

pdp11

RT-11 V02C
AUGUST 1976

SOFTWARE PERFORMANCE SUMMARY
DEC-11-XSPSA-G-D

digital

The material in this document is for informational purposes only and is subject to change without notice. DIGITAL Equipment Corporation assumes no responsibility for any errors which may appear in this document. Software binaries and sources are provided only under licenses. The standard terms and conditions, OEM Agreement, and/or quantity discount agreement contain the licenses for all binaries other than for the DECsystem-10.

RT-11 V02C
AUGUST 1976

SOFTWARE PERFORMANCE SUMMARY
DEC-11-XSPSA-G-D

digital

SOFTWARE SERVICES

Copyright 1976 DIGITAL EQUIPMENT CORPORATION

Betty Steinfeld
Editor

Roxanne Alexander
Publications Coordinator

TRADEMARKS OF DIGITAL CORPORATION
MAYNARD, MASSACHUSETTS

COMPUTER LABS	GLC-8	IDACS	CDP	DECSYSTEM-10
DDT	KA10	INDAC-8	RAD-8	MASSBUS
DEC	LAB-K	QUICKPOINT	RSX	TYPESET-11
DECUS	PDP	DIBOL	RSTS	CLASSIC
DIGITAL	SABR	EDUSYSTEM	DECCOMM	DECAL
EDGRIN	TYPESET-8	OMNIBUS	PHA	XVM
FLIP CHIP	UNIBUS	OS/8	LAB-8	
FOCAL	DECTAPE	COMTEX-11	DECSET	

RT-11 V02C
Software Performance Summary

- 1.0 INTRODUCTION
- 2.0 GENERAL FORMAT OF THE ARTICLES
 - 2.1 Filing
 - 2.2 System Components and Related Products
 - 2.3 System and Related Product Documentation
- 3.0 SOFTWARE PERFORMANCE REPORTS
 - 3.1 Software Performance Report Guidelines
- 4.0 RT-11 SYSTEM NOTES

1.0 INTRODUCTION

The Software Performance Summary is a compendium of information which provides a customer with a maintenance notebook on current software documentation and the status of known software problems. The notebook is supplemented with articles in the monthly Digital Software News which should be filed in the appropriate sections of the Software Performance Summary.

2.0 GENERAL FORMAT OF THE ARTICLES

Each article is formatted so that the subject is easily recognized. Figure 1 shows an overall example of the format.

Digital Software News, June 1975

Title and SPR Number, if Appropriate

The text of the article may take several forms:

1. An article of a general nature; i.e., clarification of the use of a software feature.
2. Problem description. Here the problem and its effects will be described and followed by a solution or disposition. When no solution is available, known methods of avoiding the problem will be given when possible.

SOFTWARE PRODUCT (1)	VERSION (1A)	
COMPONENT (2)	VERSION (2A)	
SUBPROGRAM OR ADDITIONAL INFORMATION (2B)	SEQUENCE (3)	PAGE OF (3A)
NEW (4)	REPLACEMENT ARTICLE (5)	ORIGINAL DATE (5A)

Figure 1. Article format.

2.1 Filing

This introductory material should be filed at the beginning of the notebook.

A system has been devised to help you file each article in its proper place. The key is Figure 2 below.

SOFTWARE PRODUCT (1)		VERSION (1A)	
COMPONENT (2)		VERSION (2A)	
SUBPROGRAM OR ADDITIONAL INFORMATION (2B)		SEQUENCE (3)	PAGE OF (3A)
NEW (4)	REPLACEMENT ARTICLE (5)	ORIGINAL DATE (5A)	

Figure 2. Coding block.

Each month the Digital Software News should be taken apart and inserted into the pages in your notebook.

First, the articles are classified by software product (1). All articles should be filed under the appropriate major heading.

Secondly, the software product is broken down by its components (2). See section 2.2 for the list of software components.

Finally, the article is referenced by sequence number (3). As an article is added to each component, it is assigned the next higher sequence number. An asterisk (*) to the right of the sequence number signifies that the article contains a patch. A dagger (†) to the right of the sequence number signifies that the article is a replacement.

Additional information in the coding block is presented to further clarify the article and is not specifically for filing:

- (1A) Version of the software product.
- (2A) Version number of the component.
- (2B) Other information helpful to the user.
- (3A) Page number and pages in the article.
- (4) An "X" in this block indicates a new article.
- (5) A number in this block indicates an article republished for revision or correction and specifies the number of the revision. For example, the second revision of an article which originally appeared in June 1974 is shown in Figure 3.

(5A) Original date of a revised article.

NEW <input type="checkbox"/>	REPLACEMENT ARTICLE <input checked="" type="checkbox"/> 2	ORIGINAL DATE June 1974
---------------------------------	--	----------------------------

Figure 3. Coding block showing the second revision.

2.2 System Components and Related Products

RT-11	ASEMBL	VS02-12
	BATCH	V01-04
	CBUILD	V01-01
	CREF	V01-04
	DUMP	V02-02
	EDIT	V02-12
	EXPAND	V02-02
	FILEX	V02-02
	HANDLERS	N/A
	LIBR	V03-03
	LINK	V04-04
	MACRO	VM02-12
	MONITOR	RT-11 SJ V02C-02
		RT-11 FB V02C-02
	ODT	V01-02
	PATCH	V01-02
	PATCHO	N/A
	PIP	V04-06
	SRCCOM	V01-03
	SYSLIB	V6
	SYSMAC.8K	N/A
	PTBUILD	V02-01
	MBUILD	V02-03
	MSBOOT	V01-05
	SYSTEM	
	INFORMATION	N/A
BASIC/RT-11	None	V01B-01
BASIC/RT-11 Ext.	None	V01
FORTRAN/RT-11	None	V01C
GAMMA-11 F/B		V02
LA-11	SPARTA	V03
	Modules	TTYR11
		QA
		OUTF
		FFT
		ADSAM
		GTDISP
		GENS
		VRDISP

Other sources to generate SPARTA

IV-11 Plotting Package

MU BASIC/RT-11

SSP-11/RT-11

RT-11 Software Performance Summary

2.3 System and Related Product Documentation

RT-11 Documentation Directory	DEC-11-ORDDA-A-D
RT-11 System Generation Manual	DEC-11-ORGMA-A-D
RT-11 System Message Manual	DEC-11-ORMEA-A-D
RT-11 System Release Notes	DEC-11-ORNRA-A-D
RT-11 Reference Manual, Addendum 1, Update 2	DEC-11-ORUGA-C-D, DN1, DN2
RT-11 System Reference Card	DEC-11-ORRCA-C-D
RT-11 Software Support Manual, Update 1	DEC-11-ORPGA-B-D, DN1
BASIC/RT-11 Language Reference Manual	DEC-11-LBACA-D-D, DN2
BASIC/RT-11 Release Notes	DEC-11-LBRNA-A-D
FOCAL-11 User's Manual	DEC-11-LFOCA-F-D
FORTTRAN/RT-11 Release Notes	DEC-11-LFRNA-A-D
PDP-11 FORTRAN Language Reference Manual	DEC-11-LFLRA-C-D
RT-11/RSTS/E FORTRAN IV User's Guide	DEC-11-LRRUA-A-D
GAMMA-11 Reference Manual	DEC-11-MGFBA-A-D
GAMMA-11 Reference Card	DEC-11-MGCSA-A-D
SPARTA Reference Card	DEC-11-ALRCA-C-D
Laboratory Applications-11 Release Notes	DEC-11-ALRNA-A-D
Laboratory Applications-11 Program Manual	DEC-11-ALPMA-A-D
Laboratory Applications-11 Module Manual	DEC-11-ALMMA-A-D
LV-11 Plotting Package BASIC/RT-11 Manual	DEC-11-LBACA-D-DN1
BASIC-11 Language Reference Manual	DEC-11-LIBBA-B-D
MU BASIC/RT-11 User's Manual	DEC-11-LIBRA-A-D
MU BASIC System Installation Guide	DEC-11-LIBMA-A-D
Scientific Subroutine Manual	DEC-11-ARSMA-A-D

3.0 SOFTWARE PERFORMANCE REPORTS

Each new installation is provided with Software Performance Report (SPR) forms. The SPR form enables users to suggest enhancements to or report problems with Digital Equipment Corporation software or documentation. When a problem is encountered, an SPR should be completed and mailed to the local SPR Center.

Responses will be sent to the name and address appearing on the form. Additional SPR forms may be obtained by writing to the local SPR Center. (See the list inside the back cover.)

3.1 Software Performance Report Guidelines

Please fill out the SPR form completely. It is important that the machine configuration be included--the system device type, the amount of core in use, and all the peripherals on the machine. If your system has LPS, include all the options it contains (LPSKW, LPSAM-SG, SPSAD-NP, etc.). The name and version number of the operating system and the program in use, if any, is absolutely essential.

An adequate and clear description of the problem is very important and will certainly speed the processing of the SPR. Supplement the description by including the Teletype printout, pictures of the display if involved, and actual copy of the user programs that show the problems. The printout should ideally start from the system bootstrap and include the least number of commands or instructions that cause the problem.

Before submitting a Software Performance Report, the user should review the Digital Software News to ensure that the problem has not already been published.

4.0 RT-11 SYSTEM NOTES

Before including the patches found in the Software Performance Summary, please consult the RT-11 System Release Notes (DEC-11-ORNRA-A-D), the RT-11 FORTRAN Release Notes (DEC-11-LFRNA-A-D), the BASIC/RT-11 Release Notes (DEC-11-LBRNA-A-D), and any other Release Note documents associated with optional RT-11 software. They contain additional patches which should be installed in the system and articles describing restrictions and operational procedures.

This is a complete listing of all RT-11 V02C and related product articles that have appeared through the August 1976 issue of the PDP-11 Digital Software News. Missing Sequences pertain to other versions of RT-11 and related products.

4.0 SYSTEM NOTES

Sequence

BASIC/RT-11 V01B

HALT OR OTHER SYSTEM FAILURE AFTER USE OF BASIC EXTENSIONS	1
BUFFER STORAGE OVERFLOW ERROR	2*
BASIC/RT-11 USED WITH EAE HARDWARE INCORRECTLY HANDLES THE VALUE -32768	3*
CALL TO DFIX CAUSES DISPLAY TO GO BLANK	4*

COS-350 V02

CURSOR POSITIONING QUIRKS	2
ONERROR TRAP	3*
DEVICE HANDLER NOT RESIDENT	4
SEND MESSAGE DOCUMENTATION ERROR	5

DICOMP

MOUNT MESSAGE OCCURS	2*
EXTRANEIOUS CHARACTERS WITHOUT BEING FLAGGED ERROR	3*

SORTM

FILE DESCRIPTION OF TAGSORT OUTPUT	4
REMOVAL OF FILE NAMING RESTRICTION UNDER TSD	5*
COUNT CAUSES ERROR 16	6*
TAGSORT MAY FAIL	7*
CONTROL COUNTS	8*

TSD

ERROR WHEN FILES OPEN IN UPDATE MODE	11*
SUPERSEDING EXISTING FILE ERROR	12*
RANDOM RECORD READ	13*
ONERROR TRAP	14*
SINGLE USER IN UPDATE MODE	16*
SUPERSEDING EXISTING FILE, EXCEEDING MAXIMUM NUMBER DEVICES, AND ACCEPT STATEMENT VIA A FILE	17*
BAD BINARY FILE ERROR MESSAGE ON GOOD FILES	18*
OPTIMIZATION OF READS AND WRITES	19*
READING FILE OF LENGTH 0 RESULTS IN LINE PRINTER HANG	21*
LOCKED BLOCKS	22*
LINEPRINTER STOPS EXECUTION IF TURNED OFF	23*

FOCAL/RT-11 V01B

FOR COMMAND WITHOUT AN ARGUMENT	1
OPERATE COMMAND CAUSES ERROR	4
FCLK ROUTINE GIVES INCORRECT TIME	5*

*Article contains a patch.

