The Cache Buffer

*

RSTS/E SIG DECUS U.S. CHAPTER

DECUS
Subscription Service

DECUS

November - December 1983

Newsletter of the

RSTS/E Special Interest Group

The Cache Buffer *

Newsletter of the RSTS/E Special Interest Group

Ray Gebbi, Editor

Volume 10 Number 2 December - November 1983

SYMPOSIUM REPORT — PART II

--- Beginning on Page 9

SYMPOSIUM HANDOUTS

--- Beginning on Page 22

Plus Other Useful Pieces of Information .....

In This Issue

SIG Officers

SIG Chairman

Bruce L. Gaarder
Macalester College
1600 Grand Ave.
St. Paul, MN 55105
612-696-6138

Symposia Coordinator

Scott Daily
Great Lakes Chemical Corp.
P.O. Box 2200
West Lafayette, IN 47906
317-463-2511

Newsletter Editor

Ray E. Gebbie
Guntert Sales
P.O. Box 1688
Stockton, CA 95201
209-464-8712

Wishlist Coordinator

Jeff Killeen
Information Design & Management
45B N. Main St.
Sherborn, MA 01770
617-653-2500

BASIC SIG Liaison

James Hodges
Merrell-National Labs
2110 E. Galbraith Rd.
Cincinnati, OH 45215
513-948-9111 x2566

EDUSIG Liaison

George Wyncott
Purdue University
Mathematical Sciences Bldg.
West Lafayette, IN 49707
317-494-1787

Small-Users Sub-SIG

Scott Pandorf
Kittle's
8600 Allisonville Rd
Indianapolis, IN 46250
317-849-5300

Library Coordinator

Mark Hartman
ImPrint Systems, Inc.
303 W. Katella, Suite 303
Orange, CA 92667
714-771-6480

INDENT Sub-SIG

Dean Goranson
Cincinatti Gear Co.
Cincinatti, OH 45227
513-271-7700

Tape Copy

Carl Hauger
P.O. Box 4052
Greenville, DE 19807
302-655-3544

If there is anyone who was left out, or if there are any corrections, let me know as soon as possible, so that the next newsletter can be correct.

From The Editor

The RSTS world is changing very fast, and we will have to change our ideas about data processing when the newer generations of PDP-11's arrive. The trend is toward machines with great processing capabilities but with more limitations as to number of simultaneous users.

These new machines will have processors with capabilities beyond any current PDP-11's, yet will sell at a fraction of the cost. They will have the memory capacity and power of the 11-7Ø. But their physical size will be much smaller. There will not be room on the back of the new machines for dozens of terminal ports. FCC regulations will bring on limitations in the number of physical ports on these machines. It may be that 8-1Ø ports will be the practical limit.

Our thinking has to be turned toward multiple machines with greater power but fewer users. We will then connect these machines using networking software (such as Decnet). It seems to me that this arrangement will be more efficient, as there will be much more computing power available to each user then on today's machines. This will be done at a much lower cost also.

It will no longer be "think big", and see how much we can get a single machine to do. We will have to "think small", and spread the tasks over our new smaller and more powerful machines. *

How To Submit To "The Cache Buffer"

I urge you to submit articles for the newsletter on a machine-readable medium. I will promptly return all media. The ideal medium for me is 8ØØ BPI tape, but I can handle with greater effort 16ØØ BPI tape or WPS 8 floppy. The format of the article can be RUNOFF, RNO, or plain text file. With machine-readable articles, I can print them myself and include headlines and page numbering and the print quality of the newsletter can be more uniform. But if you are unable to provide it using one of the above formats or media, by all means send it anyhow. For .RNO files, please use the following print settings:

```
.PAGE SIZE 6Ø,8Ø
.LEFT MARGIN 8
.RIGHT MARGIN 72
.SPACING 1
```

For other formats, use a page size of 6Ø lines by 65 characters, single spaced, one side of the paper only.

I have had to revise the schedule and issue dates for future newsletters as follows:

ISSUE	DEADLINE
Jan-Feb 1984	Dec. 1, 1983
Mar-Apr 1984	Feb. 1, 1984
May-Jun 1984	Apr. 1, 1984
Jul-Aug 1984	June 1, 1984

Send your submissions to:

Ray Gebbie
RSTS Sig Newsletter Editor
Guntert Sales Div., Inc.
P.O. Box 1688
Stockton, CA 952Ø7
2Ø9-464-8712

*

Letters to "The Cache Buffer"

Dear Cache Buffer:

A problem has recently come to light that I think the RSTS/DIBOL community should be made aware of. The problem concerns RSTS/E Version 8.Ø SORT and any data files that use a '^Z' to indicate end of file (in our case it was MCBA type files). The problem with the new SORT-11 package is that, apparently, RMS Version 2 has been changed so that it no longer recognizes the '^Z' as EOF hence when you sort using the new SORT your data file will become 'corrupt' with what appears to be the control Z and the rest (of the block) sorted to the beginning of the file. This will obviously cause problems with any follow-on programs.

We have submitted an SPR to DEC on this problems, but initial reaction has not been promising. In this light we have come up with a 'work around' that appears to do the job.

We copied the old "$SORT.TSK" to a file on our version 8.Ø disk called "$SRTV18.TSK" and then modified all of our DIBOL programs to use this task instead of "$SORT .TSK". It would appear that you could copy the old sort down 'on top' of the new sort, but this may cause problems for new (V2.Ø) RMS files and the like, so I wouldn't recommend doing it.

S.W. Pandorf
Director Management
 Information Systems
Kittle's
86ØØ Allisonville Rd
Indianapolis, IN 4625Ø
317-849-53ØØ

Dear Cache Buffer:

The following patch will allow STK (and also 7.2 SLOTKB) to turn on data cacheing for the task builder's work file. For very large task builds this batch can have, depending on the system load, the effect of making STK cpu bound. For the 'File to patch?' prompt answer with DDn:$STK.TSK, TKB.TSK or SLOTKB.TSK. Note that future patches for STK may require the use of a larger offset address into PATCH. Note also that the data cacheing will only work for privileged users as NO attempt is made here to drop any temporary privileges. System performance could be adversely affected by this patch due to heavy activity in tne cache in XBUF.

Brian Nelson
University of Toledo
28Ø1 W. Bancroft St
Toledo, OH 436Ø6

```
RUN $ONLPAT
Command file name ? STKPAT
File to patch?
Base address? .OPFNB+41Ø+1574
Offset address? 14Ø
Base     Offset   Old      New?
??????   ØØØ14Ø   ØØØØ2Ø   ? <LF>
??????   ØØØ142   ØØØØ43   ? <LF>
??????   ØØØ144   ØØ14Ø2   ? <LF>
??????   ØØØ146   Ø127Ø4   ? 4737
??????   ØØØ15Ø   ØØØØ32   ? PATCH+1Ø
??????   ØØØ152   13276Ø   ? ^Z
Offset address? ^Z
Base address? PATCH
Offset address? 1Ø
Base     Offset   Old      New?
??????   ØØØØ1Ø   ØØØØØØ   ? 127Ø4
??????   ØØØØ12   ØØØØØØ   ? 32
??????   ØØØØ14   ØØØØØØ   ? 12765
??????   ØØØØ16   ØØØØØØ   ? 1ØØ4ØØ
??????   ØØØØ2Ø   ØØØØØØ   ? 22
??????   ØØØØ22   ØØØØØØ   ? 2Ø7
??????   ØØØØ24   ØØØØØØ   ? ^C
```

The follow letter refers to the article entitled "Basic-Plus-II Solution?" in the August issue of "The Cache Buffer". This article included some programs that could be used to patch Basic-Plus-II programs for running under V8.Ø of RSTS. I had used RUNOFF to format the article and did not notice that it had stripped off the up-arrow in all of the control Z's in the programs. --Editor

Dear Cache Buffer:

Thank you very much for your kind and speedy help regarding my Resident Library renaming problem. The program listings you sent were much appreciated. They got me right on track.

I don't know if it might have just been something in the way you printed out my listing, but there was one error (that showed up 12 times). The line came out:

```
\ PRINT #2%, "Z"
```

whereas it (quite obviously to me) should have been:

```
\ PRINT #2%, "^Z"
```

Finally, since I can never do things without 'improving' them, you might be interested in a modified version of the program I put together. It sort of combines all 3 of your programs into one by adding a short dialog at the beginning to define the old and new resident library names and also allowing for either command file or interactive entry of the task specs. It also does the patching immediately rather than building a command file for using ONLPAT later. In a very limited timing test (modifying 2 tasks each being 29 blocks long and having 3 occurances of the library name), I found that your original versions took 3 CPU seconds whereas my 'jazzed up' version did it in about 1 CPU second. HERE IT IS -

