit-switching data networks are a relatively low-cost way of introducing data communications on a small scale, especially for terminal and workstation links to computers. Because they are point-to-point systems, they cannot share lines as in packet-switching networks and so cannot be similarly expanded. Once a circuit-switching data network reaches its capacity—say, 1,000 simultaneous communications over 2,000 lines—another data switch must be installed to grow any further.

Circuit switching is most suitable for voice communications: users normally talk to only one other person at a time, need only the physical connection to operate, and the sessions are usually quite short. In contrast, packet switching is most suitable for computer-to-computer communications: one computer may be called on to address simultaneously two or more other machines (perhaps all of them in the network), more than the physical connection is needed to operate (high-level protocols above the physical layer in a packet-switching LAN provide file transfer, task-to-task communication and other common computer-to-computer functions), and computers are likely to be actively using lines for hours at a time (which can cause blocking in point-to-point communications).

HIGH-SPEED PACKET SWITCHING DATA LANs

High-speed packet-switching LANs present telecommunications managers with a wide variety of alternatives in combinations of data rate, number of nodes, expandability, flexibility, ease of installation and operation, and original cost.

Fortunately, the standardization effort on packet-switching LANs began early on with establishment of the IEEE Project 802 in 1980. In view of the complexities of data communications, one of the 802 committee's first steps was to define a local network reference model. The reference model at left in Figure 1 is functionally identical to the data link and physical layers of the ISO model at right. The IEEE has further divided the data link layer into logical link control and medium access control functions.

Logical link control sets up, maintains and terminates a logical link between nodes in a network. Medium access control insures that all devices at nodes in the network can share a single transmission medium. The physical layer performs tasks associated with the specific physical medium—twisted-pair wire, baseband coaxial cable, broadband coaxial cable, or fiber optics.

Before long, the 802 committee also saw that it would be necessary to standardize on more than one functional implementation at the medium access control and physical layers of the network model. Standards have thus far been approved for two access methods: IEEE 802.3 on CSMA/CD (Carrier-Sense Multiple Access with Collision Detection), 802.4 on token bus, and 802.5 on token ring. Two transmission media have been defined for each of these access methods: baseband and broadband coaxial for CSMA/CD; single-channel broadband and broadband coaxial for token bus; and twisted-pair wire and baseband coaxial for token ring.

High-speed data transmission at 10 mbps is defined by the IEEE 802.3 CSMA/CD baseband standard, which was approved in 1982. The IEEE standard and the Ethernet baseband specification, published jointly by Digital, Xerox and Intel in 1980, are virtually identical. To date, Digital alone has shipped hardware for more than 7,000 Ethernet nodes, destined for networks varying widely in number of nodes and physical size. Individual baseband Ethernets may have up to 1,023 nodes and extend up to 2,800 meters between sending and receiving nodes.

Many current Ethernet installations take advantage of the LAN's high data rate for communications between minicomputers, superminis and larger machines. In addition, Ethernet vendors have over recent years developed devices such as terminal servers, twisted-wire local interconnects, repeaters, routers and gateways that extend Ethernet communications both within and beyond the range of a single cable. Figure 4 illustrates these extensions in Digital's implementation of Ethernet.

Within an Ethernet, each terminal server links as many as 32 terminals to multiple hosts (in contrast to a single host connection). Local interconnects (called DELNIs) do the same with as many as eight desktop computers, workstations and multi-user micros, and as many as eight DELNIs can be cascaded off one node. Repeaters link an Ethernet in one department to individual Ethernets in other departments on different floors of the same building or on the same campus.

Outside the LANs themselves, routers and gateways allow extending Ethernet communications to DEC DNA wide area networks, X.25 public packet-switching networks, and even IBM SNA networks. This is possible because the Ethernet baseband protocol has been implemented along with the X.25 protocol and DDCMP protocol at the data link and physical layers of the DECnet Phase IV implementation of DNA network architecture.

BROADBAND FOR VOICE, DATA, VIDEO

High-speed packet-switching data transmission is the same to end users—and in many ways to telecommunications managers—whether implemented on baseband or broadband cable. The significant differences have to do with whether the LAN cable plant is to carry data alone or voice and video as well.

The decision to establish a broadband LAN depends mainly on a number of factors having little to do with the current needs of data transmission. Has a broadband cable already been installed for video communications or has video itself become a vital need that must be met? Is there a management consensus that broadband is the communications technology of choice for the future?

Broadband Ethernet is based on a set of data transmission techniques that implement 10-mbps CSMA/CD medium access control on broadband coaxial cable. These techiques, which are compatible with the IEEE baseband standard, were defined in a Digital proposal to the IEEE 802.3 committee and are now being considered as a draft standard. Because it is also implemented at the two lowest layers of the DNA network architecture, a broadband Ethernet can be linked into extended LANs and wide area networks like those in Figure 4.

Since Digital's broadband Ethernet is compatible with its own baseband Ethernet implementation, much of the same network hardware is used for both: controllers, routers, gateways, and terminal servers. This permits non-disruptive growth from baseband to broadband technology and then on to links with wide area networks.

The Ethernet baseband and broadband transceivers generally perform the functions of the physical layer and the controllers the functions of the data link layer (except that in the Digital implementaiton, the controller handles the physical layer function of Manchester encoding and decoding). The broadband transceiver has the additional physical-layer task of modulating digital data onto the broadband carrier and demodulating carrier-frequency data.

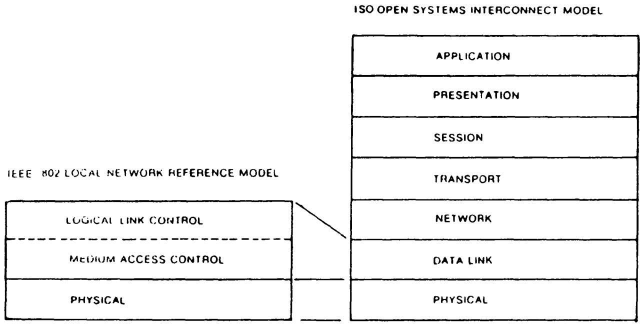

ISO OPEN SYSTEMS INTERCONNECT MODEL
APPLICATION
PRESENTATION
SESSION
TRANSPORT
NETWORK
DATA LINK
PHYSICAL
IEEE 802 LOCAL NETWORK REFERENCE MODEL
LOGICAL LINK CONTROL
MEDIUM ACCESS CONTROL
PHYSICAL

FIGURE 1

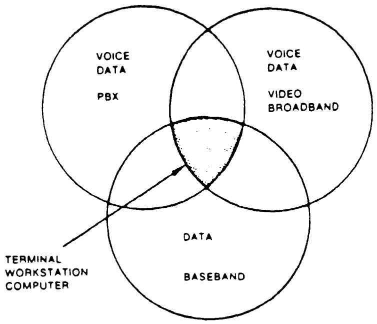

VOICE
DATA
PBX
VOICE
DATA
VIDEO
BROADBAND
DATA
BASEBAND
TERMINAL
WORKSTATION
COMPUTER

FIGURE 2

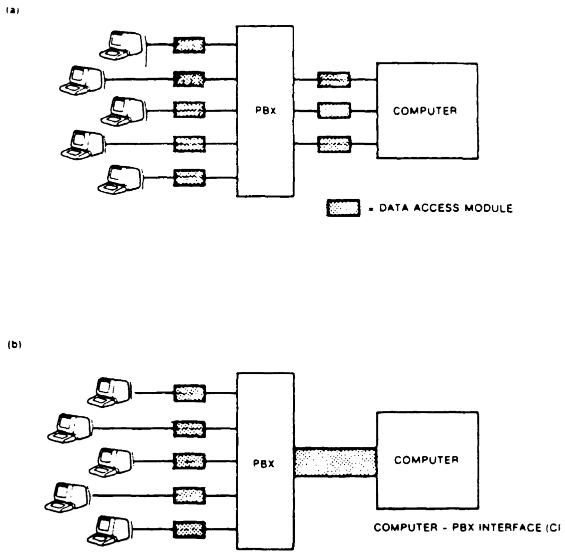

(a)
PBX
COMPUTER
= DATA ACCESS MODULE
(b)
PBX
COMPUTER
COMPUTER - PBX INTERFACE (CI

FIGURE 3

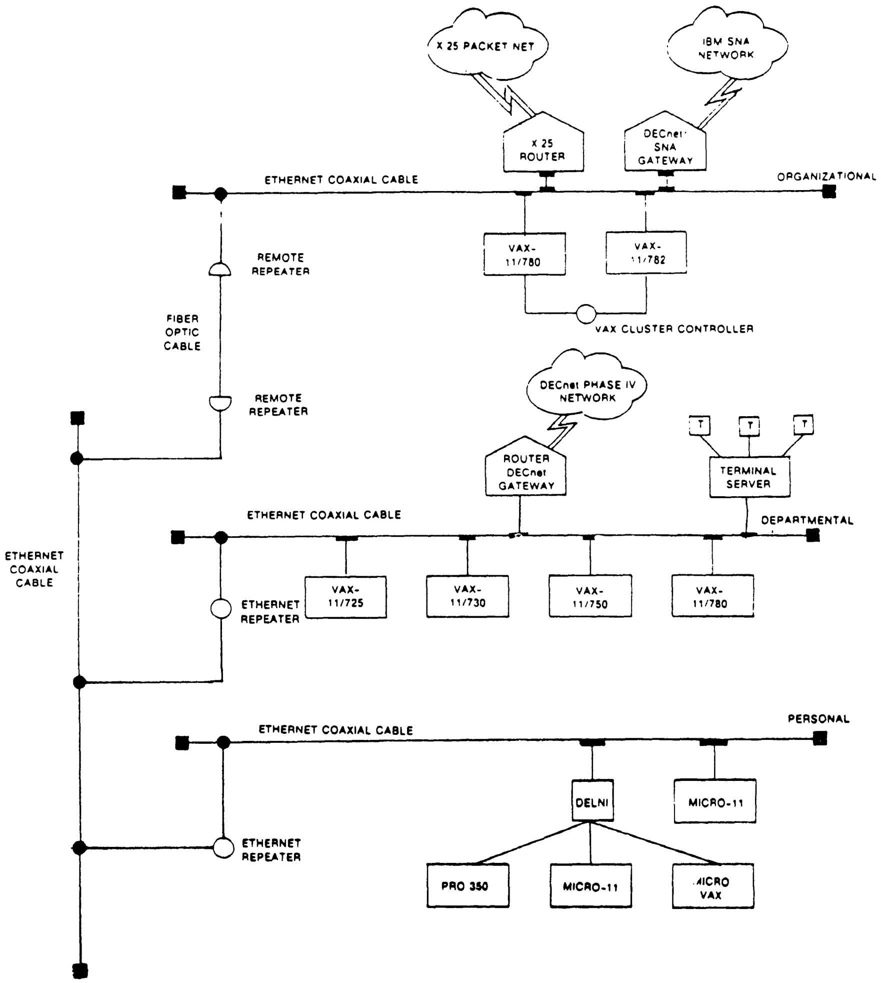

X 25 PACKET NET
IBM SNA NETWORK
X 25 ROUTER
DECnet SNA GATEWAY
ETHERNET COAXIAL CABLE
ORGANIZATIONAL
VAX-11/780
VAX-11/782
REMOTE REPEATER
FIBER OPTIC CABLE
VAX CLUSTER CONTROLLER
REMOTE REPEATER
DECnet PHASE IV NETWORK
T
T
T
ROUTER DECnet GATEWAY
TERMINAL SERVER
ETHERNET COAXIAL CABLE
DEPARTMENTAL
ETHERNET COAXIAL CABLE
VAX-11/725
VAX-11/730
VAX-11/750
VAX-11/780
ETHERNET REPEATER
ETHERNET COAXIAL CABLE
PERSONAL
DELNI
MICRO-11
ETHERNET REPEATER
PRO 350
MICRO-11
MICRO VAX

FIGURE 4

DATAGRAM

DATAGRAMs are short messages, comments, requests, or answers that are published in NETwords. Please fill in the sections below and send the DATAGRAM to:

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Title: DECNET SECURITY INFO

Message:
I AM TRYING TO FIND ANY REPORTS
OR PAPERS CONCERNING SECURITY
CONSIDERATIONS WHEN USING DECNET.
I AM PARTICULARLY INTERESTED IN
SECURITY ISSUES WITH REGARD TO
Dept. of Defense Guidelines.

ANY INFORMATION ON THIS subject
Would be APPRECIATED.

Your Name: Rich Stoehe
Address: MITRE CORP
1259 LAKE PLAZA, Colo Spgs, ~~8~~CO 80916
Telephone: 303 ~~5~~ 576 2602

If this is a reply to a previous DATAGRAM, what #? __

Signature: ________ **Date:** 29-JAN-86

DATAGRAM

DATAGRAMs are short messages, comments, requests, or answers that are published in NETwords. Please fill in the sections below and send the DATAGRAM to:

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Title: DECnet/Ethernet Performance and Capacity Models, Rules of Thumb, etc.

Message: We have just begun to collect some data about the usage of our DECnet/ethernet and we wish to use that data for performance analysis and capacity planning. In order to do that, we need some capacity definitions (What percentage of the 10Mb bandwidth can we expect to actually use?), rules of thumb, performance and capacity models (What if we add one more node?), etc. We would appreciate hearing from anyone having such information.

Your Name: Jim Haskett
Address: Memorial Hall 330, Indiana University
Bloomington, Indiana 47405
Telephone: 812-335-0421

If this is a reply to a previous DATAGRAM, what #? ___

Signature: ______ **Date:** 04/02/86

OFFICE AUTOMATION

OA

OFFICE AUTOMATION SIG STEERING COMMITTEE

Chairman
Katherine 'Kit' Trimm
Pivotal, Inc.
Tucson, AZ

Vice Chairman
Ralph Bradshaw
Johnson and Johnson
Raritan, NJ

Communications Committee Representative
E. Catherine Ditamore
ARA Services
Philadelphia, PA

Symposium Coordinator
Mitch Brown
Gen Rad, Inc.
Waltham, MA

Special Projects
Gene LeClair
HQ Dept. of Army
Washington, DC

BOF Coordinator
Ray Kaplan
University of Arizona
Tucson, AZ

Newsletter Editor
Therese LeBlanc
Wheeling, IL

Library
Bob Hassinger
Liberty Mutual Research Center
Hopkington, MA

Tape Copy Coordinator
Randall Buck
Columbia Savings
Irvine, CA

ALL-IN-1 Working Group
Leon E. Ottley
Evans and Sutherland
Salt Lake City, UT

Symposia Assistant
Sal Gianni
Northeast Utilities
Hartford, CT

Store Coordinator
Mike Jackson
Air Force Operational
Test and Evaluation Center
Kirtland AFB, NM

Personal Computer SIG Liaison
Cheryl Johnson
Grinnell College
Grinnell, IA

Networks SIG Liaison
Gene LeClair
HQ Dept. of Army
Washington, DC

DECUS Europe OA SIG
Andreas Verbay
Telinco AG
Spiegelstrasse 20

Digital Counterparts
Les Agigian
Digital Equipment Corporation
Merrimack, NH

Geof Bock
Digital Equipment Corporation
Merrimack, NH

Session Notes
Martha Rudkin
GMF Robotics
Troy, MI

In This Issue

From The Editor..1
- Therese LeBlanc

Why Reordering a Forms Library Doesn't Work........................2
- Bart Z. Lederman

Shutting ....2 - 6
- Gregory N. Brooks

Patches for ALL-IN-1 Version 2.0..................................6 - 7
- Gregory N. Brooks

From The Editor

Hello everyone, welcome to the June issue. We have two technical articles for you regarding ALL-IN-1; a non-interactive shutdown procedure file and several patches for V.2.0. Many thanks to the people who contributed articles for this issue. And speaking of contributors (here's my soap-box for the month) is there anyone out there who has created an automated office using products other than ALL-IN-1? As much as I enjoy publishing all the goodies I receive about ALL-IN-1 I can't help but think that some of you are holding out on me. If you have created an automated office by combining several DEC systems and/or software packages, please submit an article describing what you have put together and how you did it. I think that everyone would be interested in hearing what your company has done.

The post-symposium information (from the Spring Symposium in Dallas) will be appearing in the July issue. Make sure to check it for the new SIR list for you to vote on.

Regards,

Therese

Therese LeBlanc
275 London Place
Wheeling, IL 60090
(312) 459-1784

WHY REORDERING A FORMS LIBRARY DOESN'T WORK WITH ALL-IN-1

One of the options in the Forms Development section of ALL-IN-1 V2.0 is reordering a forms library. This submits OA$LIB:REORDER.COM as a batch job. This command procedure could probably be used interactively outside of A-1. Unfortunately, it requires the presence of a task called TRIM which is supposed to be supplied with A-1. The Telephone Support Center informs me that DEC forgot to put it on the distribution kit: they are suppling a copy, and state that this will be fixed in the next release.

An examination of the command definition for TRIM indicates that it may well be worth having on it's own account as it should be very useful for processing the output of directory listings and library listings into a useful form for generating command procedures.

- Bart Z. Lederman -

ALL-IN-1 HINTS AND KINKS

Shutting down All-in-1 non-interactively

Gregory N. Brooks
Washington University
St. Louis, MO.

All-in-1 provides a facility for shutting down the All-in-1 system interactively, but does not provide such a facility to do so from a command procedure or a batch job.

Occasionally it is necessary to be able to shut down All-in-1 non-interactively. Such as when performing an automatic image backup and restore of a disk drive. If All-in-1 is not shut down before performing the restore (when the disk is dismounted and then mounted foreign) the files A1.EXE, MEMRES.FLC, A1.TXL, and OAFORM.FLC will still be open and will cause VMS to go into the infamous mount/dismount state. To clear this condition, the system must be rebooted, not to mention that your restore operation will have to be started again once the system is rebooted.

Also, I feel a bit safer in having my All-in-1 system shutdown as part of SYSHUTDWN.COM. I never encountered a problem before I included an All-in-1 shutdown as part of the site specific shut down, but it just seems to be proper, and tidy to shut down All-in-1 as part of SYSHUTDWN.COM. That way I am always sure All-in-1 was shut down properly before the system goes down.

To accomplish the non-interactive shut down of All-in-1, I studied, and modified two command procedures, the All-in-1 startup file, A1V2START.COM (I am using the Bert Roseberry version as published in the OASIG Newsletter) and the the SMSHUTDN.COM from OA$LIB that is used for the interactive shut down.

I reversed all the steps of A1V2START.COM so that all the All-in-1 libraries and TXL's are removed and the installed images for All-in-1 are removed. I call the file A1V2STOP.COM.

After studying the OA$LIB:SMSHUTDN.COM I found that the FMS form used when shut down is selected passes two important parameters to the command procedure that are used to define the symbols, MNT (time till shutdown) and DOWNTIME (duration of All-in-1 shut down). I made permanent symbol definitions so interaction would not be required.

SMSHUTDN.COM uses mailboxes to communicate with All-in-1 when it is run interactively. I removed all the mailbox references so the procedure does not require one to be actually running All-in-1. I call my version A1SHUTDOWN.COM.

A1SHUTDOWN.COM also contains a lot of code it probably does not really need, But I have not spend the time it would require to determine exactly what code is not useful.

Please note that A1V2STOP.COM does call A1SHUTDOWN.COM so it is not a self sufficient procedure. I opted to call A1SHUTDOWN.COM instead of imbedding the code from A1SHUTDOWN.COM within A1V2STOP.COM since A1SHUTDOWN.COM is so useful just by itself and leaving it as one file required less effort, if and when, it required any updating or enhancements.

The listings are included below. I have tried to make appropriate comments where they seemed useful. I hope to include them on the OASIG symposia tape for Dallas.

```
$!                       A1V2STOP.COM
$!
$       V := 'F$VERIFY( 0 )
$!======================================================================
$!
$!      File: A1V2STOP.COM                    Original: 27-DEC-1985
$!
$!      Description:
$!      Shutdown and de-assign and removed installed images and libraries.
$!
$!      Changes:
$!
$!      Author: Gregory N. Brooks 12-27-85
$!
$!======================================================================
$!
$       ON CONTROLY THEN GOTO EXITPROC
$       set noon                        ! Ignore install errors
$       Say := write sys$output
$ !
$       disk := YOUR DISK FOR ALL-IN-1 HERE
$ !
$       say "All-in-One: Removing All-in-One forms"
$ !
$ ! Run ALLIN1 to remove the forms and TXL libraries
$ !
$       set noon
$       define/nolog sys$output nl:
$       define/nolog sys$error nl:
$       ALLIN1/NOINIT
oa$fbtremovelib oa$lib:memres
oa$fbtremovelib oa$lib:oaform
oa$txlremove
$       deassign sys$output
$       deassign sys$error
$ !
$ !
$       set on
$ !
$ !
$       say "Shutting down All-in-1"
$ !
$ @sys$com:a1shutdown.com       !CALLS A1SHUTDOWN.COM HERE TO SHUT
$                               !DOWN ALL-IN-1
$
$       say "All-in-One: Removing All-in-One installed images"
$!
$       install :== $install/command
$!
$! Remove All-in-One
$!
$       set noon
$       define/nolog sys$output nl:
$       define/nolog sys$error nl:
$       install remove 'disk'[ALLIN1.LIB]a1
$       install remove sys$share:dpeshr
$       install remove sys$message:dpemsg
$       install remove sys$message:fdvmsg
$       install remove sys$share:fdvshr
$       install remove sys$message:xportmsg
$       deassign sys$output
$       deassign sys$error
$       set on
$!
$ !
$ EXITPROC:
$       V = F$VERIFY( V )
```

```
$!                       A1SHUTDOWN.COM
$!
$       curpriv = f$setprv("SYSPRV,TMPMBX,SYSNAM,WORLD,OPER")
$       if .not. f$priv("SYSPRV,TMPMBX,SYSNAM,WORLD,OPER") then goto nopriv
$ !
$!      DEFINE SYMBOLS MNT AND DOWNTIME
$
$       mnt = 1                 !SMALLEST INCREMENT OF TIME POSSIBLE
$       downtime := UNKNOWN     !UNKNOWN TIME PERIOD SPECIFIED
$ !
$ !     find the name of the allin1 image
$
$       vers = f$getsyi("version")
$       if f$extract(0,1,vers) .eqs. "Z" .or. -
           f$extract(0,1,vers) .eqs. "X" .or. -
           f$extract(0,2,vers) .eqs. "V4" -
        then -
            goto v4parse
$ v3parse:
$       a1image = f$parse("oa$image","sys$system:.exe") - ";"
$       goto continue
$
$ v4parse:
$       a1image = f$parse("oa$image","sys$system:.exe",,,"noconceal")- ";"
$
$
$ continue:
```

```
$!
$! See if we're already shut down.
$!
$       if f$parse(alimage,,,"NAME") .eqs. "SMA1" then goto alreadyshut
$ !
$! this part gets the current time and adds the length of time
$! before allin1 goes down to it .It writes this time to the file accessed by
$! users during shutdown. And is written as part of the shutdown message
$! sent to the users screen.
$
$ time = f$time()
$ mins = f$extract(15,2,time)
$ hours= f$extract(12,2,time)
$ mins = mnt + f$integer(mins)
$
$ gethours:
$       if mins .lt. 60 then goto timefin
$       mins = mins - 60
$       hours = hours + 1
$       if hours .ge. 24 then hours = 0
$       goto gethours
$
$ timefin:
$       if mins .le. 9 then mins = "0" + f$string(mins)
$       if hours .le. 9 then hours = "0" + f$string(hours)
$       fintime = f$string(hours) + ":" + f$string(mins)
$
$
$ notice:       ! Prepare notice program
$ !
$       OPEN/WRITE Z OA$DATA:SMSHUTDN.TMP
$       WRITE Z "ALL-IN-1 is going down for maintenance at ''fintime' "
$       WRITE Z "and will be down for ''downtime'"
$       CLOSE Z
$
$       SET PROTECTION=(W:RE) OA$DATA:SMSHUTDN.TMP
$ !
$ shutsenders:
$ !
$       WRITE SYS$OUTPUT ""
$       WRITE SYS$OUTPUT "      Shutting down ALL-IN-1 senders"
$       WRITE SYS$OUTPUT ""
$!
$!=========================================================================
$!
$!              THIS ADDITION IS SITE SPECIFIC
$!
$!      CHECK TO SEE IF WE ARE DOING AN IMAGE BACKUP OF DISK
$!
$ LOGICAL = F$TRNLNM("BACKUP$RESTORE","LNM$SYSTEMTABLE")
$ IF LOGICAL .EQS. "YES" THEN OA$MTISTATUS := "STOPPED" !WE ARE
$ IF LOGICAL .EQS. "NO" THEN OA$MTISTATUS :== "ON HOLD" !WE ARE NOT
$!
$       assign/system "''OA$MTISTATUS'" OA$MTISENDERCONTROL
$       assign/system "''OA$MTISTATUS'" OA$MTIFETCHERCONTROL
```

```
$       define/system  oa$image  OA$LIB:SMA1.EXE
$!
$!DEFINES SENDER CONTROL SO FILES WILL NOT BE OPENED ON IMAGE BACKUP
$!===========================================================================
$ !
$ !     find the users running the ALL-IN-1 image and ask each current
$ !     user to exit. Create a file of all current users and the cputime
$ !     used.
$
$
$create/fdl=sys$input:
file
        name                alusers.tmp
        organization        indexed
record
        format              variable
        size                40
key 0
        changes             no
        duplicates          no
        prologue            3
        seg0length          8
        seg0position        0
        type                string
$
$       open/read/write proc alusers.tmp;0
$       first = 1
$       context = ""
$       write sys$output "scanning for ALL-IN-1 users to ask to log off"
$ loop:
$       pid = f$pid(context)
$       image = f$getjpi("''pid'","IMAGNAME") - ";"
$       if "''image'" .eqs. "''alimage'" then goto R1
$       if pid .eqs. "" then goto R2
$       goto loop
$ !
$ R1:
$       if first .ne. 1 then goto R1A
$       cputim = f$getjpi(pid,"CPUTIM")
$       write proc pid,cputim
$ R1A:
$       term = f$getjpi(pid,"terminal")
$       msg := -
"Please exit. Going down in ''mnt' minutes (at ''fintime') for ''downtime'"
$       REPLY/bell/terminal="''term'"  "''msg'"
$       goto loop
$ R2:
$       IF first .eq. 1 then close proc
$       first= first+1
$
$       IF 'mnt' .LT. 10 THEN GOTO R3
$       write sys$output -
        "WAIT   00:05:00.0      ! Wait 5 minutes before next announcement"
$       WAIT    00:05:00.0      ! Wait 5 minutes before next announcement
$       mnt = mnt - 5
$       GOTO loop
$ R3:
$       write sys$output -
        "WAIT   00:01:00.0      ! Wait 1 minute before next announcement"
$       WAIT    00:01:00.0      ! Wait 1 minute before next announcement
$       mnt = mnt - 1
$       IF 'mnt' .GT. 1 THEN GOTO loop
$
$ R3A:
$       open/write/read proc alusers.tmp;0
$       alrunning = "no"
$       firstonefound ="no"
$       context = ""
$       usercount = 0
$
$ getuser:
$       pid = f$pid(context)
$       image = f$getjpi("''pid'","IMAGNAME") - ";"
$       if "''image'" .eqs. "''alimage'" then goto running
$       if pid .eqs. "" then goto endlist
$       goto getuser
$
$ running:
$       usercount = usercount +1
$       if alrunning .eqs. "no" then alrunning = "yes"
$       if firstonefound .eqs. "yes" then goto more
$       firstonefound = "yes"
$       write sys$output -
        "process name           cpu time used since start of shutdown"
$       goto details
$
$ more:
$       if usercount .LT. 20 then goto details
$       write sys$output "there are more users"
$       write sys$output "process name          cpu time used"
$
$ details:
$       cputim2 = f$getjpi(pid,"cputim")
$       procnam = f$getjpi(pid,"prcnam")
$       read/key='pid proc record
$       cputim1 = f$integer(record - f$extract(0,8,record))
$       cpuused = cputim2 - cputim1
$ !     write out the process names in a neat table.
$       a = ". . . . . . . . . . . . "
$       l = f$length(procnam)
$       a[0,l+1] := "''procnam'"
$       write sys$output a,cpuused
$       goto getuser
$
$ endlist:
$       close proc
$       if alrunning .eqs. "no" then goto finish
$       answer := Y
$       ans = f$extract(0,1,answer)
$       if ans .eqs. "y" then goto killem
```

```
$       if ans .eqs. "Y" then goto killem
$       write sys$output "processes running ALL-IN-1 not stopped "
$       goto finish
$
$ killem:
$       write sys$output "all processes running ALL-IN-1 are now being stopped"
$       kill = "$oa$lib:SMSTOP"
$       kill 'alimage
$
$ finish:
$       delete alusers.tmp;
$       write sys$output "ALL-IN-1 successfully shut down"
$
$ notice2:      ! Prepare notice program saying ALL-IN-1 is down
$ !
$       OPEN/WRITE Z OA$DATA:SMSHUTDN.TMP
$       WRITE Z "ALL-IN-1 went down for maintenance at ''fintime' "
$       WRITE Z "and is down for ''downtime'"
$       CLOSE Z
$
$       SET PROTECTION=(W:RE) OA$DATA:SMSHUTDN.TMP
$       set protection=(s:rwed) OA$DATA:SMSHUTDN.TMP;
$       purge oa$data:smshutdn.tmp
$       goto goback
$
$ nopriv:
$       set proc/priv =('curpriv')
$       write sys$output -
"you do not have the privileges required to shut down ALL-IN-1"
$       goto goback
$!
$alreadyshut:
$       exit
$
$ goback:
$       exit
$ !     ===========================
$ !     A 1 S H U T D O W N . C O M
$ !     ===========================
$ !
$ !     Utility to shut down ALL-IN-1 for system maintenance
$ !
$ !     written by Catriona Cooke      date: sep/oct 1984
$ !
$ !     modified by Gregory N. Brooks  date: dec/jan 1985/1986
$ !     modified by Gregory N. Brooks  date: March 6, 1986
$ !
$ !
$ !     This procedure has been modified to be used to shutdown All-in-1
$ !     without needing to be run interactively.
$ !
$ !     mnt = length of time before ALL-IN-1 is shut down
$ !     downtime = length of time before ALL-IN-1 comes back up
$ !
$ !     It redefines the logical OA$IMAGE to point to a basic program
$ !     (which will be run instead of ALL-IN-1 by anyone trying to get in to
$ !     ALL-IN-1 while the shut-down is in progress) and it asks people to
$ !     log out of ALL-IN-1.
$ !
$ !     All processes still running All-in-1 are deleted when shutdown is
$ !     complete.
$ !
$ !     AFTER SHUTDOWN
$ !     --------------
$ !     The following need to be reset by hand
$ !       define/system OA$IMAGE OA$LIB:A1 (or wherever the image is)
$ !       ASSIGN/SYSTEM "RUNNING" OA$MTISENDERCONTROL
$ !       ASSIGN/SYSTEM "RUNNING" OA$MTIFETCHERCONTROL
$ !
```

PATCHES FOR ALL-IN-1 VERSION 2.0

Gregory N. Brooks
Washington University
St. Louis, MO.

There are a number of patches for All-in-1 version 2 available thru Digital Software Services. However, you will not be receiving them automatically. You must ask for the patches or Software Services must recommend you install the patch (or patches) to correct a problem you are encountering.

The patches are available from the Atlanta office but they must be obtained through your local Software Services representative(s). They copy them from off on of the Atlanta VAXes on the Digital Network and send you a copy on the proper media.

The patch kit I received also contained an updates file cabinet verification program. Included with this new program was a description of why a file cabinet verification program was needed and some helpful info on when to use the file cabinet verification program.

The patches I obtained correct the following problems:

Displaying the output from a DATATRIEVE procedure using an FMS form through the ALL-IN-1/DATATRIEVE interface failed.

VMS 4.2 caused a problem with the Electronic Messaging Fetcher. The fetcher looped if a remote mail message containing an attachment was received, and the message was not delivered.

Using a single or double quote in the Location field of the Schedule Meeting form caused ALL-IN-1 to crash. The patch allows names to be entered with accent marks, indications of ownership, or other special punctuation in the Location field.

NOTE: The patch applies to ALL-IN-1 on both VMS V3 and V4.

The files included with the patch are as follows:

Filename	Description
OAFLO.OBJ	Corrects DATATRIEVE problem
OAFBT.OBJ	Corrects DATATRIEVE problem
OADTR.OBJ	Corrects DATATRIEVE problem
OAMTIRDM.OBJ	Corrects VMS V4.2 problem
OACMMA.OBJ	Corrects TIME MANAGEMENT problem
OACMME.OBJ	Corrects TIME MANAGEMENT problem
OASTR.OBJ	Corrects TIME MANAGEMENT problem
OAMESS.OBJ	Patch version identifier
PATCHINS.COM	Command procedure used to install patch
PATCHINS.DOC	Documentation to be used to patch V2.0

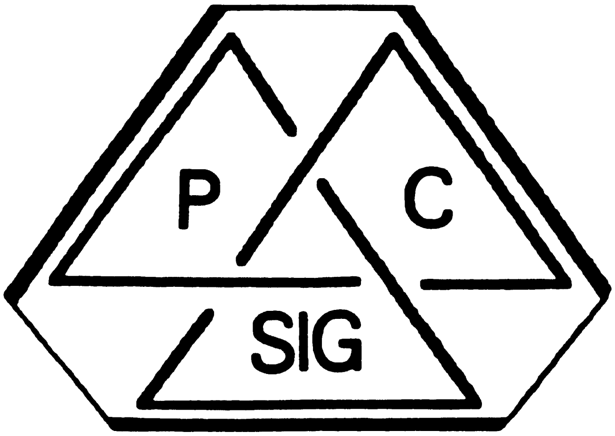
The following is a listing of the save sets of the patches I received. The first save set is for the file cabinet verification program. The second save set contains the patch kit files to correct the problems listed above.

```
Listing of save set(s)

Save set:          A2REPAIR.BCK
Written by:        WINKLEMAN
UIC:               [011,130]
Date:              21-FEB-1986 09:01:29.29
Command:           BACKUP/VERIFY FCBREPAIR.MEM,FILECAB.TXT,
FINDFILE.OBJ,SMFCVRPH1.OBJ,SMFCVRPH2.OBJ,SMFCVRPH3.OBJ,
SMFCVRSCH.COM MFAO:A2REPAIR.BCK
Operating system:  VAX/VMS version V4.2
BACKUP version:    V4.2
CPU ID register:   01388DDA
Node name:         MDVAX3::
Written on:        MFAO:
Block size:        8192
Group size:        10
Buffer count:      3

[WINKLEMAN.A1]FCBREPAIR.MEM;2
[WINKLEMAN.A1]FILECAB.TXT;10
[WINKLEMAN.A1]FINDFILE.OBJ;4
[WINKLEMAN.A1]SMFCVRPH1.OBJ;4
[WINKLEMAN.A1]SMFCVRPH2.OBJ;5
[WINKLEMAN.A1]SMFCVRPH3.OBJ;2
[WINKLEMAN.A1]SMFCVRSCH.COM;5

Total of 7 files, 274 blocks
End of save set


Save set:          A1PATCH.BCK
Written by:        WINKLEMAN
UIC:               [011,130]
Date:              21-FEB-1986 09:02:05.42
Command:           BACKUP/VERIFY A1PATCH.DOC,OACMMA.OBJ,OACMME.OBJ,
OADTR.OBJ,OAFBT.OBJ,OAFLO.OBJ,OAMESS.OBJ,OAMTIRDM.OBJ,OASTR.OBJ,
PATCHINS.COM,PATCHINS.DOC MFAO:A1PATCH.BCK
Operating system:  VAX/VMS version V4.2
BACKUP version:    V4.2
CPU ID register:   01388DDA
Node name:         MDVAX3::
Written on:        MFAO:
Block size:        8192
Group size:        10
Buffer count:      3

[WINKLEMAN.A1.EXPAND]A1PATCH.DOC;2
[WINKLEMAN.A1.EXPAND]OACMMA.OBJ;3
[WINKLEMAN.A1.EXPAND]OACMME.OBJ;1
```

```
[WINKLEMAN.A1.EXPAND]OADTR.OBJ;1
[WINKLEMAN.A1.EXPAND]OAFBT.OBJ;1
[WINKLEMAN.A1.EXPAND]OAFLO.OBJ;1
[WINKLEMAN.A1.EXPAND]OAMESS.OBJ;1
[WINKLEMAN.A1.EXPAND]OAMTIRDM.OBJ;1
[WINKLEMAN.A1.EXPAND]OASTR.OBJ;1
[WINKLEMAN.A1.EXPAND]PATCHINS.COM;12
[WINKLEMAN.A1.EXPAND]PATCHINS.DOC;6

Total of 11 files, 292 blocks
End of save set
```

If you are experiencing any of the problems this patch is touted to cure, need the updated file cabinet reorganization program, or just would like to have them in hand before the problems occur, I would suggest you contact your Software Services representative right away. It takes about two weeks (at least in my case) to get the patches copied from Atlanta and to your door.

Personal Computer Special Interest Group



Newsletter

PC

Personal Computing Special Interest Group Steering Committee

Chairman

Barbara Maaskant
UT Health Science Center
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-7351

Symposia Coordinator

Rick Eliopoulis
5258 Vickie Drive
San Diego, CA 92109
(619) 225-7867

DECmate Working Group Chair

Cheryl Johnson
Grinnell College
Noyce Computing Center
Grinnell, IA 50112-0813
(515) 236-2570

Pro Working Group Chair

Thomas R. Hintz
U. of Florida
IFAS Computer Network
1022 McCarty Hall
Gainesville, FL 32611
(904) 392-5181

Rainbow Working Group Chair

Lynn Jarrett
Union Tribune Pub.Co.
P. O. Box 191
San Diego, CA 92108
(619) 299-3131 (x1130)

PCXX Coordinator

Frederick G. Howard
Eastman Kodak Company
901 Elmgrove Road D345-LP
Rochester, NY 14650
(716) 724-5331

Session Notes Editor

Alan Bruns
Allied Electronics
401 E. 8th Street
Fort Worth, TX 76102
(817) 336-5401

National LUG Organiation Rep

Pierre Hahn
SUNY HSC-T10-02808101
Stony Brook, NY 11794
(516) 444-1362

PC Librarian

Ron S. Hafner
Hafner and Associates
P. O. Box 2924
Livermore, CA 94550
(415) 449-4178

Campground Coordinator

Jim Wilson
National Tech. Inst. for the
Deaf / RIT
Box 9887
Rochester, NY 14623
(716) 475-4178

RX/RSX Liason

Peter Flack
Computer Sciences Corp.
P. O. Box 12233
Research Triangle Park, NC
27709
(919) 541-4669

Members-at-Large

Jim Christine
SPSS, Inc.
1815 S. Coyler
Berwyn, IL 60402
(312) 329-3580

Russ Wertenberg
Sandia National Labs
Div 8352
Livermore, CA 94550
(415) 422-2663

Michael Bowers
University of California
Animal Science Department
Davis, CA 95616
(916) 752-6136

Caroline M. Mack
9007 Mears Street
Fairfax, VA 22031
(703) 280-4404

Digital Counterparts

DECmate

Ron Gemma
Digital Equipment Corporation

PRO

Lynn Olsen
Digital Equipment Corporation
146 Main Street
ML21-2/U12
Maynard, MA 01754

Rainbow

Katrina Holman
Digital Equipment Corporation
LJ02/I3
30 Porter Road
Littleton, MA 01460
(617) 486-2451

Table of Contents

PC SIG Steering Committee
Submitting Articles to the PC SIG Newsletter
From The PC SIG Chair
Contact List
Disk Dangers -- You Get What You Pay For
Hanging In There
Orphan Computers
Trivia Questions at DECUS - Big D

Rainbow

From the Working Group Chair, Lynn Jarrett
New Products
Wish List
From the DEC Counterpart, Katrina Holman
Rainbow Users Groups
Articles:
    Hints, Tricks, and Curiosities
    The Rainbow Workbench
    The Tech Corner
    Hardware News
    Public Domain Software Summaries
    DEC Rainbow Winchester Disks Technical Summary
    BSSes
    Word Perfect and the Rainbow
    Product Review: DM110 Hard Disk Tape Backup System
    Interfacing DEC Rainbow to Laboratory Equipment

DECmate

From the Working Group Chair, Cheryl Johnson
New Products
Wish List
Articles:
    Can YOU Help These People??
    Going From VAX to DECmate... In a hurry

PRO

From the Working Group Chair, Thomas Hintz
New Products
Wish List
Articles:
    Memory Expansion for the PRO-350
    PRO 350 Memory Modification
    Needed! Updates for the PRO Software List
    Background Printing Under P/OS -- 2nd Installment
    Serial Communication Between PRO 350 and CAMAC
    Fortran Graphics Application of PRO 350

Submitting Articles to the PC SIG Newsletter

The PC SIG newsletter serves as your means to share information related to personal computing hardware and software with other members of the SIG. We depend on your contributions for survival. If you have a tutorial or item of general interest, a new and better way to do something, a routine you developed and are willing to share, a related enjoyable story, etc., we would like to publish it in our newsletter. The PC SIG newsletter is presently published four times a year. With sufficient contributions we hope to increase the frequency with which we publish. This depends on you.

There are several ways you can submit your articles:

- o The PC Postscript form on the back of this newsletter.
- o Mail a Camera-Ready Hard-copy
- o For those of you with access to DCS send articles to username MAASKANT or JOHNSON_C.
- o Floppy Submissions should be on RX50's in WPS for the DecMate, ASCII, Select, WordPerfect, WPS80 or WPS+ for the Rainbow.
- o Files can also be uploaded to the FIDO bulletin Board: (315) 589-7361 Hitchiker's Guide 107/23 Williamson NY Fritz Howard
- o For those of you on Compuserve I can be reached as "Barbara Maaskant 70007,2263".

Contributions can be sent to:

(Interim Editor)
Cheryl Johnson
Grinnell College
P.O. Box 805
Grinnell, IA 50112-0813
(515) 236-2570

Barbara Maaskant (SIG Chair)
UTHSCSA-Computing Resources
7703 Floyd Curl Drive
San Antonio, Texas 78284
(512) 691-7351

Fritz Howard (PCXX Coordinator)
Eastman Kodak Company
121 Lincoln Ave.
Rochester, N.Y. 14650
(716) 263-2363

From The Chairman

Barbara Maaskant
University of Texas Health Science Center at San Antonio

By the time you read this The Spring Symposia will have come to an end. A lot has been accomplished by the SIG since December and most of these are manifested in Symposia as special activities, sessions presented, information collection and volunteer solicitation. The complexion of several standard activities changed and several new events took place.

The PC MAGIC session, first held in Anaheim, was repeated in Dallas. This was a very enjoyable and successful endeavor and will probably be a regular event. Part of the extreme success of MAGIC is due to our excellent panelists. Ted Needleman (Contributing Editor for Hardcopy Magazine) made a special effort to get to the Dallas Decus to once again serve as a panelist. Ted also wrote a very nice article about the SIG in the January Issue of Hardcopy. As a result, I received almost 100 requests for additional information, several returns on the volunteer form and one personal visit. I thank Ted for his enthusiastic support.

New products are always priority news. We have been acutely aware of the imminent announcement of the PCXX and really hoped for a Dallas debut. Although that was not possible the SIG has received strong support from Digital in terms of planning. This has enabled us to identify a new Working Group Chairman and begin support endeavors for our newest members. Fritz Howard, who many of you remember as PSS Coordinator, Campground Coordinator and Librarian has accepted this position. Fritz has done a great job in all the positions he has held and his professionalism and thorough organization is certain to make the new working group a success.

The SIG will also be supporting IVIS an "Interactive Video Workstation" which has a PRO 350 or PRO 380 as the microcomputer base. IVIS supports Training and Point-of-Information/Sale applications incorporating the the overlay of computer text and graphics on a video image. More information on all products within the PC SIG will be available in this and following issues.

A number of position changes within the SIG have taken place. Ron Hafner is the new Librarian. Ron did an excellent job on the first Public Domain diskette distribution and is organizing a permanent library. We are very pleased Ron has taken this job. Software submissions for the PC distribution, questions and suggestions should be directed to Ron. Jim Wilson is the New Campground Coordinator. Jim previously served as Member at Large and agreed to take on Campground Coordinator duties shortly after Anaheim. Those of you who were at Anaheim and Dallas know how much energy and creativity Jim puts into a job and how grateful we are to have him as a volunteer. The Campground is the focus of PC activities at Symposia and volunteers are always needed.

Sometimes your name only gets mentioned because you change jobs. I would like to mention Katrina Holman because her interface to the PC SIG has not changed. Katrina is the Rainbow Counterpart to the PC SIG from Digital and donates a lot of her time to increasing the success of the SIG. Those of you who were at Dallas know how supportive Katrina is. For those of you who don't get an opportunity to attend symposia she has been available on the phone. The PC SIG really appreciates having Katrina on the steering committee.

Several changes have occurred regarding the PC Newsletter, as well. We will attempt to publish more often. I have received many requests for a PC section in every issue rather than just quarterly, as we are presently contracted to do. The job of newsletter editor is VERY difficult and time consuming. It is a major compliment to Carolyn Mack (who has been editing our newsletter through March) that readers would like to see more. A successful newsletter is dependent upon your submissions. There are new instructions for the submission of articles in this issue. Please help us by sending your articles. This issue was edited by Cheryl Johnson. Cheryl, as you know, is the Decmate Working Group chair and is serving as editor on an interim basis. Cheryl's efforts and fine job in getting this issue together are sincerely appreciated. Anyone interested in the Editor position or that of Associate Editor should contact Cheryl.

There are three new forms in the newsletter. The PC SIG Volunteer Form, PC Postscript Submission Form and the Contact Sign Up Form. The Postscript is for short submissons, responses or comments about articles and letters to the editor you would like placed in that section. This is a way for you to share those little items you thought did not quite justify an article. This is also a way to ask for help. If you want to help your fellow SIG members please sign up to be a Software Contact. It is not time comsuming and provides a service to the SIG as well as a means to pick up extra information and meet people with a common interest.

Lots more went on at Dallas and I will only mention a few items. A first was what will hopefully be a new service at the Lug level. RainboWare debuted on a MicroVax in the Exhibit Hall Rainbow Booth. From the Campground individuals could dial in for software inquiries. A FIDO node was again set up in the campground. This is very enthusiastically received capability and allows us to communicate with Sysops and others who cannot attend Symposia over the FIDO network. We shared a suite with the OA SIG in the Loews Anatole Hotel. This allowed both SIGS to participate in more activities. The SIG Reception was Thursday evening and the theme "PC Roundup" was most appropriate.

The PC Public Domain Software distribution at Symposia is still alive. We will be posting procedures on several ways to submit software and the required release form. We also are hoping to have something in the way of an aknowledgement for contributors in time for the San Francisco distribution. It has been very well received. At Dallas 2000 diskettes were sold.

All activities, new projects and services are the result of a lot of time and effort on the part of our volunteer group and counterparts. Many of the new approaches and changes were a result of your suggestions. Please continue to communicate with me or any member of the steering committee regarding your suggestions and ideas for strengthening the SIG.

Barbara Maaskant

CONTACT LIST

If you have any questions about the hardware or software listed, feel free to call the persons below. Questions should be the sort that can be answered over the telephone in a few minutes. Contacts cannot be expected to act as consultants nor can they be expected to return long distance phone calls. (Volunteers who would like to be added to the contact list should contact the editor with their name, specialties, and a phone number.)

ALL-IN-1, V. 1 & 2	Andrew L. Wax	(212) 701-5231
BASIC	Guy Stefan-Romano	(415) 753-5581
COMMUNICATIONS/PRO	John R. Lynn	(714) 620-6667 (after 7 PM PST)
CORVUS OMNIDRIVE	Kenneth Lemay	(219) 483-0001
DBASE	Guy Stefan-Romano	(415) 753-5581
DBASE II	Paul Bottone	(516) 928-0933 (after 7 PM EST)
DBL (DIBOL TYPE)	Philip Rice	(503) 369-2706
DECMATE	Cheryl Johnson	(515) 236-2570
	Gary Swarthout	(607) 974-3152
DECSPELL	Cheryl Johnson	(515) 236-2570
ECONOMETRIC PROGRAMMING	Jeff Mackie-Mason	(617) 253-8717
FCS-EPS	Mark Lewis	(212) 563-5656
GRAPHWRITER	Lynn Jarrett	(619) 293-1130
HARD DISK/Rainbow	Kenneth Lemay	(219) 483-0001
HARDWARE/PRO 350	John Pratchios	(703) 385-5880 x3887
HARDWARE/Rainbow	Bob Catt	(717) 794-2430
LOTUS 1-2-3	Lynn Jarrett	(619) 293-1130
	Joseph L Marks	(404) 875-9211
	Miles Bridgford	(312) 496-4582
LOW LEVEL I/O ON PRO	John R Lynn	(714) 620-6667
LVP-16 PLOTTER	Lynn Jarrett	(619) 293-1130
MASS-11	Lynn Jarrett	(619) 293-1130
MEDICAL OFFICE APPL/DM	J Kaufman	(201) 937-7816
MEMORY ADDITIONS	Tom Tugman	(703) 280-4404
	Caroline Mack	(703) 280-4404
MULTIPLAN	Guy Stefan-Romano	(415) 753-5581
P/OS	Dr. Homer Baker	(602) 251-3830
PRO APPLICATIONS	Tom Hintz	(904) 392-5180
PRO OFF WORKSTATIONS	Carol Britton	(805) 277-4808
PRO SERIES	Tom Hintz	(904) 392-5180
ROBIN (VT180)	John Comella	(617) 897-5922
SAMNA II	P Chmielewski	(203) 777-5537
SCIENTIFIC PROGRAMMING	Jeff Mackie-Mason	(617) 253-8717
	Thomas R LeBone	(603) 880-9097
STATISTICAL PROGRAMMING	Jeff Mackie-Mason	(617) 253-8717
TK!FILES	Miles Bridgford	(312) 496-4582
TK!SOLVER	Miles Bridgford	(312) 496-4582
TURBO PASCAL	Thomas R LeBone	(603) 880-9097
UCSD PASCAL	Guy Stefan-Romano	(415) 753-5581
WORDPERFECT	Caroline Mack	(703) 280-4404
WORDSTAR	Guy Stefan-Romano	(415) 753-5581
WPS	Cheryl Johnson	(515) 236-2570

Disk Dangers - You Get What You Pay For
By Theodore Needleman

Now that most of you know about "Needleman's Law," it's time that you found out about "the Needleman Maneuver." This maneuver consists of removing one's walking appendage from one's speaking orifice.

Back in October's column, I mentioned that I had heard of some problems using disks with hub rings. Boy, did I get feedback on this one! Anxious readers wrote in to ask if the brand of disks they were using were okay, and several disk manufacturers contacted me about their brands. As the reason for writing this column is to pass on information that I feel you'd want to know, I make no apology for pointing out a potential problem. At the same time, my intent was not to start a panic. I've investigated this a bit further, and my findings to date indicate that the problem seems to be confined to real "cheapo" disks. For some reason, some of these seem to damage the spindle gripper on the RX50. Mind you, all of this is hearsay. I've never actually seen a drive that has been damaged by using a particular brand of disks! But something else to keep in mind is that in this life, you get what you pay for (if you're lucky). If you are using good quality disks, from a reputable manufacturer, odds are that they will not damage your drive, hub rings or not.