FORTRAN/RT-11 V01C-03

CLARIFICATION: INTERFACING ASSEMBLY LANGUAGE ROUTINES TO FORTRAN	1
FLOATING MULTIPLY FAILS TO DETECT UNDERFLOW IN NHD VERSION OF OTS	42*
COMPILING MULTIPLE PROGRAM UNITS FROM A SINGLE CASSETTE FILE	43
STAND-ALONE FORTRAN STACK USAGE	44
WRITING ON READ-ONLY FILE	46
WRITING BEYOND END OF RANDOM ACCESS FILE	47
ASYNCHRONOUS I/O, EVENT DRIVEN I/O, AND FORTRAN PROGRAMS	49*
OBJECT TIME FORMATTING WITH H FORMAT SPECIFICATION, FORMATTING	
RECORD WRITING GREATER THAN 132 CHARACTERS IN LENGTH, MAY FAIL	51*
OBJECT TIME ENCODE/DECODE	52*
CLARIFICATION OF I/O LIST ELEMENTS	53
CALL ASSIGN WITH FILE NAME TERMINATED WITH SPACES ABORTS	55*
I-FORMAT CONVERSION ERROR	56*
J=J-J GIVES INCORRECT RESULTS	57*
LISTING FILES DIRECTED TO MAGTAPE	58*
CALL CLOSE ON INACTIVE UNIT	59
ARITHMETIC STATEMENT FUNCTIONS WITH NO ARGUMENTS	60
COMPUTED GO TO	61
CLARIFICATION: COMPARING ASCII DATA ITEMS	62
IBEF NOT PROPERLY DECREMENTED	63
LPS DEVICE CONFLICT CAUSED BY CALL SETR AFTER CALL RTS	64

LA-11 V03

PEAK.MAC	
WIDE PEAKS	1*
PEAK PROBLEMS AND CORRECTIONS	2*

SPARTA

LPS AND AR-11 VECTOR AND STATUS REGISTER	1
USING SPARTA AND FLOATING POINT BUFFERS	2
AR-11 TIMING PROBLEMS WITH ADSAM AND SPARTA	3*
FFT SCALING CORRECTION	4
SCALE FACTOR CORRECTION FOR SPARTA COMMANDS FAC AND FCC	5*
DATA DISPLAYS USING LA-11	6
DATA PREPARATION FOR SPARTA COMMANDS FAC AND FCC	7
SPARTA CORRECTIONS FOR POINT-PLOT DISPLAY	8*
ADDING COMMANDS TO SPARTA	9*
CORRECTION FOR THE DFV COMMAND WITH POINT PLOT DISPLAY (VR14\$)	10*
GENERAL SUBROUTINE MODULE FOR EAE	11*

THRU

HOW TO START DATA ACQUISITION WHEN CSTART EQUALS ZERO	1
---	---

MU BASIC/RT-11 V01-01

BUILDING MU BASIC/RT-11 UNDER RT-11 V02C	1*
REMOTE TERMINAL SUPPORT ON MODEMS	2*
OVERLAY... LINE WORKS INCORRECTLY	3

REMOTE/RT-11 V01

SCHEDULER DOES NOT PROPERLY SET PROCESSOR PRIORITY	1*
NOEDIT- Ø HALTS	2*
NURSERS=1 STAYS IN A FILE MESSAGE LOOP	3*
INCORRECT SWAP AREA ALLOCATION FOR FOUR OR MORE USERS	4*

*Article contains a patch.

REMOTE/RT-11 V01

REBOOT FROM STAELLITE DURING EDIT HANGS HOST	5*
HARD ERROR ON LOOKUP IS FATAL	6*
SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL	7*
ONE SECOND TIMER FOR LINE TIMEOUTS IS SET INCORRECTLY	8*
LINE FEEDS MAY CAUSE SYSTEM ERRORS--ASSEMBLY ERROR WITH DIAL AND NODDC	9*
PROPER GENERATION OF REMOTE IS DEPENDENT ON MODULE ORDER	10*
ASCII CODES 173 AND 174 DO NOT PRINT	11*
IMPROPER FILLER HANDLING FOR VT05	12*
SYSTEM CRASHES IF RUN IN FOREGROUND WITHOUT /N	13*

RT-11 V02C-02

EDIT	
EDIT ERRORS OCCUR WHEN THE FIRST CHARACTER IN THE TEXT BUFFER IS A LINE FEED	5*
CHARACTER IS LOST WHEN EXECUTING A READ COMMAND	6*
EXTRA TEXT APPENDED TO EDIT OUTPUT	7*
HANDLER	
PATCHING LP VECTOR	2*
LP HANGS SYSTEM	3
LINK	
PERFORMANCE IMPROVEMENT IN LINKER	1*
PATCHO	
?ERR 61 MESSAGE FROM PATCHO	2*
MONITOR	
ERROR IN F/B HRESET CODE	13*
FIS EXCEPTION ERROR	17*
MIDNIGHT ROLLOVER FOR F/B MONITOR MALFUNCTIONS	19*
SYSTEM INFORMATION	
LOW SPEED READER SUPPORT	1*

MANUALS

AR-11 USER'S GUIDE (DEC-11-HARUG-B-D)	
PROGRAM EXAMPLE ERROR IN AR11 USER'S GUIDE	1*
RT-11 SYSTEM GENERATION MANUAL (DEC-11-ORGMA-A-D)	
REPLACEMENT PAGES	1
RT-11 SYSTEM MESSAGE MANUAL (DEC-11-ORMEA-A-D)	
REPLACEMENT PAGES	1
RT-11 SYSTEM REFERENCE MANUAL (DEC-11-ORUGA-C-D, DN1, DN2)	
REPLACEMENT PAGES	5
RT-11 SYSTEM RELEASE NOTES (DEC-11-ORNRA-A-D)	
REPLACEMENT PAGES	1

*Article contains a patch.

Halt or Other System Failure After Use of BASIC Extensions

PROBLEM:

The BASIC Extensions will not always turn off all real-time and/or display devices on exit when multiple devices are used. This may result in a halt or other system failure when another program is run.

ANALYSIS:

The BASIC Extensions software does not allow for multiple real-time and/or display devices to be turned off automatically on exit. It does, however, turn off or stop the last device initialized.

SOLUTION:

All programs utilizing more than one real-time and/or display device should stop or turn off these devices before the program executes a STOP statement.

If the program terminates abnormally with a BASIC error message, issue an immediate mode CALL "DSTP" in the case of the display or CALL "SETR" with a rate of zero for the LPS or AR-11. If the program is terminated with a CTRL/C, attempt to reenter BASIC and issue the previously described statements.

The new BASIC Extensions to be released early in 1976 will handle this problem correctly.

SOFTWARE PRODUCT BASIC/RT-11	VERSION V01B-01
COMPONENT N/A	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE PAGE 1 OF 1 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>
ORIGINAL DATE January 1976	

Buffer Storage Overflow Error

PROBLEM:

BASIC/RT-11 on systems with more than 16K of core under certain circumstances may improperly produce a Buffer Storage Overflow (?BSO) error.

SOLUTION:

Four lines after the tag FREEGET in BASICR is the code BVS FREER which should be BCC FREER.

The following patches fix the problem both in save modules and object modules. Underlined characters are output by the system. Characters that are not underlined are to be typed by the user. <CR> indicates carriage return. These patches change the version of BASIC/RT-11 to V01B-02A.

Enter the following patch for BASIC.SAV. The same patch applies to BASGT.SAV, BASLPS.SAV, and BGTLP.SAV. Substitute the appropriate name for BASIC.SAV in the following patch to fix these modules.

```
.R PATCH<CR>

PATCH V01-02

FILE NAME--
BASIC.SAV<CR>
*500;0R
*0,6/ 40      101<CR>
*0,2436/     102407  103007<CR>
*E
```

SOFTWARE PRODUCT BASIC/RT-11		VERSION V01B-01	
COMPONENT N/A		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 2*	PAGE 1 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

Buffer Storage Overflow Error

Enter the following patch for BAS8K.SAV. The ?BOTTOM ADDR WRONG? should be ignored.

.R PATCH<CR>

PATCH V01-02

FILE NAME--

BAS8K.SAV/O<CR>

?BOTTOM ADDR WRONG?

*1136;ØR

*Ø,6/ 4Ø 1Ø1<CR>

*Ø,2126/ 1Ø24Ø7 1Ø3ØØ7<CR>

*E

Enter the following patch for BASGTO.SAV. The same patch applies to BGTLPO.SAV. The ?BOTTOM ADDR WRONG? message should be ignored.

.R PATCH<CR>

PATCH V01-02

FILE NAME--

BASGTO.SAV/O<CR>

?BOTTOM ADDR WRONG?

*1414;ØR

*Ø,6/ 4Ø 1Ø1<CR>

*Ø,2436/ 1Ø24Ø7 103007<CR>

*E

SOFTWARE PRODUCT BASIC/RT-11		VERSION V01B-01	
COMPONENT N/A		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 2*	PAGE 2 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

Buffer Storage Overflow Error

Enter the following patch to fix BASICR.OBJ, the object module of BASICR with string support.

```
.R PATCHO<CR>
*DEC
*OPEN
ENTER INPUT FILE*BASICR.OBJ
ENTER OUTPUT FILE*BASICR.OBJ
*WORD BASICR+6=1Ø1<CR>
*WORD BASICR+2436=#-74771<CR>
*EXIT<CR>
ENTER CHECKSUM: 164463
STOP--
```

Enter the following patch to fix BASNSR.OBJ, the object module of BASICR with no string support.

```
.R PATCHO<CR>
*DEC<CR>
*OPEN<CR>
ENTER INPUT FILE*BASNSR.OBJ<CR>
ENTER OUTPUT FILE*BASNSR.OBJ<CR>
*WORD BASICR+6=#1Ø1<CR>
*WORD BASICR+2126=#-74771<CR>
*EXIT<CR>
ENTER CHECKSUM: 163643
STOP--
```

SOFTWARE PRODUCT BASIC/RT-11		VERSION V01B-01	
COMPONENT N/A		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 2*	PAGE 3 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

BASIC/RT-11 USED WITH EAE HARDWARE INCORRECTLY HANDLES THE VALUE
-32768 (SPR 11-6259 SJ)

PROBLEM: BASIC/RT-11 V01B-02 when used with EAE hardware does not
handle the value -32768 correctly.

DISPOSITION: Make the following patch to the object module FPMP.EAE,
then use LINK to create a new BASIC.SAV.

NOTE: Underlined characters are output by the system. Characters
that are not underlined are input by the user. <CR> implies carriage
return.

```
._R PATCHO <CR>  
*OPEN <CR>  
ENTER INPUT FILE*FPMP.EAE <CR>  
ENTER OUTPUT FILE*FPMP.EAE <CR>  
*WORD 2230=#167 <CR>  
*WORD 2232=#2104 <CR>  
*WORD 4340=#1001 <CR>  
*WORD 4342=#134 <CR>  
*WORD 4344=#5401 <CR>  
*WORD 4346=#-75376 <CR>  
*WORD 4350=#167 <CR>  
*WORD 4352=#-2120 <CR>  
*WORD 4354=#5066 <CR>  
*WORD 4356=#2 <CR>  
*WORD 4360=#12716 <CR>  
*WORD 4362=#-34000 <CR>  
*WORD 4364=#134 <CR>  
*EXIT <CR>  
ENTER CHECKSUM: 176550 <CR>  
  
STOP --
```


CALL TO DFIX CAUSES DISPLAY TO GO BLANK (SPR 11-4839 DA)

When DFIX is called with the GT ON option, an argument that is too large for core causes the display screen to go blank.