Robert Holz
Technical Support Specialist
Allied Data
172Ø Sleater-Kinney Rd
P.O. Box 24Ø6
Olympia, WA 985Ø7
2Ø6-456-3535

```
1       ! *** CHGRLR.B2S &
        ! Program to change Resident Library references in tasks. &
        ! Input to the program are: &
        !       1) The name of the currently existing resident library &
        !       2) The name of the new resident library &
        !       3) The name of an indirect command file containing the &
        !               names of the program files to be modified. &
        ! The indirect command file is usually a directory file created &
        ! by PIP using a command similar to - &
        !       PIP TASK.DIR=DRØ:[*,*]*.TSK/DI:NA:EX &
        ! For modifying just a few programs however, it may be easier to &
        ! just type in the file names interactively.  To do so simply: &
        !       1) Use "KB:" for the 'indirect command file name' &
        !       2) Then type one file-spec each time you are prompted &
        !       3) Use 'CTRL/Z' to exit &
        ! Note that when you are interactively entering file specs, you &
        ! may do so in either of two formats: &
        !       1) Enter each spec. completely (DEV:[ACCT]NAME.EXT) &
        !       2) Enter just the 'DEV:[ACCT]', then (when re-prompted) &
        !         enter just the 'NAME.EXT' on a separate line. Note that &
        !         when you enter the device/acct separately the program &
        !         will continue to use that DEV:[ACCT] until you explicitly &
        !         enter a new one.  This can be faster if you have a number &
        !         of file to do in the same directory. &
        ! The files are patched immediately. &

10 INPUT " Name of current resident library to be changed"; RLNAME$ &
\  RLNAME$ = MID$( SYS(CHR$(6%)+CHR$(-1Ø%)+RLNAME$), 7%, 4%) &
\  INPUT "New name for this resident library will be    "; NEW.RLNAME$ &
\  NEW.RLNAME$ = MID$( SYS(CHR$(6%)+CHR$(-1Ø%)+NEW.RLNAME$), 7%, 4%) &
\  PRINT "Enter indirect command file name <KB:>"; &
\  INPUT LINE I$ &
\  I$ = EDIT$( I$,39%) &
\  I$ = "KB:" IF LEN(I$)=Ø% &
\  OPEN I$ FOR INPUT AS FILE 1% &
\  ON ERROR GO TO 1ØØØ &
\  UNTIL EOF% &
\      PRINT "Enter program file specification: "; IF I$="KB:" &
\      INPUT LINE #1%, A$ &
\      LOC.DOT% = INSTR( 1%, A$, ".") &
\      LOC.COLON% = INSTR( 1%, A$, ":") &
\      IF  LOC.DOT%=Ø%  THEN &
                ACCT$ = EDIT$( A$, 4%)  IF LOC.COLON%<>Ø% &
       ELSE &
                FIL$ = LEFT$( A$, LOC.DOT%+3%) &
\               ACCT$ = "" IF LOC.COLON%<>Ø% &
\               PRINT "...working on "; ACCT$; FIL$; "..."; &
\               GOSUB 1ØØ      ! Process the file &
\               PRINT " - no references found" IF NUM.FIXED%=Ø% &
\               PRINT " - found "; NUM.FIXED%; " references" IF NUM.FIXED% &
\      !ENDIF
 99 NEXT &
\  CLOSE 1% &
\  GOTO 32767 &

 1ØØ   ! Subroutine to process a selected file replacing all references &
       ! to "RL.NAME$" with "NEW.RLNAME$". &
       ! &
       ! The global input variables used are: &
       !       ACCT$, FIL$, RLNAME$, NEW.RLNAME$ &
       !     (Note: RLNAME$ & NEW.RLNAME$ must each be 4 bytes long) &
       ! The global results produced are: &
       !       NUM.FIXED% (and of course the modified file on disk) &
       ! The 'local' variables used are: &
       !       R$, D$, N%, I%, EOF%, GOT.ONE% &
       !   (Note of these need to be preserved from one call to the next) &
       !   (EOF% will ALWAYS be = Ø% at the end of the subroutine) &

 1Ø1            OPEN ACCT$ + FIL$ FOR INPUT AS FILE 3%, MODE 5% &
 \              FIELD #3%, 512% AS R$ &
  \             EOF%, NUM.FIXED%, N% = Ø% &
  \             UNTIL EOF% &
  \                     N% = N% + 1% &
\                   GET #3%, BLOCK N% &
\                   GOT.ONE%, I% = INSTR( 1%, R$, RLNAME$) &
\                   WHILE I%
\                          FIELD #3%, I%-1% AS D$, 4% AS D$ &
\                          LSET D$ = NEW.RLNAME$ &
\                          I% = INSTR( I%+4%, R$, RLNAME$) &
\                          NUM.FIXED% = NUM.FIXED% + 1%
\                   NEXT &
\                   PRINT N%; IF GOT.ONE%     !May be removed if desired &
\                   PUT #3%, BLOCK N%  IF GOT.ONE% &
\                   EOF% = -1%  IF N%>1% AND NUM.FIXED%=Ø% &
\            NEXT &
\            CLOSE #3%
119          EOF% = Ø% &
\            RETURN &

1ØØØ  IF     ERR=5%  THEN &
             RESUME 119 &
      ELSE IF ERR<>11%  THEN &
             ON ERROR GO TO Ø &
      ELSE &
             EOF% = -1% &
\            RESUME &

32767 END
```

Dear Cache Buffer:

I want to compliment you on the fine first subscription service edition of "The Cache Buffer". Thank you very much for including the new DECUS Library offerings for RSTS. I feel that the SIG Newsletters are the best advertising forum the Library has. I hope this will become a regular feature of your newsletter.

Ardoth A. Hassler
DECUS U.S. Program Library
Coordinator
The Catholic Univ. of America
Washington, D.C. 2ØØ64

Do your friends a favor ..... give the
Subscription Service Order Form
on page 55 of this issue.
Your friends will Thank you.

New DECUS Library Offerings for RSTS Users

DECUS# DESCRIPTION

11-628 WRDBX -- Subroutine for Dollar conversion
A DIBOL Subroutine to accept a decimal field and convert to English-language, dollars and cents string. For example, it will convert "1178565" to "ELEVEN THOUSAND SEVEN HUNDRED EIGHTY FIVE AND 65/1ØØ DOLLARS".

11-629 Roger Williams College Collection
Misc. BASIC-PLUS programs. Includes:
XYPLOT.BAS -- 2-dim cartesian plotting program capable of plotting 3 simultaneous functions on a GIGI (VK-1ØØ).
ISOMET.BAS -- cartesian plotting program for a single function in X and Y in isometric projection on a GIGI.
ACOUNT.BAS -- system accounting program using output of MONEY and produces reports of CPU time, terminal time, etc.
SUMMER -- used in conjuction with ACOUNT in academic environment.
CIPHER.BAS -- encryption/decryption program which can operate on any file including program source code.

11-633 TREK: A Space Game
A BASIC-PLUS program developed from SPACWR using features of the VT52 and VT1ØØ terminals.

11-637 DIBOL String-Interpreting Subroutines
2 DIBOL subroutines with demonstration programs. Includes an algebraic-string evaluator and a Boolean evaluator.

11-639 Additions to the Roger Williams College Collection
Includes the following:
GRAPH.FOR -- Fortran IV subroutine which will plot up to 5 simultaneous functions of one independent variable.
FORTRA.HLP, ISOMET.HLP, XYPLOT.HLP -- Text files for use with the RSTS/E HELP facility.
RPGBCP.BAS -- Creates batch control files for compiling, taskbuilding and running RPG-II programs.

11-SP-18 (revision) C Language System, Second Master Release
A general-purpose programming language revised for use under RSTS/E V7.2/V8.Ø.

DECUS Library programs are available through:
DECUS, U.S. Chapter
One Iron Way, MRO2-1/C11
Marlboro, MA Ø1752

The latest Program Library Catalog contains a number of programs of interest to RSTS users. Request the catalog and order forms from the DECUS Office. The programs are available at a nominal cost, that is used to cover the expenses of maintaining the library. *

Spring '83 Symposium Report -- Part II

This is a continuation of the Symposium Report began in the last issue. These articles were produced as part of the DECUS Scribe Service. This project envolved a group of students from the St. Louis area who attended selected sessions and produced reports. Please note that the articles were not reviewed by the speakers before publication. It is hoped that this project will continue for many Symposia. We must give our thanks to Ralph Stammerjohn, of Monsanto in St. Louis, whose efforts made the Scribe project a success. *

How to make BASIC Run Faster

Richard Baldwin
North County
Computer Services, Inc.
Escondido, CA

Bill Tabor, Session Chairperson
Racal-Milgo
Miami, FL

Reported by Todd Spangler
DECUS Scribe Service

One of the main objects in the computer world is to minimize the amount of time required to perform a job. The problem has almost always rested in the hands of the programmer. It is the job of the programmer to make the code as efficient as possible, both in memory conservation and CPU time. In the presentation by Richard Baldwin of North County Computer Services, Inc., the problem is taken from the hands of the programmer and placed in the hands of the hardware people.

In the BASIC language there is one option to be considered. By placing a floating point processor or floating point accelerator on the CPU, the speed of many commands can be quickened. Using the processor, the math operations and related internal functions such as SIN, COS, TAN, etc., will be handled much faster. To demonstrate this increase, an experiment using a simple FOR-NEXT loop was tried.

In this experiment, the FOR-NEXT loop was run without any additional instructions in the loop and timed by itself. From this, the CPU time calculated with statements in the loop minus the FOR-NEXT time will be the time used of the instruction. By iterating the FOR-NEXT loop twenty to thirty thousand times, accurate time calculations can be made and compared with the run times of the same set of instructions using a floating point accelerator. When the figures are compared, one can see that there is almost always a significant difference when the accelerator is used.

There are four variables to be considered which influence CPU time. The type of machine and the type of compiler play an important role. Also, the hardware options and the capabilities of the programmer will be crucial to the operations of the machine. The experiment shows that virtually no matter what machine or type of BASIC is used, the program floating point operations will always run faster when using the floating point accelerator. Otherwise, variances will be machine dependent and can be found using similar FOR-NEXT loops. In general, the floating point accelerator will enhance the operations of the computer. *

Macro and BP2 V2 Tutorial

Wef Fleischman
Software Techniques Inc.
Los Alamitos, CA

Bill Tabor, Session Chairperson
Racal-Milgo
Miami, FL

Reported by Gene Mitchell
DECUS Scribe Service

"A high-level language like BASIC is almost always the best implementation language for maintenance and transportability reasons. But you should recognize the potential areas for improved performance and enhanced efficiency afforded by MACRO."

Such was the conclusion reached by Wef Fleischman of Software Techniques, Inc., the speaker at a session entitled "MACRO and BP2 V2 Tutorial." Topics of interest were focused into MACRO vs. BP2, likely applications of both, and programming techniques.

A comparison of the two languages showed some of the better features of BASIC:

1. Well organized, logical structure.

2. Protection of programmer from pitfalls. For instance, BP2 requires all variables to be explicitly and correctly spelled. Earlier BASIC systems allowed for misspelled variables, which could have adverse consequences on other sections of a program.

3. Good maintenance characteristics--many people have a knowledge of the language.

4. Transportable code--can migrate to many different machines.

MACRO-11, on the other hand, said Fleischman, is by comparison confusing to many. Drawbacks include:

1. MACRO-11 does not impose a logical structure.

2. It is sometimes difficult to understand. Programmers will occasionally have difficulty reading each other's code.

3. Debugging is an art. Debugging MACRO programs can adversely affect even logically independent parts.

4. MACRO-11 is not very transportable. MACRO is great for getting at "intimate details" of the PDP-11, but not useful on other machines.

But BASIC has some faults that do not arise in MACRO. While BASIC is more logical, it may not be optimally efficient. In addition, BASIC may not allow some access features that MACRO will. Unneeded overhead may be incurred as a result of the overly structured BASIC program.
MACRO allows several things that BASIC may not:

1. Unrestricted access to the underlying machine and operating system.

2. Greater speed and efficiency in many cases.

3. The performance, in certain cases, of special data manipulations.

4. Explicit control of memory is allowed.

Fleischman offered three examples of incorporating MACRO with BASIC. The first was initialization of static data. First the BASIC program was given:

```
1ØØØ  DECLARE INTEGER PART
      MAP (PARTS) STRING DESCRIPTION (7)=8

2ØØØ  INPUT "PART NUMBER "; PART
      PRINT "DESCRIPTION "; DESCRIPTION (PART)
      GOTO 2ØØØ

32767 END
```

Next, the MACRO equivalent was provided:

```
      .TITLE  PARTDF

      .PSECT  PARTS,RW,D,GBL,REL,OVR

      .ASCII  "BREATHER"
      .ASCII  "CABLE   "
      .ASCII  "CHOKE   "
      .ASCII  "COVER   "
      .ASCII  "PISTON  "
      .ASCII  "SOLENOID"
      .ASCII  "SPRING  "
      .END
```

Problems of sharing data between MACRO and BP2 systems were addressed, with two solutions being offered:

1. Through COMMONs or MAPs. According to Fleischman, the .PSECT directive in MACRO corresponds directly to the BASIC MAP or COMMON statement.

