The Software Dispatch

RT-11

December 1982

AD-C740C-34

software
digital

RT-11 SOFTWARE DISPATCH

The RT-11 Software Dispatch complements the RT-11 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).

PRODUCTS SUPPORTED in the RT-11 SOFTWARE DISPATCH

BASIC-11/RT-11 V2	**FORTRAN IV/RT-11 V2.5**	**MSB/FORTRAN IV V1**
CTS-300 V6/V7	**GAMMA-11 F/B V3.1**	**RT-11 V4**
DECnet-RT V1.1	**LSP-11 V1.1**	**RT-11 2780 3780**
FMS-11/RT-11 V1.1	**MSB11 V1.2**	**Protocol Emulator V4**
		SSP-11 V1.3

DISTRIBUTION
The RT-11 Software Dispatch is directed to one software contact for each software product. No mailing will be made to addresses without a software contact name. **Address change requests should be sent to the nearest DIGITAL field office. Include the new address and mailing label from the most recently received publication.**

Software binary 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.

Ann Owens, Associate Editor

TRADEMARKS of DIGITAL EQUIPMENT CORPORATION
Maynard, Massachusetts

DEC	DECwriter	PDT
DECUS	DIBOL	RSTS
DIGITAL LOGO	EDUsystem	RSX
DECnet	IAS	UNIBUS
DECsystem-10	MASSBUS	VAX
DECSYSTEM-20	PDP	VMS
		VT

RT-11 Software Dispatch, December 1982

TABLE OF CONTENTS

PRODUCT AVAILABILITY DATES - RT-11

DECEMBER 1982

The following are dates products have become available.
Customers who are in warranty or have a Software Product Service
contract during the month the product became available are
eligible to receive the update. Customers who are eligible and
have not received the update should contact their local Digital
office.

Autopatch is distributed to Software Product Service Basic
contract customers and to Self-Maintenance contract customers who
have selected this option. Autopatch will be installed for
DECsupport contract customers as part of their Preventive
Maintenance.

PRODUCT	VERSION	AVAILABLE
CTS-300	7.0	MAR 82
CTS-300 AUTOPATCH	A	SEP 82
DECNET-RT	2.0	MAR 82
DECTYPE-300	1.2	APR 82
GAMMA-11	3.2	AUG 82
LSP-11	1.2	NOV 81
MACDBG	1.0	MAR 82
MACSYS-RT	1.0	JUL 82
MU-BASIC	2.1	SEP 81
PROM/RT-11	2.0	AUG 82
QUILL	1.1	SEP 82
SPETS-11	1.0	JUL 82
SSP-11	1.3	NOV 81
RT-11 AUTOPATCH	G	AUG 82
RGL FEP/RT-11	1.0	MAY 82
QUILL	1.0	MAR 82

SPR USER LETTER

Submitted by Sheila Hatchell, 8/11 Administration

How to Make the Best Use of the SPR Form

What We Can Do for You:

1. Blank SPR forms are returned with each SPR acknowledgement and are available upon request in the desired quantities through the SPR Administration (P.O. Box F) and your local office/SPR Center.

2. Copies of the SPR acknowledgement and answer are sent to the appropriate DIGITAL Office/SPR Center for their information.

3. STATUS FOR SUBMITTED SPRs IS PROVIDED UPON REQUEST.

4. SPRs marked PROBLEM/ERROR will have a response for DIGITAL SUPPORTED products. These SPRs should refer to suspected deficiencies in the software.

5. SPRs marked SUGGESTION are forwarded to the pertinent software group for information purposes, and are responded to at their discretion.

What You Can Do for Us:

1. Fill out the form completely either by typing or printing clearly. **PLEASE INCLUDE YOUR SOFTWARE SERVICE CUSTOMER NUMBER IN THE ADDRESS BOX.**

2. Limit only one problem per SPR form. Several problems on an SPR can lengthen the turnaround time.

3. WHENEVER POSSIBLE, SUBMIT AN SPR WITH ATTACHMENTS, SUCH AS MACHINE READABLE DATA, DETAILED INSTRUCTIONS ON HOW TO REPRODUCE THE PROBLEM, PROGRAM AND/OR DATA FILES, LISTINGS, AND CONSOLE LOG.

4. It would be helpful to all concerned if problems with patches are reported as soon as possible.

5. For security SPRs, it is imperative that the DO NOT PUBLISH box be marked.

6. It would be helpful if tapes submitted with SPRs are labeled (track and density), and have a directory attached.

7. Complete the questionnaire that is supplied with each SPR answer. Your feedback is essential in monitoring the quality of our responses.

8. SPRs should not be used for problems concerning software policy, software distribution, or hardware. The local office should be contacted in these cases.

CTS-300 V07   Seq 52.9.2 M
  for RT-11 V4.0
TSD LINE PRINTER SPOOLER
LPTSPL   1 of 5
VB07-01A

PATCH 19: ERRORS IN LINE PRINTER SPOOLER PACKAGE (LG)

1. If RT-11 has been started without a system date and a
    satellite goes off-line sending a message to LPTSPL, then the
    error "?DIBOL-E7--NT-Subscript error" is generated.

    Patch 19 corrects this so that the above situation does not
    result in an Error 7.

2. Errors logged by the spooler in LPQLOG.LPQ sometimes can't be
    seen by QUE. LPTSPL doesn't force the write of error
    messages to the log file correctly if the error record
    written is within the first block of the file.

    Patch 19 ensures that the first block of the log file is
    always written.

3. A problem exists with the line printer spooler if there is
    more than one default lineprinter and if one of the default
    satellites finds its associated lineprinter busy. When this
    occurs the satellite opens the printer once it is free, but
    does not release it if there are no longer any files to be
    printed. As a result, any program directly outputting to
    this printer receives device in use errors.

    Patch 19 corrects this so that the lineprinter will remain
    open only if there is something to print.

4. The QUE/INTERRUPT statement does not work as documented. The
    command:

         QUE/I/PA=1/DEV=LP

    should interrupt the job printing at LP: and restart it from
    page 1. Printing is interrupted, and resumes on a new page,
    but continues from the point at which it was interrupted.

    Patch 19 corrects this so that QUE/INTERRUPT interrupts the
    file being printed and resumes printing at the requested page
    number.

CTS-300 V07   Seq 52.9.2 M
   for RT-11 V4.0
TSD LINE PRINTER SPOOLER
LPTSPL   2 of 5
VB07-01A

The version number of LPTSPL changes to VB07-01B.

Using the editor, create the following files as shown. Name them
as indicated in the comment line that is the first line of each
file. Then, to install the patch, follow the procedure shown
following the file.

Corrections are made to the source modules using the SLP (Source
Language Patch) program. Please note that the first record in
each patchfile is the comment and the last record is "/". You
must terminate each line in those files with a carriage return,
including the last line "/".

NOTE

The DISPLAY statement in patch file P019A.PAT appears as
two lines in the patch (the second line being
"-01B',13,10,10)"), but should be typed as one line. In
other words, do not press the carriage return key after
the words "Spooler - VB07"; type the line continuously.

4

CTS-3ØØ VØ7
   for RT-11 V4.Ø
TSD LINE PRINTER SPOOLER
LPTSPL
VBØ7-Ø1A

Seq 52.9.2 M

3 of 5

```
;PO19A.PAT
-1,1
-133,133
\
        DISPLAY(TT,13,10,'Time-Shared DIBOL Line Printer Spooler - VB07
-01B',13,10,10)
/


;PO19B.PAT
-122,122
        IF ((NUMRC(SCHED).EQ.0).AND.(NUMRC(DFALT).EQ.0)) GOTO FLUSH
-127,127
OP1,    IF (NUMRC(SCHED).EQ.0) GOTO FLUSH
-156,156
\
        IF (HI.EQ.0) GOTO FLUSH          ;IF NONE THAT ARE OK TO  PRINT LEAVE
/


;PO19C.PAT
-132
        IF (DATE.NE.'             ') GOTO T3A        ;CHECK FOR NO DATE
        ERDAY(1,6)=                                  ;SET TO ZERO IF IT IS
        GOTO T5A
T3A,
-140
T5A,
-147
        READ (LOG,TEMPP,LMAX)           ;FORCE THE WRITE
-211
        UNLOCK LOG
/


;PO19D.PAT
-1,1
-87
        INTR=
-96,96
-119,119
        IF (PP2.EQ.0) PP2=999
-121
        ONERROR ERR5
-122
        NPG=NPG-1
        IF (NPG.EQ.0) GOTO P2
-125
        GOTO PG3
-188
        ONERROR ERR5
/
```

CTS-3ØØ VØ7
   for RT-11 V4.Ø
TSD LINE PRINTER SPOOLER
LPTSPL
VBØ7-Ø1A

```
;PO19E.PAT
-126,126
AYR,     D2
-198,198
        IF  (MON.EQ.0)  AMON='XXX'
        IF  (MON.NE.0)  AMON=MONTH(MON)
/


.RENAME (LPSPL2,LPSPL5,LPSPL9,L*SAT,QUE2).DBL .OLD
 Files renamed:
DK:LPSPL2.DBL   to DK:LPSPL2.OLD
DK:LPSPL5.DBL   to DK:LPSPL5.OLD
DK:LPSPL9.DBL   to DK:LPSPL9.OLD
DK:LPSAT.DBL    to DK:LPSAT.OLD
DK:LQSAT.DBL    to DK:LQSAT.OLD
DK:LRSAT.DBL    to DK:LRSAT.OLD
DK:LSSAT.DBL    to DK:LSSAT.OLD
DK:QUE2.DBL     to DK:QUE2.OLD

.R SLP
*LPSPL2.DBL=LPSPL2.OLD,PO19A.PAT
*LPSPL5.DBL=LPSPL5.OLD,PO19B.PAT
*LPSPL9.DBL=LPSPL9.OLD,PO19C.PAT
*LPSAT.DBL=LPSAT.OLD,PO19D.PAT
*LQSAT.DBL=LQSAT.OLD,PO19D.PAT
*LRSAT.DBL=LRSAT.OLD,PO19D.PAT
*LSSAT.DBL=LSSAT.OLD,PO19D.PAT
*QUE2.DBL=QUE2.OLD,PO19E.PAT
*^C

.R DICOMP
*LPSPL2=LPSPL2/O

     NO  ERRORS DETECTED
*LPSPL5=LPSPL5/O

     NO  ERRORS DETECTED
*LPSPL9=LPSPL9/O

     NO  ERRORS DETECTED
*LPSAT=LPSAT/O

     NO  ERRORS DETECTED
*LQSAT=LQSAT

     NO  ERRORS DETECTED
*LRSAT=LRSAT/O

     NO  ERRORS DETECTED
*LSSAT=LSSAT/O

     NO  ERRORS DETECTED
*QUE2=QUE2/O

     NO  ERRORS DETECTED
*^C
```

CTS-3ØØ VØ7
  for RT-11 V4.Ø
TSD LINE PRINTER SPOOLER
LPTSPL
VBØ7-Ø1A

```
.R LINK
*LPTSPL.TSD/B:100000=LPTSPL,TDIBOL/P:500.//
*LPSPL1,LPSPL7,QUP,QUPM/O:1
*LPSPL2/O:1
*LPSPL3/O:1
*LPSPL4/O:1
*LPSPL5/O:1
*LPSPL6/O:1
*LPSPL8/O:1
*LPSPL9/O:1
*//
*LPSAT.TSD/B:100000=LPSAT,TDIBOL/P:500.
*LQSAT.TSD/B:100000=LQSAT,TDIBOL/P:500.
*LRSAT.TSD/B:100000=LRSAT,TDIBOL/P:500.
*LSSAT.TSD/B:100000=LSSAT,TDIBOL/P:500.
*QUE.TSD/B:100000=QUE,TDIBOL/P:500.//
*QPRS1,QPRS2
*QUP,QUPM
*QAGN
*QIN1,QIN1M/O:1
*QIN2/O:1
*QUE1/O:1
*QUE2/O:1
*QUE3/O:1
*QUE4/O:1//
*^C

.R REDUCE
*LPTSPL/N
*LPSAT/N
*LQSAT/N
*LRSAT/N
*LSSAT/N
*QUE/N
*^C
```

7

CTS-300 V07   Seq 52.15.5 M
    for RT-11 V4.0
MACRO SORT   1 of 5
SORT.SAV V07-OD
SORT.TSD V07-OC

PATCH 20: INCOMPLETE SORT, FILESPEC ERRORS (LG)

The following errors exist in both the single-user and
time-shared versions of the Macro Sort:

1. If a data file is being sorted where a record(s) ends on a
    block boundary, it is possible that the sort will stop
    before all the records have been reordered.

    Patch 20 corrects this so that the sort continues to
    completion.

2. If two or more filenames are specified in the Sort INPUT
    statement, indicating a merge, the error "?DIBOL-E18-T--File
    not found" or the error "?DIBOL-E17--T-Bad file
    specification" may be generated even though the files exist
    and are specified correctly.

    Also, if either the input or output file specification is a
    truncated version of the other (e.g. INPUT=PRGNAM.ABC,
    OUTPUT=PRGNAM.AB) the error "?DIBOL-E18-T--File not found"
    may be generated. If, in addition, a TAGS sort is specified,
    the errors "?SORT-E21--W-TAGSORT type invalid" and
    "?SORT-E26--F-SORT mode file name/type conflict" are
    incorrectly reported, rather than the Error 18.

    Patch 20 causes the requested merge or sort to be performed
    without reporting the errors described above.

The version number of SORT.SAV changes to V07-OE, and SORT.TSD
changes to V07-OD.

Using the editor, create the following files as shown. Name them
as indicated in the comment line that is the first line of each
file. Then, to install the patch, follow the procedure shown
following the files.

9

CTS-3ØØ VØ7
   for RT-11 V4.Ø
MACRO SORT
SORT.SAV VØ7-OD
SORT.TSD VØ7-OC

Seq 52.15.5 M

2 of 5

```
;P020A.MAC

            .TITLE   SORTD
            .CSECT   SORTD
IFNXT=30270
IFF=30274
ITOP=30304

P020:
            .=       .+3662
            JMP      P020A

            .PSECT   $P010
            .=       .+12
            JMP      P020B

            .PSECT   $P020A
P020A:      CMP      ITOP,IFF
            BHIS     2$
            MOVB     @ITOP,R0
            JMP      P020+3666
2$:         JMP      P020+3674
P020B:      CMP      IFNXT,IFF
            BHIS     3$
            JMP      P020+3222
3$:         JMP      P020+3234
            .END


;P020B.MAC

            .TITLE   SORTD
            .CSECT   SORTD
IFNXT=110564
IFF=110570
ITOP=110600

P020:
            .=       .+3734
            JMP      P020A

            .PSECT   $P010
            .=       .+12
            JMP      P020B

            .PSECT   $P020A
P020A:      CMP      ITOP,IFF
            BHIS     2$
            MOVB     @ITOP,R0
            JMP      P020+3740
2$:         JMP      P020+3764
P020B:      CMP      IFNXT,IFF
            BHIS     3$
            JMP      P020+3274
3$:         JMP      P020+3306
            .END
```

CTS-3ØØ VØ7
  for RT-11 V4.Ø
MACRO SORT
SORT.SAV VØ7-OD
SORT.TSD VØ7-OC

Seq 52.15.5 M

3 of 5

```
;PO20C.MAC

            .TITLE   SORTA
            .CSECT   SORTA
            .GLOBL   INPUTL
P020:
            .=       .+450
            JSR      PC,P020A

            .PSECT   $P020B
P020A:      ADD      #INPUTL,R0
            MOV      R0,P020+462
            RTS      PC
            .END



;PO20D.MAC

            .TITLE   SORTA
            .CSECT   SORTA
            .GLOBL   INPUTL
P020:
            .=       .+460
            JSR      PC,P020A

            .PSECT   $P020B
P020A:      ADD      #INPUTL,R0
            MOV      R0,P020+472
            RTS      PC
            .END



;PO20E.MAC

            .TITLE   SORTC
            .CSECT   SORTC
            .GLOBL   INPUTL,RTMP,EOLCNT
P020:
            .=       .+11146
            JMP      P020A

            .PSECT   $P020C
P020A:      MOV      RTMP,R0
            JSR      R4,EOLCNT
            BR       1$
            .WORD    INPUTL
            .WORD    RTMP
1$:         CMP      R0,RTMP
            BNE      2$
            JMP      P020+11152
2$:         JMP      P020+11174
            .END
```

CTS-3ØØ VØ7
  for RT-11 V4.Ø
MACRO SORT
SORT.SAV VØ7-OD
SORT.TSD VØ7-OC

;PO2OF.MAC

```
        .TITLE  SORTC
        .CSECT  SORTC
        .GLOBL  INPUTL,RTMP,EOLCNT
P020:
        .=      .+11160
        JMP     PO2OA

        .PSECT  $PO2OC
PO2OA:  MOV     RTMP,RO
        JSR     R4,EOLCNT
        BR      1$
        .WORD   INPUTL
        .WORD   RTMP
1$:     CMP     RO,RTMP
        BNE     2$
        JMP     PO2O+11164
2$:     JMP     PO2O+11206
        .END
```

;PO2OV1.MAC

```
        .TITLE  SORTR
        .PSECT  SORTR

        .=      .+21
        .BYTE   'E
        .END
```

;PO2OV2.MAC

```
        .TITLE  SORTR
        .PSECT  SORTR

        .=      .+23
        .BYTE   'D
        .END
```

```
.RENAME (SRT11D,SORTD,SRT11A,SORTA).OBJ *.OLD
 Files renamed:
DK:SRT11D.OBJ  to DK:SRT11D.OLD
DK:SORTD.OBJ   to DK:SORTD.OLD
DK:SRT11A.OBJ  to DK:SRT11A.OLD
DK:SORTA.OBJ   to DK:SORTA.OLD

.RENAME (SRT11C,SORTC,SRT11R,SORTR).OBJ *.OLD
 Files renamed:
DK:SRT11C.OBJ  to DK:SRT11C.OLD
DK:SORTC.OBJ   to DK:SORTC.OLD
DK:SRT11R.OBJ  to DK:SRT11R.OLD
DK:SORTR.OBJ   to DK:SORTR.OLD
```

CTS-3ØØ VØ7   Seq 52.15.5 M
   for RT-11 V4.Ø
MACRO SORT   5 of 5
SORT.SAV VØ7-OD
SORT.TSD VØ7-OC


```
.MACRO P020A,P020B,P020C,P020D
ERRORS DETECTED:  0
ERRORS DETECTED:  0
ERRORS DETECTED:  0
ERRORS DETECTED:  0

.MACRO P020E,P020F,P020V1,P020V2
ERRORS DETECTED:  0
ERRORS DETECTED:  0
ERRORS DETECTED:  0
ERRORS DETECTED:  0

.R PAT
*SRT11D.OBJ=SRT11D.OLD/C:050507,P020A/C:035035

.R PAT
*SORTD.OBJ=SORTD.OLD/C:056025,P020B/C:036451

.R PAT
*SRT11A.OBJ=SRT11A.OLD/C:044171,P020C/C:014405

.R PAT
*SORTA.OBJ=SORTA.OLD/C:044727,P020D/C:014455

.R PAT
*SRT11C.OBJ=SRT11C.OLD/C:120064,P020E/C:023653

.R PAT
*SORTC.OBJ=SORTC.OLD/C:105012,P020F/C:023761

.R PAT
*SRT11R.OBJ=SRT11R.OLD/C:170102,P020V1/C:005024

.R PAT
*SORTR.OBJ=SORTR.OLD/C:030612,P020V2/C:005031

.R LINK
*SORT,SRT11/M:1400/B:1400=RTIO,SRT11O,SRT11R/P:500./C
*MSGLIB/C
*SRT11C/O:1/C
*SRT11A/O:1/C
*SRT11D/O:1/C
*SRT11M/O:1
*SORT.TSD,SORT=SORTR,SRTIO/B:100000/P:500./C
*SORTC/O:1/C
*SORTA/O:1/C
*SORTD/O:1/C
*SORTM/O:1
*^C

.R REDUCE
*SORT/N
*^C
```

USING FORTRAN AND LINK OPTIONS WITH RGL/FEP (MC)

1.0 PROBLEM: USING FORTRAN AND LINK OPTIONS WITH RGL/FEP

     The RGL/FEP subroutine library expects minimal run time I/O
support of at least 136 bytes. That means that when a programmer
compiles his application program (making calls to the subroutines in
RGL/FEP), the RECORD option of the FORTRAN compiler must be 136 or
more to support RGL/FEP I/O statements. Since the default maximum
record size may vary from compiler to compiler (depending upon how it
was generated), the programmer must be aware of this option and set it
if the default is not 136 or more. He may do so using the /RECORD:n
switch of the compiler.