Enter the following patch:

Relink Basic

```
R PIP
GTB.OLD=GTB.OBJ

R PATCHO
*OPEN
ENTER INPUT FILE GTB.OBJ
ENTER OUTPUT FILE GTB.OBJ
*WORD 2416=#4767
*WORD 2420=#6762
*WORD 11404=#13746
*WORD 11406=#54
*WORD 11410=#62716
*WORD 11412=#302
*WORD 11414=#5736
*WORD 11416=#207
*EXIT
ENTER CHECKSUM: 35376
```

STOP ---

```
R PIP
GTC.OLD=GTC.OBJ
```

```
R PATCHO
*OPEN
ENTER INPUT FILE GTC.OBJ
ENTER OUTPUT FILE GTC.OBJ
*WORD 2540=ZSTOPC+0
*EXIT
ENTER CHECKSUM: 144240
```

STOP ---

CURSOR POSITIONING QUIRKS

PROBLEM:

1. When doing cursor positioning for some time, the system hangs for two or three minutes before proceeding.
2. An apparent problem may appear in the RT-11 F/B monitor. A carriage return/linefeed may be inserted into the output characters every so often.

DISPOSITION:

1. The cause of this problem is in RT-11. While the exact cause is not known, it apparently occurs after outputting 32,768 characters to the terminal without doing a carriage return/linefeed. The problem can be avoided by outputting a carriage return/linefeed occasionally in the program.
2. This is because the F/B monitor inserts an automatic carriage return/linefeed at the end of what it thinks is one line. This feature may be disabled by typing `.SET TTY NO CRLF`.

ONERROR TRAP

PROBLEM: ONERROR is used to trap a soft error in a routine which has been compiled with /O option. If the soft error occurs following an internal subroutine CALL, the subsequent RETURN statement will cause unpredictable results. This problem exists in both single and multiuser DIBOL programs.

DISPOSITION: The following patch corrects this problem and changes the version number to VA02-01A.

```
.R PIP
*DIRT.OLD=DIRT.OBJ/R
*^C

.R PATCHO
*OPEN
ENTER INPUT FILE *DIRT.OLD
ENTER OUTPUT FILE *DIRT.OBJ
*POINT #DIRT
*WORD #DIRT+1162=#-600
*BYTE #DIRT+5701=#101
*EXIT
ENTER CHECKSUM: 107057

STOP --

.R LIBR
*DIBOL=DATE, TIME, DDTX, DIRT, MATH, IO/C
*JOB, ELONG, DECML, ASCII, VERSN, FLAGS/C
*ERROR, RSTAT, DELET, MONEY

*^C
```

Following the patch, it will be necessary to relink any single user program in which this condition may occur.

DEVICE HANDLER NOT RESIDENT (SY)

PROBLEM: Attempting to look up a file from a floppy in the second set of floppy drives can result in the PROGRAM NOT FOUND message being displayed.

DISPOSITION: This is because the device handler is not resident. After each initialization of the system, using the load command will force the residency of the handler and prevent this problem.

Example: .LOA DY

SEND MESSAGE DOCUMENTATION ERROR (SY)

Page 3-48 of the DIBOL-11 Language Reference Manual incorrectly states:

When used in a single-user environment, only one message can be sent and the message must not exceed 150 characters. Further, SEND can be used only for sending a message to a program that is called using the optional argument to the STOP statement (i.e.; chaining technique).

The problem occurs when the chain to the next job is done. Some portion of the end of a 150-character message will be lost. When a message is sent, it is placed at the bottom of the system stack area, because this is the only area of core that has any chance of remaining intact during a chain operation. However, in the process of performing a lookup to do the chain or if further program processing before the chain causes the stack to be pushed down into the message area after the time that the message has been sent, the end of the message will be corrupted.

The 150-character limitation was an estimate of the theoretical message size, determined before the chain feature was functioning. Doing a simple SEND message and then chain program, 124-characters of a 150-character message were preserved. This is probably the maximum that can be sent. Depending upon the program, the maximum message size will vary. No absolute figures can be given with complete assurance that they will always be true. However, 90 to 100 characters ought to be preserved even in the worst cases.

MOUNT MESSAGE OCCURS

PROBLEM: It has been suggested that a CTRL/Z be output as the last character of a compiler listing by those who print the listings with the DIBOL line printer spooler.

Currently a mount message will occur since DIBOL interprets any file without a CTRL/Z as a multivolume file.

DISPOSITION: This patch corrects this inconvenience and changes the DICOMP version number to V02-00A.

```
.R PIP
*INIT. OLD=INIT. OBJ/R
*ENCOD. OLD=ENCOD. OBJ/R
*TC
```

```
.R PATCHO
*OPEN
ENTER INPUT FILE *INIT. OLD
ENTER OUTPUT FILE *INIT. OBJ
*POINT INIT
*BYTE INIT+3003=#101
*EXIT
ENTER CHECKSUM: 31405
```

STOP --

```
.R PATCHO
*OPEN
ENTER INPUT FILE *ENCOD. OLD
ENTER OUTPUT FILE *ENCOD. OBJ
*POINT SYMBOL
*WORD ENCODS+2746=#511
*WORD ENCODS+3172=#4767
*WORD ENCODS+3174=%OUTCH3+0
*WORD ENCODS+3176=#-65055
*WORD ENCODS+3200=#32
*WORD ENCODS+3202=#5367
*WORD ENCODS+3204=%PCOUNT+0
*WORD ENCODS+3206=#661
*EXIT
```

COS-350, V02-00
DICOMP, V02-00

Seq 2
1 of 2

MOUNT MESSAGE OCCURS

ENTER CHECKSUM: 54621

STOP --

. R LINK
*DICOMP=ROOT/C
*INIT/O:1/C
*DATA/O:1/C
*PROC/O:1/C
*ENCOO/O:1/C
*DILINK/O:2/C
*DICERR/O:2/C
*FATAL/O:2

EXTRANEIOUS CHARACTERS WITHOUT BEING FLAGGED ERROR

PROBLEM: In DICOMP V02-00A, the left side of an expression can have extraneous characters without being flagged as an error. Currently, A(1,1)*L=B will compile without error and run as though the line reads A(1,1)=B.

DISPOSITION: The following patch will correct this problem by generating an error message for cases such as this. This patch will also change the DICOMP version number to V02-00B.

```
. R PIP
*INIT. OLD=INIT. OBJ/R
*PROC. OLD=PROC. OBJ/R
*TC

. R PATCHO
*OPEN
ENTER INPUT FILE *PROC. OLD
ENTER OUTPUT FILE *PROC. OBJ
*POINT PROCED
*WORD PROC+556=%XPROC+6246
*WORD PROC+6246=#4767
*WORD PROC+6250=%IGNORF+0
*WORD PROC+6252=#-53351
*WORD PROC+6254=#75
*WORD PROC+6256=#1002
*WORD PROC+6260=#167
*WORD PROC+6262=%XPROC+1344
*WORD PROC+6264=#-73343
*EXIT
ENTER CHECKSUM: 16031

STOP --

. R PATCHO
*OPEN
ENTER INPUT FILE *INIT. OLD
ENTER OUTPUT FILE *INIT. OBJ
*POINT INIT
*BYTE INIT+3003=#102
*EXIT
ENTER CHECKSUM: 31407

STOP --
```

EXTRANEIOUS CHARACTERS WITHOUT BEING FLAGGED ERROR

```
. R LINK  
*DICOMP=ROOT/C  
*INIT/0:1/C  
*DATA/0:1/C  
*PROC/0:1/C  
*ENCOD/0:1/C  
*DILINK/0:2/C  
*DICERR/0:2/C  
*FATAL/0:2
```

File Description of TAGSORT Output

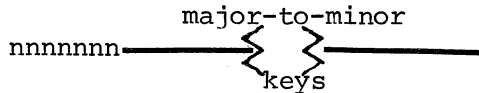
This is a further description of the output from the TAGS statements as described in the COS-350 System Reference Manual (DEC-11-OCSRA-A-D), page 10-6.

1. The TAGS:LIST statement produces a list of relative record numbers only. The output is a seven character record of the form:

bbbbbn

where: b = a blank character
 n = a numerical character

2. The TAGS:INDEX statement produces a list of relative record numbers plus the keys specified in the sort. The first seven digits represent the relative record number, and the remaining characters represent the sort fields in descending order from major to minor keys. The length of the sort fields is determined by the user's definition of the keys. The form of the TAGS:INDEX output is:



where: n = a numerical character representing the relative record numbers

major key = the outermost or highest level of sorting

minor key = descending keys down to the innermost or lowest level of sorting.

SOFTWARE PRODUCT COS-350	VERSION V02-00	
COMPONENT SORTM	VERSION V02-00	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 4	PAGE OF 1 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976

Removal Of File Naming Restriction Under TSD (CJ)

PROBLEM:

SORT does not allow output file to be the same name as the input file.

SOLUTION:

This edit will allow the SORT output to be named the same as the input file. Under Time Shared DIBOL, the following patch removes this restriction and updates the component levels accordingly:

SORTM.DBL V01-03C becomes SORTM.DBL V01-04D.
 SORTM.DBL V02-00 becomes SORTM.DBL V02-01A.

```
. R EDIT
*ERSORTM.DBL$$
*EWSORTM.DBL$$
*FSKPVOL,$-6A6L$$
;
;NOW WE REOPEN THE SOURCE FILE TO GET THE RECORD POINTER
; TO THE HEAD OF THE FILE FOR SEQUENTIAL READING.
;

                OPEN(SRC,I,INPUT)                OPEN SOURCE FILE.
*6S$$
*6K$$
*L$$
SKPVOL, IF(MAXREC.LT.ULEN)ULEN=MAXREC    SHORTEN STRING FOR
*FDETCH#0JL$$
. IFDEF DETCH                                ; IF DETACHED GIVE UP THE TERMINAL.
*U$$
*-10L5L$$
                CALL    OPNDST                    ; OPEN OUTPUT FILE.
                                                    ; TO MAKE SURE WE HAVE EVERY LITTLE BIT OF
                                                    ; SPACE REQUIRED BEFORE STARTING.
;
;NOW WE REOPEN THE SOURCE FILE TO GET THE RECORD POINTER
; TO THE HEAD OF THE FILE FOR SEQUENTIAL READING.
;
```

SOFTWARE PRODUCT COS-350		VERSION V01, V02	
COMPONENT SORTM.DBL		VERSION V01-03C, V02-01A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 5*	PAGE 1 OF 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

Removal Of File Naming Restriction Under TSD (CJ)

```

    OPEN(SRC, I, INPUT)          ; OPEN SOURCE FILE.
  . IFDEF DETCH                  ; IF DETACHED GIVE UP THE TERMINAL.
    DETACH
  . ENDC
  
```

*EX**

SOFTWARE PRODUCT COS-350		VERSION V01, V02	
COMPONENT SORTM.DBL		VERSION V01-03C, V02-01A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 5*	PAGE OF 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

COUNT CAUSES ERROR 16

PROBLEM: When the count is supplied, ERROR 16 (Channel In Use) occurs.

DISPOSITION: This patch corrects the problem and changes the version to V02-02B.

```
R EDIT
*ERSORTM.DBL##
*EWSORTM.DBL##
*FV02-0#C2#2JCB#0JL##
;          SORTM      V02-02- B
*FSKPVOL.#0J-2L2L7D#-2AISKPVOL.#0J-3L3L##
          CLOSE(SRC)          ;CLOSE THE SOURCE FILE.
;
SKPVOL, IF(MAXREC.LT.ULEN)ULEN=MAXREC  ;SHORTEN STRING FOR
          ;SMALL FILE.
;
;
X30,      MAXREC=IFAV          ;SET MAX RECORD COUNT.
SKPVOL,   CLOSE(SRC)          ;CLOSE THE SOURCE FILE.
          IF(MAXREC.LT.ULEN)ULEN=MAXREC  ;SHORTEN STRING FOR
*EX##
```

NOTE: This patch is an extension of SORTM.DBL, sequence 5, entitled "REMOVAL OF FILE NAMING RESTRICTION UNDER TSD," and must be installed after it.