2. Through passed argument lists. An argument list is a contiguous block of memory assigned by the BASIC main program when

the CALL statement is executed. Arguments can be passed by reference (REF), descriptor (DESC) or value (VAL).

The second example of incorporating MACRO with BASIC concerned how to pass data to a MACRO subprogram, and system function not supported by BP2, such as (the .NAME directive). It assumes that one has fairly lengthy processing. According to Fleischman, "This ability to change a job's name is not privileged but cannot be performed in BASIC." The MACRO code to do this is as follows:

```
     .TITLE    CHNAME

     .PSECT    CHNAME,RO,D,GBL,REL,OVR

CHNAME::

     MOV       #XRB+XRLOC,R1          ;POINT TO XRB
     MOV       2(R5),RØ               ;POINT TO DESCRIPTOR
     MOV       (RØ)+,(R1)             ;COPY LOCATION OF STRING
     MOV       (RØ),-(R1)             ;COPY LENGTH OF STRING
     MOV       (R1),-(R1)             ;DUPLICATE IN XLREN
     CLR       @#XRB+1Ø               ;NO SPECIAL LOGICAL TABLE
     .FSS                             ;CONVERT TO RAD5Ø
     TST       @#FIRQB                ;ERRORS?
     BNE       1Ø$                    ;YES, EXIT WITHOUT EFFECT
     .NAME                            ;CHANGE THE JOB NAME
1Ø$  RETURN                           ;RETURN TO CALLER

     .END
```

Calling a MACRO-11 subprogram, such as the one above, can be done with four steps:

1. Declare the external name.

2. Specify parameter list (if any).

3. Double-check parameters.

4. CALL.

An example:

```
1ØØØ  EXTERNAL SUB CHNAME (STRING BY DESC)

2ØØØ  CALL CHNAME ("STEP1")
               *
               *
               *
      CALL CHNAME ("STEP2")
               *
               *
               *
      CALL CHNAME ("STEP3")
```

*
*
*
(etc.)

The third example of incorporating MACRO with BASIC is in demonstrating special RMS file handling. The BASIC program is as follows:

```
1ØØØ    MAP (BUF) STRING KEY=1Ø%
2ØØØ    OPEN "LOAD.IDX" FOR OUTPUT AS FILE #1       &
                , INDEXED FIXED                      &
                , ACCESS MODIFY, ALLOW NONE          &
                , MAP BUF                            &
                , PRIMARY KEY                        &
                , USEROPEN BKTFIL
        PUT #1%
32767   END
```

The MACRO program is as follows:

```
        .TITLE  BKTFIL
        .PSECT  BKTFIL,RW,I,LCL,REL,OVR

        .MCALL  RABOF$,XABOF$,$STORE,$SET
        .MCALL  $FETCH,$CREATE,$CONNEC
BKTFIL::
        MOV     2(R5),RØ                ;GET ADDRESS OF FAB
        MOV     4(R5),R2                ;GET ADDRESS OF RAB
        $FETCH  R1,XAB,RØ               ;GET ADDRESS OF XAB
        $STORE  #256.,DFL,R1            ;SET DESIRED DATA FILL
        $STORE  #256.,IFL,R1            ;SET DESIRED INDEX FILL
        $CREATE RØ                      ;CREATE THE FILE
        $CONNEC R2                      ;CONNECT THE RAB
        $SET    #RB$LOA,ROP,R2          ;SET "LOAD BY FILL"
        RETURN

        .END
```

Fleischman added that in creating the RMS-11 Bucket fill control, there are four steps we must observe:

1. Specify fill factor in key XAB.

2. Create file ($CREATE/CONNECT).

3. Set fill control bit in ROP field of RAB.

4. Load records.

The session concluded with some BASIC V1 "unsupported RMS routines" and other applications for MACRO-11. Unsupported RMS routines include:

1. RETRFA - return record file address

2. GETRFA - set record by RFA
3. NULKEY - enable a null alternate key
4. SEGKEY - define a segmented key
5. BKTFIL - define bucket fill factors
6. SETROP/CLRROP - set record options
7. DEFALQ - set default extension size
8. DEFFNA - set default file name
9. SETFOP - set file options

Other applications for MACRO-11 include:

1. Passing variable number of arguments
2. Accessing job "lowcore" area
3. Patching constants via MAKSIL/ONLPAT
4. Observing/modifying BP2 OTS behavior
5. Controlling libraries explicitly (.PLAS)
6. Performing device specific I/O (.SPEC)
7. Creating "executable" files (CRBFQ)

Fleischman concluded by reemphasizing the areas that MACRO can affect towards improving performance. He also attempted to dispel the idea that MACRO is overly sophisticated for the needs of many. It is on the contrary easy to learn and use. Use of MACRO may decrease programming time or reduce the need for new equipment. *

MICRO(PDP-11) Introduction: Questions and Answers

Don Haney and Atlant G. Schmidt
Digital Equipment Corporation
Maynard, MA

Mike Allen, Session Chairperson
Lawrence Livermore
National Laboratory
Livermore, CA

Reported by J. Rick Mihalevich
DECUS Scribe Service

The Micro/PDP-11 is DIGITAL's newest multiuser microprocessor offering full PDP-11 functionality (PDP-11/23-PLUS) in a compact package that includes a 1Ø MB Winchester disk, a new controller utilizing DIGITAL's new Mass Storage Control Protocol and a new universal power supply. This presentation introduced the Micro/PDP-11 with an overview of the hardware, software and special features for service and ease of use. For more information on the specifications of the Micro/PDP-11 refer to "PDP-11 Micro/PDP-11 Handbook".

In the development of the product five program goals were maintained:

1. Low system cost
2. Low life cycle cost
3. Physically small
4. Ease of installation and use
5. Improved multiuser performance

Low system cost -- The basic system will satisfy most needs with reduced number of packaging options. The Micro/PDP-11 allows maximum use of existing components.

Low life cycle costs -- low service costs and modular upgrades are achieved by replaceable components (CPU, memory, disks, etc.) and technology advancement. Low life cycle costs are also achieved through compatibility with PDP-11 multiuser system family.

Physically small, for both technical and office use -- There is a rack mountable unit which fits standard cabinets as well as table top and under desk versions. The Micro/PDP-11 is a small development system which allows for most of the wanted small system configurations available.

Ease of installation and use -- This system has an automatic boot up with diagnostics on the diskette. The product has improvements that make the system more user friendly.

Improved multiuser performance -- The Micro/PDP-11 provides full multiuser system capability with PDP-11/23 plus CPU and LSI-11 extended bus, 4MB addressing. The product has disk storage improvements by compatibility with the 1ØMB mini-Winchester, 2 4ØØKB diskettes, and new controller which supports up to 4 logical units.

The last segment of this session was reserved for questions and answers. Major concerns with the new product were available storage and upgrading limitations to main memory. There was much concern regarding the availability of auxiliary storage devices and the corresponding media. There was mention of an upgradable CPU which is in the planing stage and the DECNET/Q-Bus Ethernet interface. *

RSTS MICRO PDP-11
Field Test Panel

Jeffrey Killeen
Information Design and Management
Sherborn, MA

Scott Daily
Great Lakes Chemical Corporation
West Lafayette, IN

Reported by Todd Spangler
DECUS Scribe Service

Problems encountered with the Micro PDP-11 were described by Scott Daily of Great Lakes Chemical Corporation and Jeffrey J. Killeen of Information Design and Management. The difference between RSTS/E and Micro RSTS is that Micro RSTS is designed to be run on a machine with smaller memory than is required for RSTS/E. Some initial problems are that there are no architectural reasons that full scale RSTS/E could not be run if there were enough disk space. FCC regulations have placed restrictions on the number of devices which may be attached to the Micro PDP-11 bus, thus severely limiting the mass storage capabilities of the Micro PDP-11. Currently, the Micro PDP-11 comes with a 1Ø Mbyte Winchester disk. However, the Micro PDP-11 is the cheapest way to configure a low end PDP-11/23 system.

Further problems are caused by the use of floppy disks as backup media for the Winchester disks, since the data storage ratio of Winchester to the floppies is 23 to 1. One solution is to have an operator feed in the floppies as needed. A far superior solution would be to get a tape drive or maintain an 8ØØKB cache buffer and cache the entire disk. Possible future innovations include the purchase of an RMØ2 to add to the system, and finding way to add another 5 inch Winchester. Fixed partitioning could also be used to limit logins when address space becomes tight. *

Q-Bus/UNIBUS Hardware Hints and Kinks

Wayne Kesling, Session Chairperson
Monsanto Research Corporation
Miamisburg, OH

Reported by Marc Caffee
DECUS Scribe Service

During this session different people from the audience presented problems they had with their system and resolutions to the problems were discussed.

One of the more frequently appearing problems occurs on VAX UNIBUSs. Sometimes failures occur on machines having more than one high speed DMA device. The problem can be fixed if the DMA device can be slowed down. The details on the timing on the DMA bus transfers were not available at the session. Similar problems apparently occur on VAX-11/78Ø using a combination of foreign DMA devices running UNIX.

There were a variety of complaints and suggestions dealing with RKØ6s and RMØ5s. In particular on the RKØ6 the supports holding the drive motor give way and the motor falls on a transformer located underneath it. However, before it does this it does make a somewhat loud noise. Don't just assume it needs some lubricant, but check and make sure the supports are not wearing out. Also on the RMØ5 there are rubber parts that the door closes against. If these are loose, pull them off before they fall into the disk. If they're not loose they probably won't cause a problem. They should be checked frequently. Also, caution should be used if you have to move a RKØ6 bus cable. If it is not handled gently there is a possibility of damaging it. Finally, there is a diagnostic available on the RKØ6 called operator intervention diagnostic. This diagnostic checks out the batteries that operate the head retractor. If you are powering up a unit that has been sitting idle for a while be sure and check this out.

Disk Drives:

The RA81 disk drive is no longer on engineering hold. However there was no sure word on when retrofits would be available on RA81s already in use. Hopefully it will be soon.

If your RLØ2 temporarily goes offline check the sector transducer. Also, the RLØ2 ready lights burn out quite frequently. Additionally, an old RLØ2 controller will not necessarily work on a 11/23+ backplane.

On an 11/6Ø running RSX the RA8Ø does not spin up after a power failure. The solution to this is to "mount" the disk. The process of accessing the disk will spin it up. This might seem strange but it's true.

RXØ2s are not supported on the 11/23+. At this time the resolution is to fix the driver in lower memory. There are at least two outside vendors that supply boards which will support floppies. In connection to the Q-Bus problem, it is possible for a parity error to occur when using the RXØ2s. This problem occurs on the 22-bit system because of the extra lines in the bus. The 18-bit machine apparently never detects this error.

NASA has conducted a study on head wear versus relative humidity for high speed disks. The conclusion is that for best head wear the humidity should be kept between 4Ø and 5Ø percent.

