RT-11

AUGUST 1979

AD-C740B-16

THE SOFTWARE DISPATCH

digital

RT-11 SOFTWARE DISPATCH

Published by
Administrative Services Group, Software Services
Digital Equipment Corporation
P.O. Box F
Maynard, MA 01754

The RT-11 Software Dispatch complements the RT-11 V3B Software Dispatch Review. New and revised Software Product Descriptions, programming notes, software problems and solutions, and documentation corrections are published here. Much of the material is developed from Software Performance Report (SPR) answers significant to the general audience and is printed here to supplement the maintenance notebook (established by the Software Dispatch Review).

PRODUCTS SUPPORTED in the RT-11 SOFTWARE DISPATCH

APL-11 V1
BASIC-11/RT-11 V2
BASIC/RT Extensions V1
COS-350/2780
CTS-300 V3, V4, V5
CTS-300 DICAM V1
CTS-300 DICAM II V1
CTS-300/DIS V1
DECnet/RT V1
FOCAL/RT-11 V1B
FORTRAN Graphics Package V1.1
FORTRAN/RT-11 Extensions V1B
FORTRAN/RT-11 LSI Extensions V1
FORTRAN IV/RT-11 V2
GAMMA-11 F/B V2, V2C
Industrial BASIC/RT-11 V1
Lab Applications-11 V3
LSP-11 V1
MSB11 V1
MSB/FORTRAN IV V1
MU BASIC-11/RT-11 V2
PDL/RT-11 V1
PEAK-11 V2
PLOT 11/RT-11 V1.1
RT-11/03 FORTRAN Extensions V1
REMOTE/RT-11 V1
RT-11 V3, V3B
RT-11 (CTS-300)/LSI-11 2780 V2
RT-11/2780 (CTS-300/2780) V2
SSP-11/RT-11 V1

DISTRIBUTION

The Dispatch is directed to one software contact for each licensed Category A and B software product for one year after installation. No Mailing will be made to addresses without a software contact name. Address changes and requests for information about maintenance service after the first year should be sent to the nearest DIGITAL Field Office. For address changes, include the new address and mailing label from the most recently received publication.

Software binaries and sources are provided under licenses only. The standard Terms and Conditions, OEM Agreement and/or Quantity Discount Agreement contain the licenses for all binaries other than DECsystem-10.

Eleanor F. Hunter, Editor
Ann Owens, Associate Editor

The material in this document is for information purposes only and is subject to change without notice. Digital Equipment Corporation assumes no responsibility for any errors which may appear in this document. Comments on the contents of this publication should be directed to your local DIGITAL Field Office.

TABLE OF CONTENTS

	SEQ.NO.	PAGE
REVISED SPR USER LETTER		1
SAMPLE OF NEW USER BUSINESS REPLY CARD		2
NEW SPR FORM		3

BASIC-11/RT-11 V2

STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES	35 N	7
MAXIMUM ARRAY SUBSCRIPT SIZE	36 N	10

CTS-300 V05

DIBOL

MISLEADING ERROR MESSAGES (PATCH 11)	9 M	11
ERRONEOUS I/O ERROR (PATCH 12)	10 M	15

REDUCE

HOW TO REDUCE PAINLESSLY	1 N	17

CTS-300 RDCP (2780/3780) V1.0

ATTEMPT TO LOAD LPX.SYS BEFORE CMX.SYS UNDER XM MONITOR	4 M	19

DECnet-RT V1.0

USER'S GUIDE DOCUMENTATION ERRORS	2.1 N	21

FORTRAN IV/RT-11 V2.1

PATCH 4	5 M	23
OPEN FAILURE WITH TYPE='OLD' - PATCH 6	7 M	25
FORTRAN LIBRARY FUNCTION ERRTST - PATCH 7	8 M	27
REGISTER ALLOCATION PATCH 8	9 M	29
I/O FROM A FORTRAN COMPLETION ROUTINE - PATCH 10	12 M	33
FORTRAN FAILS TO COMPILE DO-LOOPS - PATCH 11	13 M	37
CALL CLOSE (FORTRAN LIBRARY SUBROUTINE) - PATCH 12	14 M	39
UNFORMATTED BYTE I/O - PATCH 13	15 F	41
LIST DIRECTED INPUT ERRORS - PATCH 14	16 M	45
DISP='DELETE' OPTION - PATCH 15	17 M	47
FORMATTED RECORD OUTPUT - PATCH 16	18 M	51
COMMON SUBEXPRESSION OPTIMIZATION - PATCH 17	19 M	53
CALL ASSIGN CARRIAGE CONTROL - PATCH 18	20 M	55
NON-PLAS VIRTUAL ARRAY INITIALIZATION - PATCH 19	21 M	57
BYTE COMPARISON AND COMMON SUBEXPRESSION OPTIMIZATION - PATCH 20	22 M	59
DIRECT ACCESS READ - PATCH 21	23 M	63
COMPLEX VARIABLE TO CONSTANT COMPARISON - PATCH 22	24 M	65

GAMMA-11 F/B V2C

GAMMA-11 SYSTEMS WITH RK07 AS A DEVICE	34 M	69

TABLE OF CONTENTS (CONT.)

	SEQ.NO.	PAGE
MU BASIC-11/RT-11 V2		
STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES	41 N	71
SIZING MU BASIC-11	42 N	75
ERROR IN TABLE 4-1 OF THE USER'S GUIDE	43 N	79
RESTRICTION OF USR RESIDENCY WHEN RUNNING IN FOREGROUND	44 N	81
NOTES ON PERFORMANCE PATCHES No. 4a, No. 4b, No.4c	45 N	82
MAXIMUM ARRAY SUBSCRIPT SIZE	46 N	83
RT-11 VØ3B-ØØ		
DOCUMENTATION		
CORRECTION OF ERROR RETURNS IN .SYNCH CALL	9 M	85
EXAMPLE CODE IN .FORK DOCUMENTATION IS INCORRECT	1Ø N	86
MISCELLANEOUS		
GENERATING A SINGLE JOB MONITOR MAY CAUSE AN UNDEFINED GLOBAL	5 M	87
MONITOR		
NON-STANDARD VECTOR ADDRESSES FOR RXØ1 AND RXØ2 SECOND CONTROLLER	25 M	89
ABORT DURING COMPLETION CAUSES SYSTEM FAILURES	26 M	91
SOURCE		
DL HANDLER SEEK AND UNIT CORRECTIONS	7 M	97
UTILITIES		
LIBR /U SWITCH PROBLEM	16 M	99
IMPORTANT RESTRICTIONS FOR SQUEEZE OPERATIONS	17 M	100
RT-11 CUMULATIVE INDEX		101
SOFTWARE PRODUCT DESCRIPTION (SPDs)		111
DIGITAL EQUIPMENT COMPUTER USERS SOCIETY (DECUS)		113

NEW SPR FORM

A new SPR form is being distributed (see following). The Key areas of change are:

1. Reversal of order of priorities, 1 through 5 instead of 5 through 1.

2. Capsulized definitions of the priorities on the form.

3. Typewriter compatibility to include boxes to be Xed.

4. SPR Centers updated.

5. Use of Customer Number as part of customer's address.

   Customers will be informed of their number.

6. Administrative fields (shaded area) used in processing SPRs have been added.

digital

SOFTWARE PERFORMANCE REPORT

FIELD NO.:	CORPORATE SPR NO.:	258203

PAGE _____ OF_____

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

OPERATING SYSTEM [	VERSION	SYSTEM PROGRAM OR DOCUMENT TITLE	VERSION OR DOCUMENT PART NO.	DATE

NAME: FIRM: ADDRESS: CUST. NO.:	DEC OFFICE	DO YOU HAVE SOURCES? YES □ NO □
	REPORT TYPE/PRIORITY □ PROBLEM/ERROR □ SUGGESTED ENHANCEMENT □ OTHER	1. □ HEAVY SYSTEM IMPACT 2. □ MODERATE SYSTEM IMPACT 3. □ MINOR SYSTEM IMPACT 4. □ NO SIGNIFICANT IMPACT 5. □ DOCUMENTATION/SUGGESTION
SUBMITTED BY: PHONE:	CAN THE PROBLEM BE REPRODUCED AT WILL? YES □ NO □	
ATTACHMENTS MAG TAPE □ FLOPPY DISKS □ LISTING □ DECTAPE □ OTHER:	COULD THIS SPR HAVE BEEN PREVENTED BY BETTER OR MORE DOCUMENTATION? YES □ NO □ PLEASE EXPLAIN IN PROVIDED SPACE BELOW.	

CPU TYPE	SERIAL NO.	MEMORY SIZE	DISTRIBUTION MEDIUM	SYSTEM DEVICE	DO NOT PUBLISH □

ALL SUBMISSIONS BECOME THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION.

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

ADMINISTRATIVE SERVICES GROUP, SWS

DIRECTIONS FOR COMPLETING SPR FORM

258203

The SPR form must be filled out completely and MUST BE TYPEWRITTEN in order to ensure proper processing. The shaded areas on the form should be left blank, they will be used by DIGITAL in processing the SPR.

The following is a brief summary of the information required:

OPERATING SYSTEM/MONITOR (SOFTWARE PRODUCT)
Monitor (software product) the system program runs under and its version number (e.g. RSX–11M V3, TOPS-10 V6.03). Document Title such as OS/8 Handbook.

SYSTEM PROGRAM & VERSION (OR DOCUMENT PART NUMBER)
The program in which the problem resides, e.g. FORTRAN V5A, BASIC V1B. If a monitor, write MONITOR (module). If a documentation error is being reported, the DEC order number of the manual should be entered here (e.g. DEC-11-ORSUB-A-D).

DATE:
Date of submittal using a three character abbreviation for month (e.g. 4-APR-79)

NAME AND ADDRESS:
Fill out the name of your installation's responsible software contact and complete mailing address. The information in this block will be used to return the acknowledgment copy.

CUST. NO.:
A permanent reference number which is assigned by DIGITAL. Customers will be informed of their number.

SUBMITTED BY AND PHONE:
Enter name and phone number of the author of the SPR.

DEC OFFICE:
Enter local DEC office (or SPR Center if European or Australian).

REPRODUCIBLE AT WILL, SOURCE AND DOCUMENTATION QUESTIONS
Check appropriate boxes.

REPORT TYPE/PRIORITY
Check appropriate box for Report Type **and** Priority.

Priority Definitions are as follows:

1. Most production work cannot be run, e.g. functions/jobs which are not usable are a major use of system, e.g. system won't boot, necessary peripherals cannot be used as intended.
2. Some production work cannot be run, e.g. certain jobs/functions are not usable, performance degradation, installation has insufficient excess capacity.
3. All production work can be run with some impact on user, e.g. significant manual intervention required, extra procedures, performance degradation but installation has excess capacity.
4. All production work can be run with no significant impact on user, e.g., problem can be easily patched, simple bypass procedure exists.
5. No system modifications needed to return to normal production, e.g., suggestion, consultation, documentation error.

ATTACHMENTS:
If attachments are included with SPR, describe materials sent and insure that the number from the top of this form appears on them. Printed examples must be dark. If magtape, include track and density.

CPU TYPE:
Enter model number of the processor (e.g. 1080, 8/A, 11/70, 2040).

SERIAL #:
Enter serial # of central processor. If there are two processors, enter serial number of first.

SYSTEM DEVICE:
The device on which the monitor resides (e.g. DOS/BATCH on RK05 where RK05 is system device).

DISTRIBUTION MEDIUM:
Indicate the medium on which you receive software (e.g. 9TR Magtape, DEC Tape, RX02, RK05).

PROBLEM DESCRIPTION:
A concise description of the problem in the form of PROBLEM:, DIAGNOSIS:, CURE: (if known), with references to circumstances surrounding its occurrence should be included. **Only one problem should be stated per SPR form.** Attempt to reduce the problem to a simple test case. If you cannot, include all programs and data in machine readable form. If a patch or interim solution exists, include it.

DO NOT PUBLISH:
Check this box if you do not want your SPR published in its original form. This does not guarantee that the solution will not be published if of universal value.

SPR SUBMISSION:
Upon completion of the SPR form **remove last copy** and send remainder to the nearest SPR center. Refer to the reverse side of this instruction sheet for a listing of SPR centers.

SOFTWARE PROBLEMS OR ENHANCEMENTS

Questions, problems, and enhancements to DIGITAL software should be reported on a Software Performance Report (SPR) form and mailed to the SPR Center at one of the following Digital Offices: *(SPR forms are available from the SPR Center).*

Areas Covered	SPR Center
United States; remainder of Far East, Middle East, Africa Latin America	Administrative Services Group, SWS P.O. Box F Maynard, Ma 01754
Canada	Digital Equipment Canada P.O. Box 11500 Ottawa, Ontario Canada K2H 8K8
United Kingdom, Bahraine, Egypt, Iraq, Jordan, Kuwait, Lebanon, Libya, Qatar, Oman, Saudi Arabia, Syria, United Arab Emirates, Yemen, Arab Republic.	Digital Equipment Corp. LTD Fountain House Butts Centre GB - Reading RG17QN England
Australia-Melbourne	Digital Equipment Aust. PTY. LTD 60 Park Street So. Melbourne Victoria Australia 3205
Australia-Sydney	Digital Equipment Aust. PTY. LTD 123 125 Willoughby Rd. P. O. Box 491 Crows Nest NSW Australia 2065
Brazil	Digital Equipment Comercio Ind. Rua Batatais 429 Esq AL Campin 01423 Jardim Paulista Sao Paulo 0100 Brazil
Caribbean	De Latin America P. O. Box 11038 Fernando Juncos Sta. Santurce PR 00910
France	Digital Equipment France 18, rue Saarinen France Silic 225 F - 94528 Rungis - Cedex France
Italy	Digital Equipment S.P.A. Viale Fulvio Testi 117 I-20092 Cinisillo Balsamo Milan, Italy
Japan	Digital Equipment Corp. INTL 3rd Floor Kowa Bldg. 8-7 Sanban Cho Chiyoda Ku Tokyo 102 Japan
New Zealand	Digital Equipment N.Z. LTD P.O. Box 17093 Greenlane, Auckland 5, New Zealand
Belgium, Holland, Luxemburg	Digital Equipment B.V. KAAP Horndreef 38 NL - Utrecht/Overvecht Holland
Sweden	Digital Equipment Corp. AB Englundavägen 7 S-171 24 Solna, Sweden
Denmark	Digital Equipment Corp. APS Kristineberg 3 DK-2100 Copenhagen Ø Denmark
Finland	Digital Equipment Corp. OY PL16 SF - 02201 ESPOO 20 Finland
Norway	Digital Equipment Corp. A/S Pottenmakerveien 8 N - Oslo 5 Norway
Austria, East Germany, West Germany, Poland, Hungary, Rumania, Czechoslovakia, Russia, Bulgaria	Digital Equipment Corp. GMBH Wallsteinplatz 2 D - 8 Munich 40 West Germany
Israël	DECSYS Computers LTD. 4, Yirmiyahou Str. P.O. Box 6359 IL - Tel-Aviv 63505 Israël
Greece, Portugal, Spain, Switzerland, Yugoslavia & Sina (Morocco, Algeria, Tunisia, Cyprus, Turkey, Malta)	Digital Equipment Corp. SA 9, route des Jeunes 1211 Geneva 26 Switzerland

STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES

The BASIC-11 User's Guide gives an example of an assembly language routine that manipulates numeric values. There is, however, no example of an ALR that uses the string manipulation routines ($FIND, $ALC, $DEALC, $STORE).

