THE mini-tasker

DECUS
RT-11 SIG NEWSLETTER

AUGUST 1978 VOL. 4 NO. 3

Contributions to the newsletter and other correspondence should be sent to:

Ken Demers
MS-48
United Technologies Research Center
Silver Lane
East Hartford, Conn. 06108

or

RT-11 SIG
C/O DECUS
129 Parker Street
PK-3/E55
Maynard,Mass. 01754

FROM THE EDITOR

I would personally like to thank John Rasted for the excellent job he has done as editor of the Mini-Tasker. Our RT-11 SIG will continue to have strong leadership as John takes over for Tom Provost, who is now devoting his efforts in the Hardware Hints & Kinks area.
The RT-11 SIG has funding available to publish six issues this fiscal year. I look forward to receiving any comments or critisicm that will make this newsletter more receptive to RT-11 user needs.
The make-up of each newsletter will be a combination of the following areas: 1. From The Editor 2. User Requests 3. User Responses 4. User Input 5. DEC Input 6. Upcoming Symposium Information 7. Past Symposium Information 8. RT-11 Marketplace 9. RT-11 Patches and 10. Software Performance Reports.

KLINIEK VOOR NEUROLOGIE
RIJKSUNIVERSITEIT
GRONINGEN

Harry Haenen, dept. of neurology
University Hospital of Groningen
Oostersingel 59
GRONINGEN - HOLLAND

GRONINGEN, 8 March 1978.
OOSTERSINGEL 59
TELEFOON 050 - 13 91 23
TOESTEL

Tom Provost
MIT/LNS Bates Linear Accelerator
P.O. Box 95
MIDDLETON, Mass. 01949
U.S.A.

Dear Tom,

In our lab we have a PDP-11/34 configured with a digital plotter CALCOMP 565 and an XY-11 as interface. For software we own the PLOT-11 DEC-package under RT-11 V2C.

In a real-time environment this package is completely unsatisfactory, because it is too big (4 KW routines, a 1000 point integer buffer requires 4 KW plotbuffer space), too slow (internal real-to-integer conversion) and it suspends main-line processing during plotting. In the meantime I wrote some routines for integer buffer plotting, which are extremely fast and efficient.

I am directing myself to you for the following two questions:

1. Could you contact me with other users who had the same problem and found solutions or are looking for it ?
2. Is it not necessary that DEC changes its PLOT-11 Software Product Description as it is completely unsatisfactory in a real-time environment on a small system, and helps users by providing the sources ?

Thanking you in advance, I remain,

Harry Haenen, dept. of neurology.

To:
RT-11 SIG
Subject:
Help Mini-Tasker
I need a credit union or savings and loan program to run on a PDP 11/34. Anyone that knows where I can get something using M.U. Basic/RT-11,VT55,RK05 Disks,Teletype Model 40 Printer contact

Wayne Diffee B.A.
Dept. Head Pulmonary Function Lab.
Rose Medical Center
4567 East 9th Ave.
Denver, Colorado 80220

USER RESPONSES

Research Center, 71 Frankland Road, Hopkinton, Massachusetts 01748 • Tel. (617) 435-3452

March 29, 1978

Mr. Stan Vivian
Dept. of Pharmacology and
Therapeutics
University of Manitoba
770 Bannatyne Avenue
Winnipeg, Manitoba
Canada R3E OW3

Dear Stan:

I just saw your note in the Minitasker regarding RT-11 system overhead that severely limits maximum analog to digital sampling rates. I first noticed the exact same thing about a year ago. The first indication I had was that the Fortran Extensions routines could not reach the very modest sampling rates promised in the manual. Further investigation yielded the exact same analysis you reported. My estimate of the overhead was consistent with yours. My solution was even more brutal than yours, however. I wrote a very simple fast sample routine that simply turned off all interrupts (i.e., PS=7) during the sampling interval then restored to previous status. With this strategy I can, quite naturally, fill a buffer at the theoretical limit of 35 to 40 KHz on our 11/34 with LPS.

For reasons I have not been able to explain so far (due to a lack of source files and a shortage of time), the Lab Applications package somehow manages to overcome this problem. The THRU program in that package can do sustained Thruput to an RKØ5 on my system at the full promised 18 KHz (about 55 microseconds per sample). I have not confirmed it, but I believe the ADC code in the SPARTA part of The Lab Apps can approach the same limit I mentioned above for filling single buffers without thruput. It would be interesting to investigate how these tricks are accomplished.

I have tried to bring this problem to the attention of several friends and contacts in LDP and the RT-11 development group, but so far the answers are, "I have no idea", "I never heard of that", "I am sure that can't be" or "Maybe in FB but no way for SJ". I suspect the FORTRAN code was tested against an earlier version of RT that had less overhead. A clue might be that at one time handlers were allowed to set whatever priority they wanted for interrupts but now (V2C, I think) that was changed so that the interrupt is always at PS=7 then system calls have to be used to reduce the

priority to the desired level. This is the class of monitor design change I would start looking at if I were looking for the source of this problem.

Very truly yours,

Robert Hassinger

Unirad Corp.
4765 Oakland St.
Denver, CO 80239

John T. Rasted
JTR Associates
58 Rasted Lane
Meridian, CT 06450

Dear John,

This is regarding your letter in the April Mini-Tasker from Ray Strackbein discussing problems with ODT. ODT will use console Ø regardless of any

```
SET TT: CONSOLE=1
```

This is because it directly accesses the keyboard/printer registers at address 177560.

I suspect that the problem with ODT losing characters is due to two circumstances. When ODT processes a breakpoint it normally saves the status of the keyboard/printer registers and then disables the interrupt bit. If this is not the case when the RUN command is used, then the interrupt enable would still be on, thus the stolen characters. Still, this can only happen if the run priority of ODT is set to Ø. Thus, keyboard interrupts could still be accepted by RT-11, effectively stealing them from ODT. ODT would only get characters if it happened to be testing the TTY ready bit just as it went high, but before the interrupt could be generated.

Sincerely,

Larry A. Tepper

Larry A. Tepper
Software Project Engineer

USER INPUT

The following is submitted by N.A. Bourgeois of Sandia Laboratories Albuquerque, New Mexico. It describes the interactive use of BATCH, which he mentioned at the last DECUS Symposium. He will have copies of the files on a floppy disk at the Fall Symposium in San Francisco.

INTERACTIVE USE OF BATCH

We have developed a technique to permit casual users to run complex jobs through the use of BATCH in an interactive fashion. This technique lets BATCH to branch and/or loop in response to user's keyboard entries. The process is initiated by executing the indirect command file, BATCH.COM, and then responding to the prompting asterisk with MENU. Try executing the sample files listed below and studying the resulting log file.

```
!BATCH.COM
LOAD    BA,LP,NL
ASSIGN  LP:     LOG
ASSIGN  NL:     LST
RUN     BATCH
!EOF

$JOB/RT11       !MENU.BAT
        TTYIO
$MES ASSIGNMENTS
$MES PROMPTING MESSAGE (JOB1)
.R PIP
*TEMP.BAT<'CTY'.BAT
$CALL TEMP.BAT
        TTYIO
```

```
.R PIP
*TEMP.BAT/D
*TEMP.CTL/D
$MES DEASSIGNMENTS
$EOJ

$JOB/RT11        !JOB1.BAT
        TTYIO
$MES JOB CONTROL LANGUAGE (JCL)
$MESSAGE PROMPTING MESSAGE (Y OR N)
$CALL ANSWER.BAT
        TTYIO
        IF(A-"Y)NAY,YEA,NAY
YEA:
$MES YEA JCL
        GOTO EXIT
NAY:
$MES NAY JCL
EXIT:
$MES EXIT JCL
.R PIP
*ANSWER.CTL/D
$EOJ


$JOB/RT11        !ANSWER.BAT
        TTYIO
.R PIP
*ANS.BAT<'CTY'.BAT
$CALL ANS.BAT
        TTYIO
.R PIP
*ANS.BAT/D
*ANS.CTL/D
$EOJ

$JOB/RT11        !N.BAT
        LET A="N
$EOJ

$JOB/RT11        !Y.BAT
        LET A="Y
$EOJ
```

N. A. Bourgeois,Jr. 1736
Sandia Laboratories
PO Box 5800
Albuquerque, NM 87185

PAST SYMPOSIUM INFORMATION

Dear John:

Enclosed are copies of the handouts we picked up at the last decus. Also included is a xerox of the cover of the DECNET manuals which we picked up. The order number for these manuals might be of interest to DECNET users.

I am sending a copy of my notes on the hardware hints and kinks sessions to Tom Provost to edit as he wishes.

Hope these can be of some value.

Sincerely,

Ron Trellue

Ron Trellue
Division 9323

RT-11

EXTENDED

MEMORY

SUPPORT

WHAT?

EXTENDED MEMORY SUPPORT IS SUPPORT OF

MORE THAN 28K WORDS OF MEMORY.

WHY?

- RT-11 USER PROGRAMS LIMITED TO 32K WORDS OF PHYSICAL ADDRESS SPACE
- SEVERAL PRODUCTS COULD BENEFIT FROM HAVING A LARGER ADDRESS SPACE
  - MU/BASIC
  - FORTRAN IV
  - TIME SHARED DIBOL

HOW? (GOALS)

- PROVIDE MONITOR FACILITIES TO
  - MANAGE KT-11 MAPPING REGISTERS
  - ALLOCATE AND CONTROL EXTENDED MEMORY
- REMAIN COMPATIBLE WITH RT-11 FB

NON-GOALS

- RUN OR FRUN INTO EXTENDED MEMORY
- MODIFICATION OF CUSPS FOR EXTENDED MEMORY
- SEGMENTATION OR PAGING SUPPORT
- PROTECTION SCHEMES USING KT-11
- 22 BIT I/O

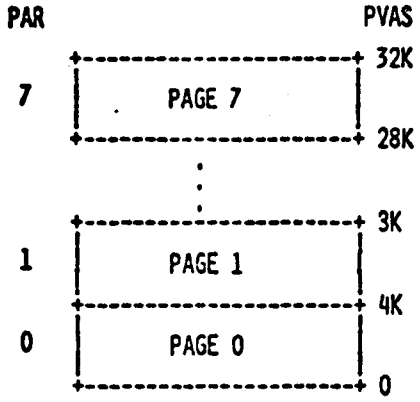
TERMS

- EXTENDED MEMORY - HARDWARE MEMORY BEYOND 28K
- PROGRAM VIRTUAL ADDRESS SPACE (PVAS) - THE SET OF ADDRESSES TO WHICH A PROGRAM CAN REFER.
- WINDOW - A DEFINED CONTINUOUS SEGMENT OF A PROGRAM'S VIRTUAL ADDRESS SPACE.
- REGION - A DEFINED CONTINUOUS SEGMENT OF PHYSICAL (I.E. HARDWARE) MEMORY
- PROGRAM PHYSICAL ADDRESS SPACE (PPAS) - THE SET OF REGIONS ALLOCATED TO A PROGRAM.
- MAPPING - ASSIGNMENT OF WINDOWS TO REGIONS.
- DYNAMIC - DEFINED AND ALLOCATED/DEALLOCATED BY REQUESTS TO THE MONITOR FROM A RUNNING PROGRAM
- STATIC - DEFINED AND ALLOCATED PRIOR TO EXECUTION; CANNOT BE CHANGED DURING EXECUTION

THE HARDWARE

- MEMORY MANAGEMENT UNIT (MMU)
  (PDP-11/34 FOR EXAMPLE)

DIVIDES PVAS INTO 8 4K SEGMENTS CALLED PAGES. ASSOCIATED WITH EACH SEGMENT IS A PAGE ADDRESS REGISTER (PAR) THAT CONTAINS THE RELOCATION CONSTANT FOR THAT PAGE.