Interfaces:

In using the DMR-11 with V3 of VMS you can go to full duplex. However, you have to remove the W17 jumper or else things will be slower than half-duplex.

The DMF-32 documentation is at odds with reality. BEWARE!!

A DLV11J can be used as an interprocessor link. If two or more of the lines are used heavily the DLV11J will write PSs and PCs all over memory. The problem involved here is really quite general in that the DLV11J came into being before 4 level interrupt. In other words, this device doesn't check other device priorities and this will cause severe problems. The resolution to the problem is to have this device (or any other similar device) placed behind the devices which have 4 level interrupt.

You might have problems if you try to connect a LP11 to a DMF-32. The LP11 has a 4Ø pin connector that doesn't fit into the DMF-32. A cable which might work is CSS LXY21-PA.

It is possible to transfer disks from a VAX to a DEC-1Ø. However there is one problem which must be overcome. The VAX MASSBUS computes parity on 18-bits. The DEC-1Ø checks parity on 16-bits. In order to make the transfer work the two high order bits must both be Ø's or 1's.

Others:

On an 11/34 after a RSX-11M 3.2 SYSGEN on RKØ5s the system refused to boot after several days. RT-11 and the diagnostics ran fine; only RSX wouldn't run. It turns out that the ribbon connector on the DZV11 raked across the memory card and caused the problem.

Diagnostics are really scarce on the VAX-11/73Ø. If you have some technical questions contact Tom Provost, M.I.T./Bates Linear Accelerator, Middleton, MA. *

Computer Center Relocation

Larry W. Hicks
R.J. Reynolds Tobacco Company
Winston-Salem, NC

Emily Kitchen, Session Chairperson
A.H. Robins Company
Richmond, VA

Reported by Scott Howell
DECUS Scribe Service

After being introduced by the session chair, Emily Kitchen, the speaker, Larry Hicks, discussed how he successfully planned and completed the relocation of the computer center of R.J. Reynolds Tobacco Company.

Hicks began the presentation by discussing the background to his particular relocation. He had to move a single DECSYSTEM-2Ø64 supporting three departments. The system was in the corporate computer center, while users and technical support were in the tobacco company, a subsidiary.

Hicks described the method to plan the move, the most important factor of which is early preparation. To do this one must:

1. Determine the major activities.

2. Determine the impact of each phase of the move.

3. Determine the lead times and critical events.

4. Break each phase into steps.
   a. identify all steps.
   b. determine step and activities dependent upon them.
   c. determine how to perform each step.
   d. estimate time for each step. (be conservative)

5. Validate information of each step.
   a. check other considerations.
   b. verify time estimations.
   c. check overlapping response.

6. Prepare preliminary time table for move.
   a. establish desired move date.
   b. determine initial activity and the earliest starting date.
   c. establish earliest move date.
   d. set realistic move date.

7. Formalize time-table.
   a. solicit comments on the preliminary time table.
      -from user department
      -from vendors
      -from other parties in move
   b. modify time table and schedule.

8. Prepare for implementation
   a. determine "go/no go" points in schedule.
   b. decide action if "no go".

9. Implement move schedule.
   a. publish implementation or move schedule.
   b. add detail.
   c. request immediate notification of problems and questions.

10. Track completion of each step.
   a. have one person with ultimate authority.
   b. review steps weekly.
   c. check daily during the last two weeks.
   d. monitor "go/no go" points.

11. Publish decisions.

12. Schedule periodical report.

The preceding are necessary steps in a computer center relocation. Hicks added several things one must do after the project is completed.

1. Evaluate move.

2. Review problems.

3. Analyze slack time.

4. Evaluate accuracy.

5. Determine move's impact.

6. Review your own action.

7. Thank involved parties in the move.

Hicks concluded his session with a summary of how to make a successful transition of a computer center.

1. Make detailed advanced planning.

2. Allow for a conservative schedule.

3. Make a close monitoring of progress.

4. Publish results. *

Giving Effective Presentations

Wesley E. Mullen
McDonnell Douglas Automation
Berkeley, MO

Adam Zavitski, Session Chairperson
Seismograph Service Corporation
Tulsa, OK

Several years ago, Mr. Mullen attended a conference where he became quite disturbed about the lack of quality in the presentations. He became so disturbed that he started taking notes. That year he attended 4 or 5 conferences and was able to continue taking notes. They were mostly negative things, things that disturbed him, so he started he started to put together a presentation that would be helpful to those called upon to present papers.

Most people have to make a presentation at one time or another. The presentation could be to a large group or to a single person. It could be an informal discussion or, as here at DECUS, a formal talk. We have probably noticed that when we begin to speak, we get weak knees, our hands shake, our voice begins to quiver, and we get this general feeling of inadequacy. The intent of this talk was to help each of you overcome these anxieties and give more effective presentations.

Mr. Mullen noted that at the Cervantes Convention center, a group of St. Louis businessmen who created the International Speaker's Hall of Fame Award have plaques honoring the recipients. The award is for professional speakers who speak throughout the country. Their criteria for judging are: enthusiastic, entertaining, exciting, interesting, informative, inspirational and motivational. Although you are not expected to be professional speakers, you should definitely think about their criteria when you are preparing and giving a presentation. Following is a discussion of the presentation itself, its preparation, techniques for giving the presentation, and the use of visual aids. If you pick up only one of the ideas presented here, you will be become a better speaker.

1. BE YOURSELF. It is very important to be yourself when you are giving a talk. If you pretend to be someone else, you won't look natural. Face your audience. Smile. Don't start your talk with an apology; you want your audience on your side. If you are prepared, they will find out soon enough. Be positive about yourself and your talk.

2. DON'T READ YOUR PAPER. Nothing is more boring than to listen to someone read word-for-word from a prepared text. There are very few people who can do this and still keep their talk interesting. If you can't keep your talk interesting, you will put the audience to sleep. Use index cards if it will help. Use an outline or use your visual aids as a guide. Ad lib. Add any thoughts that may come to you as you are giving your talk and make your talk different every time. Keep the content flexible. For instance, Mr. Mullen slipped the information on the Professional Speaker

Awards into his talk.

3. ESTABLISH AUDIENCE CONTACT. Make the audience a part of your presentation. Establish eye contact. Find out who they are. What do you have in common with them? What language do they speak? What do they want?

4. INVOLVE YOUR AUDIENCE. Use the words "we" and "you". Establish objectives for them but keep them realistic. Don't have too many objectives. It is better to have one objective and present it well than to have too many of them presented poorly. Make sure that the objectives can be accomplished in a reasonable time frame. You need to know what is going to happen because you gave the presentation. Know where your talk is going. You don't want to be like Alice talking to the Cheshire cat:

"Would you tell me please which way I ought to walk from here?"

"That depends a great deal on where you want to get to," said the cat.

"Oh, I don't care much where," said Alice.

"Then it doesn't much matter which way you walk," said the cat.

"So long as I get somewhere," Alice added as an explanation.

"Oh, you're sure to do that," said the cat. "If you only walk long enough."

Our talks are the same way.

5. DO NOT ASSUME THAT ALL OF YOUR AUDIENCE UNDERSTANDS THE ACRONYMS AND BUZZWORDS THAT YOU USE. It is quite possible that you have had a speaker use a buzzword that you did not know at a symposium.

6. BEWARE OF THE PODIUM. As you approach the podium, take it easy. Watch for wires that could trip you. Remember that the podium hides you from the audience and limits what you can do. If you do use the podium, stand erect behind the podium and don't slouch. Don't do things that might distract your audience. Don't pace. Let your hands be natural and, if you must stick them in your pocket, make sure your pockets are empty. Jingling coins or keys is distracting to the audience. Don't play with the telescoping pointer.

7. REHEARSE, REHEARSE, REHEARSE. Do a dry run with someone beforehand. It isn't enough to simply go over your notes, you need to get on your feet and vocally rehearse your talk. You should be thoroughly familiar with your talk. Rehearse enough so that you feel you will be able to roll with the punches.

8. BE FAMILIAR WITH THE AUDIO/VISUAL EQUIPMENT. Know where the on/off switch is on the overhead projector. Make sure that all of the equipment is present. Have a backup. Try the equip-

ment out. Anticipate problems.

9. BE SURE THAT YOUR AUDIENCE CAN SEE THE SCREEN. If you will block the audience's view of the screen, be prepared to step aside when you use the screen. Will any lights wash out the screen?

1Ø. EXPEND TIME PREPARING YOUR VISUAL MATERIALS. Use white space effectively. Be careful of character sizes. Characters that are too small can not be seen. Keep the visuals simple by following the K.I.S.S. (Keep It Simple Stupid) policy. The more complicated your presentation, the more things that can go wrong. Keep the attention where you want it. After your audience has seen your visual, remove it and draw the attention back to you. If you must use a "busy" visual, use dark strips to introduce the visual to the audience in sections. Alternatively, you could highlight the important features of the visual. Put the audience's attention where you want it. After spending all of that time preparing a presentation, don't send it through the airport baggage. Carry it on with you. You may lose your luggage this way, but you won't lose your presentation materials.

11. AVOID DATED SLIDES OR FOILS. Keep your materials up-to-date or avoid putting dated items on the visuals.

12. REHEARSE WITH YOUR VISUALS. Make sure that they are in the proper order. Know when to use your visuals. Make sure that they come at the right time in your talk and that you are not talking about something when you are displaying another visual.

13. USE VISUALS AS A TOOL.

14. KEEP THINGS SIMPLE.

Mr. Mullen also had some additional remarks:

Don't talk in a monotone. Speed it up and slow it down. Add inflection and pauses to your talk. If you don't think you have the audience's attention, stop talking. Silence is a good way of getting their attention again. There are problems if handouts are passed out at the start of a talk. You will find that your audience will spend most of their time reading the handouts and missing much of what you have to say.

In a dark room, speak louder. For some reason, people cannot hear as well when the lights are dimmed.