     RGL/FEP provides a program development tool called RGLLNK.
RGLLNK.FOR generates the command stream that compiles, links and runs
a simple RGL/FEP application. RGLLNK.FOR currently uses the FORTRAN
default for I/O record lengths. This update changes RGLLNK so that it
explicitly specifys the record length. RGLLNK will then work
independently of the compiler's default.

     This is the fourth update to the RGL/FEP product. If you haven't
done so already, you must complete the first three updates in order
before you proceed.

     This problem appears in both RGL/FEP as a component of FEP and in
RGL/FEP as a separate package. Likewise, this update applies to both
products.

2.0 SOLUTION

     To correct this problem, you must update the program RGLLNK.FOR.
Then you must recompile and relink it.

     The correction process is completed in three parts. The first
and third parts are common to all media. (They are described in
sections 2.2 and 2.6 respectively.) The first part sets up the update,
the third verifys the change. The second is media specific. You
could have received your RGL/FEP software in one of four ways. If you
received it on 800 bpi, 9 track magtape or on a hard disk (RK05,
RL01/2, ect.), then use the update procedure in section 2.3. If you
received RGL/FEP on nine RX01's, then use the procedure in 2.4.
Otherwise, you must have received your software on four RX02's; use
section 2.5.

     If you received your software on a hard disk or on a magtape,
then you will need to create two files on your system volume:
590104.COM and RGLLNK.001. Otherwise, you will need only create
RGLLNK.001. All the necessary files are listed in section 3.

     If your system is based on dual RX02 diskette drives, then it is
suggested that you create a system diskette with the following
utilities and drivers installed:

1. SWAP.SYS

2. RT11SJ.SYS (put this on the boot block too)

3. TT.SYS

4. DY.SYS

5. DUP.SAV

6. DIR.SAV

7. SLP.SAV

8. KED.SAV

9. MACRO.SAV

10. LIBR.SAV

11. STARTS.COM

12. FORTRA.SAV

It is also suggested that once you have built this system diskette
that you label it and reserve it exclusively for the correction of
your RGL/FEP software and store it along with your RGL/FEP
distribution kit (in some safe place).

2.1 Assumptions

     You must satify these conditions before continuing.

1. You must have a copy of the RGL/FEP distribution kit updated to
    the current patch level. This is the fourth update to the RGL/FEP
    package. You must have installed the first three updates before
    you may proceed.

2. You must have spare media of the same type and quantity as your
    current installation media. If you have a dual RX02 system then
    you will also need two extra RX02's (six total) -- one for your
    RT-11 system and one spare to update the RGL/FEP library. On the
    other hand, if your default device is a hard disk and you got
    RGL/FEP on RX02's, then you will only need five spare floppy
    diskettes. If your installation medium is a magtape, then you
    will need a spare hard disk.

3. You must have an RT-11 system, version 4.0 which has the following
    utilities installed on the system volume: LINK, FORMAT, DUP, SLP,
    LIBR. You will also need to have some editor installed such as
    KED.

4. You must have installed a FORTRAN-IV compiler, version 2.5
    (FORTRA.SAV).

5. You have installed the FORTRAN Object Time System (FORLIB.OBJ)
    into SYSLIB.OBJ.   (If not, you must make suitable changes when
    linking your FORTRAN programs).

6. You must know how to boot up your RT-11 system, how to load and
    unload the your system's devices and know how to operate a text
    editor.

7. It is assumed that these procedures will be run to completion
    without error. This includes the installation of RGL/FEP once you
    have updated your software. If you get an error in a given step,
    retry that step.   If the error persists, you should check the
    appropriate manual. You may have to backtrack to preceding steps.
    As a precautionary measure, you should copy your current software.
    In that way you can always recover from any error by starting from
    the beginning.

2.2 Preparation For Patching

     These steps are common to all of the updates for the various
media.   Refer to the RT-11 System User's Guide, RGL/FEP Release Notes
and the FEP/RT Installation User's Guide for more information.

                                NOTE

    The notation dxn: stands for a device name of a hard disk
    drive.   Examples are DL∅:,   2: or DL5:. The notation Mxn:
    stands for magtape.   Examples are MT3:   and MM∅:.   The
    notation DYn: stands for an RX∅1/2 diskette, for example DY∅:
    or DY1:. Usually, this notation will apply to the RGL/FEP
    distribution volume.

1. Boot up your system.

2. Make sure you have the utilities necessary to complete the
    updating procedure (use the DIRectory utility). A list of minimal
    software to install RGL/FEP is given in the FEP/RT Installation
    and User's Guide.

3. Make sure you have the most current level of RGL/FEP.   You may
    check this with the Software Dispatch. If you find that your
    installation media are out of date, update them by applying each
    missing patch in order.

                                 17

4. Copy your current installation media using COPY. For hard disk,
    you will need another hard disk. For magtape, you should squeeze
    the media to a hard disk (the same as when you installed RGL/FEP).
    Make sure your magtape drive is set to the 9-track, 8ØØ BPI
    default. This may be set by typing:

            SET Mx: DENSE=8ØØ
            SET Mx: DEFALT=9

    Don't use a unit number with the set command, e.g. if you are
    using MM3: type SET MM: DENSE=8ØØ and so forth. For RXØ1's
    only, you must format the diskettes to single density before
    copying them. You only need to copy diskettes 2/7 and 6/7. For
    RXØ2's, you need to copy 1/4, 2/4, and 3/4.

5. Using an editor (such as KED), create the files necessary to
    update the RGL/FEP library. If you have RGL/FEP on magtape or
    hard disk, create 59Ø1Ø4.COM and RGLLNK.ØØ1. Otherwise, just
    create RGLLNK.ØØ1. Those files are listed in section 3.Ø.

2.3 * * * Procedure For Hard Disk, Magtape   * * *

    You will need approximately 1ØØ free blocks on your system
volume. It should take you about fifteen minutes to complete this
procedure; it is suggested that you do it in one sitting. If you
have a magtape distribution and you have set up correctly, then you
have copied RGL/FEP onto a hard disk. If this is not the case, go
back to section 2.2.

1. Load your RGL/FEP disk into drive dxn:. If you have RGL/FEP on
    magtape, this disk will be the copy you made above.

2. Assign the logical name PAT to dxn:

            ASSIGN dxn: PAT:

3. Execute the patch by typing

            @59Ø1Ø4

4. If your installation medium is magtape, reinitialize your magtape
    and copy the disk onto it:

            INIT/NOQUERY Mxn:
            COPY/NOLOG ddn: Mxn:/POS:-1

5. Verify the change using section 2.6.

2.4 * * * Correction Procedure For RX01 Distribution   * * *

      This procedure is specific to RX01 distribution kits. It assumes
your system device is a hard disk. You will need approximately 100
free blocks on your system disk. It should take you about a half hour
to complete this procedure; it is suggested that you do it all in one
sitting. Using section 2.2, you should have formated and copied the
appropriate diskettes. If you have not, refer to section 2.2 now.

1. Load the RGL/FEP distribution diskette labeled 6/7 into a free
    drive and they type:

            COPY DYn:RGLLNK.FOR DK:
            RUN SLP
            RGLLNK.FOR=RGLLNK.FOR,RGLLNK.001/A
            ^C
            DELETE/NOQ RGLLNK.BAK
            FORT/CODE:THR/REC:136 RGLLNK
            LINK RGLLNK
            RENAME/NOPROT DYn:RGLLNK.FOR DYn:
            COPY/PREDEL RGLLNK.FOR DYn:
            DELETE/NOQ RGLLNK.OBJ

2. Next load the RGL/FEP distribution diskette labeled 2/7 and type:

            RENAME/NOPROT DYn:RGLLNK.SAV DYn:
            COPY/PRED RGLLNK.SAV DYn:
            RENAME/PROT DYn:RGLLNK.SAV DYn:

3. Verify your software according to section 2.6.

2.5 * * * Correction Procedure For RX02 Systems   * * *

      This procedure is specific to dual RX02 based systems. You will
need 100 free blocks on your system diskette. It should take you
about a half hour to complete this procedure; it is suggested that
you finish it in one sitting. Using section 2.2, you should have
copied the appropriate diskettes. If you have not, refer to section
2.2 now.

1. Load the diskette labeled 3/4 into a free drive and they type:

            COPY DYn:RGLLNK.FOR DK:
            RUN SLP
            RGLLNK.FOR=RGLLNK.FOR,RGLLNK.001/A
            ^C
            DELETE/NOQ RGLLNK.BAK
            FORT/CODE:THR/REC:136 RGLLNK
            LINK RGLLNK

```
        RENAME/NOPROT DYn:RGLLNK.FOR DYn:
        COPY/PREDEL RGLLNK.FOR DYn:
        DELETE/NOQ RGLLNK.OBJ
```

2. Next for diskettes 1/4 and 2/4, load each diskette and type
    (respectively)

```
        RENAME/NOPROT DYn:RGLLNK.SAV DYn:
        COPY/PRED RGLLNK.SAV DYn:
        RENAME/PROT DYn:RGLLNK.SAV DYn:
```

3. Verify the change using section 2.6.

2.6 Verifying The Update

    To check this software update, complete the following steps.

1. Reinstall the RGL/FEP subroutine library using the procedure used
    to install it initially (see the Release Notes: RGL/FEP for
    RT-11, AA-M521A-TC).   This   includes   running   the   RGL/FEP
    verification program RGLVFY.

2. You have now patched your RGL/FEP software. Save the updated
    installation volume(s) in a safe place.

3.Ø NECESSARY FILES FOR THE PATCH

    This section lists two files: an RT-11 indirect command file
590104.COM and a SLP correction file RGLLNK.ØØ1.

3.1 590104.COM (a)

     This command file is for hard disk and magtape distributions.
Note that ^C is two characters: ^ and C. This patch file assumes
that you have assigned PAT: to the correct device.


```
COPY PAT:RGLLNK.FOR DK:
RUN SLP
RGLLNK.FOR=RGLLNK.FOR,RGLLNK.001/A
^C
FORT/CODE:THR/REC:136 RGLLNK
LINK RGLLNK
DELETE/NOQ RGLLNK.OBJ
DELETE/NOQ RGLLNK.BAK
RENAME/NOPROT PAT:RGLLNK.FOR PAT:
COPY/PRED RGLLNK.FOR PAT:
RENAME/PROT PAT:RGLLNK.FOR PAT:
!* Done. Reinstall and verify.
```

3.2 SLP Correction Files

      This section lists the SLP correction file RGLLNK.001. Note that
this file must end with a carriage-return, linefeed (i.e. type RETURN
at the end of the last line). The <tab> symbol stands for the TAB
character; <cr> stands for carriage-return.

C*C*C<tab>Patch: 59.01.04<tab>Module: RGLLNK.FOR<tab>revision: 001
-50,50
<tab>logical*1 FMT1(33)<tab>!* MGC 001
-53,53
<tab>1'H','R','/','R','E','C',':','1','3','6','/',<tab>!* MGC 001
<tab>2'L','I','S','T',':','L','P',':',' '/<tab>!* MGC 001
-56,56
<tab>bldarr=23<tab>!* MGC 001
-62,62
<tab>bldarr=32<tab>!* MGC 001
-67,67
<tab>FMT1(30)=answer(1)<tab>!* MGC 001
-69,69
<tab>bldarr=29+i<tab>!* MGC 001
/<cr>

DEVELOPING RGL/FEP APPLICATIONS (MC)

1.Ø TUTORIAL: DEVELOPING RGL/FEP APPLICATIONS

     This article outlines various methods for developing RGL/FEP
applications and the options which must be present. It describes NO
changes to the RGL/FEP software.

     When developing RGL/FEP applications, a programmer must compile
his program, link it to the RGL/FEP library and run it. In general,
when he compiles a program, he can choose from twenty four different
compiler options to control how FORTRAN produces the object module.
He can specify, for example, whether arrays are to be vectored or not,
whether to produce a listing and how that listing will look.
Likewise, the linker also offers twenty four options which can
control, among other things, the generation of the executable image
and the format of the linker's listing. When compiling and linking a
program to the RGL/FEP library, the programmer is free to choose most
of the options. If he wishes, he can default most of them (which most
programmers are willing to do). But there is one option that he
should not, in general, default, unless he is sure that the default is
compatable with the RGL/FEP library. He must be sure that the FORTRAN
compiler produces run time input/output code with a maximum record
length of 136 bytes or more. If one's compiler defaults the /RECORD
option to 136 (or more) then the default is acceptable. If not, the
programmer must explicitly reset the record length to 136 or more
using the /RECORD switch. To see what options FORTRAN has used in
producing code, specify both the /LIST and /HEADER options.

     RGL/FEP provides two program development utilities: RGLOVR and
RGLLNK. With these, the programmer can compile, link and run his
application more easily. RGLOVR.COM is a command file which provides
the basic FORTRAN, LINK and RUN commands neccessary to produce an
executing RGL/FEP application using the RGL/FEP overlay scheme. As it
stands, RGLOVR.COM is NOT executable. It is a template which must be
edited. For example, suppose that a programmer wanted to build a
RGL/FEP application. This application, say, reads data from a data
file (call it subroutine READ), plots it (subroutine PLOT) and then
prints it out to a printer (subroutine PRINT). Suppose he has already
written and compiled the three subroutines READ, PLOT and PRINT and
now keeps them on DL1: (which is not his system disk). Now suppose
he wants to put them all together in a program MAIN. He knows that he
must compile MAIN. He wants to get the fullest listing FORTRAN will
allow. To save space, he doesn't want his arrays to be vectored. He
wants his integers to be I*4. When he links his program to the
subroutines, he wants a map with the program sections listed in
alphabetical order and output to a file rather than the line printer.
Because his application is large, he will overlay the RGL/FEP library.
Finally, having built an executable image, he wants to run it.

     Instead of typing all the neccessary commands to compile and link
the program, he can create a command file for his program from
RGLOVR.COM. Starting with RGLOVR.COM which looks like this:

     R LINK
     MAINPR,MAINPR=MAINPR,BLODAT,PRMLIB/A/W/P:4ØØ/B:12ØØ

```
        OV:LFIXED,OV:LOCAT2/O:1
        OV:LFREE,OV:LOCATE,OV:PPOINT/O:1
        OV:PDATA/O:1
        OV:DPAPER,OV:DEFALT/O:2
        OV:LNAXIS/O:2
        OV:LTAXIS/O:2
        OV:LNNICE,OV:LINMIN,OV:LINMAX/O:3
        OV:PRINUM,OV:PRISTR/O:3
        OV:FMINMX,OV:LGNICE/O:3
        OV:DPALOG/O:3
        OV:DPALIN/O:3//
        ^C
```

The programmer may copy it and edit it:

```
        COPY RGLOVR.COM MAIN.COM
        EDIT MAIN.COM
               .
               .
               .

        ASSIGN ddn: OV:
        @MAIN
```

Here, ddn: represents the volume where the overlay segments reside.
This resultant command file, specific to the program MAIN, will
compile, link and run MAIN with the stated options whenever it is
invoked:

```
        FORTRAN/I4/NOVECT/REC:136/SHOW:7/HEADER/STAT MAIN
        R LINK
        MAIN,MAIN=MAIN,DL1:READ,DL1:PLOT,/C
        DL1:PRINT,BLODAT,PRMLIB/A/W/P:4∅∅/B:12∅∅
        OV:LFIXED,OV:LOCAT2/O:1
        OV:LFREE,OV:LOCATE,OV:PPOINT/O:1
        OV:PDATA/O:1
        OV:DPAPER,OV:DEFALT/O:2
        OV:LNAXIS/O:2
        OV:LTAXIS/O:2
        OV:LNNICE,OV:LINMIN,OV:LINMAX/O:3
        OV:PRINUM,OV:PRISTR/O:3
        OV:FMINMX,OV:LGNICE/O:3
        OV:DPALOG/O:3
        OV:DPALIN/O:3//
        ^C
        RUN MAIN
```

        The second method uses RGLLNK. If the programmer is in the very
special situation where the RGL/FEP software is on the current system
volume, the program, also on the system volume, doesn't need to be
linked to any external object modules (other than RGL/FEP ones) and
most of the compiler and linker options are to be defaulted, then he
may use RGLLNK.COM. He need only answer the questions and the command

procedure will do the rest. If however, he wishes to change the command stream, say to link other, external modules or to use added LINK options, then he must first run RGLLNK.SAV and edit the resultant command file RGL.COM.