PAGES MUST START ON 4K BOUNDARIES AND CAN BE FROM 32 TO 4K WORDS IN LENGTH. IN MULTIPLES OF 32 WORDS.

THERE ARE TWO OPERATING MODES OF THE MMU

- KERNEL
- USER

EACH HAS ITS OWN SET OF 8 PAR'S AND ITS OWN STACK POINTER.

DIFFERENCES:

INTERRUPTS AND TRAPS VECTOR THROUGH KERNEL MODE ADDRESS SPACE

IN USER MODE - RESET INSTRUCTIONS ARE TREATED AS NOP'S. HALT INSTRUCTIONS GENERATE A TRAP TO 10.

YOU CANNOT RTI, RTT FROM USER TO KERNEL MODE.

THE SOFTWARE

RT-11 ALLOWS PROGRAMS TO

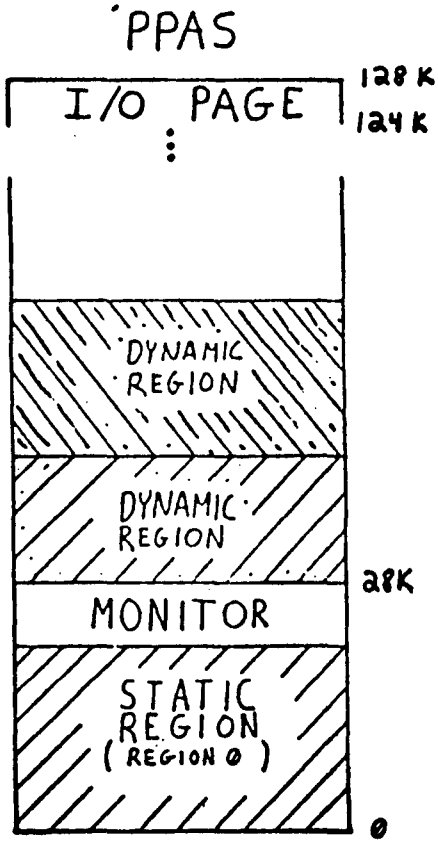
- CREATE AND ELIMINATE REGIONS (.CRRG, .ELRG)
- CREATE AND ELIMINATE WINDOWS (.CRAW, .ELAW)
- MAP AND UNMAP WINDOWS (.MAP, .UNMAP)
- GET MAPPING STATUS OF WINDOWS (.GMCX)
- GET FIXED MONITOR OFFSETS (.GVAL)

REGIONS

STATIC REGION (REGION 0) - REGION IN WHICH THE PROGRAM BASE SEGMENT RESIDES

DYNAMIC REGIONS - REGIONS CREATED BY THE .CRRG REQUEST

UP TO 4 PER PROGRAM
UP TO 96K WORDS IN LENGTH
ARE LOCATED ABOVE 28K (I.E. IN EXTENDED MEMORY)



REGIONS

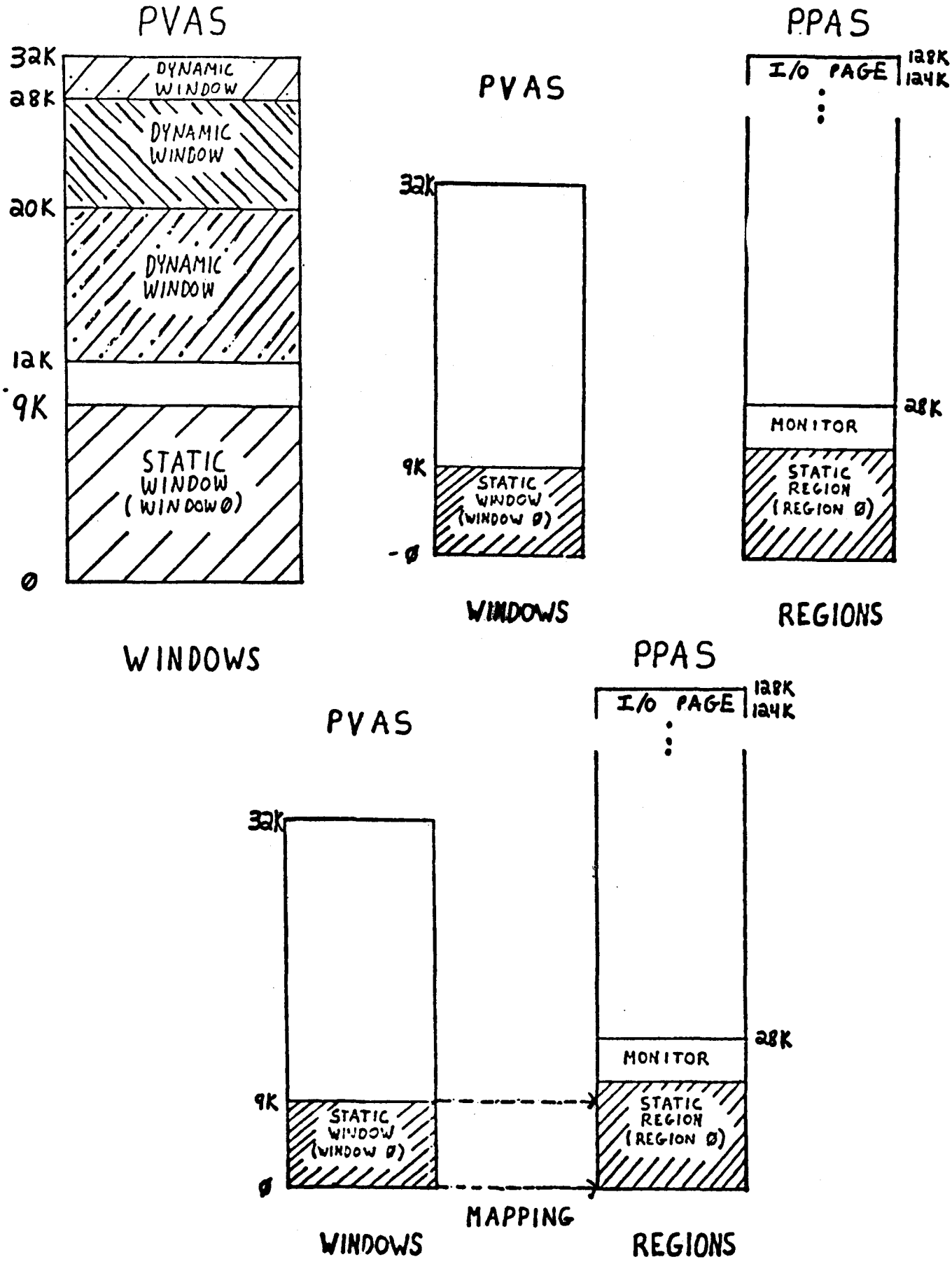
WINDOWS

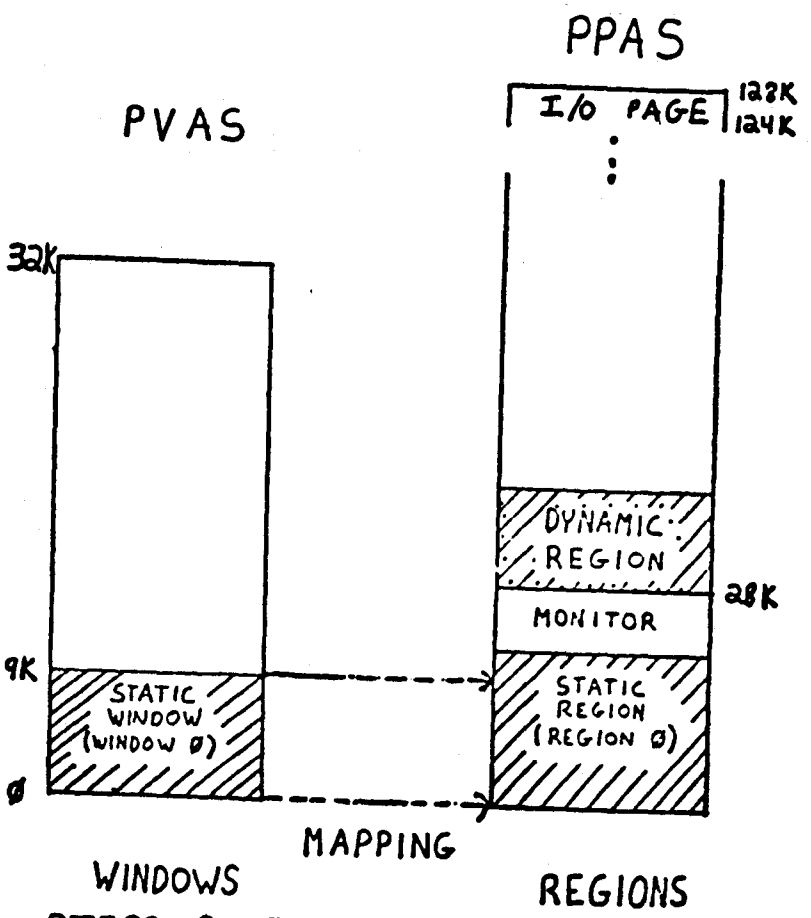
STATIC WINDOW (WINDOW 0) - EXTENDS FROM VIRTUAL 0 TO THE HIGH LIMIT OF THE PROGRAM BASE SEGMENT.