For those of you who have questions concerning disks-whether or not they be on hub rings-how disks are made or any other disk-related questions, there is someone you can call. Mike Thomas, who is with X'LNT Diskette Products, has kindly offered to field such questions. His number is (214) 783-6950 (in Texas), and he seems quite knowledgeable on the subject. I've never used his company's disks, so I can't endorse them, but I applaud both Mike and his company for their willingness to share their knowledge.

While I'm on the subject of reader mail, I've gotten a few annoyed notes about things I've mentioned here that are not available at the time the column comes out. I know I've harped on "editorial lag" many times, but once more into the breach, dear friends. The editorial lag is now running about three months (it's now early December). I know that you'd like the Corner to be as topical as possible, so would I, but it does take time to put out a magazine. One result of this lag is that sometimes things that I expect to happen, such as Don Brauns' clock board being shipped in mid-October, hit a snag. Other times, news such as DEC's $750 5-Mbyte Disk Upgrade, has been withdrawn by the time it gets mentioned. Other than not mentioning anything that might be time-sensitive, there's not a lot I can do. And I would just as soon wind up with egg on my face than miss tipping you off on something good at the earliest possible time.

Speaking of tips, looks like some interesting things are happening. The tape drive from Duncan MacDonald Inc., (714) 495-4418, has shown up, and while I've not yet installed it (like the IDrive, it uses the second floppy-controller plug), it looks well constructed and attractive. I'm also waiting for the software to use it with MS-DOS (it came with CP/M software). Duncan tells me he'll have it for me this week, so I might be able to say more next month. While I'm at DEXPO/West, I also expect to see a product called RB-Link. I'm told that this can give a fair measure of IBM compatibility to the Rainbow, though it does sound a bit expensive. It's being put out by a company called DiskTech One, and I'm hoping to get one to review.

The software emulator, Emulate-84, has still not shown up here, and I didn't feel like calling again. The people at Operating Scientist have my phone number (and my signed license agreement). I do have an alpha-test version of someone else's emulator. This one, called at the moment DIBEM, will either be put in the public domain or sold as "shareware." I have just received the second release, and (as usual) am behind in trying it out. For the latest news on this, call the Rainbow Corner Fido, (914) 425-2613, and check the bulletins when you sign on. The board has been up for about three weeks now, and has had several hundred callers! I've also duplicated about 300 disks for the LCTerm offer I mentioned in October's column. The response really knocked me out, and I'm already thinking about the next offer-maybe the IBM emulator, if it goes into the public domain (and of course, if it works).

That's it for now; got to run. I'll see you next month--or sooner if you call the Rainbow Corner Fido.

Reprinted with permission of HARDCOPY Magazine, Sedlin Publishing, Inc., P. O. Box 759, Brea, CA 92622

Hanging In There
By Jan Maaskant

For those of you who have been plagued by the inevitable phone call interrupt that aborts your long distance computer session, there IS hope. One of the first questions computer buffs ask after they purchase Call Waiting is how to avoid being knocked off the line when they are using their modem. (We know some of you have been forwarding your calls to TIME, the Office and the ZOO).

There is a better solution. The * 7 0 feature cancels the call waiting option for one phone call only. This is how it works. You enter your local communication program and invoke the telecommunication option. Immediately before dialing your host number press * 7 0. Wait for the dial tone and then dial the host. Any calls which under normal Call Waiting would interrupt your conversation are given a busy signal. You have this option if you are fortunate enough to live in a Bell System area and are serviced by an E.S.S. office. The capability has received little to no advertisement and is presently available only in some TEST areas. It is, however, an add on software modification for all E.S.S. offices so you can get it. If you want this feature CALL and make your desire KNOWN. Rumor has it that Bell is considering dropping the feature for lack of interest.

Orphan Computers
By Jim Wilson

Hi folks! In addition to my many other hats, I wear one related to helping handicapped people. I am the current Chair of a group known as ROC SIGCAPH. This stands for the ROChester chapter of the Special Interest Group on Computers And the Physically Handicapped. It is a part of ACM, and is a national organization. I am the Chair of it's first local in Rochester, NY.

We formed about two years ago, and are dedicated towards helping handicapped people in two major ways: 1) to assist handicapped people live better lives with the help of computers; 2) to help handicapped people find careers in data processing related fields. We also strive to educate the "outside world" to the needs of handicapped people.

We just started a new project that I would like to tell you about. It is our Orphaned Computer project. We are soliciting the old unused computer that you have stuffed away in your closet. The one that the family outgrew, or that you couldn't upgrade, or (heaven forbid!) was not a DEC! We are not asking for monetary donations, or for anyone to go out and buy new computers. We merely want to put some old klunkers back to work.

These computers will be handed over to organizations that assist the handicapped. These groups will then be given instructions from our volunteers on how to use their new toys. One such group is the Monroe Community Hospital. It currently has about 25 people very eager to learn more about computers, but their budget does not allow for even one. I have given a seminar there and know that these people have a real desire to learn about computers.

If you are the step-parent of one of these orphaned computers and would like to have someone else adopt it, please contact me. We are a not-for-profit organization, and thus I hope you could get a tax write-off for this gift.

I can be reached at:

James W. Wilson
National Technical Institute for the Deaf
Rochester Institute of Technology
P.O. Bos 9887
Rochester, New York 14623-0887
(716) 475-6241

Computers or information can be sent directly to me, or to the SIGCAPH mailing address (in my attention) which is:

Rochester SIGCAPH
P.O. Bos 92487
Rochester, New York 14692-0487

Dallas DECUS Symposia Trivia Questions

Editors Note: Next month we will have the answers to the trivia questions.

DECUS
Trivia Questions
for the
DEC PROfessional Minicomputer

1. What is the release date and version number of P/OS that was the first to require 512KB memory?

2. How many RED diagnostic lights are on the back of the PROfessional system unit?
3. What are the letters printed over the GREEN diagnostic light on the back of the PROfessional system unit?
4. Describe/draw the figure indicating the BRIGHTNESS control knob for the VR201 and VR241 monitors?
5. Describe/draw the figure indicating the CONTRAST control knob for the VR201 and VR241 monitors?
6. The PROfessional power switch uses 0 and 1 for ON/OFF. Which number is on the top?
7. What color stripe is used to designate the function key label strip for a PROfessional?
8. How many backplane slots are found in a PROfessional 325, 350 and 380?
9. What two special function key definitions are found on the DECmate, RAINBOW and PROfessional function key label strip, and are the same keys on the DECmate and RAINBOW but different on the PROfessional? What keys are they?
10. What color(s) are used on the FRAGILE stickers stuck to the top of the hard disk for a 5MB and 10MB drive in a PROfessional?
11. What is the figure used on the FRAGILE stickers stuck to the top of the hard disk drives for the PROfessional?

Trivia Questions
for the
DEC Rainbow Computer

1. How many keys are on a standard Rainbow keyboard?
2. In what order are the printer, video, and comm ports positioned from left to right on the back of the Rainbow?
3. How many languages come up from which to make keyboard selection?
4. How many baud rate selections are there in either XMT or RCV mode in the MODEM set-up in and name them?
5. What are the AC input selections for the Rainbow?
6. What capabilities does the extended communications option provide?
7. What is the home row key height on the keyboard?
8. What is the rate of transfer on floppy drives?
9. What is the weight of the RX50 drives?
10. What's the maximum number of characters you may store in the Answerback message in the major heading of set-up mode?

Trivia Questions
for the
DEC DECmate Computer

1. What color stripe is used to designate the function key label strip on a DECmate?
2. What is the model number for the DECmate II?
3. What is the model number for the DECmate III?
4. With what version of WPS did DECspell come with?
5. What WPS option does WS200 not have?
6. Name the DECmate terminal mode options available (exact key words)?
7. On power-up, what letters are displayed on the screen?
8. How many pages of text will a RX50 diskette hold according to Digital?
9. How many keys have the Gold writing on the fronts of them on a standard DECmate Gold Key keyboard?
10. When information is being read off of the hard disk, what color lights up on the front of the hard disk?

RAINBOW

From the Rainbow Working Group Chair, Lynn Jarrett

RAINBOW WORKS

As time goes by, it seems that all the networking of the users is beginning to pay off. Some issues have been resolved, some are in the works and there are still some that need to be pursued.

One of the problems I've seen for quite some time is that of people not speaking up. There are thousands and thousands of Rainbow owners out there, but when it comes down to voicing opinions, I don't get a lot of opinions in writing. That's SO important. When I take certain issues to DEC, they want to know how many people are voicing these concerns. Please write to me. I want to represent the majority--not only the minority.

The wish list has been a help for everyone who needs to address the problems of the users. I very much appreciate the input of all of you, so keep it coming. DEC addresses some of the problems, but a lot of the issues are also addressed by third party interests.

As of this writing, my pet peeve, the Lotus update for the Rainbow, has not happened. It's interesting that I receive so many calls on just this issue alone!! Many of you are aware that I have had meetings with Lotus and DEC regarding this subject, and several of you have written letters to these companies and requested something be done, as well. I hope that this will be resolved very soon.

Several of the software companies have sold varied packages to Rainbow users and now refuse to offer updates. It seems that they will only update the IBM compatible software. Sound familiar? Some of these packages include: Lotus, dBase II and III, Graphwriter, and the PFS series, to name a few. It just doesn't seem the fair thing to do.

Many of you have complained about support at Atlanta. Rightfully so, as far as I am concerned. They are doing a better job since DECUS Anaheim in December. The manager of the support center was there and realized that people were very unhappy with the lack of support and lack of expertise in Atlanta. I truly believe that DEC is working on making the support center better for all us. I'm watching this fairly closely myself.

Since DEC has taken their small machines out of the retail stores and is seemedly gearing toward the corporate customers with networking, etc., my concerns for their support of the Rainbow users out there are greater than they were. When the PC SIG Executive Steering Committee met with the Personal Computing Systems Group of Digital in Massachusetts recently, we were told that Digital would not let the Rainbow users down. I was glad to hear that statement, and I will continue to remind DEC that we're still out here.

Third party people are doing a lot of work on Rainbows. Disc Tech One has come out with the RB-Link and this will answer a lot of the users' needs for IBM compatibility. Of course, it's like buying a whole new PC, but it does offer compatibility if that's what you're after. Their address is:

Disc Tech One
849 Ward Drive
Santa Barbara, CA 93111
805-965-3535

I wouldn't expect DEC to come out with a clock for the Rainbow now since that's being done by third parties also. Check the following address for that item:

Rainbow Clockworks
4220 Commercial Drive, Suite 5
Tracy, CA 95376
209-836-3349

As far as a TPU, Duncan Mac Donald has come out with the DM110. (See Product Review in this newsletter).

Microsystems Engineering Corporation will be bringing out the Rainbow version of their database software package, MASS 11 Manager, this month. I just received the beta test versions for the Rainbow and the Vax and will be testing this software immediately. It should make for an excellent database package. Their address:

Microsystems Engineering Corporation
2400 W. Hassell Road, Suite 400
Hoffman Estates, IL 60195

I've had several calls and many letters from international Rainbow users. DEC sold quite a few Rainbows in Europe, and there is quite an interest there in getting information about Public Domain software, software updates and hardware. I'm enjoying those contacts that I've made and hope to continue to be able to answer the inquiries from abroad. I've "met" some very interesting people.

I am looking forward to DECUS Dallas and will report to you from there anything new and exciting.

===

New Products

===

PC Connection Program

If you are looking for a way to connect your standalone personal computers, Digital has the solution today.

With Digital's PC Connection Program, you can link your Digital, IBM, IBM-compatible, and/or Apple personal computers together, using a combination of VAX/VMS and third party software.

The PC Connection Program offers you a choice of connection methods to move files back and forth between your VAX system and personal computers. Most of today's popular software applications can be supported by these connection methods.

Currently, the PC Connection Program includes any one of four terminal emulation products and a DECnet product. VTerm II* allows an IBM PC to emulate a VT100 terminal. Poly-COM* lets your Rainbow emulate VT102 and VT52 terminals. Poly-COM 220 and 240 allow an IBM PC to emulate VT220 and VT240 terminals. MacTerminal* lets an Apple Macintosh* emulate a VT100 terminal. With these products, you can connect, transfer, and share files and documents, access and send mail, and share peripherals. The DECnet product, DECnet-DOS, allows IBM PCs to act as end nodes on the DECnet.

To support the PC Connection Program, you may use any VAX from the MicroVAX through the VAX 8600. Personal computers are connected to the VAX through asynchronous ports.

RD31 20-Mbyte Disk Drive

Looking for more disk space for you Rainbow system? Want to move up to a hard disk? Digital's new RD31 20-Mbyte hard disk offers you twice the storage capacity and faster access to data for the same price as the RD31 10-Mbyte disk drive it replaces. With a 20-Mbyte hard disk, your Rainbow is even more versatile, reliable, and easy-to-use.

The RD31 is the first of Digital's new generation, half-height 5.25 inch-high disk drives. The new disk drive delivers higher performance and reliability, more storage in less space, and lower power consumption than a full height RD51 disk drive. It offers easy, low-cost migration to 20-Mbyte storage for Rainbow users previously limited to 10-Mbytes.

Increased Storage Capacity

The RD31 disk drive is ideal if you need to store and quickly access large amounts of data, run many applications, or use large demanding programs. Higher storage enhances performance of applications such as the Rainbow Office Workstation, DESQ*, WPS-PLUS, Lotus 1-2-3*, and dBASE III*.

If you already have a hard-disk Rainbow, an RD31 disk drive allows you to double the storage capacity of your RD51-based system. And it gives you 22 percent faster average access time.
If you currently own a floppy-based Rainbow system, this new product offers you the chance to economically boost system performance. Upgrading to an RD31 hard disk drive gives you greater storage potential and faster access.

A Snap to Install

This new disk drive is a snap to install. All you do is remove your old RD51, plug in the RD31, and load the supporting software. The existing disk controller and cable operate with the RD31 without modification. The RD31 needs no preventive maintenance and gives trouble-free operation throughout its long life.

Easy Expansion

In addition to being standard equipment with the Rainbow 100+ and Rainbow 190 systems, the RD31 is available as an add-on option for floppy-based Rainbow 100A and 100B models. It is offered as an upgrade option for the Rainbow 100+ and Rainbow 190 systems already equipped with RD51 10-Mbyte disks.

The RD31 can be used with CP/M* Version 2.0 or later and MS-DOS* Version 2.11 or later operating systems. CP/M uses 10MB, and MS-DOS uses the other 10MB. A new hard disk utility program is required and is included with the hard disk.

CD Reader on Rainbow, IBM Personal Computers

The CD (Compact Disk) Reader, Digital's revolutionary compact disk reader, is now available for your Rainbow 100 and IBM personal computers. Previously, this compact disk system was available only for use with the MicroVAX I and II and the VAXstation II.

The CD Reader provides you with a more effective way to distribute and use large amounts of information. Employing optical disk technology, the CD Reader allows you to retrieve and use enormous databases, catalogues, or historical records at your own workstation. This capability saves you the cost and inconvenience of dialup lines or requests to a central computer during peak hours as well as the problems involved in dealing with masses of paper or stacks of microfiche.

High Level of Compatibility

On the Rainbow and IBM personal computers, the CD Reader uses the same small, removable optical disk known as CDROM (Compact Disk Read Only Memory) as it does with the MicroVAX. This is made possible by utilizing Uni-File, a file format developed by Digital for use with the read only media. Digital is proposing that Uni-File be used as a standard to address the need for a common structure among CDROMs. Uni-File also provides a high level of compatibility that allows the distribution of the same disks even if your organization has a variety of workstations and personal computers.

The CD Reader with its CDROM compatibility can help you simplify your electronic publishing process and capitalize on the low cost of replicating disks in large volume. By utilizing CDROM technology, you can reduce technical information to disk form and research data more cost effectively, enhancing your use of your Rainbow and IBM personal computers.

Services and Database Publications

In addition to the CD Reader, Digital offers, as part of its overall CDROM Program, Application Development Services and CDROM Database Publications.

Application Development Services provide you with a means of migrating your data onto CDROM optical disks. These services include planning and consultation, preparing data, loading databases with retrieval software, and arranging for the mastering and replication of the compact optical disks.

A CDROM Database Publication is a CDROM optical disk containing MicroBASIS* search and retrieval software and a large, full-text, structured database. The database is "owned" by a commercial publisher and was previously available only online, or on paper, microfiche, or film. These database publications are available on a subscription basis, which includes quarterly update disks and continuing subscriber support and service from Digital. There are currently ten database publications available on CDROM.

The CDROM Database Publications are intended for use on MicroVAX I and II, VAXstation II, and the IBM PC/CST.

With the addition of the CD Reader, Digital offers you the most comprehensive collection of CDROM hardware, software, services, and publication, all under one file format, Uni-File.

==

*VTerm II is a trademark of Coefficient Systems Corporation.
*poly-COM is a trademark of Polygon Associates, Inc.
*MacTerminal and Macintosh are trademarks of Apple Computer, Inc.
*DESQ is a trademark of Quarterdeck Office Systems.
*Lotus 1-2-3 is a trademark of Lotus Development Corporation.
*dBASE III is a trademark of Ashton-Tate.
*CP/M is a trademark of Digital Research, Inc.
*MS-DOS is a trademark of Microsoft Corporation.
*MicroBASIS is a trademark of Battelle Development Corporation.

==

Reprinted with Permission of Perspective, Volume 4, Number 1; Copyright Digital Equipment Corporation.

==

RAINBOW WISH LIST

Following is the Rainbow Wish List that was recently submitted to Digital by the PC SIG. This list is compiled and prioritized from users submitting their Voting Forms and/or letters to Lynn Jarrett. The Voting Forms can be found in this issue. Please take the time to mail them with your comments to Lynn. Your input is greatly appreciated.

1. A new updated version of Lotus with a decent implementation of the new version. 132 column implementation.

2. Sell the Rainbow Winchester controller as a separate item with the controller cable, and Version 3.0 Winchester utility disk.

3. Print drivers for LN03 and more non-DEC printers on all DCS software.

4. Update dBase III. DEC is still 2 major revisions away from what IBM offers.

5. Larger memory upgrades and removable hard disks at reasonable prices.

6. Ability to share hard disks with several users.

7. A second COMM port that doesn't require you to disconnect the hard disk controller to put it in.

8. Come up with an ANSI.SYS device driver like that of IBM that would give people the ability to redefine the function keys. This could also allow users the use of extended sequences for color or monochrome shading that DEC cold add as well as give the users the ability to run many IBM programs on their Rainbows.

9. A new ROM upgrade for the Rainbow that would include support for an IBM PC compatible character set. (This would allow Rainbow users to easily run all the programs that take advantage of all the IBM PC grahic characters).

10. Implement the Print Screen key so that it works in all applications.

11. MS-DOS 3.1

12. A cheaper board for 8087 support.

13. Ability to update from a 100A to 100B inexpensively.

14. More software packages!

15. Larger print buffer built into printers.

16. Ability to format a single-sided IBM-PC disk under MS-DOS.

17. Rainbow LAN without the need for a VAX.

18. Ability to upgrade to the PCxx-IBM-PC AT compatible machine at a reasonable price.

19. Full-featured GW-BASIC compiler.

20. Coordination between local DEC Field Service offices and Atlanta Hotline Support

21. RBASE 5000, Framework, Thinktank and Borland's Sidekick in Rainbow versions.

22. Q-BUS for the Rainbow.

23. Terminal mode to support the additional keys of the LK201 keyboard rather than just beep.

24. TPU (now being supplied by third party).

25. Lap-top or portable Rainbows.

26. Rainbow Draw and Paint.

27. Take copy protction off of DCS and DDS software.

28. MS-DOS installable graphics driver similar to GSX-86.

29. DEC hardware and software sold in retail stores.

30. Concurrent MS-DOS.

31. Hardware Reset key and Halt key.

32. Monitor connection for CAD products.

33. ASSIGN command.

RAINBOW WISH LIST BALLOT

Use this ballot to show which items on the Wish List are most important to you. Put the number of the most important item on the list in space 1, the next most in space 2, etc.

1._____
2._____
3._____
4._____
5._____
6._____
7._____
8._____
9._____
10._____
11._____
12._____
13._____
14._____
15._____
16._____
17._____
18._____
19._____
20._____
21._____
22._____
23._____
24._____
25._____
26._____
27._____
28._____
29._____
30._____
31._____
32._____
33._____
34._____
35._____
36._____
37._____
38._____
39._____
40._____

Please add the following to the wish list:

____
____
____
____

Comments:

Name: ____________________________________

Company: __________________________________

Address: __________________________________
____
____

Work Phone: ________________ Home Phone: ________________

RETURN BALLOTS TO:
Lynn Jarrett
Union Tribune Publishing Co.
P.O. Box 191
San Diego, CA 92108
(619) 299-3131

Attention Stay-at-Home DECUS Members:
RAINBOW LITERATURE FOR THE ASKING!

By Katrina Holman
PC SIG Counterpart for Rainbow
Digital's Personal Computing Systems Group

Those of you who attend the DECUS symposia know that one of the benefits is getting the latest literature and certain special goodies, like buttons, bags, and posters. But what about those of you who don't come to symposia? Do you feel like you've been left out in the cold? Well, I hope to remedy that.

IF YOU ATTEND LOCAL USER GROUP MEETINGS, please let your chairman know which items from the list below and how many of each you would like, and have the chair send me the request for the whole group. My office is simply not equipped to fill individual literature and information requests for lots of people, so I would prefer to send a bulk shipment to user group chairs, who can then distribute the literature at their LUG meetings.

Early in April I sent a letter with all the information in this article and copies of the materials to the chairmen of all the Rainbow local user groups I was aware of. If I missed your group, please let me know.

IF YOU DON'T BELONG TO A USER GROUP, write to me directly about which of the materials listed below you want.

(Note: Digital sales offices can also order those items listed below that have an order code in parentheses.)

Rainbow Literature to Choose from:

- BIOS listing for MS-DOS V2.11.
- BIOS listing for CP/M-86/80 V2.1.

  Note: Digital plans to include both BIOS listings in the Rainbow Technical Documentation Set when those volumes are reprinted (unspecified date).

- Plastic bags with colorful Rainbow logo (EZ-B541745).
- Rainbow Product Guide, December 1985 edition. Unfortunately, this document (which I created especially for Fall '85 DECUS) has been so popular that it is now out of stock. If I hear from you that there is still a great demand for the Rainbow Product Guide---a comprehensive list of all hardware and software options Digital sells for the Rainbow, with succinct descriptions, order codes, and U.S. prices---I will consider reprinting it.

  In the meantime, the new DECdirect Personal Computer Software Catalog, Spring/Summer 1986, contains descriptions and ordering information for Rainbow, DECmate, and Professional software packages. Call 800-258-1710 (not me!) for more information.

- Rainbow Referral Software Directory, Jan. 1985 edition (EZ-WE9CS-01). Although this document is quite dated, many people find it useful as a starting point for finding third-party software for Rainbow.

* Back issues of Perspective: Digital's Personal Computer Newsletter.

Perspective Vol. 1, No. 2, July 83 (EZ-62009-02)
Perspective Vol. 2, No. 1, Jan 84 (EA-30229-76)
Perspective Vol. 2, No. 2, May 84 (EA-30230-76)
Perspective Vol. 2, No. 3, Oct 84 (EA-30231-76)
Perspective Vol. 3, No. 1, Jan 85 (EA-30232-76)
Perspective Vol. 3, No. 2, June 85 (EA-30233-76)
Perspective Vol. 3, No. 3, Nov 85 (EJ-27836-76)
Perspective Vol. 4, No. 1, Jan 86 (EJ-27837-76)

A SELECTED TABLE OF CONTENTS

Perspective Vol. 1, No. 2, July 83 (EZ-62009-02)
- IVIS Workstation, the Interactive Video Information System
- DECmate II Word Processing System
- Real-Time Interface Options for the Professional 300 series
- Technical Perspectives by Users
  - A Subroutine to Control Rainbow 100 Screen and LA50 Printer
    - Advantages to Rainbow Programmers of Calling Assembly Subroutines from a High-level Language like MBASIC-86

Perspective Vol. 2, No. 1, Jan 84 (EA-30229-76)
- DECmate II WPS V1.1 and DECmate II CP/M 2.2 V2.0
- PC Keyboard: A Standard for Quality
- Announcing PRO/Applications Starter Kit: PRO/Spreadsheet, PRO/Graph, PRO/Data Manager, PROSE Editor
- Digital Expands Rainbow 100 Line: New Operating System Versions MS-DOS V2.05 and CP/M-86/80 V2.0, 10 MB Winchester Disk, Using MDRIVE
- Application Software
  - RS/1: An Integrated Data Analysis System for Pro 350
  - NPL Information Management System for Pro 350
  - TK!Solver for Rainbow and Pro
  - Lotus 1-2-3 for Rainbow
- Technical Perspectives by Users
  - Creating Long Lists on Your DECmate Without Cutting and Pasting
  - Rainbow to Pro Transfer
  - Hints for Hackers: Setting LA50 Printer to Enhanced Printer Mode; Using Rainbow 100 as a Terminal
- Questions and Answers from Atlanta Customer Support Center

Perspective Vol. 2, No. 2, May 84 (EA-30230-76)
- Digital Printers: LA50, LA100, and LQP02
- DECUS
- Design Standards for Digital's Personal Computers
- Application Software
  - DESIGN GRAPHIX/Executive for Pro 350
  - ATHENA/Graph for Pro 350
  - SAMNA WORD II for Rainbow
  - PFS:FILE and PFS:REPORT for Rainbow
  - Multiplan for DECmate II
- Announcing New Products
  - Telephone Management System for Pro 300
  - RT-11 for Pro 300
  - Memory Adapter Option for Rainbow (to expand PC100-A memory beyond 256 KB)
  - Technical Character Set for Rainbow
  - Gold Key keyboard for Rainbow
  - WPS V1.5 for DECmate II
  - DECspell for DECmate II

  - The Mini-Exchange
- Technical Perspectives by Users
  - Using DECmate II to Renumber Lists
  - "Block Graphics" and Simple Line Graphics Using BASIC Program to Get Reverse Video
  - Plotting XY Graphs Using LOTUS 1-2-3 on Your Rainbow Personal Computer
  - Inside the Rainbow: A Conversation Between Chips

Perspective Vol. 2, No. 3, Oct 84 (EA-30231-76)
- Digital and PRO/Videotex Star in Volvo International
- Communicating with Your Personal Computer
- Network Services
- Announcing New Products
  - PRO/DECnet
  - The DECNA Ethernet Controller for Pro 300
  - PRO/VENIX
  - Rainbow Personal Presentation System using Graphwriter Software and Polaroid Palette Hardware
  - DECmate II Master Menu
  - Accessories and Supplies
- Application Software
  - PRO/Office Workstation
  - dBASE II for Rainbow and DECmate
  - MS-BASIC Compiler, MS-BASIC Interpreter, MS-Pascal Compiler, and MS-FORTRAN Compiler for Rainbow
  - Context MBA for Rainbow
  - The Digital Accounting System on the DECmate II
  - DECmate DBMS
- Technical Perspectives by Users
  - Using GSX-86 with Assembly Language on Your Rainbow
  - How to Call Assembly Subroutines from dBASE II
  - Care and Feeding of Your DECmate
  - Using Your DECmate II to Print Addresses on Envelopes
  - Using the List Document to Control Printed Output on Your DECmate II

Perspective Vol. 3, No. 1, Jan 85 (EA-30232-76)
- Computer Graphics
- Announcing New Products and Newsbriefs
  - Digital Receives ICP 200 Award
  - Professional 380
  - DECmate II Word Processor
  - WPS V2.0 for DECmate II and III
  - LQP03
  - Letterprinter 210
  - Rainbow MS-DOS V2.11
  - Rainbow ReGIS
  - Rainbow 8087 Numeric Data Coprocessor
- Application Software
  - GrafTalk for Rainbow and DECmate
  - GraphPlan for Rainbow
  - Matrix-Aids for Rainbow and DECmate
  - PROSE PLUS for Pro 300
  - Corgraph, Fonts, and Graphics for Pro 300
  - PRO/SIGHT for Pro 300
- Technical Perspectives by Users
  - DECmate Disk Maintenance Utilities
  - Exchanging Data Between DECmate II DBMS and WPS List Processing
  - Creating Numbered, Multiple Versions of Documents from Your Lists on DECmate

  - List Processing Time and Effort Savers for DECmate
- Questions and Answers from Atlanta Customer Support Center

Perspective Vol. 3, No. 2, June 85 (EA-30233-76)
- Digital's Electronic Store
- Integrated Laboratory Automation with Pro 300
- Managing Information in Today's Office
- Announcing New Products
  - Rainbow 190
  - Rainbow Office Workstation software
  - WPS-PLUS/Rainbow
  - DECnet-Rainbow
  - DECnet-DOS for IBM personal computers
  - DECmate Office Workstation
  - LCP01 ink-jet color printer
  - LN03 laser printer
  - LA210 Letterprinter
- Application Software
  - AutoCAD 2 for Rainbow
  - dBASE III for Rainbow
  - Symphony for Rainbow
  - Overhead Express for Rainbow and DECmate
  - Synergy for Pro 300
  - PHOENIX-PRO Project Management
- Technical Perspectives by Users
  - Writing Large Programs with Pro Tool Kit
  - Creating Tables With Multiple Columns in WordStar on Rainbow
  - User-defined Keys in DECmate Word Processing

Perspective Vol. 3, No. 3, Nov 85 (EJ-27836-76)
- The VAX Family
- Announcing New Products
  - MicroVAX II
  - VAXstation II
  - DESQ for Rainbow 100+
  - PRO/DBA
  - PRO/Associate
  - Graphics Terminal Emulator V2.0 for DECmate
- Application Software
  - VAXstation and Interleaf Publishing Software
  - GCLISP for Rainbow
  - SAMNA WORD III for Rainbow
  - CT*OS for Pro
  - TYPEasy for DECmate
- Technical Perspectives by Users
  - An Alphabetized Index for Your Disk on DECmate
  - DECmate and Rainbow Communications: Sending and Receiving Files
  - Using DECmate WPS and Rainbow poly-COM
  - Using the DELETE Key as the Backspace Key in MBASIC
  - Using UDKs on the DECmate
  - Printer Code Procedures for Your Pro 350
- Questions and Answers from Atlanta Customer Support Center

Perspective Vol. 4, No. 1, Jan 86 (EJ-27837-76)
- Executive Roundtables Help Entrepreneurs
- Evolution of a Technology: Computer Generations
- Announcing New Products
  - AI VAXstation
  - PC Connection Program
  - RD31 20 MB Winchester Disk for Rainbow

  - CD-Reader for Rainbow and IBM Personal Computers
  - DECmate/WPS V2.1
  - P/OS Hard Disk V3.0
- Application Software
  - SuperCalc3 for Rainbow
  - 20/20 Integrated Spreadsheet for Rainbow, Pro 300, MicroVAX, and VAX systems
  - Multi Journal Accounting (MJA) System on Pro 300
  - RDM 300 Data Management for Pro 300 hard disk systems
  - Horizon Spreadsheet for ULTRIX-32 on MicroVAX
- Technical Perspectives
  - Helpful Hints for MBASIC and SELECT Programs on Rainbow
  - Using WPS and WPSCONV to Create and Edit MBASIC Programs on DECmate
  - DECmate II Checkbook Balancing Program Using List Processing Capabilities
  - Rainbow ReGIS Patch that Allows Program to Send Terminal Identifier of VT125, VT240, or VT102
- Questions and Answers from Atlanta Customer Support Center

RAINBOW USER GROUPS

CALIFORNIA

BAKERSFIELD

Librarian:
Glenna S. Cochran
2900 Whitley Drive
Bakersfield, CA 93309
(805) 397-4676

INLAND EMPIRE PC LUG

Chairman:
Jan Snyder
General Dynamics
380 Veronica Street
Upland, CT 91786
(714) 620-7511, x 118/1101

Newsletter Editor:
Georgene Turok
1590 Lakewood Way
Upland, CA 91786
(714) 985-6200

SACRAMENTO VALLEY LUG RAINBOW SIG

Chairman:
Robert Walraven
1309 Notre Dame Drive
Davis, CA 95616

Newsletter Editor:
Walter L. Battaglia
Publications
c/o Yolo Expert Software
1111 Kennedy Place
Suite 4
Davis, CA 95616
(916) 758-8490

Rainbow Librarians:
Mike Bowers
(916) 752-6136
Robert Walraven
(916) 752-3300

RAINBOWS PACIFIC LOCAL USER'S GROUP

Richard Davis
home (213) 941-7059
office (714) 732-9308
(meets in Covina, CA)

Newsletter Editor:
Karen Danforth
PO Box 22786
Van Nuys, CA 91404
(818) 787-8736

By FIDO (213) 204-2996

SAN DIEGO AREA RAINBOW LOCAL USER'S GROUP

Chairman:
Rick Eliopoulos
Advanced Software Applications
5258 Vickie Drive
San Diego, CA92109
(619) 488-2116/5258

BY FIDO: (619) 488-2116

SAN FRANCISCO BAY AREA DEC PC USER GROUP

PO Box 12561
Northgate Station
San Rafael, CA 94913-2561

Chairman:
Dale W. Miller
(415) 472-6531

Librarian:
Kelley Cooley
(415) 981-3268

BY FIDO: (415) 981-3325

SANTA BARBARA AREA DEC PC LUG

Chairman:
Rick Vincent
253 Aspen Way
Santa Barbara, CA 93111
(805) 964-9744

SILICON VALLEY DEC PC USER'S GROUP

Chairman:
Seth Goldberg
PO Box 7309
Menlo Park, CA 94026
office (415) 854-3300, x 2874
home (415) 365-7223

Newsletter Editor:
Bob Garrett
c/o John Fluke Mfg.
2300 Walsh Avenue
Santa Clara, CA 95051
office (408) 727-0513
home (415) 792-3916
via modem (408) 720-0629

Librarians:
Gregory Howe
(408) 730-0886
Thomas Kallock
(408) 732-5129

SOUTHERN CALIFORNIA PC LUG

Chairman:
Bill C. Davis
BCOM
145 1/2 West 219th Street
Torrance, CA 90501
(213) 618-1083

COLORADO

DENVER RAINBOW USER'S GROUP

Brad Paulsen
Denver Aurora Community College
DACC
Box 250
Denver, CO 80204
(303) 556-2966

John C. Foster
Foster Sales
22074 Crestmoor Road
Golden, CO 80401
(303) 526-0088

PIKES PEAK LOCAL USER GROUP

William Sheaver
GTE Telecom, Inc.
102 South Teljon, Suite 300
Colorado Springs, CO 80903

CONNECTICUT

HARTFORD RAINBOW USER'S GROUP

PO Box 10387
West Hartford, CT 06110

Chairman:
Reginald Dionne
161 Silo Road
Bristol, CT 06511
(203) 282-0131
(203) 583-0003

NEW HAVEN RAINBOW LUG

Chairman:
William B. Leng
Southern Connecticut State Univ.
501 Crescent Street
New Haven, CT 06515
(203) 397-4625

Librarian:
Dan Aldrich
PO Box 5019
New Haven, Ct 06525
(203) 453-1392

NECRUG (Northeastern Connecticut Rainbow User's Group)

Chairman:
Howard Roberts
67 Route 6
Andover, CT 06232
office (203) 486-3920

Newsletter Editor:
Prof. Wilbur J. Widmer
Department of Civil Engineering
Box U-37
University of Connecticut
Storrs, CT 06268
(203) 486-4014

DISTRICT OF COLUMBIA, MARYLAND AND VIRGINIA

WASHINGTON AREA RAINBOW USER'S GROUP

Chairman:
N. Jay Bassin, PhD
9514 MIdwood Road
Silver Spring, MD 20910
(301) 280-4404

Newsletter Editor:
Caroline M. Mack
9007 Mears Street
Fairfax, VA 22031
(703) 280-4404

Librarian:
Paul Fontaine
8742 Brae Brook Drive
Lanham, MD 20706

BY FIDO: (703) 359-6549

WASHINGTON AREA PRO USER'S GROUP

Sharon Weiss
255 Congressional Lane
Rockville, MD 20852
(703) 692-1380

FLORIDA

CENTRAL FLORIDQA DEC USER'S GROUP (PC SIG)

Robert Thomson
Martin Marietta
Canaveral Operations
PO Box 1399
Cocoa Beach, FL 32931

HAWAII

HAWAII RAINBOW USER'S GROUP

(Inactive)
Professor Russell Yost
University of Hawaii
Tropical Agriculture Department
3190 Maile Way (St. John's 017)
Honolulu, HI 96822
(808) 948-7066

ILLINOIS

CHICAGOLAND AREA PC/WPS LUG

Newsletter Editor:
Jim Christine
1815 Cuyler
Berwyn, IL 60402
office (312) 329-3580
home (312) 788-4763

BY FIDO: (312)490-9206

KENTUCKY

KENTUCKY DEC PC LOCAL USER GROUP (being formed)

Kenneth Alan Presley
CICA, Inc.
4513 Flintlock
Louisville, KY 40216
(502) 448-2719

MASSACHUSETTS

Boston Computer Society DEC PERSONAL COMPUTER USER'S GROUP

Boston Computer Society
One Center Plaza
Boston, MA 02108
BCS (617) 367-8080

Chairman:
Karl Rosenberger
15 Willowbrook Drive
Framingham, MA 01701
(617) 697-1865

Newsletter Editor:
Annamarie Gilmore
44 Meadow Lane
Bridgewater, MA 02324
(617) 256-4638

Librarian:
Cliff Danielson
14 Davis Road
Chelmsford, MA 01824

MICHIGAN

DEC RAINBOW USER'S GROUP

Jeffrey Weihl
College of Human Ecology
Michigan State University
East Lansing, MI 48824
(517) 353-3717

MINNESOTA

MINNEAPOLIS-ST. PAUL PC LUG

(Being formed)
John Oilandre
Central Region LUG Coordinator
University of Minnesota
Dept. of Epidemiology
Stadium Gate 27
Minneapolis, MN 55455
(612) 376-4066

MISSOURI

ST. LOUIS PC LUG

Chairman:
Ken Kaplan
Data Research Associates
9270 Olive Boulevard
St. Louis, MO 63132-3276
(314) 432-1100

BY FIDO: (314) 576-2743

NEBRASKA

OMAHA RAINBOW USER'S GROUP

President:
Dr. John P. Maloney
4839 Pine Street
Omaha, NE 68106
home (402) 558-5134

Treasurer:
Shirley Bohaty
1343 Bel Aire Blvd.
Wahoo, NE 68066
(402) 433-4766

Newsletter Editor/Librarian:
Vaughn Call
1423 N. 105th St.
Omaha, NE 68114
home (402) 493-4051
office (402) 498-1492

NEW HAMPSHIRE

NEW HAMPSHIRE RAINBOW USER'S GROUP (being formed)

Paul A Gilberti
91 Mammoth Road
Windham, NH 03087
(603) 883-1512

BY FIDO: (603) 883-1596

NEW MEXICO

DEC PC LOCAL USER'S GROUP OF ALBUQUERQUE, NEW MEXICO

Chairman:
Stan Dutler
7204 Oralee, NE
Albuquerque, NM 87109
office (505) 846-3596
home (505) 822-8285

BY BBS (505) 831-0205

NEW YORK

LONG ISLAND LUG

Chairman:
Al (Fred) Scholldorf
Reuters Ltd.
90 Davids Road
Hauppauge, NY 11788
(516) 435-7116

NEW YORK METRO DEC PC LUG

Co-Chairman:
Bob Bennett
DEC User's Group of Greater New York
697 West End Avenue, #98
New York, NY 10025
home (212) 866-5154

GREATER ROCHESTER AREA LUG (PC SIG)

Gary Griswold
B&G Associates
PO Box 81
Webster, NY 14580
office (716) 477-7009
home (716) 872-3522

Newsletter Editor:
Frederick (Fritz) Howard
2988 Ridge Road, W.
Williamson, NY 14589
(315) 589-3981

BY FIDO: (315) 589-7361

NORTH CAROLINA

RESEARCH TRIANGLE LUG
PC Special Interest Group

Jack Brickley
PO Box 2713
Chapel Hill, NC 27515
office (919) 929-7791
home (919) 542-5494

OHIO

CENTRAL OHIO PC LUG

Chairman:
Dana Klebes
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201
(614) 424-4947

CENTRAL OHIO LOCAL USERS GROUP

Robert L. Lindsay
Lancaster City Schools
Administrative Offices
345 East Mulberry Street
Lancaster, OH 43130
(614) 687-7304

NORTHEAST OHIO PC USER GROUP

Khin M. Yin
2197 Oakdale Road
Cleveland Heights, OH 44118
(216) 932-5914

SOUTWEST OHIO RAINBOW USER GROUP

Gerald E. Maupin
1767 Wood Road
Lebanon, OH 45036
Voice (513) 425-3788

BY FIDO: (513) 425-3001

STARL COUNTY LOCAL SCHOOL'S RAINBOW USERS GROUP

Ronna Haer
Jackson High School
7600 Fulton Street, NW
Massillon, OH 44646
(216) 837-3501

PENNSLYVANIA

DELAWARE VALLEY DEC-PC USER GROUP

Chairman:
Roland Spressart
RSPE Engineers
89 Signal Hill Road
Holland, PA 18966
(215) 968-3494

Newsletter Editor:
Tom Deahl
MICRODOC
815 Carpenter Lane
Philadelphia, PA 19919
(215) 848-4545

Librarian:
W. C. (Scotty) Brooke
(215) 646-3300

EDINBORO UNIVERSITY RAINBOW USER'S GROUP

Robert Rhodes
Political Science Department
Edinboro University
Edinboro, PA 16444
(814) 732-4111

ST. JOSEPH'S UNIVERSITY RAINBOW USER'S GROUP

Chairman:
Dr. Val Herzfeld
5600 City Avenue
Philadelphia, PA 19131
(215) 899-7665

UNIVERSITY OF PENNSYLVANIA DEC RAINBOW USER'S GROUP

President:
George Theall
801 S. 47th Street, #304
Philadelphia, PA 19143
(215) 662-0558 (8-11pm)

Newsletter Editor:
Chad D. Graham
Pennrug
306 LRSM X3231 Walut Street
University of Pennsylvania
Philadelphia, PA 19104
office (215) 898-8509

Librarian:
Gerd Muehllehner
(215) 277-3073

SOUTH CAROLINA

SOUTH CAROLINA INDUSTRIAL LUG
(all, incl. PC)

Jim Skoler
Celanese Fibers Operations
PO Box 2000
Florence, SC 29501
(803) 393-7441

TENNESSEE

MIDDLE TENNESSEE DEC PC LUG

Newsletter Editor:
Donald Goss
Rte. 3, Box 165
Gallatin, TN 37066
(615) 452-7484

TEXAS

DALLAS RAINBOW USER'S GROUP

Chairman:
Ken Kattner
Kadex Corporation
11311 North Central Expwy
Suite 300
Dallas, TX 75243
(214) 991-8657

HOUSTON DEC PC LOCAL USER'S GROUP

Chauncey Hacker
Zebec Data Systems
2425 Fountainview
Suite 160
Houston, TX 77057
(713) 782-3480
(713) 933-6992

UTAH

SALT LAKE'S RAINBOW USER'S GROUP
(a SIG of the Salt Lake LUG)

J. R. Westmoreland
6748 Acoma Road
Midvale, UT 84047
(801) 262-5251

Linda Stringer (801) 292-7312
Bruce L. Gold (801) 487-2806

VIRGINIA

RICHMOND USERS GROUP

Chairman:
Bill Myers
Department of Chemistry
University of Richmond
Richmond, VA 23173
home (804) 320-1500
office (804) 285- 6321

WASHINGTON STATE

DEC BOEING USER'S GROUP (D.BUG)

Dennis Byrne
Boeing Company
3722 SW Admiral Hwy
Seattle, WA 98126
(206) 655-6191

SOUTH SOUND LOCAL USER'S GROUP

Chris Meyers
Tacoma News Tribune
1950 So. State St., PO Box 1100
Tacoma, WA 98411
(206) 597-8511

WEST VIRGINIA

WEST VIRGINIA LOCAL USER'S GROUP

Dr. Richard Hudson
Alderson-Broddhus College
Computer Center
Phillipi, WV 26416
(304) 457-1700, x 224

WISCONSIN

NORTHEAST WISCONSIN LUG (PC SIG)

Steve Peschke
Network System Design
300 North Main Street
3rd Floor
Oshkosh, WI 54901

GERMANY

GERMAN PC SIG

Dr. Otto Titze
Institut fur Kernphysik THD
TH Darmstadt
Schlossgartenstrasse 9
6100 Darmstadt
Germany
Telephone (0 6151) 16 33 23

EUROPE

EUROPEAN PC SIG

Paul Sawyer
School of Chemical Engineering
University of Bath
Bath, Avon, BA27AY England

AUSTRALIA

DECUS AUSTRALIA PC SIG

Chapter Office
PO Box 384
Chatswood, New South Wales, 2067
Australia
Phone 412-5252

ISRAEL

H. Heman
Digital Equipment, Ltd.
Digital House, Accadia Junction
PO Box 2033
47633 Herzliya, Israel

COMPUSERV

Compuserv has an active group of DEC PC users. Compuserv subscribers can type GO PCS-16 at the ! prompt and select sub-area 6. A real time conference is held on Tuesday evenings. Also has a bulletin board and software area.

ROG--ROBIN OWNER'S GROUP

Robin Owner's Group
PO Box 492
Rollinsford, NH 03869-0492

Chairman:
John Comella
2 Mockingbird Lane
Maynard, MA 01754
home (617) 897-5922
office 264-3430

Newsletter Editor:
Steve Dourson
5850 Waterloo
Washington Township, OH 45459

American Bar Association

DEC User Group
Lawerence Eisenberg
(818) 788-0354

Kelly Frey
Harwell Barr Martin and Stegall
PO Box 2960
Nashville, TN 37219-0960

David Sykes
Duane, Morris and Heckcher
One Franklin Plaza
Philadelphia, PA 19102
ABA CONFER #170

%%? LAW MUG (Lawyer's Minicomputer
User's Group)
Paul Bernstein
National Coordinator of ABA
Microcumputer User's Group
(312) 280-8180 (computer)

NABBSCO--NATIONAL ASSOCIATION OF BULLETIN BOARD SYSTEM OPERATORS

FidoNet BBS (312) 729-0012

NATIONAL DECUS ADDRESS

DECUS (Digital Equipment User's Society)
Anne Foley
249 Northboro Road (BP02)
Marlboro, MA 01752
(617) 480-3289

Hints, Tricks, and Curiosities
By Sanford Hersh

The following are some answers to questions that may be hard to find answers for or to questions you didn't think of asking. The information is usable for CP/M and MS-DOS as well as for the Select series of word processors. If anyone can use any of the information than I will have done my good deed for the day.

? How can I speed up the screen output under MS-DOS version 2.05?

! Change the contents of memory address of 40:19 to 13. (This will probably only appear to work if you are using "jump scroll".) Don't do this under version 2.11 as it will lock up the console. [ref. "Digital Review" Vol. 3 No. 2 November 1985 page 141]

? I've tried to log onto a diskette drive that has no disk and even after I answer ABORT (under MS-DOS) or CANCEL (under CP/M) the drive continues to make noise. What do I do?

! Log onto a disk drive that doesn't exist (such as C: or D:) then when you answer ABORT or CANCEL the noise in the other drive will stop. (This method doesn't seem to work under MS-DOS 2.11 as it doesn't let you abort! It will also be necessary to type a command to that disk such as DIR when under MS-DOS 2.05.) Now log onto an existing drive that has a diskette or to your hard disk.

? I've tried to do a hard disk restore and the restore utility just gives me this message: "Sorry, no files have been matched." Have I lost all of the files that took me hours to back up?

! Don't panic. You've probably backed up your disk using the "/A" option. To restore your files you have to use the "/A" option in the restore utility as well.

? How can I get a 7 bit device to work, such as a VT100 or a KSR terminal, connected to the printer port of the 8-bit Rainbow.

! There seems to be an error in the 7M parity setting on the Rainbow. Do not use this setting. Any other setting will probably work with the corresponding setting on the other device. The safest approach is to use the 8N setting for both devices. (Part of the problem may be that the Rainbow has two modes and the parity setting has no effect on them. The first mode is 7-bit and is only available through the VT-100 terminal emulation mode that is built into firmware. Otherwise the Rainbow is always in 8-bit or VT200 mode regardless of the parity settings.)

? How can I display only the names of files in the directory that have no extensions?

! Type: DIR *. (Include the period.)

? How do I compare a string to something in the environment (those definitions that are returned when you type SET <cr>) from a BAT file?

! Put the name between two %s (e.g. %COMSPEC%).

? How do I compare something to nul from within a BAT file (e.g. a nul argument)?

! Surround both sides of the comparison with a common symbol such as "/" (e.g. %4/=/).

? How I can I type a "DO" loop directly from the keyboard without creating a BAT file.

! Use the FOR command with only one leading % instead of the two leading %% as you would in a BAT file (e.g. FOR %f IN (FOO BAR BLEACH) DO DEL %f).

? How can I temporarily turn the MS-DOS prompt off?

! Type ECHO OFF. This can be typed from the keyboard.

? What are the two ways of listing a CP/M file so that it prints with line numbers?

! Method I: PIP PRN:=filename Method II: PIP LST:=filename[N]

? For text editing I use Red/Select/Freestyle. Is there some way to move to the nth line or the mth page without having to continuously push the cursor and screen keys?

! All you have to do is preceed the cursor and screen keys with a number and you will jump to that position. (Don't do this in "insert mode" as the number will become part of the text.) e.g. If you want to go sixteen screens back, type: 16<Prev Screen>

? Select had an Alt mode where you could create an "ASCII" document, is there a way to do that with Freestyle?

! Create the file in the using Mail option of Freestyle. Unfortunately this mode recognizes the back-slash as an "escape" character.

? While Freestyle I can't seem to tell it to have the left margin in column one, how can I do it?

! Use the Mail option of Freestyle.

? Is there some way to capture a "printer image" file from Freestyle so that additional copies of the text can be printed without having to go into Freestyle? i.e. Can a file be created that contains all of the special escape and control sequences that enable a printer to do its tricks?

! There is a menu option that enables you to do background printing. When the option is exercised it creates a file that ends with the extension "SPL". This is the file that you want. Rename it and it is saved.

? How can I go to another branch of the same level of the directory tree in MS-DOS?

! Type: cd ..\branch_name (Note: There is a space between cd" and "..". There is no space between ".." and "\".)

? How can I temporarily 'push' in a different version of COMMAND?

! Follow the word COMMAND with a path. (e.g. COMMAND B:\GARP\) The COMMAND.COM file in that path will be used until it is 'pop'ed by typing EXIT.

? How can I call an internal MS-DOS command directly from assembler.

! Use the /C option after executing COMMAND and than follow it with the command. e.g. COMMAND /C DIR (You may also have to release some space for it to execute.)