Take the case above. The programmer runs RGLLNK.SAV to produce RGL.COM. He then renames RGL.COM to MAIN.COM and edits it. He starts with:

```
FORT/CODE:THR MAIN
R LINK
MAIN=MAIN,BLODAT,PRMLIB/B:1200/A/W/P:1000//
LFIXED,LOCAT2/O:1
LFREE,LOCATE,PPOINT/O:1
PDATA/O:1
DPAPER,DEFALT/O:2
LNAXIS/O:2
LTAXIS/O:2
LNNICE,LINMIN,LINMAX/O:3
PRINUM,PRISTR/O:3
FMINMX,LGNICE/O:3
DPALOG/O:3
DPALIN/O:3//
^C
DEL MAIN.OBJ/NOQ
RUN MAIN
DEL MAIN.SAV
```

He wishes to do the same thing as before: compile the main program with several options and link them to subroutines on DL1:. He produces:

```
FORTRAN/I4/NOVECT/REC:136/SHOW:7/HEADER/STAT MAIN
R LINK
MAIN,MAIN=MAIN,DL1:READ,DL1:PLOT,/C
DL1:PRINT,BLODAT,PRMLIB/A/W/P:400/B:1200
LFIXED,LOCAT2/O:1
LFREE,LOCATE,PPOINT/O:1
PDATA/O:1
DPAPER,DEFALT/O:2
LNAXIS/O:2
LTAXIS/O:2
LNNICE,LINMIN,LINMAX/O:3
PRINUM,PRISTR/O:3
FMINMX,LGNICE/O:3
DPALOG/O:3
DPALIN/O:3//
^C
RUN MAIN
```

The only difference between this command procedure and the one generated from RGLOVR.COM is that the overlay segments are assumed to be on DK:, the system volume.

    Choosing between the various methods is subject to taste. In the case where the RGL/FEP libraries, object modules and applications all reside on the same disk and the applications are all self-contained programs, then one may use RGLLNK.COM.   In the case where added options are neccessary, but overlays are not, using RGLLNK.SAV to produce a command file and editing it is probably easiest. Finally, when one needs options and overlays, then copying and editing RGLOVR.COM is probably best.

RESTRICTIONS WITH GSAVE FILES (MC)

1.0 PROBLEM: RESTRICTIONS WITH GSAVE FILES

     This article describes the nuances of the GSAVE-GCLOSE-GLOAD
subsystem in RGL/FEP.   It reviews its use, explains it's intent and
describes two restrictions.

1.1 Tutorial: Saving VT125 REGIS Protocol In A File

     The RGL/FEP library is a collection of FORTRAN callable
subroutines that generate graphics for the VT125 graphics terminal.
The VT125 is driven by a graphics protocol known as ReGIS: Remote
Graphics Instruction Set. ReGIS instructions are sequences of ASCII
characters that define certain primative graphic objects (lines,
circles, curves) and their attributes (shading and color, line
patterns, ect.).   The RGL/FEP subroutine library extends the
capability of the VT125 ReGIS based terminal. In particular, it
provides capabilities for transforming coordinates, plotting data and
interactively locating positions on the screen. It also provides an
environment to make the generation of graphics images easie..

     RGL/FEP also provides the capability of simultaneously generating
a display on the VT125 and saving the ReGIS strings that produced that
display in an ASCII file. The file can then be "played back" at a
later time from another RGL/FEP application. In this way, the file
can be considered to be like a macro (with no parameters).   Whenever
it is needed to create a picture (or part of one), it can be displayed
using GLOAD. That picture will appear on the screen (usually) far
quicker than it would take to regenerate it. To create such a ReGIS
file, the user simply creates a RGL/FEP application program (in
FORTRAN) and debugs it until he gets the picture he wants. When he is
satisfied with the result, he simply inserts a GSAVE call (after the
initial call to INITGR) at the start of the application and a GCLOSE
call at the end. Most ReGIS commands generated by the program will be
saved in the designated file. For example:

        PROGRAM exampl
            .
            .
            .
        CALL INITGR(5)
        CALL GSAVE('later.reg',)
        CALL INITGR(5)   !* Optional - only if the programmer wants
            .   !* to reset attributes when he loads the file.

            .   !* All picture generating strings
            .   !* between GSAVE and GCLOSE
            .   !* go to file LATER.REG.
        CALL GCLOSE
        END

The programmer could later use the file LATER.REG in the following
way:

```
      PROGRAM useit
           •
           •
           •
      CALL INITGR(5)
           •
           •
           •
      !* Displays picture generated by program EXAMPL
      CALL GLOAD('later.reg', )
           •
           •
           •
      END
```

The GSAVE, GCLOSE and GLOAD subroutines were intended to allow the
user to save certain commonly used pictures and quickly display them
rather than constantly regenerating them.

    GSAVE, however, does not generate all the ReGIS strings
neccessary to recreate the picture outside of the RGL/FEP subroutine
library. Specifically, if the user edits a file generated by GSAVE
(adding the ReGIS prefix <esc>Pp and suffix <esc>) and TYPEs it
directly to his VT125 terminal, he may get a result which is different
then if he GLOADed the file from a RGL/FEP supported FORTRAN program.
In general, RGL/FEP provides for certain graphics figures called
"macrographs" to be saved inside the VT125 memory for later use.
These macrographs are always available to RGL/FEP subroutines but are
transparent to the user. If the user sends the ReGIS strings from a
file to the VT125 terminal outside of RGL/FEP, then the macrographs
are not defined and hence not available therefore the picture may
appear different.

    ReGIS text files are generated by RGL/FEP and are meant to be
used by RGL/FEP. All other uses of those files may produce unexpected
results without the environment that the RGL/FEP library provides.

    RGL/FEP also provides the capability of locating various points
on the VT125 screen. The programmer can query RGL/FEP about the
current location (GETLOC), or can move a cursor around the screen
interactively (LOCATE, LFIXED or LFREE). If the programmer makes a
call to any of the cursor routines, he can not predict where a user
will move the cursor (since these routines are interactive). In
general, any call to LOCATE, LFIXED or LFREE could produce different
cursor movements hence different ReGIS strings. Saving these strings
in a text file would be useless: the cursor would no longer be
interactive and would move in only one fixed pattern. Therefore, if
the programmer calls any of the cursor routines while saving ReGIS
strings in a file with GSAVE, the ReGIS strings associated with those
routines will NOT be saved. In short, all ReGIS strings generated by
calls to LOCATE, LFIXED and LFREE will NOT be saved in a GSAVE file.

   This feature can be quite useful when generating a program to
produce GSAVE files.   Take the following (perhaps trivial) example:
the program BOXSAV. BOXSAV will save the ReGIS strings neccessary to
produce a box which might perhaps serve as a frame for later graphics.
BOXSAV positions the cursor to the corners of the box and then draws a
box.   When the program GSAVE's the ReGIS instructions, it will save
only the BOX command and not the cursor movement:

```
      c
      c
      c     BOXSAV
      c
      c     This program produces frames for graphics interactively.
      c
            PROGRAM boxsav
      c
            LOGICAL*1 file(50)       !* string - name of GSAVE file
            REAL x1,y1,x2,y2         !* coordinates of the frame
      c
            CALL initgr(5)
            CALL clrscr
            CALL clrtxt
      c
            CALL linetx(1,1,
           1 'Interactive Frame Generation: BOXSAV')
      c
            CALL getnam ( file )
            CALL gsave(file, )
      c
            x1 = 0
            y1 = 0
            CALL getpos(x1,y1,1)
            x2 = x1
            y2 = y1
            CALL getpos(x2,y2,2)
            CALL box(x1,y1,x2,y2)
      c
            CALL gclose
      c
            END
      c
            SUBROUTINE getnam ( file )
            LOGICAL*1 file(1)
      c
            WRITE (5,1000)
      1000  FORMAT (' Please input a GSAVE file name: ',$)
            READ(5,2000) n,(file(i),i=1,n)
      2000  FORMAT(Q,132A1)
            file(n+1)=0
            WRITE(5,3000) (file(i),i=1,n)
      3000  FORMAT(' GSAVE file: ',132A1)
      c
            END
```

```
c
        SUBROUTINE getpos(x,y,k)
        REAL x,y
        INTEGER k
c
10      CONTINUE
        CALL LOCATE (x,y,key)
        IF (key .eq. "177 ) GOTO 10
c
        CALL clrtxt
        WRITE (5,1000) "033,k,x,y
1000    FORMAT(' ',A,'[24;1fPosition ',
        1 I2,' is (',f6.1,',',f6.1,')' )
c
        RETURN
        END
c
c
```

This program could be used to generate various sized frames
interactively.

USING FORTRAN AND LINK OPTIONS WITH RGL/FEP (MC)

1.∅ PROBLEM: USING FORTRAN AND LINK OPTIONS WITH RGL/FEP

     The RGL/FEP subroutine library expects minimal run time I/O
support of at least 136 bytes. That means that when a programmer
compiles his application program (making calls to the subroutines in
RGL/FEP), the RECORD option of the FORTRAN compiler must be 136 or
more to support RGL/FEP I/O statements. Since the default maximum
record size may vary from compiler to compiler (depending upon how it
was generated), the programmer must be aware of this option and set it
if the default is not 136 or more. He may do so using the /RECORD:n
switch of the compiler.

     RGL/FEP provides a program development tool called RGLLNK.
RGLLNK.FOR generates the command stream that compiles, links and runs
a simple RGL/FEP application. RGLLNK.FOR currently uses the FORTRAN
default for I/O record lengths. This update changes RGLLNK so that it
explicitly specifys the record length. RGLLNK will then work
independently of the compiler's default.

     This is the fourth update to the RGL/FEP product. If you haven't
done so already, you must complete the first three updates in order
before you proceed.

     This problem appears in both RGL/FEP as a component of FEP and in
RGL/FEP as a separate package. Likewise, this update applies to both
products.

2.∅ SOLUTION

     To correct this problem, you must update the program RGLLNK.FOR.
Then you must recompile and relink it.

     The correction process is completed in three parts. The first
and third parts are common to all media. (They are described in
sections 2.2 and 2.6 respectively.) The first part sets up the update,
the third verifys the change. The second is media specific. You
could have received your RGL/FEP software in one of four ways. If you
received it on 8∅∅ bpi, 9 track magtape or on a hard disk (RK∅5,
RL∅1/2, ect.), then use the update procedure in section 2.3.   If you
received RGL/FEP on nine RX∅1's, then use the procedure in 2.4.
Otherwise, you must have received your software on four RX∅2's; use
section 2.5.

     If you received your software on a hard disk or on a magtape,
then you will need to create two files on your system volume:
59∅1∅4.COM and RGLLNK.∅∅1. Otherwise, you will need only create
RGLLNK.∅∅1. All the necessary files are listed in section 3.

     If your system is based on dual RX∅2 diskette drives, then it is
suggested that you create a system diskette with the following
utilities and drivers installed:

31

1. SWAP.SYS

2. RT11SJ.SYS (put this on the boot block too)

3. TT.SYS

4. DY.SYS

5. DUP.SAV

6. DIR.SAV

7. SLP.SAV

8. KED.SAV

9. MACRO.SAV

10. LIBR.SAV

11. STARTS.COM

12. FORTRA.SAV

It is also suggested that once you have built this system diskette
that you label it and reserve it exclusively for the correction of
your RGL/FEP software and store it along with your RGL/FEP
distribution kit (in some safe place).

2.1 Assumptions

     You must satify these conditions before continuing.

1. You must have a copy of the RGL/FEP distribution kit updated to
    the current patch level. This is the fourth update to the RGL/FEP
    package. You must have installed the first three updates before
    you may proceed.

2. You must have spare media of the same type and quantity as your
    current installation media. If you have a dual RXØ2 system then
    you will also need two extra RXØ2's (six total) -- one for your
    RT-11 system and one spare to update the RGL/FEP library. On the
    other hand, if your default device is a hard disk and you got
    RGL/FEP on RXØ2's, then you will only need five spare floppy
    diskettes. If your installation medium is a magtape, then you
    will need a spare hard disk.

3. You must have an RT-11 system, version 4.Ø which has the following
    utilities installed on the system volume: LINK, FORMAT, DUP, SLP,
    LIBR. You will also need to have some editor installed such as
    KED.

4. You must have installed a FORTRAN-IV compiler, version 2.5
    (FORTRA.SAV).

5. You have installed the FORTRAN Object Time System (FORLIB.OBJ)
    into SYSLIB.OBJ. (If not, you must make suitable changes when
    linking your FORTRAN programs).

6. You must know how to boot up your RT-11 system, how to load and
    unload the your system's devices and know how to operate a text
    editor.

7. It is assumed that these procedures will be run to completion
    without error. This includes the installation of RGL/FEP once you
    have updated your software. If you get an error in a given step,
    retry that step. If the error persists, you should check the
    appropriate manual. You may have to backtrack to preceding steps.
    As a precautionary measure, you should copy your current software.
    In that way you can always recover from any error by starting from
    the beginning.

2.2 Preparation For Patching

      These steps are common to all of the updates for the various
media.   Refer to the RT-11 System User's Guide, RGL/FEP Release Notes
and the FEP/RT Installation User's Guide for more information.

                                NOTE

    The notation dxn: stands for a device name of a hard disk
    drive. Examples are DLØ:, ⌐⌐2: or DL5:. The notation Mxn:
    stands for magtape. Examples are MT3:   and MMØ:.   The
    notation DYn: stands for an RXØ1/2 diskette, for example DYØ:
    or DY1:. Usually, this notation will apply to the RGL/FEP
    distribution volume.

1. Boot up your system.

2. Make sure you have the utilities necessary to complete the
    updating procedure (use the DIRectory utility). A list of minimal
    software to install RGL/FEP is given in the FEP/RT Installation
    and User's Guide.

3. Make sure you have the most current level of RGL/FEP.   You may
    check this with the Software Dispatch. If you find that your
    installation media are out of date, update them by applying each
    missing patch in order.

4. Copy your current installation media using COPY. For hard disk,
    you will need another hard disk. For magtape, you should squeeze
    the media to a hard disk (the same as when you installed RGL/FEP).
    Make sure your magtape drive is set to the 9-track, 800 BPI
    default. This may be set by typing:

            SET Mx: DENSE=800
            SET Mx: DEFALT=9

    Don't use a unit number with the set command, e.g. if you are
    using MM3:   type SET MM: DENSE=800 and so forth. For RX01's
    only, you must format the diskettes to single density before
    copying them.   You only need to copy diskettes 2/7 and 5/7. For
    RX02's, you need to copy 1/4, 2/4, and 3/4.

5. Using an editor (such as KED), create the files necessary to
    update the RGL/FEP library.   If you have RGL/FEP on magtape or
    hard disk, create 590104.COM and RGLLNK.001.   Otherwise, just
    create RGLLNK.001. Those files are listed in section 3.0.

2.3 * * * Procedure For Hard Disk, Magtape   * * *

     You will need approximately 100 free blocks on your system
volume.   It should take you about fifteen minutes to complete this
procedure; it is suggested that you do it in one sitting.   If you
have a magtape distribution and you have set up correctly, then you
have copied RGL/FEP onto a hard disk. If this is not the case, go
back to section 2.2.

1. Load your RGL/FEP disk into drive dxn:. If you have RGL/FEP on
    magtape, this disk will be the copy you made above.

2. Assign the logical name PAT to dxn:

            ASSIGN dxn: PAT:

3. Execute the patch by typing

            @590104

4. If your installation medium is magtape, reinitialize your magtape
    and copy the disk onto it:

            INIT/NOQUERY Mxn:
            COPY/NOLOG ddn: Mxn:/POS:-1

5. Verify the change using section 2.6.

2.4 * * * Correction Procedure For RXØ1 Distribution   * * *

        This procedure is specific to RXØ1 distribution kits. It assumes
your system device is a hard disk. You will need approximately 1ØØ
free blocks on your system disk. It should take you about a half hour
to complete this procedure; it is suggested that you do it all in one
sitting. Using section 2.2, you should have formated and copied the
appropriate diskettes. If you have not, refer to section 2.2 now.

1. Load the RGL/FEP distribution diskette labeled 6/7 into a free
    drive and they type:

            COPY DYn:RGLLNK.FOR DK:
            RUN SLP
            RGLLNK.FOR=RGLLNK.FOR,RGLLNK.ØØ1/A
            ^C
            DELETE/NOQ RGLLNK.BAK
            FORT/CODE:THR/REC:136 RGLLNK
            LINK RGLLNK
            RENAME/NOPROT DYn:RGLLNK.FOR DYn:
            COPY/PREDEL RGLLNK.FOR DYn:
            DELETE/NOQ RGLLNK.OBJ

2. Next load the RGL/FEP distribution diskette labeled 2/7 and type:

            RENAME/NOPROT DYn:RGLLNK.SAV DYn:
            COPY/PRED RGLLNK.SAV DYn:
            RENAME/PROT DYn:RGLLNK.SAV DYn:

3. Verify your software according to section 2.6.

2.5 * * * Correction Procedure For RXØ2 Systems * * *

        This procedure is specific to dual RXØ2 based systems. You will
need 1ØØ free blocks on your system diskette. It should take you
about a half hour to complete this procedure; it is suggested that
you finish it in one sitting. Using section 2.2, you should have
copied the appropriate diskettes. If you have not, refer to section
2.2 now.

1. Load the diskette labeled 3/4 into a free drive and they type:

            COPY DYn:RGLLNK.FOR DK:
            RUN SLP
            RGLLNK.FOR=RGLLNK.FOR,RGLLNK.ØØ1/A
            ^C
            DELETE/NOQ RGLLNK.BAK
            FORT/CODE:THR/REC:135 RGLLNK
            LINK RGLLNK

```
RENAME/NOPROT DYn:RGLLNK.FOR DYn:
COPY/PREDEL RGLLNK.FOR DYn:
DELETE/NOQ RGLLNK.OBJ
```

2. Next for diskettes 1/4 and 2/4, load each diskette and type
    (respectively)

```
RENAME/NOPROT DYn:RGLLNK.SAV DYn:
COPY/PRED RGLLNK.SAV DYn:
RENAME/PROT DYn:RGLLNK.SAV DYn:
```

3. Verify the change using section 2.6.

2.6 Verifying The Update

To check this software update, complete the following steps.