DYNAMIC WINDOWS - WINDOWS CREATED BY THE .CRAW REQUEST

UP TO 7 PER PROGRAM
FROM 32-28K WORDS IN LENGTH
START ON 4K BOUNDARY
CAN'T OVERLAP OTHER WINDOWS

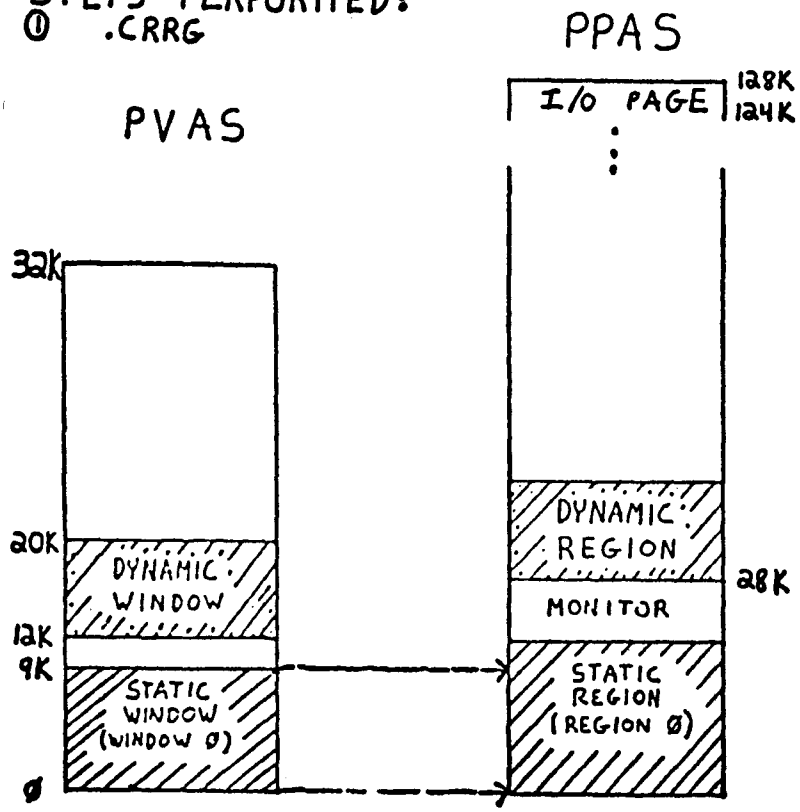
NOTE: SEVERAL WINDOWS CAN MAP TO OVERLAPPING PORTIONS OF A REGION.


PVAS
32K
28K
20K
12K
9K
0
DYNAMIC WINDOW
DYNAMIC WINDOW
DYNAMIC WINDOW
STATIC WINDOW (WINDOW 0)
WINDOWS
PVAS
32K
9K
0
STATIC WINDOW (WINDOW 0)
WINDOWS
PPAS
I/O PAGE
128K
124K
28K
MONITOR
STATIC REGION (REGION 0)
REGIONS
PPAS
PVAS
I/O PAGE
128K
124K
32K
28K
MONITOR
9K
0
STATIC WINDOW (WINDOW 0)
STATIC REGION (REGION 0)
MAPPING
WINDOWS
REGIONS



STEPS PERFORMED:

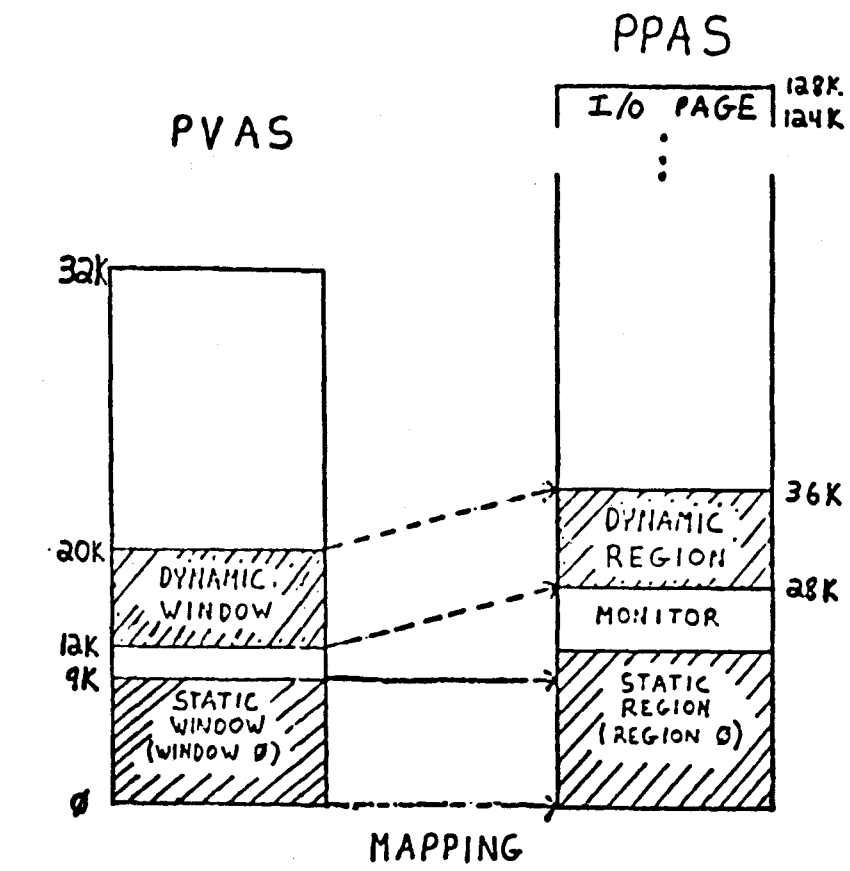
① .CRRG



STEPS PERFORMED:

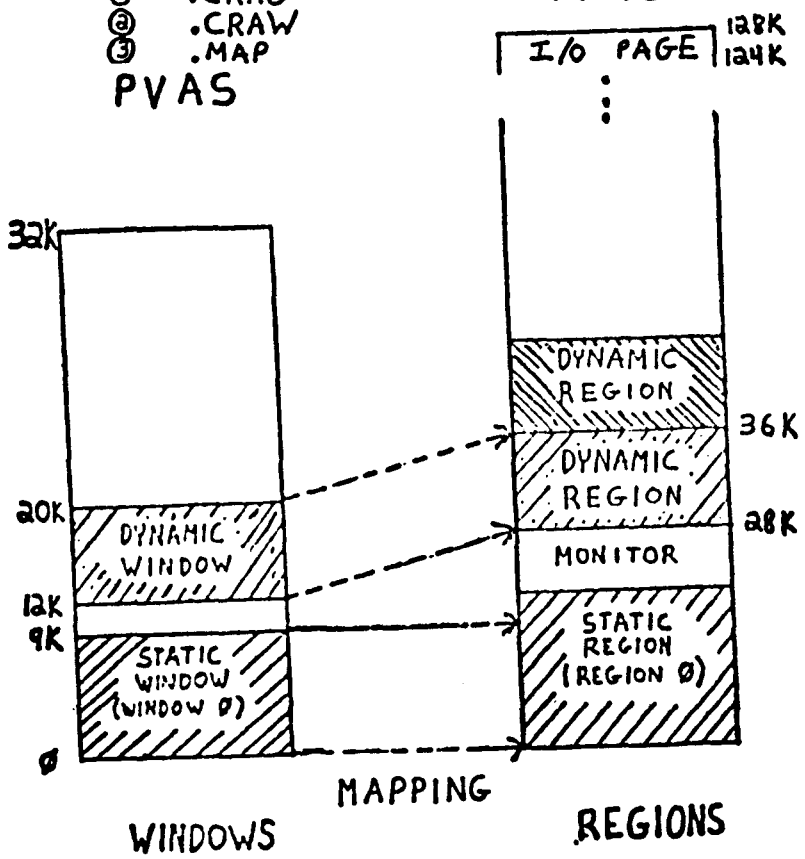
① .CRRG

② .CRAW



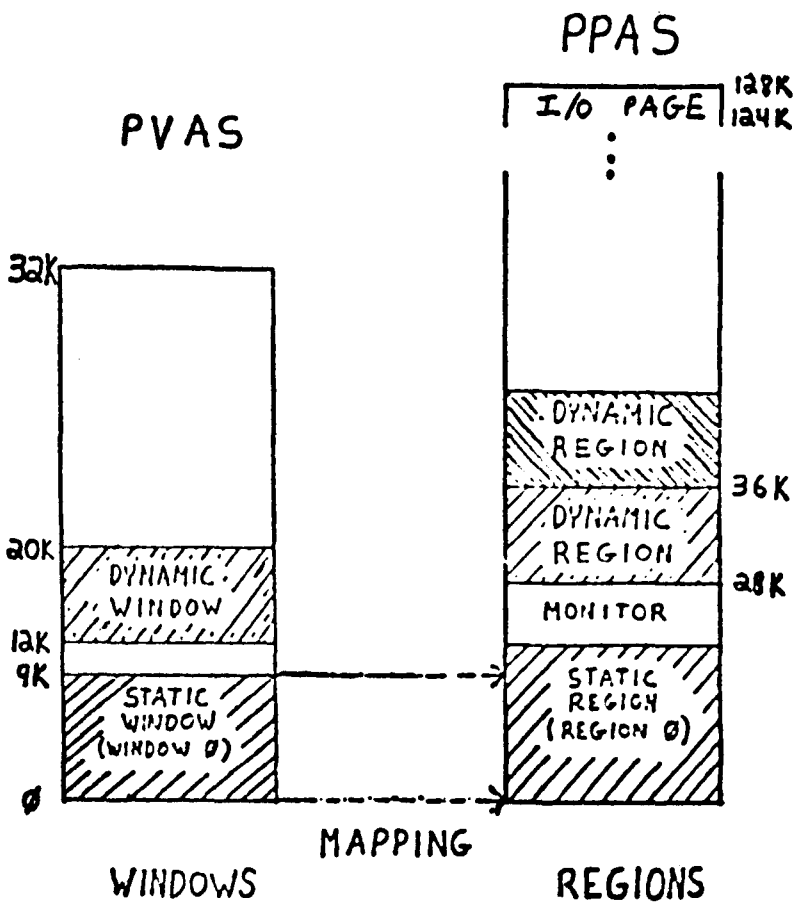
STEPS PERFORMED:

① .CRRG
② .CRAW
③ .MAP



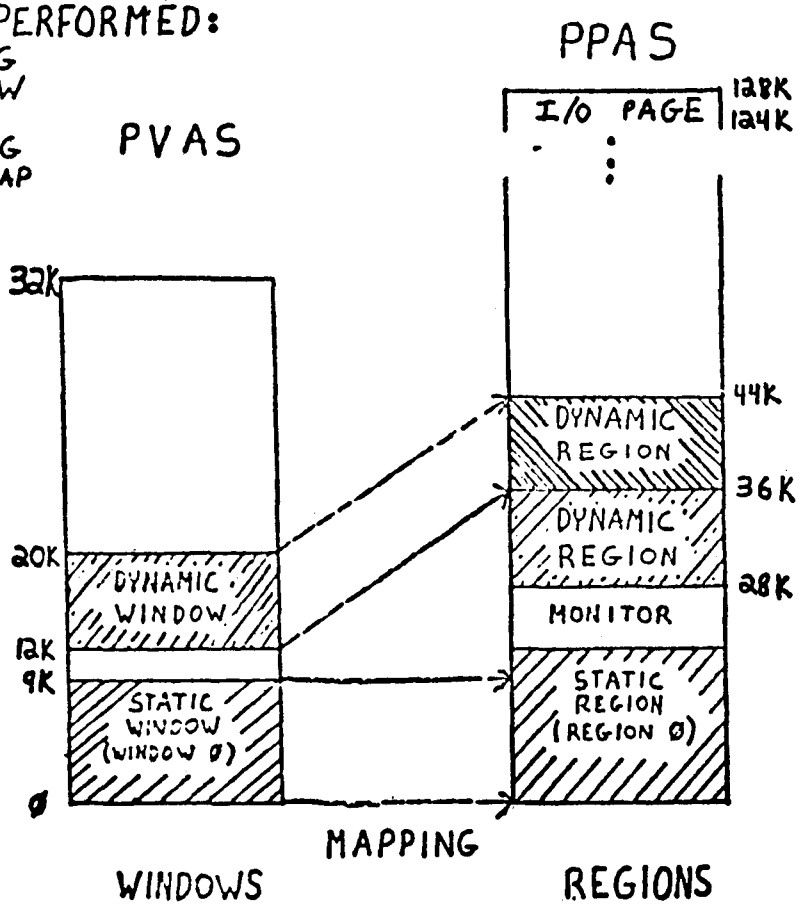
STEPS PERFORMED:

① .CRRG
② .CRAW
③ .MAP
④ .CRRG



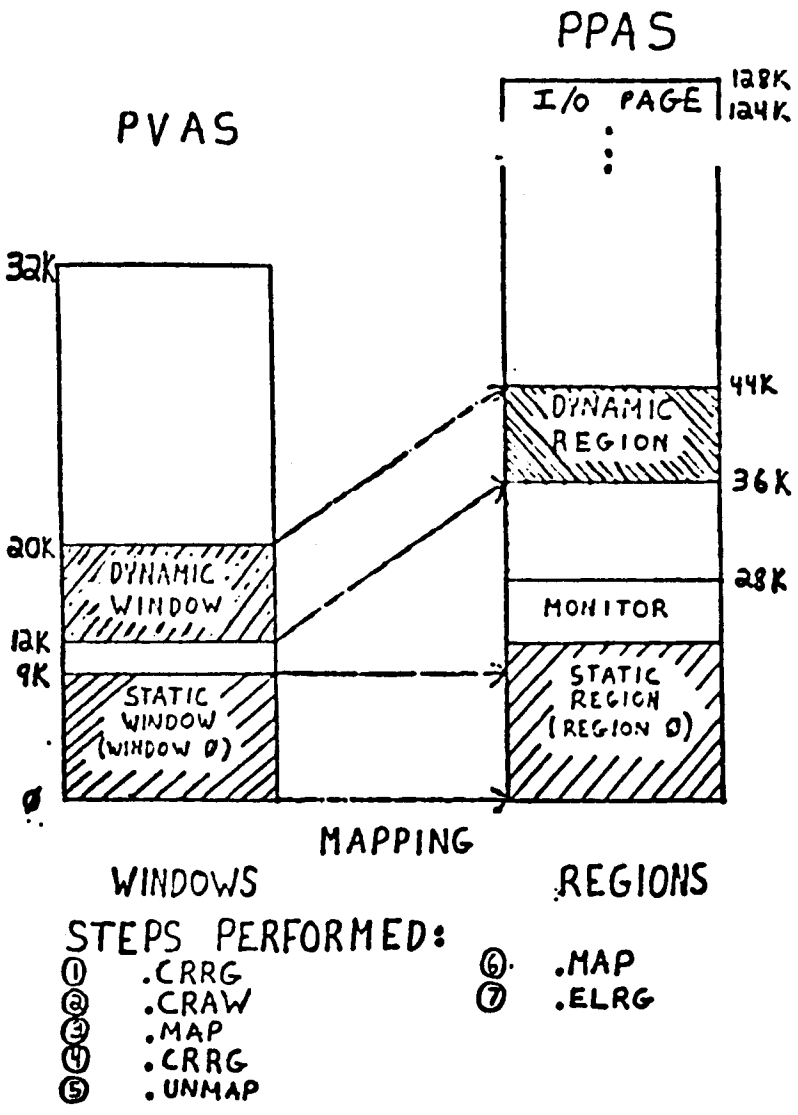
STEPS PERFORMED:

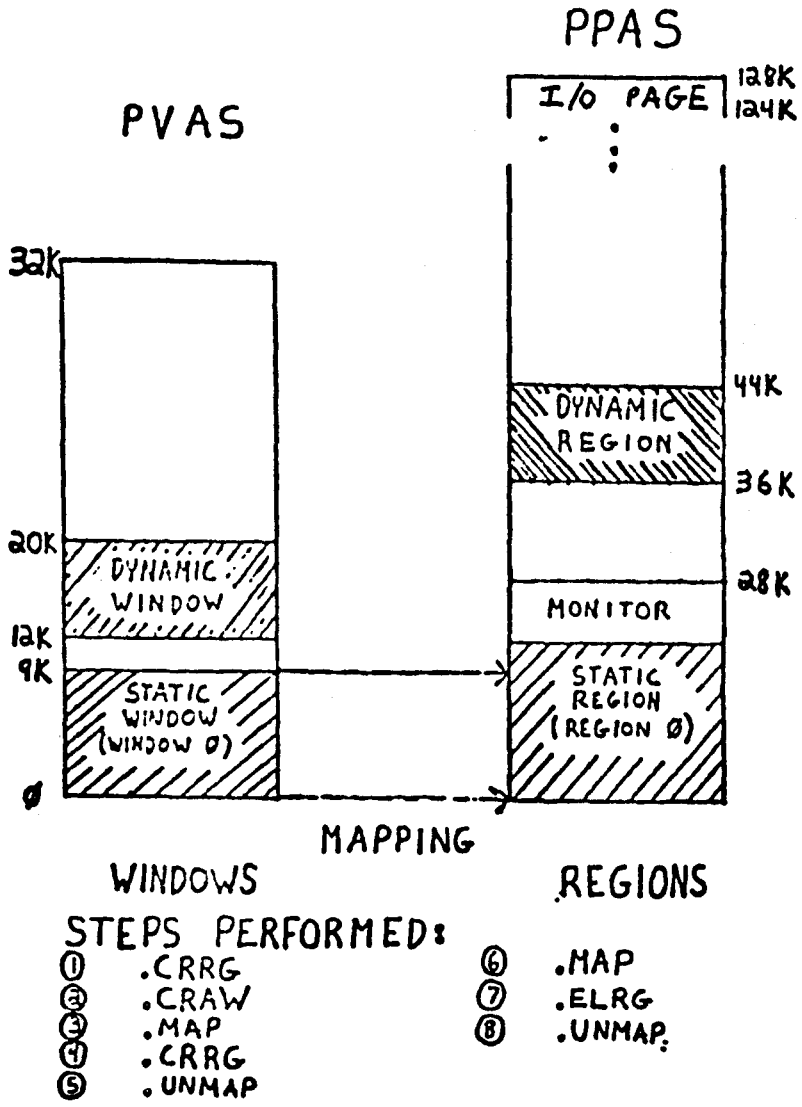
① .CRRG
② .CRAW
③ .MAP
④ .CRRG
⑤ .UNMAP

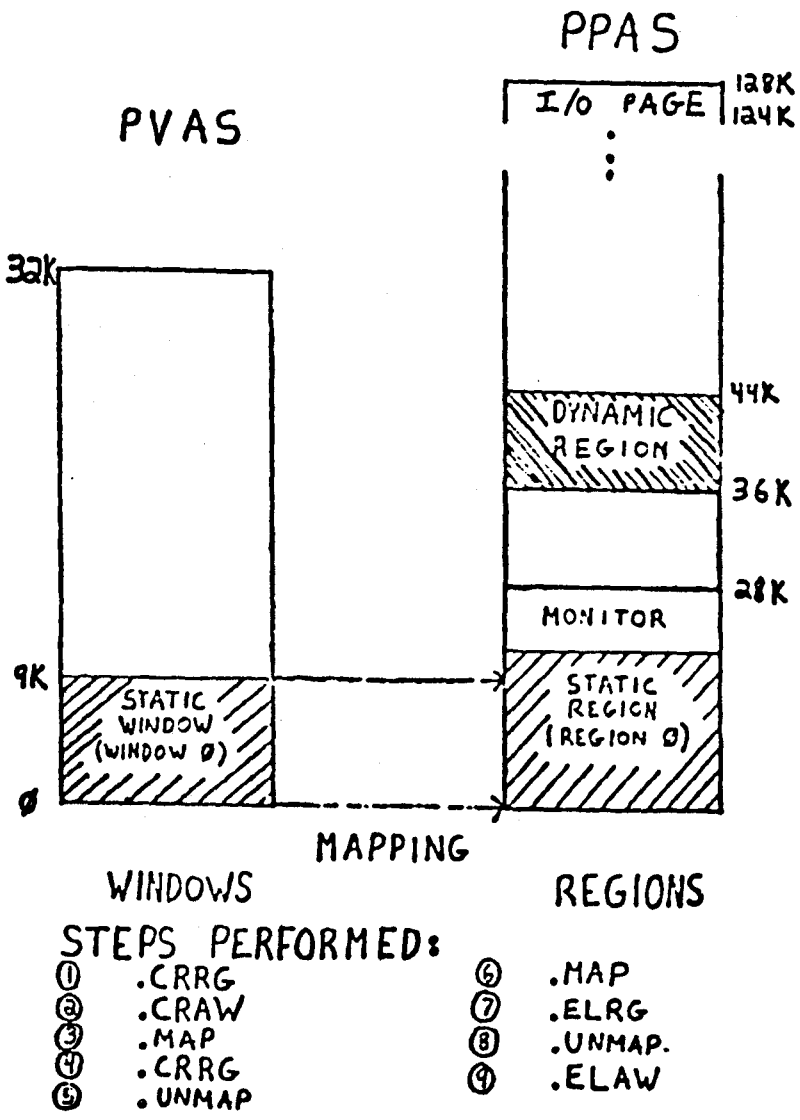


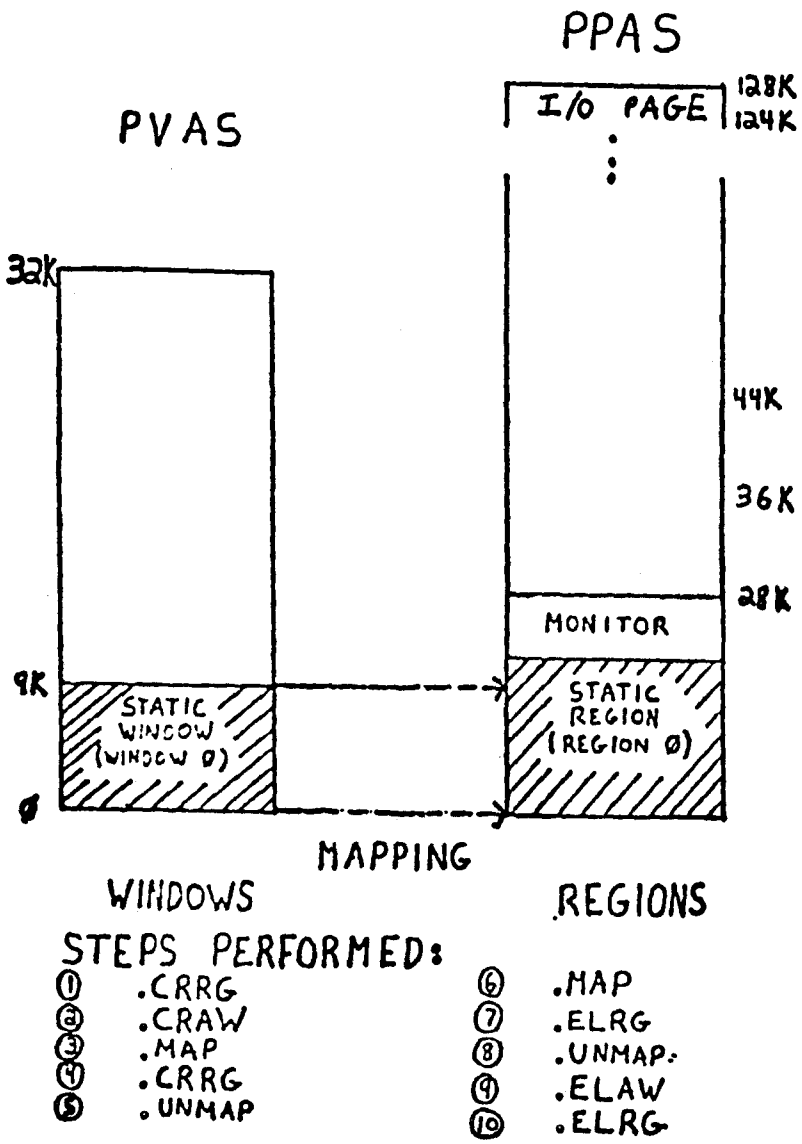
STEPS PERFORMED:

① .CRRG
② .CRAW
③ .MAP
④ .CRRG
⑤ .UNMAP
⑥ .MAP


PPAS
PVAS
I/O PAGE
128K
124K
32K
44K
DYNAMIC REGION
36K
20K
DYNAMIC WINDOW
28K
MONITOR
12K
9K
STATIC WINDOW (WINDOW 0)
STATIC REGION (REGION 0)
0
MAPPING
WINDOWS
REGIONS
STEPS PERFORMED:
① .CRRG
② .CRAW
③ .MAP
④ .CRRG
⑤ .UNMAP
⑥ .MAP
⑦ .ELRG


PPAS
PVAS
I/O PAGE
128K
124K
32K
44K
DYNAMIC REGION
36K
20K
DYNAMIC WINDOW
28K
MONITOR
12K
9K
STATIC WINDOW (WINDOW 0)
STATIC REGION (REGION 0)
0
MAPPING
WINDOWS
REGIONS
STEPS PERFORMED:
① .CRRG
② .CRAW
③ .MAP
④ .CRRG
⑤ .UNMAP
⑥ .MAP
⑦ .ELRG
⑧ .UNMAP.


PPAS
PVAS
I/O PAGE
128K
124K
32K
44K
DYNAMIC REGION
36K
28K
MONITOR
9K
STATIC WINDOW (WINDOW 0)
STATIC REGION (REGION 0)
0
MAPPING
WINDOWS
REGIONS
STEPS PERFORMED:
① .CRRG
② .CRAW
③ .MAP
④ .CRRG
⑤ .UNMAP
⑥ .MAP
⑦ .ELRG
⑧ .UNMAP.
⑨ .ELAW



MEMORY LAYOUT AND MAPPING

MONITOR IN HIGH END OF LOWER 28K

EXECUTES IN KERNEL MODE
MAPS I/O PAGE AND LOWER 28K
KERNAL VECTOR SPACE IS LOW 256 WORDS

USR ALWAYS RESIDENT

RUNS IN KERNEL MODE

KMON IS PRIVILEGED BACKGROUND JOB

RUNS IN USER MODE
MAPPED TO I/O PAGE AND LOW 28K LIKE RMON

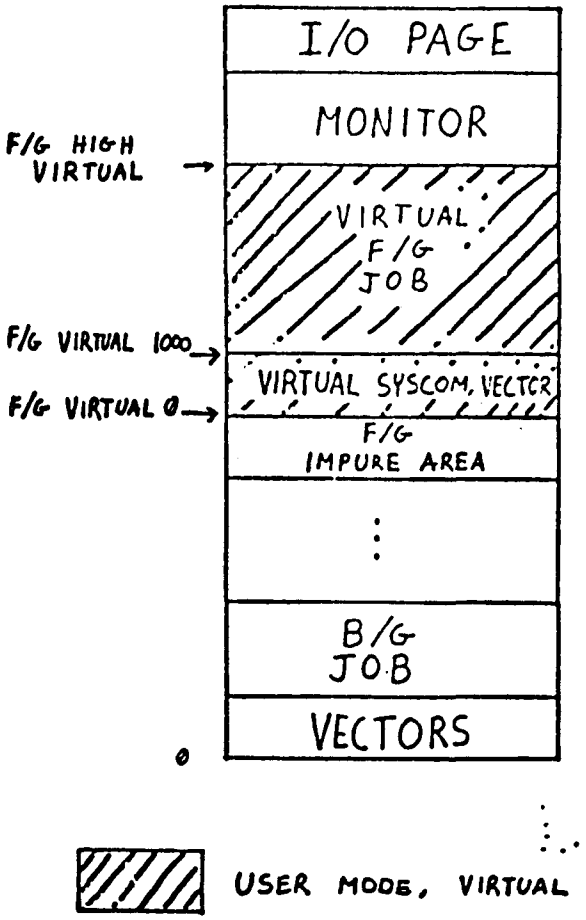
FOREGROUND AND BACKGROUND RUN IN USER MODE

TWO TYPES OF USER MODE MAPPING

PRIVILEGED (COMPATIBILITY) MAPPING
- DEFAULT MAPPING - V02C COMPATIBLE
- FULL ACCESS TO VECTORS, MONITOR, I/O PAGE
- JOB MAPPED TO I/O PAGE, LOWER 28K
- NO WINDOW 0 / REGION 0 CONCEPT

VIRTUAL MAPPING
- FULL 32K VIRTUAL ADDRESS SPACE AVAILABLE FOR PROGRAM AND DATA
- NO PRIVILEGED ACCESS TO MONITOR, I/O PAGE
- JOB MAPPED STARTING AT USER VIRTUAL 0
- LOW 500 BYTES - VIRTUAL VECTOR, SYSCOM AREA
- WINDOW 0 / REGION 0 CONCEPT -
  - REGION 0 STARTS AT LOCATION 500 (8)* AND EXTENDS TO SYSLOW
  - WINDOW 0 MAPS VIRTUAL 0 - TOP OF PROGRAM BASE
  - VIRTUAL SPACE ABOVE WINDOW 0 CAN BE MAPPED

* IN V3B. SLIGHTLY DIFFERENT IN V03.



ADDITIONAL EXTENDED MEMORY SUPPORT

CAN ISSUE EMT'S FROM EXTENDED MEMORY

I/O SUPPORTED WITHIN PVAS
- CAN UNMAP BUFFER ONCE EMT IS ISSUED
- COMPLETION ROUTINES MUST STAY MAPPED
- BUFFERS CAN'T CROSS REGION BOUNDARY

FOREGROUND AND BACKGROUND CAN ACCESS EXTENDED MEMORY SIMULTANEOUSLY

RESTRICTIONS

MEMORY LIMITED TO 128K WORDS

NO PROGRAM LOADING INTO EXTENDED MEMORY

RESTRICTIONS ON PROGRAMMED REQUESTS:

.CDFN - CHANNEL AREA IN LOW 28K

.QSET - QUEUE SPACE IN LOW 28K

.SETTOP - ONLY UP TO CURRENT HI VIRTUAL

.CNTXSW --+
.DEVICE --+----NOT AVAILABLE TO VIRTUAL JOBS

.SFPA ----+
.TRPSET --+----TRAP ADDRESS MUST STAY MAPPED

COMPLETION ROUTINES MUST STAY MAPPED

RESTRICTIONS (CONTINUED)

FOUR DYNAMIC MEMORY REGIONS PER PROGRAM
EACH REGION ≤ 96K, MULTIPLE OF 32 WORDS

SEVEN DYNAMIC WINDOWS PER PROGRAM
EACH WINDOW ≤ 28K, MULTIPLE OF 32 WORDS
ALIGNED ON 4K VIRTUAL ADDRESS BOUNDARY

INTERRUPT SERVICE ROUTINES IN PRIVILEGED JOBS ONLY
MUST RESIDE IN LOW 28K

USR ALWAYS RESIDENT, NOSWAP

HANDLERS MUST RESIDE IN LOW 28K
ALL HANDLERS MUST BE LOADED

HOW TO:

- CREATE REGIONS
- CREATE WINDOWS
- MAP WINDOWS TO REGIONS

REGIONS ARE CREATED BY THE .CRRG PROGRAMMED REQUEST:

```
.CRRG     AREA[,RGADR]
```

WHERE AREA IS THE ADDRESS OF A TWO WORD EMT ARGUMENT BLOCK:

RGADR IS THE ADDRESS OF THE REGION DEFINITION BLOCK

INPUT: SIZE OF REGION (IN 32 WORD BLOCKS)
OUTPUT: REGION ID OR, IF NOT SUCCESSFUL, SIZE OF LARGEST REGION AVAILABLE

REGION DEFINITION BLOCK

A 3 WORD BLOCK WHICH CAN BE CREATED AT ASSEMBLY TIME BY INVOKING THE MACRO:

```
.RDBBK    RGSIZ
```

WHERE RGSIZ IS THE SIZE OF THE REGION IN 32 WORD BLOCKS

REGION DEFINITION BLOCK

	OFFSET
REGION ID	R.GID
SIZE OF REGION	R.GSIZ
STATUS	R.GSTS

MNEMONIC	EXPLANATION
R.GID	ID NUMBER FOR REGION; SUPPLIED BY MONITOR
R.GSIZ	SIZE OF REGION IN 32 WORD BLOCKS
R.GSTS	REGION STATUS WORD; SUPPLIED BY MONITOR

REGION STATUS WORD BITS

MNEMONIC	EXPLANATION
RS.CRR	=1 IF REGION WAS SUCCESSFULLY CREATED
RS.UNM	=1 IF ONE OR MORE WINDOWS WERE UNMAPPED AS A RESULT OF ELIMINATING THIS REGION
RS.NAL	=1 IF REGION SPECIFIED IS NOT ALLOCATED AT THIS TIME

TO DEFINE THE OFFSETS AND STATUS BITS

WITHOUT CREATING THE ACTUAL BLOCK.