? How can I remove garbage from the end of a text file? (This is especially a problem with files that have been moved from CP/M to MS-DOS as CP/M retains files in blocks of 128 bytes whereas MS-DOS appears to keep the actual byte count.)

! Type COPY FILE+. (Where FILE is the filename. The plus sign is required and there is no space before it.) This will copy a file over itself. It essentially replaces the character count in register CX with the the character position that preceeds ^Z. Do this only with text files. If you do this with command files you will destroy them!

? How about combining two files onto one of the first two. I understand that under MS-DOS you can COPY FILE1+FILE2 FILE1 or COPY FILE1+FILE2 FILE2. Should I expect any problems when I do this?

! Ugh! Here's what happens. If you COPY FILE1+FILE2 FILE1 then there will be no problems unless there is insufficient disk space. If there isn't enough disk space than file1 will disappear! If COPY FILE1+FILE2 FILE2 then file1 will replace file2. This is identical to COPY FILE1 FILE2.

? How can I disable ^C under MS-DOS for the duration of printing?

! Type ^X. This is probably a bug. If you accidently type ^X or use the Cancel key there is no way of interrupting a long document without rebooting the system.

? How can I enter the escape character under MS-DOS?

! Use the the <Interrupt> key (F6) instead of the <ESC> key F11.

? How can I set the pointer to after the last line when editing a file when using EDLIN?

! Type #<cr>.

++

The following are questions that I have no answers for. If anyone knows any of the answers I would appreciate it if they would let me know.

? How can I enter ^N from the keyboard?

? How can I enter characters from the C1 character set from the keyboard?

? How can I ask questions from a BAT file and then use the answers?

? After typing PRINT and choosing the output device, how can I revise my choice?

? How can I change/add something to the environment from assembeler?

? How can I display a string that contains a "$" if using the MSDOS string function?

? How do you ECHO the words ON and OFF?

? While using Freestyle, how can I force an exact number of spaces between words when in right justification mode?

? Is there a version of Freestyle that is bug free? Is there anyone in Select who doesn't think they write perfect code?

Reprinted with permission of GRANOTES Newsletter

The Rainbow Workbench

by Alan I. Vymetalik, AVSD, Ltd.

RND() NOTES AND RUMORS

Welcome back from a long break, Rainbow folks! Over the last month, there has been some interest in favor of this column. So, I guess I'll write a few more.

There has been a recent flurry of excitement for the Rainbow (see the Hardware News below) and I hope that it continues. There are about 400,000 Rainbows out in the world, somewhere, and those owners are finally standing up and demanding to be noticed (about time!).

Well, it's almost official. "By the time the leaves turn green", DEC says it will announce the replacement (sob!) for the Rainbow (code-named PC-XX). It appears that new machine will be a PC AT-like machine that will run at 8Mhz versus IBM's 6Mhz.
Also, DEC will be releasing MS-DOS Version 3.1. This version should also run on a regular Rainbow and give some additional power. Another rumor is that DEC and Lotus are still discussing the release of the new version of 1-2-3. It seems Lotus feels no one out there cares about the Rainbow anymore. Come on Lotus users!

Reprinted with permission of GRANOTES Newsletter

The Tech Corner

By Alan Vymetalik

This month, I would like to try to clear up some of the mystery of reading the Rainbow keyboard from your own programs. You can easily add the routines shown in Listings 1 and 2 to your programs under MS-DOS or CP/M.

The function keys on the Rainbow keyboard generate a string of characters called an 'escape sequence'. Each time a function key is pressed, a series of up to 5 characters is sent from the keyboard. However, the console drivers in standard MS-DOS or CP/M recognize only one byte of data at a time.

Pressing the REMOVE function key generates four bytes of data: the Escape Character (ASCII 27) and the following characters:
[3~. These four characters would each have to be scanned one at a time using normal means. This turns out to be cumbersome at best.

Listing 1 shows an MBasic-86 program for reading the keyboard at the time of an INPUT$ statement. It traps the first character and then sets up a dummy line, Y$, of INKEY$ functions to trap any following characters. If the incoming string of characters matches the check, a branch is made.

This routine is effective at trapping any key. But, to check for other keys you will have to write tests for each full 'escape sequence'. Also, it has no way of knowing if you pressed any combination of SHIFT, CONTROL and/or CAPS LOCK. All your program knows is what key was hit.

In Listing 2, you will find a Turbo Pascal procedure that takes advantages of the firmware (that part of the Rainbow's operating system stored in Read-Only- Memory on the motherboard). Using these particular interrupts, the routine will tell you which key was pressed and the current state of the keyboard. This allows us to get at the operating system and firmware without having to write any assembly code.

Listing 3 shows the DEC-codes for every function key. Missing from the list are the codes for the "Hold Screen" and "Set-Up" keys. You cannot access these keys from your program because the Rainbow's keyboard filters the key codes out before the operating system gets a chance to see them.

The idea behind "ReadKey" is rather simple. We can user the Turbo Pascal built-in procedure call:

```
INTR(interrupt #,record of registers)
```

(where the interrupt # is a particular value supplied by DEC as the entry point to their firmware routines and the register record is a Pascal data structure) to access the ROM firmware. The interrupt call accepts data from the program by having certain values set before the call and it returns data in other portions of the record after the call.

The routine in Listing 2 starts off assuming that no key has been pressed. It will remain in a tight scanning loop of the keyboard until there is an key pressed. If a normal character (like 'A' or '$') is pressed it is put in the Level-1 buffer and "ReadKey" will return immediately. If it was a function key, "ReadKey" clears out the buffer of 'escape sequence' characters (found in the Level-2 Buffer) before exiting. This leaves the keyboard buffer in a cleared state for the next time you call "ReadKey." DEC supplies both of these buffers so that a program can access the keyboard directly or like the example in Listing 1.

Finally, we retrieve the keyboard status and character from the returned variable: Regpack.AX. Data for these two bytes of information are stored in the upper and lower halves, respectively, of this variable. An 8088 register is 16 bits wide and thus can hold two 8-bit bytes of information.

When finished, the routine "ReadKey" returns all necessary data in two bytes. Call this routine in place of Pascal's Read statement and check one byte for the keyboard status and one character for the key pressed whenever you want to test for function keys and/or characters.

The Pascal routine may appear slightly confusing, at first. Study the comments and, if you have Turbo, test it out. The routine can be easily implemented in any other language that provides a means of calling the firmware.

For additional programming examples, firmware functions, and other Rainbow hardware features, download the RBFUNCT.ARC, RBSYSTEM.ARC, HIASCII.ARC source archives from "The Hitchhiker's Guide".

==

Reprinted with permission of GRANOTES Newsletter

==

Hardware News

By Alan Vymetalik

The recent buzz word has been "Rainbow-IBM hardware compatibility". Disk Tech One has announced the RB-Link series of IBM plug-compatible Rainbow extension boxes. My opinion of the demo made by the President of DTO was something close to ecstasy! Seeing the new version of Lotus running using the Rainbow's keyboard and monitor was electric!

A side note on the new RB-Link series: If Data Tech One can produce their RB-Link at a reasonable price, owning one would be one of the best moves on the part of the Rainbow owner. Not only would he or she have a super microsystem but, with the flick of a software switch, have instant and two-way, fully-compatible access to the IBM PC world for less than owning two separate machines.

These additions to the Rainbow line, DEC's release of a 20M hard disk as the new Rainbow standard, and DMI's streaming-tape backup unit have shown renewed interest for the Rainbow. It seems the Rainbow is finally achieving some of the support so long denied it.

==

Reprinted with permission of GRANOTES Newsletter

==

Public Domain Software Summaries

By Alan Vymetalik

For lack of space and comments from readers, I am holding off presenting a Software Ratings Guide. I will hopefully start it sometime soon.

This month we will look at several utilities for the Rainbow. The first two are by Christopher LaRosa (the creater of DRAW V1.4 for the DEC Rainbow). These two tools stand out for their clear documentation, ease of use, and compactness of code; a must for the serious Rainbow or PC user's toolbox.

ATTR V1.0 is a file attribute modifier. You can set or clear any attribute of a file (read-only, hidden, system, archived). This tool is essential to make sure you protect files from being accidentally erased.

STDIR V1.2 is a directory sorting program that produces a listing of the current directory and/or all directories on the disk. Use it on the root directory of your hard disk and print out every file. Options include: sort on file extension, ignore the files and disk space and just give totals, display hidden and system files, or doing only the current directory.

DVORAK V1.01 by Jim Beveridge is a Rainbow-specific system-level remapping of the QWERTY keyboard (originally designed to slow typists down so they wouldn't jam the keys) to the Dvorak method. Tests have shown incredible improvement in typing speed when using the Dvorak method. While this utility is defintitely worthwhile, two nasty bugs are found: auto-repeating causes the original QWERTY key to reappear and there is no way, short of rebooting, to clear the program.

CHKMEM V1.0 is Jay Jervey's utility for checking on how much memory your system is gobbling up. Jay Jervey, author of KEDT, has created a very useful program to check up on memory-resident programs, ram-disk sizes, and free memory without having to use DOS's ChkDsk program. This utility is a must!

RESETRB V1.1 from Alan Minchew and Rob Elliot is a handy program to reset all Rainbow screen attributes to their defaults. How many times has your screen zapped into VT100-graphics mode when you're dialing other computers or running bad software? Or, how many times has ill-software turned off your cursor; never to reappear? Using this utility will keep you from having to reboot.

HISTORY V2.4 is Bryan Higgins' latest attempt at producing a memory-resident program that allows you to recall, edit, and re-issue previously typed DOS commands. Specifically written for the Rainbow, the user can recall commands (up to about 512 bytes worth) by using the up and down arrow keys and other function keys to edit the lines. Many different options exist for searching for the most recent occurence and certain string patterns allowing an efficient manner in which to locate old commands.

HISTORY does a wonderful job but we find the DO key (which MS-DOS uses to retrieve the last command and, thus, we're very used to it) remapped to allow for a single-character search mode of the old commands. Typing DO and a prompt appears asking for a character. Typing any letter lets the function work. Typing a function key HANGS the system. So, not only has a very familiar function key been needlessly changed, but its replacement can be fatal! I am looking forward to the next release. Until then, beware of small bugs lurking in HISTORY.

That wraps it up for this month. I still invite anyone to submit questions and ideas to either the editor or to me, personally, at AVSD, Ltd., 646-1 University Avenue, Rochester, New York, 14607-1232, (716)-461-2446. I will repeat the questions and comments I receive and those requesting so will remain anonymous.

==

```
10 X$ = INPUT$(1)                               ' trap first character no
                                                ' matter what it is
20 Y$ = INKEY$ + INKEY$ + INKEY$ + INKEY$       ' possibility of 5 chars.
                                                ' in code sequence
30 IF Y$ = "[3~" THEN GOTO 60                   ' was it the REMOVE key?
40 PRINT "You hit some other key"               ' ...no go?
50 GOTO 10
60 PRINT "You pressed the REMOVE key"           ' success!
70 END
```

Listing 1 - Rainbow MBasic-86 Program for Trapping the REMOVE Key.

==

```
{  Procedure "ReadKey"  Written by Alan I. Vymetalik, AVSD, Ltd.

      This  routine  maintains  a tight loop scanning the Rainbow keyboard
until  a  key  is  pressed.  The  routine  returns  the character and any
information  on whether it was a FUNCTION, CONTROL, SHIFT, or CAPS LOCKed
key in a single byte.

      (Note: a value marked FFh or $FF means FF in hex or 255)

   Value for FLAG in AX Register (AH half):

        15    . . .    11  10   9   8     <- bit position
      +---+---+---+---+---+---+---+---+--
      |   NOT USED    | X | X | X | X |   character in lower 8 bits
      +---+---+---+---+---+---+---+---+--

      Bit 8  = 1 if Function  key was pressed
      Bit 9  = 1 if Shift        "    "     "
      Bit 10 = 1 if Control      "    "     "
      Bit 11 = 1 if Caps Lock    "    "     "

    For example: if just a function key was pressed, FLAG would be 1.  If
it  was  a  Control-Function  key,  FLAG  would be 0101 in binary or 5 in
decimal.  And so on ... }

Procedure ReadKey( Var ch : char; Var flag : byte );

Const
  FirmInt = 24;                     { 24 for MS-DOS, 40 for CP/M }

Var

{ Required interrupt variable.  The firmware function, in this rou-
  tine,  is loaded into the DI register and data is returned in the
  AX register and interrupt status in the CX register. Data must be
  re-loaded before calling the INTR() procedure   because registers
  are modified by the firmware routine.  Unreferenced registers are
  not used.}
  regpack = Record
               AX,BX,CX,DX,BP,SI,DI,DS,ES,Flags : Integer
            End;

Begin { Procedure ReadKey }

  regpack.CX := 0;                       { set to no key pressed }

  While regpack.CX = 0 Do
    Begin                        { stay looping 'til a key is hit }

      regpack.DI := 6;             { Firmware call - Level-1 Read }
      Intr(FirmInt,regpack);                      { call the ROM }

         { The CX register, upon return, contains 1 of 3 values:
                00h - No key was pressed,
                01h - A  Level-2  sequence exists (an 'escape
                      sequence'), buffer is not empty, or,
                FFh - Character value is in the AX register. }

      If regpack.CX = 1 Then
        Begin

         { CX Register - Settings:
                00h - Level 2 Buffer is clear
                FFh - Level 2 Buffer has data in it }

          regpack.CX := $00FF;          { Clean up Level-2 buffer }
          While regpack.CX = $00FF Do                { until empty }
            begin
              regpack.DI := 2;          { Firmware - Level-2 read }
              Intr(FirmInt,regpack)                 { call the ROM }
            End { Clear Level-2 }

        End { test for level-2 characters }

  End; { waiting for a key }

{ All but the keys in Listing 3 will return their normal codes.  If
  the FLAG shows it to be a function key,  reconvert  the  character
  value with the ORD function to get the number (in Listing 3). The
  functions HI and LO get the  upper and lower  8-bit halves  of an
  integer. }

  ch     := Chr(Lo(regpack.AX));
  flags  := Hi(regpack.AX)

End; { Procedure ReadKey }
```

Listing 2 - Turbo Pascal Keyboard Scanning Routine

==

0	HELP	27	FIND	62	Keypad 5
1	DO	29	INSERT HERE	65	Keypad 6
2	COMPOSE	31	REMOVE	68	Keypad 7
3	PRINT SCREEN	33	SELECT	71	Keypad 8
5	F4	35	PREV SCREEN	74	Keypad 9
7	INTERRUPT	37	NEXT SCREEN	77	Keypad -
9	RESUME	39	UP ARROW	80	Keypad ,
11	CANCEL	41	DOWN ARROW	83	Keypad .
13	MAIN SCREEN	43	RIGHT ARROW	86	Keypad ENTER
15	EXIT	45	LEFT ARROW	89	Keypad PF1
17	ADDTNL OPTIONS	47	Keypad 0	92	Keypad PF2
19	F17	50	Keypad 1	95	Keypad PF3
21	F18	53	Keypad 2	98	Keypad PF4
23	F19	56	Keypad 3		
25	F20	59	Keypad 4	101	BREAK

F11 ESC key, F12 BS key, and F13 LF key return
the normal ASCII codes. Values are in decimal.

Listing 3 - Special Rainbow Function Key Codes Returned by Firmware

This came to me thru a long chain of forwards, so I'm not sure of the original source, but I thought it might clear up some of the new product rumors.

==

Reprinted with permission of GRANOTES Newsletter

==

DEC Rainbow Winchester Disks Technical Summary

By Brian Orr
Jan 30, 1986

This Document contains some technical Winchester specs that have been gleaned from the Seagate ST225 OEM manual, Seagate ST506 Service manual and other technical papers I could lay my hands (eyes?) on. Hope it helps the inquisitive and does not scare off the meek!

Mounting Requirements for Winchester Series in the Rainbows.

The ST506, ST412, & ST225 hard disks may be mounted on their sides either left or right or horizontal with the spindle motor down. The drive should NOT be tilted from front to back, in any position, by more than +/-5 degrees.

Here comes the important part that nobody bothers to tell you but makes a lot of sense...

****
Data should be read with the drive mounted in the same orientation as it was during the write operation!!!!
****

In other words, format your disk in the same position it will be used in. If you decide to change the orientation at a later time you might consider reformatting in the new position. (Backup up your disk first though.)

Winchester Specifications

	5MB ST506 Full Height	10MB ST412 Full Height	20MB ST225 Half Height
UNFORMATTED	6.38MB	12.76MB	26.62MB
Per Surface	3.19MB	3.19MB	6.41MB
Per Track	10,416 Bytes	10,416 Bytes	10,416 Bytes
Per Sector	NA	NA	NA
Formatted	5.0MB	10.0MB	20.15MB
Per Surface	2.5MB	2.5MB	5.04MB
Per Track	8,192 Bytes	8,192 Bytes	8,192 Bytes
Per Sector	256 Bytes	256 Bytes	256 Bytes
Sectors/Track	32	32	32

You can format an ST225 to have 512 Byte sectors in which case the following Specs apply.

Formatted			21.42MB
Per surface			5.35MB
Per Track			8,704 Bytes
Per Sector			512 Bytes
Sectors/Track			17

Access Time			
Average	85ms	85ms	65ms
Maximum	205ms	205ms	150ms
Track to Track	<16.67ms	<16.67ms	<20.0ms

Access time is defined as as the time from the leading edge of the last Step Pulse received to SEEK COMPLETE (including carriage settling). The period between Step Pulses must be between 5 microseconds and 200 microseconds.

Average access time is measured over a 205 track seek. (one-third stroke) calulations assume the following:

1. Nominal temperature and power
2. The average is taken from an inward on-third stroke, plus an outward one-third stroke.

Rotational Speed	3600RPM	3600RPM	3600RPM
Recording Density	9,074 BPI	9,074 BPI	9,827 BPI
Flux Density	9,074 FCI	9,074 FCI	9,827 FCI
Track Density	345 TPI	345 TPI	588 TPI
Cylinders	306	306	615
Tracks	612	1,224	2,460
Read/Write heads	2	4	4
Discs	1	2	2
Data Transfer Rate	5.0Mbits/sec	5.0Mbits/sec	5.0Mbits/sec
Recording Scheme	MFM	MFM	MFM

DC Power Requirements

12 Volts DC			
Voltage Tolerance	+/-5%	+/-5%	+/-5%
Power-on Max Current	3.5A	3.5A	2.2A
Typical Current	1.6A	1.6A	0.9A
5 Volts DC			
Voltage Tolerance	+/-5%	+/-5%	+/-5%
Power-on Max Current	1.7A	1.7A	1.2A
Typical Current	1.1A	1.1A	0.8A
Total Power-on Power	50.5Watts	50.5Watts	32.4Watts
Total Typical Power	24.7Watts	24.7Watts	14.8Watts
Ambient Temp			
Operating	40-122deg F	40-122deg F	50-113deg F
Non-operating	-40-140def F	-40-140deg F	-40-140deg F
Reliability			
MTBF	11,000hrs	11,000hrs	20,000hrs
PM	none	none	none
MTTR	30 min	30 min	30 min
Service Life	5 yrs	5 yrs	5 yrs

Take a good look at the MTBF hours of the full height verses the half height. If you have a choice I would get an ST213 instead of the ST412 or just go 20MB and get an ST225.

Read Error Rates 1 bit in

Recoverable	10e10bits	10e10bits	10e10bits
Non-recoverable	10e12bits	10e12bits	10e12bits
Seek Errors	10e6 seeks	10e6 seeks	10e6seeks

note: "e" above stands for Exponent 10e6 is 1,000,000.

A media defect is a read error when data, which has been correctly written, cannot be recovered within 16 retries.

There will be no more than eight defects per surface for a maximum total of thirty-two per drive. Cylinder Zero will be free of defects.

The October 29, 1985 edition of the Seagate ST225 manual states that the new half height ST213 may be used as a replacement to the full height ST412. Voltage requirements and connections are identical to the ST225.

ST213 Specs

Tracks 1230
Cylinders 615
Read/Write heads 2
Discs 1

Shipping Zone

The ST225 employs a shipping zone located from cylinders 615 to 670 to preserve data integrity during shipping/transport. The Read/write heads may be parked in the shipping zone by issuing a seek to any cylinder between 615-670. The drive may then be powered down.

Upon power-up the drive will recalibrate to Track 0. If the heads are parked while power is still applied, any STEP pulse will cause the unit to recalibrate to Track 0.

ST225 Drive Configuration Jumpers

Bottom of drive is facing up.
(Printed Circuit Board Facing UP)

```
  SHUNT BLOCK Configuration
 ---------------------------
 |  1  3  5  7  9  11 13 15  |
 |  o  o  o  o  o  o  o  o   |
 |                           |
 |  2  4  6  8  10 12 14 16  |
 |  o  o  o  o  o  o  o  o   |   Shorting blocks are used to short
 |                           |   the jumpers. To select an option.
 ---------------------------
```

1 -> 2 Selects Radial Operation
Drive is always selected

3 -> 4 Selects Recovery Mode
Enables Read write heads to microstep

5 -> 6 Selects Write Fault
Internally latchs write fault (not a normally selected option)

7 -> 8 Selects Life Test
Used for testing only! Stepper motor will continually seek between Track 0 and Max track.

9 -> 10 Enables DRIVE SELECT 4
For Daisy Chaining Disks

11 -> 12 Enables DRIVE SELECT 3
For Daisy Chaining Disks

13 -> 14 Enables DRIVE SELECT 2
For Daisy Chaining Drives

15 -> 16 Enables DRIVE SELECT 1 ***** THIS IS THE ONLY BLOCK
For Daisy Chaining Drives ***** SHORTED ON THE
******** RAINBOW ST225 ***********

BBSes

By Rob Elliott
Fido #115/100
CIS [70675,1204]

This is a list of all the BBSes running in, or having a reasonably sized section for, DEC Rainbow 100 computers. BBSes with unlisted numbers, or BBSes that are otherwise inaccessible, are not listed here. Send any and all update, corrections, etc. to me via CompuServe or through Fido 115/100.

Coding: All boards are assumed to be 300/1200 baud. A "+" before the phone number means the board has 2400 baud support also. All boards are assumed to be Fidos. If one isn't, the type of BBS program is listed instead of a node number. Node numbers of -1 cannot be reached via FidoNet at the moment.

FIDO Log-on: Upon connection, wait 5 full seconds then 300/1200 baud callers hit RETURN twice, 2400 hit SPACE twice, to start the BBS.

Node Phone Number	Name	Location	Sysop
333 602-952-2146	Rainbow BBS	Pheonix AZ	Jim Koshner
36 +213-204-2996	Rainbow Data	Culver City CA	Don Brauns
666 +714-537-7355	Revelation 13	Garden Grove CA	Mike Hamilton
3603 213-644-1963	Rainbow Brite	Hawthorne CA	Bruce Headly

410	Medic	714-964-0454	Huntington Beach CA	Phil St. Erne
350	SD Rainbow LUG	619-488-2116	San Diego CA	Rick Eliopoulos
1	Fido's Board	+415-864-1418	San Francisco CA	Tom Jennings
13	Vern's Fido	408-923-5565	San Jose CA	Vern Crawford
610	Sam's Rainbow	408-720-0629	Sunnyvale CA	Sam Rogers
383	Southbay_RB	+213-618-8454	Torrance CA	Glenn Bowes
430	Oberon_Systems	+805-643-0982	Ventura CA	Scott Johnson
210	Rainbow's End	203-453-1089	Guilford CT	Matt Gertner
51	Stockware	302-655-6342	Wilminton DE	Van D. Olmstead
46	Joe's Rainbow	305-321-2369	St. Mary FL	Joe Clayton
199	Illini Data	+312-759-5402	Bolingbrook IL	Rob Elliott
2	CLP-Fido	301-484-2831	Pikesville MD	John Madill
-1	Beauty Board	+301-776-2300	Laurel MD	John Raum
14	WayStar	617-481-7147	Marlboro MA	Kevin Porter
27	Daves Fido	+617-632-1861	Fardner MA	David Rene
44	NECS	617-646-3610	Arlington MA	Dave Mitton
45	Midnite DEC	+617-787-3033	Boston MA	David Strickler
310	Dave's Annes	617-874-4325	Westminster MA	David Rene
202	DEC-Line	617-721-1688	Winchester MA	Bill MacNeil
TOPS20 DEC MARKET		617-467-7437	Marlboro MA	Bernard Eiben

367	RBG-ENG HootNet	617-486-2286	Littleton MA	Bruce Gibson
-1	Thieve's World	616-343-0996	Kalamazoo MI	Ian Schirado
16	Mikes Board	+314-726-3448	St. Louis MO	Mike Mellinger
17	DCA BBS	314-962-0395	St. Louis MO	Jon Wichman
22	PCLUG	+314-576-2743	St. Louis MO	Ken Kaplan
51	Decus Central	+314-432-4129	St. Louis MO	Ken Kaplan
82	DEC-House	609-429-6630	Cherry Hill NJ	Brian Sietz
317	DEC-WARE II	201-750-3748	Iselin NJ	David Horowitz
ROS	Albuq. PC-LUG	505-831-0205	Albuquerque NM	Eloy Gonzales
79	DEC-WARE	+212-535-8924	New York NY	Matt Kanter
-1	Scherer's	502-425-9941	Dublin OH	David Orr
489	Catt House Fido	717-794-5268	Blue Ridge SummitPA	Bob Catt
348	Day Rainbow	01146-54-266988	Karlstad SWEDEN	Conny Johnson
-1	Big D Fido	+214-392-1121	Dallas TX	Dennis Forcier
67	DEC Domain	806-795-0102	Lubbock TX	Alan Minchew
918	DEC Bronson	806-742-5328	Lubbock TX	Bronson Johnson
483	Wash-a-RUG	703-359-6179	Fairfax VA	Kurt Reisler
74	The Bear's Den	+703-671-0598	Falls Church VA	Kurt Reisler
90	MidNet	608-233-8449	Madison WI	Mike Mansfield

Word Perfect and the Rainbow

by Bob Garrett

A few months ago, a salesman from SSI Software demonstrated Word Perfect 4.0 at at our group meeting. The features he presented were enough to convince me that this package was far superior to the word processors I've had the misfortune to use in the past. He also claimed that the next release (version 4.1) would incorporate major improvements (thesaurus, graphics, etc.) making it even more powerful. In November of last year, I purchased Word Perfect 4.0 and installed it on my Rainbow 100B. The salesman who sold me the package promised that SSI would mail version 4.1 as soon as it was released, provided I let them know I was interested.

For the past couple months I've been using 4.0 and have been very pleased. I am certainly no expert, however I've used a number of it's features and have been quite happy with the results. Two weeks ago, I received my copy of Word Perfect 4.1 and as they said, it is much more powerful. Some of the features that impress me the most are:

Automatic Backup - You can specify how often you would like to have your documents backed up. While you work the program will periodically do the backup, displaying a "please wait" prompt while it does so. I usually look at the keyboard while I type and miss the warning message. However, all the text I've typed while the backup was being performed is inserted once its done. This backup is for catastrophic failures like power outages, etc. You still need to save your documents when you're done.

All Functions Available While Editing - Some of the word processors I've used forced you to leave the edit session before you can print the document, check spelling or get a directory. This takes time and can be particularly exasperating when you discover a mistake in the text. Word Perfect allows you to use ALL its functions while you are still editing. You can send a copy to the printer, then continue editing while it's being printed.

Speller - The spell program does all the usual things you would expect, by checking words, pages, document, etc., but it also checks for double words. Once it finds a misspelled word, a list of possible alternatives (based on the phonetic spelling) are displayed. One of these words can then be easily substituted for the incorrect one. A supplemental dictionary can be created by choosing the 'Add Word' feature when the word can't be found in the main dictionary.

Thesaurus - At first, I thought this was a nice 'gimick', but it works so fast and so well that I would not want to write anything without it. Both the dictionary and the thesaurus can be placed in main memory to get maximum speed when looking up a word. This means you might need som extra memory, but it's well worth it.

Switch Documents - Word Perfect allows you to switch between two documents while you're editing. I didn't think I'd use this feature very often either, but I've found many cases where I've needed it. With 4.1 you can display portions of both while you wor, with the 'Window' function. Not only does this allow you to view them at the same time, but you can copy portions of text between the two quite easily.

File Utilities - The 'List Files' function gives you a directory similar to 'Maint' (CPM) and allows you to retrieve, delete, rename, print, import ASCII text, look at, change directory, copy or search the selected file(s) for word patterns.

Document Formatting - You can choose a number of document formats. There are a couple different types of outlines, and columns can be either newspaper style (for text that flows up and down through the columns) or parallel (e.g., side-by-side translations, scripts).

As you might be able to tell by now, I'm very impressed with this software. Rather than go on and on about the various features, I'll be glad to discuss it at the user group meeting.

Reprinted with permission of Silicon Valley PC User's Group

Product Review: DM110 Hard Disk Tape Backup System

By Karin Froide
San Diego Union-Tribune Pub. Co.

Rainbow users have long suffered the indignity of being relegated to the dark ages when it comes to innovations available in the peripherals market. In comparison to devices marketed for those "other" computers, third-party peripherals available for the Rainbow are sparse to say the least. Considering the reliability and technological soundness of the DEC PCs, it's a shame that once we have purchased a DEC machine, we are forced to beg, badger, and cajole vendors to provide us with state-of-the-art peripherals.

One of the most annoying rituals associated with personal computers is the process of backing up hard disk files to floppy disks. Because this is such a tedious, time-consuming and inconvenient task, more than one user has probably procrastinated one day too long and paid a dear price for that procrastination.

But someone has finally stepped in to fill a much-needed void for Rainbow 100+ users.

Duncan Mac Donald Inc., a small Southern California firm, has developed a hard disk tape backup system called the DM110 Tape Backup Unit. Recently I was given the opportunity to test one of these units and with few minor exceptions found it to be a device worth considering as an adddition to your system.

The DM110 package includes the unit itself, the software, and a manual outlining installation and operating procedures.

Visually the DM110 unit has the appearance of a stand-alone floppy disk drive. Its relatively small size, 3-1/4 by 5-3/4 by 12 inches, will not add much additional clutter to your workspace.

Internally, the unit houses a Cipher Data Products Model 525 Floppy Tape cartridge tape subsystem, as well as its own 40-watt power supply and an adapter board. The DM110 utilizes the extra port on the Rainbow's floppy disk controller and the adapter board provides the interface through which data is transferred between the Winchester and the tape subsystem. The power supply is simply extra insurance that the unit will not put additional strain on the Rainbow's own internal power supply.

The tape cartridge itself will hold approximately 25 MB of data on 6 tracks of approximately 4.2 MB apiece.

The software is straightforward, offering the operator 3 choices including formatting a blank tape, backing up to the tape, and restoring the data to the hard disk. During my test, the formatting procedure took about 25 minutes. Backing up the data from my disk (about 8 MB) took about 12-15 minutes, and restoring that data took about the same length of time.

One of my initial concerns with the unit came when it was time to restore the backed up data to my hard disk. Despite the fact that I had backed up all my files to floppies, I was hesitant to attempt the restoration process on my own hard disk since this was a new device about which I knew nothing. As a result, I attempted to restore the files to another computer's empty hard disk. As it turned out, this didn't work either, mainly because the disk to which I was attempting to restore my files was a 20 MB disk and I had backed up on a 10 MB. I called Duncan Mac Donald to register my concerns and he assured me that the reason for failure was due to the incompatibility in sizes of the two hard disks.

Since I couldn't find another 10 MB disk to dump the tape onto, I bit the bullet and finally gathered the courage to restore the tape to my own hard disk. Prior to restoring my files, I wrote down the byte count so that I might verify that figure after the test. After the test, I found that the byte count had not been altered at all. Both the software and the hardware worked beautifully during the test. I didn't find a single garbled file, and it certainly was much easier to handle than a fistful of floppies.

As is the case with any tape backup system, while the computer is engaged in the backup process, the system is unavailable for use. While inconvenient, this is not an insurmountable problem. Provided you choose a time to perform the backup operation when you wouldn't be using the computer much anyway (start it up and go to lunch, for example), this is an excellent system.

If there was one thing that I didn't find satisfactory about the DM110, it was the fact that the software did not include Selective File Restoration. Without this function a user could conceivably spend all his time backing up since each time you changed data in even one file, your original backup tape becomes invalid.

However, prior to writing this review, I spoke to Duncan Mac Donald about this problem. He said his company had already addressed this issue and that they will have Selective Restoration by Partition by the middle of April and Selective File Restoration will be available around the beginning of May. He has also addressed my initial concerns with attempting to restore data to my hard disk by including a verification process during the Restore operation.

Overall, my conclusion was that this was a very useful and much-needed peripheral for the Rainbow 100+, and that it will be even better once the Selective Restoration process is available. Who knows? Rainbow users may yet be able to shed that medieval image.

Additional information regarding the DM110 unit may be obtained from Duncan Mac Donald, Duncan Mac Donald Inc., 29276 Ridgeview, Laguna Niguel, CA 92677. The phone number is 714-495-4418.

Interfacing DEC Rainbow to Laboratory Equipment

By John L. Butler, Fishery Biologist
Southwest Fisheries Center
La Jolla, CA 92038

I recently interfaced a DEC Rainbow with an HEI Video Coordinate Digitizer through the Comm port. Data is sent to the Rainbow from the peripheral device via RS232 cable. Since data transmission is intermittent, the problem is similar to accepting data from a modem except that no crossover cables are required. The problems that I encountered were related to setting up the Comm port and the inability of higher languages (BASIC) to query the port fast enough to ensure data integrity. The DEC Hotline ws helpful up to a point on this problem but critical information was obtained from Jack Brown, a systems engineer at SWFC and Steve Green, a graduate student at UCSC.

The Comm port in the DEC Rainbow is not a simple data port like that found in other microcomputers. It is controlled by an Intel Chip. The PC100 specifications say that this is an INTEL 7201 but DEC provides documentation for the INTEL 8274. There are two addresses for the Comm port. The status port is 042H and the data port is 040H. The status port provides information to the computer indicating that data is ready at the data port. My program queries that status port and the console alternately and reads data from the data port when it arrives.

Before a reliable status can be obtained from the 8274 or 7201 whichever it is, the status port must be reset with sequences of commands. These commands are:

```
OUT 42H,18H
OUT 42H,14H
OUT 42H,44H
OUT 42H,13H
OUT 42H,C1H
OUT 42H,15H
OUT 42H,EAH
```

The baud rate at the Comm port is also under program control by writing to the 8088 port 06H. The statement OUT 06H,88H sets the port to read and write at 1200 baud, OUT 06H,EEH is 9600 baud. The programs that I have been working with were originally written in BASIC for an 8-bit CPM computer. I transferred these programs to the Rainbow but found that a simple loop with the read statement INP 40H was too slow to capture data consistently. The solution was to write an Assembly language program which was linked to the CBASIC program with LINK86. The subroutine must be defined as an external subroutine in CBASIC:

```
DEF  HEITWO(C1.ADR%),C2.ADR%,C3.ADR%) EXTRNAL
     C1% = 0
     C2% = 0
     C3% = 0
     P1% = VARPTR(C1%)
     P2% = VARPTR(C2%)
     P3% = VARPTR(C3%)
CALL HEITWO(P1%,P2%,P3%)
.pa
```

This CBASIC code defines HEITWO as an external function and passes data from the subroutine to the integers C1% to C3%.

The assembly routine was written by Jack Brown and looks like this:

```
PUBLIC   HEITWO

DATA PORT      EQU  40H      ;SERIAL DATA PORT
STATUS PORT    EQU  42H      ;SERIAL STATUS PORT
TXRDY          EQU  4        ;TRANSMITTER BUFFER EMPTY FLAG BIT
RXRDY          EQU  1        ;RECEIVED CHARACTER READY FLAG BIT
BDOS           EQU  OEOH     ;BSDOS INTERR

         CSEG                ;CODE SEGMENT

HEITWO:
         POP  CB RETURN      ;SAVE CBASIC RETURN POINTER

         MOV  CX,3           ;SET THE COUNTER

;INPUT ROUTINE

CHAR LOOP:

CALL           CONST          ;CHECK FOR KEYSTROKE
OR             AL,AL          ;SET FLAGS ON STATUS BYTE
JNZ            EXIT           ;ABORT ON EXIT CHAR
CALL           SERIN          ;GET A CHARACTER
XOR            AH,AH          ;ZERO THE HIGH BYTE
POP            BX             ;GET VARIABLE ADDRESS
MOV            [BX],AX        ;AND INSERT CHAR INTO MEMORY
DEC            CX             ;STEP DOWN THE COUNTER
JNZ            CHAR LOOP      ;LOOP UNTIL DONE
CALL           SERIN          ;GET CARRIAGE RETURN
CALL           SERIN          ;GET LINE FEED
JMP            EXIT           ;RETURN TO CBASIC

;GET THE STATUS OF THE SERIAL PORT

SERST:

IN             AL,STATUS PORT ;GET SERIAL INPUT STATUS
AND            AL,RXRDY       ;MASK FOR RECEIVE FLAG BIT
RET
;WAIT HERE FOR A CHAR FROM SERIAL PORT

SERIN:

CALL           SERST          ;GET INPUT STATUS
OR             AL,AL          ;SET CONDITION FLAGS
JZ             SERIN          ;WAIT FOR SERIAL INPUT IF 0
IN             AL,DATA PROT   ;ELSE GET BYTE
AND            AL,07FH        ;MASK OFF HI BIT
RET                           ; AND RETURN
```

```
CONST:

PUSH           CX             ;QUERY CONSOLE
PUSH           ES
MOV            CL,3
MOVE           DL,OFFH
INT            BDOS
POP            ES
POP            CX
RET
                              ; AND RETURN

EXIT:

OR             CX,CS           ;SET FLAGS ON CX REGISTER
JZ             EXIT1           ;SKIP IF COUNT ALREADY 0
POP            AX              ;REMOVE EXTRA POINTERS FROM STACK
DEC            CX              ;STEP DOWN THE COUNTER
JMP            EXIT

EXIT1:

PUSH           CB RETURN       ;REPLACE CBASIC RETURN POINTER
RET                            ;AND RETURN TO CBASIC
DSEG                           ;DATA SEGMENT

CB RETURN    RW      1         ;CBASIC RETURN ADDR STORAGE

      END
```

This subroutine will read three integers from the data port and strip the carriage return and line feed. It must be assembled with a relocating assembler such as RASM and linked to the BASIC program with a linker such as LINK86. This may seem like a lot of work to read in a few numbers from a peripheral device but once the system works you can collect a lot of data. We use this system to collect data on the age and growth of fishes, but it could as well be used to monitor any laboratory equipment.

DECmate

From the DECmate Working Group Chair, Cheryl Johnson

I have received many calls and letters from DECmate user's out ther this past month on a number of issues. I have been contacting DEC and letting them know what the user's are saying. If you have questions, or software or hardware suggestions, please LET ME KNOW! You can fill out the DECmate wish list ballot or the PC Postscript in the tear-out section in the back of the newsletters.

New products for the DECmate have been few these last months. We have all been looking forward to WPS Version 2.1. As you'll notice in the new products section, that's the only product listed. Some of the very nice features of Version 2.1 that we have had on our wish lists in the past are:

1) VT102 print port support. You can now print a document from your host to the printer connected to your DECmate through the printer port. This is something that we've been waiting for since the DECmate was introduced!

2) Editing and Indexing User Defined Keys. You now have the ability to edit an user defined key. This one has been on every wish list I've seen!

3) Increased use of fonts on the LN03. This feature of WPS Version 2.1 allows you to use other font families (with additional cartridges) to print documents.

As you will notice with our current wish list, there are many other enhancements to WPS that we are letting DEC know about.

The DECmate section is even sparser than usual this month due to the fact that gathering and editing a newsletter is VERY time consuming. I have included an article that will allow you to set up a command file on the VAX to move you back to the DECmate easily. You may however, wish to contact your system manager for help setting this up.

If you have an article that is DECmate or PDP-8 related, please send it to me. We need more articles on DECmates!

In the next issue I will report to you on the DECmate sessions in Dallas.

New Products

DECmate/WPS Version 2.1

DECmate/WPS, Version 2.1, makes it possible for you to integrate DECmate word processing into all kinds of office environments. Whether your office is small, large, or in between, you can start off with a single DECmate and expand, as needed, into VAX-based applications.

DECmate/WPS, Version 2.1, offers you a number of enhancements over previous versions. These include additional Gold Keys and function keys, which provide faster performance and WPS-PLUS user compatibility; host local printer control that allows the host to direct output to the DECmate printer in CX mode (printer port control); and transporting user-defined keys (UDKs) and system settings which allow stored rulers and printer settings and UDKs to be transferred from Version 2.0 system diskettes to Version 2.1 system diskettes.

DECmate/WPS, Version 2.1, also provides you with editing, indexing, and printing functions for your UDKs, making them easier to use. With these functions, you can erase, index, and remove keystrokes any place within the UDK, see the first line of each UDK, and print all of the UDKs.

Another Version 2.1 enhancement is a word counter that automatically counts words within a document or region, a feature that helps you write to editors' guidelines. Expanded document destination codes to print to the communications port are now also available, providing you with printer support with the mini-exchange. The mini-exchange allows you to use letter-quality printers for you DECmate/WPS documents. In addition, Version 2.1 offers specific LN03 (firmware level 4.3) support. With WPS 2.1 and the LN03 printer, you can produce hardcopy output in a variety of print styles and sizes.

With its ease-of-use and increased productivity, DECmate/WPS Version 2.1 add considerably to the power and depth of your DECmate's word processing capabilities.

Reprinted with Permission of Perspective, Volume 4, Number 1; Copyright Digital Equipment Corporation.

DECmate Wish List

If you have items you want to include on the wish list, send to the DECmate Working Group Chair, Cheryl Johnson (address in the Steering Committee list).

1. Global Search and Replace: The ability to do more than one search and replace string at a time.

2. Text Wrap: Text should automatically wrap when editing so that the user doesn't have to advance through it.

3. Additional Paste Buffers: The ability to save and rearrange multiple segments of text at the same time.

4. Spelling Error Detection: The ability to check spelling of text easily and accurately without using CP/M (to be included on DECmate word processing software, possibly as an optional feature), -- or rearranging document in any way. The current DECspell is very slow and doesn't seem to be very reliable.

5. Simple Math Logic: The ability to have five-function math (addition, subtraction, multiplication, division and percentages) on DECmate word processing software, possibly as an optional

feature. Would like the ability to check columns of numbers and perform simple math operations with a minimal number of keystrokes. The current math software seems very complicated to use.

6. UDK's: The ability to actually do the UDK as the user is setting it up, not just displaying the typed keystrokes.

7. ENTER key: Would like it moved closer to the normal typing keypad.

8. Print Specific Pages: The ability to print any specified page when reset pages are used; i.e., print section II, page 8.

9. Graphics: The ability to easily draw vertical and horizontal lines around text that will print out to form boxes and charts, to be an inclusion in DECmate software.

10. Background Transmission: The ability to continue to use the system while transferring documents from the VAX, converting documents, using list processing and other forground functions.

11. Working with a "COPY" document on disk, not the actual copy, so that if the user makes an error, the original can be recalled and used again.

12. Multiple Wraps: The ability to have multiple wraps so the user doesn't have to keep changing rulers; possibly by using the GOLD TAB to indicate the specific tab wrap that you desire at that time.

13. Sub- and Superscripts Displayed: Show sub- and superscripts on the screen by actually moving the character up or down.

14. Printer Stop: The printer should have the ability to detect when the ribbon breaks (not just runs out), and stop.

15. Printer Detects Paper Out: The ability for the printer to detect when no paper is in the printer so it will not print on the platen.

16. Screen Scrolling: Have screen scroll a single character at a time in wide documents, rather than jump.

17. Wide Screen Setting to Stay with the Document: With Version 2.0, the user has the ability to chage the size of the screen but is not allowed to store the chosen setting with the document. The chosen setting should stay with the Editor Menu (Screen Width narrow or wide), to be stored with the document chosen for.

18. Cutting Text of Any Size: To have the ability to cut any amount of text.

19. Replacement Characters on Draft Printer: The ability to print replacement characters on a draft printer going through communications, particularly double underscores.

20. Advance Forward: To be able to advance directly to the end of a word, sentence, or segment of text.

21. Document View while Printing: The ability to view or edit a document while other pages of it are being printed.

22. Multiple Simultanious Print Queue: The ability to queue severl nonconsecutive pages of a document to the print at the same time; i.e., queue pages 2,6,8; pr at least be able to send page 8 of a document to the printer while page 2 of the same document is being printed.

23. Underline Recognized as Part of a Word: The ability for the DECmate to recognize underlines before text that are entered using select and the underline key to be recognized a part of the word, i.e., ______________846. This would make the system back up to the beginning of the underlines when backup word is pressed.

24. Larger Capacity in DEC DX Account: The ability to put more than 200 documents in a DEC DX account.

25. Recreate Index: The ability to automatically recreate the index of the DEC DX account if it becomes corrupt.

26. Caps-Lock Sound Recognition: A different keying sound when the Lock key is on to singal the operator.

27. Replacement Character View: The ability to see numbers/characters which have replacement characters, such as double underscores, on the screen.

28. Gold Get Document Page: Use Gold Get Docmt (Page) to get only certain specified pages of a document.

29. Page markers: Not removed when rulers or text are changed.

30. Delete Page Key: The ability to delete an entire page by specifying instead of selecting.

31. Double Underscore with Decimal Tabs: The ability to use double underscoring with decimal tabs without having the text moved over.

32. Simple Column Logic: The ability to set up multiple columns on a page by putting more than one set of marging in a ruler. This would allow the use of the sheet feeder in multicolumn printing.

33. Column Centering: The ability to easily center headings over a column of text.

34. A place to get detailed information addressing software for applications such as PDP8 real time data acc replacement.

35. A way to split out DX and compile it into OS278.

36. Address the comm port under COS 310.

37. FMS: The ability to use print screen with FMS forms that have line drawings.

38. Printer Compatibility: Full compatibility with the LA210 and LN03.

39. Gold Q for Quit on DECmate and Rainbow WPS-Plus.

40. The ability to use non-DEC printers.

41. WPS Index: The ability to call up the index in alphabetical order by document name.

42. Paste Buffer: The ability to add information on to the end of a paste buffer before pasting, i.e., cut a paragraph, then add another paragraph that was later in the text, then paste both paragraphs elsewhere in the document.

43. The ability to install ANYTHING on master menu (especially Type Easy).

44. TypeEasy on Main Menu: Have type easy be part of WPS, include TypeEasy as an option on the main menu like sort, listing processing, etc.

45. File without Changes: The ability to file a document without saving the changes.

46. Numeric Keypad: The ability to use the numeric keypad to enter numbers while in the editor.

47. The abililty in DECspell to add a word to your personal dictionary as you come to the word in your document.

48. WPS should have the ability to interrupt and stop an UDK from continuing to run when the operating system returns an UDK error.

Help!! Can YOU Help These People???

Looking for a third-party billing system for WPS on a DECmate I:
Contact: Carol Traub
Small Business Support Systems
5 Highland
Sharon, MA 02067

Looking for other DECmate users in the St. Louis area working with WPS, 5 Meg hard disk, CP/M, and Multi-Plan:
Contact: Jim Seibert
5277 Trailview Drive
Florissant, MO 63033

Going From VAX to DECmate... In a hurry

By Cheryl Johnson

I have found this little trick useful when going back and forth between the VAX (VMS) and the DECmate (WPS). You may want to call your VAX System Manager and get her/his help in setting this up on your computer.

Using the editor, EDT on the VAX, create a file, QUICK.COM.
Key in the following commands: (be sure that they are all uppercase)

```
$ SET NOON
$ SET NOVERIFY
$ WRITE SYS$OUTPUT "<ESC>[3!~"
$ LO
```

When you enter <ESC> you need to press the ESC key twice. Don't type in the characters.

At this point, you need to edit your LOGIN.COM file on the VAX, again I used EDT. Key in the following line:

```
$ quick :== @QUICK.COM
```

Now give it a try. When you are at the DCL prompt ($), type QUICK. The command file should automatically log you off the VAX and put you back to the DECmate main menu.

DECMATE WISH LIST BALLOT

se this ballot to show which items on the Wish List are most important to ou. Put the number of the most important item on the list in space 1, the ext most in space 2, etc.

.____
.____
.____
.____
.____
.____
.____
.____
9.____
10.____
11.____
12.____
13.____
14.____
15.____
16.____
17.____
18.____
19.____
20.____
21.____
22.____
23.____
24.____
25.____
26.____
27.____
28.____
29.____
30.____
31.____
32.____
33.____
34.____
35.____
36.____
37.____
38.____
39.____
40.____
41.____
42.____
43.____
44.____
45.____
46.____
47.____
48.____
49.____
50.____
51.____
52.____

lease add the following to the wish list:

____
____
____
____

omments:

ame: ____________________

ompany: ____________________

ddress: ____________________
____
____

ork Phone: ____________ Home Phone: ____________

ETURN BALLOTS TO: Cheryl Johnson
Grinnell College
P.O. Box 805
Grinnell, IA 50112-0810
(515) 236-2570

PRO

From the PRO Working Group Chair, Thomas Hintz

1986 PRO-300 Wish List Results

Thomas R. Hintz
University of Florida
Institute of Food and Agricultural Sciences
Gainesville, Florida 32611

All the score sheets are in and the votes have been tallied for the first PRO-300 wish list evaluation. I received responses from all over the U.S and from as far away as the Netherlands and Australia. Thanks for all of your input. In addition, most ballots that were received contained additional items to be added to the wish list. Since they were not a part of the original list and available for everyone to vote on they were not included in the tables provided.

I won't try to interpret or formulate any opinions about the results. I will leave that job up to you. However, some of these wish list items have been answered by new features in V3 of P/OS and other sources.

In a future article the unanswered wish list items will be added to the new items that were submitted with the last ballots. If you have additional items that you would like to see added they should be sent to me. You will again have a chance to voice your opinion about various items requested by PRO users. Don't miss your chance to make yourself heard.

Table 1 - priority based on points

Table 2 - priority based on votes

Table 3 - priority based on both points and votes

New Products

P/OS Hard Disk V3.0

Digital's new P/OS Hard Disk V3.0 provides you with new networking options and software as well as new supported products not previously available for your Professional operating system.

New Networking Options

With the P/OS Hard Disk V3.0 networking options, you can group your Professional workstations into a local network in which one Professional system acts as a file and print server to the others. The typical configuration contains one server and four to eight workstations, depending on application mix and shared disk usage. With this combination, you can obtain powerful local processing and shared group resources. Non-shared resources can also be stored on the server and accessed by a workstation registered on that server. PRO/DECnet V2.0 is a prerequisite to this functionality.

In addition, a local Professional network that contains a file and printer server can now be part of a wide area network that links VAXs, PDP-11s, and other systems. Whatever your special requirements, your Professional workstation team configuration can be easily tailored to meet them.

New Software Options

P/OS Hard Disk V3.0 supports the PRO/Tool Kit V3.0 and all its layered languages. PRO/Tool Kit V3.0 features improved integration of the operating system and the development environment, streamlining many programming tasks and allowing more complete customization of the development environment to the requirements and preferences of the programmer.