This article provides an example of an ALR that accepts two string arguments; the first is a string (eg "BOSTON" or A$) that will be stored in, the variable named as the second argument (eg B$). If the first argument is null (eg "") the original contents of the named variable are preserved.

To accomplish this the ALR uses each of the string manipulation routines provided by BASIC-11 as follows:

$FIND	returns the location and length of the string in arg1
$ALC	allocates a temporary string
$STORE	moves the string in arg1 to the temporary string
$STORE	moves the temporary string to the variable named in arg2
$DEALC	deallocates the temporary string and resets the stack

The routine is functionally similar to a direct string assignment statement in BASIC-11;

```
        ie. CALL CAF1('BOSTON',B$)
is equivalent to:
            B$='BOSTON'
```

It may be used as a model upon which to base more complex (and more useful !) ALR's. However, it does not necessarily represent the most efficient way to code the routine. It is intended as a clear example of how to set up the parameters required by the BASIC-11 string manipulation routines.

```
        .TITLE  CAF1

        .PSECT  SUBRS,RO,I

        .GLOBL  CAFTAB
CAFTAB: .BYTE   4
        .ASCII  'CAF1'
        .EVEN
        .WORD   CAF1


        .GLOBL  $ARGER,$BOMB
        .GLOBL  $ALC,$DEALC,$FIND,$STORE

;
;       THIS ROUTINE TAKES TWO STRING ARGUMENTS, THE FIRST IS READ,
;       THE SECOND MUST BE WRITEABLE:
;       EG.
;               CALL CAF1('BOSTON',B$) OR
;               CALL CAF1(A$,B$)
;
;       FUNCTION: THE ROUTINE WILL REPLACE THE CONTENTS OF THE STRING
;                 NAMED IN THE SECOND ARGUMENT WITH THE STRING SPECIFIED
;                 IN THE FIRST ARGUMENT UNLESS THE FIRST ARGUMENT IS NULL.
;
;
CAF1:   CMPB    (R5)+,#2            ;ARE THERE 2 ARGUMENTS?
        BEQ     2Ø$                 ;YES
1Ø$:    JMP     $ARGER              ;NO
2Ø$:    CMPB    (R5)+,#2Ø2          ;ARE WE BEING CALLED BY B-11
                                    ;WITH ARG DESCRIPTORS?
        BNE     99$                 ;NO

3Ø$:    MOV     -4(R5),R4           ;GET PTR TO 1ST ELEMENT IN ADL
        JSR     PC,GETDSC           ;GET DESCRIPTOR WORD OF 1ST ARGUMENT
        BIC     #16Ø2ØØ,R3          ;IS IT A STRING?
        CMP     #11ØØ,R3
        BNE     1Ø$                 ;NO
        JSR     PC,GETDSC           ;YES - GET DW OF 2ND ARG
        BIC     #16ØØØØ,R3          ;IS IT A WRITEABLE STRING?
        CMP     #11ØØ,R3
        BNE     1Ø$                 ;NO

6Ø$:    MOV     -4(R5),RØ           ;MOVE PTR TO 1ST ADL ELEMENT TO RØ
        MOV     (RØ),RØ             ;MOVE 1ST ARG'S DESCR. PTR TO RØ
        ADD     #2,RØ               ;MAKE RØ PTR TO 1ST ARG'S STR REF PTR
        MOV     (RØ),RØ             ;MOVE STR REF PTR TO RØ
        MOV     #1ØØ,R1             ;MOVE 1ØØ TO R1
        SUB     #2,R5               ;RESTORE R5
        JSR     PC,$FIND            ;FIND STRING
        BCS     7Ø$                 ;BRANCH IF ERROR IN $FIND

61$:    TST     R1                  ;TEST FOR NULL STRING
        BEQ     8Ø$                 ;BRANCH IF NULL
        MOV     R1,RØ               ;STRING LENGTH
        MOV     #1ØØ,R1
        JSR     PC,$ALC
        BCS     71$                 ;BRANCH IF ERROR IN $ALC
```

```
62$:    MOV     SP,R1           ;MOVE TEMP STR'S STR REF PTR TO R1
        MOV     -2(R5),RØ       ;MOVE PTR TO 1ST ADL ELEMENT TO RØ
        MOV     (RØ),RØ         ;MOVE 1ST ARG'S DESCR. PTR TO RØ
        ADD     #2,RØ           ;MAKE RØ PTR TO 1ST ARG'S STR REF PTR
        MOV     (RØ),RØ         ;MOVE STR REF PTR TO RØ
        MOV     #1ØØ,R2         ;MOVE 1ØØ TO R2
        JSR     PC,$STORE       ;STORE STRING INTO TEMP STRING
        BCS     73$             ;BRANCH IF ERROR IN $STORE

63$:    MOV     R1,RØ           ;MAKE RØ STR REF PTR OF TEMP STR
        MOV     -2(R5),R1       ;MOVE PTR TO 1ST ADL ELEMENT TO R1
        ADD     #2,R1           ;MAKE R1 POINT TO 2ND ADL ELEMENT
        MOV     (R1),R1         ;MOVE 2ND ARG'S DESCR. PTR TO R1
        ADD     #2,R1           ;MAKE R1 PTR TO 2ND ARG'S STR REF PTR
        MOV     (R1),R1         ;MOVE 2ND ARG'S STR REF PTR TO R1
        MOV     #1ØØ,R2         ;MOVE 1ØØ TO R2
        JSR     PC,$STORE       ;STORE TEMP STRING INTO 2ND ARG
        BCS     73$             ;BRANCH IF ERROR IN $STORE

64$:    JSR     PC,$DEALC
        BCS     72$             ;BRANCH IF ERROR IN $DEALC
        RTS     PC              ;RETURN

7Ø$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $FIND'
        .EVEN

71$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $ALC'
        .EVEN

72$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $DEALC'
        .EVEN

73$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $STORE'
        .EVEN

8Ø$:    RTS     PC              ;RETURN

99$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN CALLING ID'
        .EVEN


;
;GETDSC RETURNS NEXT ARGUMENT'S DESCRIPTOR WORD (DW)

;INPUTS:
;       R4 POINTS TO THE WORD IN DESCRIPTOR LIST

;OUTPUTS:
;       R3 CONTAINS THE DW FOR CURRENT ARG
;       R4 IS UPDATED TO POINT TO NEXT ELEMENT IN ADL

GETDSC: MOV     (R4)+,R3        ;GET DESCRIPTOR
        BIT     #1,R3           ;IS IT A POINTER?
        BNE     1Ø$             ;NO
        MOV     (R3),R3         ;YES - GET ACTUAL DESCRIPTOR
1Ø$:    RTS     PC
        .END
```

BASIC-11/RT-11 V2 Seq 36 N

1 of 1

MAXIMUM ARRAY SUBSCRIPT SIZE (CF)

There is some confusion about the size of arrays and virtual arrays in BASIC-11. The following information should help users decide how best to use arrays and how to calculate maximum array sizes.

- In-memory arrays (e.g. DIM X(a), Y(b,c))

The maximum size of an array depends on the amount of memory available to store the array. In any case, the theoretical maximum subscript value is 32766.

Note that BASIC-11 arrays start with element Ø, not element 1. Therefore, an array DIMensioned with the following statement:

```
DIM X(4ØØØ)
```

will contain 4ØØ1 elements, and an array DIMensioned with the following statement:

```
DIM Y(24,3)
```

will contain 1ØØ elements, (25*4).

- Virtual arrays (e.g. DIM #1,X(a) and DIM #1,Y(b,c))

The maximum subscript value for a one-dimensional virtual array is 32766.

Note that BASIC-11 virtual arrays start with element Ø, not element 1, exactly like in-memory arrays. Therefore, the maximum number of array elements in a one-dimensional virtual array is 32767.

The maximum subscript values for a two-dimensional virtual array are determined by the following formula:

```
(b+1)*(c+1) must be less than 32768
```

where b and c are the subscripts for the virtual array DIMensioned with the following statement:

```
DIM #1,Y(b,c)
```

Note that the above formula, restated, specifies that the maximum number of array elements in a two-dimensional virtual array is 32767, which corresponds to the maximum number of array elements in a one-dimensional virtual array.

CTS-3ØØ V5
REDUCE

Seq 1 N

1 of 1

HOW TO REDUCE PAINLESSLY (LG)

We would like to clarify a point concerning the REDUCE program that may be causing minor headaches for some users. The documentation for REDUCE (CTS-300 System User's Guide (AA-C747B-TC), Chapter 11) strongly implies that you can use an input and an output file specification in this manner:

```
     .R[U] REDUCE
     *device:outputfilename.type = device:inputfilename.type
```

This is NOT so. REDUCE will accept only an input file specification. If you provide both an input and an output file specification, the output file specification (everything to the left of the equals sign) will be ignored by REDUCE.

For example, if you run REDUCE and enter:

```
     RK2:YRFILE.XYZ=RK1:MYFILE.TSD
```

the result will be a REDUCEd MYFILE.TSD located on RK1:.

DECnet-RT V1.Ø
   for RT-11 FB/XM V3.Ø
DOCUMENTATION

Seq 2.1 N
1 of 1
Supersedes article dated May 79

USER'S GUIDE DOCUMENTATION ERRORS (SPR 21942, WMD)

There are several documentation errors in the DECnet/RT User's Guide regarding the descriptions of the Macro Argument Blocks.

1) On page 8-38 following 'Argument Block:'

   A) The ".chan" number cannot be Ø or NSP will return an error.

   B) The "-3" in byte number nine should be a "-2".

   C) There is an additional zero word at byte 1Ø. Necessary for proper operation.

2) On page 8-36 following 'Argument Block:' there should be an additional two words of zero at bytes 16. and 18.

UNFORMATTED BYTE I/O - PATCH 13 (WM)

Problem:

Unformatted Byte I/O allocates words rather than bytes.

Solution:

1. Type in the following MACRO file and name it PAT13.MAC

```
                .TITLE      $UIO
                .IDENT      /03/
                .PSECT      OTS$I

S=.
.=S+424
                JSR         PC,PAT13A
.=S+514
                NOP
                NOP
                NOP
                NOP
                NOP
.=S+630
                NOP
                NOP
                NOP
                NOP
                NOP
.=S+644
                JSR         PC,PAT13B
.=S+734
OCHECK:
.=S+770
PAT13A:
                MOVB        123(R3),R1
                CMP         R1,#1
                BEQ         7$
                INC         16(R0)
                BIC         #1,16(R0)
7$:
                RTS         PC

PAT13B:
                DEC         22(R0)
                JSR         PC,OCHECK
                RTS         PC

                .END
```

2. Assemble the patch using MACRO-11.

   ```
   .R MACRO
   *PAT13=PAT13
   *↑C
   ```

3. Install the patch, using PAT, to the most recently patched OTS file: OTSCOM.OBJ

   Note: Make a copy of OTSCOM before you patch it just in case something goes wrong.

   ```
   .R PAT
   *OTSCOM=OTSCOM,PAT13
   ```

4. Rebuild the OTS using the procedure described in the FORTRAN Installation Guide.

5. Test the patch by compiling, linking, and running the following program.

   Note: If you have a file named FTN1.DAT rename it before running the following program or you will lose your current file.

```
                BYTE B(4)
                INTEGER*2 A(4)
                A(1)='AB'
                A(2)='CD'
                A(3)='EF'
                A(4)='GH'
                WRITE(1)A
                REWIND 1
                READ(1)B
                WRITE(7,100)B
        100     FORMAT(1X,4A1)
                END
```

which should produce the following result.

```
                ABCD
                STOP --
```

STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES

The MU BASIC-11 User's Guide gives an example of an assembly language routine that manipulates numeric values. There is, however, no example of an ALR that uses the string manipulation routines ($FIND, $ALC, $DEALC, $STORE).

This article provides an example of an ALR that accepts two string arguments; the first is a string (eg "BOSTON" or A$) that will be stored in the variable named as the second argument (eg B$). If the first argument is null (eg "") the original contents of the named variable are preserved.