TAGSORT MAY FAIL

PROBLEM: The tagsort may fail when the total length of the sort keys is more than nine characters.

Resulting error messages may be VOLUME ESTIMATE EXCEEDED when none was supplied or subscript error messages.

A general loss of control occurs because of a misplaced assignment statement which overwrites some of the control variables.

DISPOSITION: This patch relocates the statement and deletes it from its current position and changes the version number to V02-03C.

```
R EDIT
*ERSORTM.DBL##
*EWSORTM.DBL##
→ *FV02-0#C3#2JCC#0JL##
;          SORTM    V02-03- C
→ *FX69C.#2A0JL##
          . ENDC
→ *I          HOLD(N)=RKEY    ;SAVE IN HOLD AREA
   $$
→ *-5L5L##
          . IFDEF  ZK
X69B,    READS(N,RKEY,X200)
X69C,    IF(RKEY.EQ.EOSMRK)GO TO X72
          CALL    TKSARE    ;EXTRACT KEYS AND SAVE SECONDARY RECORD.
          HOLD(N)=RKEY    ;SAVE IN HOLD AREA
          . ENDC

          . IFNDEF  ZK
X69B,    READS(N,RECORD,X200)
X69C,    IF(RECORD.EQ.EOSMRK)GO TO X72
→ *FMSELECT.#-7AL#-4L4L##
          HOLD(N)=RKEY
          . ENDC
          . IFDEF  KEYS
          KTBL8(N)=RKEY8
          . ENDC
          HOLD(N)=RKEY

          RETURN
          . ENDC
```

COS-350, V02
SORTM.DBL, V02-02B

Seq 7*
1 of 2

TAGSORT MAY FAIL

```
→ *K##  
→ *-4L4L##  
  .ENDC  
  .IFDEF  KEYS  
    KTBL8(N)=RKEY8  
  .ENDC  
  
    RETURN  
  .ENDC  
  ,+++++++  
→ *EX##
```

CONTROL COUNTS

PROBLEM: SORTM.DBL V02-00 expands the control counts to six digit fields. Two clear instructions were missed in the editing of these changes. When using more than four work units, it is possible to get VOLUME ESTIMATE EXCEEDED.

DISPOSITION: This change extends the length of the clear to cover all of the counters and changes the version number to V02-04D.

```

R EDIT
*ERSORTM.DBL$$
*EWSORTM.DBL$$
*FV02-0#0JL$$
;
;          SORTM   V02-03- C
*G-0#C4#2JCD#0JL$$
;
;          SORTM   V02-04- D
*FX31, #10A2LG1, #2DI54#AG1, #2DI54#0J-4L4L$$
      CNTA(1,27)=
      CNTB(1,27)=
      NPASS=                ; CLEAR PASS OR STAGE COUNTER.
      SUMB=                 ; CLEAR TEMP ACCUM.
                          ; CLEAR DISTRIBUTION TABLE FOR PRE-SORT.
      CNTA(1,54)=
      CNTB(1,54)=
;
; CALCULATE THE FIRST DISPLACEMENT VALUE TO BE USED IN
; THE DIMINISHING DISPLACEMENT SORT AT PSORT.
*EX$$
```


Error When Files Open in UPDATE Mode

PROBLEM:

A given file is currently open both in Input and Update mode. The use of the update mode is then terminated by closing the file (leaving the file open only in INPUT mode). At this point, the core being used for record locking information is then deallocated. If while in this state, the same file is now opened up in UPDATE Mode, TSD falsely uses the old record locking information area and fails to request a new area in core.

SOLUTION:

This patch corrects this problem and changes the version number to TSD VB02-00A.

TSD users should run SYSGEN after putting in the patches.

```
. R PIP
*DIO. OLD=DIO. OBJ/R
*DT0. OLD=DT0. OBJ/R
*^C

. R PATCH0
*OPEN
ENTER INPUT FILE *DIO. OLD
ENTER OUTPUT FILE *DIO. OBJ
*POINT #DIO
*WORD #DIO+776=#2534
*EXIT
ENTER CHECKSUM: 22500

STOP --

. R PATCH0
*OPEN
```

SOFTWARE PRODUCT COS-350		VERSION V02	
COMPONENT TSD		VERSION VB02-00	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 11*	PAGE 1 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

Error When Files Open in UPDATE Mode

```

ENTER INPUT FILE *DT0.OLD
ENTER OUTPUT FILE *DT0.OBJ
*POINT DT0
*BYTE DT0+4301=#101
*EXIT
ENTER CHECKSUM: 63243
  
```

STOP --

.ASS LP:LOG

.ASS LP:LST

.LOAD BA

.LOAD LP

.R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: <2>

DO YOU WANT REMOTE TERMINAL SUPPORT: <N>

ARE THEY ALL STANDARD TERMINALS: <Y>

NUMBER OF PROGRAMS TO RUN: <4>

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: <Y>

NUMBER OF MESSAGES IN MEMORY: <8>

NUMBER OF DEVICES ALLOWED: <4>

SOFTWARE PRODUCT COS-350		VERSION V02	
COMPONENT TSD		VERSION VB02-00	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 11*	PAGE 2 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

Error When Files Open in UPDATE Mode

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USR IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

SOFTWARE PRODUCT COS-350	VERSION V02
COMPONENT TSD	VERSION VB02-00
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE PAGE 11* 3 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>
	ORIGINAL DATE April 1976

SUPERSEDING EXISTING FILE Error

PROBLEM:

After attempting to trap a SUPERSEDING EXISTING FILE error with processing being continued in the manner described below, the following has been discovered:

1. If the ONERROR trap routine attempts to close the existing file channel, the system hangs.
2. Any further attempt to use the file or device by another program results in FILE IN USE or DEVICE IN USE.
3. If the current program tries to open the given file for input, the result is a CHANNEL IN USE ERROR.

SOLUTION:

This patch corrects the problem and changes the version number to TSD VB02-00B.

```

CC
.R PIP
*DI0.OLD=DI0.OBJ/R
*DT0.OLD=DT0.OBJ/R
*CC

.R PATCH0
*OPEN
ENTER INPUT FILE *DI0.OLD
ENTER OUTPUT FILE *DI0.OBJ

*POINT #DI0
*WORD #DI0+1550=#4767
*WORD #DI0+1552=X#DI0+11256
*WORD #DI0+11256=#1403
  
```

SOFTWARE PRODUCT COS-350		VERSION V02	
COMPONENT TSD		VERSION VB02-00A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 12*	PAGE 1 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

SUPERSEDING EXISTING FILE Error

```
*WORD $DIO+11260=#4767
*WORD $DIO+11262=%$DIO+2402
*WORD $DIO+11264=#-73340
*WORD $DIO+11266=#207
*EXIT
ENTER CHECKSUM: 102671
```

STOP --

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DT0.OLD
ENTER OUTPUT FILE *DT0.OBJ
*POINT DT0
*BYTE DT0+4301=#102
*EXIT
ENTER CHECKSUM: 63245
```

STOP --

. ASS LP LST

. ASS LP LOG

. LOR BA LP

. R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

SOFTWARE PRODUCT COS-350		VERSION V02	
COMPONENT TSD		VERSION VB02-00A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 12*	PAGE 2 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

SUPERSEDING EXISTING FILE Error

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD.SAV
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

SOFTWARE PRODUCT COS-350	VERSION V02	
COMPONENT TSD	VERSION VB02-00A	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 12*	PAGE 3 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976

RANDOM RECORD READ

PROBLEM: Let two jobs open the same file in update mode. Let the first job randomly read some given record; i.e., record 56, then unlock the record and let the second job randomly read the same record, change its contents, and write it out thereby unlocking the record. The second job terminates, leaving the first job as the only user of the given file. The first job now rereads the same record (record 56) to see if it was changed. The read is not actually done, because the information is assumed to be still in core due to the first read. Thus, the first job thinks the record was not changed.

DISPOSITION: The following patch corrects this problem and changes the version number to TSD VB02-00C.

```
      CC
      .R PIP
      *DIO.OLD=DIO.OBJ/R
      *DTO.OLD=DTO.OBJ/R
      *CC

      .R PATCHO
      *OPEN
      ENTER INPUT FILE *DIO.OLD
      ENTER OUTPUT FILE *DIO.TMP
      *POINT #DIO
      *WORD #DIO+6012=#4767
      *WORD #DIO+6014=%#DIO+11270
      *WORD #DIO+11270=#12763
      *WORD #DIO+11272=#-1
      *WORD #DIO+11274=#156
      *WORD #DIO+11276=#5712
      *WORD #DIO+11300=#-77376
      *WORD #DIO+11302=#5063
      *WORD #DIO+11304=#156
      *WORD #DIO+11306=#12712
      *WORD #DIO+11310=#-1
      *WORD #DIO+11312=#207

      *EXIT
      ENTER CHECKSUM: 37527

      STOP --
```

RANDOM RECORD READ

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DIO.TMP
ENTER OUTPUT FILE *DIO.OBJ
P*QINT $DIO
*WORD $DIO+6150=#167
*WORD $DIO+6152=%$DIO+11314
*WORD $DIO+6154=#240
*WORD $DIO+6156=#240
*WORD $DIO+7432=#240
*WORD $DIO+11314=#-55016
*WORD $DIO+11316=#1
*WORD $DIO+11320=#13
*WORD $DIO+11322=#1005
*WORD $DIO+11324=#5763
*WORD $DIO+11326=#156
*WORD $DIO+11330=#-77376
*WORD $DIO+11332=#167
*WORD $DIO+11334=%$DIO+6160
*WORD $DIO+11336=#167
*WORD $DIO+11340=%$DIO+6266
*EXIT
ENTER CHECKSUM: 40763
STOP --
. R PATCHO
*OPEN
ENTER INPUT FILE *DIO.OLD
ENTER OUTPUT FILE *DIO.OBJ
*POINT DIO
*BYTE DIO+4301=#103
*EXIT
ENTER CHECKSUM: 63247
STOP --

. R PIP
*DIO.TMP/D
*^C
```

RANDOM RECORD READ

. ASS LP LST

. ASS LP LOG

. LOR BA LP

. R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USR IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

RANDOM RECORD READ

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD.SAV
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION VB02-00C

*

ONERROR TRAP

PROBLEM: ONERROR is used to trap a soft error in a routine which has been compiled with /O option. If the soft error occurs following an internal subroutine call, the subsequent RETURN statement will cause unpredictable results. This problem exists in both single and multiuser DIBOL programs.

DISPOSITION: The following patch corrects the problem and changes the version number to TSD VB02-00D.

```
.R PIP
*DDIRT.OLD=DDIRT.OBJ/R
*DTO.OLD=DTO.OBJ/R
*+C
```

```
.R PATCH0
*OPEN
ENTER INPUT FILE *DDIRT.OLD
ENTER OUTPUT FILE *DDIRT.OBJ
*POINT $DIRT
*WORD $DIRT+1016=#-726
*EXIT
ENTER CHECKSUM: 54234
```

STOP --

```
.R PATCH0
*OPEN
ENTER INPUT FILE *DTO.OLD
ENTER OUTPUT FILE *DTO.OBJ
*POINT DTO
*BYTE DTO+4301=#104
*EXIT
ENTER CHECKSUM: 63251
```

STOP --

```
.ASSIGN LP:LOG
```

```
.ASSIGN LP:LST
```

```
.LOAD LP.BA
```

```
R SYSGEN
```

ONERROR TRAP

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: <2>

DO YOU WANT REMOTE TERMINAL SUPPORT: <N>

ARE THEY ALL STANDARD TERMINALS: <Y>

NUMBER OF PROGRAMS TO RUN: <4>

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: <Y>

NUMBER OF MESSAGES IN MEMORY: <8>

NUMBER OF DEVICES ALLOWED: <4>

NUMBER OF CHANNELS OPEN PER PROGRAM: <10>

LOCK THE USER IN CORE: <N>

DO YOU WANT DDT: <N>

ARE YOU SATISFIED WITH ALL ANSWERS: <Y>

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION V02-000
*R RTEXT

SINGLE USER IN UPDATE MODE

PROBLEM: A single user has a given file opened in UPDATE mode. Some given record is read, modified, and then written out.