1. Reinstall the RGL/FEP subroutine library using the procedure used
    to install it initially (see the Release Notes: RGL/FEP for
    RT-11, AA-M521A-TC).   This   includes. running   the   RGL/FEP
    verification program RGLVFY.

2. You have now patched your RGL/FEP software.   Save the updated
    installation volume(s) in a safe place.

3.Ø NECESSARY FILES FOR THE PATCH

This section lists two files: an RT-11 indirect command file
59Ø1Ø4.COM and a SLP correction file RGLLNK.ØØ1.

36

3.1 590104.COM (a)

This command file is for hard disk and magtape distributions.
Note that ^C is two characters: ^ and C. This patch file assumes
that you have assigned PAT: to the correct device.

```
COPY PAT:RGLLNK.FOR DK:
RUN SLP
RGLLNK.FOR=RGLLNK.FOR,RGLLNK.001/A
^C
FORT/CODE:THR/REC:136 RGLLNK
LINK RGLLNK
DELETE/NOQ RGLLNK.OBJ
DELETE/NOQ RGLLNK.BAK
RENAME/NOPROT PAT:RGLLNK.FOR PAT:
COPY/PRED RGLLNK.FOR PAT:
RENAME/PROT PAT:RGLLNK.FOR PAT:
!* Done. Reinstall and verify.
```

3.2 SLP Correction Files

This section lists the SLP correction file RGLLNK.ØØ1. Note that this file must end with a carriage-return, linefeed (i.e. type RETURN at the end of the last line). The <tab> symbol stands for the TAB character; <cr> stands for carriage-return.

```
C*C*C<tab>Patch: 59.Ø1.Ø4<tab>Module: RGLLNK.FOR<tab>revision: ØØ1
-5Ø,5Ø
<tab>logical*1 FMT1(33)<tab>!* MGC ØØ1
-53,53
<tab>1'H','R','/','R','E','C',':','1','3','6','/',<tab>!* MGC ØØ1
<tab>2'L','I','S','T',':','L','P',':',' '/<tab>!* MGC ØØ1
-56,56
<tab>bldarr=23<tab>!* MGC ØØ1
-62,62
<tab>bldarr=32<tab>!* MGC ØØ1
-67,67
<tab>FMT1(3Ø)=answer(1)<tab>!* MGC ØØ1
-69,69
<tab>bldarr=29+i<tab>!* MGC ØØ1
/<cr>
```

DEVELOPING RGL/FEP APPLICATIONS (MC)

1.0   TUTORIAL: DEVELOPING RGL/FEP APPLICATIONS

     This article outlines various methods for developing RGL/FEP
applications and the options which must be present. It describes NO
changes to the RGL/FEP software.

     When developing RGL/FEP applications, a programmer must compile
his program, link it to the RGL/FEP library and run it. In general,
when he compiles a program, he can choose from twenty four different
compiler options to control how FORTRAN produces the object module.
He can specify, for example, whether arrays are to be vectored or not,
whether to produce a listing and how that listing will look.
Likewise, the linker also offers twenty four options which can
control, among other things, the generation of the executable image
and the format of the linker's listing. When compiling and linking a
program to the RGL/FEP library, the programmer is free to choose most
of the options. If he wishes, he can default most of them (which most
programmers are willing to do).   But there is one option that he
should not, in general, default, unless he is sure that the default is
compatable with the RGL/FEP library. He must be sure that the FORTRAN
compiler produces run time input/output code with a maximum record
length of 136 bytes or more. If one's compiler defaults the /RECORD
option to 135 (or more) then the default is acceptable. If not, the
programmer must explicitly reset the record length to 136 or more
using the /RECORD switch. To see what options FORTRAN has used in
producing code, specify both the /LIST and /HEADER options.

     RGL/FEP provides two program development utilities: RGLOVR and
RGLLNK.   With these, the programmer can compile, link and run his
application more easily. RGLOVR.COM is a command file which provides
the basic FORTRAN, LINK and RUN commands neccessary to produce an
executing RGL/FEP application using the RGL/FEP overlay scheme. As it
stands, RGLOVR.COM is NOT executable. It is a template which must be
edited. For example, suppose that a programmer wanted to build a
RGL/FEP application.   This application, say, reads data from a data
file (call it subroutine READ), plots it (subroutine PLOT) and then
prints it out to a printer (subroutine PRINT). Suppose he has already
written and compiled the three subroutines READ, PLOT and PRINT and
now keeps them on DL1: (which is not his system disk). Now suppose
he wants to put them all together in a program MAIN. He knows that he
must compile MAIN. He wants to get the fullest listing FORTRAN will
allow. To save space, he doesn't want his arrays to be vectored.   He
wants his integers to be I*4.   When he links his program to the
subroutines, he wants a map with the program sections listed in
alphabetical order and output to a file rather than the line printer.
Because his application is large, he will overlay the RGL/FEP library.
Finally, having built an executable image, he wants to run it.

     Instead of typing all the neccessary commands to compile and link
the program, he can create a command file for his program from
RGLOVR.COM. Starting with RGLOVR.COM which looks like this:

```
R LINK
MAINPR,MAINPR=MAINPR,BLODAT,PRMLIB/A/W/P:400/B:1200
```

```
       OV:LFIXED,OV:LOCAT2/O:1
       OV:LFREE,OV:LOCATE,OV:PPOINT/O:1
       OV:PDATA/O:1
       OV:DPAPER,OV:DEFALT/O:2
       OV:LNAXIS/O:2
       OV:LTAXIS/O:2
       OV:LNNICE,OV:LINMIN,OV:LINMAX/O:3
       OV:PRINUM,OV:PRISTR/O:3
       OV:FMINMX,OV:LGNICE/O:3
       OV:DPALOG/O:3
       OV:DPALIN/O:3//
       ^C
```

The programmer may copy it and edit it:

```
       COPY RGLOVR.COM MAIN.COM
       EDIT MAIN.COM
             .
             .
             .

       ASSIGN ddn: OV:
       @MAIN
```

Here, ddn: represents the volume where the overlay segments reside.
This resultant command file, specific to the program MAIN, will
compile, link and run MAIN with the stated options whenever it is
invoked:

```
       FORTRAN/I4/NOVECT/REC:136/SHOW:7/HEADER/STAT MAIN
       R LINK
       MAIN,MAIN=MAIN,DL1:READ,DL1:PLOT,/C
       DL1:PRINT,BLODAT,PRMLIB/A/W/P:400/B:1200
       OV:LFIXED,OV:LOCAT2/O:1
       OV:LFREE,OV:LOCATE,OV:PPOINT/O:1
       OV:PDATA/O:1
       OV:DPAPER,OV:DEFALT/O:2
       OV:LNAXIS/O:2
       OV:LTAXIS/O:2
       OV:LNNICE,OV:LINMIN,OV:LINMAX/O:3
       OV:PRINUM,OV:PRISTR/O:3
       OV:FMINMX,OV:LGNICE/O:3
       OV:DPALOG/O:3
       OV:DPALIN/O:3//
       ^C
       RUN MAIN
```


     The second method uses RGLLNK. If the programmer is in the very
special situation where the RGL/FEP software is on the current system
volume, the program, also on the system volume, doesn't need to be
linked to any external object modules (other than RGL/FEP ones) and
most of the compiler and linker options are to be defaulted, then he
may use RGLLNK.COM. He need only answer the questions and the command

procedure will do the rest. If however, he wishes to change the
command stream, say to link other, external modules or to use added
LINK options, then he must first run RGLLNK.SAV and edit the resultant
command file RGL.COM.

     Take the case above. The programmer runs RGLLNK.SAV to produce
RGL.COM. He then renames RGL.COM to MAIN.COM and edits it. He starts
with:

```
     FORT/CODE:THR MAIN
     R LINK
     MAIN=MAIN,BLODAT,PRMLIB/B:1200/A/W/P:1000//
     LFIXED,LOCAT2/O:1
     LFREE,LOCATE,PPOINT/O:1
     PDATA/O:1
     DPAPER,DEFALT/O:2
     LNAXIS/O:2
     LTAXIS/O:2
     LNNICE,LINMIN,LINMAX/O:3
     PRINUM,PRISTR/O:3
     FMINMX,LGNICE/O:3
     DPALOG/O:3
     DPALIN/O:3//
     ^C
     DEL MAIN.OBJ/NOQ
     RUN MAIN
     DEL MAIN.SAV
```

He wishes to do the same thing as before: compile the main program
with several options and link them to subroutines on DL1:. He
produces:

```
     FORTRAN/I4/NOVECT/REC:136/SHOW:7/HEADER/STAT MAIN
     R LINK
     MAIN,MAIN=MAIN,DL1:READ,DL1:PLOT,/C
     DL1:PRINT,BLODAT,PRMLIB/A/W/P:400/B:1200
     LFIXED,LOCAT2/O:1
     LFREE,LOCATE,PPOINT/O:1
     PDATA/O:1
     DPAPER,DEFALT/O:2
     LNAXIS/O:2
     LTAXIS/O:2
     LNNICE,LINMIN,LINMAX/O:3
     PRINUM,PRISTR/O:3
     FMINMX,LGNICE/O:3
     DPALOG/O:3
     DPALIN/O:3//
     ^C
     RUN MAIN
```

The only difference between this command procedure and the one
generated from RGLOVR.COM is that the overlay segments are assumed to
be on DK:, the system volume.

Choosing between the various methods is subject to taste. In the case where the RGL/FEP libraries, object modules and applications all reside on the same disk and the applications are all self-contained programs, then one may use RGLLNK.COM. In the case where added options are neccessary, but overlays are not, using RGLLNK.SAV to produce a command file and editing it is probably easiest. Finally, when one needs options and overlays, then copying and editing RGLOVR.COM is probably best.

RESTRICTIONS WITH GSAVE FILES (MC)

1.∅ PROBLEM: RESTRICTIONS WITH GSAVE FILES

    This article describes the nuances of the GSAVE-GCLOSE-GLOAD
subsystem in RGL/FEP. It reviews its use, explains it's intent and
describes two restrictions.

1.1 Tutorial: Saving VT125 REGIS Protocol In A File

    The RGL/FEP library is a collection of FORTRAN callable
subroutines that generate graphics for the VT125 graphics terminal.
The VT125 is driven by a graphics protocol known as ReGIS: Remote
Graphics Instruction Set. ReGIS instructions are sequences of ASCII
characters that define certain primative graphic objects (lines,
circles, curves) and their attributes (shading and color, line
patterns, ect.). The RGL/FEP subroutine library extends the
capability of the VT125 ReGIS based terminal. In particular, it
provides capabilities for transforming coordinates, plotting data and
interactively locating positions on the screen. It also provides an
environment to make the generation of graphics images easier.

    RGL/FEP also provides the capability of simultaneously generating
a display on the VT125 and saving the ReGIS strings that produced that
display in an ASCII file. The file can then be "played back" at a
later time from another RGL/FEP application. In this way, the file
can be considered to be like a macro (with no parameters). Whenever
it is needed to create a picture (or part of one), it can be displayed
using GLOAD. That picture will appear on the screen (usually) far
quicker than it would take to regenerate it. To create such a ReGIS
file, the user simply creates a RGL/FEP application program (in
FORTRAN) and debugs it until he gets the picture he wants. When he is
satisfied with the result, he simply inserts a GSAVE call (after the
initial call to INITGR) at the start of the application and a GCLOSE
call at the end. Most ReGIS commands generated by the program will be
saved in the designated file. For example:

        PROGRAM exampl
            .
            .
            .
        CALL INITGR(5)
        CALL GSAVE('later.reg',)
        CALL INITGR(5)   !* Optional - only if the programmer wants
            .   !* to reset attributes when he loads the file.

            .   !* All picture generating strings
            .   !* between GSAVE and GCLOSE
            .   !* go to file LATER.REG.
        CALL GCLOSE
        END

The programmer could later use the file LATER.REG in the following
way:

```
      PROGRAM useit
         .
         .
         .
      CALL INITGR(5)
         .
         .
         .
      !* Displays picture generated by program EXAMPL
      CALL GLOAD('later.reg', )
         .
         .
         .
      END
```

The GSAVE, GCLOSE and GLOAD subroutines were intended to allow the user to save certain commonly used pictures and quickly display them rather than constantly regenerating them.

GSAVE, however, does not generate all the ReGIS strings neccessary to recreate the picture outside of the RGL/FEP subroutine library. Specifically, if the user edits a file generated by GSAVE (adding the ReGIS prefix <esc>Pp and suffix <esc>) and TYPEs it directly to his VT125 terminal, he may get a result which is different then if he GLOADed the file from a RGL/FEP supported FORTRAN program. In general, RGL/FEP provides for certain graphics figures called "macrographs" to be saved inside the VT125 memory for later use. These macrographs are always available to RGL/FEP subroutines but are transparent to the user. If the user sends the ReGIS strings from a file to the VT125 terminal outside of RGL/FEP, then the macrographs are not defined and hence not available therefore the picture may appear different.

ReGIS text files are generated by RGL/FEP and are meant to be used by RGL/FEP. All other uses of those files may produce unexpected results without the environment that the RGL/FEP library provides.

RGL/FEP also provides the capability of locating various points on the VT125 screen. The programmer can query RGL/FEP about the current location (GETLOC), or can move a cursor around the screen interactively (LOCATE, LFIXED or LFREE). If the programmer makes a call to any of the cursor routines, he can not predict where a user will move the cursor (since these routines are interactive). In general, any call to LOCATE, LFIXED or LFREE could produce different cursor movements hence different ReGIS strings. Saving these strings in a text file would be useless: the cursor would no longer be interactive and would move in only one fixed pattern. Therefore, if the programmer calls any of the cursor routines while saving ReGIS strings in a file with GSAVE, the ReGIS strings associated with those routines will NOT be saved. In short, all ReGIS strings generated by calls to LOCATE, LFIXED and LFREE will NOT be saved in a GSAVE file.

This feature can be quite useful when generating a program to produce GSAVE files. Take the following (perhaps trivial) example: the program BOXSAV. BOXSAV will save the ReGIS strings neccessary to produce a box which might perhaps serve as a frame for later graphics. BOXSAV positions the cursor to the corners of the box and then draws a box. When the program GSAVE's the ReGIS instructions, it will save only the BOX command and not the cursor movement:


```
c
c
c       BOXSAV
c
c       This program produces frames for graphics interactively.
c
        PROGRAM boxsav
c
        LOGICAL*1 file(50)        !* string - name of GSAVE file
        REAL x1,y1,x2,y2          !* coordinates of the frame
c
        CALL initgr(5)
        CALL clrscr
        CALL clrtxt
c
        CALL linetx(1,1,
        1 'Interactive Frame Generation: BOXSAV')
c
        CALL getnam ( file )
        CALL gsave(file, )
c
        x1 = 0
        y1 = 0
        CALL getpos(x1,y1,1)
        x2 = x1
        y2 = y1
        CALL getpos(x2,y2,2)
        CALL box(x1,y1,x2,y2)
c
        CALL gclose
c
        END
c
        SUBROUTINE getnam ( file )
        LOGICAL*1 file(1)
c
        WRITE (5,1000)
1000    FORMAT (' Please input a GSAVE file name: ',$)
        READ(5,2000) n,(file(i),i=1,n)
2000    FORMAT(Q,132A1)
        file(n+1)=0
        WRITE(5,3000) (file(i),i=1,n)
3000    FORMAT(' GSAVE file: ',132A1)
c
        END
```

```
c
      SUBROUTINE getpos(x,y,k)
      REAL x,y
      INTEGER k
c
10    CONTINUE
      CALL LOCATE (x,y,key)
      IF (key .eq. "177 ) GOTO 10
c
      CALL clrtxt
      WRITE (5,1000) "033,k,x,y
1000  FORMAT(' ',A,'[24;1fPosition ',
     1 I2,' is (',f6.1,',',f6.1,')')' )
c
      RETURN
      END
c
c
```

This program could be used to generate various sized frames
interactively.

This is a complete listing of all articles for RT-11 V4.0 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!

Unassigned articles are indicated: UNASSIGNED.

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.

+ = Articles appeared in the RT-11 Software Dispatch Review, March 1980.

*The "Autopatch Kit" column in the list which follows indicates the first RT-11 V4.0 Autopatch Kit in which the associated patch was included. Unless otherwise indicated, the patches also appear in subsequent Autopatch Kits as well. Note that Autopatch Kit "G" is the latest kit available from the SDC.