Beware of the "opening joke" or the "opening story". You could kill your entire talk if it is not done well. In the opening part of your presentation, you want to capture your audience, you want to get their attention. *

RSTS V8.0 Technical Changes

by Fred Knight

RDSØ

```
      MFD                    UFD's                   FILES

    -------                 -------                 -------
/--|--      |               |       |------------->|       |
|  |        |-------------->|       |------\        -------
\->|        |               -------        |
   |        |                              |        -------
/--|--      |               --------       \------>|       |
|  |        |-------------->|        |----\         -------
\->|        |               --------      |         -------
    -------                               \------->|       |
                                                    -------

   LINKED LIST             LINKED LIST             DATA
```

RDS1

```
  MFD             GFD'S           UFD'S          FILES

 -------         -------         -------         -------
|       |       |       |------>|       |------>|       |
|       |------>|       |----\   -------         -------
|       |        -------     |   -------         -------
|POINTS |                    \->|       |------>|       |
|       |                        -------         -------
|  TO   |        -------         -------
|       |------>|       |------>|       |
|  THE  |        -------         -------
|       |
| GROUP |        -------         -------         -------
|       |       |       |------>|       |------>|       |
|       |------>|       |----\   -------         -------
|       |       |       |-\  |   -------         -------
|       |        -------  |  \->|       |------>|       |
|       |                 |      -------         -------
|       |                 |
 -------                  |      -------         -------
                          \---->|       |------>|       |
                                 -------         -------

DIRECT          DIRECT          LINKED          DATA
ACCESS          ACCESS           LIST
```

COMPATIBILITY

o V8.Ø WILL SUPPORT BOTH RDSØ and RDS1 DISKS

o DSKINT CREATES ONLY RDS1 DISKS

o V7.2 DOES NOT SUPPORT RDS1 DISKS

o IN-PLACE CONVERSION PROGRAM SUPPLIED WITH V8.Ø
(DSKCVT)

o 3RD PARTY SOFTWARE MAY NEED CHANGING

RDS1 VS. RDSØ

o UFD's ARE IDENTICAL

o MFD IS DIFFERENT
(Direct access and sorted)

o RDS1 SUPPORTS MAXIMUM ACCOUNTS
(64516 accounts)

o BETTER PERFORMANCE

o RPØ4/Ø5 INCREASE BY APPROXIMATLY 46ØØ BLOCKS

o RPØ6 INCREASES BY APPROXIMATLY 62ØØ BLOCKS

PACK LABEL

```
                                OFFSET
 -----------------------
!           1           !          Ø
!-----------------------!
!          -1           !          2
!-----------------------!
!      DCN OF MFD       !          4
!-----------------------!
! DISK VERSION NUMBER   !          6
!-----------------------!
!          PCS          !          1Ø
!-----------------------!
!        STATUS         !          12
!-----------------------!
!        PACK-ID        !          14
!- - - - - - - - - - - -!
!       IN RAD-5Ø       !          16
!-----------------------!
!       RESERVED        !          .
!      FOR FUTURE       !          .
!         PACK          !          .
!      ATTRIBUTES       !
 -----------------------           776
```

BLOCK 1 OF RDS1 DISKS

PACK STATUS WORD

```
  15    14    13    12    11    1Ø    9    8   7   6   5   4   3   2   1   Ø
 ---------------------------------------------------------------------------
!    !     !     !     !     !     !     !   !   !   !   !   !   !   !   !   !
!    !     !     !     !     !     !     !   !   !   !   !   !   !   !   !   !
!    !     !     !     !     !     !     !   !   !   !   !   !   !   !   !   !
 ---------------------------------------------------------------------------
  !     !     !     !     !     !     !    \_______________  _______________/
  !     !     !     !     !     !     !                    \/
  !     !     !     !     !     !     !                 RESERVED
  !     !     !     !     !     !     !
  !     !     !     !     !     !     \--  NFF
  !     !     !     !     !     !
  !     !     !     !     !     \--  RESERVED
  !     !     !     !     !
  !     !     !     !     \--  DLW
  !     !     !     !
  !     !     !     \--  READ ONLY PACK
  !     !     !
  !     !     \--  RDS1 PACK
  !     !
  !     \--  PRIVATE/SYSTEM
  !
  \--  MOUNTED
```

RDS1 MFD

```
 -----------------------------
|                             |
|    RESERVED BLOCKETTES      |
|           FOR               |      BLOCK Ø
|    FUTURE GROUP/PACK        |
|        ATTRIBUTES           |
|                             |
|-----------------------------|
|                             |
|      1 WORD/GROUP           |
|      DCN OF GFD             |      BLOCK 1
|                             |
|                             |
|                             |
|-----------------------------|
|                             |
|      1 WORD/GROUP           |
|    GROUP ATTRIBUTE          |      BLOCK 2
|        POINTER              |
|       (RESERVED)            |
|                             |
|-----------------------------|
|                             |
|    RESERVED BLOCKETTES      |
/           FOR               /      BLOCK 3-END
/    FUTURE GROUP/PACK        /
|        ATTRIBUTES           |
|                             |
 -----------------------------

   MFD CLUSTERSIZE = 4,8,16
```

```
        BLOCK Ø IN THE MFD

                                    OFFSET
 ----------------------
|          Ø           |              Ø
|----------------------|
|         -1           |              2
|----------------------|
|          Ø           |              4
|----------------------|
|          Ø           |              6
|----------------------|
|          Ø           |             1Ø
|----------------------|
| PACK ATTRIBUTE POINTER|            12
|----------------------|
|   255.   |   255.    |             14
|----------------------|
|   'MFD' IN RAD-5Ø    |             16
|----------------------|
|       RESERVED       |              .
/    FOR BLOCKETTES    /              .
/                      /              .
|----------------------|
| 1 |   CLUSTERSIZE    |            76Ø
|----------------------|
|   DCN Ø OF MFD       |            762
|----------------------|
|   DCN 1 OF MFD       |            764
|----------------------|
|   DCN 2 OF MFD       |            766
|----------------------|
|   DCN 3 OF MFD       |            77Ø
|----------------------|
|   DCN 4 OF MFD       |            772
|----------------------|
|   DCN 5 OF MFD       |            774
|----------------------|
|   DCN 6 OF MFD       |            776
 ----------------------
```

```
          BLOCK 1 IN THE MFD

                                 OFFSET
  -----------------------
  |   DCN FOR GROUP Ø GFD |        Ø
  |-----------------------|
  |   DCN FOR GROUP 1 GFD |        2
  |-----------------------|
  |   DCN FOR GROUP 2 GFD |        4
  |-----------------------|
  |   DCN FOR GROUP 3 GFD |        6
  |-----------------------|
 /            .           /        .
 /            .           /        .
  |-----------------------|
  | DCN FOR GROUP 253 GFD |      772
  |-----------------------|
  | DCN FOR GROUP 254 GFD |      774
  |-----------------------|
  |           Ø           |      776
  -----------------------

  CONTAINS THE DCN FOR EACH
  GROUP FILE DIRECTORY THAT
  EXISTS.  IF Ø, THEN THAT
  GROUP DOES NOT EXIST.
```

```
                        RDS1 GFD

   -------------------------------
   |                             |
   |       BLOCKETTES FOR        |
   |    UFD NAME/ACCOUNTING/     |        BLOCK Ø
   |      ATTRIBUTE ENTRIES      |
   |                             |
   |-----------------------------|
   |                             |
   |          1 WORD/USER        |
   |        DCN OF THE UFD       |        BLOCK 1
   |                             |
   |                             |
   |-----------------------------|
   |                             |
   |         1 WORD/USER         |
   |    NAME BLOCKETTE POINTER   |        BLOCK 2
   |                             |
   |                             |
   |-----------------------------|
   |                             |
   |       BLOCKETTES FOR        |
   /    UFD NAME/ACCOUNTING/     /        BLOCK 3-END
   /     ATTRIBUTE ENTRIES       /
   |                             |
   |                             |
   -------------------------------

         GFD CLUSTERSIZE = 4,8,16
```

```
        BLOCK Ø IN THE GFD

                                    OFFSET
     -------------------------
    !           Ø             !      Ø
    !-------------------------!
    !          -1             !      2
    !-------------------------!
    !           Ø             !      4
    !-------------------------!
    !           Ø             !      6
    !-------------------------!
    !           Ø             !      1Ø
    !-------------------------!
    !           Ø             !      12
    !-------------------------!
    !  GROUP # !     255.     !      14
    !-------------------------!
    !   'GFD' IN RAD-5Ø       !      16
    !-------------------------!
    !        RFSERVED         !      .
    /     FOR BLOCKETTES      /      .
    /                         /      .
    !-------------------------!
    ! 1 !    CLUSTERSIZE      !      76Ø
    !-------------------------!
    !   DCN Ø OF GFD          !      762
    !-------------------------!
    !   DCN 1 OF GFD          !      764
    !-------------------------!
    !   DCN 2 OF GFD          !      766
    !-------------------------!
    !   DCN 3 OF GFD          !      77Ø
    !-------------------------!
    !   DCN 4 OF GFD          !      772
    !-------------------------!
    !   DCN 5 OF GFD          !      774
    !-------------------------!
    !   DCN 6 OF GFD          !      776
     -------------------------
```

```
        BLOCK 1 IN THE GFD

                                    OFFSET
     -------------------------
    !  DCN FOR USER   Ø UFD  !        Ø
    !------------------------!
    !  DCN FOR USER   1 UFD  !        2
    !------------------------!
    !  DCN FOR USER   2 UFD  !        4
    !------------------------!
    !  DCN FOR USER   3 UFD  !        6
    !------------------------!
    /           .            /        .
    /           .            /        .
    !------------------------!
    ! DCN FOR USER  253 UFD  !       772
    !------------------------!
    ! DCN FOR USER  254 UFD  !       774
    !------------------------!
    !           Ø            !       776
     -------------------------
```

CONTAINS THE DCN FOR EACH
USER FILE DIRECTORY IN THIS
GROUP. IF Ø, THEN THAT USER
DOES NOT HAVE A UFD. THAT
ACCOUNT MAY STILL EXIST HOWEVER.

```
        BLOCK 2 IN THE GFD

                                     OFFSET
  -----------------------
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER Ø   |             Ø
 |-----------------------|
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER 1   |             2
 |-----------------------|
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER 2   |             4
 |-----------------------|
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER 3   |             6
 |-----------------------|
 /           .           /             .
 /           .           /             .
 |-----------------------|
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER Ø   |            772
 |-----------------------|
 |  NAME BLOCKETTE       |
 |  POINTER FOR USER Ø   |            774
 |-----------------------|
 |           Ø           |            776
  -----------------------

 CONTAINS THE NAME BLOCKETTE
 POINTER FOR EACH USER IN THIS
 GROUP.  IF Ø, THEN THAT ACCOUNT
 DOES NOT EXIST.
```

```
   BLOCK 3 TO END OF THE GFD

                                   OFFSET
  -----------------------
 |  8 WORD BLOCKETTE     |            Ø
 |-----------------------|
 |  8 WORD BLOCKETTE     |            2Ø
 |-----------------------|
 |  8 WORD BLOCKETTE     |            4Ø
 |-----------------------|
 |           .           |            .
 /           .           /            .
 /           .           /            .
 |           .           |
 |-----------------------|
 |  8 WORD BLOCKETTE     |            74Ø
 |-----------------------|
 | 1 |   CLUSTERSIZE     |            76Ø
 |-----------------------|
 |  DCN Ø OF GFD         |            762
 |-----------------------|
 |  DCN 1 OF GFD         |            764
 |-----------------------|
 |  DCN 2 OF GFD         |            766
 |-----------------------|
 |  DCN 3 OF GFD         |            77Ø
 |-----------------------|
 |  DCN 4 OF GFD         |            772
 |-----------------------|
 |  DCN 5 OF GFD         |            774
 |-----------------------|
 |  DCN 6 OF GFD         |            776
  -----------------------
```

```
        GFD NAME BLOCKETTE                GFD ACCOUNTING BLOCKETTE

                              OFFSET
  -----------------------              -----------------------
 |  ATTRIBUTE POINTER    |      Ø     |           1           |
 |-----------------------|            |-----------------------|
 | GROUP # |  USER #     |      2     |    CPU TIME (LSB)     |
 |-----------------------|            |-----------------------|
 |           Ø           |      4     |     CONNECT TIME      |
 |-----------------------|            |-----------------------|
 |           Ø           |      6     |     KCT'S (LSB)       |
 |-----------------------|            |-----------------------|
 |   PROT  |   STATUS    |      1Ø    |     DEVICE TIME       |
 |-----------------------|            |-----------------------|
 |       ACCESS COUNTS   |      12    |  CPU & KCT'S (MSB)    |
 |-----------------------|            |-----------------------|
 |  ACCOUNTING POINTER   |      14    |   LOGGED OUT QUOTA    |
 |-----------------------|            |-----------------------|
 |     DCN OF UFD        |      16    |    UFD CLUSTERSIZE    |
  -----------------------              -----------------------

 DCN MUST AGREE WITH BLOCK 1
```