To accomplish this the ALR uses each of the string manipulation routines provided by MU BASIC-11 as follows:

$FIND	returns the location and length of the string in arg1
$ALC	allocates a temporary string
$STORE	moves the string in arg1 to the temporary string
$STORE	moves the temporary string to the variable named in arg2
$DEALC	deallocates the temporary string and resets the stack

The routine is functionally similar to a direct string assignment statement in MU BASIC-11;

```
        ie. CALL CAF1('BOSTON',B$)
is equivalent to:
            B$='BOSTON'
```

It may be used as a model upon which to base more complex (and more useful !) ALR's. However, it does not necessarily represent the most efficient way to code the routine. It is intended as a clear example of how to set up the parameters required by the MU BASIC-11 string manipulation routines.

```
        .TITLE  CAF1

        .PSECT  SUBRS,RO,I

        .GLOBL  CAFTAB
CAFTAB: .BYTE   4
        .ASCII  'CAF1'
        .EVEN
        .WORD   CAF1


        .GLOBL  $ARGER,$BOMB
        .GLOBL  $ALC,$DEALC,$FIND,$STORE

;
;       THIS ROUTINE TAKES TWO STRING ARGUMENTS, THE FIRST IS READ,
;       THE SECOND MUST BE WRITEABLE:
;       EG.
;               CALL CAF1('BOSTON',B$) OR
;               CALL CAF1(A$,B$)
;
;       FUNCTION: THE ROUTINE WILL REPLACE THE CONTENTS OF THE STRING
;                 NAMED IN THE SECOND ARGUMENT WITH THE STRING SPECIFIED
;                 IN THE FIRST ARGUMENT UNLESS THE FIRST ARGUMENT IS NULL.
;
;
;
CAF1:   CMPB    (R5)+,#2        ;ARE THERE 2 ARGUMENTS?
        BEQ     2Ø$             ;YES
1Ø$:    JMP     $ARGER          ;NO
2Ø$:    CMPB    (R5)+,#2Ø2      ;ARE WE BEING CALLED BY B-11
                                ;WITH ARG DESCRIPTORS?
        BNE     99$             ;NO

3Ø$:    MOV     -4(R5),R4       ;GET PTR TO 1ST ELEMENT IN ADL
        JSR     PC,GETDSC       ;GET DESCRIPTOR WORD OF 1ST ARGUMENT
        BIC     #16Ø2ØØ,R3      ;IS IT A STRING?
        CMP     #11ØØ,R3
        BNE     1Ø$             ;NO
        JSR     PC,GETDSC       ;YES - GET DW OF 2ND ARG
        BIC     #16ØØØØ,R3      ;IS IT A WRITEABLE STRING?
        CMP     #11ØØ,R3
        BNE     1Ø$             ;NO

6Ø$:    MOV     -4(R5),RØ       ;MOVE PTR TO 1ST ADL ELEMENT TO RØ
        MOV     (RØ),RØ         ;MOVE 1ST ARG'S DESCR. PTR TO RØ
        ADD     #2,RØ           ;MAKE RØ PTR TO 1ST ARG'S STR REF PTR
        MOV     (RØ),RØ         ;MOVE STR REF PTR TO RØ
        MOV     #1ØØ,R1         ;MOVE 1ØØ TO R1
        SUB     #2,R5           ;RESTORE R5
        JSR     PC,$FIND        ;FIND STRING
        BCS     7Ø$             ;BRANCH IF ERROR IN $FIND

61$:    TST     R1              ;TEST FOR NULL STRING
        BEQ     8Ø$             ;BRANCH IF NULL
        MOV     R1,RØ           ;STRING LENGTH
        MOV     #1ØØ,R1
        JSR     PC,$ALC
        BCS     71$             ;BRANCH IF ERROR IN $ALC
```

```
62$:    MOV     SP,R1           ;MOVE TEMP STR'S STR REF PTR TO R1
        MOV     -2(R5),RØ       ;MOVE PTR TO 1ST ADL ELEMENT TO RØ
        MOV     (RØ),RØ         ;MOVE 1ST ARG'S DESCR. PTR TO RØ
        ADD     #2,RØ           ;MAKE RØ PTR TO 1ST ARG'S STR REF PTR
        MOV     (RØ),RØ         ;MOVE STR REF PTR TO RØ
        MOV     #1ØØ,R2         ;MOVE 1ØØ TO R2
        JSR     PC,$STORE       ;STORE STRING INTO TEMP STRING
        BCS     73$             ;BRANCH IF ERROR IN $STORE

63$:    MOV     R1,RØ           ;MAKE RØ STR REF PTR OF TEMP STR
        MOV     -2(R5),R1       ;MOVE PTR TO 1ST ADL ELEMENT TO R1
        ADD     #2,R1           ;MAKE R1 POINT TO 2ND ADL ELEMENT
        MOV     (R1),R1         ;MOVE 2ND ARG'S DESCR. PTR TO R1
        ADD     #2,R1           ;MAKE R1 PTR TO 2ND ARG'S STR REF PTR
        MOV     (R1),R1         ;MOVE 2ND ARG'S STR REF PTR TO R1
        MOV     #1ØØ,R2         ;MOVE 1ØØ TO R2
        JSR     PC,$STORE       ;STORE TEMP STRING INTO 2ND ARG
        BCS     73$             ;BRANCH IF ERROR IN $STORE

64$:    JSR     PC,$DEALC
        BCS     72$             ;BRANCH IF ERROR IN $DEALC
        RTS     PC              ;RETURN

7Ø$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $FIND'
        .EVEN

71$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $ALC'
        .EVEN

72$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $DEALC'
        .EVEN

73$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN $STORE'
        .EVEN

8Ø$:    RTS     PC              ;RETURN

99$:    JSR     R1,$BOMB
        .ASCIZ  'ERROR IN CALLING ID'
        .EVEN


;
;GETDSC RETURNS NEXT ARGUMENT'S DESCRIPTOR WORD (DW)

;INPUTS:
;       R4 POINTS TO THE WORD IN DESCRIPTOR LIST

;OUTPUTS:
;       R3 CONTAINS THE DW FOR CURRENT ARG
;       R4 IS UPDATED TO POINT TO NEXT ELEMENT IN ADL

GETDSC: MOV     (R4)+,R3        ;GET DESCRIPTOR
        BIT     #1,R3           ;IS IT A POINTER?
        BNE     1Ø$             ;NO
        MOV     (R3),R3         ;YES - GET ACTUAL DESCRIPTOR
1Ø$:    RTS     PC
        .END
```

SIZING MU BASIC-11

This article contains information about the components of MU BASIC-11. It is intended to help users estimate the relative cost of including certain options in their system. All figures should be regarded as approximate.

1. Comparison table of V1 and VØ2 Partition Sizes:

The following are comparable user partition size figures for the two versions of MU BASIC (figures are in Kwords):

MU BASIC Version	RT-11 Version	Monitor	No. of Users 1	2	3	4
1	2C	SJ	13.1	6.3	4.1	3.Ø
1	2C	FB	11.2	5.4	3.4	2.5
1	3B	SJ	12.9	6.2	4.Ø	2.9
1	3B	FB	11.1	5.1	3.3	2.4
2* (average)	3B	FB	6.8	3.2	2.Ø	1.5
2* (minimal)	3B	FB	1Ø.6	5.2	3.3	2.4

The total memory available to 4 users varies from 1ØØØØ to 5ØØØ words (free of overhead), depending on link and configuration options. This range represents a minimal workable system to a 'full' configuration with most options included and a generous number of channels and buffers, respectively.

NOTE

The 'average' system referred to above includes a resident USR. If the USR is declared non-resident, then around 2ØØØ words are freed for user partition space. The cost in performance of declaring the USR non-resident is discussed later in section 3.

For details of the 'minimal' system referred to above, see section 2 below.

2. 'Minimal' MU BASIC-11 VØ2 system

The 'minimal' system referred to above excludes all space consuming link-time options (e.g. PRINT USING, SUBSTITUTE, RESEQUENCE, transcendental functions, double precision and long error messages), all non-essential configuration-time optional functions (e.g. TAB, RCTRLO, TTYSET, RND, ABS, SGN, BIN, OCT, LEN, ASC, CHR$, POS, SEG$, VAL, TRM$, STR$, PI, INT, DAT$, CLK$ and ERR), and takes minimal but workable settings for all other configuration parameters. The USR is set to swap, thereby saving 2ØØØ words (see section 3 below).

The cost in terms of space of including link-time options is shown below (in decimal words):

PRINT USING	2Ø	SUBSTITUTE	196
RESEQUENCE	199	Long Error Messages	72Ø
Trnscndntl Functions	574	Double Precision	1Ø35

The cost in terms of space of including configuration-time optional functions is shown below (in decimal words):

TAB	25	SYS	198*	RCTRLO	4
ABORT	2Ø*	TTYSET	352	(R)CTRLC	18*
RND	42	ABS	9	SGN	21
BIN	29	OCT	33	LEN	11
ASC	17	CHR$	15	POS	97
SEG$	81	VAL	37	TRM$	26
STR$	25	PI	7	INT	8
DAT$	11	CLK$	11	ERR	22

*indicates a function that is needed even in a minimal system (by INIT.BØØ)

Total: about 11ØØ decimal words
Total for minimal system: about 87Ø decimal words

These figures can be used to calculate the space that will be available for user partitions according to the following formula:

$$U= S - ((T + V) / n)$$

Where: U= Usable partition size per user
S= Space per user shown for minimal system in section 1
T= Total cost of options selected from tables above (remember that SYS, ABORT and (R)CTRLC are already included in the minimal system)
V= Value dependent on whether USR is resident, no. of channels and buffers selected at configuration time (see sections 3 & 4 below)
n= No. of users

Example: A 4 user system with: resident USR (see section 3 below), more User Channels etc. (V= 218Ø, see section 4 below); PRINT USING, RESEQUENCE, SUBSTITUTE and the transcendental functions included at link-time; SYS, ABORT, (R)CTRLC, TTYSET, LEN, POS, SEG$, STR$, CHR$, ERR, DAT$, CLK$ included at configuration-time;

$$U= 24ØØ - ((162Ø+218Ø)/4)= 24ØØ - 95Ø= 145Ø \text{ words per user}$$

3. USR - To swap or not to swap?

The bare facts are as follows:

The USR takes up 2ØØØ words of memory when resident (non-swapped). This means that there are 2ØØØ words that may be used either for user partition space or for a resident USR. If the USR is set to swap, you get 2ØØØ words more user partition space, but pay a penalty in terms of performance. If the USR is kept resident, you optimize system response time, but lose user partition space.

To help you decide whether to swap or not swap the USR, the following approximate timings for login response will help; the system activity during simultaneous multiple login is fairly representative of general system performance.

Figures (in seconds) are taken from 11/Ø3 with dual RXØ1; timed from HELLO to READY, 4 users, simultaneous login, MUBAS with performance improvements:

Format: USR resident/USR swap

No. of System Buffers specified in config.	1	2
With NOTICE file	43/1Ø4	23/82
Without NOTICE file	32/66	19/44

4. Cost of Configuration-time options

The cost of configuration-time options for XM systems is explained in section 2.6 of the MU BASIC-11 Release Notes (System Data Area). For FB systems, the figures are similar and are reproduced below:

Guideline	Add this no. of words	Example	Add
For resident USR	2ØØØ	US:Y	2ØØØ
For every device type	7	LPØP DXØP,1P	7 7
For highest device unit no.	(unit no.)+1	DXØP,1P	1+1=2
For highest terminal number	(term. no.)+1	AØ-15	15+1=16
For each system buffer	(size of system buffer)+1Ø	2,256	512+2Ø=532
For system I/O area	(size of I/O area)+3	256	256+3=259
For every channel (FB)	(no. of channels +1)*16	UC:16 SC: 4	(2Ø+1)*16=336
For every channel (XM)	(no. of channels +1)*19	UC:16 SC: 4	(2Ø+1)*19=399

MU BASIC-11/RT-11 V2 Seq 43 N

1 of 2

ERROR IN TABLE 4-1 OF THE USER'S GUIDE (CF)

Following is a corrected version of Table 4-1, page 4-11 of the MU BASIC-11 User's Guide. The changes are indicated by an asterisk (*) to the left of the changed line.

Table 4-1
Using String Access Routines

Routine	Program Setup	Result with No Errors Detected	Result with Errors Detected

$FIND
(return location and length of string)
	R0<-string reference pointer
	R1<-100
	R5<-initial value
	Execute: JSR PC, $FIND
		R0 = address of first string character
		R1 = length of string
		R2 = 100
		R3,R4,R5 unchanged
		C-bit = 0 (char)
		Z-bit = 1 if a null string (R1=0)
			R0 contains error code:
			if R0=1, R1 did not equal 100
			if R0=2, R5 did not contain correct initial value
			R3,R4,R5 unchanged
			C-bit = 1

$ALC
(allocate temporary string)*
	R0<-required string length
	R1<-100
	R5<-initial value
	Execute: JSR PC, $ALC
	✱
		R0 = address of first string character
		R1 = length of string
		R2 = 100
		R3,R4,R5 unchanged
		C-bit = 0 (char)
		Z-bit = 1 if a null string (R1=0)
		SP = string reference pointer
		stack contains several words of internal pointers. Remove these words from the stack by the $DEALC routine
			R0 contains error code:
			if R0=0, indicates insufficient free space for requested string
			If R0=1, R1 did not equal 100
			if R0=2, R5 did not contain correct initial value
			R3,R4,R5 unchanged
			C-bit = 1

$STORE (store value of a string in a second string, make first string null)	R0<-string reference pointer of string to be copied R1<-string reference pointer of receiving string R2<-100 R5<-initial value Execute: JSR PC, $STORE	R0,R1,R2,R3,R4,R5 unchanged C-bit = 0 string whose pointer was in R0 is null string whose pointer was in R1 contains former value of the other string	R0 contains error code: if R0=1, R2 did not equal 100 if R0=2, R5 did not contain correct initial value R1,R2,R3,R4,R5 unchanged C-bit = 1

$DEALC
(remove from stack the internal pointers produced by $ALC routine)*
✱
	Return stack to the state that it was immediately following $ALC routine. Do this by removing any words you have added to the stack since calling the $ALC routine; this ensures that the string reference pointer is in the SP.
	R2<-100
	R5<-initial vlaue
	Execute: JSR PC, $DEALC
		R0,R1,R2,R3,R4,R5 unchanged
		C-bit = 0
		Stack returned to the state that existed before $ALC was called
			R0 contains error code:
			if R0=1, R2 did not equal 100
			if R0=2, R5 did not contain correct initial value
			R1,R2,R3,R4,R5 unchanged
			C-bit = 1
			Stack

✱ * Any temporary string created by $ALC must be removed by $DEALC before the ALR ends.