New Supported Products

This new version of the P/OS Hard Disk adds support for several new devices such as the LN03 laser printer. Support for the DECtouch touch screen terminal, which originally shipped with the DECtouch hardware, is also now incorporated into the P/OS kit. This new version of P/OS includes support for alternate input devices, such as mouses and bit pads, through the Positional Device Interface. Other new supported devices are the Quad Serial Line Unit (SLU) and PRO Expander.

The Quad SLU is a single-slot option that provides you with four additional RS232 serial lines for your Professional. You can use the SLU to connect any serial device supported by the P/OS operating system, including terminals, printer, or plotters. You can use the SLU to attach multiple printers to a single Professional workstation or workstation network. In applications developed with the PRO/Tool Kit or the Professional Host Tool Kit, you can use the SLU to link multiple terminals to the Professional for multiterminal applications at a low cost.

Another new option available with P/OS Hard Disk V3.0 is PRO Expander, and external hard disk storage box that can securely contain a separately purchased hard disk for Professional 380 systems. Your Professional 380 can now have dual hard disk drives for a total storage capacity of up to 66 Mbytes.

Current users with the Update Service will automatically receive Version 3.0. Current users who do not have the Update Service need to purchase an H-kit in order to receive the new version.

Reprinted with Permission of Perspective, Volume 4, Number 1; Copyright Digital Equipment Corporation.

1986 DEC PROfessional 300 wish list voter results

(priority based on votes)

```
Points Votes  Both  Item
------ -----  ----- -----------------------------------------------------------------------------------------------------
   2     1      1   Diskette formatting capability
   8     2      4   Disk compression utility
   3     3      3   Ability to connect two hard disks to the bus, preferably utilizing the same controller
  10     4      6   A method to unfragment disk space without a complete reload
   1     5      2   External disk(s)
   7     6      5   BUS extension to provide more expansion slots
  11     7      9   Warm restart for P/OS
  21     8     12   Master index to toolkit documentation
   6     9      7   Streaming tape backup
  11    10     10   Removable hard disk
  12    11     11   Terminal emulation with full VT24X (e.g. downloadable characters, etc.)
  18    12     13   Menu item to execute applications from diskette so that install/deinstall is not required
   4    13      8   Standalone backup
  20    14     17   A search command similar to what exists on VMS for finding text strings in a file or files
  32    15     23   Instructions for deleting unused application options to conserve disk space
  15    16     14   DCL sources on Toolkit
  16    17     16   Batch spooling facility on toolkit for compilation/link
  22    18     21   DECNET support for the communications port
  25    19     22   Full window/multi-task support for user developed applications (i.e. SIDEKICK)
  13    20     15   SIGHT - convert multiple objects into single one
  17    21     20   BATCH processing support
  29    22     25   TPU (Text Processing Utility) for P/OS like the announced but not yet delivered text editor for VMS
  38    23     31   I & D space support for 325 and 350
  14    24     19   Virtual terminal support
  30    25     27   Image backup of Winchester disk
  34    26     30   Ability to dial into the TMS and the communications port from a remote terminal
  27    27     26   DECNET command terminal should be compatible with VMS 4.x (e.g. should allow command line editing)
  36    28     32   PRO/FMS compatible with VMS/FMS
   9    29     18   RT-11 emulator executing under P/OS
  19    30     24   Full VT125 emulation
  26    31     28   Access to the DZ driver or use of the other track/sector combinations supported by RX50 controller
  33    32     34   Enhanced CAD software (not Design Graphics Executive!)
  37    33     36   Asynchronous dial-up DECNET
  39    34     39   A more elaborate PRO/COMMUNICATIONS that makes full use of the TMS and voice box
  24    35     29   FPJ-11 floating Point for 380
  31    36     35   Wild card specification for PFT
  35    37     38   More compatibility between FMS screen control keys (function keys) and the P/OS menu control keys
  41    38     40   SIGHT - rotation of figures and text
  43    39     41   Graphics from VAX using PRO/COMMUNICATIONS V2.x
  44    40     42   GIDIS to REGIS conversion
  45    41     43   Supervisor mode support for 325 and 350
  51    42     45   Full DECGRAPH/DECSLIDE support
  23    43     33   MENU sources
  28    44     37   Extending graphics for PRO/BASIC
  48    45     44   GIDIS to/from NAPLPS conversion
  54    46     51   Graphics support over DECNET
  50    48     50   PRINT SCREEN to a GIDIS file
  52    49     53   SIGHT - Larger buffer for drawing complex filled figures
  46    49     46   HELP on floppy, less on hard disk
  55    50     54   Maintain color setting when going between menues
  47    51     49   Complete documentation and listing of the VT102 and GIDIS (TFW) code
  49    52     52   Output CORE graphics to GIDIS files (from PRO/BASIC)
  53    53     55   Startup detect of battery backup status
  42    54     47   VAX server for cluster of PROs
  57    55     56   A better DEC LANDER!
  58    56     58   Complete documentation set, etc. on laser disk (with cross index)
  40    57     48   Ability to spawn a BASIC compilation
  56    58     57   Reliable PASCAL compiler
  59    59     59   Videotex creation
  60    60     60   DECNET support under RT-11
  61    61     61   Coexistance of multiple OS on a single hard disk for each to run in native mode
  62    62     62   Do not clear screen after logoff with PRO/COMM TMS
-------------------------------------------------------------------------------------------------------------
R a n k ------ 1 = highest priority item, 62 = lowest priority item
P o i n t s -- rank based on voter prioritization
V o t e s ---- rank based on number of votes for item, ignoring prioritization
B o t h ------ rank based on both point rank and vote rank
```

1986 DEC PROfessional 300 wish list voter results

(priority based on points)

```
Points Votes  Both  Item
------ -----  ----- -----------------------------------------------------------------------------------------------------
   1     5      2   External disk(s)
   2     1      1   Diskette formatting capability
   3     3      3   Ability to connect two hard disks to the bus, preferably utilizing the same controller
   4    13      8   Standalone backup
   6     9      7   Streaming tape backup
   7     6      5   BUS extension to provide more expansion slots
   8     2      4   Disk compression utility
   9    29     18   RT-11 emulator executing under P/OS
  10     4      6   A method to unfragment disk space without a complete reload
  11     7      9   Warm restart for P/OS
  11    10     10   Removable hard disk
  12    11     11   Terminal emulation with full VT24X (e.g. downloadable characters, etc.)
  13    20     15   SIGHT - convert multiple objects into single one
  14    24     19   Virtual terminal support
  15    16     14   DCL sources on Toolkit
  16    17     16   Batch spooling facility on toolkit for compilation/link
  17    21     20   BATCH processing support
  18    12     13   Menu item to execute applications from diskette so that install/deinstall is not required
  19    30     24   Full VT125 emulation
  20    14     17   A search command similar to what exists on VMS for finding text strings in a file or files
  21     8     12   Master index to toolkit documentation
  22    18     21   DECNET support for the communications port
  23    43     33   MENU sources
  24    35     29   FPJ-11 floating Point for 380
  25    19     22   Full window/multi-task support for user developed applications (i.e. SIDEKICK)
  26    31     28   Access to the DZ driver or use of the other track/sector combinations supported by RX50 controller
  27    27     26   DECNET command terminal should be compatible with VMS 4.x (e.g. should allow command line editing)
  28    44     37   Extending graphics for PRO/BASIC
  29    22     25   TPU (Text Processing Utility) for P/OS like the announced but not yet delivered text editor for VMS
  30    25     27   Image backup of Winchester disk
  31    36     35   Wild card specification for PFT
  32    15     23   Instructions for deleting unused application options to conserve disk space
  33    32     34   Enhanced CAD software (not Design Graphics Executive!)
  34    26     30   Ability to dial into the TMS and the communications port from a remote terminal
  35    37     38   More compatibility between FMS screen control keys (function keys) and the P/OS menu control keys
  36    28     32   PRO/FMS compatible with VMS/FMS
  37    33     36   Asynchronous dial-up DECNET
  38    23     31   I & D space support for 325 and 350
  39    34     39   A more elaborate PRO/COMMUNICATIONS that makes full use of the TMS and voice box
  40    57     48   Ability to spawn a BASIC compilation
  41    38     40   SIGHT - rotation of figures and text
  42    54     47   VAX server for cluster of PROs
  43    39     41   Graphics from VAX using PRO/COMMUNICATIONS V2.x
  44    40     42   GIDIS to REGIS conversion
  45    41     43   Supervisor mode support for 325 and 350
  46    49     46   HELP on floppy, less on hard disk
  47    51     49   Complete documentation and listing of the VT102 and GIDIS (TFW) code
  48    45     44   GIDIS to/from NAPLPS conversion
  49    52     52   Output CORE graphics to GIDIS files (from PRO/BASIC)
  50    48     50   PRINT SCREEN to a GIDIS file
  51    42     45   Full DECGRAPH/DECSLIDE support
  52    49     53   SIGHT - Larger buffer for drawing complex filled figures
  53    53     55   Startup detect of battery backup status
  54    46     51   Graphics support over DECNET
  55    50     54   Maintain color setting when going between menues
  56    58     57   Reliable PASCAL compiler
  57    55     56   A better DEC LANDER!
  58    56     58   Complete documentation set, etc. on laser disk (with cross index)
  59    59     59   Videotex creation
  60    60     60   DECNET support under RT-11
  61    61     61   Coexistance of multiple OS on a single hard disk for each to run in native mode
  62    62     62   Do not clear screen after logoff with PRO/COMM TMS
-------------------------------------------------------------------------------------------------------------
R a n k ------ 1 = highest priority item, 62 = lowest priority item
P o i n t s -- rank based on voter prioritization
V o t e s ---- rank based on number of votes for item, ignoring prioritization
B o t h ------ rank based on both point rank and vote rank
```

1986 DEC PROfessional 300 wish list voter results

(priority based on both points and votes)

Points	Votes	Both	Item
2	1	1	Diskette formatting capability
1	5	2	External disk(s)
3	3	3	Ability to connect two hard disks to the bus, preferably utilizing the same controller
8	2	4	Disk compression utility
7	6	5	BUS extension to provide more expansion slots
10	4	6	A method to unfragment disk space without a complete reload
6	9	7	Streaming tape backup
4	13	8	Standalone backup
11	7	9	Warm restart for P/OS
11	10	10	Removable hard disk
12	11	11	Terminal emulation with full VT24X (e.g. downloadable characters, etc.)
21	8	12	Master index to toolkit documentation
18	12	13	Menu item to execute applications from diskette so that install/deinstall is not required
15	16	14	DCL sources on Toolkit
13	20	15	SIGHT - convert multiple objects into single one
16	17	16	Batch spooling facility on toolkit for compilation/link
20	14	17	A search command similar to what exists on VMS for finding text strings in a file or files
9	29	18	RT-11 emulator executing under P/OS
14	24	19	Virtual terminal support
17	21	20	BATCH processing support
22	18	21	DECNET support for the communications port
25	19	22	Full window/multi-task support for user developed applications (i.e. SIDEKICK)
32	15	23	Instructions for deleting unused application options to conserve disk space
19	30	24	Full VT125 emulation
29	22	25	TPU (Text Processing Utility) for P/OS like the announced but not yet delivered text editor for VMS
27	27	26	DECNET command terminal should be compatible with VMS 4.x (e.g. should allow command line editing)
30	25	27	Image backup of Winchester disk
26	31	28	Access to the DZ driver or use of the other track/sector combinations supported by RX50 controller
24	35	29	FPJ-11 floating Point for 380
34	26	30	Ability to dial into the TMS and the communications port from a remote terminal
38	23	31	I & D space support for 325 and 350
36	28	32	PRO/FMS compatible with VMS/FMS
23	43	33	MENU sources
33	32	34	Enhanced CAD software (not Design Graphics Executive!)
31	36	35	Wild card specification for PFT
37	33	36	Asynchronous dial-up DECNET
28	44	37	Extending graphics for PRO/BASIC
35	37	38	More compatibility between FMS screen control keys (function keys) and the P/OS menu control keys
39	34	39	A more elaborate PRO/COMMUNICATIONS that makes full use of the TMS and voice box
41	38	40	SIGHT - rotation of figures and text
43	39	41	Graphics from VAX using PRO/COMMUNICATIONS V2.x
44	40	42	GIDIS to REGIS conversion
45	41	43	Supervisor mode support for 325 and 350
48	45	44	GIDIS to/from NAPLPS conversion
51	42	45	Full DECGRAPH/DECSLIDE support
46	49	46	HELP on floppy, less on hard disk
42	54	47	VAX server for cluster of PROs
40	57	48	Ability to spawn a BASIC compilation
47	51	49	Complete documentation and listing of the VT102 and GIDIS (TFW) code
50	48	50	PRINT SCREEN to a GIDIS file
54	46	51	Graphics support over DECNET
49	52	52	Output CORE graphics to GIDIS files (from PRO/BASIC)
52	49	53	SIGHT - Larger buffer for drawing complex filled figures
55	50	54	Maintain color setting when going between menues
53	53	55	Startup detect of battery backup status
57	55	56	A better DEC LANDER!
56	58	57	Reliable PASCAL compiler
58	56	58	Complete documentation set, etc. on laser disk (with cross index)
59	59	59	Videotex creation
60	60	60	DECNET support under RT-11
61	61	61	Coexistance of multiple OS on a single hard disk for each to run in native mode
62	62	62	Do not clear screen after logoff with PRO/COMM TMS

Rank ------ 1 = highest priority item, 62 = lowest priority item
Points -- rank based on voter prioritization
Votes ---- rank based on number of votes for item, ignoring prioritization
Both ------ rank based on both point rank and vote rank

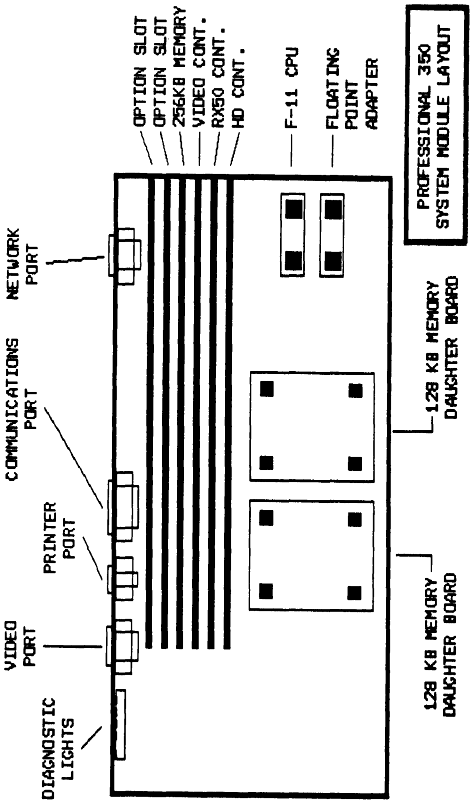
Memory Expansion for the PRO-350

by

Thomas R. Hintz
University of Florida

When the PRO-350 was announce several years ago it first appeared with 256KB memory (P/OS V1.0) and was advertised as having four expansions slots. Since all units are used as single user systems (prior to clustering with P/OS V3.0) the hard disk should not be considered an option but a mandatory item. Therefore, only three expansion slots are really available. All the room you need, right? Well, yes and no.

If you were going to limit yourself to the basic operation of the system and not add many options it was adequate. But, for those of us who needed or wanted the various options available, the expansion slots were used up very quickly. To make matters worse, it was soon discovered that the P/OS upgrade would require a minimum of 512KB memory. Although this hardware memory upgrade was provided free, it required a single 256KB board in one of the expansion slots leaving only two expansion slots available. This is not what I wanted to hear!



With the recent announcement of P/OS version 3.0, there is support for even more optional hardware that requires additional option slots. Where can it all go? Well, at the present time the only expansion box available for the PRO can only be used for an external hard disk drive. A nice feature to have but it also takes up an expansion slot. I would prefer an expander box that also gives me more option slots. Some of us want our cake and be able to eat it to.

Since expansion slots are at a premium, the recovery of just one slot would be extremely helpful. But which option should go? Why should any option have to go? Get rid of the 256KB memory board! But how, 512KB is required to boot?

Several solutions are possible for getting rid of the memory board in the expansion slot. The first method is simple but expensive. This method is chosen by those of us who do not have the time or experience to modify memory boards. It also is the prefered method if you are supported by a DEC maintenance contract since hardware modifications can and probably will void the contract.

If you fall into this first category you can purchase the 512KB memory option for the PRO-380 (MSC11-B). This option card can be used to replace one of the two 128KB daughter boards (part 5415084) connected to the main CPU board. With only one of the old 128KB daughter board and the new PRO-380 memory board (256KB memory removed from expansion slot) you now have 640KB total memory and an available option slot. Enough memory to boot with room to spare! If you want even more memory (and money is no problem!!!), both daughter boards of the PRO-350 can be replaced with the PRO-380 512KB option boards resulting in 1MB on the main board. As long as the system can find 512KB memory, the second daughter board does not even need to be installed (if sent for repair).

A second method is available for those of us who do not have extra money burning a hole in our pocket and can do some relatively simple replacements of memory chips. These 256KB chip sets can be purchased for one tenth the cost of the PRO-380 daughter board. The modification is almost trivial since DEC had the foresight to design both the daughter board and the system to handle larger chip sizes. It turns out that the system is completely self-mapping and it has the hooks to support both the 64K RAM (on daughter board) or the new 256K RAM chip set.

If the second category is more appealing to you, detailed instructions are provided in an excellent article written by Bruce Eteson. His article, "PRO 350 Memory Modification", can be found immediately following this article in the PC SIG Newsletter. Remember that this method will probably void any maintenance contract you have on your PRO. However, since you are still using the origional DEC daughterboard you might be able to negotiate with your local DEC repair facility for retaining normal maintenance on the rest of the unit? Give it a try. You might be surprised.

A third method is for those wishing to save money but do not have the time or experience to make the conversion themselves. This category can utilize the services of a third party vendor who will make the necessary modifications to your boards for you. If you are interested in having your PRO-350 daughterboards upgraded to 512KB each, most electronic technicians should be able to do the upgrade at a relatively modest cost (give them Eteson's article). The cost should be about one fourth the price of a PRO-380 daughter board.

One company that has done this conversion is Horizon Computer Services, Inc. They can be contacted at (201)420-5888 for the cost of a conversion. Other modifications to your PRO-350 can be made by this company such as disk controller upgrades. They can convert a 10MB disk controller so it can be used with a 20MB or 30MB hard disk drive. Call them for more information.

Be the first on your block to have a 1MB PRO-350 and three, available option slots!

PRO 350 MEMORY MODIFICATION

or

How to double your memory and free an option slot at the same time

by

Bruce M. Eteson
Copyright 1985

INEXPENSIVE MEMORY EXPANSION - I know a place where you can do more with less, where you can remove half of something to give you twice as much. Where? Inside the system unit of your 512-KByte Professional 350 computer! You can replace the memory chips in your Pro to multiply their capacity by four. This will allow you to have one megabyte of memory and free an option slot by removing the 256-KByte memory option card that came with your Pro.

SYSTEM REQUIREMENTS - Before I describe how this miracle is performed, I'll mention a few problems that DEC had the foresight to solve when the Pro was a twinkle in their corporate eye. DEC's design decisions made the Pro a little more expensive, but prevented the memory design from becomming obsolete when 256-Kbit dynamic random-access memory (RAM) chips became commonplace and attractive alternatives to 64-Kbit chips. Three areas in particular could have caused major headaches:

Addressing - The 256-K chips require nine address lines, one more than the 64-K chips need. Since these chips have only sixteen pins, the full address is split into two equal parts using what is known as a two-to-one multiplexer. Two parts, nine bits each for a 256-K chip and eight for a 64-K chip, are sent to the chip one after the other. They are known as the row and column addresses. DEC's Pro 350 design has nine address lines in the system-unit circuits that connect to the two memory daughterboards (little circuit boards that fit sockets on the main board and don't use option slots in the card cage). The daughterboards, in turn, have all nine address lines in their printed circuits. Finally, the 2:1 multiplexer in the system unit handles the eighteen inputs and nine outputs needed for the 256-K chips. The result? Since memory is addressed one sixteen-bit word at a time (sixteen chips in parallel), each daughterboard can hold up to 512 KBytes.

Memory Refresh - The memory chips are a variety known as dynamic RAM. The addressing circuits for the chips must be exercised frequently to prevent data loss as tiny electrical charges in the chips leak away. The refresh of all the bits in two rows takes place when a row address is imparted to the chip. This happens when a row is addressed for a read or write, but if a program somewhere else in memory is being executed it may be too long between refreshes. To overcome data loss in this situation, the Pro 350 contains circuitry that exercises the row addresses of the memory chips at a regular rate. Because the refresh takes place on two rows at a time, the counter that generates the addresses need be only seven bits for 64-K RAMs. The 256-K chips need eight, but tolerate being refreshed half as often. DEC's foresight gave the Pro a refresh counter that generates an eight-bit count.

Mapping - Most of us are familiar with the software end of memory management. Registers are loaded with the "upper" bits of the memory addresses to be accessed, then sixteen-bit "virtual" addresses are added to them to generate the 22-bit addresses used by the Pro. But, these 22-bit addresses have to be allocated to chips that accept only sixteen or eighteen bits. What's needed is logic circuitry to enable "banks" of memory. The upper bits of the 22-bit address can be used to select a bank. But how can the system tell if a daughterboard has 128-KBytes or 512 KBytes? In the Pro's case DEC provided a pin on the daughterboard connector for a logic signal. Not only that, but DEC also built the mapping logic to respond properly to the signal. You can have any combination of daughterboards with 128 or 512 KBytes!

TOOLS AND SUPPLIES - The modification of the daughterboards is quite simple if you have a good soldering iron and some experience in removing the solder from ICs inserted in plated-thru holes. You should perform this modification only if you are experienced in electronic service and safety methods. If you don't have such experience, you should work with a qualified technician. You can get by with the following tools and supplies:

Soldering Iron - A thermostatically controlled, fine tipped, pencil-type, iron will be best.

Desoldering Wick - This is fine braided wire impregnated with flux. Five or ten feet will be enough. Most solder suckers will have trouble pulling enough solder from some of the holes, especially the ones to the power busses. Note that after wicking or sucking the solder from a hole, the IC lead will often still be stuck to the side of the hole by a thin film of solder. To free the lead heat it and push it away from the wall of the hole with the soldering iron.

Solder - Use a fine size. Solder flow is sometimes better if a light coat of liquid flux is brushed on, but a good rosin-core electronic solder will usually be adequate.

DIP Sockets - These should be sixteen-pin, low-profile sockets. Each daughterboard needs sixteen sockets for its memory chips. I've been using a low-reliability type that doesn't have gold-plated contacts. The sockets aren't necessary, but after you desolder and remove a few of the chips, you'll understand why it's nice to have sockets on the board when a chip fails. The downside of sockets is that there may be occasional contact problems caused by corrosion.

Chip Puller - This is a tool like a pair of tweezers that has inward flanges at the tips. The flanges go under the ends of the chip being removed so you can pull them up and out of the board. If you try to pry out the chip with a screwdriver, you risk breaking the little capacitor next to the chip. When desoldered properly, the chips should pull out easily.

Flux Remover - Removing flux residue makes the job look professional. Use a fairly stiff brush - an old toothbrush is just fine - to completely eliminate the flux, then rinse. Avoid breathing the vapor or letting the liquid touch your skin.

256-K RAM Chips - Use a 41256-15 or equivalent. The -15 indicates a maximum access time of 150 ns. Some manufacturers use entirely different part numbers and speed codes, so don't panic if the dealer sends you chips with other numbers on them. The chips I used were NEC D41256C-15 parts. Be careful about the price you pay. The unit price for a chip should be around three or four dollars. The ads in Byte seem to be a good place to shop for a good price. For some reason DEC persists in selling 256-KByte chip sets (eight or nine chips) for the Rainbow for 695 dollars! That would be too much if they came to your house and installed them for you!

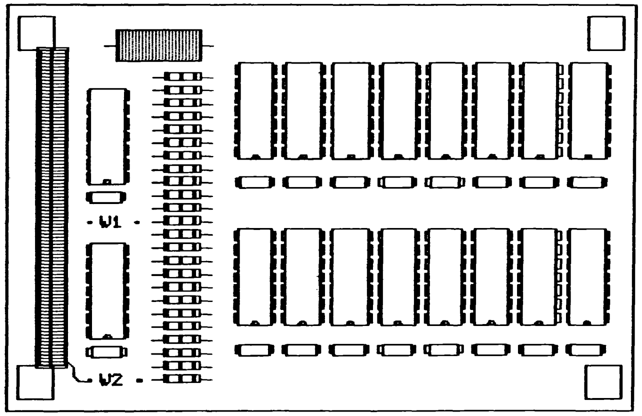


FIGURE 1. PRO-350 DAUGHTERBOARD

DAUGHTERBOARD ACCESS - Access to the daughterboards is straightforward. Turn off the system power and remove all cables from the system unit. Release the two catches - one on each end - that hold the cover. Lift the cover clear and set it aside. Disconnect the disk-drive power cords from the power supply box, the hard disk cables from the card connectors in the option card box, and the floppy disk cable from the top of the floppy drive. Press the retaining clip at the bottom center of each disk drive and slide the drive out half way. Unscrew the three knurled screws at the bottom rear of the drive trays - they hold the drive-tray part to the option-card box. Finally, slide the system board, with option-card box attached, back out of the system unit box. The power connector at the rear must be disconnected as the board is pulled out. All this will expose the daughterboards, which can be easily removed from their stand-offs. Assembly is the reverse of the removal process.

BOARD LAYOUT - Figure 1 shows approximately the component layout of a daughterboard. Note the position of the notch in each of the memory chips. The new chips must be installed in the same orientation. Be patient and take your time in removing the old chips. Don't overheat the board as you could damage it. Also, don't remove the two chips closest to the card connector.

STATIC ELECTRICITY - The chips are sensitive to electrostatic voltages. It helps to do this work on a sheet of grounded aluminum foil, especially in winter when the air is dry. The idea is to have you, the chips, the board, and the tip of the soldering iron at the same voltage. Connecting everything to ground works best. The grounds are usually done with a large series resistance to allow the static charge to leak away, but prevent dangerous currents from flowing from sources like the AC line.

JUMPERS - In addition to the new chips, two jumpers must be installed at the positions marked "W1" and "W2". The boards I've seen have "W1" and "W2" marked on the board. Three of the four contact points appear only as circular pads on the top of the board. The fourth has circuit foil going to it from under the connector. The W1 jumper is for the nineth address bit and connects pin 32 on the connector with pin 1 on each of the memory chips. The W2 jumper connects pin 39 of the connector to ground to indicate that 512 KBytes are present on the board. If bare wire is used for the jumpers, it mustn't make contact with any other parts of the circuit.

After making the change you'll notice that it takes five or six seconds longer for your Pro to do its self test after you turn it on. An added benefit of this change is that you can work around a memory fault. P/OS won't load if there's a memory fault or if there's less than 512 K in the machine. With only 512 K you're dead in the water until the memory is fixed. If a fault appears after you have the extra memory, you can remove the guilty card and keep working.

The author accepts no responsibility for direct or consequential damages to persons or equipment resulting from attempts to use the data in this article for any purpose. Persons using the tools or procedures described in this article do so entirely at their own risk. Any reproduction of all or part of this article must include this notice.

Needed! Updates for the PRO Software List

by Gary Rice

One of the issues raised at the Fall DECUS symposium was the lack of coordinated information from DIGITAL regarding the current versions of software available for personal computers. In an effort to provide this information, I have called several places and come up with a very incomplete list for the PRO computer family.

DEC Software	REV Currently Shipping
P/OS	3.0
PRO/Toolkit	3.0
PRO/Pascal	1.2
PRO/COBOL	2.4
PRO/Dibol	1.7
PRO/Comm	3.0
PRO/BASIC	1.2
PRO/Datatrieve	2.0
PROSE PLUS	2.0
PRO/SIGHT	1.0

3rd Party Software	Vendor	REV currently shipping
TK!Solver	Software Arts	1 (2A)
Supercomp-20	Access Technology	1.21
Online Disk Unfragmentor	By Hand	1.3

If you have received a shipment of software in the last month (and you DIDN'T get it in a fire sale), please compare the documented REV level to the one I have listed. If your software is more recent (or it isn't listed at all), please let me know so I can update my list. I will submit a new list to the SIG Newsletters every time I learn of a change to the info.

You can contact me in writing at:

Gary Rice
McDonnell Douglas Corp.
5701 Katella Ave.
Mail Code: K34-4W
Cypress, CA 90630

or by phone: (714) 952-6582

BACKGROUND PRINTING UNDER P/OS--2ND INSTALLMENT

By Harold Salwen, Stevens Inst. of Tech.

In the December issue, Steve Kirschner pointed out that one can print files in background without inhibiting other jobs requiring PIP by installing a second copy of PIP with a different name and spawning the print job using the new copy. A few comments are in order regarding the way he proposes doing it.

1) Spawn is not available in Command Language. This would have to be done in TOOLKIT.

2) The directory name [ZZAP00002] is dependent on the order in which applications are installed. The sure way to find the right file to edit is to use APPL$DIR:START.CMD. To avoid error messages when you leave and reenter TOOLKIT, and to make sure PIP is found, use,

```
.IFNINS PIPTT2 INSTALL/TASK:PIPTT2 APPL$DIR:PIP
```

(Not "$PIP" -- at least on my machine.)

3) Instead of editing START.CMD, one can simply type

```
INSTALL/TASK:PIPTT2 APPL$DIR:PIP
```

once (each day) after entering TOOLKIT.

4) Once PIPTT2 has been installed, one can print a file (say FILE.DOC) in background by the one line command:

```
SPAWN RUN/COMMAND:"PIP LP:=FILE.DOC" PIPTT2
```

5) Alternatively, one can use a command file PR1.CMD containing only one line:

```
SPAWN RUN/COMMAND:"PIP LP:='P1'" PIPTT2
```

by typing the command

```
@PRI FILE.DOC
```

('P1' stands for the string following @PR1 on the command line ".) ENABLE QUIET" can be added at the beginning if you don't want the SPAWN command to appear on the screen. To make PRI.CMD available in all default directories, put it in [1,2].

SERIAL COMMUNICATION BETWEEN PRO/350 and CAMAC

by Zhou Zhen-lei, John Hopkins Univ.

Oct., 1985

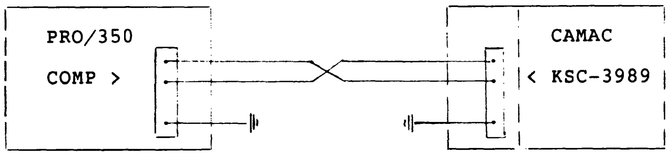
There is a serial communication port(COMP) on DEC PRO/350 microcomputer. It is the communications services that PRO/350 has been installed for the communications between PRO/350 and other computers, or behaving as(emulating) a terminal connected to a "host" system.

In this article, we place the emphasis on how to make PRO/350 COMP and a CAMAC crate controller KSC-3989 with a serial port communicate successfully. All the illustrative program segments are in Fortran-77 language.

Some characteristics for both COMP and KSC-3989 listed below:

	COMP	KSC-3989
I/O standard	RS-423(compatible with)	RS-232
Highest baudrate	19.2K	19.2K
Asynchronous mode	yes	yes
Xon/Xoff supported	yes	yes

Null-modem connection of COMP with KSC-3989:



<I> I/O format for KSC-3989 (refer to KSC-3989 manual)

```
Input format  1).  SOH CC NN,AA,FF STX EXT         (read CAMAC)
              2).  SOH CC NN,AA,FF STX             (control  CAMAC)
              3).  SOH CC NN,AA,FF STX DH,DM,DL EXT
                   (write CAMAC)

Output format      SOH CC STX DH,DM,DL EXT  (no status-reply)
              or   SOH  CC  STX  DH,DM,DL,REP  EXT  (status-reply
                   enabled)
```

SOH, STX, ETX are the communication format codes constituted by ASCII control characters, which ASCII values are in correspondence with 1, 2, 3 respectively.

CC is from the CAMAC crate number, but it appears as ASCII character which ASCII value is equal to the decimal number. For instance, if actual CAMAC crate number = 65, than CC = 'A', and so forth.

NN,AA,FF are CAMAC station number, subaddress and function code respectively, which make up one specific CAMAC command.

DH,DM,DL are 3-byte data which are only present in "write CAMAC" input and the possible response output to "read CAMAC"; REP is CAMAC status reply responded by KSC-3989 to input command.

'NN,AA,FF' or 'DH,DM,DL(REP)', all these decimal numbers and comma separaters will not be modified at all and will be transferred as ASCII characters.

So input and output both are ASCII codes. Nevertheless, we must make sense of specific definitions for them.

<II> ASCII code output at COMP

Due to lack of intrinsic dedicated function CHR$ which converts ASCII value to corresponding ASCII character on RSX-11M/M-PLUS system of PRO/350, perhaps we have to consider using DECODE, ENCODE commands to realize ASCII code conversion.

But we can make BYTE data and character data equivalent. Of course, the most convenient method is to take the advantage of BYTE specification for assigning the ASCII value to corresponding ASCII character, in order words, one BYTE variable will store one specific ASCII value; then we "write" this BYTE out of COMP according to the FORMAT of character string.

For example, output "read CAMAC" command SOH CC NN,AA,FF STX ETX ! If CC=65, NN=5, AA=1, FF=0, we can write down the following program:

```
      BYTE SOH, CC, STX, ETX
        ! CC is equivalent to CHARACTER*1.
      DATA SOH, CC, STX, ETX/1,65,2,3/
        ! CC is equivalent to 'A'
      WRITE(12,100)SOH,CC,STX,ETX
        ! If logical unit No. of COMP is 12.
100   FORMAT(1X,2A1,'5,1,0',2A1)
              !    Notice    the    '1X',    because    the    first
  ! step of output at COMP takes action as well as on line printer
```

Besides, we must also take into account of the features of serial communication, such as start-bit, data-bit, stop-bit, parity-bit and optional 'checksum'. All these must match one another between COMP and KSC-3989. KSC-3989 requires that effective data-bit be seven bits(7-bit ASCII code).

<III> COMP receives ASCII code input

If the software driver---communications services on PRO/350 not modified, routine CCRXD reads data(including ASCII control code) from the communication line at COMP. In the case of normal data input this routine must be called before any relative CAMAC command is to be issued for the reason of automatic reply of KSC-3989.
In length as following:

```
          CALL CCRXD(status,lun,efn,iosb,stadd,size,tmo)

! status      A two-word status block
! lun         The logical unit number for the communication line
! efn         An  event  flag  the system sets on completion of the
              operation
```

```
! iosb        A double-length array for the iosb
! stadd       A buffer into which the routine can place the data
! size        An integer specifying the input buffer size
! tmo         The time-out interval
```

It is necessary to discuss the choices of 'tmo'. 'tmo' is a 16-bit integer. Bits 0-7-----Number of ten-second intervals > up to 255 sec. maximum.
Bits 8-15----Number of one-second intervals >

For example: if tmo = 256+1 = 257, then time-out interval = 1+10 = 11 (sec.)

(1). If we do not specify this parameter, input is not complete until 'size' characters have been written to the buffer. It can be selected thus as we are not sure when the input data will appear.

(2). If 'tmo' is zero, the request returns immediately after transferring as many characters as are available up to the input buffer size. When Xon/Xoff was enabled at COMP, COMP will issue Xoff as input data buffer is full or CCRXD suddenly closed. Possibly this Xoff signal results in that a little part buffer not cleared up, most likely PRO/350 will receive unsolicited data which are superflous. Especially we have to deal with this problem when PRO/350 will receive data continuously. It can be selected thus as we want to clear the output buffer of KSC-3989 in the case of critical speed application.

(3). If 'tmo' is not equal zero, the request completes after the time-out interval specified. This is the general case as we can judge for how long the input data will have been collected.

Only after the previous CCRXD call completed, then the next one can start, or else we will be fail to call.

How to call CCRXD effectively and ingeniously seems to become one of the key links of the communication between PRO/350 and KSC-3989.

Despite the data types defined, we can redefine data types for some parameters:

```
   BYTE status(2),efn,iosb(4),stadd(...)    !  (...) buffer size
```

From "status" received, we will know if the CCRXD had been issued successfully. From "iosb", we will know how many characters CCRXD has just received on the communication line.

Since communications services can not respond to interrupt, probably only the polling method can be used to monitor CAMAC status. In practice, if CAMAC slot made KSC-3989 call for a break request to computer, PRO/350 will receive pertinent control character EOT(ASCII value = 4).

<IV> Operation among data received by PRO/350

Actually those data received by PRO/350 thru COMP are stored in BYTE array. But BYTE data is a 8-bit unit including the leading sign. The data of this type are not always suitable to involve in arithmetic operation, otherwise the operation could lead to an unpredicted result. Therefore, when the arithmetic operation is inevitable, those related BYTE data must be converted to INTEGER*2 type at first.

For example,

```
         BYTE stdd(16)            ! input buffer
         INTEGER*2 afstdd(16)     ! for conversion
         DO 200, I=1,16           !
200      afstdd(I)=stdd(I)        !  valid  data in afstdd for normal
                                  arithemetic operation.
```

<V> Auxiliary manual debugging

After Pro/350 entered terminal emulator, we control ECHO such that monitor screen is capable of displaying input and output messages transferred via COMP (except for ASCII control characters) simultaneously whether KSC-3989 is in Monitor-mode or Normal mode. At this time we can key in commands on keypad to send them to CAMAC by your demands. It has been proved that the of such manual debugging is very necessary. In this way, we can make PRO/350 implement communications with CAMAC step by step. The response of CAMAC will also bring us lots of convenience for troubleshooting.

Fortran-Graphics Application on PRO/350

by Zhou Zen-lei

DEC PRO/350 is a powerful microcomputer which is widely being applied in research Lab. and industry. They often meet graphics application in these areas. How to make full use of the resources of PRO/350 so that we can build up graphics programs easier and effectively, as a result, on monitor screen or line printer we will get objective images corresponding to previous programs, is a pratically valuable topic.

<I> Principles and Procedures of Graphics Programming

The Professional 300 series Core Graphics Library (CGL) is a general purpose graphics subroutine package, which makes available to the Fortran-77 programmer the powerful and versatile functions provided by PRO/350 video bitmap architecture.

The first step in graphics programming is to collect relative data, thereafter we can design pictorial representations of these data. Some arithemetic formula may be used.

The second step is to write the instructions that create the desired images.

CGL has four general types of instructions:

```
i).     Control
ii).    Viewing Transformation   / a) Global Attribute
iii).   Output Primitive--------|  b) Current Position and Marker
iv).    Attribute---------------|  c) Line
                                \  d) Text
```

<I-1>. Calling CGL routines from F-77

format: CALL CGL(inst_name,p1,p2,-----,pn)

"inst_name" is a designated acronym equivalent to an integer constant specifying the desired CGL instruction. "p1,----" are possible parameters.

e.g. : CALL CGL(GIC) is identical to CALL CGL(90),
here GIC means INITIALIZE_CORE.

<I-2>. A practical model of graphics program

They may be confused how to deal with so many instructions at the beginning. But so long as we hold on to those conventional instructions we will find the way to success.

N.B.: Most numeric data in this should be modified according to actual circumstances.

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
       PROGRAM GRAPH
C      'INCLUDE...' is necessary to take advantage of CGL package
       INCLUDE 'LB:[1,5]CGL.FTN'
       INTEGER COLOMP(24),COLOR(3)
       DATA  COLOMP/0,0,0,  7,0,0,  0,7,0,  0,0,7,  7,7,0,  0,7,7,
               7,0,7,
#                 7,7,7/, COLOR/1,4,6/
C     INITIALIZE CORE(first things first)
      CALL CGL(GIC)
C     NEW_FRAME(after 'INITIALIZE')
      CALL CGL(GNF)
C     SET_BACKGROUND_INDEX(take action after 'NEW FRAME','ERASE
       VIEWPORT')
      CALL CGL(GSBI,5)
C     SET_WINDOW(world coordinates)
      CALL CGL(GSW, 0., 100., 0., 100.)
C     SET_ORIGIN(the point addressed by the smallest world
       coordinate pair)
      CALL CGL(GSO,0)
C     SET_VIEWPORT(a portion of normalized device coordinate space)
      CALL CGL(GSV2,0.1875,0.8125,0.,1.)
C     SET_COLORMAP(set all eight colormaps RGB values)
c     CALL CGL(GSCM,COLOMP)
C     SET_WRITING_INDEX(current colormap entry)
      CALL CGL(GSWI,4)

      ---> (program 'body')

C     TERMINATE_CORE(the end of CGL)
      CALL CGL(GTC)
      END
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

We may use another instruction SET GLOBAL ATTRIBUTES, which sets the values of the entire global output primitive attribute list with a single call, instead of SET COLORMAP SET WRITING INDEX and SET BACKGROUND INDEX, etc. Notwithstanding, we prefer above separate instructions rather than the complex one for more explicit and mobile.

The corresponding Command File and Overlay Description File for F-77 graphics program are very similar to usual except for adding two items to Command File. Before calling Task Builder to link object modules with libraries, we must define global for graphics(GBLDEF=G$LUN:XX, here XX is an octal number) and insert CGLFPU into cluster libraries in Command File.

<II> Considerations about Graphics Programming

<II-1>.Data types

The data type and relative position of each parameter must match that expected by CGL. CGL cannot perform data type checking of parameter.

<II-2>.Freezing the images on screen

If we want to keep a complete image on the monitor screen, which is yielded from graphics program, sometimes we can't ignore the coming appearance of the cursor after this program completed. Because as soon as the program ends, computer returns to command status, the entire image on the screen will scroll up, correspondingly, to leave space for appearing cursor so that some top part of the current image will disappear unexpectedly. A very simple way to handle this is that we insert a 'dead' iteration just before 'END' statement, shown as the following:

```
  CALL CGL(GTC)                      Actually this program will not
  'dead' interation                  'END'   unless   you   press
  END                                "INTERRUPT DO"(as well as ^C).
```

<II-3>.Print screen

The CGL command "PRINT SCREEN", CALL CGL(GSC,x_offset,y_offset), only sends screen image in current viewport to LA50 printer. It is different from the real "PRINT SCREEN" key on the keypad.

x_offset the offset from left-margin on LA50
y_offset the offset from top-margin on LA50

The top-margin means the horizontal line by which the printing head was standing just before implementing CGL "PRINT SCREEN".

<II-4>.'Zig-Zag'

Due to the limitation of rasters scanned on screen, there are some distortions with all straight lines but horizontal ones. To some extent, the closer to horizontal direction exclusive, the worse the situation is. Possibly, some straight lines inevitably look like 'Zig-Zag' saw-tooth shaped. The plotter, which can draw continually, will overcome such inherent shortcoming of monitor screen.

<II-5>.User-defined font

Somewhere we would compare the differences between the programs in F-77 and in BASIC-PLUS-2.

relative instructions:

SET_FONT(select character font, fonts 1 through 3 can be redefined)
SET_FONT_SIZE(define size of character font)
LOAD_CHARACTER(load user-defined character)

i). Be sure to use neither TAB nor other control characters for preparing font data(user-defined character), because they cannot be displayed on screen but may occupy some positions in the user defined font as well as space character does.

If in F-77, use another BLOCK DATA subprogram to prepare the font size, font description data and others; If in BASIC-PLUS-2, user another TERMINAL-FORMAT file to prepare the same parameters as to take the advantage of LINPUT instruction. In both cases, we have considered to make these subprograms, which may be modified repeatedly, relatively independent from main programs.

ii). In general, the start point of any font is always at its top-left corner regardless of the 'window' origin-setting.

iii). As the main program is to draw the images of the user-defined character which font description data was just loaded from the subprogram, provided the corresponding DEC Multinational Character Set decimal code to this character is CH(an integer), then

```
In BASIC-PLUS-2 -----> CALL CGL BY REF(TEXT, CHR$(CH%),1)
```

It is unnecessary to worry about the lack of internal ASCII conversion function CHR$ in F-77 on PRO/350 computer with RSX-11M/M-PLUS. For instance, if CH=65, then its font position is in respect to the one where the original character 'A' in DEC Multinational Character Set loaded in 'font 0'; Nonetheless, we can write,

```
CALL CGL(GT,'A',1)
```

But at this time the main program will draw the user-defined character, not 'A'. It is that we are just expecting.

iv). In order to LOAD_CHARACTER, the main program must contain 'font editor'. This 'font editor' reads font descrition data from a subprogram and performs a string-to-integer conversion for each line in the array. The non-space bit at left-most column implies integer 32768(=2**15), which is more than the valid biggest INTEGER*2 value by one, or -32768; so far as we are concerned with the representation of INTEGER*2 values in computer memory, although the appearance of +32768 is referenced as overflow, in actual it is the same as -32768. From this point of view, we can explain why the result is almost the same whatever, +32768 or -32768, for the above purpose, if we overlooked some F-77 task-run warnings which told us integer overflow occured.

There is nothing wrong with choosing +32768 or -32768 on BASIC-PLUS-2 task running in this area.

v). A sample program for user-defined font: (on next page)

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
       BLOCK DATA OPTION
       INTEGER*2 EXTENT,COLOMP(24)
       CHARACTER SS(16)*16,CHR(10)
       COMMON /AREA1/EXTENT,COLOMP /AREA2/SS,CHR
       DATA COLOMP/0,0,0, 7,0,0, 0,7,0, 0,0,7, 7,7,0, 0,7,7,
       7,0,7,
     #             7,7,7/,EXTENT/1/,CHR(1)/' '/
C                        5432109876543210
       DATA  SS    /'        X  ', !1
     #       '      X         ', !2
     #       '      X         ', !3
     #       '      X         ', !4
     #       'XXXXXXXXXXXXXXXX', !5
     #       '      X         ', !6
     #       '      X         ', !7
     #       '      X         ', !8
     #       '      X         ', !9
     #       '      X         ', !10
     #       'XXXXXXXXXXXXXXXX', !11
     #       '      X         ', !12
     #       '      X         ', !13
     #       '      X         ', !14
     #       '      X         ', !15
     #       '      X         '/ !16
C             5432109876543210
       END
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
          PROGRAM MAIN
          INCLUDE 'LB:[1,5]CGL.FTN'
          INTEGER*2 COLOMP(24),CHAMAT(16),EXTENT
          CHARACTER SS(16)*16,CHR(10)
          COMMON /AREA1/EXTENT,COLOMP /AREA2/SS,CHR
          NX=LEN(SS(1))       ! font 'x' size
          DO 100, I=1,16 ,-1
          IF(LEN(SS(1)) .EQ. 0) GOTO 150
100       CONTINUE
          NY=16               ! font 'y' size = 16
          GOTO 170
150       NY=I-1              ! font 'y' size < 16
170       CALL CGL(GIC)       ! INITIALIZE_CORE
          CALL CGL(GNF)       ! NEW_FRAME
          CALL CGL(GSBI,5)    ! SET_BACKGROUND_INDEX
          CALL CGL(GSW, O., 100., 0., 100.)          ! SET_WINDOW
          CALL CGL(GSO,0)     ! SET_ORIGIN
          CALL CGL(GSV2,0.1875,0.8125,0.,1.)
                              ! SET_VIEWPORT_2
          CALL CGL(GSCM,COLOMP)                      ! SET_COLOR_MAP
          CALL CGL(GSWI,2)    ! SET_WRITING_INDEX
          CALL CGL(GSF,1)     ! SET_FONT TO 'font 1'
          EXTENT=31+EXTENT    ! font size
          CALL CGL(GSFS,0,NX,NY)                        ! clear font
          CALL CGL(GSFS,EXTENT,NX,NY)                ! SET_FONT_SIZE
          DO 1000,ICH=32,EXTENT          ! 'font editor' performs
string-to-integer
          DO 500,I=1,NY                                   !
conversion.
          N=0
          DO 200,J=NX-1,1,-1
          IF ((SS(I)((J+1):(J+1))).NE.(' ')) N=N.OR.(2**(15-J))
200       CONTINUE
          IF (SS(I)(1:1).NE.(' ')) N=N32768
                              ! certainly no integer overflow
          CHAMAT(I)=N
500       CONTINUE
          CALL CGL(GLC,ICH,CHAMAT)                 ! LOAD_CHARACTER
1000      CONTINUE
          CALL CGL(GSCS,35.,35.)                     ! SET_CHARSISE
          CALL CGL(GMA2,30.,60.)                       ! MOVE_ABS_2
          CALL CGL(GT,CHR(1),1)     ! TEXT  is equinalent to CALL
CGL(GT,' ',1)
          CALL CGL(GTC)       ! TERMINATE_CORE
          END
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Run task image of above program and we will get a chinese character 'shi' in green color on monitor screen.

PRO 300 SERIES WISH LIST BALLOT

Use this ballot to show which items on the Wish List are most important to you. Put the number of the most important item on the list in spsace 1, the next most in space 2, etc.

1.____
2.____
3.____
4.____
5.____
6.____
7.____
8.____
9.____
10.____
11.____
12.____
13.____
14.____
15.____
16.____
17.____
18.____
20.____
21.____
22.____
23.____
24.____
25.____
25.____
26.____
27.____
28.____
29.____
30.____
31.____
32.____
33.____
34.____
35.____
36.____
37.____
38.____
39.____
40.____
41.____
42.____
43.____
44.____
45.____
46.____
47.____
48.____
49.____
50.____
51.____
52.____
53.____
54.____
55.____
56.____
57.____
58.____
59.____
60.____
61.____
62.____
63.____
64.____

Please add the following to the wish list:

____

____

____

____

Comments:

Name: ____

Company: ____

Address: ____

____

____

Work Phone: ____________________ Home Phone: _____________________

RETURN BALLOTS TO:
Thomas R. Hintz
University of Florida
IFAS Computer Network
Building 810
Gainsville, FL 32611

RSTS

RST

RSTS

Chairman
Charles Mustain
Stark County School System
Louisville, OH

Symposium Coordinator
Scott W. Pandorf
Kittle's Home Furnishings
Indianapolis, IN

Assistant Symposium Coordinator
Wef Fleischman
Software Techniques
Cypress, CA

Newsletter Editor

Open

Library Representative
Susan Abercrombie
Ventrex Laboratories Inc.
Portland, ME

DEC Counterpart
Joel Arker
Digital Equipment Corporation
Merrimack, NH

Pre-Symposium Seminar Coordinator
Bruce Gaarder
Macalester College
St. Paul, MN

Wish Lists Coordinator
Neal E. Goldsmith
Software Techniques, Inc.
Cypress, CA

Vice SIG Chairman
Wish Lists & Tape Copy Coordinator
Philip Hunt
System Industries
Milpitas, CA

EDUSIG Liaison
George Wyncott
Purdue University Computing Center
W. Lafayette, IN

RSTS Product Planning Coordinator
Errol E. Ethier
Shrewsbury, MA

Members-At-Large
Ed Beadel
Instructional Computer Center
Oswego, NY

Scott Daily
Great Lakes Chemical Corp.
W. Lafayette, IN

Mark Gilmore
Cal State University
Long Beach, CA

Mark Hartman
Jadtec Computer Group
Orange, CA

Jeff Killeen
Information Design & Management
Hopedale, MA

Newton J. Munson
Rochester Institute of Technology
Rochester, NY

SIG ACTIVITIES REPORT

1986-87 FISCAL YEARS

Charles W. Mustain, chairman

April 11, 1986

RSTS SIG activities and services center around the traditional channels of DECUS services. They are:

1. Symposium and Pre-Symposium Sessions.
2. User-to-user help via telephone.
3. Channeling user input to DEC product development staff and product management staff.
4. User-to-user communication via newsletter.
5. Support of other activities within DECUS by providing volunteer representatives to these groups and participating in their activities and services.