Component	Autopatch Kit	Sequence	Mon/Yr
RT-11 V4.0			
MONITOR PATCHES			
ISSUING .SETTOP #-2 AND .EXIT UNDER XM MONITOR MAY			
CORRUPT SYSTEM DISK	A	1.1.1 M	Jul 80
IMPLEMENTING INTERNAL HANDLER QUEUEING IN FB AND XM MONITORS	A	1.1.2 M	Jul 80
ADDING HIGH SPEED RING BUFFER SUPPORT	A	1.1.3 M	Jul 80
CORRUPTION OF CSI TEXT UNDER XM MONITOR	A	1.1.4 M	Jul 80
MISSING COLON IN BOOT XX CAUSES SYSTEM HALT	A	1.1.5 M	Jul 80
TYPING ^U WHILE IN A ^X SEQUENCE UNDER A SYSTEM JOB	A	1.1.6 M	Sep 80
ABNORMAL TERMINATION OF FG JOB WHICH IS USING CSI	A	1.1.7 M	Nov 80
MISCELLANEOUS MRRT-11 BUGS	A	1.1.8 M	Nov 80
MRRT-11 MINIMAL FILE SUPPORT PROBLEM	A	1.1.9 M	Nov 80
INCORRECT LIMIT CHECKS ON PRIVILEGED BACKGROUND JOBS USING			
VIRTUAL OVERLAYS	A	1.1.10 M	Nov 80
MULTI-TERMINAL MONITORS DON'T ALWAYS PROCESS CTRL/F PROPERLY	A	1.1.11 M	Nov 80
MONITOR CHANGES AND CORRECTIONS	A	1.1.12 M	Dec 80
MONITOR CORRECTIONS	B	1.1.13 M	Jan 81
MONITOR UPDATES	B	1.1.14 M	Feb 81
ABORT I/O IN PROGRESS HANDLER BIT	B	1.1.15 M	Apr 81
CORRECTIONS FOR DISTRIBUTED AND SYSTEM GENERATED MONITORS	C	1.1.16 M	Jun 81
PRINT COMMAND RESTRICTION		1.1.17 R	Jul 81
UPDATES TO MONITOR FILES	D	1.1.18 M	Oct 81
CORRECTIONS TO THE MONITOR	E	1.1.19 M	Jan 82
MONITOR NOTES			
COMPLETION ROUTINE OPERATION UNDER THE SJ MONITOR		1.2.1 N	Sep 82

Component	Autopatch Kit	Sequence	Mon/Yr
DEVICE HANDLER SOURCES			
DEVICE HANDLER NOTES			
RL02s AT REV. LEVEL "F" FAIL DURING RT-11 SYSGEN		6.1.1 N	Oct 80
DD.MAC			
DD PRIMARY BOOTSTRAP PROBLEM	A	6.4.1 M	Jul 80
DL.MAC			
PATCH XM VERSION OF DL HANDLER .SPFUN GET SIZE ROUTINE	A	6.5.1 M	Dec 80
ERRORS ON RL01 DISK DRIVES AFTER DISK PACKS ARE CHANGED	B	6.5.2 M	Jan 81
DM.MAC			
ERRORS IN DM OFFSET POSITIONING AND ERROR LOGGING	A	6.6.1 M	Jul 80
DY.MAC			
DELETED DATA MARK MAY BE LOST IF BUFFER STARTS ON PAR BOUNDARY	D	6.11.1 M	Aug 81
ERROR LOGGING SUPPORT FOR DY		6.11.2 M	Oct 82
LP.MAC			
LP SET NOHANG MAY CRASH SYSTEM	A	6.12.1 M	Sep 80
LS.MAC			
LS SET NOHANG MAY CRASH SYSTEM	A	6.13.1 M	Sep 80
PROBLEMS WITH LS HANDLER	B	6.13.2 M	Jan 81
USING AN LA120 TERMINAL AS A LINE PRINTER WITH THE LS HANDLER		6.13.3 N	Jul 81
SET LS NOHANG IS CURRENTLY INOPERATIVE	C	6.13.4 M	Jul 81
RACE CONDITION IN LS HANDLER	D	6.13.5 M	Aug 81
LS HANDLER SET "NOHANG" PROBLEM	E	6.13.6 M	Jan 82
PROBLEMS WITH LS HANDLER		6.13.7 M	Oct 82
PD.MAC			
CORRECTION TO PDT ERROR LOGGING SUPPORT	B	6.16.1 M	Apr 81
MAG TAPE HANDLERS			
BUFFER CLEARING ON SHORT READ IN XM MONITOR	A	6.20.1 M	Jul 80
LINKING AN XM, NON-FILESTRUCTURED TS HANDLER GENERATES			
AN UNDEFINED GLOBAL	A	6.20.2 M	Aug 80
INCORRECT READ ERROR RECOVERY IN MT HANDLER	A	6.20.3 M	Sep 80
TS-11 DOES NOT RECOVER FROM SOFT ERROR ON WRITE EOF	C	6.20.4 M	Jul 81
SYSTEM UTILITIES			
PIP.SAV			
ERRORS IN PIP	A	7.1.1 M	Sep 80
COPY/PREDELETE COMMAND		7.1.2 N	Sep 80
MATCHING FILE SPECIFICATIONS ERRORS	B	7.1.3 M	Feb 81
COPY/BINARY/WAIT AND LOG HEADER PROBLEMS	B	7.1.4 M	Apr 81
COPY/PREDELETE AND COPY/NOREPLACE WORK INCORRECTLY WITH /WAIT	C	7.1.5 M	Jun 81
ERROR WITH RENAME/NOREPLACE	C	7.1.6 M	Jul 81
/POSITION:N SWITCH FOR MAGTAPE INPUT WORKS INCORRECTLY	D	7.1.7 M	Oct 81
COPY/BINARY STOPS PROCESSING AFTER ENCOUNTERING AN OBJ LIBRARY FILE	E	7.1.8 M	Nov 81
COPYING FILES TO UNINITIALIZED DISKS		7.1.9 N	Nov 81
ALLOCATE AND DELETE WORK INCORRECTLY WITH COPY OPERATIONS	F	7.1.10 M	Feb 82
DUP.SAV			
MISSING COLON IN BOOT XX CAUSES SYSTEM HALT	A	7.2.1 M	Jul 80
SQUEEZE CREATES <UNUSED> ENTRIES OF LENGTH ZERO BEFORE			
.BAD FILES	A	7.2.2 M	Aug 80
PROBLEMS WITH COPY/DEVICE AND INITIALIZE	A	7.2.3 M	Dec 80
BOOTSTRAPPING AN UNPATCHED MONITOR FROM A PATCHED SYSTEM	B	7.2.4 N	Jan 81
.SPFUN RETURN BUFFER PROCESSED INCORRECTLY FOR RK06/7	B	7.2.5 M	Jan 81
USE OF INITIALIZE/RESTORE ON MEDIA SUPPORTING BAD			
BLOCK REPLACEMENT		7.2.6 N	May 81
PROBLEMS WITH INIT/BAD AND COPY/DEVICE	C	7.2.7 M	May 81
PROBLEMS WITH INITIALIZE COMMAND	C	7.2.8 M	Jun 81
ATTEMPT TO RESTORE UNCLOSED TENTATIVE FILES FAILS	C	7.2.9 M	Jul 81
/V WITH NO DEVICE SPECIFICATION GIVES WRONG ERROR MESSAGE	D	7.2.10 M	Sep 81
OUTPUT ERROR DURING COPY/DEVICE TO MAGTAPE CAUSES SYSTEM ERROR	E	7.2.11 M	Oct 81
USE OF COPY/DEV/FILE WITHOUT FILE SPECIFICATION	E	7.2.12 M	Nov 81
PROBLEMS WITH COPY/DEVICE USING /END	F	7.2.13 M	Apr 82

Component	Autopatch Kit	Sequence	Mon/Yr
DIR.SAV			
DIR/OUT COMMAND PRODUCES DEVICE NOT ACTIVE MESSAGE	A	7.3.1 M	Jul 80
DIR/VOL GIVES ?MON-F-TRAP TO 4	A	7.3.2 M	Dec 80
LOSS OF LAST PRINT CHARACTER IN DIRECTORY LISTING	D	7.3.3 M	Sep 81
RESORC.SAV			
RESORC MAY REPORT INCORRECT JOB NAMES ON A SHOW JOBS COMMAND	A	7.5.1 M	Aug 80
ADD CIS DETECTION CAPABILITY TO RESORC	B	7.5.2 M	May 81
PROBLEM WITH IDENTIFYING 11/23 PROCESSOR	D	7.5.3 M	Sep 81
LINK.SAV			
LINK BYTE RELOCATION AND DIRECTORY SIZE	A	7.9.1 M	Jul 80
LINK MAP PROCESSING ERROR	A	7.9.2 M	Aug 80
LINK MAP ERROR AND MULTIPLE DEFINITION LIBRARIES	A	7.9.3 M	Oct 80
RT-11 V4 LINKER RESTRICTION	B	7.9.4 R	Jan 81
LINK TRANSFER ADDRESS CALCULATION BUGS	B	7.9.5 M	Mar 81
LINK ADDITIONS AND CORRECTIONS	D	7.9.6 M	Aug 81
LINK UPGRADE	E	7.9.7 M	Nov 81
LINK ERROR IN LIBRARY MODULE TRANSFER ADDRESS PROCESSING	E	7.9.8 M	Jan 82
LINK LIBRARY MODULE PLACEMENT ERROR	E	7.9.9 M	Jan 82
LINK MULTIPLE ERROR FIXES	G	7.9.10 M	May 82
LINK REFERENCES ILLEGAL ADDRESS		7.9.11 M	Oct 82
LIBR.SAV			
A LIBR COMMAND WITH NO FILE-SPEC CAN CAUSE A SYSTEM CRASH	A	7.10.1 M	Jul 80
LIBR ERRORS	C	7.10.2 M	Jul 81
LIBR CORRUPTS FORM LIBRARY DIRECTORY	C	7.10.3 M	Jun 81
LIBR ERROR IN GENERATING ENTRY POINT TABLE	E	7.10.4 M	Jan 82
LIBR RESTRICTION		7.10.5 N	Jan 82
FILEX.SAV			
FILEX WILDCARD TRANSFERS CAUSE MONITOR TRAP	A	7.11.1 M	Aug 80
FILEX CREATES ZERO FILLED INTERCHANGE RECORDS	A	7.11.2 M	Sep 80
SIZE CALCULATION PROBLEM IN FILEX	D	7.11.3 M	Aug 81
RECORDS DROPPED BY FILEX	D	7.11.4 M	Sep 81
SRCCOM.SAV			
COMPARING TWO FILES MAY CAUSE TRAP TO 4	A	7.12.1 M	Aug 80
BLANK LINE COMPARISON FOR SLIDING MATCH	A	7.12.2 M	Dec 80
BINCOM.SAV			
BINCOM GENERATES ERRONEOUS ERROR MESSAGE	B	7.13.1 M	Apr 81
ERRONEOUS DOUBLE PRECISION CALCULATION IN BINCOM	C	7.13.2 M	Jun 81
BINCOM PLACES TAB CHARACTER AFTER OFFSET IN SIPP COMMAND FILE	E	7.13.3 M	Jan 82
DUMP.SAV			
BLOCK NUMBERS OUTPUT FROM DUMP	D	7.14.1 M	Aug 81
SLP.SAV			
TERMINATION OF PATCHING SESSION WITH SLP FATAL ERRORS	A	7.15.1 M	Nov 80
SLP GENERATES FATAL ERROR TRAP	B	7.15.2 M	Jan 81
SLP ERROR	B	7.15.3 M	Mar 81
SIPP.SAV			
CORRUPTION OF MULTI-BLOCK LOG FILES	A	7.16.1 M	Jul 80
PAT.SAV			
USE OF THE PAT UTILITY WITH RT-11 V3B PATCHES		7.17.1 N+	Mar 80
HELP.SAV			
PROBLEMS WITH HELP UTILITY	A	7.19.1 M	Nov 80
EDIT.SAV			
EDIT MISHANDLES OUTPUT FILE FULL ERROR	B	7.20.1 M	Nov 81
SYSTEM SUBROUTINE LIBRARY (SYSLIB)			
SYSLIB.OBJ			
PATCH TO ICSI	A	8.1.1 M	Oct 80
IASIGN REDEFINITIONS	A	8.1.2 M	Oct 80

Component	Autopatch Kit	Sequence	Mon/Yr
ILUN RESTRICTION		8.1.3 R	Feb 81
VIRTUAL OVERLAY HANDLER CORRECTION	E	8.1.4 M	Feb 82
SYSTEM MACRO LIBRARY			
.SPFUN PROGRAMMED REQUEST	A	9.1.1 M	Dec 80
ABORT I/O PROGRESS SUPPORT FOR SYSMAC	B	9.1.2 M	Apr 81
.CMKT PROGRAMMED REQUEST	C	9.1.3 M	Jun 81
INCORRECT EXPANSION OF .DRSET MACRO	F	9.1.4 M	Apr 82
SYSTEM GENERATION PACKAGE			
SYSGEN CREATES ONE MORE DEVICE SLOT THAN REQUESTED	A	10.3.1 M	Dec 80
ASSEMBLY ERROR AFTER SYSGEN	B	10.3.2 M	Mar 81
TERMINAL OUTPUT CORRUPTION ON DZ11 OR DZV11 LINES	F	10.3.3 M	Apr 82
DOCUMENTATION			
RT-11 SYSTEM RELEASE NOTES			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.2.1 N	Jul 80
DOCUMENTATION CORRECTIONS		11.2.2 N	Aug 80
CHANGES TO DUP /I OPTION		11.2.3 N	Apr 81
INCORRECT DUP CUSTOMIZATION PATCHES		11.2.4 N	Sep 81
RT-11 INSTALLATION AND SYSTEM GENERATION GUIDE			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.3.1 N	Jul 80
CORRECTION TO AN OPTIONAL PATCH TO LINK		11.3.2 N	Aug 80
DOCUMENTATION ERROR: REFERENCE TO RL02 OMITTED FROM SYSGEN DIALOGUE		11.3.3 N	Oct 80
INCORRECT LINK MAPS FOR DISTRIBUTED MONITORS		11.3.4 N	Dec 80
INCORRECT PATCH FOR CHANGING QUEUE WORK FILE SIZE		11.3.5 N	Dec 80
CHANGING DEFAULT NUMBER OF DIRECTORY SEGMENTS		11.3.6 N	Apr 81
INTRODUCTION TO RT-11			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.4.1 N	Jul 80
RT-11 SYSTEM USER'S GUIDE			
RT-11 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.5.1 N	Jul 80
CORRECTIONS TO SLP CHAPTER: RT-11 SYSTEM USER'S GUIDE		11.5.2 N	Oct 80
DIFFERENCES BETWEEN DEVICE COPYING COMMANDS		11.5.3 N	Dec 80
RT-11 SYSTEM MESSAGE MANUAL			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.6.1 N	Jul 80
CORRECTIONS TO SLP MESSAGES IN "RT-11 SYSTEM MESSAGE MANUAL"		11.6.2 N	Nov 80
NEW SLP ERROR MESSAGE		11.6.3 N	Feb 81
PIP ERROR MESSAGES MISSING		11.6.4 N	Oct 81
RT-11 POCKET GUIDE			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.7.1 N	Jul 80
RT-11 PROGRAMMER'S REFERENCE MANUAL			
DOCUMENTATION CORRECTIONS		11.8.1 N	Sep 80
INCORRECT PROGRAMMED REQUEST EXAMPLES		11.8.2 N	Mar 81
UNDOCUMENTED .SERR ERROR CODE		11.8.3 N	Dec 81
RT-11 SOFTWARE SUPPORT MANUAL			
RT-11 V4.0 DOCUMENTATION CORRECTIONS AND ADDITIONS		11.9.1 N	Jul 80
SOFTWARE SUPPORT MANUAL CORRECTION		11.9.2 N	Jun 81
ERROR IN DESCRIPTION OF .DRSET MACRO		11.9.3 N	Sep 81
DEBUGGING UTILITIES			
VDT.OBJ			
NOTES ON USING ODT OR VDT IN AN XM ENVIRONMENT		12.2.1 N	Jan 81
ERROR STATUS NOT SAVED/RESTORED BY VDT		12.2.2 M	Oct 82
ERROR CONTROL PACKAGE			
ERROUT.MAC			
ERROR LOGGING SUPPORT OF USER-WRITTEN HANDLERS	G	14.6.1 M	May 82
BATCH PACKAGE			
BATCH.SAV			
PATCH BATCH TO USE MONITOR SUFFIX	A	15.1.1 M	Oct 80
BATCH $CREATE IGNORES BLANK LINES		15.1.2 M	Aug 82

SPOOLING PACKAGE
QUEUE.REL

Component	Autopatch Kit	Sequence	Mon/Yr
SUPERFLUOUS LINEFEED FROM QUEUE	B	16.1.1 M	Mar 81
NARROW BANNER PAGES FROM QUEUE	C	16.1.2 F	May 81
/R FOLLOWING /S IF NO OUPTUT QUEUED MAY CAUSE FATAL ERROR IN QUEUE	D	16.1.3 M	Aug 81
ATTEMPTING TO COMMUNICATE WITH 'QUEUE' FROM A VIRTUAL JOB		16.1.4 N	Apr 82
QUEUE MAY INDICATE INCORRECT NUMBER OF COPIES ON BANNER PAGES		16.1.5 M	Sep 82

QUEMAN.SAV

Component	Autopatch Kit	Sequence	Mon/Yr
PROBLEMS WITH QUEMAN	B	16.2.1 M	Jan 81

KEYPAD EDITOR
KED

Component	Autopatch Kit	Sequence	Mon/Yr
MAKE TERMINAL SETUP OPTIONAL IF MTATCH FAILS	A	17.1.1 F	Aug 80
PROVIDE A .CHAIN INTERFACE FOR KED	A	17.1.2 F	Aug 80
PROVIDE REASONABLE ACTIONS AND ERROR MESSAGES WHEN DEALING WITH DEGENERATE FILES	A	17.1.3 M	Oct 80
SEARCH FAILS IF TARGET IF FIRST OR LAST STRING IN THE FILE	A	17.1.4 M	Nov 80
KNOWN ERRORS AND RESTRICTIONS		17.1.5 R	Dec 80
"SET SEARCH EXACT JUNK" COMMAND CRASHES KED	C	17.1.6 M	Jul 81
REPEATED USE OF THE "APPEND" FUNCTION CRASHES KED	C	17.1.7 M	Dec 81
DISABLE REVERSE VIDEO DISPLAY BY KED	E	17.1.8 F	Jul 81
FILE SAMPLE.KED OMITTED FROM DISTRIBUTION		17.1.9 N	Aug 81
KED DOCUMENTATION CORRECTION		17.1.10 N	Nov 81

K52

Component	Autopatch Kit	Sequence	Mon/Yr
MAKE TERMINAL SETUP OPTIONAL IF MTATCH FAILS	A	17.2.1 F	Aug 80
PROVIDE A .CHAIN INTERFACE FOR K52	A	17.2.2 F	Aug 80
PROVIDE REASONABLE ACTIONS AND ERROR MESSAGES WHEN DEALING WITH DEGENERATE FILES	A	17.2.3 M	Oct 80
SEARCH FAILS IF TARGET IS FIRST OR LAST STRING IN THE FILE	A	17.2.4 M	Nov 80
KNOWN ERRORS AND RESTRICTIONS		17.2.5 R	Dec 80
"SET SEARCH EXACT JUNK" COMMAND CRASHES K52	C	17.2.6 M	Jul 81
REPEATED USE OF THE "APPEND" FUNCTION CRASHES K52	E	17.2.7 M	Dec 81
NO EQUIVALENT PATCH FOR K52 FOR SEQ 17.1.8		17.2.8 N	Aug 81
FILE SAMPLE.KED OMITTED FROM DISTRIBUTION		17.2.9 N	Aug 81
KED DOCUMENTATION CORRECTION		17.2.10 N	Dec 81

AUTOMATED PATCHING FACILITY PACKAGE
PACKAGE NOTES

Component	Autopatch Kit	Sequence	Mon/Yr
AUTOPATCH SERVICE FOR RT-11		19.1.1 N	Jun 81