RESTRICTION ON USR RESIDENCY WHEN RUNNING IN FOREGROUND (CF)

When you are running MU BASIC-11 in the foreground and you have specified a resident USR in the configuration file, there are certain occasions when the USR will nevertheless be swapped out. On these occasions, the operating system will issue the message:

```
        ?MON-F-Ill usr xxxxxx
```

and halt MU BASIC-11. The circumstances under which this will happen are described below:

1) Under the operating system's default setting of USR SWAP, you attempt to load a background job which, because it is too big to fit below the USR, causes the USR to swap out. When the operating system next attempts to reload the USR for the foreground job (MU BASIC-11), MU BASIC-11 is not expecting the USR to need to be reloaded and so no USR reload address exists.

2) Under the operating system's default setting of USR SWAP, you have successfully loaded a background job which is running and which at some point attempts to expand into the space occupied by the USR (as is occasionally done by TECO, for example). This expansion causes the USR to swap out.
   When the operating system next attempts to reload the USR for the foreground job (MU BASIC-11), MU BASIC-11 is not expecting the USR to need to be reloaded and so no USR reload address exists.

One way to avoid this happening is to use the RT-11 SET command to specify a resident USR:

```
        SET USR NOSWAP.
```

This will prevent the USR from being swapped out ever, which means that any attempt to load an oversized background job will not succeed and that any attempt to expand over the USR will also fail, with no ill effects on the foreground job.

Another way to avoid this happening is to specify the USR to be non-resident in the MU BASIC-11 configuration file. This will ensure that MU BASIC-11 sets up a USR reload address within its own (foreground) region, which will make it independent of whatever the background job is doing.

Note that the first of these two 'ways around' results in faster system performance due to the perpetually resident USR, traded off against the possibility that certain background jobs may be rejected. The second allows for larger background jobs to be loaded and run, traded off against slower system performance.

MU BASIC-11/RT-11 V2

Seq 45 N

1 of 1

NOTES ON PERFORMANCE PATCHES No. 4a, No. 4b, No. 4c (CF)

Note that if you follow the recommendations laid out in Performance patches No. 4a, No. 4b, and No. 4c to COMPILE INIT.BØØ and INITH.BØØ, the resulting COMPILEd files are still subject to the standard MU BASIC-11 restriction on transportability of COMPILEd programs between different versions of MU BASIC-11 (See Language Reference Manual, section 9.1Ø, and previous article in this publication entitled "COMPILEd program transportability").

Therefore, you are advised to keep a copy of the SAVE image of those programs, so that you can later create COMPILEd versions of them using a different version of MU BASIC-11 (e.g. when changing from a single precision to a double precision MU BASIC-11).

An example of such a situation follows:

```
RUN MUBASA               (a single precision version)
     .
     .
     .
OLD INIT.BØØ             (this can be either a SAVEd version or a
                          version that was previously COMPILEd using
                          MUBASA)
SAVE INIT.SVE            (this will SAVE the program as a SAVE image)
RUN EXIT                 (exit from MUBASA)

RUN MUBASB               (a double precision version)
     .
     .
     .
OLD INIT.SVE             (this will read in the SAVE image)
COMPILE INIT.BØØ         (this will create a COMPILEd INIT.BØØ for use
                          with MUBASB)
RUN EXIT                 (exit from MUBASB)
```

MU BASIC-11/RT-11 V2 Seq 46 N

1 of 1

MAXIMUM ARRAY SUBSCRIPT SIZE (CF)

There is some confusion about the size of arrays and virtual arrays in MU BASIC-11. The following information should help users decide how best to use arrays and how to calculate maximum array sizes.

* In-memory arrays (e.g. DIM X(a), Y(b,c))

The maximum size of an array depends on the amount of memory available to store the array. In any case, the theoretical maximum subscript value is 32766.

Note that MU BASIC-11 arrays start with element Ø, not element 1. Therefore, an array DIMensioned with the following statement:

```
DIM X(4ØØØ)
```

will contain 4ØØ1 elements, and an array DIMensioned with the following statement:

```
DIM Y(24,3)
```

will contain 1ØØ elements, (25*4).

* Virtual arrays (e.g. DIM #1,X(a) and DIM #1,Y(b,c))

The maximum subscript value for a one-dimensional virtual array is 32766.

Note that MU BASIC-11 virtual arrays start with element Ø, not element 1, exactly like in-memory arrays. Therefore, the maximum number of array elements in a one-dimensional virtual array is 32767.

The maximum subscript values for a two-dimensional virtual array are determined by the following formula:

```
(b+1)*(c+1) must be less than 32768
```

where b and c are the subscripts for the virtual array DIMensioned with the following statement:

```
DIM #1,Y(b,c)
```

Note that the above formula, restated, specifies that the maximum number of array elements in a two-dimensional virtual array is 32767, which corresponds to the maximum number of array elements in a one-dimensional virtual array.

RT-11 SOFTWARE DISPATCH
CUMULATIVE INDEX
AUGUST 1979

This is a complete listing of all articles for current versions of RT-11 and related products. In the case of subordinate software, **missing sequence numbers may pertain to problems unique to interaction with previous versions of the same product or other major operating systems.**

IMPORTANT!

Retracted articles are indicated: RETRACTION.

Flags are currently being installed for all articles. The flags and definitions are as follows:

M = Mandatory Patch. These patches correct errors in the software product. All users are required to apply these patches to maintain consistent "user level" unless the accompanying article specifies otherwise.

F = Optional Feature Patch. These patches extend or configure functionality into the product. These functions will be treated as a supported part of the product for the duration of the current release and will be incorporated with any future release, unless otherwise stated.

R = Restriction. These articles discuss areas that will not be patched in the current release because they require major modification or because they are not consistent with the design of the product. Restrictions, except those described as permanent, are reviewed and modified when possible as part of the normal release cycle.

N = NOTE. These articles provide explanatory information that supplements the manual set and provide more detailed information about a program or package. They also provide procedural information to make it easier to use a program or package.

APL-11 V1

Component	Sequence	Mon/Yr
APL.SAV PROGRAM PATCHES		
ERRONEOUS "DEFINITION ERROR" DURING FUNCTION EDITING	01 M	Nov 77
LOSS OF LOWER-CASE ON RE-ENTRY TO APL-11	02 M	Nov 77
APL WORKSPACE	03 R	Nov 77
"SYSTEM ERROR"S GENERATED BY NULL LINE ELEMENTS	04	Dec 77
INTERNAL MEMORY ALLOCATION PROBLEMS	05 M	Dec 77
ERROR FOR SCALAR RESULT OF DECODE OR INNER PRODUCT OPERATION	06 M	Feb 78
SYSTEM ERROR ON PARAMETER RETURN	07 M	May 78

BASIC-11/RT-11 V2

Component	Sequence	Mon/Yr
RESEQUENCE PRODUCES AN INCORRECT PROGRAM UNDER CERTAIN CONDITIONS	01 M	Aug 78
PRINT USING	02 M	Jun 78
MAX SIZE OF LINE ENTERED TO BASIC-11	03 M	Jun 78
REM STATEMENT CONTAINING LEFT PARENTHESIS CAUSES SUBSEQUENT SPACES AND PERIODS TO BE REMOVED	04 R	Jun 78
RUN (NH) COMMAND MAY GIVE AN ERROR MESSAGE	05 M	Jul 78
TERMINAL MAY HANG	06 M	Jul 78
DATA FILES	07 M	Jul 78
SAVE DEV: AND REPLACE DEV:	08 M	Jul 78
SINGLE PRECISION HANG AND NUMERIC CONVERSION PROBLEM (PATCH F)	09 M	Aug 78
CONVERSION PROGRAM	10 M	Sep 78
OVERLAYING WHILE IN A SUBROUTINE	11 R	Nov 78
OPERATION OF CTRLC, AND RCTRLC AND SYS (6) FUNCTIONS AND THE CTRL/C COMMAND	12 N	Nov 78
BASIC-11/RT-11 V2 CONVERSION PROGRAM PATCH 1	13 M	Feb 79
OPERATION OF OLD, RUN, CHAIN AND OVERLAY WHEN THE SPECIFIED FILE IS NOT FOUND	14 N	Feb 79
CREATING AND ACCESSING VIRTUAL ARRAY FILES	15 N	Feb 79
REPUBLICATION OF PATCHES	16 N	Feb 79
PRINT USING - PATCH A	17 M	Feb 79
RESEQ - PATCH B	18 M	Feb 79
EDITING A DIM #n STATEMENT - PATCH C	19 M	Feb 79
DOUBLE PRECISION HANG - PATCH D	20 M	Feb 79
SAVE dev: AND REPLACE dev: - PATCH E	21 M	Feb 79
SINGLE PRECISION HANG AND NUMERIC CONVERSION PROBLEM - PATCH F	22 M	Feb 79
SAVE .XXX & UNSAVE .XXX - PATCH G	23 M	Feb 79

Component	Sequence	Mon/Yr
NEW - PATCH H	24 M	Feb 79
STORAGE OF THE NULL CHARACTER IN STRING VARIABLES AND VIRTUAL STRING ARRAYS	25 N	Feb 79
USE OF COMPILE COMMAND	26 N	Feb 79
RESEQ - PATCH I	27 M	Mar 79
LISTNH /OLD - PATCH J	28 M	Mar 79
SYS(1) - PATCH K	29 M	Mar 79
CALL - PATCH L	30 M	Mar 79
DOUBLE PRECISION INTEGER VARIABLES - PATCH M	31 M	May 79
FILESIZE 0 - PATCH N	32 M	May 79
INTEGERS IN DOUBLE PRECISION BASIC-11	33 M	Jul 79
REM STATEMENTS ON MULTI-STATEMENT LINES - PATCH O	34 M	Jul 79
STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES	35 N	Aug 79
MAXIMUM ARRAY SUBSCRIPT SIZE	36 N	Aug 79
BASIC/RT-11 EXTENSIONS V1		
"IPK" SUBROUTINE	01 M	Aug 77
SAMPLING A/D CHANNEL NO. 15	02 R	Aug 77
SAMPLING AR11	03 M	Sep 77
"CLRD" AND "PUTD" ROUTINES	04 M	Nov 77
"SETR" AND "WAIT" COMBINATION MAY FAIL	05	Apr 78
BASIC/RT-11 EXTENSION BUILD PROCEDURE RESTRICTION	06 R	Mar 79
CTS-300 V5		
DECFORM		
TWO PROBLEMS WITH FOCOMP	01 M	May 79
DIBOL		
TWO PROBLEMS: FILE CORRUPTION POSSIBILITY AND REPETITIVE I/O ERRORS	01 M	Apr 79
OPENING NON-STANDARD HANDLERS	02 M	Apr 79
ANOTHER FILE CORRUPTION POSSIBILITY	03 M	Apr 79
TWO PROBLEMS: OPENING 0 LENGTH FILE IN SUD AND OPENING LP IN I MODE	04 M	Jun 79
LINE PRINTER PROBLEM AND PROBLEM WITH LARGE ISAM FILE	05 M	Jun 79
I/O ERRORS AND PROBLEM WITH FMAC SUBROUTINE	06 M	Jun 79
ISAM FILE CORRUPTION	07 M	Jun 79
SHUFFLE CAUSES TRAP TO 4	08 M	Jul 79
MISLEADING ERROR MESSAGES	09 M	Aug 79
ERRONEOUS I/O ERROR	10 M	Aug 79
REDUCE		
HOW TO REDUCE PAINLESSLY	01 N	Aug 79
SORTM		
MERGE DOES NOT ACCEPT EMPTY FILES	01 M	Apr 79
CTS-300 RDCP (2780/3780), V1.0		
SENDING OF TRANSPARENT DATA AND TRANSLATION OF DATA AFTER SENDING A TRANSPARENT FILE	01 M	Jul 79
SEND A TRANSPARENT FILE AFTER RECEIVING AN ASCII DATA FILE	02 M	Jul 79
AN ACK IS RECEIVED WHEN ENQ HAS ALREADY BEEN SENT	03 M	Jul 79
ATTEMPT TO LOAD LPX.SYS BEFORE CMX.SYS UNDER XM MONITOR	04 M	Aug 79
DECnet-RT V1		
DAP		
DAP ROUTINES DO NOT ARBITRATE DAP SEGMENT SIZE PROPERLY	07 M	Jan 79
NOTES ON CHANGES TO DAP INTERFACE	09 N	Feb 79
CORRECT BUFFER POINTER ERROR	16.11 M	May 79
DDCMP		
DDCMP LINE COUNTERS OVERFLOW TO ZERO	01 O	Jul 78
DMC		
DMC LINE COUNTERS OVERFLOW TO ZERO	01 O	Jul 78
DOCUMENTATION		
USER'S GUIDE DOCUMENTATION ERRORS	2.1 N	Aug 79