NOTE: Because the number of users of this file is exactly one, optimized code neither physically writes out the record nor unlocks the blocks.

Next, the user specifies an UNLOCK, which causes the blocks to be unlocked. If the user now reads some other record, the previously modified record never gets physically written out.

DISPOSITION: The following patch corrects this problem and changes the version number to TSD VB02-00E.

```
^C

.R PIP
*DIO.OLD=DIO.OBJ/R
*DTO.OLD=DTO.OBJ/R
*^C

.R PATCH0
*OPEN
ENTER INPUT FILE *DIO.OLD
ENTER OUTPUT FILE *DIO.OBJ
*POINT $DIO
*WORD $DIO+11340=X$DIO+6240
*EXIT
ENTER CHECKSUM: 61740

STOP --

.R PATCH0
*OPEN
ENTER INPUT FILE *DTO.OLD
ENTER OUTPUT FILE *DTO.OBJ
*POINT DTO
*BYTE DTO+4301=#105
*EXIT
ENTER CHECKSUM: 63253

STOP --

.R ASS LP LST
.R ASS LP LOG
```

SINGLE USER IN UPDATE MODE

.LOA BA LP

.R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING /LP:/ FOR LOG OR LST, BE SURE IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD.SAV
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

SUPERSEDING EXISTING FILE, EXCEEDING MAXIMUM NUMBER DEVICES, AND
ACCEPT STATEMENT VIA A FILE

PROBLEM:

1. Let the SUPERSEDING EXISTING FILE flag be set via an XCALL FLAGS. If one now tries to open the lineprinter, this will produce the SUPERSEDING EXISTING FILE error message. However, this message should only occur for file oriented devices.
2. When using an ACCEPT statement to retrieve characters from a file into a decimal field or alpha field, a CTRL/Z response to the MOUNT SUCCESSOR FOR INPUT message causes IMPOSSIBLE TRAP TO 4.
3. When a system is SYSGENed for some given number of allowable devices, the system will actually allow one less device than the number specified. Attempting to specify one more device will generate the message TOO MANY HANDLERS CALLED.

DISPOSITION: The following patch corrects these problems and changes the version number to TSD VB02-00F.

```
.R PIP
*DIO.OLD=DIO.OBJ/R
*DTO.OLD=DTO.OBJ/R
*DTOINI.OLD=DTOINI.OBJ/R
*^C

.R PATCHO
*OPEN
ENTER INPUT FILE *DIO.OLD
ENTER OUTPUT FILE *DIO.OB1
*POINT #DIO
*WORD #DIO+1534=#167
*WORD #DIO+1536=%#DIO+11342
*WORD #DIO+11342=#-72012
*WORD #DIO+11344=#2
*WORD #DIO+11346=#-77773
*WORD #DIO+11350=#5764
*WORD #DIO+11352=#20
*WORD #DIO+11354=#-77774
*WORD #DIO+11356=#167
*WORD #DIO+11360=%#DIO+1542
*WORD #DIO+11362=#167
*WORD #DIO+11364=%#DIO+1710
*WORD #DIO+11366=#167
*WORD #DIO+11370=%#DIO+1554
```

SUPERSEDING EXISTING FILE, EXCEEDING MAXIMUM NUMBER DEVICES, AND
ACCEPT STATEMENT VIA A FILE

```
*EXIT
ENTER CHECKSUM: 56055

STOP --

. R PATCHO
*OPEN
ENTER INPUT FILE *DIO.OB1
ENTER OUTPUT FILE *DIO.OB2
*POINT $DIO
*WORD $DIO+4470=#167
*WORD $DIO+4472=%$DIO+11372
*WORD $DIO+11372=#-17477
*WORD $DIO+11374=#10164
*WORD $DIO+11376=#2
*WORD $DIO+11400=#167
*WORD $DIO+11402=%$DIO+4474
*WORD $DIO+4616=#240
*WORD $DIO+4644=#167
*WORD $DIO+4646=%$DIO+11404
*WORD $DIO+11404=#16401
*WORD $DIO+11406=#2
*WORD $DIO+11410=#60301
*WORD $DIO+11412=#-65167
*WORD $DIO+11414=#167
*WORD $DIO+11416=%$DIO+4654
*EXIT
ENTER CHECKSUM: 11010

STOP --
```

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DIO.OB2
ENTER OUTPUT FILE *DIO.OBJ
*POINT $DIO
*WORD $DIO+4674=#167
*WORD $DIO+4676=%$DIO+11420
*WORD $DIO+11420=#10064
*WORD $DIO+11422=#5
*WORD $DIO+11424=#-17477
*WORD $DIO+11426=#10164
*WORD $DIO+11430=#2
```

SUPERSEDING EXISTING FILE, EXCEEDING MAXIMUM NUMBER DEVICE, AND
ACCEPT STATEMENT VIA A FILE

```
*WORD $DIO+11432=#167
*WORD $DIO+11434=%$DIO+4702
*WORD $DIO+5032=#167
*WORD $DIO+5034=%$DIO+11436
*WORD $DIO+11436=#16405
*WORD $DIO+11440=#2
*WORD $DIO+11442=#60305
*WORD $DIO+11444=#16402
*WORD $DIO+11446=#6
*WORD $DIO+11450=#167
*WORD $DIO+11452=%$DIO+5040
*EXIT
ENTER CHECKSUM: 167013
```

STOP --

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DIOINI.OLD
ENTER OUTPUT FILE *DIOINI.OBJ
*POINT DIOINI
*WORD DIOINI+526=#-61011
*EXIT
ENTER CHECKSUM: 41643
```

STOP --

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DIO.OLD
ENTER OUTPUT FILE *DIO.OBJ
*POINT DIO
*BYTE DIO+4301=#106
*EXIT
ENTER CHECKSUM: 63255
```

STOP --

```
. R PIP
*DIO.OB1.DIO.OB2/D
*CC
. ASS LP LOG
. ASS LP LST
. LOR BA LP
```

SUPERSEDING EXISTING FILE, EXCEEDING MAXIMUM NUMBER DEVICES, AND
ACCEPT STATEMENT VIA A FILE

.R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT
VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS
TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY
ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING
THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL.
SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES
AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD
SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION VB02-00F

*

BAD BINARY FILE ERROR MESSAGE ON GOOD FILES

PROBLEM: While running under TSD after entering an illegal command, the next legal run program command causes the error message BAD BINARY FILE to be displayed. The legal run program command can now be entered and will be accepted.

DISPOSITION: The following patch corrects this problem and changes the version number of TSD to VB02-00G.

```
R PIP
*DT0.OLD=DT0.OBJ/R
*^C

.R PATCH0

*OPEN
ENTER INPUT FILE *DT0.OLD
ENTER OUTPUT FILE *DT0.OBJ

*POINT DT0
*WORD DT0+612=#10306
*BYTE DT0+4301=#107
*EXIT
ENTER CHECKSUM: 113061
STOP --

.R ASS LP LOG

.R ASS LP LST

.R LOR BA LP

.R SYSGEN

TSD SYSGEN V02-00
```

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

BAD BINARY FILE ERROR MESSAGE ON GOOD FILES

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING /LP/ FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION VB02-000

*

OPTIMIZATION OF READS AND WRITES

PROBLEM: Due to the Version 2 optimization of reads and writes for a single user in update mode, LPTSPL must be modified so that control information in the LPQFIL.LPQ file is updated immediately and not buffered. If all files are printed followed by LPTSPL termination (via a CTRL/Z or shutting down TSD), an LPTSPL restart may try to print a file that has already been printed.

DISPOSITION: The following patch will correct this problem and change LPTSPL version number to VB02-02A.

NOTE: Before installing this patch allow all files currently queued to print. After installing the patch, delete the file LPQFIL.LPQ from the system device.

```
. R EDIT
*ERLPTSPL.DBL##
*EWLPTSPL.DBL##
*F,A150
$I RECORD      DUMMY,X
      ,A27
*-3A#3L##
      ,A150
RECORD  DUMMY,X
      ,A27
*FVB02-02#IA#0J#1L##
      DISPLAY (3,13,10,'TIME SHARED DIBOL LINE PRINTER SPOOLER VB02-02A',13,10
*F,OUTCNT)
$I      READ      (1,DUMMY,117)
*-2A#3L##
      WRITE      (1,PRINT,OUTCNT)
      READ      (1,DUMMY,117)
      GOTO      RECV
*2F,INCNT)
$I      READ      (1,DUMMY,117)
*-2A#3L##
      WRITE      (1,PRINT,INCNT)
      READ      (1,DUMMY,117)
      ONERROR OPERR
*FVARS,1)
```

OPTIMIZATION OF READS AND WRITES

```
#I      READ      (1,DUMMY,117)
#-2A#3L##
        WRITE     (1, VARS, 1)
        READ      (1,DUMMY,117)
        CALL      BUFFIL
*FCOPCTR
#I      INCR      COPSIZ
        IF        (COPSIZ.EQ.99) COPCTR=
#F(COP#3COPSIZ#4J#I120#2D#-4A#6L##
WRT.    WRITES    (1, VARS)
        INCR      COPCTR
        INCR      COPSIZ
        IF        (COPSIZ.EQ.99) COPCTR=
        IF        (COPSIZ.LT.120) GOTO    WRT
        CLOSE    (1)
*EX##

.R DICOMP
*LPTSPL=LPTSPL

      NO  ERRORS DETECTED

*CC

.R LINK
*LPTSPL.TSD=LPTSPL.TDIBOL

*CC
```

READING FILE OF LENGTH 0 RESULTS IN LINE PRINTER HANG (SY)

PROBLEM: If an attempt is made to read a file whose length is 0 blocks, the job will hang in IO WAIT. Likewise, if output to the line printer is interrupted due to the printer going off line, the job will also hang if it attempts to continue with the output once the printer is enabled again.

DISPOSITION: This patch corrects those problems and changes the version number to TSD VB02-00H.

```
. R PIP
*DIO. OLD=DIO. OBJ/R
*DTO. OLD=DTO. OBJ/R
*^C

. R PATCH0
*OPEN
ENTER INPUT FILE *DIO. OLD
ENTER OUTPUT FILE *DIO. OBJ
*POINT $DIO

*WORD $DIO+3336=#4767
*WORD $DIO+3340=%$DIO+11454
*WORD $DIO+3342=#240
*WORD $DIO+4426=#4767
*WORD $DIO+4430=%$DIO+11454
*WORD $DIO+11454=#-74772
*WORD $DIO+11456=#16702
*WORD $DIO+11460=%$JOB+0
*WORD $DIO+11462=#42762
*WORD $DIO+11464=#4000
*WORD $DIO+11466=JOBSTS+0
*WORD $DIO+11470=#-73352
*WORD $DIO+11472=#4767
*WORD $DIO+11474=%$SLEEP+0
*WORD $DIO+11476=#207
*EXIT
ENTER CHECKSUM: 107652

STOP --
```

READING FILE OF LENGTH 0 RESULTS IN LINE PRINTER HANG

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DT0.OLD
ENTER OUTPUT FILE *DT0.OBJ
*POINT DT0
*BYTE DT0+4301=#110
*EXIT
ENTER CHECKSUM: 63261
```

STOP --

. ASS LP LOG

. ASS LP LST

. LOR LP BA

. R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

READING FILE OF LENGTH 0 RESULTS IN LINE PRINTER HANG

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN VB02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION VB02-00H

*

LOCKED BLOCKS (SY)

PROBLEM: A single user has a given file opened in UPDATE mode and some given record is read, modified, and then written out.