GFD ATTRIBUTE BLOCKETTE

```
 ------------------------
 !   POINTER TO NEXT    !        Ø
 !----------------------!
 !           !  TYPE    !        2
 !            ----------!
 !                      !
 !                      !
 !                      !        .
 !                      !
 !                      !        .
 !                      !
 !     ATTRIBUTE        !        .
 !                      !
 !                      !
 !        DATA          !
 !                      !
 !                      !
 !                      !        16
  ------------------------
```

TYPE 3 = ASCII PASSWORD
TYPE 4 = ACCOUNT DATE/TIME
TYPE * = RESERVED FOR FUTURE

STANDARD LINK WORDS

```
 15  14  13  12  11  1Ø   9   8   7   6   5   4   3   2   1   Ø
 ---------------------------------------------------------------
!   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !
!   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !
!   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !   !
 ---------------------------------------------------------------
 \_____ _______/ \___ ____/ \_______ ________/ \_____ _______/
       \/            \/             \/              \/
    BLOCK #      CLUSTER #        ENTRY #        FLAGS OR Ø
```

GENERAL MONITOR CHANGES

INIT.SYS CHANGES

o BOOTSTRAPS RE-WRITTEN
(to be able to support new disk types)

o FILL OPTION UNSUPPORTED

o UNISYS OPTION REMOVED

o LOAD OPTION UNSUPPORTED

PROGRAM SIZE

PROGRAMS MAY NOW EXPAND UP TO 32k WORDS.

INITIALIZE MODE FOR DISKS

NFS MODE 512. ALLOWS READING PAST LOGICAL END OF DISK. THIS IS USED TO READ FACTORY BAD BLOCK INFORMATION BY THE 'DSKINT' PROGRAM. IT ALSO CAUSES ERROR LOGGING TO BE SUPPRESSED.

ACTIVITY LOGGING

MONITOR CAN RETURN INFORMATION TO A USER WRITTEN PROGRAM ABOUT SELECTED REQUESTS ISSUED BY ANY PROGRAM ON THE SYSTEM.

NO-STALL MODE FOR KB'S AND LP'S

DRIVER RETURNS COUNT OF CHARACTERS NOT WRITTEN TO DEVICE INSTEAD OF STALLING PROGRAM UNTIL COMPLETION. USED BY THE NEW SPOOLING PACKAGE TO PREVENT STALLING.

NEW SYS-CALLS

NAME	CHANGE
CREATE ACCOUNT	PRE-EXTENSION & PLACEMENT OF UFD
DISK MOUNT	SINGLE USER MODE (/NOSHARE) & OVERRIDE READONLY
CLEAN DISK	REMOVED - USE ONLCLN
LOGOUT	ENFORCES QUOTA ON ALL DISKS CAN SPECIFY NO CLEANUP (CLOSE, DEASSIGN, ETC...) NON-PRIVED SELF-KILL
ATTACH	SWAP CONSOLE
DETACH	SWAP CONSOLE
ZERO ACCOUNT	DOES NOT RELEASE UFD CLUSTERS (OPTIONAL IF PRIVED)
READ ACCOUNTING DATA	ACCEPTS WILDCARD PPN'S
SEND/RECEIVE	NON-PRIVED DECLARE (USING FEATURE PATCHABLE QUOTAS) (USED BY RMSV2.Ø FOR REMOTE FILES)
SPAWN	NON-PRIVED CREATES JOB LOGGED INTO SAME ACCOUNT

CUSPS

NAME	CHANGE
DISTRIBUTION	.BAS AND .TSK'S DISTRIBUTED TO SPEED INSTALLATION
SYSGEN	MAXIMUM NUMBER OF LOGICALS INCREASED TO 100 ACTIVITY LOGGER OPTION PR:/PP: QUESTIONS COMBINED
BUILD	NEW QUESTIONS FOR BP2 WHEN CREATING ACCOUNTS, PRE-EXTENDS BY 1 CLUSTER
SYSTAT	SHOWS DEVICE FROM WHICH RUN-TIME-SYSTEMS AND RESIDENT LIBRARIES WERE ADDED SHOWS DISK STRUCTURE LEVEL
ONLCLN/MOUNT	DCL COMPATIBLE SYSTAX DCL DISK MOUNT PROGRAM REBUILD (CLEAN) AT SAME TIME AS MOUNT MOUNT /NOSHARE MOUNT /WRITE
BACKUP/DIRECT	USE WILDCARD PPN TO LOOKUP ACCOUNTS DIRECTORY LISTINGS SORTED BY PPN ON RDS1 DISKS BACENT PRE-EXTENDS DIRECTORIES BY 1 CLUSTER
REACT	NEW QUESTIONS FOR PLACEMENT AND PRE-EXTENSION ZEROS ACCOUNT DURING DELETE NEW DATA ITEMS STORED IN '$ACCT.SYS' CAN READ OLD OR NEW '$ACCT.SYS' FILES
DSKINT	CREATES RDS1 FORMAT DISKS WRITTEN IN MACRO READS FACTORY BAD BLOCK INFO DOES PATTERN CHECKS ERASES DISK IF NO PATTERNS SELECTED
DCL	INIT COMMAND FOR DISKS USES 'DSKINT' MOUNT COMMAND FOR DISKS USES 'ONLCLN' MORE LINK COMMANDS
LOGOUT	ENFORCES QUOTAS ON ALL READ/WRITE MOUNTED DISKS
LOGIN	ENFORCES QUOTAS WHEN CHANGING ACCOUNTS ;PASSWORD - REMOVED FOR SECURITY
MONEY	ACCEPTS WILDCARDS RDS1 LISTINGS AUTOMATICALLY SORTED BY PPN ALLOWS ONE LINE COMMANDS CAN BE CHAINED IN TO AND OUT OF

MICRO RSTS V8.0 Spooling Package

by Paul R. Laba

Features·

o Runs as a single detached job in 13KW memory (plus 8KW using RMS resident library version), or 22KW (using overlaid version).

o Runs significantly faster than current spooling package. Written entirely in MACRO.

o Supports concurrent printing on up to four printers (LP or KB devices).

o Prints files of any RMS format, as well as standard stream ASCII files.

o Uses a single interface program for both user and operator commands. Syntax is fully supported in DCL.

o Provides acknowledgement of all user and operator commands, with improved error reporting.

o Uses a single RMS indexed file for queued jobs and internal work entries. Improved queue management.

o Provides a Forms Definition File for maintaining all necessary form attributes.

o Supported from existing UU.SPL monitor directive.

o User interface program can be accessed via DCL, a CCL, or RUN command.

o May be run concurrent with the standard spooling package.

Communication Paths

```
                                   S P L . T S K
                         + - - - - - - - - - - - - - - - - - +
                         |                                   |     __________
                         |                    +-----------+  |    / KB or LP |
                         |                +-->| Spooler #Ø|-----> |__________|
Q U E U E . T S K        | +----------+   |   +-----------+  |         ||
+ - - - - - - - - +      | | Queue    |<--+        :         |         ||
|      User       |<------>| Manager  |<--+        :         |      =======
|    Interface    |      | +----------+   |        .         |
+ - - - - - - - - +      |                |                  |     __________
         ^               |                |   +-----------+  |    / KB or LP |
         |               |                +-->| Spooler #3|-----> |__________|
         |               |                    +-----------+  |         ||
         |               + - - - - - - - - - - - - - - - - - +         ||
         |                                                          =======
         v
      _________
    / User KB |
    |_________|
         ||
         ||
      =======
```

File Access Paths

```
_SY:[1,2]FORMS.SYS   _SY:[1,2]QUE.SYS                       LB·RMSERR.ERR

   __________          __________          __________          __________
  /          \        /          \        /          \        /          \
 /   Forms    \      /            \      /    Disk    \      /    RMS     \
|  Definition  |    |    Queue     |    |   File(s)    |    |    Error     |
|     File     |    |     File     |    |   to print   |    |     File     |
 \            /      \            /      \            /      \            /
  \__________/        \__________/        \__________/        \__________/
       ||               || /\   ||=====||       ||                  ||
       ||               || ||          ||       ||===========||     ||
       ||       ||======|| ||====||    ||                    ||     ||
       ||       ||               ||    ||==============||    ||     ||
       \/       \/               \/                    \/    \/     \/
   +----------------+       +----------------+       +----------------+
   |                |       |                |       |                |
   |      User      |       |     Queue      |       |     Print      |
   |   Interface    |       |    Manager     |       |    Spoolers    |
   |                |       |                |       |                |
   +----------------+       +----------------+       +----------------+
```

Sub-Job Scheduling

```
    Queue                                                          Message
   Manager       Spooler #Ø      Spooler #3       Null Job       Transmitter
+----------+    +----------+    +----------+    +----------+    +----------+
|          |    |          |    |          |    |          |    |          |
| "SPL.QM" |    | "SPL.SØ" |....| "SPL.S3" |    | "SPL..." |    | "SPL.XM" |
|          |    |          |    |          |    |          |    |          |
+----------+    +----------+    +----------+    +----------+    +----------+
     /\              /\              /\              /\              /\
     ||              ||              ||              ||              ||
     ||==============||==============||==============||==============||
                                     ||
                                     ||
                +------------------------------------------+
                |                                          |
                |             Sub-Job Schedular            |
                |                                          |
                +------------------------------------------+
```

User Commands

o PRINT

Submit one or more disk files to be printed.

```
$ PRINT $NOTICE.TXT  MYJOB
Job #37 entered on queue PRINT
```

Command qualifiers:

```
        /PRIORITY=n      /AFTER=date:time

        /JOB_COUNT=n     /FORMS=form-name
```

File qualifiers:

```
        /COPIES=n        /[NO]DELETE

        /[NO]TRUNCATE    /[NO]CONVERT

        /[NO]FEED        /[NO]FLAG_PAGFS
```

o DELETE/JOB

Cancel one or more print jobs on the queue by job specification.

```
$ DELETE/JOB  PRINT:[10,10]*
3 jobs PRINT·[10,10]????????? deleted
```