Component	Sequence	Mon/Yr
FAL		
CORRECT FAL PROCESSING OF END OF STREAM MESSAGE	01 M	Jan 79
FAL INCORRECTLY ALLOCATES DISC SPACE FOR FILES	02 M	Feb 79
FAL INCORRECTLY HANDLES REMOTE FILE REQUESTS	04 M	Feb 79
TIMING DEPENDENCY IN RT TO RSTS FILE TRANSFERS	17.5 M	Jul 79
MRS FIELD NOT DEFAULTED PROPERLY	17.6 M	Jul 79
FORTRAN INTERFACE		
DIFFERENCES IN RT AND RSX FORTRAN INTERFACE IMPLEMENTATIONS	01 N	Jul 78
USE OF THREADED AND INLINE FORTRAN COMPILER OPTIONS	04 R	Jan 79
FORTRAN REMOTE OPEN FOR WRITE MODIFIES FILE ATTRIBUTES	05 N	Jan 79
MODEM CONTROL		
SUPPORT OF ASYNCHRONOUS HALF DUPLEX MODEMS	01 R	Jul 78
NFARS		
DAP ROUTINES CHANGE MODE DURING FILE TRANSFER	02 M	Feb 79
CHECK FOR BLOCK MODE TRANSFER	03 M	Feb 79
DAP DEFAULTS DO NOT ALLOW RECORDS TO SPAN BLOCKS	06 O	Jan 79
ASCII FILE ACCESS TO VAX/RSX SYSTEMS	08 M	Feb 79
INVALID FILE TYPE SENT TO VAX IN ASCII TRANSFER	10 M	Mar 79
NSP		
PROTOCOL VIOLATION IN NODE INITIALIZATION	01 M	Jan 79
NFT		
NFT ASCII FILE TRANSFER TO VAX/RSX SYSTEMS	03 M	Feb 79
LOGICAL BLOCK NUMBERS NOW START AT ONE	17.5 M	May 79

FEP-11, FORTRAN ENHANCEMENT PACKAGE
ALSO PERTAINS TO: RT-11/FORTRAN UPGRADE PACKAGE FOR MINC

FEP-11 INITIAL PROBLEMS, SOLUTIONS AND HINTS	01 M	May 79

FMS-11 V1

CONSOLE TERMINAL SPECIAL MODE BIT CLEARED	01 M	Jun 79
INCORRECT MCDEMO FILE TYPES	02 O	Jun 79
TSKINI INPUT BUFFER TOO SMALL	03 M	Jun 79
ARTS ERROR MESSAGES LACK '?'	04 M	Jun 79
HANDLER FETCH CORRUPTS FROM FILE ID	05 M	Jul 79
ZERO-FILLED FIELD VALIDATION PROBLEM	06 M	Jul 79
FILED VIDEO ATTRIBUTES PROBLEM	07 M	Jul 79
FRED ERROR MESSAGES LACK'?'	08 M	Jul 79
ERROR IN SCROLL FORWARD/BACKWARD CODE	09 M	Jul 79
ERROR IN EXIT SCROLLED AREA FORWARD CODE	10 M	Jul 79

FORTRAN GRAPHICS PACKAGE, V1.1

DECGRAPHIC		
NMBR SUBROUTINE IN DECgraphic	01 R	JAN 79

FORTRAN/RT-11 EXTENSIONS V1

RUNNING PROGRAM WITH "SETR"	01 M	Oct 78
IBEF NOT PROPERLY DECREMENTED	02 R	Oct 78
LPS DEVICE CONFLICT CAUSED BY CALL SETR AFTER CALL RTS	03 R	Oct 78
IADC AFTER RTS DOES NOT WORK	04 M	Oct 78
SUBROUTINE NAMING CONFLICT	05 N	Oct 78
PLOT55 DESCRIPTION	06 N	Oct 78
ILLEGAL MEMORY REFERENCE ERROR	07 M	Oct 78
uDEVICE CONFLICT ERROR	08 R	Oct 78
TWO PROBLEMS WITH THE RT-11/FORTRAN GRAPHICS EXTENSIONS	09 M	Oct 78

Component	Sequence	Mon/Yr
FORTRAN IV/RT-11 V2		
COMPILER		
DISPOSE = 'KEEP' OPTION	01 R	Jan 79
CRASH DUMPS	02 N	Jan 79
SYNTAX ERRORS IN SOURCE PROGRAM MAY CAUSE COMPILER TO ABORT	03 M	Jan 79
SIMRT	04 M	Jan 79
SIMRT CONTINUED	05 M	Jan 79
KNOWN FORTRAN IV V2 BUGS	06 N	Jan 79
USE OF THE FIND STATEMENT	07 M	Jan 79
RAISING COMPLEX NUMBERS	08 M	Jan 79
EXTRA CHARACTERS MAY RESULT IN COMPILER TRAPPING	09 M	Jan 79
TRANSMITTING ASCII DATA	10 R	Jan 79
IN-LINE CODE	11 N	Jan 79
ERRORS OCCUR WITH NO DO LOOP	12 M	Jan 79
FORTRAN "ACCEPT" STATEMENT	13 R	Jan 79
FORTRAN IV/RT-11 V2.1		
FORTRAN IV V2.1 MAINTENANCE RELEASE	01 N	Dec 78
PATCH 1	02 M	Feb 79
PATCH 2	03 M	Feb 79
PATCH 3	04 M	Feb 79
PATCH 4	05 M	AUG 79
CARRIAGE CONTROL OPTION - PATCH 5	06 M	May 79
OPEN FAILURE WITH TYPE='OLD' - PATCH 6	07 M	Aug 79
FORTRAN LIBRARY FUNCTION ERRTST - PATCH 7	08 M	Aug 79
REGISTER ALLOCATION - PATCH 8	09 M	Aug 79
SMALLER EXECUTION-TIME PROGRAMS	10 N	Jun 79
FORTRAN OTS - PATCH 9	11 M	Jun 79
I/O FROM A FORTRAN COMPLETION ROUTINE - PATCH 10	12 M	Aug 79
FORTRAN FAILS TO COMPILE DO-LOOPS - PATCH 11	13 M	Aug 79
CALL CLOSE (FORTRAN LIBRARY SUBROUTINE) - PATCH 12	14 M	Aug 79
UNFORMATTED BYTE I/O - PATCH 13	15 F	Aug 79
LIST DIRECTED INPUT ERRORS - PATCH 14	16 M	Aug 79
DISP='DELETE' OPTION - PATCH 15	17 M	Aug 79
FORMATTED RECORD OUTPUT - PATCH 16	18 M	Aug 79
COMMON SUBEXPRESSION OPTIMIZATION - PATCH 17	19 M	Aug 79
CALL ASSIGN CARRIAGE CONTROL - PATCH 18	20 M	Aug 79
NON-PLAS VIRTUAL ARRAY INITIALIZATION - PATCH 19	21 M	Aug 79
BYTE COMPARISON AND COMMON SUBEXPRESSION OPTIMIZATION - PATCH 20	22 M	Aug 79
DIRECT ACCESS READ - PATCH 21	23 M	Aug 79
COMPLEX VARIABLE TO CONSTANT COMPARISON - PATCH 22	24 M	Aug 79
FOCAL/RT-11 V1B		
FOR COMMAND WITHOUT AN ARGUMENT	01 M	Oct 75
OPERATE COMMAND CAUSES ERROR	04 M	Aug 76
FCLK ROUTINE GIVES INCORRECT TIME	05 O	Aug 76
"LIBRARY ASK" COMMAND	06 O	Feb 77
"/Z" SWITCH	07 M	Aug 77
@START NOT WORKING WHEN DOWN-LINE LOADING	08 M	Mar 78
LIBRARIES FROM FOCAL SOURCE DISK MUST BE REFORMATTED	09 N	Aug 78
CLOCK PROBLEM FOR PAPER TAPE (STAND-ALONE) FOCAL USERS	10 M	Nov 78
FORTRAN/RT-11 EXTENSIONS V1		
RUNNING PROGRAM WITH "SETR"	01 M	Oct 78
IBEF NOT PROPERLY DECREMENTED	02 R	Oct 78
LPS DEVICE CONFLICT CAUSED BY CALL SETR AFTER CALL RTS	03 R	Oct 78
IADC AFTER RTS DOES NOT WORK	04 M	Oct 78
SUBROUTINE NAMING CONFLICT	05 N	Oct 78
PLOT55 DESCRIPTION	06 N	Oct 78
ILLEGAL MEMORY REFERENCE ERROR	07 M	Oct 78
DEVICE CONFLICT ERROR	08 R	Oct 78
TWO PROBLEMS WITH THE RT-11/FORTRAN GRAPHICS EXTENSIONS	09 M	Oct 78

Component	Sequence	Mon/Yr
FORTRAN/RT-11 EXTENSIONS V1B		
FORTRAN CRASHES AFTER RUNNING PROGRAM WITH "SETR"	01 M	Oct 78
TWO PROBLEMS WITH THE RT-11/FORTRAN GRAPHICS EXTENSIONS	02 M	Oct 78
NEGATIVE INTENSITY	03 N	Nov 78
PROGRAM TERMINATION ERROR USING RT-11 F/B	04 R	Apr 79
FORTRAN/RT-11 EXTENSIONS V2.1		
FORTRAN CRASHES AFTER RUNNING PROGRAM WITH "SETR"	01 M	Mar 79
TWO PROBLEMS WITH THE RT-11/FORTRAN GRAPHICS EXTENSIONS	02 M	Mar 79
NEGATIVE INTENSITY	03 N	Mar 79
GAMMA-11 F/B V2		
DATA ANALYSIS PROGRAM	01 M	Feb 79
STUDY PROGRAM DISPLAYS TOO MANY INDEX LINES PER PAGE	02 M	Feb 77
BASIC AND FOCAL	03 M	Feb 77
BACKGROUND PROGRAM CAN HANG THE FOREGROUND TERMINAL	04 M	Feb 77
CNTL/C UNDER SINGLE JOB MONITOR	05 M	Feb 77
CROSSHAIRS FAIL TO APPEAR IN SLICE	06 M	Feb 77
UNDOCUMENTED PROGRAMS	07 N	Mar 77
FORTRAN SUPPORT INCORRECTLY CONVERTS DATA AND TIME OF INQUISITION	08 M	May 77
"RS" COMMAND IS INCORRECTLY	09 N	Jun 77
GAMMA-11 F/B V2C		
GATED LIST MODE IMAGES	01 O	Sep 78
TU16 SUPPORT	02 M	Sep 78
PROBLEMS WITH PLAYBACK BUFFER COMMENTS AND FLOOD CORRECTIONS	03 M	Oct 78
STATIC FOREGROUND ACQUISITION FAILS ON RK06 OR RL01 SYSTEMS	04 M	Oct 78
DYNAMIC CURVE CALCULATIONS MAY FAIL	05 M	Dec 79
RK06, 7 AND RL01 FOREGROUND ACQUISITIONS PROBLEMS	06 M	Dec 78
PROBLEMS WITH FLOOD CORRECTIONS	07 M	Dec 78
PROBLEMS WITH REGION OF INTEREST	08 M	Dec 78
KW11-P REAL-TIME CLOCK INCORRECTLY INITIALIZED	09 M	Dec 78
GAMMA-11 V2C NCV11 REAL-TIME CLOCK CAN BE DISABLED	10 M	Dec 78
KW11-P REAL-TIME CLOCK RUNS TOO FAST DURING GSA STUDIES	11 M	Dec 78
BUILDING AN RL01 GAMMA-11 V2C SYSTEM	12 M	Dec 78
PREDEFINED GATED LIST MODE STUDIES	13 M	Dec 78
GATED LIST MODE DATA ACQUISITION SET-UP	14 M	Dec 78
PROBLEMS WITH MAGTAPE DISTRIBUTION	15 N	Dec 78
SUBROUTINE 'GMXG' GENERATES ILLEGAL ADDRESS MESSAGE	16 F	Jul 79
FGAMMA/BGAMMA RACE CONDITION	17 M	Feb 79
DELAYED START LIST MODE STUDIES	18 M	Feb 79
FORMATTING GATED LIST MODE STUDIES	19 M	Feb 79
SLICE PROBLEMS	20 M	Feb 79
DOUBLE INTERPOLATION OF 64 X 64 MATRIX DATA	21 M	Feb 79
GAMMA-11 AND RT-11 DATE ROLLOVER	22 M	Feb 79
PROBLEMS WITH PATIENT MONITOR AND GSA ADMIN BLOCKS	23 M	Feb 79
FOREGROUND GATED LIST MODE STUDIES FAIL	24 M	Feb 79
NCV11 JOYSTICK AND LIST MODE PROBLEMS	25 M	May 79
SYSTEM SUMMARY FOR RK07 DISKS	26 O	May 79
MORE PROBLEMS WITH FLOOD CORRECTION	27 M	May 79
TWO MINOR PROBLEMS WITH PLAYBACK BUFFERS	28 M	May 79
TRANSFER STUDY CAN CORRUPT A DISK DIRECTORY	29 M	May 79
FOUR FRAME MINIMUM FOR GSA STUDIES	30 M	May 79
GAMMA-11/BASIC PATCHES	31 M	May 79
CONTINUE ANALYSIS CA) OCCASIONALLY FAILS	32 M	May 79
ASCII STRING VARIABLE TABLE (FORTRAN AND BASIC) -- SUBROUTINE GPAR AND GPAW --	33 M	Jul 79
GAMMA-11 SYSTEMS WITH RK07 AS A DEVICE	34 M	Aug 79
LABORATORY APPLICATIONS-11 V3		
A NEW MODULE TO ENHANCE DATA FLOW WITHIN LA-11	01 N	Oct 76
HISTO.MAC		
ACQUIRING AND PROCESSING HISTOGRAM DATA	01 M	Sep 76