In support of these, the following activities were conducted in FY86 and/or are planned for FY87:

1. Three Pre-Symposium Seminars were offered in FY86. Only one was subscribed to sufficiently to be held. Plans are for two seminars in FY87.

   In FY86, 47 sessions were presented at the two National Symposia. It is anticipated that this number will grow slightly in FY87 as a result of planning activity that should produce better-targeted sessions. The SIG also provides either a campground or suite at each symposium with scheduled times for users to meet with RSTS developers. Additional times are scheduled to meet with RSTS "gurus" from the SIG. This activity will be continued in FY87.

2. Telephone support for users was largely informal in FY86, with queries generally channeled to the chairman and then "farmed out" to users who had helpful information as needed. Planning to formalize this effort and provide DECUS office with a list of RSTS folk who can answer questions will go forward in FY87.

3. Four meetings of the RSTS SIG Standing Committee for Product Planning (RSTS PPC) were held at the two National DECUS Symposia in FY 86. This is our regular annual schedule. When major changes in the RSTS product are contemplated by DEC, a "woods" meeting is sometimes scheduled. One such session is scheduled for FY87.

   PPC meetings are closed sessions under non-disclosure agreements where future hardware and software projects from DEC are discussed and input solicited very early in the product planning cycle. In addition to these closed sessions, the SIG carries out an active "Wish List" project, providing a channel for broad user input to the PPC and a medium for feedback to members on progress or lack thereof.

4. Newsletter activity for the SIG was at a long-time low in FY86. Only two issues carried RSTS articles. This has been addressed by the chair and the Steering Committee for the remainder of the year and for FY87.

   Working through the SIG counterpart, a committment has been made by the RSTS development group to provide at least six articles over the next twelve months. The RSTS SIG steering committee has committed to one article per person. This will provide a base to keep the "pipeline" full and insure at least one article per issue of the combined newsletters for the next year. It is expected that this regular production will encourage others to contribute.

5. The RSTS SIG has and expects to continue to provide volunteers to other groups within DECUS both as representatives of the SIG and as quality volunteers to assist these groups in fulfilling their missions and goals.

The following Operating Procedures for the RSTS SIG have been approved by the Steering Committee and submitted to the SIG Council and the Board of Directors of DECUS for approval as a part of the SIG Relicensing procedure. Due to the lead time for this publication, approval may have been given by the time you read this.

Charles W. Mustain, RSTS SIG chair 4/11/86

DECUS

RSTS Special Interest Group

Operating Procedures

Article I

Name of Organization

1.0 The name of this organization is the "RSTS Special Interest Group", hereinafter referred to as SIG.

Article II

Purpose and Affiliation

2.0 The SIG is established as a SPECIAL USER GROUP under the bylaws of the DECUS/U. S. Chapter.

2.1 The SIG is established, as empowered by the DECUS bylaws, to serve its members having a common interest as follows:

2.1.1 Promote the interchange of information and ideas concerning the utilization of computers, computer peripheral equipment, software, and other products or services manufactured, marketed, endorsed, or otherwise made available by or through Digital Equipment Corporation (DEC) as pertains to RSTS products.

2.1.2 Advance the art of computer usage through mutual education and exchange of ideas and information.

2.1.3 Establish standards and provide channels to facilitate the exchange of computing techniques and related information among SIG members.

2.1.4 Provide feedback to DEC concerning equipment, software, product services, product enhancement, and other needs which may arise.

Article III

Membership of SIG

3.0 Any DECUS member expressing a desire to become a member of the SIG is accepted as a member.

Article IV

Structure Of The SIG

4.0 The administration of the SIG is entrusted to the RSTS SIG Chair, the Executive Committee, and the Steering Committee.

4.1 The SIG Chair shall be the official representative of the SIG and is empowered to make decisions on behalf of the SIG in all matters not specifically delegated to another officer by this document.

4.2 All officers of the SIG shall be members of the Steering Committee for the duration of their term of office.

4.3 Ad hoc members of the Steering Committee shall be volunteers managing projects for the SIG. Their term shall be for the life of their leadership of the project.

4.4 Other active RSTS volunteers may be added as at-large members of the steering committee at the request of any SIG officer and with the consent of a majority of the SIG Executive Committee.

4.5 The SIG Executive Committee shall be composed of those officers defined in Article V of this document.

4.6 All members of the Steering Committee shall be voting members.

4.7 The SIG Steering Committee shall, at a minimum, meet at each national DECUS Symposium.

4.8 All meetings of the Steering Committee shall be open to attendance by any RSTS SIG member.

4.9 Any member may, by submitting in writing to any SIG officer at least four weeks in advance, bring a matter before the Executive Committee or Steering Committee for consideration.

Article V

SIG Officers

5.0 With the exception of the Chair, all officers are appointed by the chair subject to the following procedures:

5.1 Terms of office for all officers shall be two years from appointment or election.

5.2 Upon expiration of term or vacancy in office for any reason, the SIG Chair shall, choosing the most timely method, notify the steering committee and, via Business Meeting or DECUS SIGs Newsletter, notify the active membership of the SIG of the vacancy. The Chair shall solicit volunteers for the post.

By no later than the DECUS National Symposium first following the vacancy, the Chair shall select from those volunteering a nominee to submit to the steering committee for confirmation. In the event of non-approval, the Chair shall propose alternate choices from the list of volunteers. In the event none are available or approved, the Chair's original appointment shall prevail.

5.3 Officers of the SIG shall include official representatives of the SIG to other DECUS organizational Units, the Chairs of the Standing Committees and the SIG Chair. In the absence of sufficient volunteers, one person may hold more than one office.

The list of officers includes, but is not limited to, the following who meet the criteria in this item:

5.3.1 The Newsletter Editor is responsible for soliciting, reviewing, and/or writing RSTS contributions to the DECUS SIGs Newsletter.The editor is the representative of the SIG at all meetings of Newsletter Editors called by the organizational unit reponsible for such meetings. DECUS Publications Committee.

5.3.2 Communications Committee Representative: This person represents the SIG at all meetings and in all matters arising in this committee.

5.3.3 Library Coordinator: This person represents the SIG on the DECUS Library Committee at all meetings and in all matters arising in this committee.

5.3.4 Symposium Coordinator: This person is responsible for the symposium organization and scheduling of RSTS sessions at the National DECUS Symposia in direct cooperation with the SIG Counterpart(s) and the SIG membership. This person represents the SIG at all meetings and in all matters arising in the Symposia Committee.

5.3.5 Tape Copy Coordinator: This person is responsible for the construction and dissemination of the Symposium SIG tapes.

5.3.6 Wish-List Coordinator: This person is responsible for organizing the requests submitted to the SIG for improvement of RSTS, submitting them to DEC for review and tracking the status of each such request. This person shall be a member of the Product Planning Committee.

5.3.7 Product Planning Committee Chair: This person shall be responsible for managing the RSTS futures activity conducted under non-disclosure, confidential conditions as a cooperative activity between DECUS and DEC.

5.4 Each SIG officer should seek a volunteer to serve as vice-officer. This vice-officer shall be a non-voting member of the executive committee, except when serving in place of the respective officer, and a voting member of the steering committee.

5.5 Recall of Officers:

5.5.1 Any member of the Steering Committee or Executive Committee may be removed from office by a 2/3 majority vote of the Steering Committee. A motion to remove may be presented by any SIG officer.

5.5.2 Removal from office may occur only as the result of a formal meeting of the Steering Committee. Any individual being considered for removal shall be notified in advance of such meeting and afforded the opportunity to appear and be a part of any such meeting.

5.5.3 Petition by 50 or more members, defined as those present at a SIG Business Meeting and/or those identified as RSTS SIG members in the DECUS membership system, shall be sufficient cause for a removal hearing to be held. Petitioners shall be afforded the opportunity to have a representative speak at the hearing.

5.6 The SIG Chair shall be elected by the Executive Committee upon expiration of term or vacancy in office for any reason. The Chair's term shall be two years from election.

5.6.1 Upon vacancy in the Chair, nominations for a new chair shall be solicited from and may be made by any member of the Steering Committee, or by any active member at a business meeting of the SIG.

Article VI

Standing Committees

6.0 The Product Planning Committee is established as a Standing Committee.

6.0.1. The Product Planning Committee is entrusted with the primary planning relationship with Digital. Members are appointed by the SIG Chair and approved by Digital. The committee will consist of a representative cross section of RSTS users. It's membership will consist of 10 persons, with a core group of five selected by the SIG Chair with the advice and consent of the PPC Chair.

6.0.2. Members of the Product Planning Committee serve at the pleasure of the SIG Chair and Digital. As vacancies occur for any reason, replacements shall be appointed by the procedures in item 6.0.1, above.

6.0.3. The SIG shall abide by any restrictions placed on the membership of the Product Planning Committee by Digital. Any such restrictions take precedence over this article.

6.1 The Chairman may establish or abolish other standing committees with the approval of the Steering Committee.

Article VII

Interpretation

7.0 The SIG Chair is the final authority for interpretation of these procedures.

7.1 Where necessary in these procedures, the male gender includes the female, the singular number includes the plural, and the plural number includes the singular.

Article VIII

Amendments

8.0 Amendments to these procedures may be proposed by the Steering Committee or by written petition of fifteen members of the SIG.

8.1 Amendments shall be ratified by a two-thirds majority of the Steering Committee.

8.2 Amendments to these procedures shall not conflict with any provision of the DECUS/U. S. Chapter bylaws.

Article IX

Implementation

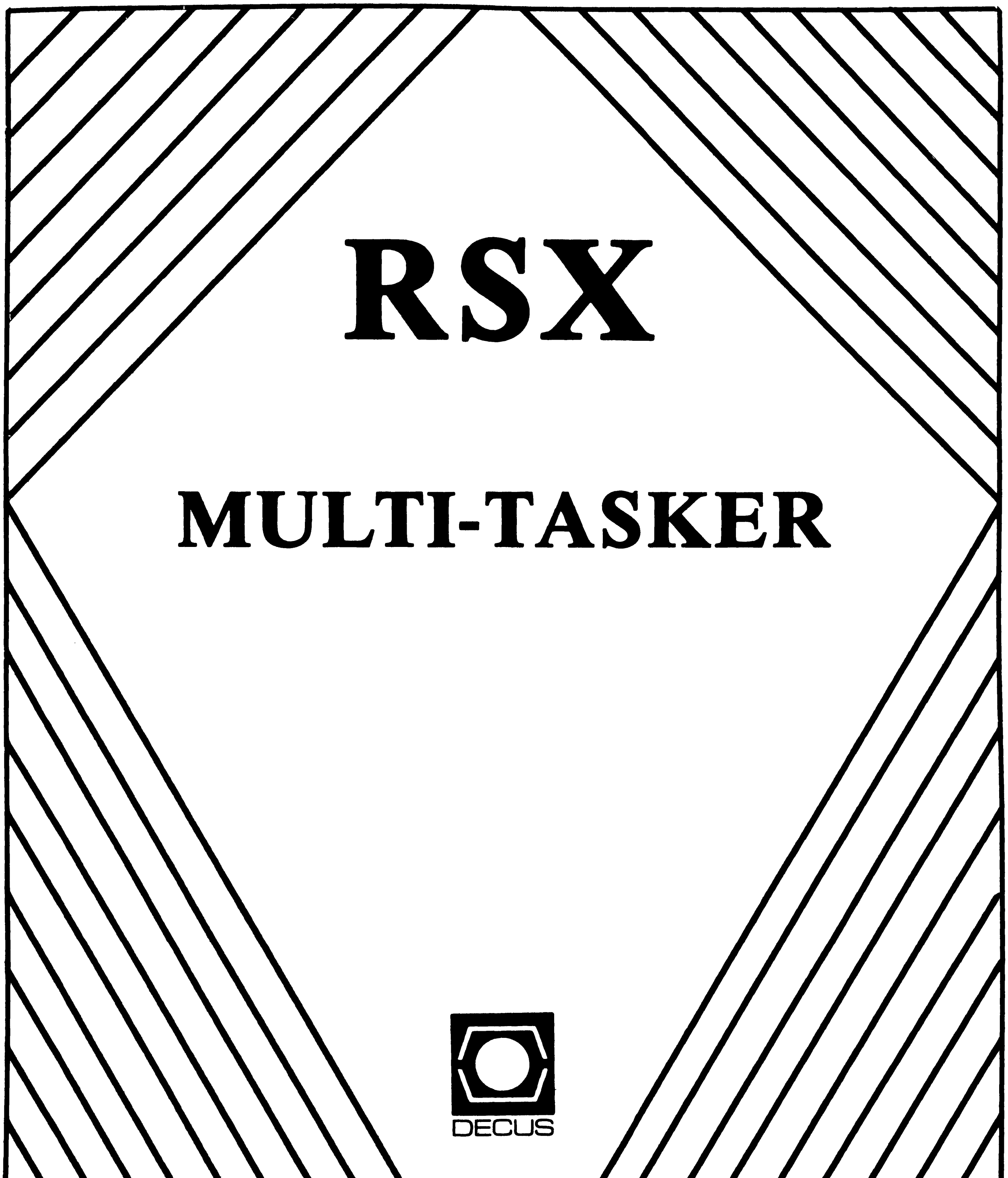
9.0 These operating procedures shall be submitted to the current SIG steering committee for approval and, upon approval by a majority, shall become effective.

9.1 Upon approval of these procedures, the current SIG Chairman shall be considered to have started a two-year term of office.

9.2 Upon approval of these procedures, members of the current Product Planning Committee who have served more than one consecutive year shall be considered to have served one year on that committee.

9.3 Upon approval of these procedures, all current members of the steering committee shall be afforded the opportunity to remain in office, commencing a two-year term. Any offices vacated shall be filled in the manner described in this document.

9.4 This article shall cease to be a part of this document following implementation of the Policies and Procedures according to its provisions.



RSX

MULTI-TASKER

DECUS

RSX

Chairman
Dan Eisner
Perkin-Elmer Corp.
Garden Grove, CA

Vice-Chairperson
Elizabeth Bailey
Tennessee Valley Authority
Muscle Shoals, AL

Symposium Coordinator
Rick Sharpe
Toledo Edison
Toledo, OH

Pre-Symposium Seminar Coordinator
Hans Jung
Associated Press
New York, NY

Communications Committee Representative
Allen Bennett
Lear Siegler Rapistan
Grand Rapids, MI

Newsletter Editor
Dominic J. DiNollo
Loral Electronics
Yonkers, NY

Store Coordinator
Bob Freeborn
Savin Corporation
Binghamton, NY

Session Note Editor
Burt Janz
Northern Telecom Inc.
Concord, NH

Librarian
Glenn Everhart
Mt. Holly, NJ

Campground Coordinator
Jerry Ethington
Prolifix Inc.
Frankfort, KY

DEC Counterparts
Tim Martin
Digital Equipment Corporation
Maynard, MA

Dick Day
Digital Equipment Corporation
Nashua, NH

Bruce Webster
Digital Equipment Corporation
Nashua, NH

Working Group Coordinator
Ed Cetron
Center for Biomedical Design
Salt Lake City, UT

Working Group Chair
Evan Kudlajev
Philadelphia Electric Co.
Philadelphia, PA

RSX Group Chair Software Clinic Coord.
Roy S. Maull
U.S. Air Force
Offutt AFB, NE

Software Clinic Coordinator
Bruce Zielinski
RCA
Moorestown, NJ

Volunteer Coordinator
Gary Maxwell
U.S. Geological Survey
Menlo Park, CA

Multi-Processors Working Group Coordinator
Bruce Mitchell
Machine Intelligence & Indus. Magic
Hudson, WI

Networks Working Group Coordinator
Mark Podany
Case Western Reserve University
Cleveland, OH

SRD Working Group Coordinator
Bob Turkelson
Goddard Space Flight Center
Greenbelt, MD

Accounting & Performance Working Group Coord.
Denny Walthers
American McGaw
Irvine, CA

Menu Coordinator
Ed Cetron
Center for Biomedical Design
Salt Lake City, UT

Members-At-Large
Jim McGlinchey
Warrenton, PA

Jim Neeland
Hughes Research Labs.
Malibu, CA

Anthony E. Scandora, Jr.
Argonne National Laboratory
Argonne, IL

Ralph Stamerjohn
Creve Coeur, MO

Jim Hopp
Carleton Financial Comp.
South Bend, IN

Table of Contents

From the Editor 2
RSX and P/OS Q & A Session - Fall 1985 3
An RSX-11M/M-Plus Application Note 10
Full Duplex Driver Hooks for ACD Processing 13
RSX Device Driver: Hints and Kinks 20

From the Editor

This is the last issue of the Multitasker for which I will be Editor. Starting with the July issue, Bruce Mitchell will be taking the reigns as Editor.

For the last year or so, my job responsibilites have taken me further and further from RSX based systems. I have reached the point of being out of touch with RSX. Besides this, after thirty months it it time for a change.

I am confident Bruce will take the Multitasker in to future. He is actively using RSX to its fullest capacity and certainly can supply good insight into what makes RSX tick, and more importantly how to make RSX for you.

I would like to thank all the people who have contributed to the Multitasker. Without contributions, the Multitasker would cease to exist. My thanks to the present and past RSX leadership who have been very supportive. Finally, thanks to you, the readers, who make this job worthwhile.

How to Contribute to the Multitasker

The Multitasker publishes articles and notes on all topics dealing with or relating to RSX based systems. If you are doing something new or innovative with RSX we would like to hear from you.

New instructions for sending contributions will be supplied by the new Editor in the next issue.

RSX and P/OS Question and Answer Session
Fall 1985 U.S. DECUS Symposium, Anaheim (Disneyland)

Gary Maxwell
U.S. Geological Survey
Menlo Park, California

Brian McCarthy & Dale Donchin, Panelists
Digital Equipment Corporation
Nashua, New Hampshire

The traditional Monday evening Question and Answer session for RSX got off to an unusually late start: 9:30 PM. This, along with the relatively early "shuttle bus curfew" accounts for the relatively low number of questions asked.

Each person who asked a question completed a form which included the question asked and the answer received. These, plus a recording of the session, are the basis for the written transcript. The editor would like to thank all the participants for taking the time to complete the forms.

Q1 How about a SET /NOTERM command for terminal devices, which would remove the terminal driver overhead which is unnecessary for non-terminal devices? (Denny Walthers, Kendall McGaw)

A1 Come to the session on using TTDRV with non-terminal devices. [Basically, one must use the various SET commands and the SF.SMC function. More functionality will be available in the next version of RSX. Ed.]

Q2 Our system console is located in a building remote from our offices. How can we get a copy of console messages stored into a file for remote access? (Mark Northrup, Miller Brewing)

A3 Use console logging, which must be generated into the system by Sysgen. Use a SET /COLOG/LOGFILE command in the system startup file. To look at the current logfile, issue another SET /COLOG/LOGFILE command to close the current file and open another.

Q3 EMT 374 is now being used in Micro/RSX. Are there any other EMT values planned for use by DEC? Can a set be reserved for users? (Jim McGlinchey, Business Info Technologies)

A3 RSX reserves EMT 370 through 377 for internal use. EMT 374 and 375 are used for recoverable/non-recoverable executive errors (bugchecks). EMT 376 and 377 are used for directives and stack switching. All others are available.

Q4 Will there eventually be support in the terminal driver for the full functionality of the DHU11, such as RXTIMER, Auto XON/XOFF, programmed I/O on short output, etc.? (Tivis Mobberley, MDB Systems)

A4 For full functionality, the user will have to modify the driver. We may make the number of data and stop bits user selectable (low priority).

Q5 Are there plans to have a backup utility that will take advantage of the streaming capability of the TU80 tape? (Helen Myers, Dow Chemical)

A5 BRU cannot take advantage of streaming because of the disk seek overhead required to read file headers. The TU80 should stream while data blocks are being transfered. Part of the problem is the thresholds in the TU80 to decide when to stream. The hueristics are unchangeable. As a further note, the TK50 does stream with BRU.

Q6 Using a spawned "RUN file" command, parent and offspring tasks sometimes lose synchronization, even though they are synchronized correctly on event flags. Any help? (Dale Lutes, Cessna Aircraft)

A6 Use the /EST=YES switch when you spawn the "RUN file" command, so the parent truly becomes connected to the offspring (instead of INS).

Q7 Does pre-genned M-Plus support user-written drivers? What is the upper level on interrupt vectors? How big is the driver partition? (Wayne Fisher, Horiba Instruments)

A7 SYSPARAM.DAT has an option for loading user-written drivers during system startup; the interrupt vector limit is 500; drivers can be loaded into GEN with the /HIGH switch so that the driver partition does not have to be changed.

Q8 I want to copy a bootable system image from an RL02 to an RC25. My MXV11 board with MXV11-B2 boot ROMs does not recognize the RC25. Is there an easy solution? (Dick Humphrey, Los Alamos National Lab)

A8 Buy a new set of MXV11-B2 ROMs - they should recognize the RC25. How to move the bootable system over is documented in the System Generation manual.

Q9 Will NFT ever duplicate PIP's functionality (in particular, support for /CD and /FO), and automatic translation of local defaults to remote defaults? (Tom Coradetti, Lawrence Livermore National Lab)

A9 PIP /FO support is upcoming in a later release of DECnet. No comment on /CD support. Logical name support in M-Plus V3.0 Update B will partially provide remote default translations.

Q10 Can we get standardization on the error code format displayed by utility tasks; e.g., all octal or all decimal? (Bruce Mitchell, MIIM)

A10 Noted. Efforts have been made in this area; required changes in the documentation are extensive. We have considered performing error translations in the Executive, but lack of user demand and required memory overhead are the current drawbacks.

Q11 What does CATCH do in the P/OS Toolkit? How can I get my own catchall to work? (Jim Bostwick, Cargill Inc.)
A11 ...CA2 is MCR. Limited user catchall support is in P/OS V3. A general catchall in P/OS is very difficult.

Q12 Please give us the .BLD and .OLB files necessary to build BRU, FMT, and BAD without disk overlays and without mounted volume support so we can put them in our own RSX-11S systems. (Tony Scandora, Argonne National Lab)
A12 It would take a couple hundred more disk blocks on the distribution. Since your request is for customizing the utilities, we suggest that you purchase a source licence.

Q13 We keep running out of pool on an M-Plus system running on an 11/23. What strategies are available to us? (Howard Roman, Systems Control, Inc.)
A13 You'll have to cut some system options out. Don't use XDT unless you really need it (or try the loadable XDT). Basically, you need to choose your Sysgen and Netgen options carefully.

Q14 Can Virtual Terminal ASTs, Exit Status Block ASTs, and ASTs from other places (QIO, Mark Time) interfere with each other? (Hans Jung, Associated Press)
A14 No. All user ASTs use the same delivery queue, and are guaranteed to be serialized.

Q15 At a previous session, I found out that P/OS developers have written a "screen mode ZAP." Will we get this in RSX? (Kreigh Tomaszewski, Amway Corp.)
A15 It was a hack program we wrote - not intended to be a generally useful tool. It won't see the light of day.

Q16 I have a Brand-X tape drive with a Brand-Y controller that emulates a TM11. It worked correctly under 11M, but under M-Plus it sometimes missed BOT and sometimes skipped tape marks. My workaround was to put a delay in the MTDRV initiator. Is this a problem? (Al Tyrrill, Digital Consulting)
A16 We think it's a known bug, and it's probably fixed in M-Plus V3.0.

Q17 Does M-Plus V3.0 support loadable crash dump drivers? If not, do you have plans to add it? (Helen Myers, Dow Chemical)
A17 It can be done by taking the crash dump support off of the pre-generated M-Plus kit and using the hooks in the Executive for loadable crash dump drivers. See us offline for details.

Q18 If a parent task exits before an offspring task, an Offspring Control Block becomes orphaned. Can you add support to clean up orphan OCBs? (Kreigh Tomaszewski, Amway Corp.)
A18 The behaviour is deliberate, so that offspring tasks can pass the OCB to their own offspring. An example is TKB chaining to CRF.

Q19 What is the formula to make the system image file as small as possible so it still works? I'm trying to put a system image on an RX50. (Phil Tripp, General Electric Co.)
A19 After SAVing a version of the system, go into VMR and do a PAR command. Determine the highest address used by SAV (add the base address to the length). Convert this number to decimal, giving the minimum system size in bytes. Divide this number by 512. (the number of bytes in a disk block), and add two blocks for the task image label blocks. This result should work as the size to use for the system image file.

Q20 Will an 11M system generated for an 11/73 run on an 11/83 (in the general case)? (Jesse Ursery, EG&G ORTEC)
A20 Yes. A problem that could be encountered is the bug with booting an 11/83 from an RQDX1. This is fixed in the next release.

Q21 On an 11/70 or 11/44 running 11M V4.1C or V4.1E, during the Sysgen phase of building the system help files, LBR hangs. If you hit RETURN, LBR aborts with a BPT error, followed by a checkpoint error message for MCR. After this, the system still works. If PMT is removed, this scenario doesn't happen. What's wrong? (Joseph M. O'Neil, Jr., Computer Science and Applications, Inc.)
A21 We don't know what could be happening. Please submit an SPR.

Q22 When using the XM: driver under 11M V3.2, an IO.KIL on one side while the other side was sending a 1-byte buffer caused both sides to hang for an indeterminant time. Also, could we see more rigorous DMC/DMR diagnostics, such as conversational diagnostics which allow testing of large (4K) block transfers? (Tom Becker, Informatics)
A22 Please submit an SPR for your problem. No comment on improving the diagnostics.

Q23 BROadcast crashes the system with an odd address trap when the system has two DHU11s with consecutive addresses. The crash occurs in the YHV port driver. If one of the DHU11s is removed, the problem does not occur. (Tivis Mobberley, MDB Systems)
A23 Please bring your crash dumps to the RSX campground so we can look at it.

Q24 Can we please have a tool which stores terminal session dialog in a log file? This would benefit every user, make problems easier to solve, and would make SPRs easier to generate. (Thomas Coradetti, Lawrence Livermore National Lab)

A24 Noted. A consensus felt that the VM: driver can support this function; also, there was an unsupported task (LOG) provided on M-Plus V1.0 kits which used the VT: driver. A straightforward solution does not exist.

Q25 (Answered in Q11)

Q26 When will there be a print spooling utility in P/OS with reasonable functionality? (Jim Bostwick, Cargill Inc.)
A26 Look at the V3 distributed print server.

Q27 It has been announced that the Pro Host Tool Kit will go away soon. Could this be replaced by RSX utilities? Can Macro programs work with the RSX macro library? Are P/OS compilers the same as the RSX versions? Could special P/OS symbols go into RSX TKB to explicitly use the [1,5] libraries? Do the RSX Forms Editor and Forms Utility work on P/OS forms? Basically, can you fold P/OS support into RSX? (Tony Scandora, Argonne National Lab)
A27 It would probably all work. We developers will discuss the situation among ourselves.

Q28 Is there anything that forces the use of RMS in DECnet under M-Plus? (Wayne Fisher, Horiba Instruments)
A28 If Netgen is done correctly, it will not call for RMS. Check [137,10]DECPRM.CMD for a flag symbol which would cause RMS use by default.

Q29 In 11M, an indirect command file of the form:

```
.Xqt Install Task/Run=Rem/Cmd=....
.Wait Ins
```

would work to start a task and continue the command file. Under M-Plus, this doesn't work. Leaving out the ".Wait" causes "MCR - Task active" messages; leaving the ".Wait" in causes Indirect to hang. How should I write the command file? (Hans Jung, Associated Press)
A29 This is a known problem. Use a workaround such as ".Delay 3S" for now. It may be fixed, but it is not a high priority item.

Q30 Are there any plans for 11M, M-Plus, or Micro/RSX distribution for the Professional series hardware? (Jesse Ursery, EG&G ORTEC)
A30 No. (Technically, it can be done.)

Q31 We would like TU78 support under M-Plus. We have coerced field service into hooking one up, dual-ported with a VAX. Can you do it? (Phil Tripp, General Electric Co.)
A31 We would need more user demand before providing support. PDP-11 TU78 diagnostics would have to be written: an expensive proposition.

Q32 When I throw away my DZ11s, DL11s, etc., and connect my terminals, bar code readers, and other foreign processors to a terminal server, what terminal driver I/O functions will continue to work? (Rick Van Eman, Ford Kentucky Truck)
A32 It should all continue to work.

Q33 I generated an M-Plus system for an 11/73 with two RLV12 controllers. Only one of the controllers is actually plugged in. CON hangs when I try to bring all the devices online. What's happening? (Tim Gabryel, Finnigan Corp.)
A33 You've stumbled onto some opaque documentation. The RLV12 has five registers (in 22-bit mode), not four, so if you configure the CSR for the second RLV12 as the documentation implies, the registers overlap. CON then "sees" a CSR where it thinks the second RLV12 is located, and it hangs trying to bring the non-existent controller and units online. Bump up the CSR for the second RLV12.

Q34 Please support the use of FLX on the Professional, including foreign mounts and dismounts. (Robert Gezelter)
A34 The device types "DW:" and "DZ:" would need to be added to FLX. This can be done by patching the device table in FLX.

Q35 Would you consider support for cache bypass in PLAS directives (bit 15 of the PDR), and/or save/restore of the Cache Control Register across context switches for the PDP-11/34 (which doesn't implement bit 15 of the PDR)? (David Berg, Fermilab)
A35 No for the CCR save/restore. Support for bit 15 of the PDR is enabled as part of the multi-processor conditionals. It would take some work to untangle it. Submit a menu item request.

Q36 Please support an option to define a different default library UIC for TKB and Macro. (Robert Gezelter)
A36 It could be done. Submit as a menu item request.

Q37 Please give us remote print queues over DECnet. (Applause from audience) (Tony Scandora, Argonne National Lab)
A37 Noted.

Q38 Can you support making CONSOLE.LOG a fixed record length RMS indexed file? The logger could unlock the current record when nothing is queued. Monitoring tasks could search the file on alternate keys. The logger would always use one file. A utility would be used to purge the file based on date. (Rick Van Eman, Ford RTP)
A38 No response.

Q39 I came across an undocumented PLAS Window Status bit mask, WS.EDS. Executive sources call this "Effective Data Space." When used, it causes the mapping to occur in I-space for non-I/D tasks, and in D-space for I/D tasks. Is this a permanent feature? I also noticed that for I/D tasks, when you map to I-space, and the corresponding D-space APR is unused, the Executive maps the D-space along with the I-space. Will I run into trouble when I explicitly remap that D-space APR? (Gary Maxwell, USGS)

A39 WS.EDS should be documented as a permanent feature. Mapping the D-space along with I-space is done to keep old programs working. You can remap the D-space APR after this happens.

Q40 Can you support building the Batch Processor against RMSRES? (Robert Gezelter)

A40 No. RMSRES is considered an optional item on M-Plus. [This question is followed up on Q44. Ed.]

Q41 Please give us a cluster overlay runtime package that does not have a default library: it stays mapped to whatever it's mapped to until it has to remap. This would help with a language OTS and FMS in a cluster with FCS or RMS in Supervisor mode. A lot of mapping is currently wasted in such a task. (Tony Scandora, Argonne National Lab)

A41 Noted.

Q42 Will there be an M-Plus print symbiont to use LAT printer servers? (Arnold De Larisch, Florida Atlantic University)

A42 We have no such plans as of yet.

Q43 Could we get more support to manipulate the error log file, LOG.ERR, such as optimized handling of a very large log file and improved performance from RPT? (Joe O'Neil, CSA, Inc.)

A43 We suggest that you create a new LOG.ERR file each month (or whenever it gets big). Also, consider using Batch for generating reports. Remember that RPT scans the entire log file when date selection is used. RPT is much faster if you give it an explicit packet range.

Q44 Please give us at least the option to build the Batch Processor against RMSRES to reduce the ineffeciencies of using overlayed RMS. (Robert Gezelter)

A44 Please submit as a menu item.

An RSX-11M/M-Plus Application Note

Richard J. Garrick
Systems Engineer
Tailor-Made Systems, Inc.

In many applications it is desirable to have access to MCR and DCL commands while a program is in execution. The RSX executive provides the SPAN directive which is an excellent tool for accessing a command line interpreter (CLI) from application programs. The SPAWN directive allows a running task, the parent task, to request the running of a second task, the offspring task, and can pass a command line to the offspring task. The SPAWN directive can be called from FORTRAN, BASIC-PLUS-2, and MACRO-11.

The SPAWN directive is a powerful tool and can take 12 parameters to clarify a particular purpose. To execute system level commands it is necessary to specify only 5 parameters with the syntax shown in figure 1. The name of the offspring task is the first parameter of interest. The task name can have a length of 6 characters and must be expressed in RAD50 format. The RAD50 format is a limited character set, internal to DEC, designed to fit 3 characters into a word versus 2 characters per word for standard ASCII. MCR's taskname is MCR ... and DCL's task name is ... DCL. DCL has a prototype task name, meaning that the system creates a separate copy of the CLI each time a command is entered from a terminal set to that CLI. In a conventional task CLI, like MCR, all commands from all terminals associated with the CLI are queued to a single copy of the task.

The next parameter specified is an event flag to be cleared when the SPAWN directive is issued and set when the offspring task exits. While the event flag specification is optional, it is a powerful synchronization tool. The offspring task does not act like a subroutine but becomes a full-fledged task competing for system resources like the parent task. In the given example, see figure 2, a request to spawn a directory listing command is followed by a prompt to input a file to be deleted. Without waiting for the directory listing command to be executed, the prompt for the file to be deleted would appear and the program would wait for the user's input before the directory listing would appear. Upon supplying input, the directory listing would appear. To solve this problem, another executive directive is needed. The Wait For Single Event Flag directive is issued to block the parent task and to wait for the offspring task to exit to set the specified event flag (see fig. 1 for syntax). As a side note, each task has its own unique set of event flags numbered 1 through 32. Try experimenting with the example by commenting out the Wait

For Single Event Flag directive.

The next two parameters are the command line to be queued for the offspring task and the length of the command line respectively. The command line can be any valid command for the spawned CLI. The length of the command line is limited to 79 characters on RSX-11M and 255 characters on RSX-11M-PLUS.

The last parameter specified receives the directive status word. The directive status word returns a positive value for successful directive acceptance and specific negative error codes for directive failures. Directive failure can be caused by such things as an invalid event flag number or privilege violation.

While a very useful application has been presented, only a fraction of the SPAWN directive's capabilities have been addressed. The reader is encouraged to read Section 5.3.71 of the RSX-11M/M-PLUS Executive Reference Manual for the SPAWN directive description and Section 2.2 of the same manual for a discussion of event flag synchronization.

```
CALL SPAWN BY REF (TASKNAME,,,EVENTFLAG,,,,COMMAND,LENGTH,,,STATUS)

   TASK_NAME - NAME OF OFFSPRING TASK IN RAD50 FORMAT
               (TASK MUST BE INSTALLED)

   EVENT_FLAG - EVENT FLAG NUMBER USED FOR SYNCHRONIZATION

   COMMAND - STRING CONTAINING COMMAND SENT TO OFFSPRING TASK

   LENGTH - LENGTH OF COMMAND

   STATUS - STATUS OF DIRECTIVE (RETURNS ERROR CODE IF DIRECTIVE FAILS)

CALL WAITER (EVENT_FLAG,STATUS)

   EVENT_FLAG - SPECIFIES WHAT EVENT FLAG TO WAIT FOR

   STATUS - STATUS OF DIRECTIVE
```

DIRECTIVE SYNTAX FOR BASIC-PLUS-2
FIGURE 1

```
10      DECLARE STRING COMMAND1,COMMAND2, FILE_NAME
        DECLARE INTEGER TASK_NAME(1%),LENGTH1,LENGTH2,DSW
     !
     ! SPECIFY THE MCR TASK NAME IN RAD50 FORMAT
     !
20      TASK_NAME(0%)=20938%           !RAD50 FORMAT  20938="MCR"
        TASK_NAME(1%)=-19588%          !RAD50 FORMAT -19588="..."
     !
```

```
     ! ISSUE DIRECTORY LISTING COMMAND FIRST
     !
30      COMMAND1="PIP /LI"
        LENGTH1=LEN(COMMAND1)
        CALL SPAWN BY RED (TASK_NAME(),,,1%,,,COMMAND1,LENGTH1,,,DSW)
        IF (DSW < 0) THEN       !CHECK FOR DIRECTIVE ERROR
                PRINT "ERROR IN FIRST SPAWN DIRECTIVE"
                STOP
        END IF
     !
     ! PROMPT FOR AND ACCEPT FILENAME TO BE DELETED
     !
50      INPUT LINE "ENTER COMPLETE FILE SPECIFICATION TO BE
        DELETED";FILE_NAME
     !
     ! ISSUE COMMAND TO DELETE SPECIFIED FILE
     !
60      COMMAND2="PIP "+FILE_NAME+"/DE"
        LENGTH2=LEN(COMMAND2)
        CALL SPAWN BY REF (TASK_NAME(),,,1%,,,,COMMAND2,LENGTH2,,,DSW)
        IF (DSW < 0) THEN       !CHECK FOR DIRECTIVE ERROR
                PRINT "ERROR IN SECOND SPAWN DIRECTIVE"
                STOP
        END IF
        CALL WAITFR(1%,DSW)
        IF (DSW < 0) THEN       !CHECK FOR DIRECTIVE ERROR
                PRINT "ERROR IN SECOND WAIT DIRECTIVE"
                STOP
        END IF
     !
     ! PRINT MESSAGE ON TERMINAL THAT THE DIRECTORY LISTING COMMAND WILL BE
     ! EXECUTED AGAIN AS A CHECK AND GIVE THE USER TIME TO READ IT
     !
70      PRINT LF;"THE DIRECTORY LISTING WILL BE REPEATED TO VERIFY
        DELETION"LF SLEEP 2%
     !
     ! REISSUE DIRECTORY LISTING COMMAND
     !
80      CALL SPAWN BY REF (TASK_NAME(),,,1%,,,,COMMAND1,LENGTH1,,,DSW)
        IF (DSW < 0) THEN       !CHECK FOR DIRECTIVE ERROR
                PRINT "ERROR IN THIRD SPAWN DIRECTIVE"
                STOP
        END IF
        CALL WAITFR(1%,DSW)
        IF (DSW < 0) THEN       !CHECK FOR DIRECTIVE ERROR
                PRINT "ERROR IN THIRD WAIT DIRECTIVE"
                STOP
        END IF
        END
```

PROGRAMMING EXAMPLE
FIGURE 2

Full-Duplex Terminal Driver Hooks For ACD Processing

Fall DECUS 1985
Dale R. Donchin

NOTE WELL:

This document is supplied as informational material only. It is not guaranteed to be accurate and is always subject to change, although for the most part it seems to be pretty good. The ACD interface and any user-written ACDs are not supported by Digital. The source code of the ACD loader task is supplied on the RSX-11M-PLUS distribution kit as [12,10]ACD.MAC.

The list below details the hook points within the Full Duplex Terminal Driver. For each hook point there is a unique entry in the ACD dispatch table. The bits in the UCB offset U.AFLG describe the action which should occur when the ACD returns control to the driver. The default action is set by the driver before calling the ACD. In this way the ACD can execute just a RETURN instruction if it does not want to alter the normal processing which would occur at that particular point in the driver had the ACD not been there. Alternatively, the ACD can manipulate those bits to obtain the desired action. Each hook point described below lists the bits which pertain to it, including the default setting.

NOTE:

In all cases, the ACD is passed a pointer to offset U.TSTA in the UCB via R5. This is how the ACD knows for which terminal the processing is being done.

All registers except for R0 must be preserved. The APR 6 mapping must also be preserved.

1. I/O request acceptance

Entered when a task has issued an I/O request. The context of the task is still available for validation of buffer parameters. At this point the ACD can change the I/O function, such as changing a read function to a read with prompt function. In this case, the prompt buffer address and length are passed in A.SMAP, A.SBUF and A.SLEN. This entry point is called only if the appropriate bit in A.MAS or AMAS+2 is set for the function code specified by the request.

INPUT:
R1 - Function code * 2
R3 - I/O packet

OUTPUT:
UA.ALL - Set if this request should be accepted. If clear, refuse the request returning the error code passed in the first word of A.IOS

DEFAULT:
UA.ALL

2. I/O request dequeue

Entered when a previously received I/O request is extracted from the queue and is about to be processed. The context of the task which issued the request is not necessarily available. This entry point is called only if the appropriate bit in A.MAS or A.MAS+2 is set for the function code specified in the request.

INPUT:
R1 - Function code * 2
R3 - I/O packet

OUTPUT:
UA.ALL - Set if this request should be accepted. If clear, refuse the request returning the error code passed in the first word of A.IOS

DEFAULT:
UA.ALL

3. Powerfail

Entered when power is restored after a powerfail has occurred.

4. Input request completion

Entered when an input request terminates (successfully or not).

INPUT:
R3 - I/O packet
A.IOS - Reason for completion

OUTPUT:
UA.TRA - Transfer characters from the ACD buffer as described by A.SMAP, A.SBUF and A.SLEN to the task's read buffer. If not set, characters are transferred from the terminal driver's buffers.
A.IOS - Reason for completion (can be changed by the ACD)

5. Output request completion

Entered when an output request terminates (successfully or not).

INPUT:
R3 - I/O packet
A.IOS - Reason for completion

OUTPUT:
A.IOS - Reason for completion (may be changed by the ACD)

6. Connection

Entered when a terminal is set to this ACD. Any initialization and memory allocation necessary for this link can be done at this point.

INPUT:
R0 - ACB address
A.IOS - Low byte of first word is zero

OUTPUT:
A.IOS - Low byte of first word may be set nonzero to refuse the connect request. The nonzero value is returned to the task requested the connection as the error status.

7. Disconnection

Entered when a terminal is unlinked from this ACD. Any clean-up of the link should be done at or by this point.

INPUT:
R0 - ACB address

8. Input character reception

Entered when an input character is received.

INPUT:
R2 - Character received

OUTPUT:
R2 - Character received (may be changed by the ACD)
UA.ACC - Set if the character is to be accepted. If clear, the character is thrown away without any user notification
UA.PRO - Do the standard processing of this character, if not, treat the character as pass-all (no interpretation)
UA.ECH - Set if this character will be echoed when processed
UA.TYP - Set to force the character into the typeahead buffer

DEFAULT:
UA.ACC UA.PRO UA.ECH

9. Input character processing

Entered when an input character is processed.

INPUT:
R2 - Character to process

OUTPUT:
R2 - Character to process (may be changed by the ACD)
UA.ECH - Set if an echo should occur
UA.SPE - Set if a special (ACD supplied) echo is to be output. In this case the echo buffer and length are passed in A.IMAP, A.IBUF and A.ILEN
UA.PUT - Set if the character is to be put into the input buffer
UA.COM - Set if the input request should be completed. In this case the reason for completion must be specifed in A.IOS. The ACD will be entered again at the input completion entry point
UA.CAL - Set if the ACD needs to be called back (at entry point #10) after the echo completes to deallocate its buffers and/or begin another echo string

DEFAULT:
UA.ECH UA.PUT

10. Interrupt level callback

Entered when the ACD has indicated by setting the UA.CAL bit that it needs to be called back after input character echo has completed, perhaps to initiate more echo from another buffer. This entry point is called after the previous echo completes.

INPUT:
R2 - Character being echoed

OUTPUT:
UA.CAL - Set if the ACD needs to be called back again at this entry point after the echo described by A.IMAP, A.IBUF and A.ILEN has been initiated and completed

11. Output translation

Entered during output I/O request processing just before the characters from the task's buffer are copied to terminal driver buffers. This gives the ACD the opportunity to determine what characters will be output.

INPUT:
R3 - I/O packet address

OUTPUT:
UA.TRN - Set if the ACD wants to translate the output stream. If this bit is not set, then the task's buffer will be output. If UA.TRN is set, the ACD is expected to set up A.SMAP and A.SLEN to point to a buffer of the characters to be output instead. After the characters are copied from the ACD specified buffer to the driver buffers, the ACD is called back at this same entry point. The ACD may set UA.TRN to transfer another buffer (possibly part or all of the user task buffer by setting up A.SMAP to point to it) or just return to begin the output of the driver's buffers.

12. System state callback

Entered when the ACD has indicated by setting the UA.TRA bit that it needs to be called back after the previous buffer has been transferred to the task during input I/O completion. This entry point is called after the previous transfer completes. The ACD can use this entry point to know when to deallocate its buffer and/or to transfer more characters from another buffer.

OUTPUT:
UA.CAL - Set if the ACD needs to be called back at this entry point after the ACD buffer described by A.SMAP, A.SBUF and A.SLEN has been appended to the task buffer

Important Note About A.RES

The symbol A.RES defines the start of ACD specific offsets. These offsets can be used to contain any information the ACD needs to keep on a per-line basis. It is assumed that the prototype ACB has been created with A.FLEN at least equal to the size of a full ACD (32 bytes). Any additional offsets needed should be added to A.FLEN. For example, if the ACD needs 2 additional word offsets, the prototype ACB should be created with A.FLEN = 32+4 = 36 octal bytes.

Restrictions on the ACD

1. If an ACD wishes to transfer input characters from its own buffers to a task's read buffer, it must not set the UA.PUT flag during input character processing. Instead it must buffer the characters itself, set the UA.COM flag when the read is complete, and set the UA.TRA flag at the input completion entry point.

2. All buffer transfers must be of a non-zero length, not exceeding 4K words.

3. The ACD is responsible for not transferring more characters to the task's input buffer than the task has requested.

4. At the moment, the ACD cannot use the UA.TRA mechanism to transfer characters which are part of an unsolicited input request.

Format of the Ancillary Control Block (ACB)

Prototype (linked off $ACDHD listhead)

```
         +---------------------------------+
A.REL    |       Relocation bias           |                 0
         +---------------------------------+
A.DIS    |    Dispatch table address       |                 2
         +---------------------------------+
A.MAS    |       Function maskword         |                 4
         +                                 +
A.MAS+2  |          doubleword             |                 6
         +---------------------------------+
A.NUM    | ID number      |  ACB length    |  A.FLEN         10/11
         +---------------------------------+
A.LIN    |          Link word              |                 12
         +---------------------------------+
A.ACC    | Access count  |  Status byte    |  A.STA          14/15
         +---------------------------------+
```

Full ACB (linked off U.ACB)

```
         +---------------------------------+
A.REL    |       Relocation bias           |                 0
         +---------------------------------+
A.DIS    |    Dispatch table address       |                 2
         +---------------------------------+
A.MAS    |       Function maskword         |                 4
         +                                 +
A.MAS+2  |          doubleword             |                 6
         +---------------------------------+
A.NUM    | ID number      |  ACB length    |  A.FLEN         10/11
         +---------------------------------+
A.IMAP   |  Interrupt buffer APR bias      |                 12
         +---------------------------------+
A.IBUF   |  Interrupt buffer address       |                 14
         +---------------------------------+
A.ILEN   |  Interrupt buffer length        |                 16
         +---------------------------------+
A.SMAP   | System state buffer APR bias    |                 20
         +---------------------------------+
A.SBUF   | System state buffer address     |                 22
         +---------------------------------+
A.SLEN   | System state buffer length      |                 24
         +---------------------------------+
A.IOS    |          I/O status             |                 26
         |                                 |                 30
         +---------------------------------+
A.RES    | Reserved for use by the ACD     |                 32
         |                                 |
         V                                 V
```

ACD offset definitions extracted from PKTDF$

```
;+
; ANCILLARY CONTROL BLOCK (ACB) DEFINITIONS
;-
.=0
A.REL:'L' .BLKW 1               ;ACD RELOCATION BIAS
A.DIS:'L' .BLKW 1               ;ACD DISPATCH TABLE POINTER
A.MAS:'L' .BLKW 2               ;ACD FUNCTION MASK WORDS
A.NUM:'L' .BLKB 1               ;ACD IDENTIFICATION NUMBER
A.FLEN:'L'.BLKB 1               ;LENGTH IN BYTES OF FULL ACB
A.LIN:'L' .BLKW 1               ;ACD LINK WORD
A.ACC:'L' .BLKB 1               ;ACD ACCESS COUNT
A.STA:'L' .BLKB 1               ;ACD STATUS BYTE
A.PLEN='B'.                     ;LENGTH IN BYTES OF PROTOTYPE ACB
;
.=A.LIN                         ;FULL ACB OVERLAPS PROTOTYPE ACB
A.IMAP:'L' .BLKW 1              ;ACD INTERRUPT BUFFER RELOCATION BIAS
A.IBUF:'L' .BLKW 1              ;ACD INTERRUPT BUFFER ADDRESS
A.ILEN:'L' .BLKW 1              ;ACD INTERRUPT BUFFER LENGTH
A.SMAP:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER RELOCATION BIAS
A.SBUF:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER ADDRESS
A.SLEN:'L' .BLKW 1              ;ACD SYSTEM STATE BUFFER LENGTH
A.IOS:'L' .BLKW 2               ;ACD I/O STATUS
A.RES='B'.                      ;START OF ACB RESERVED FOR USE BY THE ACD
;
; DEFINE THE FLAG VALUES IN THE OFFSET U.AFLG
;
UA.ACC='B'1                     ;ACCEPT THIS CHARACTER
UA.PRO='B'2                     ;PROCESS THIS CHARACTER
UA.ECH='B'4                     ;ECHO THIS CHARACTER
UA.TYP='B'10                    ;FORCE THIS CHARACTER INTO TYPEAHEAD
UA.SPE='B'20                    ;THIS CHARACTER HAS A SPECIAL ECHO
UA.PUT='B'40                    ;PUT THIS CHARACTER IN THE INPUT BUFFER
UA.CAL='B'100                   ;CALL THE ACD BACK AFTER THE TRANSFER
UA.COM='B'200                   ;COMPLETE THE INPUT REQUEST
;
UA.ALL='B'400                   ;ALLOW PROCESSING OF THIS I/O REQUEST
UA.TRN='B'1000                  ;TRANSLATE CHARACTERS FROM OUTPUT QIO
UA.TRA='B'2000                  ;TRANSFER CHARACTERS WHEN I/O COMPLETES
;
; DEFINE THE ACD ENTRY POINTS (OFFSETS INTO THE DISPATCH TABLE)
;
.=0
A.ACCE:'L' .BLKW 1              ;I/O REQUEST ACCEPTANCE ENTRY POINT
A.DEQU:'L' .BLKW 1              ;I/O REQUEST DEQUEUE ENTRY POINT
A.POWE:'L' .BLKW 1              ;POWER FAILURE ENTRY POINT
A.INPU:'L' .BLKW 1              ;INPUT COMPLETION ENTRY POINT
A.OUTP:'L' .BLKW 1              ;OUTPUT COMPLETION ENTRY POINT
A.CONN:'L' .BLKW 1              ;CONNECTION ENTRY POINT
A.DISC:'L' .BLKW 1              ;DISCONNECTION ENTRY POINT
A.RECE:'L' .BLKW 1              ;INPUT CHARACTER RECEPTION ENTRY POINT
A.PROC:'L' .BLKW 1              ;INPUT CHARACTER PROCESSING ENTRY POINT
A.TRAN:'L' .BLKW 1              ;OUTPUT QIO CHARACTER TRANSLATION ENTRY POINT
A.CALL:'L' .BLKW 1              ;CALL ACD BACK AFTER TRANSFER ENTRY POINT
;
; DEFINE THE STATUS BITS IN A.STA OF THE PROTOTYPE ACB
;
AS.DLT='B'1                     ;ACD IS MARKED FOR DELETE
AS.DIS='B'2                     ;ACD IS DISABLED
```

Chris Isbell
Kenda Electronic Systems
Nutsey Lane
Southampton, Hampshire, England

1.0 INTRODUCTION

This article describes some techniques and tricks which were developed, borrowed or discovered in the course or writing a number of device drivers for RSX-11M and RSX-11M/PLUS. Some of the items are aimed at the programmer writing their first (hopefully simple) device driver. Other items are concerned with drivers for complex and often high speed hardware such as real time data acquisition devices or high speed communication links.