----

o DELETE/ENTRY

Cancel one or more print jobs on the queue by entry (job) number.

```
  $ DELETE/ENTRY 10,11
  Job #10 deleted
  Job #11 deleted
```

----

o SHOW QUEUE

Display jobs either currently running or waiting to run. Jobs to be displayed can be selected by

o queue name

o job name

o owner (PPN)

o form name

The default SHOW QUEUE command displays a brief listing of all of your jobs in the queue:

```
$ SHOW QUEUE

            Listing of jobs PRINT:[10,10]?????????

Job 17  [10,10]  MYJOB  Status AFTER  Pri 128  Form NORMAL  Pos 10th
Job 19  [10,10]  TEST   Status READY  Pri  64  Form CHECKS  Pos 12th
```

You can use the /FULL qualifier to obtain a complete status of your jobs:

```
$ SHOW QUEUE/FULL

            Listing of jobs PRINT:[1Ø,1Ø]?????????

Job 17  [1Ø,1Ø]  MYJOB  Status AFTER  Pri 128  Form NORMAL  Pos 1Øth
  Submitted 17-Jan-83 Ø9:17 am, Start after 17-Jan-83 11:59 pm
    _SY:[1Ø,1Ø]MYFILE.LST /COP=2
    -SY:[1Ø,1Ø]WRKFIL.TMP /COP=2 /DEL
```

You can select jobs by PPN or job name:

```
$ SHOW QUEUE [1Ø,*]TEST

            Listing of jobs PRINT:[1Ø,*]TEST

Job 16  [1Ø,12]  TEST   Status STARTED Pri 128  Form NORMAL  Pos 8th
Job 19  [1Ø,1Ø]  TEST   Status AFTER   Pri 64   Form CHECKS  Pos 12th
Job 25  [1Ø,31]  TEST   Status READY   Pri 128  Form NORMAL  Pos 18th
```

You can select jobs by form name:

```
$ SHOW QUEUE/ALL/FORMS=CHECKS

        Listing of jobs PRINT:[*,*]????????? with form CHECKS

Job 15  [ 1,2 ]  BILL   Status READY   Pri 128  Form CHECKS  Pos 5th
Job 19  [1Ø,1Ø]  TEST   Status AFTER   Pri 64   Form CHECKS  Pos 12th
```

Operator Commands

o START/QUEUE/MANAGER

Start up the spooling services package. Causes a single detached job to be spawned.

```
$ START/QUEUE/MANAGER
Spooling Services started at 1Ø:51 AM
```

o STOP/QUEUE/MANAGER

Shut down the spooling services package, either immediately or at the completion of all active jobs.

```
$ STOP/QUEUE/MANAGER
Spooling Services shutting down immediately
```

o INITIALIZE/PRINTER

Define a device for printing. Up to four devices (LP or KB) may be defined at any one time.

```
$ INITIALIZE/PRINTER/FORMS=CHECKS/NOSHAREABLE  KB33:
Printer  KB33: initialized nonshareable with form CHECKS
```

o DELETE/PRINTER

Remove a printer from the list of defined print devices. Another device can then be initialized.

```
$ DELETE/PRINTER KB33:
Printer _KB33: deleted
```

o STOP/PRINTER

Stops printing on a specified printer, either immediately, at the end of the current file, or at the end of the current job.

```
$ STOP/PRINTER LPØ:/JOB_END
Printer _LPØ: marked as stopped
```

o START/PRINTER

Resume printing following a STOP/PRINTER command.
You can indicate that you want to resume printing

- o where it was stopped
- o at start of next job
- o at start of current job
- o at relative page number (forward or backward)
- o with new form
- o at start of current file
- o at absolute page number

You can resume printing at the point where printing was stopped (default):

```
        $ START/PRINTER LPØ:
        Printer _LPØ: marked as started
```

You can resume printing with a new form. New form attributes do not take effect until the start of a new job:

```
        $ START/PRINTER LPØ:/FORM=CHECKS
        Printer _LPØ: marked as started with form CHECKS
```

You can resume printing at the next job, causing any current job to be aborted:

```
        $ START/PRINTER LPØ:/NEXT_JOB
        Printer _LPØ: marked as started
```

You can resume printing at the start of the current file copy:

```
        $ START/PRINTER LPØ:/TOP_OF_FILE
        Printer _LPØ: marked as started
```

You can resume printing at the start of the current job copy:

```
        $ START/PRINTER LPØ:/RESTART
        Printer _LPØ: marked as started
```

You can resume printing at a specified page in the current file copy·

```
$ START/PRINTER LPØ:/PAGE=25
Printer _LPØ· marked as started
```

You can resume printing at a page relative to the page where printing was stopped·

```
$ START/PRINTER LPØ:/BACKSPACE=3
Printer _LPØ: marked as started

$ START/PRINTER LPØ:/FORWARDSPACE=1Ø
Printer _LPØ: marked as started
```

Restrictions

o No BATCH processing

The current package only provides print services. Batch processing is planned for a future release.

o No job modification

The SET QUEUE command has not yet been implemented.

o No multiple queues

Only the (generic) queue PRINT is recognized in the first release.

o No definable operators

You must be privileged, i.e., running from a privileged account to issue any of the operator commands.

o No operator services console

No communication services (e.g. OPSER) is provided with the first release of the package.

o No printing of remote (DECnet/E) files

Files to be printed must reside on the (local) system from which the PRINT command is issued.

o No status display of spoolers

Only the status of jobs on the queue can be displayed.

o No definable job characteristics

Only the characteristic /FORMS is defined in the first release.

o No forms alignment procedures

Form alignment templates are being considered for a future release.

o No loadable fonts for printers

Support for automatic setup of programmable printers is being considered for a future release.

Spring '83 Symposium Handouts from the RSTS Developers

Memory Usage on RSTS Systems

by Jim Carey

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

1.Ø INTRODUCTION --

We define large memory RSTS/E systems as meaning that you have plenty of memory to do whatever you want with it. The problem becomes one of striking an optimal balance between all of the RSTS performance parameters rather than just making do with what you can get. This discussion deals with RSTS requirements, and the performance tradeoffs to be considered by a the system manager in order to optimize his system for its operating environment.

2.Ø ELEMENTS IN MEMORY --

o Your RSTS Monitor SIL

o The Default RTS

o XBUF (extended buffering for the monitor)

o Libraries

o Other RTS's

o Jobs

3.Ø SIZE REQUIREMENTS OF ALLOCATED MEMORY

3.1 Your Monitor SIL

The RSTS SIL requires memory starting at Ø. Sizes of RSTS monitors vary from about 35K up to systems containing over 7ØK of resident code. The size of the monitor varies according to the hardware configuration it was genned for, the software features selected, and the number of small buffers you have selected from both the SYSGEN dialogue, plus any additional small buffers added in INIT.SYS.

3.2 Selecting the small huffer count --

Some rules of thumb·

- o Go for 1Ø buffers per job (this is an extremely system dependent number, 1Ø is good for a first stab).
- o Try to keep about 1ØØ free buffers (FIP pool and general pool) available during peak use hours.
- o Experiment. The BUFFERS suboption of INIT's DEFAULT option is a very quick way of varying this number.

RSTS use of small buffers:

- o 3 small buffers required for every job on the system. Taken from General pool.
- o The first time a file is opened it requires 2 small buffers (FCB and WCB), additional opens on that file only require 1 small buffer per open. These buffers are allocated from the FIP pool.
- o 1 buffer is required for each Receiver ID (RIB). It comes from the general buffer pool.
- o 1 buffer is used by each PMB. General pool.
- o An active terminal may use 4-6 small buffers.
- o A CCL uses up one small buffer from the FIP pool.
- o Other uses of small buffers include DSQ's for disks, and some buffering for the line printer driver.

Reasonable numbers of small buffers for your system may vary from six or seven up to twenty-five or more per job, according to everything about your system.

Some obvious things to consider:

- o How many files does the average job have open?
- o How many CCL's do you use?
- o How many receivers are declared, and how much activity will they see?
- o Keep in mind that TOO MANY small buffers will also impact performance, but not nearly as much as too few.

3.3 XBUF -- RSTS extended buffer space

Uses for XBUF:

- o LPDVRX -- uses XBUF. We estimate that at most it will use about 2K.
- o DECNET uses XBUF for·
  - line receive buffers (about three per Kw)
  - Routing -- (max circuits + 2) * (max nodes + 2) = number of words used.
  - Pending transmit and receive buffers.

  Two circuits on a three node network would use about 3K of XBUF, and five circuits on a 250 node network would use something on the order of 10K to 15K of XBUF.
- o FMS uses XBUF for its control structures, and requires about 3Kw/16 terminals on the system.
- o Send/Receive uses XBUF. Extremely system dependant. Messages are kept in XBUF waiting to be serviced. Variables include:
  - System loading.
  - Priority of receivers.
  - Size and number of pending messages.
- o Caching -- data and directory
- o RJ2780
- o EMT logging (a new feature for V8.0) also uses XBUF for messages waiting to be logged. Varies with amount of logging requested.

Some rules of thumb for XBUF·

- o Use less than 10% of total memory if not performing Data Caching.
- o Use up to 25% of memory if performing data caching.
- o Don't use more than 100Kw of XBUF unless performing extensive data caching.
- o Try to keep enough memory left as user space for 1/4 to 1/3 of all jobs on the system to be in memory at any given time.

4.Ø PLACEMENT IN MEMORY OF RSTS STRUCTURES --

Keep in mind that the goal of carefully placing these structures is to make the user job space in memory as contiguous as possible. A single large chunk of memory can be allocated more completely than several chunks adding up to the same amount of actual memory.

4.1 The monitor

It will go from Ø up to wherever its top is.

4.2 The Default RTS

Normally it is fine to leave this where INIT.SYS automatically places it. It must be placed between Ø and 124K in memory.

4.3 XBUF

Placing XBUF in V8.Ø is a little bit more challenging because RSTS requires 8Kw of contiguous space between 6ØK and 124K in for a monitor loader. This loader can't use XBUF space, or monitor memory. The loader can be put in place of the DEFAULT RTS, or in normal user memory. Also remember that XBUF must be allocated between the top of the monitor, and 512K in memory.

There are three cases to consider·

1) Monitor resident size greater than 6ØK and Default RTS size greater than 8K (for example, BASIC-PLUS). Allocate XBUF from the top of the space required for the RTS, as the loader will fit fine in the RTS area.

2) Memory size 512K or less. Set XBUF so that it is pushed up against the top of memory, and user space is between the monitor and XBUF. This means no fragmentation and plenty of room for the loader.

3) More memory than 512K, and not room in RTS space for loader between 6ØK and 124K. Set XBUF so that its top is at 512K. This will break the memory into two large chunks, which can be more effectively used than a large piece, and a small one.