Component	Sequence	Mon/Yr
LABMAC.SML		
ERRONEOUS MACRO	01 M	Sep 77
INCLUDING LABMAC.SML IN SYSMAC.SML	02 M	Mar 79
PEAK.MAC		
WIDE PEAKS	01 M	Mar 76
PEAK PROBLEMS AND CORRECTIONS	02 M	Jul 76
ARITHMETIC CORRECTION FOR PEAK AREA	03 M	Dec 76
MISSING PATCH IN RELEASE NOTES	04 M	Oct 77
SPARTA		
LPS AND AR-11 VECTOR AND STATUS REGISTER	01 N	Dec 75
USING SPARTA AND FLOATING POINT BUFFERS	02 N	Feb 76
AR-11 TIMING PROBLEMS WITH ADSAM AND SPARTA	03 O	Feb 76
FFT SCALING CORRECTION	04 M	Feb 76
SCALE FACTOR CORRECTION FOR SPARTA COMMANDS FAC AND FCC	05 M	Mar 76
DATA DISPLAYS USING LA-11	06 N	Mar 76
DATA PREPARATION FOR SPARTA COMMANDS FAC AND FCC	07 N	Apr 76
SPARTA CORRECTIONS FOR POINT-PLOT DISPLAY	08 M	Apr 76
ADDING COMMANDS TO SPARTA	09 M	May 76
CORRECTION FOR THE DPV COMMAND WITH POINT PLOT DISPLAY	10 M	Jun 76
GENERAL SUBROUTINE MODULE FOR EAE	11 O	Jun 76
INCORRECT PHASE ANGLE CALCULATION	12 M	Oct 76
"MOU" AND "MIN" COMMANDS CAN BE READ OUT AND IN CORRECTLY	13 N	Jan 77
MULTIPLE SYNCH PULSES	14 M	Jan 77
AUTO AND CROSS CORRELATION	15 M	Jan 77
ALLOCATING MORE THAN 16K BUFFERS IN SPARTA	16 M	Feb 77
A/D SAMPLING: FAST MODE	17 M	Jul 77
A/D SAMPLING: FAST MODE EXIT	19 M	Mar 78
SCALE FACTOR PRINT FOR THE FFT	20 M	Jan 79
SWEEP.MAC		
SWEEP SAMPLING: FAST MODE	01 M	Aug 77
THRU		
HOW TO START DATA ACQUISITION WHEN CSTART EQUALS ZERO	01 N	Jun 76
MULTICHANNEL SINGLE RATE SCHMIT TRIGGER SWITCH BOUNCE	02 M	Dec 76
CONTINUOUS SAMPLING: CONDITIONAL ASSEMBLY ERRORS	03 M	Jul 77
CONTINUOUS SAMPLING: DMA WITH DUAL SAMPLE + HOLD	04 M	Jul 77
DOCUMENTATION CORRECTIONS	05 M	Nov 77

LSP-11 V1

PATCH NO. 1 - GENERAL CORRECTIONS NO. 1	01 M	Jun 79
PATCH NO. 2 - PEAK CORRECTION NO. 1	02 M	Jun 79
PATCH NO. 3 - PEAK CORRECTION NO. 2	03 M	Jun 79

LV11/RT-11 PLOTTING PACKAGE V2

SUBROUTINE PLOT DOES NOT CORRECTLY REPRODUCT VT11 PICTURE	01 M	Apr 78

MSB-11 V1.0

MSB-11 SOFTWARE ON THE PDP-11/03	01 M	Jul 79

MU BASIC/RT-11 V1

BUILDING MU BASIC/RT-11 UNDER RT-11 V2C	01	Feb 76
REMOTE TERMINAL SUPPORT ON MODEMS	02	May 76
OVERLAY... LINE WORKS INCORRECTLY	03	May 76
USING IMMEDIATE MODE "GOSUBs"	04	Dec 76
CLOCK LOSES TIME ON RT-11 WHEN RUNNING MU BASIC	05	Jul 77
REM STATEMENTS	06	Feb 78
ADDITIONAL FILES ON RELEASE KIT (MUB*.*)	07 N	May 78
MU BASIC/RT-11 SYSTEM INSTALLATION GUIDE		
REPLACEMENT PAGES	01	Jan 77
REPLACEMENT PAGES	02 N	Jan 78
REPLACEMENT PAGES	03 N	Jan 78

Component	Sequence	Mon/Yr
MU BASIC-11/RT-11 V2		
MU BASIC-11/RT-11 V2 CONVERSION PROGRAM	01 R	Nov 78
OPERATION OF CTRL/C, RCTRLC AND SYS (6) FUNCTIONS AND THE CTRL/C COMMAND	02 N	Nov 78
MEMORY REQUIREMENTS OF OPTIONAL FUNCTIONS ETC.	03 O	Nov 78
MU BASIC-11/RT-11 V2 RELEASE NOTES AND INSTALLATION GUIDE CHANGES	04 N	Dec 78
ORDER OF COMMON STATEMENTS AT START OF MUCNFG.BOO, MUCNF1.BOO, MUCNF2.BOO	05 M	Dec 78
OPERATION OF OLD, RUN, CHAIN AND OVERLAY WHEN THE SPECIFIED FILE IS NOT FOUND	06 N	Feb 79
CREATING AND ACCESSING VIRTUAL ARRAY FILES	07 N	Feb 79
STORAGE OF THE NULL CHARACTER IN STRING VARIABLES AND VIRTUAL STRING ARRAYS	08 N	Feb 79
USE OF COMPILE COMMAND	09 N	Feb 79
MU BASIC-11/RT-11 V2 CONFIGURATION PROGRAM PATCH 1	10 O	Feb 79
CHAINING WITH COMMON -PATCH A	11 M	Feb 79
VIRTUAL FILE I/O - PATCH B	12 M	Feb 79
SYS (1,n) FUNCTION - PATCH C	13 M	Feb 79
RESEQ - PATCH D	14 M	Feb 79
VALUES IN PATCHES A, B, C	15 N	Feb 79
LISTNH / OLD - PATCH E	16 M	Mar 79
CALL - PATCH F	17 M	Mar 79
MU BASIC-11 DEVICE INDEPENDENCE FOR INIT.BOO - SPECIIAL PATCH YY1	18 M	May 79
DOUBLE PRECISION INTEGER VARIABLES - PATCH G	19 M	May 79
INPUT #/PRINT # - PATCH H	20 M	May 79
OLD OF A ZERO BLOCK FILE - PATCH I	21 M	May 79
ADDITION TO PATCH B - PATCH J	22 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 1	23 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 2	24 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 3	25 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 4a	26 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 4b	27 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 4c	28 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 5	29 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 6	30 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 7	31 M	May 79
MU BASIC-11/RT-11 V2 PERFORMANCE IMPROVEMENT PATCH NO. 8	32 M	May 79
DEVICE MNEMONIC PROBLEM - PATCH K	33 M	Jul 79
CLOSE - PATCH L	34 M	Jul 79
REM STATEMENTS ON MULTI-STATEMENT LINES	35 M	Jul 79
DEASSIGNING A TERMINAL - PATCH N	36 M	Jul 79
OVERLAYING THE ERROR MESSAGE MODULE - SPECIAL PATCH WW1	37 M	Jul 79
UNEQUAL USER PARTITION SIZE ALLOCATION - SPECIAL PATCH XX1	38 M	Jul 79
HOW TO CHANGE INIT.BOO's DEVICE AFTER INSTALLING SPECIAL PATCH YY1	39 M	Jul 79
INTEGERS IN DOUBLE PRECISION MU BASIC-11	40 M	Jul 79
STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES	41 N	Aug 79
SIZING MU BASIC-11	42 N	Aug 79
ERROR IN TABLE 4-1 OF THE USER'S GUIDE	43 N	Aug 79
RESTRICTION OF USR RESIDENCY WHEN RUNNING IN FOREGROUND	44 N	Aug 79
NOTES ON PERFORMANCE PATCHES NO. 4a, NO. 4b, NO. 4c	45 N	Aug 79
MAXIMUM ARRAY SUBSCRIPT SIZE	46 N	Aug 79
PDL/RT-11 V1B		
CLARIFICATION OF SEARCH FAILURE IN SUBROUTINE FIND	01 N	Jul 78
FIND SUBROUTINE	02 R	Jul 78
PATCHES TO PDL	03 M	Jul 78
SUBROUTINE QKGT	04 M	Jul 78
PDL SUBROUTINE 'RDAA'	05 M	Sep 78
PDL PEAK ALGORITHM WILL NOT RECOGNIZE VALID PEAKS	06 M	Sep 78
PEAK-11 V1		
"MREPRT" AND "REPRT" GET CONFUSED	01 M	Aug 78

Component	Sequence	Mon/Yr
REMOTE/RT-11 V1		
SCHEDULER DOES NOT PROPERLY SET PROCESSOR PRIORITY	01 M	May 76
NOEDIT- 0 HALTS	02 M	May 76
NUSERS=1 STAYS IN A FILE MESSAGE LOOP	03 M	May 76
INCORRECT SWAP AREA ALLOCATION FOR FOUR OR MORE USERS	04 M	May 76
REBOOT FROM SATELLITE DURING EDIT HANGS HOST	05 M	Jun 76
HARD ERROR ON LOOKUP IS FATAL	06 M	Jun 76
SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL	07 M	Jun 76
ONE SECOND TIMER FOR LINE TIMEOUTS IS SET INCORRECTLY	08 M	Aug 76
LINE FEEDS MAY CAUSE SYSTEM ERRORS--ASSEMBLY ERROR WITH DIAL AND NODDC	09 M	Aug 76
PROPER GENERATION OF REMOTE IS DEPENDENT ON MODULE ORDER	10 M	Aug 76
ASCII CODES 173 AND 174 DO NOT PRINT	11 M	Aug 76
IMPROPER FILLER HANDLING FOR VT05	12 O	Aug 76
SYSTEM CRASHES IF RUN IN FOREGROUND WITHOUT /N	13 O	Aug 76
"UNSAVE" COMMAND CAUSES SYSTEM ERRORS	14 M	Dec 76
FLET WILL REMOVE MORE THAN ONE USER FROM THE WAIT QUEUE	15 M	Dec 76
STACK FOR USER THREE IMPROPERLY SET	16 O	Dec 76
SECONDARY MODE LOADS DO NOT OPERATE PROPERLY	17 M	Jan 77
@START COMMAND GIVEN ON TERMINAL WITHOUT SATELLITE CAUSES CRASH	18 O	Jan 77
"RTSIM" DOES NOT SUPPORT 50 Hz LINE CLOCK	19 O	Jan 77
CHANNEL ACTIVE ERROR	20 M	Mar 77
THREE WORDS LOST ON DOWNLINE LOAD	21 M	Mar 77
CSISPC NOT PROPERLY SIMULATED	22 M	May 77
EXCEEDING CHARACTERS PER LINE LIMIT	23 M	Oct 77
UNASSIGNED	24	XXX XX
@RE IN THE SATELLITE DOES NOT WORK	25 R	Mar 78
"HANG" CONDITIONS	26 R	Apr 78
UANSSIGNED	27	XXX XX
USING KG-11 CRC CALCULATOR	28 M	Aug 78
PASTE CAUSES LINE DUPLICATION	29 M	Aug 78
"DAISY CHAIN" ARRANGEMENT IN RTSIM.MAC	30 M	Aug 78
OPTIONAL RMON IS OMITTED FROM RTS1M BY DEFINING NORMON=0	31 M	Oct 78
DL-11 ERROR AND CRC ERROR IN HOST	32 M	Oct 78
RT-11 V3		
DOCUMENTATION		
TYPOGRAPHICAL ERRORS	01 N	Mar 78
ERROR IN FOREGROUND/BACKGROUND DEMONSTRATION	02 M	Aug 78
THE /LIST OPTION FOR THE DIBOL, FORTRAN, AND MACRO KEYBOARD MONITOR COMMANDS	03 M	Nov 78
EDIT		
EDIT DOES NOT OPERATE CORRECTLY UNDER XM MONITOR	01 M	Mar 78
MACRO		
.NARG FAILS WHEN AUTOMATIC LABEL GENERATION IS USED	01 M	Apr 78
MISCELLANEOUS		
GETSTR AND PUTSTR ROUTINES FOR IN-LINE CODE	01 M	Jun 78
ERROR IN THE CONCAT ROUTINE	02 M	Jun 78
ERROR IN MTATCH ROUTINE	03 M	Nov 78
ODD RING BUFFER SIZES CAUSE ASSEMBLY ERRORS	04 R	Jun 79
MONITOR		
INCORRECT IDENTIFIER IN .TWAIT REQUEST CAUSES PROBLEMS	01 M	Mar 78
.CHAIN, .EXIT FROM VIRTUAL JOB; USR MOVING INTO PAR1 AREA	02 M	Apr 78
PATCH TO INTERRUPT EXIT ROUTINE	03 M	Apr 78
IMPROPER HANDLING OF THE KW11-P CLOCK	04 M	May 78
SPECIFYING 50-CYCLE CLOCK SUPPORT DURING SYSGEN OPERATIONS	05 M	Jun 78
EDITORS AND V3B MONITORS	06 M	Jun 78
TYPING NON-ASCII FILES TO CONSOLE AFTER ISSUING A GTON HANGS THE SYSTEM	07 M	Jun 78
LINK/FRUN FAILS WHEN PROGRAM IS OVERLAYED AND USES LIBRARIES	08 M	Jul 78
MULTITERMINAL CORRECTIONS	09 M	Aug 78
PATCH TO XM ADDRESS CHECKING	10 M	Aug 78
FIXES FOR TWO FB/XM PROBLEMS	11 M	Aug 78
TERMINATING CONSOLE OUTPUT	12 M	Aug 78
ISSUING SEEKS TO DX HANDLER IN XM CAUSES RANDOM SYSTEM FAILURES	13 M	Oct 78
CERTAIN EXTENDED MEMORY REQUESTS CANNOT BE ISSUED FROM BOTH MAINLINE CODE AND COMPLETION ROUTINES	14 M	Oct 78