NOTE: Because the number of users of this file is exactly one, optimized code does not unlock the blocks once the write is issued nor physically write out the record.

A second user now opens up the same file in UPDATE mode. Although the first user's record is immediately written out, the corresponding blocks were not unlocked. Hence, the second user would get a RECORD LOCK error if he attempted to read the given record.

DISPOSITION: This patch corrects this problem and changes the version number to TSD VB02-00I.

```
. R PIP
*DIO. OLD=DIO. OBJ/R
*DT0. OLD=DT0. OBJ/R
*CC

. R PATCHU
*OPEN
ENTER INPUT FILE *DIO. OLD
ENTER OUTPUT FILE *DIO. OBJ
*POINT $DIO
*WORD $DIO+1236=#4767
*WORD $DIO+1240=X*DIO+11500
*WORD $DIO+1242=#240
*WORD $DIO+11500=#16400
*WORD $DIO+11502=#4
*WORD $DIO+11504=#16000
*WORD $DIO+11506=#14
*WORD $DIO+11510=#5720
*WORD $DIO+11512=#4767
*WORD $DIO+11514=X*DIO+7426
*WORD $DIO+11516=#42762
*WORD $DIO+11520=#400
*WORD $DIO+11522=JOBSTS+0
*WORD $DIO+11524=#207
*EXIT
ENTER CHECKSUM: 176416

STOP --
```

LOCKED BLOCKS

```
. R PATCHO
*OPEN
ENTER INPUT FILE *DT0.OLD
ENTER OUTPUT FILE *DT0.OBJ
*POINT DT0
*BYTE DT0+4301=#111
*EXIT
ENTER CHECKSUM: 63263
```

STOP --

. ASS LP LOG

. ASS LP LST

. LOA LP BA

. R SYSGEN

TSD SYSGEN V02-00

EACH OF THE FOLLOWING QUESTIONS IS FOLLOWED BY A DEFAULT VALUE IN PARENTHESIS. THIS VALUE WILL BE USED IF A <CR> IS TYPED IN ANSWER TO THE QUESTION. IF A QUESTION MARK OR ANY ILLEGAL RESPONSE IS TYPED, FURTHER INFORMATION CONCERNING THE CURRENT QUESTION WILL BE PRINTED AT THE TERMINAL. SUITABLE DEFAULT VALUES FOR BUFFERS, DEVICES, MESSAGES AND OPEN CHANNELS HAVE BEEN CHOSEN FOR A STANDARD TSD SYSTEM. USE THESE UNLESS YOU HAVE SPECIAL NEEDS.

NUMBER OF TERMINALS: (2)

DO YOU WANT REMOTE TERMINAL SUPPORT: (N)

ARE THEY ALL STANDARD TERMINALS: (Y)

NUMBER OF PROGRAMS TO RUN: (4)

DO YOU WANT THE STANDARD SIZE TERMINAL BUFFER: (Y)

NUMBER OF MESSAGES IN MEMORY: (8)

NUMBER OF DEVICES ALLOWED: (4)

LOCKED BLOCKS

NUMBER OF CHANNELS OPEN PER PROGRAM: (10)

LOCK THE USER IN CORE: (N)

DO YOU WANT DDT: (N)

ARE YOU SATISFIED WITH ALL ANSWERS: (Y)

TSD SYSGEN V02-00 CONTINUING

IF YOU ARE USING 'LP:' FOR LOG OR LST, BE SURE
IT IS ON LINE AND READY.

ENTER THE NAME OF THE SAVE FILE ?TSD
AFTER THE MESSAGE 'END BATCH' IS DISPLAYED
ON YOUR TERMINAL REINITIALIZE YOUR SYSTEM

END BATCH

.R TSD

TSD VERSION VB02-00I

*

LINEPRINTER STOPS EXECUTION IF TURNED OFF (SY)

PROBLEM: If the line printer is put off line or turned off in the middle of printing a file, the line printer spooler will stop execution.

DISPOSITION: This patch will cause the LPTSPL program to loop until the line printer is ready and then print the proper information. This will correct the problem and change the LPTSPL version number to VB02-00B.

```
..R EDIT
*ERLPTSPL.DBL##
*EWLPTSPL.DBL##
*FVB02-02A*-1D#IB#0J#1L##
      DISPLAY (3,13,10,'TIME SHARED DIBOL LINE PRINTER SPOOLER VB02-02B',13,10)
*F
PRINT,$3A#FL$-1D#5D#IX##
*1A#IX.$1A#I      ONERROR BLANK
$-4A#5L##
      IF      (COPSIZ.EQ.0)  GOTO      BLANK
      ONERROR X
X.      DISPLAY (2,10(1,COPSIZ))
      ONERROR BLANK
BLANK.  DISPLAY (2,13,TERM)
*FOFFERROR$-7D#INERROR LOOP$-1A#3L##
BUFFIL, COPTM1=
      ONERROR LOOP
LOOP.   DISPLAY (2,0)
*EX##

.R DICOMP
*LPTSPL=LPTSPL

      NO  ERRORS DETECTED
*^C

.R LINK
*LPTSPL.TSD=LPTSPL.TDIADL

*^C
```


FOR Command Without an Argument

PROBLEM:

Execution of the FOR command without an argument (i.e., FOR I = 1,100) within a scheduled routine (via FQUE) may leave FOCAL in a state which will halt the processor if execution of the program is attempted in a second time (via GO).

DISPOSITION:

Until a patch is published for this problem, ensure that a semi-colon follows a FOR command without an argument.

SOFTWARE PRODUCT FOCAL/RT-11		VERSION V01	
COMPONENT N/A		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 1	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE October 1975	

OPERATE COMMAND CAUSES ERROR (SPR 11-8762 DA)

Consider the following Focal program:

```
1.10 ERASE
1.20 SET X=X+1
1.30 O T
1.40 TYPE X,!
1.50 O L
1.60 TYPE X,!
1.70 GOTO 1.20
```

The above program causes a great deal of device handler interaction at the monitor level (i.e., .FETCHes and .RELEASEs) between the TT: and the LP:. After approximately 50 loops of this program (assuming a 16K system), an error #23 (Symbol table shuffle error) will occur. This is due to the way Focal/RT-11 handles free core.

To solve the problem, load the specific device handlers before running Focal/RT-11 (i.e., use the .LOAD LP command at the monitor level for use of the OPERATE L command in Focal).

FCLK ROUTINE GIVES INCORRECT TIME (SPR 11-8900 DA)

Focal FCLK routine will give incorrect time for periods over nine minutes.

The following patches must be installed. Relink Focal.

```
.R PIP
*FOCAL1.OLD=FOCAL1.OBJ
*^C
```

```
.R PATCH0
*OPEN
ENTER INPUT FILE *FOCAL1.OBJ
ENTER OUTPUT FILE *FOCAL1.OBJ
*WORD 14000=#06167
*EXIT
ENTER CHECKSUM: 44776
```

```
STOP --
```

Paper Tape Users:

4K version: Change location 10656 Fr: 6367 to: 6167

8K version: Change location 11514 Fr: 6367 to: 6167

Clarification: Interfacing Assembly Language Routines to FORTRAN
(SPR 31-3461)

User-written assembly language routines which interface to the FORTRAN Object Time System must be aware of the location of the RT-11 USR (User Service Routines).

If a user routine requests a USR function (e.g., .LOOKUP, .ENTER), or if the USR is invoked by the FORTRAN Object Time System, the USR will be swapped into memory if it is nonresident. The FORTRAN object-time system is designed so that the USR may swap over it. User routines must be written to allow the USR to swap over them or must be located outside the region of memory into which the USR will swap.

User interrupt service routines and completion routines must be further restricted to be located where the USR will not swap, because of the asynchronous nature of these routines.

The USR (if in a swapping state) will always swap over the area of memory which starts at the program initial stack pointer address; the USR occupies 2K words. Interrupt and completion routines (and their data areas) must not be located in this area.

To remove these restrictions, the user must make the USR resident by either specifying the /U switch to the FORTRAN compiler or by issuing the RT-11 Version 2 console command SET USR NOSWAP before executing the program.

SOFTWARE PRODUCT FORTRAN/RT-11		VERSION V01C	
COMPONENT N/A		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 1	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE	

FLOATING MULTIPLY FAILS TO DETECT UNDERFLOW IN NHD VERSION OF OTS
(SPR 11-6790 GH)

PROBLEM: The NHD version fails to detect underflow when the result is very close to the minimum number representable (approximately $10E-38$).

DISPOSITION: This problem is different from the earlier floating multiply patch to V01B. The following patch corrects this problem.

This is patch #1 to the RT-11 FORTRAN IV object time system (OTS). The user should keep track as to which patches have been installed in his OTS; there is currently no patch level indicator.

In order to apply this patch, the user should verify that the module NHD.OBJ is on the system device. If this is not the case, then this module should be copied from the distribution media.

This patch corrects a problem in the no hardware version of the FORTRAN IV OTS. The floating multiply does not detect underflow for some cases which are near the lower boundry (approx $10E-38$).

In the following, the user types the underscored text; lines are terminated by a carriage return.

```
.R PATCHO
-----
*OPEN
-----
ENTER INPUT FILE*NHD.OBJ
-----
ENTER OUTPUT FILE*NHD.NEW
-----
*POINT FMULS
-----
*WORD 116=#4767
-----
*WORD 120=#200
-----
*WORD 122=#240
-----
*WORD 322=#-77772
-----
```

FLOATING MULTIPLY FAILS TO DETECT UNDERFLOW IN NHD VERSION OF OTS

```
*WORD 324=#32704
-----
*WORD 326=#77400
-----
*WORD 330=#1003
-----
*WORD 332=#62716
-----
*WORD 334=#-14
-----
*WORD 336=#207
-----
*WORD 340=#-72774
-----
*WORD 342=#62704
-----
*WORD 344=#77401
-----
*WORD 346=#207
-----
*EXIT
-----
ENTER CHECKSUM: 105326
-----
STOP --

.R PIP
-----
*NHD.OBJ/R=NHD.NEW
-----
*^C          (control C is typed here)
--
```

At this point the module NHD.OBJ should be copied onto the backup media for possible future use. If the libraries being used use the NHD hardware option, they should be rebuilt. That procedure is described in the RT-11 System Generation Manual (DEC-11-ORGMA-A-D) and is reproduced below.

FLOATING MULTIPLY FAILS TO DETECT UNDERFLOW IN NHD VERSION OF OTS

If the default FORTRAN library (FORLIB) was built for no special hardware (ie NHD), then the following procedure is used to re-build it.

- 1) Copy the files UNI.OBJ, OTSCOM.OBJ, V2NS.OBJ (V2S.OBJ for subscript checking) from the distribution media if not already on the system device.
- 2) Build the library as follows:

```
.R LIBR
-----
*FORLIB=UNI,OTSCOM,V2NS,NHD/G   (V2S may be substituted for V2NS)
-----
ENTRY POINTS:
$ERRS
-----
$ERRTB
-----
      (type a <CR> for the third line. It must be blank.)

$FIO   ILL INS
$END2  ILL INS
$END2  ILL INS
$CLO2  ILL INS
$FIO2  ILL INS
$FIO2  ILL INS

*^C   (type control C here)
--
```

If NHDLIB was built when FORTRAN was installed, it should be re-built using the above procedure substituting the name NHDLIB for the name FORLIB in the first line to LIBR.

COMPILING MULTIPLE PROGRAM UNITS FROM A SINGLE CASSETTE FILE (SPR 11-5483 RB)

PROBLEM: FORTRAN will not recognize multiple programs in a single input file if that file is on a cassette. Only the first program unit (FUNCTION, SUBROUTINE, or main program) in the file is processed.

ANALYSIS: This problem is caused by the fact that multiple program units cannot be processed from the same input file unless that file resides on a device which allows random access input. This is true of the RT-11 magtape handler, but not the cassette handler.