4.4 Libraries --

Libraries are only in memory when in use unless marked with /STAY. Use this switch only for very heavily used libraries.

Consider frequency and concurrency of use. Place most heavily used libraries on the edge of user space: either at the top of memory, or close to XBUF and the Default RTS. Place other libraries in descending order of use, leaving the least used libraries farthest into user space. This will reduce the fragmentation of user space due to active library positioning.

4.5 Other RTS's

o With RTS's, generally allowing RSTS to handle placement will work better than assigning addresses, and forcing /STAY.

o Mark /STAY only if permanently in use.

o Add Run-time systems in proper order (.BAC,.TSK,.SAV,....,.TEC, then other .BAC's,etc.), and put the most heavily used RTSs first.

4.6 User Job's

Only additional aid is to install the First Fit Feature Patch. If you have lots of memory, your system will perform better. In V7.2 it is patch 3.1.2 F: in V8.Ø it is patch 3.1.1 F.

RSTS Disk Performance Optimization

by Gary Barton

The basic goal is to minimize the number of disk seeks and the distance moved on each one.

Every system is different and requires different strategies for optimum performance.

Observation

Planning

Experimentation

This talk will touch upon the following items:

- o Cluster size
- o Pack characteristics
- o Placed files
- o Directories
- o Contiguous files
- o Pre-extended files
- o REORDR
- o BACKUP and RESTORE
- o Data caching
- o Multiple disk systems
- o Monitor overhead

Explanation of terms

- o Cluster
- o Device Cluster
- o Device Cluster Size
- o Pack Cluster
- o Pack Cluster Size
- o File Cluster
- o File Cluster Size

A small cluster size means:

- o More retrieval entries
- o Bigger directories
- o Less wasted space

A large cluster size means:

- o Wasted space
- o May not be able to extend the file
- o More contiguous disk blocks
- o Lower FIP overhead

What is a good cluster size?

filesize / 7 rounded up to
a power of 2

Why is this a good cluster size?

Spring '83 Symposium Handouts from the RSTS Developers

PACK CHARACTERISTICS ala DSKINT

Pack cluster size?

o A large cluster size reduces the size of SATT.SYS

o This sets the minimum cluster size for any file or directory.

SATT.SYS Base?

o Yes, choose this carefully

MFD cluster size?

o If you have the space it should be 16.

Pre-extend directories?

o Yes, placed directories should always be pre-extended.

Create account [1,N]?

o No, do this later.

Date last modified?

o Penalty for date of last access.

o Possible gain through periodic REORDRing.

New files first?

o Pack conditions

o REORDR more often

PLACED FILES

The 'center' of a disk

- o Where is the center of the disk?
- o Heavily accessed files
- o Is the center really best?

Directories

- o Placement and pre-extension
- o Directories known in advance
- o Directory cluster size
- o Small directories
- o Ordered directories

What files should be placed?

- o SATT.SYS
- o Swap files
- o UFD's and GFD's
- o Frequently used [1,2] files
- o LIB's and RTS files
- o OVR, ERR
- o Others

Contiguous files

- o Cluster size
- o Directory overhead
- o Permanent files
- o Free space

Pre-extend files wherever desirable.

Multiple disk systems

o Separate active files

o Avoid multi-disk public structures

Monitor overhead

o FIP does things one at a time.

o SAT searches can be expensive.

REORDER

o Done on a regular basis.

o Places name blockettes in as few disk blocks as possible.

o Attempts to make retrieval pointers more contiguous.

o Optionally sorts the directories.

BACKUP and RESTORE

o Done on a regular basis.

o All files are implicitly contiguous.

o Files are reasonably well ordered.

o Free space is contiguous.

Data caching

o Size of the cache

o Random files caching

o Sequential file caching

o Guidelines

MICRO RSTS V8.0

by Nancy V. Burns

- Tailored for new Micro PDP-11 hardware
- "Application-only" subset of RSTS/E V8.0
- Distributed as part of RSTS/E V8.0
- Supplied pre-sysgened (can be installed in less than one hour)
- Built on a host RSTS/E system and transferred to micro PDP-11 on a transfer disk
- Occupies a maximum of 2.9 MB (29%) of the micro PDP-11's system disk
- New Micro RSTS Application Developer's Guide available

Why Micro RSTS?

RSTS/E:	7.0 MB (70% of RL02 or RD51)
Micro RSTS:	2.9 MB (29% of RL02 or RD51)

10 MB is too small to hold RSTS/E, application code, and user data. Therefore, the RSTS/E subset - Micro RSTS - is provided.

The Micro PDP-11 Hardware

- 11/23-PLUS CPU board (KDF11B)
- 256KB Q-bus memory (MSV11-P)
- 2 DL terminal lines
- 4-line terminal multiplexer (DZV11)
- 10 MB Winchester disk (RD51)
- 2 x 400 KB floppy disks (RX50)
- Integrated in a very small package

Optional Micro PDP-11 Hardware

- Two additional 4-line multiplexers (14 lines total)
- RL02 disk subsystem (10 MB per drive)
- Additional Q-bus memory (256KB / 512KB boards)

Monitor Configuration

Terminals:

- 2 DL lines
- 3 DZV11 multiplexers
- 4 Pseudo-keyboards
- No 2741 support
- Multi-terminal service
- Echo-control
- One-line status report (CTRL/T)
- FMS support
- Multiple private delimiters

Disks:

- RD51/RX50 mass storage subsystem
- 4 RL02 drives

Miscellaneous:

- One printer
- No DECnet/E network support
- No RJ2780 support
- No monitor statistics
- RSX as the default run-time system
- RSX directives

Features:

- 10 jobs maximum
- 300 small buffers
- 50 system-wide logicals
- FIP buffering
- Directory caching
- Resident libraries
- All FIP overlays resident

Micro RSTS V8.0 Software

- Monitor (MICRO.SIL) and INIT.SYS
- RT-11, BASIC, and RSX run-time systems
- RMS resident library
- DCL keyboard monitor.
- RT-11 cusps:

    PIP.SAV
    ONLCLN.SAV

- Standard cusps:

    LOGIN
    UTILTY
    SYSTAT
    LOGOUT
    INIT
    REACT
    SHUTUP
    TTYSET
    DSKINT

- Error package (ERRDIS, ERRCPY, ERRDET, ERRINT)
- Backup package (BACKUP)
- Text editor (EDT)
- Micro-RSTS spooling package (SPL)
- Sort package (SORT)
- Installation Certification Procedure (ICP)

DECUS U.S. Chapter Contacts

FOR INFORMATION ON:

Ordering Material from DECUS:

For general information before placing an order,
contact Order Processing . . . . (617) 467-4135

For information about an order you have *placed but have not yet received,* contact Order
Processing . . . . (617) 467-4135

For information about an order you have *placed and have received,* contact DECUS Library . . . . (617) 467-4254

For authorization to return defective media,
contact the DECUS Library . . . . (617) 467-4254

Library Information:

For information on how to submit a program for inclusion into the DECUS Library, or to check on the status of a newly submitted program,
contact the Submissions Coordinator . . . . (617) 467-4177

Membership Information:

To become a member, to check membership status, or to notify DECUS of a change of address or other membership information,
contact DECUS Membership Group . . . . (617) 467-4168

DECUS SYMPOSIA, contact DECUS Meetings Planner . . . . (617) 467-7469

DECUSCOPE, Newsletters, and other DECUS Publications,
contact Publications Administrator . . . . (617) 467-4143

DECUS Special Interest Groups and Local Users Groups . . . . (617) 467-4889

DECUS SUBSCRIPTION SERVICE ORDER FORM

RETURN TO: Subscription Service
DECUS
One Iron Way, MRO2-1/C11
Marlboro, MA 01752

- All checks payable to DECUS
- All orders MUST be paid in full
- No refunds will be made
- Prices indicated are FY'84 prices

Check # ____________
Bank # ____________
Amount $ ____________

Name ______________________________ DECUS Membership No. ____________
(First) (Last)

Company/Affiliation ______________________________

Mailing Address ______________________________ Mail Stop ____________

City ____________ State/Country ____________ Zip Code ____________ Phone ( ) ____________

PUBLICATIONS SUBSCRIPTION SELECTIONS

CODE	PUBLICATION	CODE	PUBLICATION
MSL	MUMPS/STRUCTURED LANGUAGES NEWSLETTER	RST	RSTS NEWSLETTER
LHS	LABS/HMS/SITE MGMT NEWSLETTER	LGS	LARGE SYSTEMS NEWSLETTER
OAD	OA/DIBOL/COBOL/GRAPH NEWSLETTER	EDU	EDUSIG NEWSLETTER
VAX	VAX/VMS NEWSLETTER	DTR	DATATRIEVE NEWSLETTER
RSX	RSX/IAS NEWSLETTER	NTW	NETWORKS NEWSLETTER
RT	RT11 NEWSLETTER	SOS	SS&OS NEWSLETTER
SPR	Spring Proceedings	BAS	BASIC NEWSLETTER
FAL	Fall Proceedings	APL	APL NEWSLETTER
		ALL	ALL PUBLICATIONS PRODUCED

	Insert Code From Above:	Check One:		
BASIC PLAN: This plan allows you to receive one (1) selection for one year	□	□	Member/DIGITAL Employee	$ 12.00
		□	Non Member	$ 24.00
STANDARD PLAN: This plan allows you to receive up to three (3) selections at one low price.	□ □ □	□	Member/DIGITAL Employee	$ 25.00
		□	Non Member	$ 50.00
DELUXE PLAN: This plan allows you to receive up to six (6) selections for one year.	□ □ □ □ □ □	□	Member/DIGITAL Employee	$ 45.00
		□	Non Member	$ 90.00
ALL: This will allow you to receive all publications listed above for one year for only one price.	ALL	□	Member/DIGITAL Employee	$120.00
		□	Non Member	$240.00

TOTAL AMOUNT OF ORDER $____________

I understand that neither DECUS nor Digital Equipment Corporation is responsible for any publication not published by a Special Interest Group or the contents of any publication published by a Special Interest Group. I also understand that there will be no refunds even if I decide to cancel my subscription.

Signature ______________________________ Date ____________

DIGITAL Employees Only: Badge No. ______________________________ C.C. ____________

Cost Center Manager's Signature ______________________________ C.C. ____________

INPUT/OUTPUT Submission Form

A SIG Information Interchange

Please reprint in the next issue of The Cache Buffer

Caption: ____

Message: ____

____

____

____

____

____

____

____

____

____

____

Contact:

Name ____

Address ____

____

Telephone ____________________________

If this is a reply to a previous I/O, which number? ______

Signature ______________________ Date ______________________

Mail this form to: Editor, THE CACHE BUFFER, DECUS, MRO2-1/C11,
One Iron Way, Marlboro, MA Ø1752 USA

Tear out to submit an INPUT/OUTPUT item

Editor, THE CACHE BUFFER
DECUS, MRO2-1/C11
One Iron Way
Marlboro, MA Ø1752
USA