Component	Sequence	Mon/Yr
THE "RUN" AND "GET" MONITOR COMMANDS DO NOT CORRECTLY LOAD THE PORTION OF A PROGRAM THAT OVERLAYS KMON	15 M	Oct 78
DX SJ MONITOR BOOTSTRAP CORRECTIONS	16 O	Oct 78
TYPING CTRL/O TO THE CONSOLE TERMINAL SOMETIMES CRASHES	17 M	Nov 78
LINK CAUSES ODD MONITOR ADDRESS TRAP	18 M	Nov 78
CHAINING FROM A VIRTUAL JOB AND RELATED PROBLEMS	19 M	Dec 78
DIRECTORY CORRUPTION	20 M	Dec 78
FIXES FOR FB/XM PROBLEM IN V03.02	21 M	Apr 79
CORRECTION TO "DIRECTORY CORRUPTION" PATCH	22 M	May 79
FLOPPY SYSGEN WITH KW11-P CLOCK	23 M	May 79
INPUT FILE LOST WHEN USING CSIGEN	24 M	Jun 79
SOURCES		
UNRESOLVED DIFFERENCES IN DEMOX1.MAC	01 M	Aug 78
DISTRIBUTED MAGTAPE HANDLER CORRECTIONS	02 M	Sep 78
MAGTAPE XM AND FSM CORRECTIONS	03 M	May 79
SYSTEM HANDLERS		
DM HANDLER CORRECTIONS	01 M	Oct 78
DM SYSTEM HANDLERS CORRECTIONS	02 M	Dec 78
DM HANDLER ERROR HANDLING CORRECTIONS	03 M	Jan 79
DM CTO AND SPFUN 376 CORRECTIONS	04 M	May 79
UTILITIES		
DUP DEFAULT FILE SIZE AND NULL FILE TYPES ARE INCORRECT	01 M	Mar 78
DIR MAY INCORRECTLY LIST DIRECTORIES OF MAGTAPES	02 M	Mar 78
/L OPTION TO PIP MAY CUASE SYSTEM CRASH	03 M	Mar 78
LINK OUTPUT INVALID IF OBJ HAS AN EMPTY GSD RECORD	04 M	Mar 78
PAT GIVES FATAL ERROR IF OBJ HAS AN EMPTY RECORD	05 M	Apr 78
UNASSIGNED	06	XXX XX
EDIT VT11 DISPLAY FUNCTIONS WILL NOT OPERATE UNDER XM MONITOR	07 M/R	Apr 78
TRANSFERS IN INTERCHANGE FORMAT WHEN NO SYSTEM DATE IS GIVEN	08 M	Jun 78
DUP SCAN RATE FOR FLOPPY	09 M	Jun 78
DUP /I AND /W SWITCHES DO NOT WORK PROPERLY	10 M	Jun 78
LINK/FRUN FAILS WHEN PROGRAM IS OVERLAYED AND USES LIBRARIES	11 M	Jul 78
DUP DOES NOT DIFFERENTIATE BETWEEN DELETED .BAD FILES AND PERMANENT ONES	12 M	Jul 78
ERRORS IN FILEX INTERCHANGE FORMAT	13 M	Jul 78
LINK PRODUCES INCORRECT .LDA FILES	14 M	Sep 78
DUP DOES NOT DETECT END OF SEGMENT IF IT IS FIRST ENTRY IN A DIRECTORY SEGMENT DURING A SQUEEZE OPERATION	15 M	Oct 78
LIBR CLEARING OF LOCATION ZERO	16 M	Oct 78
LINK ERROR IN PSECTS MOVED TO ROOT	17 M	Oct 78
PIP ERRONEOUSLY DELETES FILES	18 M	Oct 78
LIBR BLOCK BOUNDARY PROBLEM	19 M	Dec 78
LINK CAN CAUSE TRAP TO 4	20 M	Feb 79
CORRECTIONS TO FILEX	21 M	May 79

RT-11 V3B

Component	Sequence	Mon/Yr
DOCUMENTATION		
ERROR IN FOREGROUND/BACKGROUND DEMONSTRATION	01 M	Aug 78
THE /LIST OPTION FOR THE DIBOL, FORTRAN, AND MACRO KEYBOARD MONITOR COMMANDS	02 M	Nov 78
UPDATE PAGES	03 N	Dec 78
RT-11 SOFTWARE SUPPORT DOCUMENTATION	04 M	Feb 79
SUMMARY OF UPDATES FOR RT-11 V03B DOCUMENTATION	05 M	Feb 79
NEW DEVICE RELEASE DOCUMENTATION, RT-11 V03B	06 N	Jun 79
.FORK AND .SYNCH BLOCK DOCUMENTATION	07 N	Jul 79
CORRECTION OF ERROR RETURNS IN .SYNCH CALL	09 M	Aug 79
EXAMPLE CODE IN .FORK DOCUMENTATION IS INCORRECT	10 N	Aug 79
MISCELLANEOUS		
ERRORS IN THE SYSGEN CONDITIONAL FILE	01 M	Jul 78
ERROS IN MTATCH ROUTINE	02 M	Nov 78
ODD RING BUFFER SIZES CAUSE ASSEMBLY ERRORS	03 R	Jun 79
INCORRECT NULL HANDLER DEVICE IDENTIFIER	04 M	Jun 79
GENERATING A SINGLE JOB MONITOR MAY CAUSE AN UNDEFINED GLOBAL	05 M	Aug 79
MONITOR		
SOURCE PATCHING PROCEDURES FOR V3B	01 M	Aug 78
MULTITERMINAL CORRECTIONS	02 M	Aug 78
SINGLE JOB TIMER SUPPORT CORRECTIONS	03 M	Aug 78

Component	Sequence	Mon/Yr
FIXES FOR TWO FB/XM PROBLEMS IN VP3B	04 M	Aug 78
TERMINATING CONSOLE OUTPUT	05 M	Aug 78
EDITORS AND V03B MONITORS	06 O	Aug 78
SEEK IN RK DRIVER	07 M	Aug 78
RL01 CONTROLLER VECTOR AT 160	08 M	Aug 78
FPU EXCEPTION HANDLING IN XM MONITOR	09 M	Sep 78
TWO EXTENDED MEMORY MONITOR PROBLEMS	10 M	Oct 78
TYPING CTRL/O TO THE CONSOLE TERMINAL SOMETIMES CRASHES RT-11	11 M	Oct 78
DX SJ MONITOR BOOTSTRAP CORRECTIONS	12 O	Oct 78
THE EDIT AND HELP MONITOR COMMANDS FAIL AFTER A VIRTUAL JOB HAS RUN	13 M	Nov 78
DIRECTORY CORRUPTION AND .UNPROTECT CORRECTIONS	14 M	Jan 79
FB AND XM MONITOR CLOCK SUPPORT	15 M	Apr 79
CHANGING CLOCK RATE ON GENERATED MONITORS	16 M	Apr 79
MULTI-TERMINAL CORRECTIONS TO DECREASE INTERRUPT LATENCY	17 M	Apr 79
FIXES FOR FB/XM PROBLEM IN V03B.00	18 M	Apr 79
FLOPPY SYSGEN WITH KW11-P CLOCK	19 M	May 79
DISTRIBUTED FB MONITOR CLOCK SUPPORT	20 M	May 79
OPTIONAL PATCH TO IMPROVE PERFORMANCE ON PDP-11/03 SYSTEMS	21 O	May 79
DISTRIBUTED PD AND DD FB MONITORS CLOCK SUPPORT	22 M	May 79
OPTIONAL PATCH TO IMPROVE PERFORMANCE ON PDP-11/03 AND PDT SYSTEMS FOR DD AND PD FB MONITORS	23 O	May 79
INPUT FILE LOST WHEN USING CSIGEN	24 M	Jun 79
NON-STANDARD VECTOR ADDRESSES FOR RX01 AND RX02 SECOND CONTROLLER	25 M	Aug 79
ABORT DURING COMPLETION CAUSES SYSTEM FAILURES	26 M	Aug 79
SOURCES		
UNRESOLVED DIFFERENCES IN DEMOX1.MAC	01 M	Jul 78
ISSUING SEEKS TO DX HANDLER IN XM CAUSES RANDOM SYSTEM FAILURES	02 M	Sep 78
DISTRIBUTED MAGTAPE HANDLER CORRECTIONS	03 M	Sep 78
DY HANDLER DOUBLE DENSITY ONLY SUPPORT	04 M	Apr 79
DL QUEUE ELEMENT AND XM ZERO FILL CORRECTIONS	05 M	Apr 79
MAGTAPE XM AND FSM CORRECTIONS	06 M	May 79
DL HANDLER SEEK AND UNIT CORRECTIONS	07 M	Aug 79
SYSTEM HANDLERS		
RL01 HANDLER CORRECTIONS	01 M	Sep 78
ISSUING A SEEK TO THE DY HANDLER CAUSES THE SYSTEM TO CRASH	02 M	Oct 78
DM HANDLER CORRECTIONS	03 M	Oct 78
DM SYSTEM HANDLERS CORRECTIONS	04 M	Dec 78
DY HANDLER SPFUN CORRECTION	05 M	Dec 78
DM HANDLER ERROR HANDLING CORRECTIONS	06 M	Jan 79
RL01 PATCH CLARIFICATION	07 N	Jan 79
DM CTO AND SPFUN 376 CORRECTIONS	08 M	May 79
UTILITIES		
ERRORS IN FILEX INTERCHANGE FORMAT	01 M	Jul 78
LINK PRODUCES INCORRECT .LDA FILES	02 M	Sep 78
LIBR CLEARING OF LOCATION ZERO	03 M	Oct 78
LINK ERROR IN PSECTS MOVED TO ROOT	04 M	Oct 78
DUP DOES NOT DETECT END OF SEGMENT	05 M	Oct 78
COPY/DEVICE FAILS ON DISK TO MAGTAPE	06 M	Oct 78
LINK CAUSES MONITOR ODD ADDRESS TRAP	07 M	Nov 78
LIBR BLOCK BOUNDARY PROBLEM	08 M	Jan 79
EDIT ESCAPE CODE CORRECTION	09 O	Dec 78
ERROR IN ODT	10 M	Feb 79
ERROR IN EDIT	11 M	Feb 79
LINK CAN CAUSE TRAP TO 4	12 M	Feb 79
CORRECTIONS AND ADDITIONS TO FILEX	13 M	May 79
RESORC DISPLAYS STATUS OF FIRST 14 TERMINALS	15 M	Jun 79
LIBR /U SWITCH PROBLEM	16 M	Aug 79
IMPORTANT RESTRICTIONS FOR SQUEEZE OPERATIONS	17 M	Aug 79

RT-11/2780 V2

CORRECTIONS TO 2780 PACKAGE	01	Sep 77
RUNNING 2780 ON RT-11 V3	02	Nov 77
PATCHING THE 2780 IN RT-11 V3	03 M	Jan 79
CHECK FOR ZERO LENGTH RECORD	04 M	Jan 79
RESTRICTION OF THE CONSOLE AS AN INPUT/OUTPUT DEVICE	05 R	Jan 79

Software Product Description

PRODUCT NAME: FORTRAN IV/RT-11, Version 2.1 **SPD 12.10.9**

DESCRIPTION:

FORTRAN IV is an extended FORTRAN implementation based on American National Standard (ANSI) FORTRAN, X3.9 - 1966. It operates under the RT-11 operating system. The PDP-11 FORTRAN IV language includes the following extensions to the ANSI standard:

- General expressions allowed in all meaningful contexts
- Mixed-mode arithmetic
- BYTE data type for character manipulation
- ENCODE, DECODE statements
- PRINT, TYPE, ACCEPT input/output statements
- Direct-access unformatted input/output DEFINE FILE statement
- Comments allowed at end of each source line
- PROGRAM statement
- OPEN and CLOSE file access control statements
- List-directed input/output

Additionally, virtual arrays are supported on systems with memory management directives. Virtual arrays are memory-resident, and require enough main memory to contain all elements of all arrays.

The PDP-11 FORTRAN IV compiler is a fast, one-pass compiler. Compiler options allow program size (threaded code) versus execution speed (in-line code) tradeoffs. FORTRAN IV compiler optimizations include:

- Common subexpression elimination
- Local code tailoring
- Array vectoring
- Optional in-line code generation for integer and logical operations

MACRO-11 assembly language subroutines may be called from FORTRAN IV programs.

Object Time System:

FORTRAN IV includes a set of object modules, called the Object Time System (OTS), that are selectively linked with compiler-produced object modules to produce an executable program.

The RT-11 system provides several special features for FORTRAN IV. FORTRAN programs may be developed under RT-11 and output in absolute binary format for execution on a stand-alone PDP-11 system with minimal peripherals, or for loading into ROM or PROM memory.

Using SYSLIB, the RT-11 FORTRAN system subroutine library, all features of the RT-11 monitor are available to FORTRAN programs. Additionally, SYSLIB provides subroutines which support extensive character string manipulations, where the characters are stored as variable-length strings in BYTE arrays.

MINIMUM HARDWARE REQUIRED:

Any valid RT-11 configuration (32K bytes of memory are required for string support)

EIS hardware is required for virtual arrays.

OPTIONAL HARDWARE:

FORTRAN IV supports all devices supported by the operating system.

FORTRAN IV generated code can be selected to support the following arithmetic hardware options:

KE11-A Extended Arithmetic Element
KE11-B Extended Arithmetic Element
KE11-E Extended Instruction Set
KE11-F Floating Instruction Set
KEV11 Extended Arithmetic Chip

The FORTRAN IV OTS additionally supports the FP11 floating point processor.

PREREQUISITE SOFTWARE:

RT-11 operating system, Version 3B or later

OPTIONAL SOFTWARE:

FORTRAN/RT-11 Extensions
PLOT 11/RT-11
SSP-11, Scientific Subroutine Package
FMS-11 Forms Management System

TRAINING CREDITS:

None

SUPPORT CATEGORY:

B — Software Support will be provided as stated in the Software Support Categories Addendum to this SPD.

ORDERING INFORMATION:

All binary licensed software, including any subsequent updates, is furnished under the licensing provisions of DIGITAL's Standard Terms and Conditions of Sale, which provide in part that the software and any part thereof may be used on only the single CPU on which the software is first installed, and may be copied, in whole or in part (with the proper inclusion of the DIGITAL copyright notice and any DIGITAL proprietary notices on the software) only for use on such CPU. All source licensed software is furnished only under the terms and conditions of a separate Software Program Sources License Agreement between Purchaser and DIGITAL.

Options with no support services are only available after the purchase of one supported license. When a software license is ordered without support services, the category of support applicable to such software is Category C.

A single-use license only option is a license to copy the software previously obtained under license, and use such software in accordance with DIGITAL's Standard Terms and Conditions of Sale. The category of support applicable to such copied software is Category C.

Source and/or listing options are only available after the purchase of at least one supported license and after a source license agreement is in effect.

The following key (C, D, E, G, Q, R, T, Y, Z) represents the distribution media for the product and must be specified at the end of the order number, e.g., QJ813-AD = binaries on 9-track magnetic tape.