FMS-11/RT-11 V1.1

Component	Autopatch Kit	Sequence	Mon/Yr
ANNOUNCING FMS-11/RT-11 V1.1		33.1 N	Aug 80

FRED V1.1

Component	Autopatch Kit	Sequence	Mon/Yr
ZERO IMPURE AREA SIZE PROBLEM		33.3.1 M	Sep 81

BASIC-11/RT-11 V2.0

INTERPRETER

Component	Autopatch Kit	Sequence	Mon/Yr
REPUBLICATION OF PATCHES		35.1.1 N+	Mar 80
PRINT USING - PATCH A	A	35.1.2 M+	Mar 80
RESEQ - PATCH B	A	35.1.3 M+	Mar 80
EDITING A DIM #n STATEMENT - PATCH C	A	35.1.4 M+	Mar 80
DOUBLE PRECISION HANG - PATCH D	A	35.1.5 M+	Mar 80
SAVE dev: AND REPLACE dev: - PATCH E	A	35.1.6 M+	Mar 80
SINGLE PRECISION HANG AND NUMERIC CONVERSION PROBLEM - PATCH F	A	35.1.7 M+	Mar 80
SAVE .XXX & UNSAVE .XXX - PATCH G	A	35.1.8 M+	Mar 80
NEW - PATCH H	A	35.1.9 M+	Mar 80
RESEQ - PATCH I	A	35.1.10 M+	Mar 80
LISTNH / OLD - PATCH J	A	35.1.11 M+	Mar 80
SYS(1) - PATCH K	A	35.1.12 M+	Mar 80
CALL - PATCH L	A	35.1.13 M+	Mar 80
DOUBLE PRECISION INTEGER VARIABLES - PATCH M	A	35.1.14 M+	Mar 80
FILESIZE 0 - PATCH N	A	35.1.15 M+	Mar 80

Component	Autopatch Kit	Sequence	Mon/Yr
INTEGERS IN DOUBLE PRECISION BASIC-11		35.1.16 N+	Mar 80
REM STATEMENTS ON MULTI-STATEMENT LINES - PATCH O	A	35.1.17 M+	Mar 80
INT FUNCTION - PATCH P FOR SINGLE USER BASIC-11	A	35.1.18 M	Nov 80
RETRACTED		35.1.19 M	May 81
PRINT USING - PATCH R FOR SINGLE USER BASIC-11	B	35.1.20 M	Jan 81
OMITTING TRIG FUNCTIONS FROM BASIC-11	B	35.1.21 N	Jan 81
STRING CONCATENATION - PATCH S FOR SINGLE USER BASIC-11	B	35.1.22 M	Mar 81
PROBLEM WITH BASIC-11 PATCH Q		35.1.23 N	May 81
INTEGER COMPARISON - PATCH T FOR SINGLE USER BASIC-11 (REFERENCE SEQ 35.1.26 M)		35.1.24 M	Sep 82
PASSING STRING ARGUMENTS TO ALRs - PATCH U FOR SINGLE USER BASIC-11		35.1.25 M	Sep 82
REVISION TO PATCH "T" FOR SINGLE USER BASIC-11 (REFERENCE SEQ 35.1.24 M)		35.1.26 M	Oct 82
USING "CHAIN" WITH "COMMON" MAY CAUSE SYSTEM CRASH - PATCH "V" FOR BASIC-11		35.1.27 M	Oct 82
UTILITIES			
CONVERSION PROGRAM		35.2.1 M+	Mar 80
BASIC-11/RT-11 V2 CONVERSION PROGRAM PATCH 1		35.2.2 M+	Mar 80
DOCUMENTATION			
OVERLAYING WHILE IN A SUBROUTINE		35.3.1 R+	Mar 80
OPERATION OF CTRLC, RCTRLC AND SYS(6) FUNCTIONS AND THE CTRL/C COMMAND		35.3.2 N+	Mar 80
OPERATION OF OLD, RUN, CHAIN, AND OVERLAY WHEN THE SPECIFIED FILE IS NOT FOUND		35.3.3 N+	Mar 80
CREATING AND ACCESSING VIRTUAL ARRAY FILES		35.3.4 N+	Mar 80
STORAGE OF THE NULL CHARACTER IN STRING VARIABLES AND VIRTUAL STRING ARRAYS		35.3.5 N+	Mar 80
USE OF COMPILE COMMAND		35.3.6 N+	Mar 80
STRING MANIPULATION IN ASSEMBLY LANGUAGE ROUTINES		35.3.7 N+	Mar 80
MAXIMUM ARRAY SUBSCRIPT SIZE		35.3.8 N+	Mar 80
NEW MANUAL AVAILABLE FOR BASIC-11/RT-11		35.3.9 N	May 81

MicroPower/Pascal V1.0

Component	Autopatch Kit	Sequence	Mon/Yr
ANNOUNCING MICROPOWER/PASCAL V1.0		37.1.1 N	Apr 82
BUILDING AN APPLICATION THAT USES THE FILE SYSTEM		37.1.2 M	May 82
COMPILER			
INCORRECT PASCAL DEFINITION OF THE PCB RECORD		37.5.1.7 N	Nov 82
CHANGE IN DATA TYPE STRUCTURE_DESC IN PASCAL INCLUDE FILE		37.5.1.8 N	Nov 82

MicroPower/Pascal V1.1

Component	Autopatch Kit	Sequence	Mon/Yr
MISCELLANEOUS NOTES			
ANNOUNCING MicroPower/Pascal V1.1		37.1.1.1 N	Sep 82
MIB			
MIB MAY GIVE A HARDWARE READ ERROR DURING KERNAL INSTALLATION		37.3.3.1 N	Aug 82
PAXM/PAXU/KERNAL			
SERA REQUEST FOR DISCONNECT MAY FAIL IN A MAPPED SYSTEM		37.4.1.1 N	Aug 82
ILLEGAL ADDRESS ARGUMENT CAN CAUSE UNPREDICTABLE RESULTS		37.4.1.2 N	Aug 82
DISPATCH TO UNMAPPED STACK OVERFLOW EXCEPTION IS INCORRECT		37.4.1.3 N	Aug 82
STOPPED PROCESSES ARE PLACED IN THE INACTIVE QUEUE		37.4.1.4 N	Aug 82
PROCESS ON INACTIVE QUEUE DOES NOT HAVE POINTER TO EXCEPTION FRAME		37.4.1.5 N	Aug 82
DISCONNECT FROM INTERRUPT REQUEST MAY CORRUPT KERNEL FREE POOL		37.4.1.6 N	Sep 82
MULTIPLE EXCEPTIONS IN A PROCESS CAN CAUSE UNPREDICTABLE RESULTS		37.4.1.7 N	Sep 82
UNMAPPED, MIXED ROM/RAM POWER-UP IS INCORRECT		37.4.1.8 N	Nov 82
PASCAL COMPILER			
CONFORMANT ARRAYS AND SINGLE CHARACTER LITERALS		37.5.1.1 N	Aug 82
FORMAL PARAMETER LISTS WITH DEFAULT VALUES		37.5.1.2 N	Aug 82
ATTRIBUTE [CONTEXT(MMU)] DOES NOT WORK		37.5.1.3 N	Aug 82
ACCESSING UP-LEVEL LOCAL VARIABLES FROM [TERMINATE] PROCEDURES		37.5.1.4 N	Aug 82
CALLING THE ROUND (OR TRUNC, UROUND, UTRUNC) FUNCTION WITH NON-STATIC VARIABLES		37.5.1.5 N	Aug 82
INCORRECT CODE GENERATED FOR STRUCTURED FUNCTION RESULTS		37.5.1.6 N	Nov 82

Component	Autopatch Kit	Sequence	Mon/Yr

OTS
KEF-11 FLOATING POINT STATUS WORD IS INCORRECTLY INITIALIZED — 37.6.1.1 N — Aug 82
THE NATURAL LOG FUNCTION RETURNS INCORRECT RESULTS — 37.6.1.2 N — Aug 82
UROUND AND UTRUNC FUNCTIONS DO NOT WORK CORRECTLY — 37.6.1.3 N — Nov 82
PACK AND UNPACK FUNCTIONS ARE MISSING FROM THE OTS LIBRARY — 37.6.1.4 N — Nov 82

XL (SERIAL LINE) DRIVER
ERROR IN "DISCONNECT TRANSMIT RING BUFFER" FUNCTION — 37.8.1.1 N — Aug 82
BLOCK MODE READ REQUEST RETURNS INCORRECT DATA — 37.8.1.2 N — Aug 82

DOCUMENTATION
RENAMING LIBXXX.OBJ TO SYSLIB IS NO LONGER RECOMMENDED — 37.10.1.1 N — Sep 82
RENAMING COMM.SML OR COMU.SML TO SYSMAC.SML IS NO LONGER RECOMMENDED — 37.10.1.2 N — Sep 82
PROCESSOR JUMPERS FOR POWER-UP MODES SHOWN ON PAGE 1-4 OF THE
  MicroPower/Pascal INSTALLATION GUIDE ARE INCORRECT — 37.10.1.3 N — Sep 82
MicroPower/Pascal V1.0 SYSTEM USER'S GUIDE DOES NOT GIVE FULL
  INFORMATION ON THE LINDF$ MACRO FIELDS — 37.10.1.4 N — Sep 82
DLV-11 PREFIX MODULE EXAMPLE SHOWS INCORRECT CSR — 37.10.1.5 N — Sep 82

MU BASIC-11/RT V2.1

INTERPRETER
MU BASIC V2.1 MAINTENANCE RELEASE AVAILABLE — — Mar 82
UNWARRANTED ISSUANCE OF "TOO MANY CHANNELS" ERROR - PATCH A FOR
  MULTI-USER BASIC-11 — 38.1.1 M — Jul 82
"ERR" VALUE IMPROPERLY UPDATED WHEN USING "ON ERROR GOTO nnnnn" -
  PATCH B TO MULTI-USER BASIC-11 — 38.1.2 M — Jul 82
"RESEQ" FOLLOWING "DEL nnnnn" RESULTS IN "Mon-F-Trap to 10 000002" -
  PATCH C TO MULTI-USER BASIC-11 — 38.1.3 M — Jul 82
PROGRAMS RETRIEVED USING "OLD filename" OR "RUN filename"
  ARE SOMETIMES CORRUPTED - PATCH "D" FOR MU BASIC-11 — 38.1.4 M — Sep 82
IMPROPER FILE EXTENSION CREATED FOR COMPILED FILES WHEN MU BASIC-11
  IS CONFIGURED FOR DOUBLE-PRECISION - PATCH "E" FOR MU BASIC-11 — 38.1.5 M — Sep 82
REVISION TO PATCH "F" FOR MULTI-USER BASIC-11 — 38.1.6 M — Oct 82
PROBLEMS DEASSIGNING PREVIOUSLY ASSIGNED TERMINAL -
  PATCH "G" FOR MU BASIC-11 — 38.1.7 M — Oct 82

FORTRAN IV/RT-11 V2.5

COMPILER
ANNOUNCING PDP-11 FORTRAN IV/RT-11 V2.5 — — 45.1.1 N — Sep 80
THE COMPILER INCORRECTLY PARSES SOME EXPRESSIONS IN I/O LISTS — A — 45.1.2 M — Nov 80
THE COMPILER INCORRECTLY CONVERTS INTEGER TO BYTE IN
  LOGICAL EXPRESSIONS — A — 45.1.3 M — Nov 80
THE COMPILER GENERATES INCORRECT CODE FOR EQUIVALENCED ARRAYS
  (PAT 12) — D — 45.1.4 M — Sep 81
THE COMPILER INCORRECTLY INTERPRETS COMMENTS WITH TABS (PAT 17) — E — 45.1.5 M — Nov 81
MISSING END IN MAIN PROGRAM CAN CAUSE COMPILER CRASH (PAT 18) — E — 45.1.6 M — Nov 81
THE COMPILER INCORRECTLY OPTIMIZES ARRAY ELEMENTS PASSED AS
  ARGUMENTS (PAT 20) — E — 45.1.7 M — Dec 81
THE COMPILER INCORRECTLY PARSES PARENTHESES IN QUOTED STRINGS
  (PAT 21) — E — 45.1.8 M — Jan 82
THE COMPILER CRASHES WHILE ACCESSING AN ODD ADDRESS IN PAT 12
  (PAT 22) — E — 45.1.9 M — Jan 82
CORRECTION FOR CONTINUATION LINES PRECEEDED BY COMMENTS (PAT 27) — F — 45.1.10 M — Apr 82
BOUNDS CHECKING OF INTERNAL BUFFER IN OPTIMIZER (PAT 29) — G — 45.1.11 M — Jun 82
COMPILER HANGS WHEN ERRORS OCCUR IN STATEMENT FUNCTIONS (PAT 31) — G — 45.1.12 M — Jun 82
INCORRECT BYTE TO INTEGER CONVERSION — — 45.1.13 M — Aug 82
COMPILER GENERATES FATAL ERROR IN REGISTER ALLOCATOR — — 45.1.14 M — Aug 82

OTS
THE OTS DOES NOT SET DEFAULT CARRIAGE CONTROL FOR SERIAL
  LINE PRINTER — B — 45.2.1 M — Jan 81
THE LUN IS NOT SAVED WHEN AN ERROR OCCURS WHILE OPENING A FILE — B — 45.2.2 M — Jul 81
PATCH TO ALLOW THE PLACEMENT OF THE FORTRAN OTS WORK AREA
  BETWEEN THE PROGRAM'S HIGH LIMIT AND THE BASE OF THE FIRST
  VIRTUAL OVERLAY FOR PRIVILEGED FORTRAN JOBS — B — 45.2.3 F — Feb 81
BOUNDARY CONDITION ON FORMATTED I/O CORRUPTS I/O (PAT 6) — B — 45.2.4 M — Mar 81

Component	Autopatch Kit	Sequence	Mon/Yr
DEFAULT CARRIAGE CONTROL FOR IMPLIED SEQUENTIAL ACCESS FILES (PAT 7)	C	45.2.5 M	Jul 81
STANDALONE FORTRAN YIELDS RUN-TIME ERROR 64 (PAT 8)	B	45.2.6 M	Apr 81
DISPOSE = 'KEEP' NOT RECOGNIZED WITH READONLY OPEN PARAMETER (PAT 9)	C	45.2.7 M	Jul 81
THE DATE ROUTINE DOES NOT PERMIT BYTE ALIGNED PARAMETERS (PAT10)	C	45.2.8 M	Jul 81
IMPLICIT READ FAILURE MAY HALT PROCESSOR (PAT 11)	C	45.2.9 M	Jul 81
FPU DOUBLE PRECISION SINE/COSINE MODULE ERRORS (PAT 13)	D	45.2.10 M	Sep 81
EMBEDDED BLANKS OVERRIDE THE ICNT PARAMETER IN THE ASSIGN ROUTINE	D	45.2.11 M	Oct 81
THE DEFAULT CARRIAGE CONTROL FOR THE ASSIGN ROUTINE IS INCORRECT	D	45.2.12 M	Oct 81
CORRECTION FOR UNIT CLOSING (PAT 16)	E	45.2.13 M	Nov 81
LIST DIRECTED INPUT CONVERSION ERROR (PAT 19)	E	45.2.14 M	Dec 81
BOUNDARY CONDITION ON FORMATTED I/O CORRUPTS I/O IN PAT 6 (PAT 23)	F	45.2.15 M	Feb 82
BOUNDARY CONDITION ON FORMATTED I/O BACKSPACE CORRUPTS I/O	F	45.2.16 M	Feb 82
CORRECTION OF ASSIGN FILENAME HANDLING WHEN ICNT EQUALS ZERO	F	45.2.17 M	Feb 82
CONVERSION ERROR WHILE READING COMPLEX NUMBER FROM FILE (PAT 26)	F	45.2.18 M	Apr 82
CORRECTION TO ALLOW CLOSING OF UNIT RECORD DEVICES (PAT 28)	G	45.2.19 M	Jun 82
PREMATURE CLEARING OF ERR= BRANCH WHEN EOF IS ENCOUNTERED (PAT 30)	G	45.2.20 M	Jun 82
UIOBYT PREMATURELY DETERMINES END OF BLOCK (PAT 32)	G	45.2.21 M	Jul 82
INVALID DATA IS ACCEPTED DURING LIST DIRECTED I/O		45.2.22 M	Nov 82

GAMMA V3.1

Component	Autopatch Kit	Sequence	Mon/Yr
FGAMMA-FRAMES 3 TO 10 OF GSA STUDY SOMETIMES CORRUPT		49.2.1 M	Jul 81
SYSTEM MAY HANG WHEN DISK SQUEEZED		49.2.2 M	Oct 81
STATIC STUDIES ON LARGE DEVICES		49.2.3 M	Jan 82
STATIC STUDY ACQUISITION ON LARGE DEVICES		49.4.1 M	Jan 82
ISOMETRIC DISPLAY IMAGES USE INCORRECT INTENSITY LEVELS		49.5.1 M	Oct 81
SLICE - LAST POINT IS NOT PLOTTED		49.5.2 M	Nov 81
SLICE - <CR>, <LF> NOT ISSUED AFTER PRINTING SLICE DATA		49.5.3 M	Jan 82
DYNAMIC CURVE RECALCULATION IN REGIONS OF INTEREST		49.5.4 M	Aug 82
TRANSFER STUDY IN SELECTIVE STEP MODE		49.8.1 F	Mar 82
GAMMA-11 DOCUMENTATION CORRECTIONS AND ADDITIONS		49.10.1 N	Mar 82
PATCHING THE RT-11 MONITOR FOR GAMMA-11		49.11.1 M	Nov 81
ERROR IN THE BASIC SUPPORT ROUTINE GPMR		49.12.1 M	Aug 82
ERRORS IN THE BASIC SUPPORT ROUTINES GPLR AND GPF		49.12.2 M	Aug 82
ERROR IN FORTRAN SUPPORT SUBROUTINE GPMR		49.13.1 M	Mar 82
ERRORS IN THE FORTRAN SUPPORT ROUTINES GPLR AND GPF		49.13.2 M	Mar 82