All of the examples are from actual device drivers. Some have been edited to remove code which is incidental to the point being demonstrated or to replace device specific macros with in-line example code. In all the examples the device name has been changed to XX:.

It is assumed that the driver to be debugged is a loadable driver on a mapped system. The driver will be mapped by kernel active page register five, and have a starting virtual address of 120000 octal. In systems using separate I and D space the driver is mapped by both the I and D space APRs. The examples given are from device drivers written for RSX-11M/PLUS, but they should still be applicable on RSX-11M systems, except where noted.

2.0 VIRTUAL AND PHYSICAL ADDRESSES

One of the things which makes life interesting on a mapped PDP-11 system is the large number of ways of addressing memory. In the case of a buffer passed to a device driver, the following addresses may be encountered.

2.1 User task virtual address.

These are usually mapped by user mode D space APRs. Drivers are not usually concerned with these addresses. They are only valid in the context of the user task which issued the request. This context can only be assumed before the I/O packet is queued. A driver is usually only called at this point if it needs to address check and relocate special user buffers.

2.2 Driver address double-word for non-DMA devices.

The first word contains a value to be placed in kernel D space APR 6. The second word contains a virtual address assuming the APR 6 mapping. The driver only needs to save and restore APR 6 in an interrupt service routine before becoming a system process. (Ie:- Between INTSV$ and calling $FORK.) If the user buffer is smaller than 4K words less 32 words, then the whole of the buffer is mapped by APR 6. If it is larger then the buffer must be re-mapped beyond this limit. This is done by subtracting a suitable number from the virtual address. This should be an exact multiple of 100 octal and be smaller than 4K words less 32 words. This value divided by 100 octal should be added to the APR 6 mapping.

```
;
; Copy special registers to user buffer
; (R4 points to device registers)
;
        mov     r1,-(sp)        ;;;Save register
        mov     @£kisar6,-(sp)  ;;;Save current mapping
        mov     u.buf(r5),@£kisar6 ;;;Map user buffer
        mov     u.buf+2(r5),r1  ;;;Address user buffer
        mov     xxsps(r4),(r1)+ ;;;Copy first value
        mov     xxdat(r4),(r1)+ ;;;Copy second value
        mov     (sp)+,@£kisar6  ;;;Restore mapping
;
; Update buffer virtual address for next time
;
        add     £size,u.buf+2(r5) ;;;Update virtual address
        cmp     u.buf+2(r5),£<160000-size> ;;;See if off APR
        blo     20$             ;;;Branch if O.K.
        sub     £<20000-100>,u.buf+2(r5) ;;;Knock off almost 4K words
        add     £177,u.buf(r5)  ;;;Update mapping to compensate
20$:    mov     (sp)+,r1        ;;;Restore registers
```

2.3 Physical memory address.

This value is only of concern to drivers for DMA devices. In the case of eighteen bit RSX-11M systems (maximum memory 124K words) and twenty two bit Q-bus systems, this address is the one passed to the device for the transfer. Normally the executive converts the virtual address in the double word at U.BUF(R5) into a physical address in the same locations.

2.4 Unibus address

In the case on Unibus systems with twenty two bit memory (11/24, 11/44 etc.) the twenty two bit physical address must be converted into an eighteen bit Unibus address, and that part of the Unibus address space must be mapped into the physical memory address space using Unibus mapping registers. This is handled by executive subroutines called by the driver. These routines assume that the size in bytes of the buffer is in U.CNT(R5). Drivers which alter the value in U.CNT(R5) should only do this after Unibus mapping registers have been allocated.

Devices which need to perform DMA operations into kernel D space addresses below 120000 (the driver UCB, or a buffer in pool for example) do not need to assign Unibus mapping registers. The executive assigns the lowest Unibus mapping registers so that they always map this part of physical memory. Drivers which wish to perform DMA operations into kernel I space or their own APR 5 address space are on their own.

On a quiet system, a device driver will usually get the lowest free Unibus mapping register. This will give a Unibus address in the low sixteen bits of the Unibus address space. (A value of 120000 octal is typical.) This can be a little disconcerting at first. If you are reduced to looking at a crash dump to assist in debugging a driver, it is helpful to know that the executive numbers Unibus mapping registers from zero while CDA numbers them from one.

2.5 Converting between addresses

The user mode virtual address of the data buffer is only valid in the context of the user task. The conversion of this address to a kernel APR 6 address double word is covered in the RSX driver documentation. This can only be done by drivers which perform special buffer validation before queuing the I/O packet. Note that user (and supervisor) mode APRs are not valid after an I/O packet has been dequeued.

Conversion of the APR 6 address double word to a physical address is performed transparently by the executive for DMA device drivers. Drivers which also need to access the user buffer under program control can use the routine $RELOP as shown in the example below. (On systems with Unibus mapping registers this operation should be performed before converting the physical address to a Unibus address.)

```
;
; Convert buffer physical address into APR6 mapping and offset
;
        clr     r0                      ;Offset into buffer
        call    $relop                  ;Form address doubleword
;
; (The user buffer is now mapped by APR 6 and pointed to by R1)
;
        mov     @£kisar6,u.bufv(r5)     ;Get APR6 bias
        mov     r1,u.bufv+2(r5)         ;Get virtual address
```

A physical address is usually converted to a Unibus address using the executive routines $STMAP to assign Unibus mapping registers and $MPUBM to actually map the Unibus address to physical memory.

```
;
; Set up Unibus map
;
        .if df m$$ext
        call    $stmap                  ;Get UNIBUS mapping registers
        call    $mpubm                  ;Set up UNIBUS map
        .endc   ;df m$$ext
;
; The byte count in U.CNT can now be converted to a word count. The
; buffer length in bytes is used by $STMAP to calculate the correct
; number of mapping registers to allocate
;
        clc                             ;Get ready to roll
        ror     u.cnt(r5)               ;Get word count
```

```
        bne     15$                     ;Branch if non-zero
        mov     £ie.spc&377,r0          ;Report invalid user buffer
        call    $ioalt                  ;Finish the operation
        jmp     xxini                   ;Try for more work
15$:
```

2.6 Physical memory layout

RSX-11M/PLUS systems with I and D space support and Unibus mapping registers have the bottom 8K bytes of physical memory mapped by both kernel I space and D space APR 0. This includes trap and interrupt vectors, the system stack and ICB pool. This is followed by kernel instruction space which is mapped by I space APRs 1 to 4 inclusive. On the next 32 word boundary above the kernel instruction space is kernel data space mapped by D space APRs 1 to 4 inclusive. This includes the executive static data area and primary pool. The lowest numbers Unibus mapping registers are statically set up so that Unibus addresses in the range 0 to 120000-1 map to the corresponding addresses in kernel D space. The values in key mapping registers for a typical system are given below.

Kernel I space

APR number	Value	Physical address	Virtual address
0	0	0	0
1	200	20000	20000
2	400	40000	40000
3	600	60000	60000
4	1000	100000	100000
7	177600	17760000	160000 (I/O page)

Kernel D space

APR number	Value	Physical address	Virtual address
0	0	0	0
1	740	74000	20000
2	1140	114000	40000
3	1340	134000	60000
4	1540	154000	100000
7	177600	17760000	160000 (I/O page)

Unibus map

Number	Value	Kernel D space address
0	0	0
1	74000	20000
2	114000	40000
3	134000	60000
4	154000	100000

3.0 BUFFER LENGTH

Unless the driver performs special buffer checks before the I/O packet is queued, the executive checks that the user buffer has a valid address and that it lies wholly within the user task's virtual address space. The device driver should generally check explicitly for a buffer length of zero and take appropriate action. A buffer length of zero sent to most DMA devices would result in a transfer of 64K bytes. This transfer could either corrupt memory outside the user task's address space or produce a non existent memory error on the device if it runs off the end of physical memory. Drivers for non DMA devices should also take care to avoid this trap.

On a mapped system the user buffer is only partially mapped if it is larger than 4K words less 32 words. The executive routines $GTBYT, $PTBYT, $GTWRD and $PTWRD all update the buffer address double word to remap the buffer if required. If a driver manipulates the user buffer directly it should either limit the buffer size to 4K words less 32 words or remap the buffer as described above.

When manipulating the buffer length it should be treated as an unsigned number. In particular beware of using the ASR instruction to convert it from a byte count to a word count. The instructions CLC followed by ROR are usually used.

4.0 INTERRUPT SERVICE ROUTINES

In a large number of drivers the interrupt service routine is a common entry point required to handle a number of different conditions. This often leads to obscure code containing a large number of conditional branches based on bit settings on one or a number of status variables.

An alternative method is the use of a simple interrupt dispatcher which jumps to an address in a word in the UCB. The overhead of this dispatcher is typically very small compared to the hardware interrupt latency time together with the register saving and mapping performed before the driver itself is entered. Using this method the interrupt code for a particular operation can follow the code which instigated the operation resulting in a much clearer layout of the source code. In the examples given the dispatch variable is called U.GOTO(R5). This can either be a device specific extension to the UCB or can be equated to an existing UCB offset, eg U.CW2(R5).

```
        bis     £<intebl!go>,u.buf(r5)  ;Set input bits
        mov     u.cnt(r5),rl            ;Get byte count
        neg     rl                      ;Negate count
        mov     rl,xxbc(r2)             ;Output to device
        mov     u.buf+2(r5),xxba(r2)    ;Output bus address
        mov     £trint,u.goto(r5)       ;Set up interrupt pointer
        movb    s.itm(r4),s.ctm(r4)     ;Set up timeout count
        mov     u.buf(r5),(r2)          ;Set it all going
        return                          ;Wait for some action
;
; Transfer function interrupt enters here
;
trint:
        call    $fork                   ;;;Create a system process
        mov     u.scb(r5),r4            ;Address SCB
        mov     s.csr(r4),r2            ;Address device CSR
```

The dispatcher code should handle extraneous interrupts gracefully. In the example below they are simply ignored.

```
.page
.sbttl  $XXINT  interrupt service routine
; WRITTEN BY:- Chris Isbell     02-NOV-85

; *****     DESCRIPTION          *****
; Entered as a result of DMA completion.
;
; *****   ENTRY CONDITIONS       *****
; Hardware interrupt.
;
; *****   EXIT CONDITIONS        *****
;  Dispatch to address in offset u.goto of the UCB
;
; *****  SUBROUTINES CALLED      *****
; $INTSV
;
;-

$xxint::
        intsv$  XX,PR5,x$$x                     ;;;Generate interrupt save code
        mov     u.goto(r5),-(sp)                ;;;Action routine address
        bne     10$                             ;;;Branch if OK
        mov     £20$,(sp)                       ;;;Set a dummy routine address
10$:    mov     £20$,u.goto(r5)                 ;;;Multiple interrupt block
20$:    return                                  ;;;Go to action routine
                                                ;;;return = jmp @(sp)+
```

5.0 MAINTAINING CONTEXT BETWEEN OPERATIONS

Some hardware requires a lot of setting up before it can be used. This setting up may only be required once for a large number of subsequent operations. A typical example of this would be a laboratory instrument. In this case it is desirable if the driver can recognise that a sequence of operations are related. The driver can then reject operations depending on the hardware being set up before the set up operation has been performed.

A convenient method for dealing with this situation is to require the user to attach the device before allowing any other operations. By setting the appropriate bit in the UCB, the driver can be called on attach and detach. These functions indicate to the driver that the hardware can be reset loosing its context.

In the following example the driver uses U.CW2(R5) to indicate the hardware context. A zero value means that there is no hardware context. This word is also set to zero on power fail and when the driver is placed on line.

```
        cmpb    i.fcn+l(rl),£io.att/256. ;See if attach function
        beq     10$                      ;Branch if so
        cmpb    i.fcn+l(rl),£io.det/256. ;See if detach function
        bne     20$                      ;Branch if not
;
; Process attach/detach
;
10$:    clr     u.cw2(r5)                ;Flag hardware not set up
        mov     £reset,@s.csr(r4)        ;Reset the hardware
        mov     £is.suc&377,r0           ;Return success code
        br      30$                      ;Take the common exit
;
; Process read/write
;
20$:    mov     £ie.dna&377,r0           ;Assume device not attached
        cmp     u.att(r5),i.tcb(rl)      ;See if it is ours
        beq     40$                      ;Branch if it is
30$:    call    $ioalt                   ;Reject the operation
        br      xxini                    ;Try for more work
40$:
```

If the hardware context will be lost on power failure the driver should always be called on power fail. It should reset its indication of the hardware context.

6.0 SETTING TASK EVENT FLAGS

In high speed data acquisition applications using double buffering performance can be increased it the driver is only called once for a large number of transfers into the same buffers. This avoids the considerable overhead of repeated QIO operations. This can be achieved by writing the driver so that it sets an event flag every time the user task needs to be notified that there is data to process. The driver then continues transferring data into another part of the data buffer. The driver can also force the executive to schedule the user task. In the example below the driver also increments the high byte of the first I/O status block word. This can be used by the user task to check for a data over run condition where data is being transferred by the driver faster that it can be processed.

```
;
; Flag buffer event in i/o status block
;
```

```
        mov     u.scb(r5),r4    ;Address SCB
        mov     s.pkt(r4),rl    ;Address i/o packet
        tst     i.iosb(rl)      ;See if i/o status block specified
        beq     50$             ;Branch if not
        mov     i.iosb+2(rl),@£kisar6 ;Map the i/o status block
        mov     i.iosb+4(rl),r0 ;Address of i/o status block
        incb    l(r0)           ;Increment the flag byte
50$:
;
; Set the user specified event flag
;
        tstb    u.efn(r5)       ;See if an event flag was specified
        beq     60$             ;Branch if not
        mov     r5,-(sp)        ;Save UCB address
        movb    u.efn(r5),r0    ;Event flag number
        mov     i.tcb(rl),r5    ;TCB address of task
        call    $setf           ;Set the event flag
        call    $setrt          ;Force this task to run next
        mov     (sp)+,r5        ;Restore UCB address
60$:
        return                  ;Wait for the next interrupt
```

The event flag number passed by the user should be validated. The minimum value is one and the maximum value depends on whether or not the executive includes support for group global event flags. (Note that there is a much higher overhead if group global event flags are used. They should be avoided in very high speed operations.) A value of zero can be used to indicate that no event flag is specified.

```
        .if df G$$GEF                   ;Max. EFN depends on group
                                        ; global event flag support
        maxefn  =       96.             ;Maximum event flag number
        .iff
        maxefn  =       64.             ;Maximum event flag number
        .endc   ;df G$$GEF
;
; Validate user specified event flag number
;
40$:    movb    (rl)+,r0                ;Get the event flag number
        cmp     r0,£maxefn              ;Validate it
        blos    50$                     ;Branch of valid
        mov     £ie.flg&377,r0          ;Error - invalid event flag
        br      1000$                   ;Clean up
50$:    mov     r0,u.efn(r5)            ;Save event flag number
```

7.0 AVOIDING RACE CONDITIONS CAUSED BY MULTIPLE INTERRUPTS

If a driver changes from interrupt state to system state by calling $FORK while the hardware is still running and interrupts are enabled a potential race condition exists. If another interrupt occurs before fork processing for the first interrupt is complete then the second interrupt must not call $FORK. This will corrupt the fork queue and crash the system.

It is possible for the code after the call to $FORK to determine if another interrupt has occurred and take appropriate action. In the example below the offset in the fork block which holds the saved program counter is cleared when the driver is entered in system state. The interrupt service routine regards a non-zero value in this word as an indication that a fork request is pending. If this is the case it clears the word in the fork block containg the saved value of R4. The code executed after fork processing has completed checks for a zero value in R4 indicating that multiple interrupts have occurred.

```
;
; Create a system process (fork) if there is not already a fork request
; outstanding for this device.
;
        mov     u.scb(r5),r4    ;;;Address SCB
        tst     s.frk+2(r4)     ;;;See if fork already pending
        beq     10$             ;;;Branch if not
        clr     s.frk+6(r4)     ;;;Clear fork block R4
        return                  ;;;Dismiss interrupt
10$:    call    $fork           ;;;Create a system process
        clr     (r3)            ;Reset fork PC
;
; Check for a failed fork request from the interrupt service routine
;
        tst     r4              ;Check for double fork
        bne     47$             ;Branch if all is in order
        mov     £ie.dao&377,r0  ;Error code
        call    $ioalt          ;Finish the operation
        jmp     rrini           ;Try for more work
47$:
```

8.0 GETTING INTO THE DRIVER USING XDT

A useful debugging aid is to place a NOP instruction in the driver code at a point just after an I/O packet has been dequeued.

```
xxini:
        gtpkt$  XX,x$$x,,,single        ;Get some work
;
; The following registers are returned by $GTPKT
;
; R5 -- UCB address
; R4 -- SCB address
; R3 -- Controller index (not used)
; R2 -- Physical unit number (not used)
; R1 -- I/O packet address
;
; The I/O packet parameters are as follows:-
;
; I.PRM+0 -- User buffer address (high order)
; I.PRM+2 -- User buffer address (low order)
; I.PRM+4 -- User buffer length (bytes)
;
        nop                             ;Debugging aid
        cmpb    i.fcn+1(r1),£io.att/256. ;See if attach function
        beq     10$                     ;Branch if so
```

Changing the NOP to a BPT using the MCR OPEN command will cause XDT to gain control as soon as the driver successfully obtains an I/O packet to process.

Determine the offset of the NOP instruction from the start of the driver by looking at the listing file produced by MACRO-11. (In this example it is 64 octal.)

```
        OPEN 64/DRV=XX:<CR>
        00000064/000240 3<ESC>
```

Performing I/O to the driver will cause XDT to prompt a breakpoint exception on the system console. At this point the BPT instruction can be changed back to a NOP if required and a relocation register can be set up as in the following example.

```
        BE:120064
        XDT>120000;R
```

```
        XDT>0,64/000003 240<CR>
        XDT>
```

9.0 ASSEMBLING DEVICE DRIVERS

Device drivers and their database files are assembled using the executive macro library and the system prefix file. It is often convenient to include a driver specific prefix file containing definitions for the number of controllers and the symbol which defines the driver as loadable. For a device called XX: this file would be similar to this example.

```
.TITLE  XXPFX
.IDENT  /V1.0A/
; *****     DESCRIPTION          *****
; Prefix file for XX (example) device driver
;

    X$$X        =       1       ;One XX controller
    LD$XX       =       0       ;Using a loadable driver
```

With later versions of MACRO-11 these files can be referenced from the main body of the source code using the .INCLUDE and .LIBRARY directives. This has three main advantages. The command line to assemble the driver is simplified. The command line does not depend on the directory containing the driver source code. The .CROSS and .NOCROSS directives can be used to exclude all the symbols generated by these files but not referenced elsewhere from a cross reference listing.

The included files should be referenced before the .TITLE directive for the main body of the driver source module. If this is not done then the driver will have the wrong title (RSXMC for example). The example below shows a typical beginning for a driver.

```
        .nlist
        .nocross
        .include        /lb:[11,10]rsxmc.mac/
        .include        /xxpfx.mac/
        .cross
        .list
        .title  XXDRV
        .ident  /V1.0A/
```

The library file can be included in the source code just before any symbol definition macros from it are used.

```
;
; System definitions
;
        .nocross
        .library        /lb:[1,1]exemc.mlb/
        .mcall  ucbdf$,scbdf$,pktdf$,hwddf$
        ucbdf$  sysdef=yes
        scbdf$  sysdef=yes
        pktdf$
        hwddf$  sysdef=yes
        .cross
```

10.0 CONCLUSION

The information in this article is believed to be accurate and is hopefully useful to other RSX device driver writers. If you discover any errors or have any comments on this article please spare the time to forward them to me.

THE mini-tasker

DECUS
RT-11 SIG NEWSLETTER

RT-11 Mini-Tasker

Letter "FROM THE EDITOR" 1-2
Contents of the Fall 1985 RT-11 SIG tape 3-11
Fall '85 Tape—Online Distribution 13-14
Modula-2 Programming System 15-18

RT SIG Steering Committee

SIG Chairman
John T. Rasted
JTR Associates
Meriden, CT

Symposium Coordinator
Ned Rhodes
Software Systems Group
Rockville, MD

Communications Committee Representative
FMS Contact
Susan Rasted
Software Dynamics, Inc.
Wallingford, CT

Newsletter Editor
COBOL Contact
Bill Leroy
The Software House Inc.
Atlanta, GA

Pre-Symposium Seminar Coordinator
Suite Manager
Bruce Sidlinger
Sidlinger Computer Corp.
San Antonio, TX

Standards Coordinator
Kenneth L. Aydlott
Teledyne Hastings-Raydist
Hampton, VA

Digital Counterpart
Diana L. Miller
Digital Equipment Corporation
Maynard, MA

Tape Copy Generation Contact
Ralston Barnard
Sandia Laboratories
Albuquerque, NM

APL Contact
Doug Bohrer
Bohrer & Company
Wilmette, IL

MACRO Contact
Nick Bourgeois
NAB Software Services, Inc.
Albuquerque, NM

Teco Contact
Product Planning Contact
John Crowell
Crowell Ltd.
Los Alamos, NM

DECNET Contact
Ken Demers
Adaptive Automation
New Haven, CT

UNIX Contact
Wish List Contact
Bradford Lubell
L.A. Heart Lab
Los Angeles, CA

RT-11 Hardware Contact
C Contact
Carl Lowenstein
University of Calif., San Diego
San Diego, CA

TSX Contact
Jack Peterson
Horizon Data Systems
Richmond, VA

Library Contact
Tape Copy Distribution
Tom shinal
General Scientific Corp.
Rockville, MD

BASIC Contact
Ed Stevens
EMDA Inc.
Pasadena, CA

CAMAC Contact
J.W. Tippee
Kinetic Systems Inc.
Lockport, IL

Personal Computer Contact
LUG Contact
William K. Walker
Monsanto Research Corp.
Miamisburg, OH

FORTRAN Contact
RUNOFF Contact
Robert Walraven
Multiware, Inc.
Davis, CA

FROM THE EDITOR

Auto Sysgen on a Floppy

With many APOLOGIES to Paul F. Sullivan of POLAROID CORPORATION, Cambridge, Massachusetts - for my leaving out a credit to him and his company for submitting the article last month on "Auto Sysgen on a Floppy" in the April 1986 Mini-Tasker. It was a wonderful article on an interesting subject, and if you missed it, go back and read it.

Trivia

From time to time, one must forget the cares and woes of life, and relax with a bit of trivia. With thanks to Diana Miller of the RT-11 Development Group, DEC put together an interesting DECUS program last year on PDP-11 trivia. Following are some of the TECO facts of life.

1. What does TECO stand for?
2. The TECO mnemonic H, what is its origin?
3. The TECO mnemonic T, what is its origin?
4. What was TECO designed for?
5. The TECO mnemonic V, what is its origin?
6. The TECO mnemonic Y, what is its origin?
7. Who is the RT-11 TECO contact?

Fortran-77, Pascall and miscellaneous tips

For those of you using, or thinking of using FORTRAN-77, Bob Walraven has submitted an article on the differences between V5.0 and V5.0A. In addition, he has submitted articles on VMS file transfers, and BAD block tech tips.

If this is of interest to you, and you would like to thank Bob for his efforts, please let him (or me) know. Also, please send me your own thoughts or comments for publication.

The Mini-Tasker

As we begin to look at the second year of the Mini-Tasker being included in "The Big One", we would be interested in your comments as to how well it has worked out for you, the reader. Please let me know how you feel.

Trivia - Answers

1. Tape Editor and Corrector Program
2. Whole tape
3. Type to flexowriter
4. Editing paper tapes on a PDP-1
5. Video display of current line
6. Yank a paper tape
7. John Crowell of Crow4ell, Ltd.* (the 4 is silent) * but not very

And finally, I am always looking for something to print.

Please send your submissions to The Mini-Tasker to me at:

Bill Leroy
The Software House, Inc.
2964 Peachtree Road, NW #320
P. O. Box 52661
Atlanta, GA 30355-0661

404/231-1484 or DCS (LEROY)

Bill Leroy

From the DECUS U.K., Ireland & Middle East RT Special Interest Group
September 1985:

* * *

Completion and Interrupt Routines in Oregon Software Pascal-2

The problem:

Mixing Oregon Software Pascal-2 main-line code with ANY, repeat ANY, completion or interrupt level code WHATSOEVER usually results in a program that crashes.

If the completion/interrupt code is pasmac-ed or Pascal, then stack overflow errors can occur. If the code is straight MACRO-11, then multiple errors can occur, UNLESS YOU DO NOT USE THE USER STACK AT ALL. (Which also means you do not call any routines from your completion/ interrupt code.)

The explanation:

If the main-line code is doing file I/O (and probably other functions too, but the file I/O is the one we've found), then Pascal is calling a library routine called P$LOWS, which (for some inexplicable reason) replaces the stack pointer with the address '#P$STAC-2', having first saved the original stack pointer in P$STAC.

The solution:

With pasmac-ed or Pascal code, EVEN WITH ",check=0" or "/nocheck" (respectively), the routine p$75 (save registers) still checks for stack overflow. Since the stack is now down in the psect P$RTS, this will always give a stack overflow error. You can get round THAT by:

1) in pasmac-ed code, never save more than three registers - pasmac doesn't use p$75 to save three or less registers. If you do need more than three registers, then don't us pasmac 'save'; instead, put in a call to p$75L immediately after the 'begin' and call p$77 immediately before the 'endpr', or, alternatively, save the registers on the stack yourself (if you can spare the room in your code.)

2) in Pascal code, do a /MACRO compilation, and edit all the p$75 calls to p$75L calls.

However, this will not help unless you use very little stack (and are lucky), because the stack is still down in the psect P$RTS, and will eventually overwrite the data in P$AREA. P$AREA contains EMT work areas for EMTs called by Pascal. (Typically, there are about 150 words between the beginning of P$AREA and the top of the Pascal internal stack.)

When you overwrite important data in P$AREA, you can cause multiple errors (not suprisingly).

So, this is our solution.

Front-end all completion routines with the following code, and then you can call Pascal routines, use the stack, nest/queue completion routines to your heart's content.

(Note: we have NOT addressed the PAR1 restriction in the XM monitor, except by ensuring that the Pascal run-time support and our completion and interrupt code are not in PAR1's region.)

```
;
;**********************************************************************
;
        .title  MacroRoutines
        .globl  P$STAC
;
;**********************************************************************
;
        .psect  p$code,ro
        proc    MacRtn,check=0
;
; do not use pasmac 'save' or 'rsave' - do them by hand as specified
; below
;
        begin
        mov     r3,-(sp)        ; save r3 (could be any register - use
                                ; one whose original contents you do
                                ; not need in your routine)
        mov     sp,r3           ; save original stack pointer in r3
        cmp     sp,#P$STAC      ; are we using Pascal internal stack?
        bhi     1$              ; if yes, then
        mov     P$STAC,pc       ;    restore 'normal' stack pointer
                                ; end if
1$:     mov     r3,-(sp)        ; push original stack pointer
;
; note - the register you used above does not now have its original
;        contents - e.g., if you are in a completion routine and you
;        used r0, the completion routine id is no longer in r0
;
        mov     r0,-(sp)        ; save the registers now, if required
        mov     r1,-(sp)        ; note - you could call p$75 now, if
        mov     r2,-(sp)        ; you really wanted to
                |
                |
            your code
                |
                |
endMacRtn:
        mov     (sp)+,r2        ; restore registers saved above
        mov     (sp)+,r1        ; (or call p$77, if you used p$75 to
        mov     (sp)+,r0        ; save them

        mov     (sp),sp         ; restore original stack pointer
        mov     (sp)+,r3        ; restore r3
        endpr
;
```

```
;*************************************************************************
;
        .end
```

We would be very interested to hear from anyone who has found other problems (and fixes!) in using Oregon Software Pascal-2 for REAL TIME software.

Sally Antill   Vickers Design & Projects
                      Wessex House, Market Street
Tel : 0703 619722   Eastleigh, Hants S05 4FD

-o- -oOo- -o-

RT11 and RD5x/RQDX1 Bad Block Tech Tip

Are you experiencing bad block problems on an RD51/52 and RQDX1 combination running RT11 V5.0 or V5.1? If so, then read on.

The RQDX1 performs controller initiated bad block replacement (BBR). Most of the tim~re the data is recoverable and the replacement takes place transparently. Sometimes the data is not recoverable and in this case the block is replaced and the 'Forced Error Flag' is set in the replacement block's header. This results in an error being generated when the block is read. The block is NOT bad at this point and the original bad block has been replaced. DO NOT reformat the disk, this is the very worst action that you could take. All that is needed at this point is to write on the block and the forced error flag will be cleared and the block is completely usable. Quite simply, the replaced block needs to be written to.

RT11 Engineering are aware of this problem and are working to produce changes in RT11 and to provide software tools to allow easier recovery from this problem. Please do not submit SPRs on the problem or call the CSC. The problem is known.

RT11 (TSX+) - VMS File Transfer

A short and simple expose on a method of transferring between RT11 (TSX+) and VMS when your only means of transfer is by mag-tape.

In our case we have an 11/24 (v5.1 RT11, 5.1C TSX+) with a Cipher M890 and a VAX 11/750 (VMS 4.1) with a TU80. Apologies to anyone with any other weird and/or wonderful equipment (and I know there are bound to be some in the RTSIG!) you'll have to play around. Right, on VMS version 4.0 we were all pleased to see the EXCHANGE utility, weren't we? This did actually mention RT11 (what's RT11? -- VMS SIG) unlike most of the other 25 or so documentation volumes with V4. EXHANGE will let you mount RT11 virtual disks:

```
MOUNT/VIRTUAL LOGNAM TEMP.DSK
```

and then copy the files inside out into RMS format, or you can create virtual disks:

```
INITIALIZE/CREATE  TEMP.DSK/ALLOC:1000
```

and copy into them. It's almost like having a sub-set of RT11 inside VMS.

As usual, everything is quite simple and straight forward converting Files-11 files, but EXCHANGE will only access RT11 'block addressable' devices, which of course mag-tape is not. Well done VMS.

To copy, mount the tape with a 512 byte blocksize:

```
MOUNT/BLOCKSIZE=512 MS0:
```

then copy the virtual disk(s) onto contiguous disk area:

```
COPY/CONTIGUOUS MS0:VDISK1.DSK OUTPUT DEV:VDISK1.DSK
```

Another problem is that I haven't found a convenient way of copying virtual disks that have file names with fewer than six characters, if there is more than one file per tape. Wildcards can be used if there is only one file per tape. More experimentation needed.

Malcolm Collier

FORTRAN-77/RT-11
Differences between versions V5.0 and V5.0A

The original release of FORTRAN-77/RT-11 is version V5.0, and the second release of FORTRAN-77/RT-11 is version V5.0A. A number of minor problems with V5.0 have been corrected in V5.0A. This document describes in detail the differences between these two releases.

The version of the compiler is printed in the listing output header, and is also displayed on the terminal when the compiler is run in CCL mode and a return with no other text is typed in response to the prompt.

1.0 FORTRAN-77 COMPILER DIFFERENCES

1. CHARACTER Data in COMMON

   Character data cannot be in a common block with other data types. While this is stated in the manual, V5.0 does not flag this error, and furthermore behaves very erratically following such an error.

   V5.0A detects this error and prints an error message.

2. Object File and Listing File Overflow

   V5.0 does not check for overflow of output files. As a result, if there was not enough room for the object file, no error message is generated and a corrupted object file is produced.

   V5.0A checks to see if output files attempt to overflow their allocated space. On an object file overflow a 'FATAL 06' error message is issued and the incomplete object module is deleted. On a listing file overflow a 'FATAL 07' error message is issued and the listing file is closed. If the compiler was run in CCL mode, then after either error the compiler will accept another command line. This behavior is consistent with FORTRAN-IV on RT-11 except that FORTRAN-IV does not purge incomplete object files.

3. Extra Spaces in Filename

   V5.0: The filename in the header and in the final error message will have extra spaces when the device or file name is shorter than the maximum.

   V5.0A: The extraneous spaces in the filename have been eliminated.

4. Error Message Differences

   The following changes were made to the message file F77COM.MSG between V5.0 and V5.0A:

   1. The words "I/O" in errors 77 and 80 were changed to "OPEN/CLOSE" and the words "or IF" were added to error 79 to make the actual messages consistent with the documentation in Appendix C of the FORTRAN-77/RT-11 User's Guide.

   2. Text for error messages 107 through 116 (FORTRAN-77 extensions information messages), described in Appendix C of the FORTRAN-77 User's Guide, was added. The CCL switch /Y now causes FORTRAN-77 extensions to be correctly reported. (NOTE: you must use the /W switch in conjunction with the /Y switch - see p. 1-16 of the User's Guide.)

   3. Error 94 (mixed numeric and character elements in COMMON) was changed from class W to class E. This was done because the compiler currently generates corrupt code if this error occurs. The documentation on page C-11 of the User's Guide should be changed accordingly.

5. Error After Compiler Identification

   V5.0: When CCL mode is used and a RETURN without other text is typed, the compiler identifies itself, but does not work after that unless it is restarted.

   V5.0A: The problem in V5.0 has been corrected.

6. Blank Lines in the Source

   V5.0: A blank line in the source is ignored, and does not appear in the listing output.

   V5.0A: A blank line in the source appears as a blank line in the listing.

7. INCLUDE with Illegal Name

   V5.0: If an INCLUDE statement has a filename with illegal characters in it, the system will crash or a fatal error will result.

   V5.0A: If an INCLUDE statement has a filename with illegal characters in it, an error message is generated.

8. INCLUDE Compatability with RSX

   V5.0: The argument of an INCLUDE statement can only begin with a valid RT-11 filename.

   V5.0A: For compatibility with RSX, the filename in an INCLUDE statement may be proceeded by the notation [n,m]. If the left bracket is detected, text is ignored until a right bracket is encountered.

9. Virtual Array PSECT

   V5.0: If the job has no virtual arrays, then the vsect $VIRT is not generated, so the symbol $VIRSZ is not defined.

   V5.0A: The virtual array vsect was renamed from $VIRT to . VIR. because this name has special meaning to the linker. The vsect is always generated even when the job has no virtual arrays, so that the symbol $VIRSZ will always be defined. If the job has no virtual arrays, then $VIRSZ has a value of zero.

10. /A CCL Switch

    V5.0: when the statistics switch, /A, is used, the operating system is always reported to be RT-11 even if it is actually TSX+.

    V5.0A: TSX+ is reported properly.

11. /B CCL Switch

    V5.0: using the non-XM version of the compiler in conjunction with the /B (debugger switch) will often crash the system.

    V5.0A: the /B switch works properly.

12. /Y Switch

    V5.0: the /Y switch (CCL extensions) does not work if no argument has been supplied.

    V5.0A: the /Y switch may be used without an argument.

13. PSECT Ordering

    V5.0A may generate some empty PSECTs in object modules in order to establish desirable PSECT ordering.

14. Extraneous error message

    V5.0: An extraneous (and sometimes incorrect) error message sometimes appears on the terminal when the code has errors.

    V5.0A: The extraneous error message has been eliminated.

15. Size of Non-XM compiler

    In V5.0A the overlay structure of the non-XM version of the compiler has been optimized. Although the .SAV file is bigger than for V5.0, it requires about 500 less words of memory.

2.0 OBJECT TIME SYSTEM (OTS)

1. DIRECT ACCESS I/O

   V5.0: If records are written to a file in reverse order and immediately followed by reads in ascending order, the last record written is occasionally lost.

   V5.0A: The problem in V5.0 has been eliminated.

2. SEQUENTIAL UNFORMATTED READ

   V5.0: There were two problems with the sequential unformatted READ statement. First, if a record starts on word 0 of a block other than the first block of the file, record pointers for subsequent blocks (if present) became misaligned, and erroneous reads occur. Second, if when reading a record, the I/O list contains fewer bytes than the record in the file, the unused part of the record is not discarded. As a result, when reading the next record, bytes from the unused part of the previous record will be used first, giving erroneous results.

   V5.0A: The problems with V5.0 have been fixed. Furthermore FORTRAN-77 sequential unformatted I/O is now fully compatible with its FORTRAN-IV counterpart. (FORTRAN-77 byte variables are stored one per byte, whereas the default in FORTRAN-IV is to store byte variables one per word. You must use the optional module UIOBYT.OBJ to store one byte per byte in FORTRAN-IV.)

3. $ FORMAT SPECIFIER

V5.0: If a sequential formatted I/O statement on any logical unit uses the $ format specifier, the next line output to TT: behaves as if the $ specifier acts on the terminal's LU. This causes this line to overwrite the last written line on TT:.

V5.0A: This problem has been corrected.

4. INITIAL LINE FEED BEFORE TERMINAL INPUT

Specifying CARRIAGECONTROL='NONE' on the terminal's LU should disable a line-feed from being issued when a READ statement is executed. V5.0 fails to do this, but V5.0A handles this correctly.

5. "REWIND" OF TT: OR LP:

V5.0 gives a 'rewind error' message if an attempt is made to REWIND a non file-structured device.

V5.0A simply dumps the buffer, and produces no error.

6. STATUS='UNKNOWN' KEYWORD IN 'OPEN' STATEMENT

V5.0 always creats a NEW file when a STATUS='UNKNOWN' keyword specifier is used in the OPEN statement.

V5.0A first checks to see if the file already exists, and opens with a STATUS='OLD' if a file is found.

7. UNNECESSARY PSECT REMOVED

In V5.0 of the OTS, an unnamed PSECT of 211 words was accidentally present.

This has been removed from V5.0A.

8. ASSIGNing LU's

V5.0: The RT-11 command ASSIGN TT 4 does not direct I/O on LU 4 to the terminal.

V5.0A: This problem has been corrected.

9. Global Name Conflicts

V5.0: The OTS globals LIMIT, WNDOW, WINMAP, FTRAP, RTOPEN, and QSET have legal FORTRAN variable names, so a name conflict can occur.

V5.0A: The OTS globals have been renamed to LIMIT$, WNDOW$, WNMAP$, $FTRAP, $RTOPN, and $QSET to avoid naming conflicts.

10. Address of Work Area

For V5.0A a new FORTRAN-callable OTS function has been added to return the address of the OTS work area ($OTSVA). The routine is called with the statement I = IAOTSA().

11. Virtual Array Initialization

V5.0: The virtual array initialization routine $VINIT is included in all jobs even if no virtual arrays are used.

V5.0A: The virtual array initialization routine $VINIT has been moved to VIRTXM so that it is included only if VIRTXM is linked with the program. Otherwise a dummy $VINIT is linked with the program from F77OTS.

12. Unnamed PSECT

V5.0: The OTS routines DATE and IDATE are in an unnamed psect.

V5.0A: The OTS routines DATE and IDATE are in the psect $$OTSI.

13. ENDFILE as first file I/O

V5.0: When an ENDFILE is performed on a logical unit before any other I/O statement is executed, the ENDFILE mark is a sequential formatted one only. If the file is then accessed as an unformatted file, an error will occur.

V5.0A: When an ENDFILE is performed on a logical unit before any other I/O statement is executed, a combination sequential formatted and unformatted ENDFILE mark is written so that either form of access can be used to the file.

14. VIRTUAL Jobs

V5.0: There are three problems associated with virtual FORTRAN-77 jobs:

1. When a FORTRAN job is run, one of the first things it does is issue a .SETTOP macro to gain some free memory to be used as a work area. This work area is used to build a line buffer, the logical unit control tables (LUBs), and the actual I/O buffers for disk file access. If the job is VIRTUAL, then the space obtained by the .SETTOP begins on the next 4K boundary after the program high limit. Since the work area must not be overlayed, this means that a whole 4K page must be allocated to the work area. In other words, the size of a virtual job cannot exceed 28K (or 24K if virtual arrays are used, since they need a 4K page also). Forcing the work area into the root segment, so that the 4K region can be recovered, requires a SIPP patch to the .SAV file. This patch is discussed in F77/RT Application Note 1.

2. FORTRAN jobs require additional queue elements to be allocated for I/O. For a privileged (i.e., non-virtual) job the queue elements are created in the job's work area obtained from free memory. For a virtual job, the queue elements must be allocated in the first 4K of memory, otherwise they can be destroyed. Normally the PSECT ordering forced by OTS is sufficient so that this will happen, but for some very large jobs, the PSECT where the queue elements is allocated gets pushed into the second 4K of memory, and as a result they get destroyed when the job runs. If this problem is suspected, the LINK map should be examined to be sure that the variable $QBLK is in the first page of memory. If it is not, then the procedures described in F77/RT Application Note 1 should be followed.

3. As discussed above, FORTRAN virtual jobs without virtual arrays attempt to allocate the work area in the top 4K of memory. For reasons we cannot determine at this time, if a .SETTOP is performed that includes the last 32 words of memory, then the job sometimes aborts when run. To circumvent this problem, the .SAV file should have a SIPP patch applied that causes the .SETTOP to not ask for the top 32 words of memory. This patch is described in F77/RT Application Note 1.

V5.0A: The three problems with V5.0 discussed above have all been addressed in V5.0A:

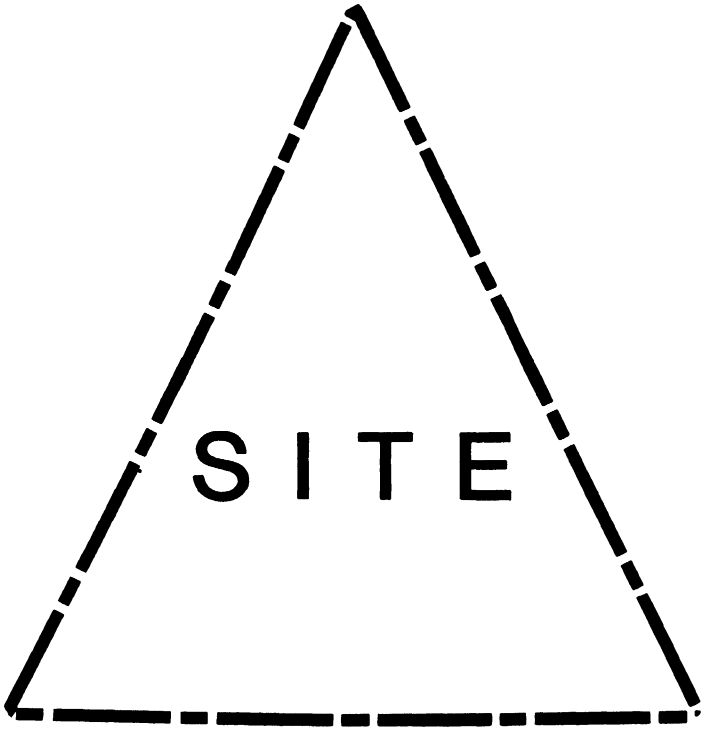
1. If the job uses virtual arrays, then the module VIRTXM must be linked with the job, and this module automatically forces the work area to be allocated in the root. The default size of the work area is big enough for a sequentially formatted I/O record length of 136. (the default), six simultaneously open logical units (the default), and six i/o buffers (enough for all six logical units if single buffered). If this is not adequate, or the job is virtual without virtual arrays, then the following module should be modified as appropriate and linked with the job in the root.

```
*************************************************************

   .TITLE  VIMPUR  Work Area Allocation for Virtual XM

;-------------------------------------------------------------
;       Allocate Fortran work space (line buffer, channel
;       table, I/O block buffers, etc.) in low mem psect.
;-------------------------------------------------------------

       UNITS = 6.               ; Number of LUNs
        LRECL = 136.             ; Line Buffer Length

        NSIZE = 512.+40.+1.

        .PSECT  $$OTSS,RW,D,LCL,REL,CON
$IOBKF::

;  Allocate bytes: linebuffer+UNITS*(Buffer+ChannelTb+LUB)

        .BLKB   UNITS*NSIZE+LRECL
        .EVEN
$IOBKL::

        .END
*************************************************************
```

2. The PSECT ordering has been modified so that the PSECT in which $QBLK is defined is assigned earlier, assuring that it will be in the first 4K of memory.

3. The problem with the last 32 words of memory still exist, but OTS now automatically avoids the last 32 words in the appropriate case so that no SIPP patch is necessary.

DECUS



SITE

SIT

SITE

Chairman
HMS SIG Liason
RT SIG Liason
David Hunt
Lawrence Livermore National Lab
MS L-230
P.O. Box 808
Livermore CA. 94550
(802) 656-3190

Symposia Coordinator
VAX SIG Liason
Michael Weaver
OJCS/JAD/TSD
Room 1D940
Washington D.C. 20301-5000
(202) 694-6772

Communications Committee Representative
AI SIG Liason
Terry C. Shannon
Digital Review
160 State St.
6th Floor
Boston, MA. 02109
(617) 367-7190

Newsletter Editor
Networks SIG Liason
OA SIG Liason
Gregory N. Brooks
Washington University
Behavior Research Labs
1420 Grattan St.
St. Louis, MO. 63104
(314) 241-7600 ext. 257

Session Note Editor
Large SIG liason
Gary Bremer
Emerson Electric Co.
8100 W. Florisant
St. Louis, MO. 63136
(314) 553-4448

Product Coordinator
HMS SIG liason
Jim Corrigan
KI Research
2116 E. Arapaho
Suite 521
Richardson, TX.
(214) 960-5674 or (714) 633-9308
Assistant Library Coordinator
RSTS SIG Liason
Timothy Frazer
Specialized Bicycle Components
15130 Concord Circle #77
Morgan Hill, CA. 95037
(408) 779-6229

Library Coordinator
DMS SIG Liason
Large SIG Liason
Larry W. Hicks
Relational Database services
P.O. Box 644
121 S. Main St.
Kernersville, NC. 27285-0644
(919) 996-4882

Hardware Coordinator
HMS SIG Liason
Emily Kitchen
A.H. Robbins Co.
1211 Sherwood Ave. RT-2
Richmond, VA. 23220
(804) 257-2925

Staff Management
Adam Zavitski
720 N. Zenith
Tulsa, OK. 74127

Pre-Symposia Seminar Coordinator
Phillip Ventura

Members-At_Large

George Hilton
Compucard International
777 Summer St.
Stamford, CT. 06091
(203) 324-9261

Debbie Boole
Texas Instruments
13510 N. Central
M/S 437
Dallas, TX. 75266
(214) 995-4629

DEC Counterparts

Sonja Sokal
Digital Equipment Corp.
Stow MA.

Lil Holloway
Digital Equipment Corp.
Bedford MA.

Susan Porada
Digital Equipment Corp.
Marlboro, MA.

Gary Siftar
Digital Equipment Corp.
Tulsa, OK.

UNISIG

DECUS

UNI

UNISIG Chair

James W. Livingston, Jr.
Measurex Corporation
1 Results Way
Cupertino, CA 95014
408-255-1500 x5556
ihnp4!decwrl!jwl

UNISIG Symposia Coordinator

Stephen M. Lazarus
Ford Aerospace, MS X-20
3939 Fabian Way
Palo Alto, CA 94303
415-852-4203
ihnp4!fortune!wdl1!sml

UNISIG Session Notes Editor

Kurt L. Reisler
Hadron Incorporated
9990 Lee Highway
Fairfax, VA 22030
703-359-6100
decvax!seismo!hadron!klr

UNISIG Newsletter Co-editor

William Toth
Harvard-Smithsonian
Center for Astrophysics
60 Garden Street, P-353
Cambridge, MA 02138
617-495-7181
harvard!hrvsmth!toth

UNISIG Newsletter Co-editor

James W. Livingston, Jr.

UNISIG Administrative Daemon

Dorothy Geiger
The Wollongong Group
49 Showers Drive, 451
Mountain View, CA 94040
415-948-1003
ihnp4!decwrl!dgeiger

UNISIG Tape Librarian

Carl Lowenstein
Marine Physical Laboratory
Scripps Institute of Oceanography, P-004
La Jolla, CA 92093
619-294-3678
(ihnp4|decvax|akgua|dcdwest|ucbvax)
!sdcsvax!mplvax!cdl

UNISIG Usenet Liason

Joe Kelsey
FlexComm Corporation
711 Powell Ave. SW
Renton, WA 98055
allegra!fluke!joe

UNISIG Standards Coordinator

Jeff Gilliam
National Semiconductor
2900 Semiconductor Drive, MS C2303
Santa Clara, CA 95051
408-721-3801
ihnp4!nsc!voder!jeff

UNISIG Minister Without Portfolio

Norman Wilson
Bell Laboratories, 2C-529
600 Mountain Avenue
Murray Hill, NJ 07974
201-582-2842
(decvax|ihnp4)!research!norman

UNISIG DEC Counterpart

Roseann Maclean
DEC, MKO2-1/H10
Continental Boulevard
Merrimack, NH 03054
603-884-5702
decvax!maclean

NEWSLETTER OF THE VAX SYSTEMS SIG

Pageswapper

Our Mascot

VAX SYSTEMS
SIG
DECUS
U.S. CHAPTER

VAX

General material for publication in the Pageswapper should be sent (US mail only -- no "express" services please) to:

Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station
Cambridge, MA 02139-0901
USA

Preference is given to material submitted as machine-readable text (best is Runoff source). Line length should not exceed 64 characters. Please do not submit program source, as that is better distributed on the VAX SIG tape.

For information about on-line submission to the Pageswapper, dial (in the United States):

(617) 262-6830

and log in with the username PAGESWAPPER.

Change of address, reports of non-receipt, and other circulation correspondence should be sent to:

DECUS U.S. Chapter
Attention: Publications Department
249 Northboro Road (BPO2)
Marlborough, MA 01752
USA

Only if discrepancies of the mailing system are reported can they be analyzed and corrected.

Editor's Workfile

by Larry Kilgallen, Pageswapper Editor

On-line Pageswapper Submissions -

Resources are now available to accomplish this longstanding obvious goal. The one hitch is that the caller must pay for the phone call to Boston. To find out more, dial (617) 262-6830 (in the United States) using a 1200 baud modem and give PAGESWAPPER as your username. There is a registration protocol which involves sending you back an individual username in the mail, more for purposes of establishing an authentication base for publication than for computer security.