DISPOSITION: At the present time, the following restriction applies to source files on cassettes: A cassette file which contains multiple program units must be transferred to a random access device prior to compilation; if compilation from the cassette is required, then each source program unit should be placed in a separate file.

This problem is being examined for possible future solution.

STAND-ALONE FORTRAN STACK USAGE (SPR 11-8110 RB)

PROBLEM: If the stack pointer (SP) has an odd value or is zero when a stand alone FORTRAN program is started, the program will halt.

ANALYSIS: This situation is uncommon, as the SP would almost never become zero under normal circumstances. However, the FORTRAN main program executes a JSR instruction as its first operation; hence at least one word of valid stack storage is required to start a stand alone routine.

NOTE: Once the program has started, the stack is reset by the FORTRAN OTS; hence a minimal amount of space is required for startup.

DISPOSITION: This problem is being examined. Meantime, the following restriction applies to stand alone programs:

When starting a stand alone program, the SP must not be odd or zero.

WRITING ON READ-ONLY FILE (SPR 11-6514 GH)

PROBLEM: When a CALL ASSIGN is done to make a file read-only, a subsequent write (or an ENDFILE statement) to that file will issue the error message and then perform the write, thereby destroying the contents of the file beyond that point.

DISPOSITION: This is due to a design error in the error processor of the OTS. Due to the complexity of the problem, it is not patchable. It will, however, be corrected in the next release of FORTRAN (V02) to be available in February or March 1977.

WRITING BEYOND END OF RANDOM ACCESS FILE (SPR 11-8129 GH)

PROBLEM: If a record is written beyond the end of a Random Access file using the END = option, control will be transferred to the specified label but subsequent I/O to that file will attempt to write out that buffer (i.e., the one beyond the end of the file) and will cause ERROR #24. Closing the file either explicitly (with CALL CLOSE) or implicitly (as is done when the program exits) will also result in error #24 and possibly will cause the system to crash.

ANALYSIS: When control is transferred by the END = option, the buffer is not invalidated; therefore, subsequent I/O will attempt to write it out again so that the buffer might be reused. Also, when the file is closed, the last buffer is written out which, if it is beyond the end of the file, will cause ERROR #24. If the error is detected when the file is closed, the program may crash the system due to incorrect error recovery.

DISPOSITION: This problem is currently under investigation by the FORTRAN development group. Until such time as a more permanent solution is found, the following restriction applies:

When writing records to a Random Access file (of record size greater than 255 words), do not write beyond the end of file; the end of file may be detected, however, with the END = option on a Random Access Read statement.

ASYNCHRONOUS I/O, EVENT DRIVEN I/O, AND FORTRAN PROGRAMS
(SPR 11-8152 GH)

PROBLEM: When FORTRAN programs open a file, a wait is done on every channel before the file is opened. If event driven I/O (or asynchronous I/O) is outstanding, the system will hang until the I/O is complete thereby violating the expectations of said I/O.

DISPOSITION: When a FORTRAN program opens a file, it locks the USR into core for a brief period of time so the .LOOKUPs, .ENTERs, and .FETCHs may be done with a minimal amount of swapping. The USR is locked into the lower part of the FORTRAN program image. If a FORTRAN subroutine were the interrupt handler for any event driven I/O, it clearly would not be good for an interrupt to occur while the routine is swapped out and the USR is loaded in its place. Hence, a wait is done on each channel until it is safe to lock the USR into core.

If event driven I/O is being used, then all files should be opened before said I/O is initiated. This may be accomplished by writing a null record on the lun; e.g.,

```
100          WRITE(5,100)  
            FORMAT('+',$)
```


OBJECT TIME FORMATTING WITH H FORMAT SPECIFICATION, FORMATTING RECORD WRITING GREATER THAN 132 CHARACTERS IN LENGTH, MAY FAIL (SPR 11-8103 GH)

PROBLEM:

1. Object time format using the H format specification may cause a trap to 4 or other errors.
2. Writing formatted records greater in size than the internal buffer (132 unless specified otherwise with compiler /R switch) may cause program to halt.

DISPOSITION: The following patch corrects these problems. This is OTS patch 2. The user should keep track as to which patches have been installed in his OTS; there is currently no patch level indicator.

In order to apply this patch, the user should verify that the module OTSCOM.OBJ is on the system device. If this is not the case, this module should be copied from the distribution media.

The user types the underscored text; lines are terminated by a carriage return.

```
. R PATCHO
-----
*OPEN
-----
ENTER INPUT FILE*OTSCOM.OBJ
-----
ENTER OUTPUT FILE*OTSCOM.NEW
-----
*POINT OBJFMT
-----
*WORD 1134=#-65300
-----
*WORD 1136=#4767
-----
*WORD 1140=#-212
-----
*WORD 1142=#5302
-----
*WORD 1144=#-77405
-----
*POINT FORMAT
-----
```

OBJECT TIME FORMATTING WITH H FORMAT SPECIFICATION, FORMATTING RECORD
WRITING GREATER THAN 132 CHARACTERS IN LENGTH MAY FAIL (SPR 11-8103 GH)

```
*WORD 1510=#3375
-----
*EXIT
-----
ENTER CHECKSUM: 61312
-----

STOP --

.R PIP
-----
*OTSCOM.OBJ/R=OTSCOM.NEW
-----
*^C          (control C is typed here)
--
```

At this point the module OTSCOM.OBJ should be copied onto the backup media for possible future use. The FORTRAN libraries should be re-built. That procedure is described in the RT-11 system generation manual and is reproduced below.

In the following ??? represents the hardware module appropriate for your configuration. e.g. NHD, EAE, EIS, FIS, or FPU.

1) Copy the files UNI.OBJ,???.OBJ,V2NS.OBJ (V2S.OBJ for subscript checking) from the distribution media if not already on the system device.

2) Build the library as follows:

```
.R LIBR
-----
*FORLIB=UNI,OTSCOM,V2NS,???/G  (V2S may be substituted for V2NS)
-----
ENTRY POINTS:
$ERRS
-----
$ERRTB
-----
(type a <CR> for the third line. It must be blank.)
```

OBJECT TIME FORMATTING WITH H FORMAT SPECIFICATION, FORMATTING RECORD
WRITING GREATER THAN 132 CHARACTERS IN LENGTH MAY FAIL (SPR 11-8103 GH)

```
$FIO2  ILL INS  
$END2  ILL INS  
$END2  ILL INS  
$CLO2  ILL INS  
$FIO2  ILL INS  
$FIO2  ILL INS
```

```
*^C    (type control C here)
```

```
--
```

OBJECT TIME ENCODE/DECODE

PROBLEM: Object time ENCODE/DECODE does not restore the free space pointer properly. This may result in trap to 4, ?M-ILL ADDR, or other errors.

DISPOSITION: The following patch corrects this problem. This is OTS patch 3. The user should keep track as to which patches have been installed in his OTS; there is currently no patch level indicator.

In order to apply this patch, the user should verify that the module OTSCOM.OBJ is on the system device. If this is not the case, this module should be copied from the backup media where it was stored in the last patch.

The user types the underscored text; lines are terminated by a carriage return.

```
.R PATCHO
-----
*OPEN
-----
ENTER INPUT FILE*OTSCOM.OBJ
-----
ENTER OUTPUT FILE*OTSCOM.NEW
-----
*POINT OBJENC
-----
*WORD 16=#16346
-----
*WORD 20=#14
-----
*WORD 22=#4767
-----
*WORD 24=%$OBJFM+0
-----
*WORD 26=#12663
-----
*WORD 30=#14
-----
*WORD 32=#16366
-----
*WORD 34=#22
-----
```

OBJECT TIME ENCODE/DECODE

```
*WORD 36=#4
-----
*WORD 40=#207
-----
*EXIT
-----
ENTER CHECKSUM: 134341
-----
STOP --

.R PIP
-----
*OTSCOM.OBJ/R=OTSCOM.NEW
-----
*^C          (control C is typed here)
--
```

At this point the module OTSCOM.OBJ should be copied onto the backup media for possible future use. The FORTRAN libraries should be re-built. That procedure is described in the RT-11 system generation manual and is reproduced below.

In the following ??? represents the hardware module appropriate for your configuration. e.g. NHD, EAE, EIS, FIS, or FPU.

- 1) Copy the files UNI.OBJ, ??? .OBJ, V2NS.OBJ (V2S.OBJ for subscript checking) from the distribution media if not already on the system device.
- 2) Build the library as follows:

```
.R LIBR
-----
*FORLIB=UNI,OTSCOM,V2NS,???/G  (V2S may be substituted for V2NS)
-----
ENTRY POINTS:
$ERRS
-----
$ERRTB
-----
      (type a <CR> for the third line. It must be blank.)
```

OBJECT TIME ENCODE/DECODE

\$END2 ILL INS
\$END2 ILL INS
\$FIO2 ILL INS
\$CLO2 ILL INS
\$FIO2 ILL INS
\$FIO2 ILL INS

*^C (type control C here)

--

CLARIFICATION OF I/O LIST ELEMENTS (SPR 11-8279 GH)

PROBLEM: The description of simple I/O lists in the FORTRAN Language Reference Manual, DEC-11-LFLRA-B-D, Section 5.2.1, does not include the construct of simple I/O list elements separated by commas and grouped by parentheses.

DISPOSITION: The manual will be changed to read:

A simple I/O list element consists of a single variable, array reference, array element, constant, or expression. A simple I/O list consists either of a simple I/O list element or a group of two or more simple I/O list elements separated by commas and enclosed by parentheses.

CALL ASSIGN WITH FILE NAME TERMINATED WITH SPACES ABORTS (GH)

CALL ASSIGN WITH A FILE NAME TERMINATED WITH SPACES ABORTS WITH
?ERR 36 BAD FILE SPECIFICATION STRING.

THE FOLLOWING PATCH CORRECTS THE PROBLEM.

THIS IS PATCH #4 TO THE RT-11 FORTRAN IV V01C-03 OBJECT TIME SYSTEM (OTS).
THE USER SHOULD KEEP TRACK OF WHICH PATCHES HAVE BEEN INSTALLED IN HIS OTS;
THERE CURRENTLY IS NO PATCH LEVEL INDICATOR.

1. COPY THE FILE OTSCOM.OBJ FROM THE BACKUP MEDIA (WHERE IT WAS STORED IN
THE PREVIOUS PATCH) IF NOT ALREADY ON THE SYSTEM DEVICE.
2. IN THE FOLLOWING THE USER TYPES THE UNDERScoreD TEXT; LINES ARE
TERMINATED WITH A CARRIAGE RETURN.

```
.R PATCHO
-----
*OPEN
----
ENTER INPUT FILE*OTSCOM.OBJ
-----
ENTER OUTPUT FILE*OTSCOM.NEW
-----
*POINT ASSIGN
-----
*WORD 64=#-56751
-----
*WORD 66=#40
-----
*WORD 70=#1404
-----
*WORD 72=#-65756
-----
*EXIT
----
ENTER CHECKSUM: 61265
-----
STOP --
```

```
.R PIP
-----
*OTSCOM.OBJ/R=OTSCOM.NEW
-----
*^C
```

3. AT THIS POINT THE MODULE OTSCOM.OBJ SHOULD BE COPIED ONTO SOME BACKUP MEDIA FOR POSSIBLE FUTURE USE.
4. THE FORTRAN LIBRARY SHOULD NOW BE RE-BUILT. THAT PROCEDURE IS DESCRIBED IN THE RT-11 SYSTEM GENERATION MANUAL AND IS REPRODUCED BELOW.

IN THE FOLLOWING ??? REPRESENTS THE HARDWARE MODULE APPROPRIATE FOR YOUR CONFIGURATION. E.G. NHD, EAE, EIS, FIS, OR FPU.