CTS-300 V6.0

Component	Autopatch Kit	Sequence	Mon/Yr
DBUILD			
CORRECTION FOR THREE DECFORM PROBLEMS		51.2.1 M	Oct 81
DECFORM			
PROBLEM WITH DECFORM AND THE VT100		51.4.1 M	Nov 80
CORRECTION FOR THREE DECFORM PROBLEMS		51.4.2 M	Oct 81
DECFORM WITH VT100 TERMINAL CAUSES BAD CHARACTER ON TYPE-AHEAD		51.4.3 M	Nov 81
DIBOL			
TWO CORRECTIONS TO XCALL PAK/UNPAK		51.5.1 M	Aug 81
DICOMP			
FOUR DICOMP ERRORS FIXED		51.6.1 M	Oct 81
DKED			
TWO PROBLEMS WITH DKED		51.7 M	Aug 80
DKED SELECT/CUT AND KEYPAD ERRORS		51.7.2 M	Sep 80
DKED INCORRECTLY HANDLES CONTINUED LINES		51.7.3 M	Oct 81
POSSIBLE BOTTOM OF SCREEN CORRUPTION USING DKED		51.7.4 M	May 82

ISMUTL

	Sequence	Mon/Yr
CORRECTIONS FOR ISAM UTILITY ERRORS	51.8.1 M	Nov 81
ISMUTL GIVES INCORRECT ERROR MESSAGES IF INSUFFICIENT MEMORY AVAILABLE	51.8.2 M	Apr 82

LPTSPL

	Sequence	Mon/Yr
TSD SPOOLER GETS CONFUSED	51.9.1 M	Nov 80

SORTM

	Sequence	Mon/Yr
SORT SENDS MESSAGES INDISCRIMINATELY	51.14.1 M	Jan 81

SUD

	Sequence	Mon/Yr
CORRECTIONS TO DIBOL RUN TIME SYSTEMS	51.16.1 M	Jan 81
PROBLEMS WITH XCALL RENAM AND ERROR 6	51.16.2 M	Feb 81
NO ERROR 22 RETURNED	51.16.3 M	Nov 81
DIBOL STACK OVERFLOW ON OPEN	51.16.4 M	Nov 81
PROBLEMS WITH STACK OVERFLOW AND INCREMENT	51.16.5 M	Dec 81
SUD MESSAGES OVER 100 CHARACTERS IN LENGTH ARE NOT RECEIVED CORRECTLY	51.16.6 M	Feb 82
ISAM FILE RECORD COUNT REVERTS TO 0	51.16.7 M	Apr 82
A SUD PROGRAM DOING AN XCALL MAY RESULT IN A TRAP TO 4 OR 10	51.16.8 M	Jul 82
ERRORS IN DATA FORMATTING WITH MASK	51.16.9 M	Oct 82

TDIBOL

	Sequence	Mon/Yr
PROBLEM WITH XCALL PAK	51.17 M	Aug 80
PROBLEM UNPACKING DATA	51.17.2 M	Sep 80
TWO CORRECTIONS TO XCALL PAK/UNPAK	51.17.3 M	Aug 81

TSD

	Sequence	Mon/Yr
CORRECTIONS TO DIBOL RUN TIME SYSTEMS	51.18.1 M	Jan 81
PROBLEMS WITH XCALL RENAM AND ERROR 6	51.18.2 M	Feb 81
INCORRECT TERMINAL WIDTHS AND CIS PROBLEM	51.18.3 M	Aug 81
CORRECTION TO TSD/XMTSD	51.18.4 M	Sep 81
CORRECTION FOR ISAM PROBLEM	51.18.5 M	Oct 81
"SEND" STARTS MULTIPLE JOBS	51.18.6 M	Oct 81
NO ERROR 22 RETURNED	51.18.7 M	Nov 81
DIBOL STACK OVERFLOW ON OPEN	51.18.8 M	Nov 81
PROBLEMS WITH STACK OVERFLOW AND INCREMENT	51.18.9 M	Dec 81
CORRECTION FOR SIDE EFFECTS FROM PATCH 27	51.18.10 M	Feb 82
LINE PRINTER IS SOMETIMES INCORRECTLY CONSIDERED IN USE	51.18.11 M	Feb 82
ISAM FILE RECORD COUNT REVERTS TO 0	51.18.12 M	Apr 82
TSD AND XMTSD HANG AFTER ATTEMPT TO ILLEGALLY START UP JOB	51.18.13 M	May 82
ERRORS IN DATA FORMATTING WITH MASK	51.18.14 M	Oct 82

XMTSD

	Sequence	Mon/Yr
CONFLICT BETWEEN XMTSD AND RT-11 OVER CHANNEL 16	51.20 M	Aug 80
CORRECTIONS TO DIBOL RUN TIME SYSTEMS	51.20.2 M	Jan 81
PROBLEMS WITH XCALL RENAM AND ERROR 6	51.20.3 M	Feb 81
PATCH FOR XMTSD WITH CIS	51.20.4 M	Apr 81
INCORRECT TERMINAL WIDTHS AND CIS PROBLEM	51.20.5 M	Aug 81
XMTSD HANGS WHEN LP IS OFF-LINE	51.20.6 M	Sep 81
CORRECTION TO TSD/XMTSD	51.20.7 M	Sep 81
CORRECTION FOR ISAM PROBLEM	51.20.8 M	Oct 81
"SEND" STARTS MULTIPLE JOBS	51.20.9 M	Oct 81
NO ERROR 22 RETURNED	51.20.10 M	Nov 81
DIBOL STACK OVERFLOW ON OPEN	51.20.11 M	Nov 81
PROBLEMS WITH STACK OVERFLOW AND INCREMENT	51.20.12 M	Dec 81
CORRECTION FOR SIDE EFFECTS FROM PATCH 27	51.20.13 M	Feb 82
LINE PRINTER IS SOMETIMES INCORRECTLY CONSIDERED IN USE	51.20.14 M	Feb 82
ISAM FILE RECORD COUNT REVERTS TO 0	51.20.15 M	Apr 82
XMTSD GIVES INCORRECT ERROR WHEN NO ROOM FOR I/O BUFFER	51.20.16 M	Apr 82
TSD AND XMTSD HANG AFTER ATTEMPT TO ILLEGALLY START UP JOB	51.20.17 M	May 82
ERRORS IN DATA FORMATTING WITH MASK	51.20.18 M	Oct 82

DOCUMENTATION

	Sequence	Mon/Yr
CTS-300 VERSION 6 IS RELEASED	51.21 N	Aug 80
TWO RT-11 PATCHES MODIFIED FOR CTS-300 USE	51.21.2 N	Oct 80
RT-11 PATCH TO LS.MAC MODIFIED FOR CTS-300 USE	51.21.3 N	Feb 81
ADDITIONS TO CTS-300 DOCUMENTATION ON PRINT UTILITY	51.21.4 N	Mar 81
LIST OF SEQUENCE NUMBERS FOR CTS-300 V6	51.21.5 N	Mar 81
SOME NOTES ON RT-11 PATCH SEQ 6.13.3 M TO LS.MAC FOR CTS-300 USERS	51.21.6 M	Jul 81

Component	Autopatch Kit	Sequence	Mon/Yr
SOME NOTES ON RT-11 PATCH SEQ 6.13.4 M TO LS.MAC FOR CTS-300 USERS		51.21.7 N	Aug 81
SOME NOTES ON RT-11 PATCH SEQ 6.13.5 M TO LS.MAC FOR CTS-300 USERS		51.21.8 N	Aug 81
AVOIDING POSSIBLE PROBLEM WITH ISAM FILES		51.21.9 N	Dec 81
SOME NOTES ON RT-11 PATCH SEQ 6.13.6 M TO LS.MAC FOR CTS-300 USERS		51.21.10 N	Feb 82
RESTRICTION FOR CTS-300		51.21.11 R	Apr 82
LS.MAC			
SPECIAL CTS-300 PATCH FOR LS.MAC		51.23.1 M	Feb 81
CORRECTION TO CTS-300 PATCH 11 (SEQ 51.23.1 M) TO LS.MAC		51.23.2 M	Jun 81
SYSTBL.CND			
RT-11 PATCH TO SYSTBL.CND MODIFIED FOR CTS-300 USE		51.25.1 M	Mar 81
RT-11 PATCH SEQ 10.3.2 M TO SYSTBL.CND MODIFIED FOR CTS-300 USE		51.25.2 M	Apr 81
RT-11 PATCH SEQ 10.3.3 M TO SYSTBL.CND MODIFIED FOR CTS-300 USE		51.25.3 M	May 82

CTS-300 V7.0

Component	Autopatch Kit	Sequence	Mon/Yr
DOCUMENTATION			
CTS-300 VERSION 7 IS RELEASED		52.1.1 N	Apr 82
XMTSD RUN-TIME SYSTEM SIZE		52.1.2 N	Jun 82
CHANGING THE DEFAULT TIME SLICE VALUE FOR XMTSD		52.1.3 N	Jun 82
RELINK DIBOL PROBLEMS FOR CTS-300 V7		52.1.4 N	Jun 82
PATCH LEVEL FOR KED/K52 CLARIFIED		52.1.5 N	Aug 82
NO ROOM FOR SOME BACKGROUND JOBS UNDER CTS-300 V7		52.1.6 N	Nov 82
CTS-300 AUTOPATCH KIT A PATCH LEVEL		52.1.7 N	Nov 82
DIBOL RUN-TIME SYSTEMS			
PATCH 5: VARIOUS TSD AND XMTSD PROBLEMS		52.3.1 M	Jun 82
PATCH 6: ISAM FILE RECORD COUNT REVERTS TO 0		52.3.2 M	Jun 82
PATCH 11: TWO RUN-TIME SYSTEM ERRORS		52.3.3 M	Oct 82
PATCH 13: TWO PROBLEMS: ISAM STORE/WRITE AND LPQUE STATEMENT		52.3.4 M	Oct 82
PATCH 18: RUN-TIME SYSTEM PROBLEMS WITH USR LOCKED, COMPILATION ERRORS		52.3.5 M	Nov 82
DIBOL/TDIBOL			
PATCH 2: POSSIBLE INCORRECT RESULTS FROM THE INSTR ROUTINE		52.4.1 M	Apr 82
DICOMP V07-00			
PATCH 16: CONDITIONAL COMPILATION ERROR, AND NO ERROR FOR LARGE RECORD		52.5.1 M	Nov 82
DKED			
PATCH 8: POSSIBLE BOTTOM OF SCREEN CORRUPTION USING DKED		52.6.1 M	Jul 82
ISMUTL			
PATCH 15: RUNNING ISMUTL IN AUTO-CREATE MODE		52.7.1 M	Oct 82
TSD LINE PRINTER SPOOLER			
PATCH 12: LINE PRINTER SPOOLER PROBLEMS WITH DELETE AND /FLUSH		52.9.1 M	Oct 82
PATCH 19: ERRORS IN LINE PRINTER SPOOLER PACKAGE		52.9.2 M	Dec 82
ERMSG.TXT			
PATCH 9: INCORRECT ERROR MESSAGES FOR SORT IN ERMSG.TXT		52.10.1 M	Jul 82
DIBOL SORT			
PATCH 7: ERROR RECEIVED WHEN PERFORMING A LEGAL SORT		52.14.1 M	Jul 82
MACRO SORT			
PATCH 1: TWO SORT PROBLEMS EMERGE UNDER CERTAIN CONFIGURATIONS		52.15.1 M	Jun 82
PATCH 3: SINGLE USER SORT MAY LEAVE TEMPORARY FILES ON DISK		52.15.2 M	Jul 82
PATCH 10: TWO MACRO SORT PROBLEMS		52.15.3 M	Aug 82
PATCH 17: PROBLEMS WITH HANDLING ERRORS IN SORT COMMAND FILE		52.15.4 M	Nov 82
PATCH 20: INCOMPLETE SORT, FILESPEC ERRORS		52.15.5 M	Dec 82
SYSTBL.CTS			
PATCH 4: TERMINAL OUTPUT CORRUPTION ON DZ11 OR DZV11 LINES		52.16.1 M	Jun 82

Component	Autopatch Kit	Sequence	Mon/Yr
LS.CTS			
PATCH 14: PROBLEMS WITH LS HANDLER		52.17.1 M	Oct 82

CTS-300 DICAM (3271) V3.1

INCORRECT ACK SENT IN CONVERSATIONAL MODE		55.1.1 M	Jul 81
LOOP WHEN CLOSE IS ISSUED WITH OUTSTANDING I/O REQUESTS		55.1.2 M	Jul 81

CTS-300 RDCP (2780/3780) V2.0

ABNORMAL TERMINATION AND LISTING PROBLEMS		56.1.1 M	Dec 80
SUBSCRIPT ERROR IN RDCP EDITOR		56.1.2 M	Dec 80
MEMORY CORRUPTION PROBLEM		56.1.3 M	Dec 80

DECtype-300 V1.1

REPEATED USE OF THE PASTE FUNCTION WILL CAUSE AN ERROR 28		57.1.1 M	Jun 82

RGL/FEP

INVALID LABELS FOR DATA RANGE OF 0.1 TO 1.0		58.1.1 M	Aug 82
ERROR CALLING LOCATE, LFIXED OR LFREE TWICE IN SUCCESSION		58.1.2 M	Aug 82
USING FORTRAN AND LINK OPTIONS WITH RGL/FEP		58.1.3 M	Dec 82
DEVELOPING RGL/FEP APPLICATIONS		58.1.4 M	Dec 82
RESTRICTIONS WITH GSAVE FILES		58.1.5 M	Dec 82

RT-11/FORTRAN ENHANCEMENT PACKAGE for MINC (FEP)

INVALID LABELS FOR DATA RANGE OF 0.1 TO 1.0		59.1.1 M	Aug 82
ERROR CALLING LOCATE, LFIXED FOR LFREE TWICE IN SUCCESSION		59.1.2 M	Aug 82
USING FORTRAN AND LINK OPTIONS WITH RGL/FEP		59.1.3 M	Dec 82
DEVELOPING RGL/FEP APPLICATIONS		59.1.4 M	Dec 82
RESTRICTIONS WITH GSAVE FILES		59.1.5 M	Dec 82

REAL-11/MNC			
UNDEFINED GLOBAL DRSW10 IN MNCLIB		59.4.1 M	Jul 82
DATA SENT BY THE MAIN PROGRAM IS CORRUPTED BY THE SRQ ROUTINE		59.5.1 M	Jul 82
IBSRQ SKIPS INSTRUMENT ADDRESS IF SRQ ROUTINE DEFAULTED		59.5.2 M	Jul 82
SRQ ROUTINE AND TIMEOUT VALUE NOT CLEARED ON EXIT		59.5.3 M	Jul 82
SYSTEM CRASHES IF THE IB DRIVER IS NOT LOADED		59.5.4 M	Jul 82
CAN'T SPECIFY TALKER WHEN LISTENERS DEFAULTED, AND INCORRECT RECEIVE		59.5.5 M	Jul 82
CANNOT USE SECONDARY ADDRESSES IN RANGE 96. to 126.		59.5.6 M	Jul 82

QUILL V1.0

QUILL.SAV/TSD			
PATCH 1: PRIMARY FILE BLOCK COUNT PROBLEM		60.1.1 F	Nov 82
QBUILD.SAV/TSD			
PATCH 2: QBUILD WORK FILE SIZE ALLOCATION		60.2.1 F	Nov 82

Software Product Description

PRODUCT NAME: RT-11 MDE/T-11, Version 1.0
Microcomputer Development Environment

DESCRIPTION:

RT-11 MDE/T-11 is a hardware/software in-circuit emulation development system for the Micro T-11 16-bit microprocessor. RT-11 MDE/T-11 offers features for observing, controlling, and simulating the hardware interface between the processor and external circuitry, without impacting the speed, performance, or operating environment of the Micro/T-11. RT-11 MDE/T-11 includes a MACRO-11 source language debugger.

Features:

- Ability to EXAMINE and DEPOSIT memory and register contents

- Full symbolic support, using program and user defined symbols

- MACRO-11 line assembly and disassembly

- Ability to load memory image files developed under VAX/VMS into the Micro/T-11 address space

- Simulation of RAM/ROM in portions or all of the Micro/T-11 address space

- Data display/entry in various radices and formats

- VT100 split-screen status display automatically updated

- Predefined and user-definable VT100 and LA120 keypad command keys

- Nested indirect command files, logging, and an internal HELP facility

- Error and warning messages

- Hardware diagnostics

- System exerciser command file

RT-11 MDE/T-11 contains a hardware interface that observes and controls the Micro/T-11 signals in real-time to implement the following features:

- Micro/T-11 Bus cycle tracing (1024 cycle capacity)

- Protection of memory against read and write access by the Micro/T-11

- Counted single-instruction STEPPING

- Counted hardware BREAKPOINTS, TRACEPOINTS, and WATCHPOINTS

- Event-driven external trigger signal

- 16-line external probe support (for event detection and tracing)

- Masked Micro/T-11 signal pattern event detection (state analysis)

MINIMUM HARDWARE REQUIRED:

Any valid RT-11 XM system configuration with

- An EIA RS232 serial port

- 64K words of memory

OPTIONAL HARDWARE:

- Additional 32K byte MDE-BM Memory Simulator board

- Up to two additional MDE-BD State Analyzer boards

PREREQUISITE SOFTWARE:

RT-11 Operating System, Version 4.0, with multi-terminal support

OPTIONAL SOFTWARE:

None

TRAINING CREDITS:

None

SUPPORT CATEGORY:

DIGITAL SUPPORTED
RT-11 MDE/T-11 is a DIGITAL Supported Software Product.

SOFTWARE INSTALLATION:

CUSTOMER INSTALLED
RT-11 MDE/T-11 is a software product engineered to be installed by the customer and includes other Software Product Support services listed below.

SOFTWARE PRODUCT SUPPORT:

RT-11 MDE/T-11 includes standard warranty services as defined in the Software Support Categories Addendum of this SPD, except that no Newsletter or on-site remedial service will be provided.

software

-2-

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.

A single-use, license-only option is a license to copy the software previously obtained under license.

RT-11 MDE/T-11 software is not available as a separate product. This software is provided as a standard part of each RT-11 MDE/T-11 system.

The following key (E, G, H, Q, X, Y, Z) represents the distribution media for the product and must be specified at the end of the order number, e.g., QJA61-HE = binaries on RK05 Disk Cartridge.

E = RK05 Disk Cartridge
G = TU58 DECtape II Cartridge*
H = RL02 Disk Cartridge
Q = RL01 Disk Cartridge
X = RX02 Double Density Diskette
Y = RX01 Floppy Diskette
Z = No hardware dependency

* The TU58 is to be used in a stand-alone, lightly loaded environment. If used as a file device in a heavily loaded environment, it can degrade system performance.