C	=	DECtape
D	=	9-Track 800 BPI magtape (NRZI)
E	=	RK05 Disk cartridge
G	=	TU58 DECtape II cartridge
Q	=	RL01 Disk cartridge
R	=	Microfiche
T	=	RK06 Disk cartridge
Y	=	RX01 Floppy diskette
Z	=	No hardware dependency

QJ813 -A— Single-use license, binaries, documentation, support services (media: C, D, E, G, Q, T, Y)

QJ813 -C— Single-use license, binaries, documentation, no support services (media: C, D, E, G, Q, T, Y)

QJ813 -D— Single-use license only, no binaries, no documentation, no support services (media: Z)

Source/Listing Options

QJ813 -E— All sources (media: D, E, Q, T)

QJ813 -F— Listings (media: R)

Upgrade Options

The following option is available as an upgrade kit from MSB/FORTRAN IV for use on the same single CPU on which MSB/FORTRAN IV is licensed. The license previously granted for MSB/FORTRAN IV shall be extended to cover this upgrade.

QJE06 -A— Single-use license, binaries, documentation, support services (media: Y)

Update Options

Users of FORTRAN IV/RT-11, Version 1C or Version 2.0, whose specified Support Category warranty has expired may order the following software update at the then current charge for such update, for use under the existing license. Except where the medium is designated as Z,,the update is distributed in source or binary form on the appropriate medium. A software update where the medium is designated as Z grants the user of FORTRAN IV/RT-11, Version 1C, the right to copy the previously ordered QJ813-H or QJ813-W software update for use on an additional single CPU for which a FORTRAN IV/RT-11 license has been obtained.

QJ813 -H— Binaries, documentation (media: C, D, E, Q, T, Y)

QJ813 -H— Right to copy for single-use (under existing license), no binaries, no documentation, no support services (media: Z)

Users of FORTRAN IV/RT-11, Version 1C or Version 2.0, whose specified Support Category warranty has not expired may order under license the following software update for the then current media charge. The update is distributed in source or binary form on the appropriate medium and includes no installation or other services unless specifically stated.

QJ813 -W— Binaries, documentation (media: C, D, E, Q, T, Y)

Source/Listing Update Options:

The following options are available to licensed users as updates to source/listing options. The update is distributed in source form on the appropriate medium and includes no installation or other services unless specifically stated otherwise.

QJ813 -N— Sources update (media: D, E, Q, T)

Miscellaneous Options

QJ813 -G— Documentation only kit (media: Z)

ADDITIONAL SERVICES:

None

The Digital Equipment Computer Users Society

DECUS

DECUS, the Digital Equipment Computer Users Society, was established in March of 1961 to advance the effective use of DIGITAL computers. It is a voluntary, not-for-profit users group, supported in part by Digital Equipment Corporation.

OBJECTIVES

The objectives of the Society are to advance the effective utilization of computers, computer peripheral equipment, and software manufactured and marketed by Digital Equipment Corporation, by promoting the interchange of information concerning their uses; advance the art of computation through mutual education and exchange of ideas and information; establish standards and provide channels to facilitate the exchange of computer programs among DECUS members; provide feedback to the computer industry on equipment and software needs; and to reduce the duplication of development efforts.

ACTIVITIES

1. SYMPOSIA

Symposia are held throughout the year in each of the DECUS Chapters. These meetings provide a forum for users of DIGITAL computers to meet with other users and with DIGITAL management, engineers, and Software Services and Field Service representatives. They are an opportunity for users to participate in DIGITAL Product Workshops and Product Planning feedback sessions. The technical papers and presentations from each symposium are published as DECUS Proceedings after each meeting and provide a permanent record of the meetings activities.

2. SPECIAL USER GROUPS

DECUS encourages subgrouping of users with common interests and/or geographical proximity.

Special Interest Groups (SIGs) promote the interchange of specialized information and have no geographical limitations. Specializations may be for application areas, subject areas (such as languages), or specific operating systems. A group of users must petition the Chapter Executive Board for recognition as a Special Interest Group. The group must have a chairman, and its organization must meet the guidelines of the Chapter Executive Board.

Examples of active SIGs are users of RSX-11, RSTS, RT-11 users, business system users, etc. For additional information, contact your Chapter Executive Secretary.

One of the most successful subgroupings are Local Users Groups (LUGs). There are numerous active LUGs in Australia, Canada, Europe, and the U.S. Local User Groups are basically geographic in nature; however, they may be geographic and specific as well.

The largest Special User Group is composed of users of the DECsystem-10 and DECsystem-20.

3. STANDARDS

DECUS promotes user activity in reviewing DIGITAL standards. Users are given the opportunity to comment on DIGITAL standards prior to their finalization.

4. PROGRAM LIBRARY

One of the major activities of the users group is the DECUS Program Library. The Library contains programs written and submitted by users and is maintained and operated separate from the Digital Software Distribution Center. A wide range of software is available, including languages, editors, numerical functions, utilities, display routines, and various other types of application software.

DECUS (continued)

Library catalogs, updated periodically, contain descriptive abstracts and ordering information.

Information and forms for submitting programs to the Library may be obtained from local DECUS offices.

Programs are available to all members on a request basis. Orders for programs are made on DECUS Library Order Forms and directed to the local DECUS Chapter office. Information on the nominal service charge applied to most programs is published in the Library Catalogs.

As of January 1979, the Library contained approximately 1500 active software packages.

MEMBERSHIP

Membership in DECUS is voluntary and is not subject to a membership fee. Members are invited to take an active interest in the Society by contributing to the Program Library, to DECUSCOPE, and by participating in its Special User Groups and symposia. There are two types of membership: Installation Membership and Associate Membership.

INSTALLATION

An organization, institution, or individual that has purchased, leased, or has on order a computer manufactured by Digital Equipment Corporation is eligible for Installation Membership in DECUS. Membership status is acquired by submitting a written application to the appropriate Chapter Executive Secretary for approval by the Chapter Executive Board.

On acceptance of the application for membership, literature covering numerous DECUS services is sent to the Installation Delegate for reference and aid in maintaining active participation in the Society.

ASSOCIATE

Any person, who is not an appointed Installation Delegate, who has a bona fide interest in DECUS is eligible for Associate Membership.

Like Installation Members, Associate Members receive DECUSCOPE, the Society's quarterly newsletter, automatically. They may receive other DECUS material on request. Written application indicating desire to join must be submitted to the appropriate Chapter Executive Secretary for approval by the Chapter Executive Board.

On acceptance of the application for membership, literature covering the numerous DECUS services is sent to the member for reference and to enable active participation in the Society.

To obtain a membership form for DECUS, please return this form to the appropriate Chapter office listed below.

NAME: ______________________________

COMPANY: ______________________________

ADDRESS: ______________________________

____

CITY: ______________________________

STATE/COUNTRY: ____________________ ZIP: __________

Membership form Requested (check one):

□ Installation □ Associate

February 1979

I obtained this form from ______________________________.

DECUS OFFICES

DECUS Australia
P.O. Box 491
Crows Nest, New South
Wales 2065
Australia

DECUS Canada
P.O. Box 11500
Ottawa, Ontario K2H 8K8
Canada

DECUS Europe
C.P. 510
12, avenue des Morgines
CH-1213 Petit-Lancy 1,
Geneva, Switzerland

DECUS U.S. and
Office of the Executive Director
One Iron Way
Marlboro, Massachusetts 01752
USA

SOFTWARE PROBLEMS OR ENHANCEMENTS

Questions, problems, and enhancements to DIGITAL software should be reported on a Software Performance Report (SPR) form and mailed to the SPR Center at one of the following DIGITAL Offices: (SPR forms are available from the SPR Center).

AREAS COVERED	SPR CENTER
United Kingdom Middle East	DIGITAL EQUIPMENT CORP., Ltd. Fountain House, Butts Center RG1 7QN READING / England
Austria, Poland, Hungary, Rumania, East Germany, West Germany, Russia, Czechoslovakia, Bulgaria	DIGITAL EQUIPMENT CORP., Gmbh Wallensteinplatz 2 8 MUNICH 40 / Germany
Israel	DECSYS COMPUTERS, LTD. Yirmiyahou Street 4 TEL AVIV 63505 / Israel
France	DIGITAL EQUIPMENT FRANCE Silic 225 18, rue Saarinen 94528 RUNGIS Cedex / France
Italy	DIGITAL EQUIPMENT SPA Viale Fulvio Testi 117 20092 CINISELLO/BALSAMO (Milan) Italy
Denmark	DIGITAL EQUIPMENT CORP. APS Kristineberg 3 2100 COPENHAGEN 0 / Denmark
Finland	DIGITAL EQUIPMENT CORP. OY P.L. 16 02201 ESPOO 20 / Finland
Norway	DIGITAL EQUIPMENT CORP. A/S Pottenmakerveien 8 OSLO 5 / Norway
Sweden	DIGITAL EQUIPMENT CORP. A.B. Englundavagen 7 17124 SOLNA 1 / Sweden
Switzerland, Spain, Greece, Portugal, Yugoslavia, Cyprus, Algeria, Morocco, Malta, Tunisia, Turkey	DIGITAL EQUIPMENT CORP. S.A. 9, route des Jeunes 1211 GENEVE 26 / Switzerland
Holland, Belgium, Luxemburg	DIGITAL EQUIPMENT BV Kaap Hoorndreef 38 UTRECHT/OVERTRECHT / Holland
United States, remainder of Far East, Africa, Latin America	Administrative Services Group, SWS P.O. Box F Maynard, MA 01754
Canada	Digital Equipment Canada P.O. Box 11500 Kanata Canada K2H 8K8 Ontario
Australia (Melbourne)	Digital Equipment Aust. Pty., LTD. 70-74 Park Street South Melbourne, Victoria Australia 3205
Australia (Sydney)	Digital Equipment Aust. Pty., LTD. 123 - 125 Willoughby Road P.O. Box 491 Crows Nest NSW Australia 2065
Brazil	Digital Equipment Comercio Ind. Rua Batatais 429 Esq AL Campin 01423 Jardim Paulista Sao Paulo 0100 Brazil
Caribbean	De Latin America P.O. Box 11038 Fernando Juncos Sta. Santurce PR 00910
Japan	Digital Equipment Corp., INTL 3rd Floor - Kowa Building 8-7 Sanban Cho Chiyoda Ku Tokyo 102 Japan
New Zealand	Digital Equipment Corp., LTD Challenge House - 3 Wolfe Street P.O. Box 2471 Auckland New Zealand 10010

DIGITAL EQUIPMENT CORPORATION, Corporate Headquarters: Maynard, Massachusetts 01754, Telephone: (617)897-5111—SALES AND SERVICE OFFICES: UNITED STATES—ALABAMA, Huntsville • ARIZONA, Phoenix and Tucson • CALIFORNIA, El Segundo, Los Angeles, Oakland, Ridgecrest, San Diego, San Francisco (Mountain View), Santa Ana, Santa Clara, Stanford, Sunnyvale and Woodland Hills • COLORADO, Englewood • CONNECTICUT, Fairfield and Meriden • DISTRICT OF COLUMBIA, Washington (Lanham, MD) • FLORIDA, Ft. Lauderdale and Orlando • GEORGIA, Atlanta • HAWAII, Honolulu • ILLINOIS, Chicago (Rolling Meadows) • INDIANA, Indianapolis • IOWA, Bettendorf • KENTUCKY, Louisville • LOUISIANA, New Orleans (Metairie) • MARYLAND, Odenton • MASSACHUSETTS, Marlborough, Waltham and Westfield • MICHIGAN, Detroit (Farmington Hills) • MINNESOTA, Minneapolis • MISSOURI, Kansas City (Independence) and St. Louis • NEW HAMPSHIRE, Manchester • NEW JERSEY, Cherry Hill, Fairfield, Metuchen and Princeton • NEW MEXICO, Albuquerque • NEW YORK, Albany, Buffalo (Cheektowaga), Long Island (Huntington Station), Manhattan, Rochester and Syracuse • NORTH CAROLINA, Durham/Chapel Hill • OHIO, Cleveland (Euclid), Columbus and Dayton • OKLAHOMA, Tulsa • OREGON, Eugene and Portland • PENNSYLVANIA, Allentown, Philadelphia (Bluebell) and Pittsburgh • SOUTH CAROLINA, Columbia • TENNESSEE, Knoxville and Nashville • TEXAS, Austin, Dallas and Houston • UTAH, Salt Lake City • VIRGINIA, Richmond • WASHINGTON, Bellevue • WISCONSIN, Milwaukee (Brookfield) • INTERNATIONAL—ARGENTINA, Buenos Aires • AUSTRALIA, Adelaide, Brisbane, Canberra, Melbourne, Perth and Sydney • AUSTRIA, Vienna • BELGIUM, Brussels • BOLIVIA, La Paz • BRAZIL, Rio de Janeiro and Sao Paulo • CANADA, Calgary, Edmonton, Halifax, London, Montreal, Ottawa, Toronto, Vancouver and Winnipeg • CHILE, Santiago • DENMARK, Copenhagen • FINLAND, Helsinki • FRANCE, Lyon, Grenoble and Paris • GERMAN FEDERAL REPUBLIC, Cologne, Frankfurt, Hamburg, Hannover, Munich, Nuremburg, Stuttgart and West Berlin • HONG KONG • INDIA, Bombay • INDONESIA, Djakarta • IRELAND, Dublin • ITALY, Milan, Rome and Turin • IRAN, Tehran • JAPAN, Osaka and Tokyo • MALAYSIA, Kuala Lumpur • MEXICO, Mexico City • NETHERLANDS, Utrecht • NEW ZEALAND, Auckland and Christchurch • NORWAY, Oslo • PUERTO RICO, Santurce • SINGAPORE • SPAIN, Madrid • SWEDEN, Gothenburg and Stockholm • SWITZERLAND, Geneva and Zurich • UNITED KINGDOM, Birmingham, Bristol, Epsom, Edinburgh, Leeds, Leicester, London, Manchester and Reading • VENEZUELA, Caracas •