INVOKE THE MACRO

.RDBDF

EXAMPLE 1

CREATE A 3K WORD REGION IN EXTENDED MEMORY

```
RGADR     .RDBBK    3*1024./32.
AREA:     .BLKW     2
               ;
               ;
               ;
FOO:      .CRRG     #AREA,#RGADR
          BCC       AOK             ;NO ERRORS
          CMPB      @#52.#5         ;ERROR 6?
          BEQ       NORCB           ; YES
          CMPB      @#52.#7         ;ERROR 7?
          BEQ       NOSPACE         ; YES
     (ERROR 10 - ILLEGAL REGION SIZE)
NORCB:
     (ERROR 6 - NO FREE REGION CONTROL BLOCKS)
NOSPACE:
     (ERROR 7 - REGION OF REQUESTED SIZE NOT
                AVAILABLE)
AOK:
```

WINDOWS ARE CREATED BY THE .CRAW PROGRAMMED REQUEST:

.CRAW AREA [,WADR]

WHERE AREA IS THE ADDRESS OF A TWO WORD EMT ARGUMENT BLOCK:

36	2
WADR	

WADR IS THE ADDRESS OF THE WINDOW DEFINITION BLOCK

INPUT: BASE PAR
SIZE OF WINDOW (IN 32 WORD BLOCKS)
OUTPUT: BASE VIRTUAL ADDRESS

WINDOW DEFINITION BLOCK

A 7 WORD BLOCK WHICH CAN BE CREATED AT ASSEMBLY TIME BY INVOKING THE MACRO:

.WDBBK WWAPR, WWSIZ [,WWRID,WWOFF,WWLEN,WWSTS]

WINDOW DEFINITION BLOCK

BASE PAR	WINDOW ID
BASE VIRTUAL ADDRESS	
WINDOW SIZE	
REGION ID	
OFFSET INTO REGION	
LENGTH TO MAP	
STATUS	

OFFSET	
W.NAPR	W.NID
W.NBAS	
W.NSIZ	
W.NRID	
W.NOFF	
W.NLEN	
W.NSTS	

MNEMONIC	EXPLANATION
W.NID	WINDOW IDENTIFIER (MONITOR)
W.NAPR	STARTING PAR OF WINDOW
W.NBAS	BASE VIRTUAL ADDRESS OF WINDOW (MONITOR)
W.NSIZ	SIZE OF WINDOW IN 32 WORD BLOCKS

WINDOW DEFINITION BLOCK, CONTINUED

MNEMONIC	EXPLANATION
W.NRID	ID OF THE REGION TO BE MAPPED TO; SAME AS R.GID RETURNED BY .CRRG
W.NOFF	OFFSET INTO REGION AT WHICH TO START MAPPING THE WINDOW
W.NLEN	LENGTH OF THE WINDOW TO MAP IN 32 WORD BLOCKS
W.NSTS	WINDOW STATUS BITS

WINDOW STATUS WORD BITS

MNEMONIC	EXPLANATION
WS.MAP	MAP THE WINDOW TO THE SPECIFIED REGION AFTER CREATING IT, SAVING AN EXTRA .MAP REQUEST
WS.CRW	ADDRESS WINDOW WAS SUCCESSFULLY CREATED
WS.UNM	ONE OR MORE WINDOWS WERE UNMAPPED TO CREATE AND MAP THIS WINDOW
WS.ELW	ONE OR MORE WINDOWS WERE ELIMINATED TO CREATE THIS WINDOW

TO DEFINE THE OFFSETS AND STATUS BITS

WITHOUT CREATING THE ACTUAL BLOCK,

INVOKE THE MACRO:

```
WDBDF
```

EXAMPLE 2

CREATE A 3K WORD WINDOW STARTING AT THE PAGE
MAPPED BY PAR 7 AND MAP IT STARTING AT OFFSET 0
INTO THE REGION CREATED IN THE PREVIOUS EXAMPLE

```
RGADR:    .RDBBK    3*1024./32.
AREA:     .BLKW     2
WADR:     .WDBBK    7,3*1024./32.,0,0,0,WS.MAP
               ;
               ;
               ;
FOOBAR:   MOV  #RGADR,R1       ;R1 -  REG DEF BLK
          MOV  #WADR,R2        ;R2 -  WIN DEF BLK
          MOV  R.GID(R1),W.NRID(R2)
                               ;PUT REG ID INTO
                               ;WINDOW BLOCK
          .CRAW     #AREA,#WADR
                               ;CREATE AND MAP
                               ;WINDOW
          BCC  AOK
     (HANDLE ERROR CONDITIONS).
AOK:
```

WE HAVE A WINDOW MAPPED TO A REGION

ALLOCATED IN EXTENDED MEMORY.

NOW WHAT ????

EXAMPLE 3

READ FOUR BLOCKS (4000 OCTAL WORDS) FROM
CHANNEL 5 INTO THE REGION. (ASSUME .LOOKUP
ON CHANNEL 5 ALREADY PERFORMED)

```
RGADR:    .RDBBK    3*1024./32.
WADR:     .WDBBK    7,3*1024./32.,0,0,0,WS.MAP
BUFF = 7*4096.*2         ;BUFFER ADDRESS -
                         ;BUFF = 160000 (8)
RAREA:    .BLKW     5    ;EMT ARGUMENT BLOCK
               ;
               ;
               ;
FOOTOO:   .READW    #RAREA,#5,#BUFF,#4000,BLKNO
          BCC       OKREAD
     (HANDLE ERROR CONDITIONS)

OKREAD:
```

EXAMPLE 4

WRITE TWO 5000 (8) WORD BUFFERS TO A DISK FILE.
THE BUFFERS ARE IN DIFFERENT XM REGIONS.
THE DISK FILE IS OPEN ON CHANNEL 7.

```
RGADR1:   .RDBBK    6*1024./32.     ;6K REGION
RGADR2:   .RDBBK    10*1024./32.    ;10K REGION
WADR:     .WDBBK    6.4*1024./32.,0,0,5000/32.
;
;    WINDOW STARTS AT PAR 5, 140000 (8). IS 4K
;    WORDS IN LENGTH, BUT ONLY 5000 (8) WORDS
;    WILL BE MAPPED
;
AREA:     .BLKW     5          ;EMT ARGUMENT BLOCK
               ;
               ;
               ;
FOO:      .CRRG     #AREA,#RGADR1  ;CREATE REG 1
          BCC       1$
     (HANDLE ERROR CONDITIONS)
1$:       .CRRG     #AREA,RGADR2   ;CREATE REG 2
          BCC       2$
     (HANDLE ERROR CONDITIONS)
2$:  (CREATE WINDOWS, MAP TO BUFFERS, FILL THEM)
```

```
;
;    AT THIS POINT, BOTH BUFFERS ARE FULL.
;    LET'S ASSUME WE MUST RECREATE THE WINDOW
;
     MOV  RGADR1+R.GID.WADR+W.NRID ;PUT REG ID
                                   ;IN WIN BLK
     MOV  #WS.MAP,WADR+W.NSTS ;SET MAP BIT
     .CRAW     #AREA,#WADR    ;CREATE AND MAP
     BCC  3$                  ;WINDOW
  (HANDLE ERROR CONDITIONS)
3$:  JSR  PC,PUTBUF           ;PUT OUT BUFFER
;
;    NOW MAP TO 2ND BUFFER, THEN WRITE IT OUT
;
     MOV  RGADR2+R.GID,WADR+W.NRID ;PUT 2ND REG
                                   ;ID IN WIN BLK
     .MAP      #AREA,#WADR    ;MAP TO SECOND BUF
     BCC  4$
  (HANDLE ERROR CONDITIONS)
4$:  JSR  PC,PUTBUF           ;WRITE OUT 2ND BUF
          ,
          ;
          ,
PUTBUF:   .WRITW    #AREA,#7,#6*4095.*2,#5000.BLKNO
     BCC  10$
  (HANDLE ERROR CONDITIONS)
10$: RTS  PC
```

SUMMARY

EXTENDED MEMORY SUPPORT IS IMPLEMENTED THROUGH

HARDWARE	SOFTWARE
MEMORY MANAGEMENT UNIT	XM MONITOR AND HANDLERS
EIS	USER DATA STRUCTURES
	REGION DEF BLOCK
	WINDOW DEF BLOCK
	PROGRAMMED REQUESTS:
	.CRRG
	.ELRG
	.CRAW
	.ELAW
	.MAP
	.UNMAP
	.GMCX
	.GVAL

HOW TO USE IT:

CREATE REGION DEFINITION BLOCK
    INVOKE .RD3BK TO CREATE BLOCK & DEFINITIONS, OR
    INVOKE .RD3DF FOR DEFINITIONS ONLY, THEN
        CREATE BLOCK YOURSELF
CREATE REGIONS
    ISSUE .CRRG TO CREATE REGIONS
    ISSUE .ELRG TO ELIMINATE REGIONS
CREATE WINDOW DEFINITION BLOCK
    INVOKE .WD3BK TO CREATE BLOCK & DEFINITIONS, OR
    INVOKE .WD3DF FOR DEFINITIONS ONLY, THEN
        CREATE BLOCK YOURSELF
CREATE WINDOWS
    ISSUE .CRAW TO CREATE AND OPTIONALLY MAP
        WINDOWS
    ISSUE .ELAW TO ELIMINATE WINDOWS
MAP WINDOWS TO REGIONS
    ISSUE .MAP (OR .CRAW) TO MAP WINDOWS
    ISSUE .UNMAP TO UNMAP WINDOWS
    A WINDOW IS IMPLICITLY UNMAPPED BY
        ANOTHER .MAP REQUEST