MDETS -R— MDE hardware, RT-11 MDE/T-11 Host Development Software, single-use license, binaries, documentation, support services. Includes a license to use RT-11 MDE/T-11 software on MDE hardware (media: E, G, H, Q, X, Y)

Update/Unsupported Options

Users of RT-11 MDE/T-11 whose specified Support Category warranty has expired may order under license the following software option as an update to an earlier version. The option may also be purchased for use on a second or subsequent CPU, in conjunction with a binary, single-use, license-only option. Options are distributed in binary form on the appropriate medium and include no installation or other services unless specifically stated.

QJA61 -H— Binaries, documentation (media: E, G, H, Q, X, Y)

QJA61 -H— Right to copy for single-use, no binaries, no documentation (media: Z)

Miscellaneous Options:

QJA61 -G— Documentation-only kit (media: Z)

ADDITIONAL SERVICES:

The following post-warranty Software Product Services for this software product are available to licensed customers:

• Self-Maintenance Service
• Basic Service

The prerequisite being the purchase of the equivalent level RT-11 Software Product Service. Customers should contact their local DIGITAL office for additional information on the availability of these services.

Software Product Description

PRODUCT NAME: DECtype-300, Version 2.0
Word Processing Application

DESCRIPTION:

DECtype-300 is a word processing application designed to run on the CTS-300 Operating System, permitting concurrent word processing and data processing in a multiuser environment. This feature extends the data processing system with word processing capabilities.

The DECtype-300 editor creates and maintains documents stored on the full range of disk devices supported by CTS-300. Storage available for documents will vary depending upon other data storage requirements on the same disk media.

Two DECtype-300 terminals, running concurrently, is the maximum number recommended on any CTS-300 configuration.

Features

- Concurrent word processing and data processing
- Center screen editing allowing view of previous and following text
- Menu driven structure
- Special editing keypad
- Editing features:
    - Cut and paste blocks of text
    - Insert bodies of text and/or boilerplates from library files
    - Insert abbreviations, phrases and words from abbreviation files
    - Swap transposed characters and words
    - Delete and rubout by character and word, rubout by sentence and line
    - Search and replace capability
    - Wide document editing and printing
    - Four function math capabilities
    - Jump to page
    - Edit while viewing special control characters
- Full control of tabs, margins, justification, and pagination:
    - Automatic centering of text on a line
    - Discretionary pagination control
    - Decimal and right-adjusted tabs
- Ability to support up to four draft and/or letter quality printers

- Underlines and bolding appear on screen
- Stop printer menu
- Selectable pitch on the letter quality printer
- Underlined and overstruck (bold) printout
- Superscript and subscript
- List processing of data prepared by DECtype-300 and other data processing programs
- Date and time stamp
- Document statistics
- User-defineable keys for predetermined and repetitive operations

MINIMUM HARDWARE REQUIRED:

Any valid CTS-300 configuration with 128KB of memory supporting one VT100-NA or VT102-NA Terminal with the advanced video option, one line printer and two RX02 Diskette Drives. The VT100-WA or VT102-WA Word Processing Terminals are recommended for DECtype-300.

OPTIONAL HARDWARE:

- LQPSE-FA Letter Quality Printer (Serial)
- LQP02-AA Letter Quality Printer with LQPX2-AA bidirectional tractor
- Any valid line printer supported by the prerequisite software
- LA100 Letterprinter 100, supported as a draft printer
- Additional VT100-WAs or VT102-WAs
- Any valid disk storage device supported by the prerequisite software
- VT1XX-CE Keyboard upgrade kit for VT100-NA to VT100-WA

PREREQUISITE SOFTWARE:

CTS-300 Operating System, Version 7.0

OPTIONAL SOFTWARE:

None

TRAINING CREDITS:

None

SUPPORT CATEGORY:

DIGITAL SUPPORTED

DECtype-300 is a DIGITAL Supported Software Product.

digital
software

SOFTWARE INSTALLATION:
CUSTOMER INSTALLED

DECtype-300 is a software product engineered to be installed by the customer and includes other Software Product Support services listed below.

SOFTWARE PRODUCT SUPPORT:

DECtype-300 includes standard warranty services as defined in the Software Support Categories Addendum of 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.

A single-use, license-only option is a license to copy the software previously obtained under license.

The following key (H, Q, V, X, Z) represents the distribution media for the product and must be specified at the end of the order number, e.g., QJ038-AH = binaries on RL02 Disk Cartridge.

H = RL02 Disk Cartridge
Q = RL01 Disk Cartridge
V = RK07 Disk Cartridge
X = RX02 Double Density Diskette
Z = No hardware dependency

English Language

QJ038 -A— English single-use license, binaries, documentation, support services (media: H, Q, V, X)

QJ038 -D— English single-use license-only option, no binaries, no documentation, no support services (media: Z)

French Language

QJ068 -A— French single-use license, binaries, documentation, support services (media: H, Q, V, X)

QJ068 -D— French single-use license-only option, no binaries, no documentation, no support services (media: Z)

Update/Unsupported Options

Users of DECtype-300 whose specified Support Category warranty has expired may order under license the following software option as an update to an earlier version. The option may also be purchased for use on a second or subsequent CPU, in conjunction with a binary, single-use, license-only option. Options are distributed in binary form on the appropriate medium and include no*installation or other services unless specifically stated.

English Language

QJ038 -H— Binaries, documentation (media: H, Q, V, X)

QJ038 -H— Right to copy for single use, no binaries, no documentation (media: Z)

French Language

QJ068 -H— Binaries, documentation (media: H, Q, V, X)

QJ068 -H— Right to copy for single use, no binaries, no documentation (media: Z)

Users of DECtype-300 on D315-A-WA, -WD, -WY, or -WZ 64KB systems whose specified Support Category warranty has expired may order under license the following combination software update and hardware upgrade at the prevailing rate. The software update is distributed in binary form on the appropriate medium and includes no installation or other services unless specifically stated. The hardware upgrade includes a 64KB memory module with no installation or other services unless specifically stated.

DS3AA -H—Binaries, documentation, MSV11-DD memory module (media: X)

DS3AA -H—Right to copy for single use, no binaries, no documentation, MSV11-DD memory module (media: Z)

Miscellaneous Options

QJ038 -G— English documentation-only kit (media: Z)

QJ068 -G— French documentation-only kit (media: Z)

ADDITIONAL SERVICES:

Basic Service is available to licensed customers as a post-warranty Software Product Service for this software product.

The prerequisite being the purchase of the equivalent level CTS-300 Software Product Service. Customers should contact their local DIGITAL office for additional information on the availability of this service.

DIGITAL SOFTWARE LICENSING

DATA SHEET

Digital protects its investment in software by treating software as proprietary information. We do not transfer ownership of software to our customers—we license them to use our software under the terms and conditions of our various software license agreements.

Key principles of the terms and conditions of Digital software binary license agreements are:

- Customers must have a binary license to use any of our binary software products.

- A Binary License allows ONE customer to run ONE software product on the CPU on which the software is first installed.

- Digital retains title and ownership.

- Software may only be transferred to another party with written permission from Digital.

- A customer may reproduce the software, if necessary, but only for use on the specific CPU licensed to use that software.

- The use of an update version of the software on the same CPU licensed to use the product requires that the customer purchase a software update option if out of warranty or not covered by a software service contract.

HOW A CUSTOMER OBTAINS A BINARY LICENSE

A customer obtains a Digital binary license upon execution of the transactions described below:

- Accepting Digital's standard terms on the customer's purchase order
- Referencing a Digital quotation number on the face of the purchase order
- Executing the binary license agreement and returning it to Digital (usually only executed for a transfer or if no license is apparent)

OEM and other volume customers can reference a valid Digital purchasing agreement on their purchase orders. These agreements include:

- Quantity Discount Agreement (QDA)
- OEM Contract
- GSA Contract
- Master Agreement containing the applicable software licensing clause.
- Purchase and Sales agreement containing the appropriate licensing clause.

DIGITAL-SUPPORTED BINARY LICENSE OPTION

This is a standard binary license that includes media, manuals, documentation and warranty packaged together. A 90 day warranty, as specified in the SPD Addendum, is the support received (unless different warranty conditions are specified in the SPD).

CUSTOMER-SUPPORTED BINARY LICENSE OPTION

This is a standard binary license which includes media, manuals, documentation but no warranty support. It is only offered when a Digital-supported license option is not offered.

LICENSE-ONLY OPTION

A license only option is a standard binary license, but has no media, manuals, documentation or support. Software products can be ordered at a considerable cost reduction but the customer must have first purchased a previous license with media for that software product.

The license-only option is a one time right to copy. It is a license to run a single software product on one additional specific CPU using a copy of the software the customer made from the original licensed product. Customers can order additional copies of the documentation.

OUT-OF-WARRANTY UPDATE OPTION

A customer with a binary license may order a product update for each licensed CPU. An additional fee is charged for each product update or for each one time right to copy update for each licensed CPU.

CPU BACKUP LICENSE PROVISION

There is a specific provision that permits customers to use the software on another processor should the licensed machine be inoperative. This provision is intended to allow customers whose computer is "down" to use another machine while theirs is being repaired. This provision does not permit the use of the software on a "HOT" backup unless a second license is obtained.

SOURCE LICENSING

Sources in machine readable, listing or microfiche form are only available on selected products when specified on a Software Product Description. A source license agreement is required and must be approved by the Digital Contracts Department.

MODIFICATION TO DIGITAL LICENSED SOFTWARE

Modification made to Digital-licensed software does not exempt the software product from Digital licensing or sublicensing terms and conditions or from payment of licensing fees to Digital. Every line of code from a software product falls under the terms and conditions of the license. Only those modifications that are not part of the original software are the customer's property.

LICENSE TRANSFER

Digital's licensing agreement does not allow the transfer of software from one end user to another or from one CPU to another without prior permission from Digital.

OEM SUBLICENSING

SUBLICENSES DEFINED

Sublicenses are contractual agreements between the Digital OEM and the OEM's customers under which the customers are licensed to use Digital's binary software. In the OEM agreement, Digital gives the OEM the right to sublicense, allowing the OEM to provide Digital's binary software without case-by-case permission for use on a system furnished by the OEM.

The OEM is permitted to create a new license and is not required to disclose the identity of its customer to Digital (unless services, etc., from Digital is required). However, for sublicensed software, Digital must always receive a license fee from the OEM.

A sublicense agreement must include the terms and conditions specified in Digital's standard OEM terms and conditions of sale, in addition to whatever added value the OEM might include in the agreement. Digital supplies a sample sublicense for the OEM to use in conjunction with or integrated into the OEM's sale agreement.

The OEM must always obtain a signed sublicense agreement from its customer before any Digital software is furnished to the customer.

OEM USE OF BINARY LICENSED SOFTWARE

A Digital OEM, in addition to being able to use the Digital licensed software on a backup machine in the case of a CPU malfunction, may also use the software on CPU's intended for resale, provided that:

* the OEM possesses the CPU; or
* the OEM uses software on only one additional CPU at a time.

This provision permits OEMs to integrate their software system before selling or shipping it. The OEM must sublicense the software to the purchaser of the system.

SUBLICENSE VS. LICENSE

A sublicense includes the same rights and responsibilities as a direct license from Digital.

The principal difference between a sublicense and a license is warranty and/or support services. The OEM who purchases a product with services owns and receives the service directly. However, Digital may provide these services to the OEM at the end user's site if requested.

◨DECUS

WHY YOU SHOULD JOIN DECUS

- SYMPOSIA
- PROGRAM LIBRARY

- TECHNICAL PUBLICATIONS
- SPECIAL USER GROUPS

DECUS (the Digital Equipment Computer Users Society), a worldwide association of customers and employees, provides a forum for the exchange of useful information, new program packages, and other innovations among those who use and supply the products of Digital Equipment Corporation.

Founded in 1961, DECUS is one of the largest and most active associations of its type in the world. Its objectives 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 of 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.

DECUS membership is free--upon application--to owners of DIGITAL computers and to their computer-interested employees. Membership carries important benefits and opportunities; among them are access to the program library; membership in local, regional, and national organizations; invitations to symposia dedicated to optimal use of DIGITAL equipment; opportunity to present papers and workshops on your own new ideas; and, finally, access to special interest groups dedicated to particular uses, languages, operating systems, and hardware configurations.

The program library maintained by DECUS contains over 1700 active software packages written and submitted by members and DIGITAL employees, and available to members for the media fee and reproduction cost only. Programs in the library range from enhanced editors and cross compilers to statistics packages and games. Of particular interest to college and university customers, for example, might be a package of programs for registration, class scheduling, dormitory management, and annual giving records. A laboratory user could take advantage of various statistical packages, or programs that perform Fourier transforms or least squares fitting. There are programs for circuit analysis, resonance simulation, blood-count evaluation, and stress testing, and scores of others which medical, scientific, or engineering customers could employ. Business people can find accounting packages, data analysis and payroll programs among the library's offerings. In addition, of course, there is a wide range of text editing, display graphics, and enhanced utility programs available.

Local, regional, and national DECUS organizations give members the opportunity to meet other DIGITAL customers and employees in an informal setting. From the monthly local meeting to the semiannual national symposium, the members can discuss their ideas, can learn what others are doing, and can give DIGITAL feedback necessary in improvement and future development of important products. Often, the national meetings in the various countries also provide the stage for major new product announcements by the company, and a showplace for interesting developments in both hardware and software technology. At any meeting a member might describe ideas and programs he has implemented, or fine tuning that has been achieved for a particular application. Members give papers, participate in panel discussions, lead workshops, or conduct demonstrations for the benefit of other members.

DECUS also publishes newsletters focusing on special interest, technical books that contain the compilation of symposia presentations; and a society newsletter.

Many members derive a particular benefit from joining DECUS Special Interest Groups. Special Interest Groups often meet as subsets of regional and national meetings, or they may meet on their own, to discuss their special interest. Here, all RSTS/E users; or everyone interested in COBOL, for example, can have a chance to get together and discuss topics of mutual importance. At present there are more than 20 Special Interest Groups (SIGs) in the U.S. alone. Many of the SIGs print newsletters and disseminate valuable technical information to members. The SIGs really are the front-line of mutual help and problem solving.

DIGITAL provides DECUS with administrative personnel and office space around the world, but the organization is run by its members, who act as speakers for conferences, planners for meetings, editorial and production talent for newsletters and minutes, and the inventors of the ideas and new programs necessary to keep the library up to date. Belonging to DECUS is a valuable adjunct to owning DIGITAL equipment on both the program exchange and the information exchange fronts.

To obtain a DECUS membership form, complete the form below and return it to the appropriate chapter office.

CHAPTER	ADDRESS
AUSTRALIA (Australia, Brunei, Indonesia, Malaysia, New Zealand, Singapore)	DECUS Australia P.O. Box 384 Chatswood NSW 2067 Australia
CANADIAN (Canada)	DECUS Canada P.O. Box 13000 Kanata, Ontario K2K 2A6 Canada
EUROPEAN (Europe, Middle East, North Africa, Russia)	DECUS Europe P.O. Box 510 12, avenue des Morgines CH-1213 Petit-Lancy 1/GE Switzerland
U.S. (U.S. and all other countries)	DECUS U.S. Chapter One Iron Way Marlboro, Massachusetts 01752 U.S.A.

Please send me a DECUS membership form.

NAME: _____
(First)   (Last/Family Name)

COMPANY: (INSTALLATION) _____

ADDRESS: _____

_____

_____

_____
(City, Town, State/Province, and Zip/Postal Code)

COUNTRY: _____

TELEPHONE: _____ TELEX _____

I obtained this form from _____

July 1980

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	Corporate Administrative Systems Group P.O. Box F Maynard, MA 01754
Canada	Digital Equipment of Canada, Ltd. P.O. Box 13000 Kanata, Ontario Canada, K2K 2A6
United Kingdom, Bahrein, Egypt, Iraq, Jordan, Kuwait, Lebanon, Libya, Qatar, Oman, Saudi Arabia, Syria, United Arab Emirates, Yemen. Arab Republic	Digital Equipment Co. Ltd. 2 Cheapside GB - Reading, Berkshire RG1 7AA England
Australia, New Zealand	Digital Equipment Aust. Pty. Ltd. P.O. Box 384 Chatswood, New South Wales 2067 Australia
Brazil	Digital Equipment Comercio e Industria Ltda. Avenida Augusto Severo, 156-A 20021 Rio de Janeiro, RJ Brazil
Caribbean	Digital Equipment Latin America P.O. Box 11038 Fernandez Juncos Station Santurce 00910 Puerto Rico
France	Digital Equipment France Cidex L225 18 Rue Saarinen F-94528, Rungis France
Italy	Digital Equipment S.p.A. Viale Fulvio Testi, 11 Ang. Via Gorki 105 I-20092 Cinisello Balsamo Milan Italy
Japan	Digital Equipment Corp. Intl. Japan Sunshine 60, P.O. Box 1135 1-1⁄ Higashi Ikebukuro 3-Chome, Toshima-Ku, Tokyo, 170 Japan
Belgium, Holland, Luxemburg	Digital Equipment B.V. Kaap Hoorndreef 38 NL-3563 AV Utrecht Holland

Sweden	Digital Equipment AB P.O. Box 1250 S-17124 Solna 1 Sweden
Denmark	Digital Equipment Corp. AS Kristineberg 3 DK-2100 Copenhagen 0 Denmark
Finland	Digital Equipment Corp. Oy PL 16 SF-02201, Espoo 20 Finland
Norway	Digital Equipment Corp. A/S Pottemakerveien 8 N-Oslo 5 Norway
Austria, East Germany, West Germany, Poland, Hungary, Rumania, Czechoslovkia, Russia, Bulgaria	Digital Equipment Corp. GmbH Rheinstrasse 28 D - 8000 Munich 40 West Germany
Israel	Decsys, Computers Ltd. 4, Yirmiyahu Str. IL-63505 Tel Aviv Israel
Greece, Portugal, Spain, Switzerland, Yugoslavia, (Morocco, Algeria, Tunisia, Cyprus, Turkey, Malta)	Digital Equipment Corp. SA 9, Route des Jeunes Case Postale 191 CH-1211 Geneva 26 Switzerland
Mexico	Digital Equipment de Mexico, S.A. de C.V. Ave. Lopez Mateos 427, 1st. Floor Guadalajara Jalisco Mexico
China	Digital Computer Hong Kong Ltd. 1303-1309 Dominian Ctr. 43-59 Queen's Road East Wanchai Hong Kong

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 •