One of my goals for this arrangement is to speed the turnaround on I/Os. The idea is that some callers may choose to peruse the as-yet-unprinted I/Os and be able to respond to them such that the response hits the press at the same time as the inquiry. We will ensure that I/Os received in the mail get put on-line so that callers can read I/Os from mailers.

The mechanism should also work reasonably well for regular articles, letters to the Pageswapper, etc.

Just a Modest VMS Proposal

edited by
Robert B. Goldstein
Eye Research Institute of Retina Foundation
and
James Downward
KMS Fusion, Inc.

At the Fall '85 Decus Symposium, I attended the "Just a Modest VAX Proposal" session. This session provided an opportunity for users to present gripes, taunts, and ideas to Digital developers. Some points were serious and others less so. Some of the items presented were sort of verbal System Improvement Requests (SIRs). I was so impressed that I decided to write this summary.

I have excluded some of the more obscure suggestions as well as those that clearly were the result of misinformation. I have indicated, when known, the category into which the judges put each proposal: Cult, Noted or UPG (Useless Product Group). The order of the items here is roughly the order in which they were presented. My apologies for any distortions.

1. DEC CUSTOMER SALES UPDATE - This should be a 'sanitized' version of the DEC internal sales update - with the phone numbers removed. This would enable a customer to specify to his salesperson the exact item of interest by noting the page number in the sales update.

2. PAY AS YOU GO SOFTWARE - DEC should consider software to be a service rather than a product. Each time a layered product is used, it should write a record into an accounting file. This file would be dumped once/month by DEC and used for billing. The amount of payment is directly linked to the amount of usage, and NOT to the type of machine for which it is licensed. This type of pricing is particularly appropriate in a clustered environment. It would also result in a usage profile for your system.

3. DCL COMMAND TRACKING - Keep an accounting record of DCL commands. Have DCL commands be trackable by the accounting utility. - Noted

4. UNBUNDLED VMS - Marketing made easy, or VMS in five easy pieces (and 42 difficult ones). Consistent with DEC's "new wave" marketing policies, a simple restructuring of VMS for fun and (especially) profit. - UPG

5. SUPPORT ALL DEC TERMINALS - The VMS command, SET TERM/DEVICE=devicename, should work for all serial devices that DEC manufactures. Currently it does not work for VT240, LN03, or LA50. - Noted

6. USE THE PHONE, DUMMY - The SPR forms have a space for the phone number of the originator of the SPR. Whoever processes the SPRs never uses the phone number, but instead responds formally, with a turnaround time of many months. - Noted

7. LIB$PUT_ERROR - Provide a LIB$PUT_ERROR analogous to LIB$PUT_OUTPUT, which would write to SYS$ERROR. - Noted

8. SYS$DEFAULT_PRINT - A logical should be added that would define the default printer and batch queues for a user. - Noted

9. SHOW ME MORE - Modify the SHOW PROCESS command so that it indicates the indirect command file currently being executed by the process. - Noted

10. BACKUP COPY UTILITY - Have a BACKUP/COPY_SAVESET command that will allow a saveset to be easily copied to and from magtape. - Noted

11. VAX SYSTEM SECURITY LEAK - VMS security bugs are posted on public mailing lists. Can we please have a secure way of finding out about security bugs in VMS? Perhaps a timely DSIN flash? - Noted

12. PAINLESS VMS UPGRADES - VMS upgrades should be made less painful, possibly by allowing it to be done on-line and possibly on a spare backup copy of the system disk.

13. VIRTUAL TERMINAL SUPPORT - Virtual terminal support needs to be improved. Some items to include are a) indication of detatchment b) indication of buffer not empty. - Noted

14. MAKE TERMINAL SERVER INFO AVAILABLE - Utilities that usually give information about the port connection should also be able to report the terminal server name and port number. Some utilities that should do this are Accounting, SHOW TERM and NCP.

15. LETS GET "TP"-ED: A TERMINAL PRINTER PORT DRIVER. VMS should provide support for printers connected through the printer ports on terminals. This device can be designated TP: . Utilities and library routines would have support for this device rather than users writing their own utilities.

16. NEW DUMP SWITCHES - Dump switches such as /FROM=1st block and /TO=last block and /BACKWARDS are needed (FROM and TO already exist as options for the /BLOCKS and /RECORDS switches - ed.) - Cult

17. CUSTOMER SUPPORT CENTER SHOULD BE TOLD FIRST - We received Fortran 4.n before Colorado, hence installation problems weren't handled well. - Cult

18. VMS SECURITY ERASE SHOULD BE ASYNCHRONOUS - We shouldn't have to wait for the $ after we delete a file. - Cult

19. ACL ON BATCH/PRINT QUEUES - ACLs exist on files, devices, global sections - how about ACL on queues? - Noted

20. PUSH BUTTON VMS - Make VMS "menu" driven - Cult

The following items were submitted but time did not permit their presentation:

1. 1 LINE COMMAND PROCEDURES - Expand DCL so that p1-p8 symbol substitution is allowed within a symbol definition. Example is PROTECT :== SET FILE 'p1'/ PROTECT=('p2')

2. WORKINGSET SIZE ADJUSTMENT - DEC encourages us to "tune" the system by using WSQUOTA, WSDEFAULT, WSEXTENT, etc. How about letting us specify these on a per IMAGE rather than a per username basis?

3. SYSTEM STATE SAVE/RESTORE UTILITIES - The purpose is to allow a VAX that is powered through a short-lived UPS power system to save its state and recover it smoothly. This would take the form of a pair of complementary VMS utilities. The first would allow all outstanding I/O to complete but no new I/O to start. When this is done, the system state is dumped to disk and the system crashed. The second utility would restore the saved state and resume execution at the point at which it was interrupted.

4. MACRO32 .INCLUDE - MACRO32 needs a .INCLUDE <module name> statement.

5. RAISE SHOW PROCESS TO NEW HEIGHTS - SHOW PROCESS should be able to show the code segment of images with large data areas. This would be SHOW PROCESS/CONT/BASE=ADDRESS.

Spring 1986 SIR Ballot Results

Gary Grebus
SIR Coordinator

It seems that the VAXSIG membership believes in the philosophy of simpler is better. The simplified SIR Ballot, published in the February 1986 issue of the Pageswapper has finally reversed the trend of declining participation. The 364 ballots received by the deadline is more than double the response received to the Fall 1985 ballot. Hopefully, participation will continue to build.

It's still hard to believe that, out of the thousands of VAXSIG members who get the SIG newsletters, so few people take the time to give their opinion. The SIR ballot is the only on-going program by which the SIG provides input to Digital. Top 10 (and other) SIR's continue to be incorporated into VMS. Digital has repeatedly encouraged the use of this channel of communication. The new ballot takes much less time to fill out. Maybe there really are only a few hundred people reading the Pageswapper?

The summary of this voting appears below. Digital's response to the top 10 requests overall will be presented at the Fall 1986 DECUS Symposium in Dallas. At the same session we will also receive a progress report on past top 10 requests that are still in the works. Look for a full report in the next issue of the Pageswapper.

Interpreting the SIR Ballot Results

The results of the System Improvement Request ballot are show on the following pages. All of the reports have the same one page format. Following the report title is the number of ballots counted for that report. The number shown on the "All Users" report is the total number of ballots which were returned. The ballots on the "11/780 Users" report is the number of ballots which checked the "11/780" blank on the ballot questionnaire, and so on.

The SIR's are listed on the page in order of points received, from highest to lowest. The entry for each SIR begins with the SIR number (from the ballot), a brief description, and the total number of votes received by that SIR. Next are listed the number of ballots which assigned positive and negative votes to the SIR. These numbers are expressed both as percentage of the total number of ballots and as a raw count. The percentage is based on the number of ballots shown at the top of the page.

The data is summarized in two different ways. First, there are a series of reports broken down by user category. The user categories are defined by the questionnaire portion of the SIR ballot. A ballot was counted in each user category which was checked off, for example "11/780 user". Finally, there are a series of reports ranking the SIR's within SIR category. The SIR categories are those shown on the ballot, for example "DCL and Utilities" and "Commercial". The reports by SIR category use the data from all ballots received.

THE TOP 50 SIR's AS RANKED BY ALL USERS Total ballots in this category: 364

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	200	55.2	0.3	201	1
9	Support linking of terminal sessions	163	47.3	2.5	172	9
20	Support a "printable" file attribute	138	39.0	1.1	142	4
25	Enhance VMS MAIL facility	136	37.6	0.3	137	1
33	Keep times synchronized on cluster	121	33.5	0.3	122	1
42	Record attempted usernames on login fail	118	35.7	3.3	130	12
35	Provide cluster-wide management tools	105	28.8	0.0	105	0
19	Add description field to file header	102	32.1	4.1	117	15
6	Support wildcards in $GETDVI	95	26.1	0.0	95	0
31	Improve DCL substitution and parsing	90	27.2	2.5	99	9
29	Enhance DCL command recall	89	28.3	3.8	103	14
5	Support TCP/IP on VMS	88	24.2	0.0	88	0
14	Support standard print file format	82	23.4	0.8	85	3
4	Allow SYSDUMP files not on sys disk	82	22.5	0.0	82	0
16	Improve call interface to job queues	75	20.6	0.0	75	0
52	Provide column editting in EDT	73	23.9	3.8	87	14
12	Provide a memory-disk for VMS	67	21.7	3.3	79	12
32	Enhance the DEASSIGN command	67	19.5	1.1	71	4
1	Provide program access to DECnet config	66	18.1	0.0	66	0
44	Prevent users from reusing passwords	63	25.0	7.7	91	28
45	Provide ACL protection on printers	59	16.2	0.0	59	0
43	Allow security alarms to be sent to file	56	15.7	0.3	57	1
36	Provide DCL facility like NCP TELL	51	14.3	0.3	52	1
38	Allow image-controlled file access	48	15.4	2.2	56	8
46	Fix handling of ACL's on installed image	47	13.2	0.3	48	1
55	Provide nesting depth counts in FORTRAN	47	12.9	0.0	47	0
28	Add restart capability to DIRECTORY	46	15.9	3.3	58	12
2	Add system wide event flag clusters	45	12.4	0.0	45	0
54	Provide overwrite mode in EDT	43	14.6	2.7	53	10
50	Allow EDT to set tabs in any column	43	15.1	3.3	55	12
24	Increase limit on foreign command size	41	11.3	0.0	41	0
27	Improve SHOW PROCESS/CONTINUOUS	41	11.5	0.3	42	1
41	Improve node authentication in DECnet	41	11.3	0.0	41	0
56	Provide "64-bit mode" qualifer for FORTR	40	12.4	1.4	45	5
48	Better control of operator logins	39	12.9	2.2	47	8
10	Add utility to list timer queue	38	10.7	0.3	39	1
3	Improve SMG support of hardcopy	38	10.7	0.3	39	1
34	Support cluster-wide spooled devices	35	9.6	0.0	35	0
7	No trailing commas in sys. svc. calls	35	13.2	3.6	48	13
40	Add DECnet End-to-end encryption	34	9.9	0.5	36	2
60	Improve debugger support for MACRO	33	9.1	0.0	33	0
18	Make SWAPPER multi-threaded	32	9.1	0.3	33	1
13	Flag autoconfig devices in SYSGEN	32	9.1	0.3	33	1
17	The VMS source kit should be cheaper	32	11.5	2.7	42	10
53	Provide SPAWN command in EDT	30	12.6	4.4	46	16
58	Provide DO-UNTIL in FORTRAN	28	10.2	2.5	37	9
51	Provide window support in EDT	27	11.3	3.8	41	14
23	Provide better control of SYS$ERROR	25	7.4	0.5	27	2
37	Support CI based intertask transfers	25	7.4	0.5	27	2
15	Improve SMG input routines	24	6.6	0.0	24	0

THE TOP 50 SIR's AS RANKED BY VAX 8600 USERS

Total ballots in this category: 86

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
33	Keep times synchronized on cluster	53	61.6	0.0	53	0
26	Show open files for a process	50	58.1	0.0	50	0
35	Provide cluster-wide management tools	45	52.3	0.0	45	0
25	Enhance VMS MAIL facility	36	41.9	0.0	36	0
9	Support linking of terminal sessions	34	43.0	3.5	37	3
4	Allow SYSDUMP files not on sys disk	30	34.9	0.0	30	0
20	Support a "printable" file attribute	30	34.9	0.0	30	0
6	Support wildcards in $GETDVI	24	27.9	0.0	24	0
38	Allow image-controlled file access	24	29.1	1.2	25	1
29	Enhance DCL command recall	23	29.1	2.3	25	2
19	Add description field to file header	23	31.4	4.7	27	4
5	Support TCP/IP on VMS	22	25.6	0.0	22	0
36	Provide DCL facility like NCP TELL	21	24.4	0.0	21	0
41	Improve node authentication in DECnet	21	24.4	0.0	21	0
1	Provide program access to DECnet config	20	23.3	0.0	20	0
43	Allow security alarms to be sent to file	20	23.3	0.0	20	0
42	Record attempted usernames on login fail	18	29.1	8.1	25	7
44	Prevent users from reusing passwords	18	29.1	8.1	25	7
45	Provide ACL protection on printers	18	20.9	0.0	18	0
40	Add DECnet End-to-end encryption	16	18.6	0.0	16	0
24	Increase limit on foreign command size	16	18.6	0.0	16	0
32	Enhance the DEASSIGN command	16	18.6	0.0	16	0
31	Improve DCL substitution and parsing	15	20.9	3.5	18	3
27	Improve SHOW PROCESS/CONTINUOUS	14	16.3	0.0	14	0
2	Add system wide event flag clusters	14	16.3	0.0	14	0
34	Support cluster-wide spooled devices	13	15.1	0.0	13	0
12	Provide a memory-disk for VMS	12	15.1	1.2	13	1
16	Improve call interface to job queues	12	14.0	0.0	12	0
14	Support standard print file format	10	12.8	1.2	11	1
28	Add restart capability to DIRECTORY	10	17.4	5.8	15	5
46	Fix handling of ACL's on installed image	9	10.5	0.0	9	0
10	Add utility to list timer queue	8	10.5	1.2	9	1
37	Support CI based intertask transfers	8	10.5	1.2	9	1
60	Improve debugger support for MACRO	8	9.3	0.0	8	0
15	Improve SMG input routines	7	8.1	0.0	7	0
55	Provide nesting depth counts in FORTRAN	7	8.1	0.0	7	0
56	Provide "64-bit mode" qualifer for FORTR	7	9.3	1.2	8	1
13	Flag autoconfig devices in SYSGEN	7	9.3	1.2	8	1
3	Improve SMG support of hardcopy	6	7.0	0.0	6	0
18	Make SWAPPER multi-threaded	6	8.1	1.2	7	1
54	Provide overwrite mode in EDT	5	11.6	5.8	10	5
48	Better control of operator logins	5	9.3	3.5	8	3
49	Add asynchronous data security erase	5	5.8	0.0	5	0
52	Provide column editting in EDT	5	15.1	9.3	13	8
17	The VMS source kit should be cheaper	4	12.8	8.1	11	7
23	Provide better control of SYS$ERROR	4	5.8	1.2	5	1
61	Provide LSE support for MACRO	4	4.7	0.0	4	0
57	Improve data structure support in FORTRA	3	5.8	2.3	5	2
58	Provide DO-UNTIL in FORTRAN	2	7.0	4.7	6	4
50	Allow EDT to set tabs in any column	2	11.6	9.3	10	8

THE TOP 50 SIR's AS RANKED BY 11/782 USERS

Total ballots in this category: 11

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	7	63.6	0.0	7	0
25	Enhance VMS MAIL facility	6	54.5	0.0	6	0
19	Add description field to file header	5	45.5	0.0	5	0
35	Provide cluster-wide management tools	5	45.5	0.0	5	0
44	Prevent users from reusing passwords	5	45.5	0.0	5	0
5	Support TCP/IP on VMS	4	36.4	0.0	4	0
6	Support wildcards in $GETDVI	4	36.4	0.0	4	0
33	Keep times synchronized on cluster	4	36.4	0.0	4	0
4	Allow SYSDUMP files not on sys disk	4	36.4	0.0	4	0
40	Add DECnet End-to-end encryption	4	36.4	0.0	4	0
42	Record attempted usernames on login fail	4	36.4	0.0	4	0
20	Support a "printable" file attribute	4	36.4	0.0	4	0
32	Enhance the DEASSIGN command	3	27.3	0.0	3	0
1	Provide program access to DECnet config	3	27.3	0.0	3	0
13	Flag autoconfig devices in SYSGEN	3	27.3	0.0	3	0
24	Increase limit on foreign command size	3	27.3	0.0	3	0
41	Improve node authentication in DECnet	3	27.3	0.0	3	0
17	The VMS source kit should be cheaper	3	27.3	0.0	3	0
18	Make SWAPPER multi-threaded	3	27.3	0.0	3	0
46	Fix handling of ACL's on installed image	3	27.3	0.0	3	0
29	Enhance DCL command recall	2	18.2	0.0	2	0
16	Improve call interface to job queues	2	18.2	0.0	2	0
12	Provide a memory-disk for VMS	2	18.2	0.0	2	0
28	Add restart capability to DIRECTORY	2	18.2	0.0	2	0
36	Provide DCL facility like NCP TELL	2	18.2	0.0	2	0
38	Allow image-controlled file access	1	9.1	0.0	1	0
39	Implement mandatory security controls	1	9.1	0.0	1	0
2	Add system wide event flag clusters	1	9.1	0.0	1	0
7	No trailing commas in sys. svc. calls	1	9.1	0.0	1	0
34	Support cluster-wide spooled devices	1	9.1	0.0	1	0
9	Support linking of terminal sessions	1	27.3	18.2	3	2
31	Improve DCL substitution and parsing	1	9.1	0.0	1	0
49	Add asynchronous data security erase	1	9.1	0.0	1	0
50	Allow EDT to set tabs in any column	1	9.1	0.0	1	0
52	Provide column editting in EDT	1	9.1	0.0	1	0
56	Provide "64-bit mode" qualifer for FORTR	1	9.1	0.0	1	0
60	Improve debugger support for MACRO	1	9.1	0.0	1	0
61	Provide LSE support for MACRO	1	9.1	0.0	1	0
11	Default to OWNER:RWED on CREATE/DIR	0	27.3	27.3	3	3

THE TOP 50 SIR's AS RANKED BY 11/780 and 11/785 USERS   Total ballots in this category: 259

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	147	56.8	0.0	147	0
9	Support linking of terminal sessions	115	47.1	2.7	122	7
33	Keep times synchronized on cluster	105	40.9	0.4	106	1
25	Enhance VMS MAIL facility	98	37.8	0.0	98	0
20	Support a "printable" file attribute	97	38.2	0.8	99	2
35	Provide cluster-wide management tools	90	34.7	0.0	90	0
42	Record attempted usernames on login fail	75	32.8	3.9	85	10
29	Enhance DCL command recall	71	30.1	2.7	78	7
6	Support wildcards in $GETDVI	70	27.0	0.0	70	0
19	Add description field to file header	69	30.5	3.9	79	10
5	Support TCP/IP on VMS	67	25.9	0.0	67	0
4	Allow SYSDUMP files not on sys disk	66	25.5	0.0	66	0
31	Improve DCL substitution and parsing	64	27.4	2.7	71	7
16	Improve call interface to job queues	58	22.4	0.0	58	0
1	Provide program access to DECnet config	56	21.6	0.0	56	0
14	Support standard print file format	52	20.5	0.4	53	1
43	Allow security alarms to be sent to file	50	19.7	0.4	51	1
44	Prevent users from reusing passwords	50	26.3	6.9	68	18
32	Enhance the DEASSIGN command	48	19.7	1.2	51	3
52	Provide column editting in EDT	48	23.6	5.0	61	13
45	Provide ACL protection on printers	46	17.8	0.0	46	0
38	Allow image-controlled file access	43	18.9	2.3	49	6
36	Provide DCL facility like NCP TELL	43	17.0	0.4	44	1
41	Improve node authentication in DECnet	37	14.3	0.0	37	0
27	Improve SHOW PROCESS/CONTINUOUS	36	13.9	0.0	36	0
2	Add system wide event flag clusters	35	13.5	0.0	35	0
46	Fix handling of ACL's on installed image	35	13.5	0.0	35	0
12	Provide a memory-disk for VMS	35	16.6	3.1	43	8
56	Provide "64-bit mode" qualifer for FORTR	34	13.9	0.8	36	2
55	Provide nesting depth counts in FORTRAN	33	12.7	0.0	33	0
28	Add restart capability to DIRECTORY	33	16.2	3.5	42	9
24	Increase limit on foreign command size	31	12.0	0.0	31	0
40	Add DECnet End-to-end encryption	30	12.0	0.4	31	1
54	Provide overwrite mode in EDT	29	14.7	3.5	38	9
50	Allow EDT to set tabs in any column	27	14.7	4.2	38	11
34	Support cluster-wide spooled devices	26	10.0	0.0	26	0
48	Better control of operator logins	25	12.4	2.7	32	7
10	Add utility to list timer queue	24	9.7	0.4	25	1
13	Flag autoconfig devices in SYSGEN	22	8.9	0.4	23	1
18	Make SWAPPER multi-threaded	22	8.9	0.4	23	1
3	Improve SMG support of hardcopy	22	8.5	0.0	22	0
23	Provide better control of SYS$ERROR	21	8.9	0.8	23	2
60	Improve debugger support for MACRO	21	8.1	0.0	21	0
37	Support CI based intertask transfers	20	8.1	0.4	21	1
17	The VMS source kit should be cheaper	20	11.2	3.5	29	9
49	Add asynchronous data security erase	19	9.3	1.9	24	5
58	Provide DO-UNTIL in FORTRAN	17	10.0	3.5	26	9
15	Improve SMG input routines	17	6.6	0.0	17	0
53	Provide SPAWN command in EDT	16	11.6	5.4	30	14
57	Improve data structure support in FORTRA	16	6.9	0.8	18	2

THE TOP 50 SIR's AS RANKED BY 11/750 USERS   Total ballots in this category: 213

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	114	54.0	0.5	115	1
9	Support linking of terminal sessions	90	45.5	3.3	97	7
20	Support a "printable" file attribute	82	39.4	0.9	84	2
25	Enhance VMS MAIL facility	78	37.1	0.5	79	1
33	Keep times synchronized on cluster	73	34.3	0.0	73	0
42	Record attempted usernames on login fail	66	33.3	2.3	71	5
35	Provide cluster-wide management tools	61	28.6	0.0	61	0
5	Support TCP/IP on VMS	56	26.3	0.0	56	0
29	Enhance DCL command recall	55	29.6	3.8	63	8
6	Support wildcards in $GETDVI	54	25.4	0.0	54	0
19	Add description field to file header	53	29.6	4.7	63	10
31	Improve DCL substitution and parsing	52	27.2	2.8	58	6
14	Support standard print file format	48	23.5	0.9	50	2
4	Allow SYSDUMP files not on sys disk	46	21.6	0.0	46	0
16	Improve call interface to job queues	43	20.2	0.0	43	0
32	Enhance the DEASSIGN command	42	21.1	1.4	45	3
1	Provide program access to DECnet config	42	19.7	0.0	42	0
52	Provide column editting in EDT	42	23.5	3.8	50	8
12	Provide a memory-disk for VMS	39	22.1	3.8	47	8
45	Provide ACL protection on printers	34	16.0	0.0	34	0
36	Provide DCL facility like NCP TELL	33	15.5	0.0	33	0
43	Allow security alarms to be sent to file	31	15.0	0.5	32	1
44	Prevent users from reusing passwords	31	22.5	8.0	48	17
41	Improve node authentication in DECnet	29	13.6	0.0	29	0
10	Add utility to list timer queue	29	13.6	0.0	29	0
2	Add system wide event flag clusters	29	13.6	0.0	29	0
46	Fix handling of ACL's on installed image	28	13.6	0.5	29	1
18	Make SWAPPER multi-threaded	26	12.2	0.0	26	0
50	Allow EDT to set tabs in any column	26	14.6	2.3	31	5
3	Improve SMG support of hardcopy	26	12.2	0.0	26	0
17	The VMS source kit should be cheaper	25	12.7	0.9	27	2
28	Add restart capability to DIRECTORY	24	14.6	3.3	31	7
38	Allow image-controlled file access	24	13.1	1.9	28	4
54	Provide overwrite mode in EDT	24	14.1	2.8	30	6
55	Provide nesting depth counts in FORTRAN	23	10.8	0.0	23	0
24	Increase limit on foreign command size	22	10.3	0.0	22	0
48	Better control of operator logins	22	12.2	1.9	26	4
7	No trailing commas in sys. svc. calls	22	13.6	3.3	29	7
40	Add DECnet End-to-end encryption	21	10.8	0.9	23	2
60	Improve debugger support for MACRO	21	9.9	0.0	21	0
27	Improve SHOW PROCESS/CONTINUOUS	20	9.9	0.5	21	1
37	Support CI based intertask transfers	19	9.9	0.9	21	2
34	Support cluster-wide spooled devices	18	8.5	0.0	18	0
11	Default to OWNER:RWED on CREATE/DIR	18	23.9	15.5	51	33
58	Provide DO-UNTIL in FORTRAN	18	10.3	1.9	22	4
15	Improve SMG input routines	18	8.5	0.0	18	0
13	Flag autoconfig devices in SYSGEN	17	8.5	0.5	18	1
57	Improve data structure support in FORTRA	17	8.0	0.0	17	0
61	Provide LSE support for MACRO	17	8.0	0.0	17	0
56	Provide "64-bit mode" qualifer for FORTR	16	9.9	2.3	21	5

THE TOP 50 SIR's AS RANKED BY 11/730 and 11/725 USERS   Total ballots in this category: 83

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	47	57.8	1.2	48	1
20	Support a "printable" file attribute	37	45.8	1.2	38	1
19	Add description field to file header	36	47.0	3.6	39	3
9	Support linking of terminal sessions	34	45.8	4.8	38	4
25	Enhance VMS MAIL facility	31	38.6	1.2	32	1
33	Keep times synchronized on cluster	29	34.9	0.0	29	0
42	Record attempted usernames on login fail	28	34.9	1.2	29	1
6	Support wildcards in $GETDVI	25	30.1	0.0	25	0
12	Provide a memory-disk for VMS	25	34.9	4.8	29	4
16	Improve call interface to job queues	22	26.5	0.0	22	0
29	Enhance DCL command recall	21	32.5	7.2	27	6
1	Provide program access to DECnet config	21	25.3	0.0	21	0
4	Allow SYSDUMP files not on sys disk	21	25.3	0.0	21	0
35	Provide cluster-wide management tools	19	22.9	0.0	19	0
31	Improve DCL substitution and parsing	18	25.3	3.6	21	3
5	Support TCP/IP on VMS	18	21.7	0.0	18	0
2	Add system wide event flag clusters	17	20.5	0.0	17	0
14	Support standard print file format	16	21.7	2.4	18	2
13	Flag autoconfig devices in SYSGEN	15	19.3	1.2	16	1
44	Prevent users from reusing passwords	15	24.1	6.0	20	5
55	Provide nesting depth counts in FORTRAN	15	18.1	0.0	15	0
38	Allow image-controlled file access	13	15.7	0.0	13	0
17	The VMS source kit should be cheaper	13	18.1	2.4	15	2
10	Add utility to list timer queue	12	15.7	1.2	13	1
43	Allow security alarms to be sent to file	12	15.7	1.2	13	1
24	Increase limit on foreign command size	12	14.5	0.0	12	0
50	Allow EDT to set tabs in any column	12	18.1	3.6	15	3
32	Enhance the DEASSIGN command	12	16.9	2.4	14	2
41	Improve node authentication in DECnet	11	13.3	0.0	11	0
18	Make SWAPPER multi-threaded	11	13.3	0.0	11	0
3	Improve SMG support of hardcopy	11	14.5	1.2	12	1
40	Add DECnet End-to-end encryption	9	12.0	1.2	10	1
52	Provide column editting in EDT	9	16.9	6.0	14	5
45	Provide ACL protection on printers	9	10.8	0.0	9	0
58	Provide DO-UNTIL in FORTRAN	9	13.3	2.4	11	2
60	Improve debugger support for MACRO	9	10.8	0.0	9	0
15	Improve SMG input routines	8	9.6	0.0	8	0
61	Provide LSE support for MACRO	8	9.6	0.0	8	0
28	Add restart capability to DIRECTORY	7	13.3	4.8	11	4
23	Provide better control of SYS$ERROR	7	9.6	1.2	8	1
57	Improve data structure support in FORTRA	7	8.4	0.0	7	0
27	Improve SHOW PROCESS/CONTINUOUS	7	8.4	0.0	7	0
48	Better control of operator logins	7	13.3	4.8	11	4
37	Support CI based intertask transfers	7	9.6	1.2	8	1
34	Support cluster-wide spooled devices	6	7.2	0.0	6	0
49	Add asynchronous data security erase	6	10.8	3.6	9	3
36	Provide DCL facility like NCP TELL	6	7.2	0.0	6	0
46	Fix handling of ACL's on installed image	5	7.2	1.2	6	1
21	Improve RMS control block macros	5	6.0	0.0	5	0
56	Provide "64-bit mode" qualifer for FORTR	4	7.2	2.4	6	2

THE TOP 50 SIR's AS RANKED BY MicroVAX USERS   Total ballots in this category: 144

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	69	48.6	0.7	70	1
25	Enhance VMS MAIL facility	66	46.5	0.7	67	1
33	Keep times synchronized on cluster	59	41.0	0.0	59	0
9	Support linking of terminal sessions	55	42.4	4.2	61	6
20	Support a "printable" file attribute	54	39.6	2.1	57	3
35	Provide cluster-wide management tools	46	31.9	0.0	46	0
4	Allow SYSDUMP files not on sys disk	45	31.3	0.0	45	0
6	Support wildcards in $GETDVI	45	31.3	0.0	45	0
5	Support TCP/IP on VMS	43	29.9	0.0	43	0
29	Enhance DCL command recall	39	29.2	2.1	42	3
1	Provide program access to DECnet config	37	25.7	0.0	37	0
19	Add description field to file header	37	31.9	6.3	46	9
42	Record attempted usernames on login fail	34	29.9	6.3	43	9
14	Support standard print file format	30	22.2	1.4	32	2
16	Improve call interface to job queues	29	20.1	0.0	29	0
12	Provide a memory-disk for VMS	29	22.9	2.8	33	4
2	Add system wide event flag clusters	29	20.1	0.0	29	0
31	Improve DCL substitution and parsing	28	23.6	4.2	34	6
43	Allow security alarms to be sent to file	28	19.4	0.0	28	0
45	Provide ACL protection on printers	26	18.1	0.0	26	0
44	Prevent users from reusing passwords	23	22.9	6.9	33	10
38	Allow image-controlled file access	23	17.4	1.4	25	2
46	Fix handling of ACL's on installed image	23	16.0	0.0	23	0
41	Improve node authentication in DECnet	21	14.6	0.0	21	0
32	Enhance the DEASSIGN command	21	17.4	2.8	25	4
27	Improve SHOW PROCESS/CONTINUOUS	21	15.3	0.7	22	1
40	Add DECnet End-to-end encryption	20	14.6	0.7	21	1
52	Provide column editting in EDT	20	17.4	3.5	25	5
24	Increase limit on foreign command size	18	12.5	0.0	18	0
13	Flag autoconfig devices in SYSGEN	17	12.5	0.7	18	1
55	Provide nesting depth counts in FORTRAN	17	11.8	0.0	17	0
10	Add utility to list timer queue	16	11.1	0.0	16	0
36	Provide DCL facility like NCP TELL	16	11.8	0.7	17	1
18	Make SWAPPER multi-threaded	16	11.8	0.7	17	1
3	Improve SMG support of hardcopy	16	11.1	0.0	16	0
23	Provide better control of SYS$ERROR	15	11.1	0.7	16	1
56	Provide "64-bit mode" qualifer for FORTR	15	13.9	3.5	20	5
37	Support CI based intertask transfers	14	11.1	1.4	16	2
60	Improve debugger support for MACRO	14	9.7	0.0	14	0
17	The VMS source kit should be cheaper	13	15.3	6.3	22	9
15	Improve SMG input routines	12	8.3	0.0	12	0
48	Better control of operator logins	12	10.4	2.1	15	3
61	Provide LSE support for MACRO	12	8.3	0.0	12	0
28	Add restart capability to DIRECTORY	11	13.2	5.6	19	8
58	Provide DO-UNTIL in FORTRAN	11	11.1	3.5	16	5
53	Provide SPAWN command in EDT	10	11.1	4.2	16	6
50	Allow EDT to set tabs in any column	10	9.7	2.8	14	4
34	Support cluster-wide spooled devices	10	6.9	0.0	10	0
49	Add asynchronous data security erase	9	7.6	1.4	11	2
57	Improve data structure support in FORTRA	8	6.3	0.7	9	1

THE SIR's IN THE CATEGORY "VMS INTERNALS" AS RANKED BY ALL USERS

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
9	Support linking of terminal sessions	163	47.3	2.5	172	9
20	Support a "printable" file attribute	138	39.0	1.1	142	4
19	Add description field to file header	102	32.1	4.1	117	15
6	Support wildcards in $GETDVI	95	26.1	0.0	95	0
5	Support TCP/IP on VMS	88	24.2	0.0	88	0
14	Support standard print file format	82	23.4	0.8	85	3
4	Allow SYSDUMP files not on sys disk	82	22.5	0.0	82	0
16	Improve call interface to job queues	75	20.6	0.0	75	0
12	Provide a memory-disk for VMS	67	21.7	3.3	79	12
1	Provide program access to DECnet config	66	18.1	0.0	66	0
2	Add system wide event flag clusters	45	12.4	0.0	45	0
10	Add utility to list timer queue	38	10.7	0.3	39	1
3	Improve SMG support of hardcopy	38	10.7	0.3	39	1
7	No trailing commas in sys. svc. calls	35	13.2	3.6	48	13
18	Make SWAPPER multi-threaded	32	9.1	0.3	33	1
13	Flag autoconfig devices in SYSGEN	32	9.1	0.3	33	1
17	The VMS source kit should be cheaper	32	11.5	2.7	42	10
23	Provide better control of SYS$ERROR	25	7.4	0.5	27	2
15	Improve SMG input routines	24	6.6	0.0	24	0
11	Default to OWNER:RWED on CREATE/DIR	17	20.3	15.7	74	57
21	Improve RMS control block macros	13	3.6	0.0	13	0
8	Provide for filtering of BELL chars	4	5.8	4.7	21	17
22	Add a per-user logical name table	1	6.9	6.6	25	24

THE SIR's IN THE CATEGORY "DCL AND UTILITIES" AS RANKED BY ALL USERS

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
26	Show open files for a process	200	55.2	0.3	201	1
25	Enhance VMS MAIL facility	136	37.6	0.3	137	1
31	Improve DCL substitution and parsing	90	27.2	2.5	99	9
29	Enhance DCL command recall	89	28.3	3.8	103	14
32	Enhance the DEASSIGN command	67	19.5	1.1	71	4
28	Add restart capability to DIRECTORY	46	15.9	3.3	58	12
24	Increase limit on foreign command size	41	11.3	0.0	41	0
27	Improve SHOW PROCESS/CONTINUOUS	41	11.5	0.3	42	1
30	Support numeric UIC's in SHOW SYSTEM	20	7.7	2.2	28	8

THE SIR's IN THE CATEGORY "VAXClusters" AS RANKED BY ALL USERS

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
33	Keep times synchronized on cluster	121	33.5	0.3	122	1
35	Provide cluster-wcde management tools	105	28.8	0.0	105	0
36	Provide DCL facility like NCP TELL	51	14.3	0.3	52	1
34	Support cluster-wide spooled devices	35	9.6	0.0	35	0
37	Support CI based intertask transfers	25	7.4	0.5	27	2

THE SIR's IN THE CATEGORY "SECURITY" AS RANKED BY ALL USERS

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
42	Record attempted usernames on login fail	118	35.7	3.3	130	12
44	Prevent users from reusing passwords	63	25.0	7.7	91	28
45	Provide ACL protection on printers	59	16.2	0.0	59	0
43	Allow security alarms to be sent to file	56	15.7	0.3	57	1
38	Allow image-controlled file access	48	15.4	2.2	56	8
46	Fix handling of ACL's on installed image	47	13.2	0.3	48	1
41	Improve node authentication in DECnet	41	11.3	0.0	41	0
48	Better control of operator logins	39	12.9	2.2	47	8
40	Add DECnet End-to-end encryption	34	9.9	0.5	36	2
49	Add asynchronous data security erase	23	8.5	2.2	31	8
39	Implement mandatory security controls	6	7.1	5.5	26	20
47	Make password validation user callable	-13	5.5	9.1	20	33

THE SIR's IN THE CATEGORY "LANGUAGES AND TOOLS" AS RANKED BY ALL USERS

SIR Nr.	SIR Description	Total Votes	% Ballots FOR	% Ballots AGAINST	Total FOR	Total AGAINST
52	Provide column editting in EDT	73	23.9	3.8	87	14
55	Provide nesting depth counts in FORTRAN	47	12.9	0.0	47	0
54	Provide overwrite mode in EDT	43	14.6	2.7	53	10
50	Allow EDT to set tabs in any column	43	15.1	3.3	55	12
56	Provide "64-bit mode" qualifer for FORTR	40	12.4	1.4	45	5
60	Improve debugger support for MACRO	33	9.1	0.0	33	0
53	Provide SPAWN command in EDT	30	12.6	4.4	46	16
58	Provide DO-UNTIL in FORTRAN	28	10.2	2.5	37	9
51	Provide window support in EDT	27	11.3	3.8	41	14
61	Provide LSE support for MACRO	24	6.6	0.0	24	0
57	Improve data structure support in FORTRA	23	6.9	0.5	25	2
62	Provide word count option on RUNOFF	9	2.7	0.3	10	1
59	Provide /INCLUDE=ALL on LINK command	3	1.1	0.3	4	1

VAX System SIG Committee List

As of January 8, 1986

Osman K. Ahmad - Large Systems Integration Working Group
Association of American Railroads
Technical Center, Research and Test Department
3140 South Federal Street
Chicago, IL 60616

Joe Angelico - Assistant Symposium Coordinator
US Coast Guard CCGD8(DT)
Hale Boggs Federal Building
500 Camp Street, New Orleans, LA 70130

Elizabeth Bailey - Volunteer Coordinator
222 CEB
Tennessee Valley Authority
Muscle Shoals, AL 35660

June Baker - Advisor
Computer Sciences Corporation
6565 Arlington Boulevard
Falls Church, VA 22046

Joe L. Bingham - Librarian
Mantech International
2320 Mill Road
Alexandria, VA 22314

Bob Boyd - Commercial Working Group
GE Microelectronics Center
MS 2P-04
Post Office Box 13409
Research Triangle Park, NC 27709

C. Douglas Brown - Security
Sandia Labs
Division 2644
P.O. Box 5800
Albuquerque, NM 87185

Jim Caddick - VAXcluster
General Datacom
Strait Turnpike
Middlebury, CT 06762-1299

Jack Cundiff - Symposium Coordinator
Horry-Georgetown
Post Office Box 1966
Conway, SC 29526

Tom Danforth - Handout Editor
Woods Hole Oceanographic Institute
Woods Hole, MA 02543

Jim Downward - Migration and Host Development, VAXintosh Working Group
KMS Fusion Incorporated
3941 Research Park Drive
Ann Arbor MI 48106

Jane Furze - Campground
3830 West Cochise
Phoenix, AZ 85064

Dennis Frayne - Real Time/Process Control Working Group
McDonnell Douglas
5301 Bolsa Avenue
Huntington Beach, CA 92646

Carl E. Friedberg - Internals Working Group
In House Systems
165 William Street
New York, NY 10038

Don Golden - Communications Committee Representative
c/o Shell Oil Company
Westhollow Research Center
Post Office Box 1380, Room D2132
Houston, TX 77001

Gary Grebus - System Improvement Request
Battelle Columbis Labs
Room 11-6011
505 King Avenue
Columbus, OH 43201-2693

B. Hancock - Network Working Group
Dimension Data Systems, Incorporated
2510 Limestone Lane
Garland, TX 75040
(214) 495-7353

Jeffrey S. Jalbert - Historian
J C C
Post Office Box 381
Granville, OH 43023
614-587-0157

Ray Kaplan - MicroVAX Working Group
Pivotal Incorporated
6892 East Dorado Court
Tucson, AZ 85715

Lawrence J. Kilgallen - Newsletter Editor
Box 81, MIT Station
Cambridge, MA 02139-0901

Margaret Knox - Chair
Computation Center
University of Texas
Austin, Texas 78712

Art McClinton - Advisor
MITRE
1820 Dolley Madison Boulevard
McLean, VA 22102

Ross W. Miller - Vice Chair and Working Group Coordinator
Online Data Processing, Inc.
N 637 Hamilton
Spokane, WA 99202

Eugene Pal - Multiprocessor Working Group
US Army
CAORA (ATOR-CAT-C)
Fort Leavenworth, KA

Susan Rehse - System Management Working Group
Lockheed Missiles
3251 Hanover Street
Palo Alto, CA 94301-1187

Bob Robbins - Advisor
Array Computer Consultants
5364 Woodvale Drive
Sarasota, FL 33582

Larry Robertson - Real Time/Process Control Working Group
Bear Computer Systems Inc.
5651 Case Avenue
North Hollywood, CA

David Schmidt - LUG Coordinator, Hardware Working Group
Management Sciences Associates
5100 Centre Avenue
Pittsburgh, PA 15232

Al Siegel - Advisor
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201-2693

D. Slater - Artificial Intelligence Working Group
Institute for Defense Analysis
1801 North Beavregard Street
Alexandria, VA 22314

INPUT/OUTPUT

A SIG Information Interchange

A form for INPUT/OUTPUT submissions is available at the back of the issue.

INPUT/OUTPUT 492

Caption: Stand-alone disk eraser

Message: I need software/procedures that will provide the means to write over every block on a disk with 3 different patterns. This software must be stand-alone and must be able to erase an RA81, RA60 and RM05. It also must be bootable on a VAX 11/780 or 11/750.

Contact: Gary Rosenfeld
Raytheon, Department 9283
6380 Hollister Avenue
Goleta, CA 93117
Telephone (805) 967-5511 x-3833

Date: January 10, 1986

INPUT/OUTPUT 493

Caption: Using incoming modems for outgoing - reply to I/O # 477

Message: This can be accomplished easily under VMS V4.n as follows:

1. Set the VAX/Modem interface speed appropriately (i.e., SET TERMINAL/PERMANENT/SPEED=1200 TXA0:)
2. Set Modem for desired speed (i.e., for DF03's push in HS switch for 1200 baud)
3. Attach hardwired terminal port to modem port using SET HOST command (i.e., SET HOST/DTE TXA0:).
4. Depress DATA/TALK switch on DF03 and dial the remote system

5. When you hear the remote's carrier tone, release the DATA/TALK switch and hang up.

You are now connected. If your modem has autodial, you can modify steps 4 & 5 to take advantage of this feature.

Contact: James E. English
Agway Incorporated
Research and Development
Box 4933
Syracuse, NY 13221
Telephone 315-477-6380

Date: January 13, 1986

Pageswapper Editor's Note

Remember that if Jane had dial-out access to a line that Fred uses for dial-in access Jane has a chance of stealing Fred's password. Terminal lines used for bi-directional dialing should be protected with ACLs and limited to only a very few highly trusted dial-out users.

INPUT/OUTPUT 494

Caption: Greek/Math characters for HP 7550 plotter

Message: Does anyone have a library of Greek and/or Mathematical symbols which can be used by the Hp 7550 plotter?

Contact: Chris Westerfeldt
Duke University Physics Department
Duke Station
Durham, NC 27706
Telephone (919) 684-8271

Date: January 14, 1986

INPUT/OUTPUT 495

Caption: Microprogramming Tools for the VAX

Message: DEC used to sell a product for writing to the WCS (writable control store) on the VAX 11/780. The product number was QE109-CX. I would appreciate talking to anyone who bought and used the product.

Contact: Harry Goldenbloome
Raytheon Company
1847 West Main Road
Portsmouth, RI 02871-1087
Telephone (401) 847-8000 x4087

Date: January 15, 1986

INPUT/OUTPUT 496

Caption: Word Processing similar to Wang

Message: We have a Wang Word Processor and we plan to phase it out. In the meantime, we are looking for VAX word processing software that is similar to the Wang since we prefer not to train our secretaries again.

Does anybody know of one?

Contact: Tarif Halabi
Post Office Box 61092
Amaroussion 15 110
Athens, Greece
Telephone 6829200

Date: January 20, 1986

INPUT/OUTPUT 497

Caption: Using incoming modems for outgoing - reply to I/O # 477

Message: We use the Hayes smartmodem 2400 for both incoming and outgoing calls. If your modem has dial-out capability, you should be able to use the SET HOST/DTE command. See the standard DEC documentation.

Contact: Danny Brooks
Systems Analyst
Box 508

Medical College of Virginia
Richmond, VA 23298
Telephone (804) 786-9026

Date: January 27, 1986

Pageswapper Editor's Note

Remember that if Jane had dial-out access to a line that Fred uses for dial-in access Jane has a chance of stealing Fred's password. Terminal lines used for bi-directional dialing should be protected with ACLs and limited to only a very few highly trusted dial-out users.

INPUT/OUTPUT 498

Caption: DECnet Security Information

Message: I am trying to find any reports or papers concerning security considerations when using DECnet. I am particularly interested in security issues with regard to Department of Defense guidelines.

Contact: Rich Stoehe
Mitre Corporation
1257 Lake Plaza
Colorado Springs, CO 80916
Telephone (303) 576-2602

Date: January 29, 1986

INPUT/OUTPUT 499

Caption: SET HOST/DTE TTxx/DIAL=(NUMBER=#,MODEM=DF224)

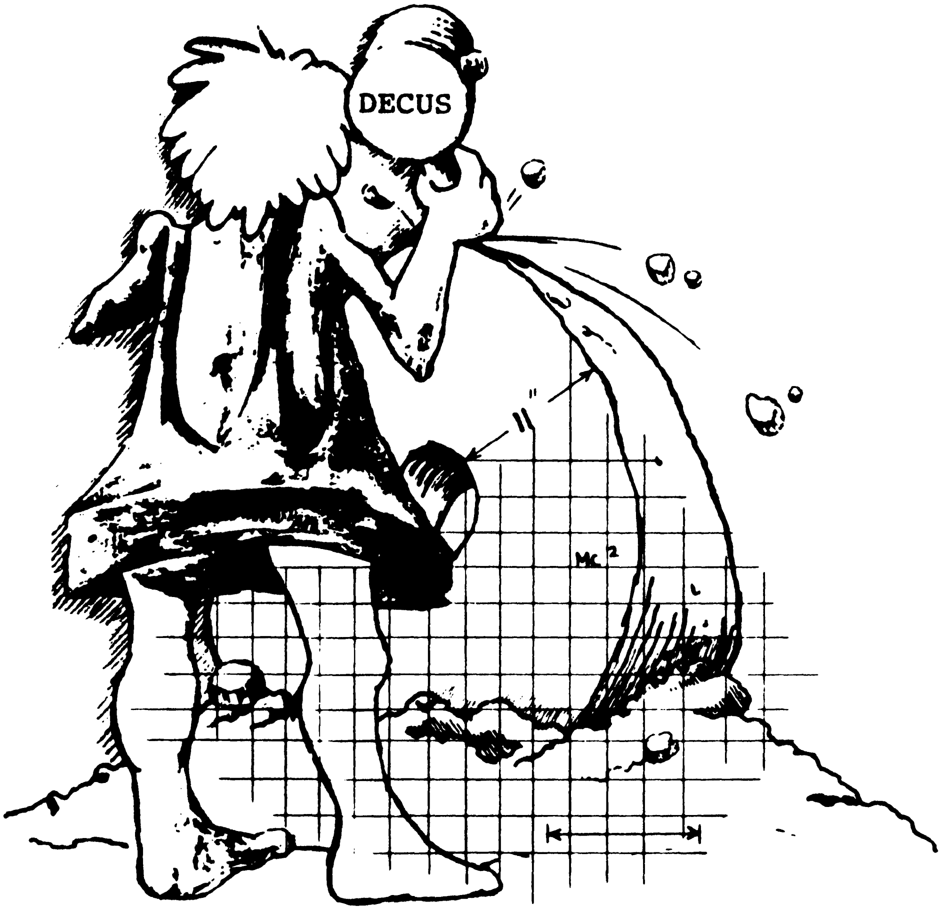
Message: Has anyone tried using a DF224 (Scholar) in a SET HOST/DTE? Does anyone have a working DTEDF224 routine (with some instructions for installing it)?

Contact: Alan Zonderman
Gerontology Research Center
National Institute on Aging
Francis Scott Key Medical Center
Baltimore, MD 21224

Date: April 11, 1986

DECUS

LIBRARY



LIB

DECUS PROGRAM LIBRARY

NEW LIBRARY PROGRAMS AVAILABLE FOR THE RAINBOW SERIES OF COMPUTERS

DECUS NO: RB-105, **Title:** SEDT: EDT/WPS Screen Editor for MS/DOS, **Version:** V2.0 February 1986

Author: Digital Equipment Corporation, **Operating System:** MS/DOS, **Source Language:** Various, **Memory Required:** 256KB **Keywords:** Editors

Abstract: SEDT is an EDT-like editor with totally compatible versions available for MS/DOS, CP/M and VMS, this being the MS/DOS version.

The program supports an EDT or WPS keypad layout, multiple buffers, windows, and journaling, with support for the keyboard used by current Digital Equipment Corporation terminals and PCs.

SEDT under MS/DOS could be the best screen editor available for the Rainbow running MS/DOS at the present time.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: RB-108, **Title:** AME86 - Application Migration Executive for CP/M to MS/DOS, **Version:** V0.7-1

Author: Digital Equipment Corporation, **Operating System:** MS/DOS, **Source Language:** Various, **Keywords:** Utilities - MS/DOS

Abstract: AME86 is an extremely useful program which allows you to run most CP/M-86 programs (.CMD files) under MS/DOS on the Rainbow.

The program simulates the CP/M environment, translating most CP/M system functions into the equivalent MS/DOS functions, achieving performance similar to that of the same program running under CP/M.

While there is no claim that your application will work, tests have been performed with such products as RED, MBASIC-86, WordStar V3, ZORK and DBASE, and applications developed with MWC-86.

On the other hand, Multiplan still has some problems with this version, so you will need to test your particular application.

Overall though, this is an extremely useful offering, letting you migrate your favorite programs to MS/DOS without time consuming conversion.

Restrictions: Needs MS/DOS V2.05 or higher.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA)

NEW LIBRARY PROGRAMS AVAILABLE FOR CP/M

DECUS NO: CPM-268, **Title:** DOSFLX: CP/M MS/DOS File Exchange, **Version:** January 1986

Operating System: CP/M, **Source Language:** Various, **Keywords:** File Management, Utilities - CP/M

Abstract: Written specifically for CP/M on the Rainbow, dsSFLX lets you transfer files between MS/DOS or PC/DOS disks and CP/M disks.