OPERATIONAL CHARACTERISTICS

*THE MONITOR (RMON, USR) RUNS IN KERNEL MODE
*KMON AND USER PROGRAMS RUN IN USER MODE
*THERE ARE TWO KINDS OF USER MODE MAPPING
    - PRIVILEGED (COMPATIBILITY) MAPPING
        THE DEFAULT MAPPING
        SIMULATES A NON-EXTENDED MEMORY
            ENVIRONMENT
    - VIRTUAL MAPPING
        SELECTED BY SETTING BIT IN THE JSW
        NO ACCESS TO MONITOR OR THE I/O PAGE
        FULL 32K ADDRESS SPACE AVAILABLE
*CHANNEL & QUEUE AREAS, INTERRUPT SERVICE ROUTINES
    MUST RESIDE IN LOW 28K
*INTERRUPT SERVICE ROUTINES IN PRIVILEGED JOBS ONLY
*I/O BUFFERS NEED NOT REMAIN MAPPED DURING I/O

SPRS

SYSTEM PROGRAM AND VERSION (OR DOCUMENT)	MONITOR AND VERSION	DATE
RT-11 BASIC V2	RT-11 V3	9 June 1978

NAME: Computing Centre
FIRM: Queen's University

ADDRESS: Dupuis Hall
Kingston, Ontario.
ZIP K7L 3N6

SUBMITTED BY: Andrew Hooper PHONE: (613)547-6254

LIST ATTACHMENTS
Program listing and output

DEC OFFICE
Ottawa

REPORT TYPE
- [X] LOGIC/CODING ERROR
- [ ] DOCUMENTATION ERROR
- [ ] SUGGESTION
- [ ] INQUIRY
- [ ] FOR YOUR INFORMATION

PRIORITY
- [ ] LOW
- [X] STANDARD
- [ ] HIGH

CAN THE PROBLEM BE REPRODUCED AT WILL?
- [X] YES
- [ ] NO

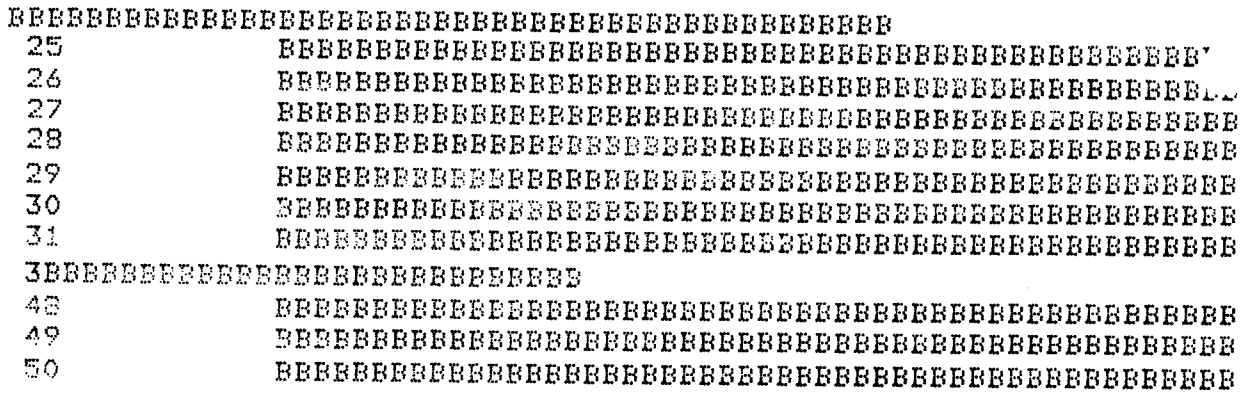
CPU TYPE	SERIAL NO.	SYSTEM DEVICE	MEMORY SIZE	DISTRIBUTION MEDIUM
PDP-11V03	WM00532	RX01	56KB	diskette

PROBLEM DESCRIPTION:

simultaneous use of a virtual array and another file causes loss of data from the other file.

Program output to console without a separate file gives full output, but when the program is changed to write to printer some blocks of output are lost.

```
10 B$='BBBBBBBBBBB'
20 B$=B$+B$+B$+B$+B$
30 DIM #1,L$(50)=127
40 OPEN 'TEST' FOR OUTPUT AS FILE #1
50 FOR L%=1 TO 50
60 L$(L%)=SEG$(B$+B$+B$,1,127)
70 NEXT L%
80 CLOSE #1
85 OPEN 'LP:' FOR OUTPUT AS FILE #2
90 OPEN 'TEST' FOR INPUT AS FILE #1
100 FOR L%=1 TO 50
110 PRINT #2,L%,SEG$(L$(L%),1,50)
120 NEXT L%
130 END
```



SYSTEM PROGRAM AND VERSION (OR DOCUMENT)	MONITOR AND VERSION	DATE
Various	RT-11 V03-02	4-MAY-78

DEC OFFICE

NAME: John Yardley
FIRM: National Physical Laboratory
ADDRESS: Queens Road,
Teddington, Middlesex,
England.
ZIP

SUBMITTED BY: John Yardley
PHONE: 01-977 3222

LIST ATTACHMENTS

REPORT TYPE
- ☐ LOGIC/CODING ERROR
- ☒ DOCUMENTATION ERROR
- ☒ SUGGESTION
- ☒ INQUIRY
- ☒ FOR YOUR INFORMATION

PRIORITY
- ☐ LOW
- ☒ STANDARD
- ☐ HIGH

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

CPU TYPE	SERIAL NO.	SYSTEM DEVICE	MEMORY SIZE	DISTRIBUTION MEDIUM
LSI-11	WM45264	RX01	20K	Floppies

As an existing OS/8 user and new RT-11 V03 user, some of my early observations and suggestions may be of interest to RT-11 users and maintainers generally.

Suggestion 1. It is unfortunate that the TT handler cannot be configured to operate in SCOPE mode under the SJ monitor. I realise that this would demand a larger resident monitor but with version 3, there is no reason why this support could not be optionally SYSGENed at the user's discretion.

Suggestion 2. Under OS/8, if you terminate a command line with a line feed rather than a carriage return, this causes the line, as reflected in the command buffer, to be re-typed on the terminal. The cursor/head is left at the end of the re-typed line to await further input or termination. In the absence of SCOPE mode for VDUs and especially for DECWRITER users, this would be extremely useful.

Inquiry 3. Under OS/8, it is fairly easy to add your own keyboard (CCL) commands since execution takes place via a "table-driven" overlay, of which a fully documented source is supplied to the user. Without diving into the un-commented monitor sources, there does not appear to be any mechanism in RT-11 to do the same. Will this information be available in the, as yet un-published, "Introduction to Advanced Programming"? Indeed, it would be most useful to know the proposed contents of this manual.

Suggestion 4. RT-11 TECO has many more features than OS/8 TECO and with the Mode Control Flags one can implement one's own editor much more easily. We have several macros which operate as different types of editor (or word processor) depending on the task at hand. It is a pity that there is not a Mode Control Flag that enables a pre-defined macro to be automatically executed if TECO detects an error condition. This would avoid the need to install time-consuming error traps in the macro.

CPU TYPE	SERIAL NO.	SYSTEM DEVICE	MEMORY SIZE	DISTRIBUTION MEDIUM
LSI-11	WMG5264	RX01	20K	Floppies

For your information 5. We spent several days generating a correct copy of the VT52 macro given in the TECO Release Notes (ORTNA-A-D), only to find that it it did not work on the delivered monitor. In order for this macro to work you must SYSGEN a monitor with escape sequence processing (difficult, if not impossible with a floppy system).

Inquiry 6. In the VEG.TEC macro (TECO Release Notes), bit 9 in the ET flag is used ie "512#2#ETET". What does it do? Furthermore, what is -1^W officially supposed to do on a VT52?

Documentation Errors 7. I don't think that the address of CONFIG in the SJ and FB monitors as given in the System Release Notes (ORNRB-A-D) is correct. It appears that the only way to get a 50 Hz clock is to do a SYSGEN. (again, hard work on a floppy system). One can, of course, obtain a map when assembling the monitor to find out where CONFIG lives, but if you do that you may just as well complete the SYSGEN process.

8. In the Advanced Programming Manual (ORPAPA-A-D), page 2-125, all the references to the JSW are inverted, ie 0 should read 1, set should read clear, etc.

9. In the System User's Guide (ORGDA-A-D) and in the Pocket Guide there is no mention of the /BEGIN option for DIRECT in the command summaries.

Suggestion 10. It would be useful to be able to incorporate Multi-Terminal support in the SJ monitor. Once more, this could be installed optionally with SYSGEN.

Suggestion 11. If you keep your .SYS files squeezed together at the bottom of the system, you don't want to re-boot the system every time you consolidate free space with a SQUEEZE. How about a NOBOOT option at the command level since it is available in PIP?

CPU TYPE	SERIAL NO.	SYSTEM DEVICE	MEMORY SIZE	DISTRIBUTION MEDIUM
LSI-11	WM45264	RX01	20K	Floppies

Documentation error 12. The System User's Guide states that the default set option for LP is CR. This is not the case, it is NOCR.

Suggestion 13. It appears that the keyboard monitor looks up the switch options before calling the utility program, so that options must be unique across all the utilities. If the utility did its own look-up, then you would need fewer letters to make the option unique (generally speaking).

I must apologise for cramming all these comments on only three SPRs but:

a) I only had three SPR forms
b) Since there are no logic/coding errors, all the points could probably be dealt with by one person.

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
RT-11	3	see below		Apr-11-78

(SEE EXAMPLE IN INSTRUCTIONS)

NAME: Fred Magee 9423
FIRM: Sandia Laboratories
ADDRESS: P. O. Box 5800
Albuquerque, NM
ZIP: 87185

SUBMITTED BY: Nick Bourgeois
PHONE: 264-8088

DEC OFFICE:

DO YOU HAVE SOURCES? YES ☐ NO ☐

REPORT TYPE
- ☐ SOFTWARE ERROR
- ☒ DOCUMENTATION ERROR
- ☐ INQUIRY
- ☐ FOR YOUR INFORMATION/SUGGESTION

PRIORITY
- ☐ LOW
- ☐ STANDARD
- ☒ HIGH

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
11/34	807	64K	~~RXD2~~ RK05	RK05	☐

REFERENCE:

RT-11	Software Dispatch	March 1978
EDIT	V03.38	SEQ 1 M

PROBLEM:

Installation of the referenced patch results in a checksum error. See the attached listings of the BATCH and LOG files.

```
$JOB/TIME/LIST/RT11                !EDIT.001
$MES NAB/SLA/1736                  07-APR-78/
$MES RT-11 SOFTWARE DISPATCH       MARCH 1978
$MES EDIT V03.38                   SEQ 1 M
        TTYIO
.COPY   DST:EDIT.SAV    PAT:EDIT.SAV
.RUN    PATCH
*PAT:EDIT/C
*3274/5037
*3276/16262
*3300/13700
*3302/54
*3304/32760
*3306/10000
*3310/300
*3312/1403
*3314/52737
*3316/170000
*3320/16262
*3322/12737
*3324/12737
*3326/16260
*3330/137
*3332/16260
*16252/12737
*16254/2362
*16256/34
*16260/137
*16262/3274
*16264/36
*16266/240
*16222\\101
*E43324
$MES CHECK THE LOG FILE FOR ERRORS
$EOJ

$JOB/TIME/LIST/RT11                !EDIT.001

10:18:10


$MES NAB/SLA/1736                  07-APR-78/

10:18:10


$MES RT-11 SOFTWARE DISPATCH       MARCH 1978

10:18:10


$MES EDIT V03.38                   SEQ 1 M

10:18:10
```

```
TIME
10:18:19


$EOJ

TIME
10:18:19



        TTYIO

COPY    DST:EDIT.SAV    PAT:EDIT.SAV

RUN     PATCH

FILE NAME--
PAT:EDIT/C
*3274/  0       5037
*3276/  0       16262
*3300/  0       13700
*3302/  0       54
*3304/  0       32760
*3306/  0       10000
*3310/  0       300
*3312/  0       1403
*3314/  0       52737
*3316/  0       170000
*3320/  0       16262
*3322/  0       12737
*3324/  0       12737
*3326/  0       16260
*3330/  0       137
*3332/  0       16260
*16252/ 16706   12737
*16254/ 164334  2362
*16256/ 12737   34
*16260/ 2362    137
*16262/ 34      3274
*16264/ 5037    36
*16266/ 36      240
*16222\ 40      101
*E

CHECKSUM?   43324

?PATCH-I-CHECKSUM ERROR

CHECKSUM?
```

OPERATING SYSTEM	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE
RT-11	3	RT-11 Software Dispatch February 1978	PAT004, pgs.97-102	20-Mar-78

(SEE EXAMPLE IN INSTRUCTIONS)

NAME: Fred Magee

FIRM: vision 9423
Sandia Laboratories

ADDRESS: Albuquerque, NM 87185

ZIP:

SUBMITTED BY: Nick Bourgeois

PHONE: 264-8088

DEC OFFICE: Albuquerque

DO YOU HAVE SOURCES? YES XX NO ☐

REPORT TYPE
- ☐ SOFTWARE ERROR
- XX DOCUMENTATION ERROR
- ☐ INQUIRY
- ☐ FOR YOUR INFORMATION/SUGGESTION

PRIORITY
- ☐ LOW
- ☐ STANDARD
- ☒ HIGH

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

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

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

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH
11/45	2087	32 KW	RK05	RK05	☐

Problem: PAT004 in the RT-11 Software Dispatch, February 1978 for the FORTRAN IV compiler/OTS aborts in the R PAT step.

Details: Patches PAT001, PAT002, and PAT003 to the FORTRAN IV Version 2 compiler and OTS were installed as described in the February 1978 RT-11 Software Dispatch. PAT004, pages 97-102 would not work. None of the patches shown on pages 98, 99, 100, 101, and 102 for the various hardware configurations would work following the procedure illustrated on page 97. Attachment I illustrates the source file PAT004.NHD for the NHD version. Attachment II shows the compile and attempted installation of the patch into the FORTRAN OTSCOM module with the resulting "?PAT-F-Unable to locate module OTINIT" error message. Attachment III is the assembled listing.

Additional Question: Since we run many different hardware configurations, is it possible to apply the patches to the same OTSCOM module or are we going to have to keep 5 copies of OTSCOM, one for each type of hardware?

```
; PAT004.NHD
; NAB/SLA/1/36                    13-MAR-78/14-MAR-78
; RT-11 SOFTWARE DISPATCH         FEBRUARY 1978
; FORTRAN IV/RT-11 V2             SEQ 5 M
; THIS IS PATCH NUMBER 4 (NHD)
; THIS PATCH FIXES THE PROBLEM WITH SIMRT (NHD)
        .TITLE  OTINIT - NHD
WRNERR=2
ERRCOD=4
STSBY1=53
        .PSECT  OTS$I
X=.
.=X+2
STADDR:
.=X+224
        MOV     R0,R4
        NOP
  X+422
        BNE     4$
.=X+462
4$:
.=X+552
```

141902

```
        JSR     PC,PAT1
        BR      .+10
.=X+1054
        JSR     PC,PAT2
        NOP
.=X+1074
        JSR     PC,PAT3
        NOP
.=X+2010
PAT1:   TST     SIMRT
        BNE     1$
        MOV     #STADDR,@#40
        BIS     #20000,@#44
1$:     RTS     PC
PAT2:   TST     SIMRT
        BNE     1$
        BISB    #WRNERR,@#STSBYT
1$:     RTS     PC
PAT3:   TST     SIMRT
        BNE     1$
        BISB    #ERRCOD,@#STSBYT
1$:     RTS     PC
        .PSECT  OTS$P
        .BLKW   19.
SIMRT:  .BLKW
        .END
```

```
DATE

16-Mar-78

.TIM
16:16:43

.R MACRO
*PAT004,LP:<PAT004
ERRORS DETECTED:  0
*^C

.R PAT
*OTSCOM<OTSCOM,PAT004
?PAT-F-Unable to locate module OTINIT
```

141902

```
 1                                      ; PAT004.NHD
 2                                      ; NAB/SLA/1736                    13-MAR-7
 3                                      ; RT-11 SOFTWARE DISPATCH         FEBRUARY
 4                                      ; FORTRAN IV/RT-11 V2             SEQ 5 M
 5                                      ; THIS IS PATCH NUMBER 4 (NHD)
 6                                      ; THIS PATCH FIXES THE PROBLEM WITH SIMR
 7                                              .TITLE  OTINIT - NHD
 8            000002                    WRNERR=2
 9            000004                    ERRCOD=4
10            000053                    STSBYT=53
11  000000                                      .PSECT  OTS$I
12            000000'                   X=.
13            000002'                   .=X+2
14  000002                                STADDR:
15            000224'                   .=X+224
16  000224    010004                            MOV     R0,R4
17  000226    000240                            NOP
18            000422'                   .=X+422
19  000422    001017                            BNE     4$
20            000462'                   .=X+462
21  000462                                4$:
22            000532'                   .=X+532
23  000532    004767    001252                  JSR     PC,PAT1
24  000536    000403                            BR      .+10
25            001054'                   .=X+1054
26  001054    004767    000754                  JSR     PC,PAT2
27  001060    000240                            NOP
28            001074'                   .=X+1074
29  001074    004767    000752                  JSR     PC,PAT3
30  001100    000240                            NOP
31            002010'                   .=X+2010
32  002010    005767    000046'         PAT1:   TST     SIMRT
33  002014    001006                            BNE     1$
34  002016    012737    000002'  000040         MOV     #STADDR,@#40
35  002024    052737    020000   000044         BIS     #20000,@#44
36  002032    000207                  1$:       RTS     PC
37  002034    005767    000046'       PAT2:     TST     SIMRT
38  002040    001003                            BNE     1$
39  002042    152737    000002   000053         BISB    #WRNERR,@#STSBYT
40  002050    000207                  1$:       RTS     PC
41  002052    005767    000046'       PAT3:     TST     SIMRT
42  002056    001003                            BNE     1$
43  002060    152737    000004   000053         BISB    #ERRCOD,@#STSBYT
44  002066    000207                  1$:       RTS     PC
45  000000                                      .PSECT  OTS$P
46                                              .BLKW   19.
47  000046                                SIMRT:  .BLKW
48            000001                            .END
```

```
OTINIT - NHD     MACRO V03.01 16-MAR-78 16:19:14 PAGE 1-1
SYMBOL TABLE

ERRCOD= 000004           PAT2    002034R      002 SIMRT   000046R      003 STSBYT=
        PAT1    002010R      002  PAT3    002052R      002 STADDR  000002R      002

. ABS.  000000      000
        000000      001
OTSSI   002070      002
OTSSP   000050      003
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  300 WORDS  ( 2 PAGES)
DYNAMIC MEMORY AVAILABLE FOR  66 PAGES
PAT004,LP:<PAT004
```

141902

FIX FOR VT52 TEC FROM ERIC MORTON

If you are running RT TECO and are using the VT52 TEC.

SET TT: NOCRLF

This should clear up most of the garbage displayed on the screen.

DECUS HAS MOVED!!

As of August 14, 1978, the DECUS International Headquarters and DECUS U.S. Chapter offices will be located at Digital Equipment Corporation in Marlboro. Our new address is:

DECUS
MR2-3/E55
One Iron Way
Marlboro, Massachusetts 01752

Marlboro is not on Centrex, a direct access telephone system, so all calls will come thru the switchboard at (617) 481-9511.

DECUS extensions are as follows:

Central Number	4100
Executive Director	4120
Admin/Finance	4122
Order Processing	4135
Accounting	4136
Membership	4166
Publications	4131
Library	4177
U.S. Chapter	4141

MOVING OR REPLACING A DELEGATE?

Please notify us immediately to guarantee continuing receipt of DECUS literature. Allow up to six weeks for change to take effect.

( ) Change of Address
( ) Delegate Replacement

DECUS Membership No.: _____

Name: _____

Company: _____

Address: _____

_____

State/Country: _____

Zip/Postal Code: _____

Mail to: DECUS - ATT: Membership
One Iron Way, MR2-3
Marlboro, Massachusetts 01752 USA

Affix mailing label here. If label is not available, print old address here. Include name of installation, company, university, etc.

DECUS

BULK RATE
U.S. POSTAGE
PAID
DIGITAL EQUIPMENT
CORPORATION

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY
ONE IRON WAY, MR2-3
MARLBORO, MASSACHUSETTS 01752 USA