DOSFLX supports Rainbow MS/DOS diskettes, Rainbow internal MS/DOS partitions, and IBM single sided 8 or 9 sector-per-track PC/DOS diskettes, keeping in mind that IBM diskettes, once written to by the Rainbow, may not be re-read on an IBM PC.

The program is extremely easy to use, completely self-documenting with a good HELP facility and very useful.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA)

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PDP-11 COMPUTER FAMILY

DECUS NO: 11-SP-89, **Title:** IAS SIG LIBRARY, **Version:** Library-B

Author: Various, **Submitted by:** Michael Robitaille, Grumman-CTEC, Inc., McLean, VA, **Operating System:** IAS, V3.0, 3.1, 3.2, **Source Language:** FORTRAN IV, FORTRAN IV-PLUS, FORTRAN 77, MACRO-11, **Keywords:** Library Tapes - IAS

Abstract: This package represents the current state of the IAS SIG LIBRARY. It contains a multitude of software programs, help files, and function libraries submitted by IAS users over the years. This library is under continuous management by the IAS SIG and will be updated as additional programs are submitted and the existing programs are tested and evaluated. "Library_B" contains 2,849 files in 87 directories requiring 45,718 blocks. Among the contents in the tape are:

- SRX - an enhanced IAS version of SRD;TDS. A suite of Task Dump Serviceswith great utility during program development and maintenance
- ATS - a utility for displaying the current active tasks in priority order on a VT100
- HLP files for most IAS commands (as of V3.0) and for some of the SIG Library
- Several disassemblers and debuggers including BUG, a full screendebugger-disassembler for the VT52
- Several games including ADVENTure and MTREK, a multi-player Star Trek
- IAS system accounting packages
- GTC - a Get Terminal Characteristics utility and much more

Notes: Developed for IAS, may work for other PDP Operating Systems.

Restrictions: The bulk of the software is untested by the submitter. Documentation for some of the programs and utilities are sketchy or absent. Objects are not supplied when source code exists. Since some source is in FORTRAN, a FORTRAN compiler is necessary for those utilities. All privileged programs assume IAS.

Media (Service Charge Code): 2400' Magnetic Tape (PC), **Format:** BRU

DECUS NO: 11-SP-90, **Title:** Symposium Tape from the RSX SIG, Fall 1985, Anaheim, **Version:** V1, March 1986

Submitted by: Glenn C. Everhart, Ph.D., **Operating System:** IAS, RSX-11D, RSX-11M, RSX-11M-PLUS, VAX/VMS. **Source Language:** BASIC-PLUS, C, FORTRAN IV, FORTRAN IV-PLUS, FORTRAN 77, VAX 11 FORTRAN, MACRO-11, PASCAL, **Keywords:** Data Communications, Programming Languages, Spreadsheet, Symposia Tapes - RSX-11, Utilities - RSX-11

Abstract: This tape contains the submissions from the Fall 1985 symposium in Anaheim for the RSX SIG collection. It is available in either BRU or VMS/BACKUP format. To obtain the VMS/BACKUP version, order DECUS No. V-SP-50.

The tape contains numerous packages, many of interest to VMS, RSTS and RT-11 sites as well as RSX sites. The top level documents for the tape are in directory [300,1] on tape. Start with BEGIN85B.DOC for further details.

Highlights of the tape:

- Complete update of all DECUS C including the most complete toolkit yet. Support for all OSs is included. This release is unsupported but contains major upgrades to improve compatibility of DECUS C with the ANSI version of the full language. Also includes a greatly improved GREP from Tom Shinal of the RT-11 SIG.
- Update to Bonner RUNOFF (DSR superset. VMS Native mode supported also. Aids to convert MACRO-11 to MACRO-32 are used and included in this kit.)
- KERMIT-11 update and a new KERMIT for IBMPC which emulates VT100, supports XMODEM, KERMIT, menus, scripts, autodialing and much more. Also communications support for systems with the half duplex TT driver.
- Faster and more powerful version of AnalytiCalc spreadsheet with additional string functions.
- Object disassemblers for RSX.
- Spelling checkers, menu systems and a complete forms management system (in FORTRAN).
- Complete revision of TED fullscreen editor for RSX, RT-11, VMS with sources in C. This editor has many powerful word processing features and a TECO-like "see-all" mode. It is faster than TECO, however.
- Graphics support systems including terminal viewgraph generators.

Numerous additional system-use aids for RSX are also included. There are several drivers, some VAX/RSX V2 AME bug work-arounds, file listers, file finders, a way to use F77 virtual arrays with I/D space, batch systems, VT200 setup systems, VTL updates, command line editors and more. The tape is one which has something for any PDP-11 or VAX system and contains much information which has never appeared elsewhere.

About 60,000 blocks are required to hold the full tape contents.

Complete sources not included.

Media (Service Charge Code): 2400' Magnetic Tape (PS), **Format:** BRU

DECUS NO: 11-824, **Title:** DISCRETE, **Version:** December 1985

Author: S. W. Provencher, Max-Planck-Inst. f. biophys. Chemie, **Submitted by:** T. Hoshiko, Case Western Reserve University, **Operating System:** RT-11 V5.01, **Source Language:** FORTRAN IV, **Memory Required:** 24K, **Keywords:** Mathematical

Abstract: DISCRETE, originally written for the IBM 370, is a FORTRAN IV program for the automatic analysis of data represented by the sum:

Y(k) = Summation [A(j)*EXP(-LAM(j)*T(k))]

for j = 0 to n and k = 1,2, ... , n, with n {= 9. Provision can be made for an unknown baseline component A(0) with LAM(0) = 0. Only the raw data (i.e., the Y(k) and T(k) are input; no potentially biased initial guesses at the A(j), LAM(j) or the number of exponential terms, n, are needed or allowed. Transforms of the data are used to obtain good starting estimates for least squares analyses of the data. DISCRETE is based on a fast and accurate method, making direct use of the knowledge that the data is being represented by a discrete sum (rather than an integral over a continuous distribution of exponentials).

DISCRETE is large for a minicomputer and the original program required about 180Kbytes. Since the RT-11 single-job monitor accomodates only 45 Kbytes, substantial changes were required. The double precision version of DISCRETE had to be used requiring more space and time for execution so that this RT-11 version can accomodate only 50 points.

Documentation available in hardcopy only.

Media (Service Charge Code): User's Manual (EA), One RX01 Diskette (KA), **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

NEW LIBRARY PROGRAMS AVAILABLE FOR THE VAX/VMS FAMILY OF COMPUTERS

DECUS NO: V-SP-50, **Title:** Symposium Tape from the RSX SIG, Fall 1985, Anaheim in VMS/BACKUP, **Version:** V1, March 1986

Submitted by: Glenn Everhart, Ph.D., **Operating System:** IAS, RSX-11D, RSX-11M, RSX-11M-PLUS, VAX/VMS, **Source Language:** BASIC-PLUS, C, FORTRAN IV, FORTRAN IV-PLUS, FORTRAN 77, VAX-11 FORTRAN, MACRO-11, PASCAL, **Keywords:** Data Communications, Programming Languages, Spreadsheet, Symposia Tapes - RSX-11, Utilities - VMS

Abstract: This tape contains the submissions from the Fall 1985 symposium in Anaheim for the RSX SIG collection. It is available in either BRU or VMS Backup formats. To obtain the BRU version, order DECUS No. 11-SP-90.

The tape contains numerous packages, many of interest to VMS, RSTS and RT-11 sites as well as RSX sites. The top level documents for the tape are in directory [300,1] on tape. Start with BEGIN85B.DOC for further details.

Highlights of the tape:

- Complete update of all DECUS C including the most complete toolkit yet. Support for all OSs is included. This release is unsupported but contains major upgrades to improve compatibility of DECUS C with the ANSI version of the full language. Also includes a greatly improved GREP from Tom Shinal of the RT-11 SIG.
- Update to Bonner RUNOFF (DSR superset. VMS Native mode supported also. Aids to convert MACRO-11 to MACRO-32 are used and included in this kit.)
- KERMIT-11 update and a new KERMIT for IBMPC which emulates VT100, supports XMODEM, KERMIT, menus, scripts, autodialing and much more. Also communications support for systems with the half duplex TT driver
- Faster and more powerful version of AnalytiCalc spreadsheet with additional string functions.
- Object disassemblers for RSX.
- Spelling checkers, menu systems and a complete forms management system (in FORTRAN).
- Complete revision of TED fullscreen editor for RSX, RT-11, VMS with sources in C. This editor has many powerful word processing features and a TECO-like "see-all" mode. It is faster than TECO, however.
- Graphics support systems including terminal viewgraph generators.

Numerous additional system-use aids for RSX are also included. There are several drivers, some VAX/RSX V2 AME bug workarounds, file listers, file finders, a way to use F77 virtual arrays with I/D space, batch systems, VT200 setup systems, VTL updates, command line editors and more. The tape is one which has something for any PDP-11 or VAX system and contains much information which has never appeared elsewhere.

About 60,000 blocks are required to hold the full tape contents.

Complete sources not included.

Media (Service Charge Code): 2400' Magnetic Tape (PS) **Format:** VMS/BACKUP

DECUS NO: VAX-162 **Title:** DR11-C VMS Device Driver **Version:** V1.3, July 1985

Submitted by: Digital Equipment Corporation **Operating System:** VAX/VMS V3.6 to V4.X **Source Language:** MACRO-32 **Memory Required:** Approx. 26KB **Keywords:** Device Handlers

Abstract: The DR11-C VMS Device Driver is a general purpose, generic driver for the DR11-C parallel I/O, non-DMA interface of a user's device to the UNIBUS on VAX/VMS systems.

This device driver supports the QIO functions READxBLK, WRITExBLK, SETMODE, SENSEMODE, SETCHAR, SENSECHAR and Type B interrupts from the DR11-C.

Assoc. Documentation: VMS Documentation Set

Media (Service Charge Code): 600' Magnetic Tape (MA) Format: VAX/ANSI

DECUS NO: VAX-163 **Title:** Escape From Manhattan **Version:** V1.1/February 1986

Submitted by: Richard Cook, Baton Rouge, LA

Operating System: VAX/VMS V3.6 **Source Language:** VAX-11 FORTRAN Keywords: Games

Abstract: Escape is an all-text computer role-playing game of adventure and weaponry, involving the one player's skill, attentiveness, thinking and luck. Based on the movie

"Escape from New York", the player takes on the character of Fox Brisskin, a man condemned to life in the newest U.S. maximum security prison, Manhattan Island. The year is 1997. Just before boarding the helicopter to Manhattan, the prison's warden offers you total reprieve of all crimes, if you can perform one duty---get the President out of Manhattan safely. The President's plane was hijacked and crashed inside the walls of Manhattan. He has survived the accident by fate and is now being held hostage by the inmates. He is due to speak with Russia and China at a summit meeting in 24 hours. If he doesn't show, global nuclear war becomes history.

Complete with introduction and menu driven options, ESCAPE pits the player against murderous (and sometimes helpful) thugs, simple and complex puzzles, a variety of deadly weapons, and a ticking death clock. Experience points are awarded for victorious battles, but health points must be maintained to stay alive. Each game is new and challenging, randomly determined at the beginning, for continued pleasure through the millennium.

A DOC file is also included with helpful playing hints.

Restrictions: Sometimes at beginning of game, the random determinator part of the initialization gets stuck in a long loop due to certain restrictions not being met. Simple patience will solve this problem.

Sources not included.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VAX/ANSI

REVISIONS TO LIBRARY PROGRAMS

DECUS NO: 20-184 **Title:** 2022 **Version:** 116B through 117B

Submitted by: David L. Wodelet, Strathcona County, Sherwood Park, Alberta, Canada T8A 3W7 **Operating System:** TOPS-20 release 6.1 **Source Language:** MACRO-10 **Software Required:** 1022 from Software House **Keywords:** Tools-Applications Development

Abstract: 2022 is a TOPS-20 front-end command parser for the System 1022 data base management system from Software House. Through use of TOPS-20 COMND% jsys, 2022 provides escape recognition and the "?" help feature for nearly all 1022 commands. The only commands NOT implemented in 2022 are those commands specific to TOPS-10 or those used exclusively within PL1022 or report programs.

Notes: Sources for Software House System 1022 are unavailable.

Changes and Improvements: New code added to support version 117B of 1022 from Software House.

Media (Service Charge Code): 600' Magnetic Tape (MA)

DECUS NO: CPM-103 **Title:** Vol. 3 Various BASIC-E Games and Programs **Version:** March 1986

Author: Various **Operating System:** CP/M **Source Language:** CBASIC, MBASIC **Keywords:** Games

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

Program Descriptions:

ACE.BAS	A game of cards and money.
AMAZE.BAS	Watch the computer build a maze to your specifications.
BAGELS.BAS	Try to guess the number that the computer is thinking.
CANNONS.BAS	Fire a cannon at distant targets.
CHASE.BAS	A game of chase in a high voltage maze.
CRAPS.BAS	Play the game of craps, but don't lose your money.
EUCLID.BAS	Let the computer find the greatest common factor of two integers that you supply.
FIB.BAS	Specify how many "Fibonacci" numbers you want and let the computer do the rest.
HELLO.BAS	Let the gem wizard solve yor problems with sound advice.
LANDER.BAS	Try to land a rocket on the moon, SAFELY!
STARTREK.BAS	Destroy enemy klingons before they get you. This game will run only on CBASIC.
STRIKE9.BAS	Take from a group of nine numbers, but don't leave yourself short or you lose.
WUMPAS.BAS	Catch the WUMPAS, but watch out for his friends.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on a Digital Rainbow Personal Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: CPM-105 **Title:** Vol. 5 BASIC-E Compilers and Interpreters, BASIC-E Programs Continued From Volume 3, Microsoft BASIC Programs **Version:** March 1986
Author: Various **Operating System:** CP/M **Source Language:** CBASIC, MBASICKeywords: BASIC, Compilers, Language Interpreters

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

BIORYTHM.BAS	Predict your physical, emotional and intellectual patterns.
BLKFR12.ASC	Try your luck at the stock market.
DECISION.BAS	Watch the bouncing ball as it makes your decisions.
EDTEXT.BAS	This program allows you to edit and print text data.
FORMAT.BAS	This is a formatter program.
OTHELLO.BAS	Play the game of OTHELLO against the computer.
RADIX.ASC	Let this program do base number conversions for you.
RECOVERY.ASC	Calculates the present value of a non-cash security.
SLOT.ASC	Take your chances on winning the big one.
SORT.ASC	This program sorts characters or numbers into order.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on aa Digital Rainbow Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: CPM-113 **Title:** Vol. 13 BASIC-E/CBASIC and Microsoft BASIC Programs and Games **Version:** March 1986

Author: Various **Operating System:** CP/M **Source Language:** CBASIC, MBASIC **Keywords:** Games

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

Program Descriptions:

NFL.BAS	You coach your football team against the computer.
ROCKET.BAS	You now have the chance to safely land a rocket on the moon.
BAGELS.BAS	Guess the number that the computer is hiding from you.
CHECKERS.BAS	Take the computer on at checkers.
DICE.BAS	Do you dare wager money on dice rolls?
KINGDOM.BAS	Rule a country, but save your life. Game of skill and hope.
TRAP.BAS	Guess a number between 1 and 100.
SWARMS.BAS	Killer bee game.
23MATCH.BAS	Who's going to end up with the last match?
BLACKJAC.BAS	This one isn't a pushover, you might lose your shirt.
WUMPAS.BAS	Try to get the wumpas from within the cave.
CHIEF.BAS	Pick a number, but don't tell the computer.
BIORYME.ASC	Find out your biorhythms for a 30 day period.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on a Digital Rainbow Personal Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: CPM-126 **Title:** Vol. 26 Microsoft BASIC and FORTRAN Games and Utilities **Version:** March 1986

Author: Various **Operating System:** CP/M **Source Language:** CBASIC, MBASIC **Keywords:** Games, Utilities - CP/M

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

Program Descriptions:

BACCRRT.ASC	Real style Vegas casino gambling.
BASEBALL.ASC	Choose the type of pitch and how your batter will swing.
BIRTHDAY.ASC	Let this program tell you how many days old you are.
CHESS.ASC	Here's one you haven't beat yet.
CIVIL WAR	Can you lead your troops to victory? CLOUD-9.ASCFool around with the elevation of a cloud.
CRAPS.ASC	Check this one out, you get to start with $25,000.
CRAZY-8.ASC	Give the game of Crazy Eights a whirl!
GALAXY.ASC	Run the program and let your imagination go as a continuous galaxy prints on your screen.
SWARMS.ASC	You must kill the swarms of killer bees.
WEATHER.ASC	Let this one predict tomorrows's weather.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on a Digital Rainbow Personal Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: CPM-127 **Title:** Vol. 27 Microsoft BASIC Games **Version:** March 1986

Author: Various **Operating System:** CP/MSource Language: CBASIC, MBASIC **Keywords:** Games

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

Program Descriptions:

ANTONYMS.ASC	The computer picks a word, and you must come up with an antonym of that word.
FOOTBALL.ASC	A very challenging game of football.
GOLF.ASC	Choose your type of club according to the conditions.
GREEKRTS.ASC	Test your knowledge of Greek terms from the list.
HANGMN-1.ASC	Guess the word or face the consequence.
HIDESEEK.ASC	Try to find four other players in one 10x10 grid.
IQUEEN.ASC	Move against the computer in a simulated game of chess.
MASTERMD.ASC	Guess a five-digit number that the computer is hiding.
MAZE.ASC	Watch the computer print out a maze, built to your specs.
MEMBRAIN.ASC	You become a cell membrain.
ROULETTE.ASC	An imitation of the famous French gambling game.
SHOP.ASC	Buy grocery items using your computer.
SNOOPY.ASC	Watch as a picture of snoopy appears on your screen.
SYNONYMS.ASC	Play a game of synonyms.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on a Digital Rainbow Personal Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: CPM-154 **Title:** Vol. 54 Games **Version:** March 1986

Author: Various **Operating System:** CP/M **Source Language:** CBASIC, MBASIC **Keywords:** Games

Abstract: All programs on this diskette require a RAINBOW microcomputer and the Microsoft MBASIC software package. All of these programs should run on MBASIC. Most of these programs should also work on CBASIC.

Program Descriptions:

ARITH.BAS	Arithmetic questions according to your grade level.
BLKJAC.BAS	Try your luck at Blackjack, Las Vegas style!
BOMBER.BAS	You are a pilot in WW II.
BUG.BAS	Build your bug before the computer builds one.
CHASE.BAS	Your must safely exit from a high voltage maze!
CHOMP.BAS	Have fun eating a giant cookie.
CRAPS.BAS	This is the game of craps, watch the computer roll for you.
DEFUSE.BAS	Defuse a bomb in a government building!
DIAMND.BAS	Watch as the computer prints pretty diamond patterns.
FISHING.BAS	Try to find where the most fish are located.
FOOTBL.BAS	Select from a number of offensive and defensive plays.
GUNNER.BAS	Fire a field artillery weapon at the enemy.
HOCKEY.BAS	Decide the types of shots to take in this fast paced game.
HORSES.BAS	Place your bets then watch the race.
INTEREST.BAS	Determine your dividends on the money you invest.
KING.BAS	You are the premier of a small communist island.
LITQZ.BAS	Test your knowledge of children's literature.
MUGWMP.BAS	Find four mugwumps hidden in a 10x10 grid.
PRIME.BAS	Print all the prime numbers from 1 - 10,000.
REVRSE.BAS	Arrange a list of numbers (1-9) in numerical order with clues from the computer.
SPORTS.BAS	Answer 10 sports questions from the year 1974.
STOCK.BAS	Play the game of the stock market.
TDRILL.BAS	A program to start or improve your typing skills.
TENNIS.BAS	Test your knowledge of the tennis world.
TICTAC.BAS	Play on a 9x9 grid against the computer.
TRAP.BAS	Guess the number the computer is hiding from you.
TTTOE.BAS	Tic tac toe against the computer.
TVPLOT.BAS	Watch as the computer will come up with plots for TV shows.
WEKDAY.BAS	Facts about dates of interest to you.
WISHES.BAS	You and the computer will write a wish poem.

Notes: As part of a General International Area (GIA) Program Library Committee project, the programs on this diskette have been certified as runnable on a Digital Rainbow Personal Computer.

Media (Service Charge Code): One RX50 Diskette (JA)

DECUS NO: VAX-6 **Title:** SPICE 3A6 **Version:** 3A6, February 1986

Submitted by: Digital Equipment Corporation **Operating System:** VAX/VMS V 4.3 or later **Source Language:** C **Memory Required:** 3MB **Software Required:** C Compiler **Keywords:** Circuit Simulation

Abstract: SPICE3 is a general-purpose circuit simulation program for nonlinear dc, nonlinear transient, and linear ac analyses. Circuits may contain resistors, capacitors, inductors, mutual inductors, independent voltage and current sources, four types of dependent sources, transmission lines, and the four most common semiconductor devices: diodes, BJTS, JFETS, and MOSFETS.

The SPICE3 version is based directly on SPICE 2G.6. While SPICE3 is being developed to include new features, it will continue to support those capabilities and models which remain in extensive use in the SPICE2 program.

Changes and Improvements: Completely rewritten in C language; program modules allow easy integration of new user-written models; new interactive user interface included.

Assoc. Documentation: Additional program references available from the Industrial Liason Program office at University of California, Berkeley.

Restrictions: U.S. Government export regulations prohibit the distribution of this program outside of the United States without the appropriate export licenses.

Media (Service Charge Code): User's Manual (EC), 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

HOW TO SUBMIT TO A SPECIFIC SECTION OF THE NEWSLETTER

The following is a listing of the Newsletter Editors with their addresses and phone numbers. All submissions to the newsletter should be submitted directly to the appropriate Editor.

ARTIFICIAL INTELLIGENCE
Terry Shannon
160 State Street
Boston, MA 02109
(617) 367-7190

BUSINESS APPLICATIONS
Thomas Byrne
L. Karp & Sons
1301 Estes
Elk Grove, IL 60007
(312) 593-5705

DATA MANAGEMENT SYSTEMS
Russ Poisson
Seed Software Corp.
2121 Eisenhower Avenue
Alexandria, VA 22314
(703) 783-4944

DAARC
Ellen Reilly
William H. Rorer
500 Virginia Drive
Ft. Washington, PA 19034
(215) 628-6547

GRAPHICS APPLICATION
Michael Anton
P.O. Box 591293
Houston, TX 77259-1293
(713) 928-4838

IAS
John Ross Roman
McDonnell Douglas
600 McDonnell Blvd.
Hazelwood, MO 63042
(314) 234-0984

LARGE SYSTEMS
Michael Joy
1st Church of Christ
Scientist
Boston, MA 02115
(617) 262-2300 x 3903

NETWORKS
Vicki Hancock
2510 Limestone Lane
Garland, TX 75040
(214) 495-7353

PERSONAL COMPUTER
Caroline Mack
9007 Mears Street
Fairfax, VA 22031
(703) 280-4404
[Upload submissions to
Wash-A-Rug Fido
(703) 359-6549]

RSX
Dominic DiNollo
Loral Electronics
Engineering Computer Center
Ridge Hill
Yonkers, NY 10710
(914) 968-2500 x2210

SITE MANAGEMENT & TRAINING
Gregory Brooks
Washington University
Behavior Research Lab.
1420 Gratton St.
St. Louis, MO 63104
(314) 241-7600 x 257

VAX SYSTEMS
Larry Kilgallen
c/o DECUS Office
219 Boston Post Road. (BP02)
Marlboro, MA 01752

APL
Doug Bohrer
Bohrer & Company
903 Ridge Road, Suite 3
Wilmette, IL 60091
(312) 251-9449

COMMERCIAL LANGUAGES
Ted Bear
RAMTEK
2211 Lawson Lane
Santa Clara, CA 95950
(408) 988-2211

DATATRIEVE
Donald E. Stern, Jr. c/o
Warner Lambert Company
10 Webster Road
Milford, Ct. 06460
(203) 783-0238

EDUSIG
Fred Bell
Taft College
29 Emmons Park Drive
P.O. Box 1437
Taft, CA 93268
(805) 763-4282

HMS
William Walker
Monsanto Research Corp.
P.O. Box 32 A-152
Miamisburg, OH 45342
(513) 865-3557

LANGUAGES & TOOLS
Alan Folsom Jr.
Fischer & Porter Company
E. County Line Road
Warminster, PA 18974
(215) 674-7154

MUMPS
Janet Berryman
2405 N. Bush
Santa Ana, CA 92706
(714) 953-1025

OFFICE AUTOMATION
Margaret Drake
Univ. of TX Health Science Ctr.
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-6105

RSTS
Bill Hobbs
ComManD. Inc.
6535 E. 82nd St., Suite 102
Indianapolis, IN 46250
(317) 842-5320

RT
Bill Leroy
The Software House, Inc.
2964 Peachtree RDNW #320
P.O. Box 52661
Atlanta, GA 30355
(404) 231-1484

UNISIG
William Toth
Harvard-Smithsonian Ctr. for
Astrophysics
60 Garden Street P353
Cambridge, MA 02138
(617) 495-7181

Bruce Bergman
UserWare International
2235 Meyer Avenue
Escondido, CA 92025-1070
(619) 741-8825

SUBMITTING ARTICLES TO THE HMS SIG NEWSLETTER

The purpose of the HMS SIG Newsletter is to serve as a forum to share information related to DEC hardware with the members of the SIG. As such, the existence of the newsletter is entirely dependent on your contributions. If you have an HHK item, a better or safer way to do something, product news, a tutorial article of general interest, etc., we are interested in publishing it in the newsletter. It is intended that the HMS SIG Newsletter be published at least four times a year.

There are several ways to submit material for the newsletter:

- The Hardware Submission Form in the back of the newsletter can be used for brief items (there is not enough room if you have a lot to say).
- You can send me camera-ready hard-copy (this saves me a lot of typing).
- I will accept submissions on floppys. I can handle RX50's or 8" diskettes (either density, single or double sided). I prefer RT-11 format, if possible, but I can probably handle RSX or VMS stuff somehow. I will return your diskette(s), of course.
- Those of you that have access to DCS can send things to username WALKER. I check DCS daily.
- I am also on CompuServe as "Bill Walker 71066,24".

In any event, if you have anything to submit, send it! If it is a mess, but I can read it, I will get it in the newsletter somehow. Finally, if you have any question about submitting material, call me. My telephone number is listed below.

Contributions can be sent to:

HMS Editor		William K. Walker
DECUS	OR	Monsanto Research Corp.
BP02	==	P.O. Box 32 A-152
249 Northboro Road		Miamisburg, OH 45342
Marlboro, MA 01752		(513) 865-3557

If you need to get something to me quickly, send it to my work address.

DECUS

DECUS SUBSCRIPTION SERVICE
SIGs NEWSLETTERS
U.S. CHAPTER MEMBERS ONLY

As a member of DECUS U.S. Chapter, you are entitled to contribute and subscribe to the DECUS monthly publication, **SIGs Newsletters.** You also have the opportunity to subscribe to the Symposia Proceedings which are a compilation of the reports from various speakers at the U.S. National DECUS Symposia.

- **No Purchase Orders will be accepted.**
- The order form below must be used as an invoice.
- All checks must be made payable to DECUS.
- All orders MUST be paid in full.
- No refunds will be made.
- The address provided below will be used for all DECUS mailings; i.e. Membership, Subscription Service and Symposia.
- SIGs Newsletters Price is for a one-year subscription beginning the month following receipt of payment.

Name ______________________ DECUS Member No. ______________
Company ______________________
Address ______________________

____

City ______________ State __________ Zip __________
Phone ______________________

Subscription Service Offering	**Qty.**	**Unit Price**	**Total**
SIGs Newsletters	______	$35.00	______
Fall '85 Proceedings (FA5)	______	15.00	______
Spring '86 Proceedings (SP6)	______	15.00	______
Fall '86 Proceedings (FA6)	______	15.00	______
Spring '87 Proceedings (SP7)	______	15.00	______
TOTAL COST OF SUBSCRIPTION			$______

☐ MASTERCARD ☐ VISA ☐ DINERS CLUB/CARTE BLANCHE®

— — — — — — — — — — — — — — — — Exp. Date ______________

I understand that there will be no refunds even if I decide to cancel my subscription.

Signature: ______________________

FOR DIGITAL EMPLOYEES ONLY		**FOR DECUS OFFICE ONLY**
Badge No. ______________	CC: __________	Check No. __________
CC Mgr. Name ______________		Bank No. __________
CC Mgr. Signature ______________		Amount $ __________

Subscription Service, DECUS (BP02), 219 Boston Post Road, Marlboro, MA 01752-1850, (617) 480-3418.

DECUS U.S.CHAPTER
APPLICATION FOR MEMBERSHIP

☐ **New Membership** ☐ **Update to current membership profile Current DECUS Member. #** ______

NOTE: PLEASE PRINT CLEARLY OR TYPE!

PLEASE PROVIDE A COMPLETE MAILING ADDRESS, INCLUDE ZIP CODE IN ACCORDANCE WITH POSTAL REGULATIONS FOR YOUR LOCALITY.

ARE YOU AN EMPLOYEE OF DIGITAL EQUIPMENT CORPORATION? ☐ **YES** ☐ **NO**

Name: ______________________________
(first) (Middle Intial) (Last/Family Name)

Company: ______________________________

Address: ______________________________

____

____

City/Town: ______________ State: ______________ Zip: ________

____

Telephone: Home ( ) ______________ Work ( ) ______________

HOW DID YOU LEARN ABOUT DECUS? Please check applicable item.

1 ☐ ANOTHER DECUS MEMBER
2 ☐ SYMPOSIA
8 ☐ DECUS CHAPTER OFFICE
10 ☐ DIGITAL STORE
4 ☐ DIGITAL SALES
5 ☐ HARDWARE PACKAGE
6 ☐ SOFTWARE PACKAGE
12 ☐ ADVERTISING
13 ☐ LOCAL USER GROUP
14 ☐ SPECIAL INTEREST GROUP
7 ☐ SOFTWARE DESPATCH
(DIGITAL Newsletter)

DO YOU WISH TO BE INCLUDED IN MAILINGS CONDUCTED BY DIGITAL (for Marketing purposes etc.?) ☐ Permission
☐ Refusal

TYPE OF DIGITAL HARDWARE USED: Please check those applicable to you.

20 ☐ DECMATE
82 ☐ DECsystem-10
83 ☐ DECSYSTEM-20
52 ☐ LSI-11
3 ☐ PDP-8 FAMILY
50 ☐ PDP-11 FAMILY
21 ☐ PROFESSIONAL
22 ☐ RAINBOW
54 ☐ VAX FAMILY
5 ☐ WPS-8
51 ☐ WPS-11

MAJOR OPERATING SYSTEMS? LANGUAGES USED: Please check those applicable to you

1 ☐ ADA
2 ☐ ALGOL
5 ☐ APL
7 ☐ BASIC
17 ☐ BLISS
19 ☐ C
22 ☐ COBOL
26 ☐ CORAL-66
28 ☐ COS
34 ☐ DATATRIEVE
35 ☐ DBMS
38 ☐ DECnet
43 ☐ DIBOL
45 ☐ DOS-11
47 ☐ FOCAL
48 ☐ FORTRAN
51 ☐ GAMMA
110 ☐ IAS
53 ☐ IQL
58 ☐ MACRO
65 ☐ MUMPS
67 ☐ OS/8
68 ☐ PASCAL
72 ☐ PL-11
92 ☐ RPG
81 ☐ RSTS/E
83 ☐ RSX
91 ☐ RMS
109 ☐ RT-11
97 ☐ TECO
70 ☐ TOPS-10
71 ☐ TOPS-20
104 ☐ VMS
107 ☐ WPS-8

TYPE OF BUSINESS (ENVIRONMENT)/COMPUTER APPLICATIONS

Please check that which best describes your business/application

21 ☐ ACCOUNTANCY
7 ☐ BANK
64 ☐ BUSINESS/COMMERCIAL
74 ☐ BUSINESS/INFORMATION SYSTEMS
57 ☐ CHEMISTRY
54 ☐ CLINICAL LABORATORY
63 ☐ COMPUTATION
11 ☐ CONSUMER ELECTRONICS
18 ☐ CONSULTANT
72 ☐ DATA ACQUISITION
52 ☐ DATA COMMUNICATIONS
13 ☐ DATA PROCESSING SERVICES
71 ☐ DATA REDUCTION
17 ☐ DIGITAL EMPLOYEE-ENGINEERING
15 ☐ DIGITAL EMPLOYEE-MARKETING
16 ☐ DIGITAL EMPLOYEE-SERVICE GROUP
60 ☐ EDUCATIONAL ADMINISTRATION
1 ☐ EDUCATION/PRIMARY
2 ☐ EDUCATION/SECONDARY
61 ☐ EDUCATION-TECHNOLOGY
3 ☐ EDUCATION/UNIVERSITY
67 ☐ ENGINEERING
65 ☐ FINANCE/ACCOUNTING
77 ☐ GOVERNMENT
75 ☐ GRAPHICS
4 ☐ HOSPITAL
62 ☐ INDUSTRIAL
55 ☐ LABORATORY/SCIENTIFIC
14 ☐ LIBRARY
58 ☐ LIFE SCIENCES
70 ☐ MANUFACTURING
79 ☐ MARKETING
59 ☐ MEDICAL RESEARCH
6 ☐ MILITARY INSTALLATION
73 ☐ NUMERICAL CONTROL
68 ☐ OEM-COMMERCIAL
78 ☐ OEM-TECHNICAL
56 ☐ PHYSICAL SCIENCES
20 ☐ RESEARCH/DEVELOPMENT
10 ☐ RETAIL
76 ☐ SOFTWARE DEVELOPMENT
53 ☐ TELECOMMUNICATIONS
19 ☐ TELEPHONE/UTILITIES
51 ☐ TIMESHARING
80 ☐ TRAINING/INSTRUCTION
66 ☐ TYPESETTING/PUBLICATION

SPECIAL INTEREST GROUP (SIGs) ENROLLMENT

I wish to participate in the following DECUS U.S. Chapter Special Interest Groups.

33 ☐ APL SIG
2 ☐ COMMERCIAL LANGUAGES
6 ☐ DATA MGMT.SYS.
5 ☐ DATATRIEVE
7 ☐ BUSINESS APPL.
8 ☐ EDUSIG
10 ☐ GRAPHICS APPL
11 ☐ HARDWARE AND MICRO
35 ☐ IAS
31 ☐ DAARC(LABS)
27 ☐ LARGE SYSTEMS
16 ☐ LANG. AND TOOLS
14 ☐ MUMPS
15 ☐ NETWORKS
34 ☐ OFFICE AUTOMATION
36 ☐ PERSONAL COMPUTER
18 ☐ RSTS/E
17 ☐ RSX
19 ☐ RT-11
32 ☐ SITE MGMT.& TRNG
21 ☐ UNISIG
26 ☐ VAX SYSTEMS

JOB TITLE/POSITION - Please check:

1 ☐ CORPORATE STAFF
2 ☐ DIVISION OR DEPARTMENT STAFF
3 ☐ SYSTEMS ANALYSIS
4 ☐ APPLICATIONS PROGRAMMING
5 ☐ SYSTEMS ANALYSIS/PROGRAMMING
6 ☐ OPERATING SYSTEM PROGRAMMING
7 ☐ DATABASE ADMINISTRATION
8 ☐ DATA COMMUNICATIONS/TELECOMMUNICATIONS
9 ☐ COMPUTER OPERATIONS
10 ☐ PRODUCTION CONTROL
101 ☐ CORPORATE DIRECTOR OF DP/MIS
102 ☐ ADMINISTRATIVE ASSISTANT
103 ☐ TECHNICAL ASSISTANT
104 ☐ SERVICES COORDINATOR
105 ☐ MANAGER
106 ☐ ANALYST
107 ☐ PROGRAMMER
108 ☐ DATABASE MANAGER
109 ☐ DATABASE ADMINISTRATOR
110 ☐ MANAGER OF DP OPERATIONS

CITIZEN OF UNITED STATES OF AMERICA? ☐ Yes ☐ No Country: ____________________

Signature: ____________________ Date: ____________________

Forward To:

DECUS U.S. CHAPTER, MEMBERSHIP PROCESSING GROUP
219 BOSTON POST ROAD
MARLBORO, MA 01752, USA
PHONE: (617) 480-3418

INPUT/OUTPUT Submission Form

A SIG Information Interchange

Please reprint in the next issue of the Pageswapper

If this is a reply to a previous I/O, which number? ________

Caption: ____

Message: ____

____

____

____

____

____

____

Contact:

Name ____

Address ____

____

____

____

Telephone ____________________________

Signature ____________________________ Date ________________

Mail this form to: Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station, Cambridge, MA 02139-0901, USA

For information about on-line submission, dial (in the United States): (617) 262-6830 and log in with the username PAGESWAPPER.

Tear out or photocopy reverse to submit an I/O item

Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station
Cambridge, MA 02139-0901
USA

System Improvement Request Submission Form

Page 1 of _____

Submittor: Firm:

Address: Phone:

How to write an SIR:
Describe the capability you would like to see available on VAX systems. Be as specific as possible. Please don't assume we know how it's done on the XYZ system. Justify why the capability would be useful and give an example of its use. If you wish, suggest a possible implementation of your request.

Abstract (Please limit to four lines):

Description and examples (use additional pages if required)

Tear out or photocopy reverse to submit an SIR

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, Ohio 43201-2693
USA

VAX Systems SIG Spring 1986 SIR Ballot

DECUS membership number ________________ (six digits)

Our site uses the following VAX models (check all that apply)

8600 ____ 11/782 ____ 11/780,11/785 ____ 11/750 ____
11/730,11/725 ____ MicroVAX ____

We use VAX's in the following applications (Check all that apply)

Business EDP ____
Education ____
Data Acquisition/Control____
Service Bureau ____
Scientific/Engineering ____
Telecommunications _____
Other ___________________________
Software Development ____
Computer Science Research ____
CAD/CAM ____
Hardware Development ____
Office Automation ____

I support the following as the most important System Improvement Requests. (List from zero to fifteen SIR's):

SIR Number:

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

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

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

I oppose the following SIR's as detrimental. (List from zero to five SIR's):

SIR Number:

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

Mail to:

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, OH 43201

To be counted, you ballot must be received by April 1.

Tear out or photocopy reverse to vote on SIRs

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, Ohio 43201-2693
USA

DATAGRAM

DATAGRAMs are short messages, comments, requests, or answers that are published in NETwords. Please fill in the sections below and send the DATAGRAM to:

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Title: ____

Message: ____

____

____

____

____

____

____

____

____

____

____

____

____

Your Name: ______________________________________
Address: ____
____
Telephone: ______________________________________

If this is a reply to a previous DATAGRAM, what #? __

Signature: ______________________________ **Date:** ____

Place
Stamp
Here

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Fold Here

OFFICE AUTOMATION SIG

SYSTEM IMPROVEMENT REQUEST SUBMISSION FORM

Name ______________________________ **Address** ______________________________

Firm ______________________________ ______________________________

Telephone ______________________________ ______________________________

INSTRUCTIONS: System Improvement Requests (SIR) can be either hardware of software; please check the category addressed by this SIR. Under ABSTRACT, give a brief definition of the capability you would like. In the DESCRIPTION section, give a detailed description and examples of what you want. Be specific; don't assume that we know how other products function. Justify the usefulness of the capability and give an example of its use.

HARDWARE IMPROVEMENT	SOFTWARE IMPROVEMENT	
DECmate ____________	ALL-IN-1 ________________	WPS ____________
PRO-Series ___________	CP/M (DECmate) __________	P/OS ___________
Rainbow ____________	CP/M (Rainbow) __________	MS-DOS _________
Other _______________	Other ____________________	

ABSTRACT ____

____

____

DESCRIPTION ____

____

____

____

____

____

____

____

E. Catherine Ditamore
ARA Services
Corp MIS
Independence Square West
Philadelphia, Pa. 19106

Product Improvement Request Submission Form

Submittor: ______________________________ DECUS Membership NO: ____________

Firm: ______________________________ Telephone: (___) ___-____

Address: ____

____

Product or Products: ____

____

How to write a PIR

A PIR should be directed at a specific product or group of products. Be sure to give the full name of the product(s) and version numbers if applicable. Describe the functionality you would like to see in as complete terms as possible. Don't assume that the PIR editors or software developers know how it is done in some other software product - state specifically how you want the software to function. Provide justification of your request and give an example of its use. If you can, suggest a possible implementation of your request.

Abstract: (Please limit to one or two short sentences.)

Description and Examples: (Use additional pages as necessary.)

affix
proper
postage

PIR Editor, Philip A. Naecker
Consulting Software Engineer
3011 North Mount Curve Avenue
Altadena, CA 91001
USA

DTR/4GL SIG Spring 1986 PIR Ballot

DECUS Membership Number: __ __ __ __ __ __

CPU Types (Check all that apply):
VAXes___ PDP-11's___ DECsystems___ Other (Specify)___________

Application Types at your site (Check all that apply):

___ Business EDP/MIS ___ Software Development ___ Education
___ Engineering/Scientific ___ Office Automation ___ Service Bureau
___ Other (Specify)__________________

Number of years using computers:____ Number of years using 4GL's: ____

Products Used (Check all that apply):

___ DTR-11 ___ VAX-DTR ___ CDD ___ TDMS ___ DBMS(any) ___ FMS
___ RS1 ___ Oracle ___ Ingress ___ Rdb
___ Others (Specify)____

____

PIR Number	Points	PIR Number	Points
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____
________	_____	________	_____

Be sure to return your ballot by July 31, 1986

affix
proper
postage

PIR Editor, Philip A. Naecker
Consulting Software Engineer
3011 North Mount Curve Avenue
Altadena, CA 91001
USA

Ask the WOMBAT WIZARD Submission Form

To submit a problem to the WIZARD, please fill out the form below and send it to:

Donald E. Stern, Jr., DTR/4GL SIG Newsletter Editor
Warner Lambert Company
10 Webster Road
Milford, CT 06460

Name:____

DECUS Membership No. ____________

Affiliation:______________________________________

Address:_______________________________________

____

Telephone Number: ________________________

Statement of Problem:

____

____

____

____

Guidelines and rules for submitting questions to the Wombat Wizard:

1. If you are trying to demonstrate a method or a concept, please simplify the procedures, records, and other information to the shortest form possible. Avoid long procedures where only a small portion of the procedure is required to demonstrate the concept.

2. Annotate your attachments. Simple comments or handwritten notes ("Everything worked until I added this statement.") go a long way toward identifying the problem.

3. Keep an exact copy of what you send. And number the pages on both copies. But send everything that is related to your question, even remotely.

4. Wombat Wizard is not the Telephone Support Center, nor is it part of DEC's Software Performance Reporting (SPR) system. Our goal is to answer "how to" or "how come" questions in an informative and instructive fashion - not to be a clearinghouse for software performance problems.

5. If you would like a direct response or would like your materials returned, please don't forget to include a stamped, self-addressed envelope large enough to hold the materials you send.

(fold here)

Donald E. Stern, Jr., DTR/4GL SIG Newsletter Editor
Warner Lambert Company
10 Webster Road
Milford, CT 06460

(fold here)

PC POSTSCRIPT

PC Postscripts are short requests, comments and responses to be published in the *Postscript Section* of the PC SIG Newsletter. Please respond to the following:

__ Y/N This is a reply to a previous Postscript. __ __ Issue Mo. ______ No.

Title:____

Message:____

____

____

____

____

____

____

____

____

____

____

____

____

Name: ____

Address: ____

____

Phone: (_____) __________-______________

Signature: ________________________________ **Date**__________

Cheryl Johnson (Interim Editor)
Grinnell College
P. O. Box 805
Grinnell, IA 50112-0810

PERSONAL COMPUTING SPECIAL INTEREST GROUP
VOLUNTEER FORM

Name______________________________
Company____________________________
Address____________________________
City____________State____________Zip Code______
Telephone__________________________

What special talents do you have?____________________

When do you attend symposia?

☐ Always
☐ East Coast Only
☐ West Coast Only

☐ Occasional Attendance
☐ Other (please specify)
____
____

Please check if you are interested in helping with any of the following activities:

Symposia Related Activities:

☐ Session Chairs____________
☐ Campground Volunteer____________
☐ Suite Volunteer____________
☐ DECUS Store____________
☐ Software Clinic____________
☐ Panels____________ (indicate topics)____________
☐ Technical Sessions____________ (indicate topics)____________

☐ Articles for Update.Daily
☐ Write letters of appreciation
☐ Equipment Setup

Ongoing SIG Activities:

☐ Working Groups____________ (indicate which groups)____________
☐ Newsletter________________________
☐ Public Domain Software Project____________________
☐ Write Software for Special SIG Needs__________________

Other SIG Activities: (please specify)____________________

Do you wish to see the PCSIG undertake any activities which it is not currently doing? Please specify.

____

Would you be willing to coordinate the activity you have listed above? ☐ Yes ☐ No

Thank you

Information Resource Sign Up Sheet
Personal Computing Special Interest Group - PC SIG

Are you willing to be an information resource for other PC SIG members? Placing your name on the *Contact List* means you are willing to answer questions within the span of a brief telephone conversation. A Contact is not expected to be a consultant. *Please Register below.* Your name and phone number (including restrictions) will be posted in the PC SIG Newsletter.

First Name:________________________ **Last Name:**____________________________________

Address:____

____

City:___________________________ **State:**_________________ **ZIP:**__________________

Phone: (_____) __________-_________________

Areas of Expertise:_______________________

____

____

____

____

Suggestions for Additional Services the SIG can Provide:

____

____

____

Barbara A. Maaskant
UTHSCSA Computing Resources
7703 Floyd Curl Drive
San Antonio, Texas 78216

DECUS PERSONAL COMPUTER SIG QUESTIONNAIRE

General:
I would like information on ______________________________
I would like to see an article
in the newsletter on ______________________________
I would like to see a symposium
session on ______________________________

I am willing to write an article(s) on: ____________________
I am willing to be contacted by PC SIG members by telephone to give
assistance/advice on: ______________________________
Phone number to call: Area Code (___) # __________ Times __________

I attend DECUS Symposiums : ____always ____Sometimes ____never
I expect to attend these symposiums ____Fall 85 ____Spring 86 ____Fall 86

I use/own: ___Rainbow(s) ___PRO(s) ___DECmate(s) ___Robin ___Other________
I use the machine(s) checked above: ___at work ___at home ___both
If a work, total number of DEC PC's at your site: _______
I also use: ___VAX ___IBM or other mainframe ___IBM/other PC
Type of use: ___business ___educational ___government ___other__________
Primary Operating System: ___MS-DOS ___CP/M ___both equally
___P/OS ___UNIX ___other__________________

I belong to a local DEC PC Group: ___yes ___no
There is a user group in my geographic area: ___yes ___no
I would like information on starting a user group: ___yes
I use a modem: ___often ___reluctantly ___never
___for work ___for pleasure ___both

Here is information on he DEC PC User Group I belong to or know of:

Name of Group ______________________________
Name of Contact Person ______________________
Address ______________________________
____
Telephone (___)______________________________
Supports ___Rainbow ___PRO ___DECmate ___Robin ___LUG ___Gold Key

Here is a DEC oriented bulletin board not on your list, or new information on a listed board:

Name of Board ______________________________
Full name of Sysop __________________________
Address if known ___________________________
City and State ______________________________
Telephone Number ___________________________
Other Info: ______________________________
Supports ___Rainbow ___PRO ___DECmate ___Robin

The subjects of greatest ineres to me are:
___word processing
___spreadsheets
___database
___graphics
___communications
___programming
___software reviews
___technical articles
___project management
___specialized vertical software
___(type)__________________
___Other:__________________
___Rainbow
___PRO
___DECmate
___Robin

___DEC Gossip and News ___Other:_______________________

If I had it to do over again, I:
___would buy another DEC Rainbow/PRO/DECmate (circle one)
___might buy another Rainbow/PRO/DECmate if it was a bargain (circle one_
___would not buy another Rainbow/PRO/DECmate (circle one)

Will you continue to subscribe at the new price of $35/year? ___yes ___no
Feel free to enclose another page(s) with comments!
Do you feel that leaving the prices out of the newsletter:
___is appropriate
___is very annoying
___makes the articles less useful

Do you feel that Decus should revise its (anti)commercialism policy?
___yes
___no

Name_____________________________
Company__________________________
Address__________________________
City/ST/ZIP______________________
Work Phone (___)_________________
Home Phone (___)_________________

Return to:
Barbara Maaskant
Computing Resources
The University of Texas Health
Science Center at San Antonio
7703 Floyd Curl Drive
San Antonio, Texas 78284

fold here, flap under---

stamp

Barbara Maaskant
Computing Resources
The University of Texas Health
Science Center at San Antonio
7703 Floyd Curl Drive
San Antonio, Texas 78284

Printed in the U.S.A.

"The Following are Trademarks of Digital Equipment Corporation"

ALL-IN-1
DEC
DECnet
DECmate
DECsystem-10
DECSYSTEM-20
DECUS
DECwriter
DIBOL

Digital logo
EduSystem
IAS
MASSBUS
PDP
PDT
P/OS
Professional
Rainbow

RSTS
RSX
RT
UNIBUS
VAX
VMS
VT
Work Processor

Copyright ©DECUS and Digital Equipment Corporation 1986
All Rights Reserved

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation or DECUS. Digital Equipment Corporation and DECUS assume no responsibility for any errors that may appear in this document.

POLICY NOTICE TO ALL ATTENDEES OR CONTRIBUTORS "DECUS PRESENTATIONS, PUBLICATIONS, PROGRAMS, OR ANY OTHER PRODUCT WILL NOT CONTAIN TECHNICAL DATA/INFORMATION THAT IS PROPRIETARY, CLASSIFIED UNDER U.S. GOVERNED BY THE U.S. DEPARTMENT OF STATE'S INTERNATIONAL TRAFFIC IN ARMS REGULATIONS (ITAR)."

DECUS and Digital Equipment Corporation make no representation that in the interconnection of products in the manner described herein will not infringe on any existing or future patent rights nor do the descriptions contained herein imply the granting of licenses to utilize any software so described or to make, use or sell equipment constructed in accordance with these descriptions.

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

Ada is a trademark of the U.S. Government, XEROX is a trademark of Xerox Corporation, IB M, PROFFS are trademarks of International Business Machines Corporation, UNIX is a trademark of AT&T Bell Laboratories, CP/M, PL/I are tademarks of Digital Research, Inc., MS-DOS is a trademark of Microsoft Corporation, TSX-PLUS is a trademark of S&H Computer Systems, Inc.

DECUS

DECUS SUBSCRIPTION SERVICE
DIGITAL EQUIPMENT COMPUTER SOCIETY
219 BOSTON POST ROAD, (BP02)
MARLBORO, MA 01752-1850

STATUS CHANGE

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

( ) Change of Address
( ) Please Delete My Membership Record
(I Do Not Wish To Remain A Member)

DECUS Membership No: ____________
Name: ____________________________
Company: _________________________
Address: __________________________
____
State/Country: _____________________
Zip/Postal Code: ___________________

Mail to: DECUS - Attn: Subscription Service
219 Boston Post Road, BP02
Marlboro, Massachusetts 01752-1850
USA

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

Bulk Rate
U.S. Postage
PAID
Permit No. 18
Leominster, MA
01453