- 4A. COPY THE FILES UNI.OBJ AND V2NS.OBJ (V2S.OBJ FOR SUBSCRIPT CHECKING) FROM THE DISTRIBUTION MEDIA IF NOT ALREADY ON THE SYSTEM DEVICE.
- 4B. IF THE HARDWARE MODULE APPROPRIATE FOR YOUR CONFIGURATION IS NHD.OBJ THEN COPY IT FROM THE BACKUP MEDIA (WHERE IT WAS STORED FROM THE FIRST OTS PATCH) IF NOT ALREADY ON THE SYSTEM DEVICE. OTHERWISE COPY THE MODULE ???OBJ FROM THE DISTRIBUTION MEDIA IF NOT ALREADY ON THE SYSTEM DEVICE.

```
.R LIBR
-----
*FORLIB=UNI,OTSCOM,V2NS,???/G (V2S MAY BE SUBSTITUTED FOR V2NS)
-----
```

ENTRY POINTS:

```
$ERRS
-----
$ERRTB
-----
```

(TYPE A <CR> FOR THE THIRD LINE. IT MUST BE BLANK.)

```
$END2 ILL INS
$END2 ILL INS
$F102 ILL INS
$CLO2 ILL INS
$F102 ILL INS
$F102 ILL INS
*^C
```

I-FORMAT CONVERSION ERROR (GH)

IF A NEGATIVE VALUE IS TRANSFERED FOR OUTPUT UNDER THE CONTROL OF AN I FORMAT SPECIFIER, AND THE NUMBER OF DIGITS IN THE MAGNITUDE OF THE RESULT IS EQUAL TO THE FIELD WIDTH SPECIFIED, THE FIELD IS NOT ASTRISK FILLED; THE NUMBER IS PRINTED WITHOUT THE LEADING MINUS SIGN.

THE FOLLOWING PATCH CORRECTS THE PROBLEM.

THIS IS PATCH #5 TO THE RT-11 FORTRAN IV V01C-03 OBJECT TIME SYSTEM (OTS). THE USER SHOULD KEEP TRACK OF WHICH PATCHES HAVE BEEN INSTALLED IN HIS OTS THERE CURRENTLY IS NO PATCH LEVEL INDICATOR.

1. COPY THE FILE OTSCOM.OBJ FROM THE BACKUP MEDIA (WHERE IT WAS STORED IN THE PREVIOUS PATCH) IF NOT ALREADY ON THE SYSTEM DEVICE.
2. IN THE FOLLOWING THE USER TYPES THE UNDERScoreD TEXT; LINES ARE TERMINATED WITH A CARRIAGE RETURN.

```
.R PATCHO
-----
*OPEN
-----
ENTER INPUT FILE*OTSCOM.OBJ
-----
ENTER OUTPUT FILE*OTSCOM.NEW
-----
*POINT CONVI
-----
*WORD 352=#4767
-----
*WORD 354=#26
-----
*WORD 404=#-72777
-----
*WORD 406=#20127
-----
*WORD 410=#26400
-----
*WORD 412=#207
-----
*EXIT
-----
```

ENTER CHECKSUM: 167777

STOP --

.R PIP

*OTSCOM.OBJ/R=OTSCOM.NEW

*^C

3. AT THIS POINT THE MODULE OTSCOM.OBJ SHOULD BE COPIED ONTO SOME BACKUP MEDIA FOR POSSIBLE FUTURE USE.
4. THE FORTRAN LIBRARY SHOULD NOW BE RE-BUILT. THAT PROCEDURE IS DESCRIBED IN THE RT-11 SYSTEM GENERATION MANUAL AND IS REPRODUCED BELOW.

IN THE FOLLOWING ??? REPRESENTS THE HARDWARE MODULE APPROPRIATE FOR YOUR CONFIGURATION. E.G. NHD, EAE, EIS, FIS, OR FPU.

- 4A. COPY THE FILES UNI.OBJ AND V2NS.OBJ (V2S.OBJ FOR SUBSCRIPT CHECKING) FROM THE DISTRIBUTION MEDIA IF NOT ALREADY ON THE SYSTEM DEVICE.
- 4B. IF THE HARDWARE MODULE APPROPRIATE FOR YOUR CONFIGURATION IS NHD.OBJ THEN COPY IT FROM THE BACKUP MEDIA (WHERE IT WAS STORED FROM THE FIRST OTS PATCH) IF NOT ALREADY ON THE SYSTEM DEVICE. OTHERWISE COPY THE MODULE ???OBJ FROM THE DISTRIBUTION MEDIA IF NOT ALREADY ON THE SYSTEM DEVICE.

.R LIBR

*FORLIB=UNI,OTSCOM,V2NS,???/G (V2S MAY BE SUBSTITUTED FOR V2NS)

ENTRY POINTS:

\$ERRS

\$ERRTB

(TYPE A <CR> FOR THE THIRD LINE. IT MUST BE BLANK.)

\$END2 ILL INS

\$END2 ILL INS

\$FIO2 ILL INS

\$CLO2 ILL INS

\$FIO2 ILL INS

\$FIO2 ILL INS

*^C

J=J-J GIVES INCORRECT RESULTS (GH)

1. STATEMENTS OF THE FORM

J=J-J

ARE INCORRECTLY OPTIMIZED TO CODE EQUIVALENT TO

J=-2*J

2. IMPROPER CODE GENERATION FOR CERTAIN SUBTRACT AND DIVIDE OPERATIONS IN THE NEIGHBORHOOD OF A CONVERSION OPERATION WHOSE RESULT IS A COMMON SUB-EXPRESSION.
3. INTERACTIONS BETWEEN STATEMENTS WHICH GENERATE TEMPORARY VARIABLES AND ERRORS IN THESE STATEMENTS MAY CAUSE THE COMPILER TO LOOP.
4. UPDATING VARIABLES IN DIFFERENT COMMON BLOCKS WITH THE SAME OFFSET YIELDS INCORRECT RESULTS.

THE FOLLOWING PATCH CORRECTS THESE PROBLEMS.

THIS IS PATCH #2 TO THE RT-11 FORTRAN IV COMPILER.

1. VERIFY THAT THE CURRENT VERSION OF THE COMPILER IS V01C-03A. THIS MAY BE DONE AS FOLLOWS:

```
.R FORTRAN
*,TT:=TT:
*^C          (TYPE CTRL C HERE)
```

```
FORTRAN IV      V01C-03A
```

IF THE CURRENT VERSION IS NOT V01C-03A, THEN THE PREVIOUS PATCH MAY HAVE BEEN OMITTED.

2. ENTER THE FOLLOWING FILE:

FILE NAME: TEST.FOR

```

COMMON /A1/ I1
COMMON /A2/ I2
I1=I2+1
D A=IFIX(1000**A)
J=J-J
B=1/(A+1)
END

```

3. CREATE A COPY OF THE COMPILER. THIS IS THE COPY TO BE PATCHED.

```

.R PIP
*OLDFOR.SAV=FORTRA.SAV
*^C

```

4. IN THE FOLLOWING THE USER TYPES THE UNDERScoreD TEXT; <CR> DENOTES THE CARRIAGE RETURN KEY; <LF> DENOTES THE LINE FEED KEY. THE RESULTING VERSION OF FORTRAN WILL BE FORTRAN IV V01C-03B.

```

.R PATCH
-----
PATCH V01-02

FILE NAME--
*OLDFOR/O<CR>
-----
*2210;1R
-----
*2260;2R
-----
*2344;3R
-----
*3046;4R
-----
*10:3,546/      20227      4737<LF>
-----
10:4,46/        12          2224<CR>
-----
*10:1,14/       0           20227<LF>
-----
10:1,16/        0           30<LF>
-----
10:1,20/        0           3003<LF>
-----
10:1,22/        0           20227<LF>
-----
10:1,24/        0           12<LF>
-----
10:1,26/        0           207<LF>
-----
10:1,30/        0           104403<CR>
-----
*11:3,1020/     402          4737<LF>
-----
11:4,320/       5720         2210<LF>
-----

```

11:4,322/	5020	402<CR>

*11:1,0/	0	407<LF>

11:1,2/ 0	12711	<LF>

11:1,4/ 0	2012	<LF>

11:1,6/ 0	5041	<LF>

11:1,10/	0	5041<LF>

11:1,12/	0	5041<LF>

11:1,14/	0	5041<LF>

11:1,16/	0	5741<LF>

11:1,20/	0	20137<LF>

11:1,22/	0	2152<LF>

11:1,24/	0	101366<LF>

11:1,26/	0	207<CR>

*14:2,540/	16700	4737<LF>

14:3,456/	177130	2210<CR>

*14:2,1576/	42702	4737<LF>

14:4,1012/	177600	2234<CR>

*14:1,0/	0	13700<LF>

14:1,2/ 0	2154	<LF>

14:1,4/ 0	11501	<LF>

14:1,6/ 0	42701	<LF>

14:1,10/	0	177600<LF>

14:1,12/	0	120027<LF>

14:1,14/	0	125<LF>

14:1,16/	0	103401<LF>

14:1,20/	0	5026<LF>

14:1,22/	0	207<LF>

14:1,24/	0	42702<LF>

14:1,26/	0	177600<LF>
14:1,30/	0	20227<LF>
14:1,32/	0	103<LF>
14:1,34/	0	1003<LF>
14:1,36/	0	22121<LF>
14:1,40/	0	162716<LF>
14:1,42/	0	16<LF>
14:1,44/	0	207<CR>
*16:4,1632/	12721	4737<LF>
16:4,1634/	35405	2210<CR>
*16:1,0/	0	12700<LF>
16:1,2/ 0	2554<LF>	
16:1,4/ 0	5720<LF>	
16:1,6/ 0	100015<LF>	
16:1,10/	0	22037<LF>
16:1,12/	0	2564<LF>
16:1,14/	0	1012<LF>
16:1,16/	0	22037<LF>
16:1,20/	0	2566<LF>
16:1,22/	0	1007<LF>
16:1,24/	0	5010<LF>
16:1,26/	0	62737<LF>
16:1,30/	0	177770<LF>
16:1,32/	0	2544<LF>
16:1,34/	0	5726<LF>
16:1,36/	0	137<LF>
16:1,40/	0	5716<LF>
16:1,42/	0	12721<LF>


```

16:1,44/      0      35405<LF>
-----
16:1,46/      0      207<CR>
-----
*3324;5R
-----
*3366;6R
-----
*20:5,660/    21014  1011<LF>
-----
20:6,620/     1010   21014<LF>
-----
20:6,622/     14000  404<CR>
-----
*21:6,604/    21014  1011<LF>
-----
21:6,606/     1010   21014<LF>
-----
21:6,610/     14000  404<CR>
-----
*22:5,564/    21014  1011<LF>
-----
22:6,524/     1010   21014<LF>
-----
22:6,526/     14000  404<CR>
-----
*1:2,2506\    101    102<CR>
-----

```

```

*E
-

```

- TEST THE PATCH BY COMPILING THE TEST PROGRAM (TEST.FOR) AND OBTAINING THE GENERATED CODE LISTING BOTH WITH AND WITHOUT THE DEBUG OPTION (COMPILER /D OPTION). PRINT THE LISTINGS WHEN THEY ARE COMPLETE.

```

E.G.   .R OLDFOR
        *,TESTD=TEST/L:ALL/D
        [ .MAIN.] ERRORS: 1, WARNINGS: 0
        *,TEST=TEST/L:ALL
        **C           (CTRL C IS TYPED HERE)

```

- IF THE COMPILER ABORTS, LOOPS, OR THE GENERATED CODE IS NOT THE SAME AS IN THE LISTING BELOW, REAPPLY THE PATCH STARTING AT STEP 3 (ABOVE).
- ONCE THE COMPILER HAS BEEN SUCCESSFULLY PATCHED, REPLACE THE ORIGINAL WITH THE PATCHED VERSION AS OUTLINED BELOW:

```

E.G.   .R PIP
        *FORTRA.SAV/R=OLDFOR.SAV
        **C

```

- NOW THAT THE PATCH IS INSTALLED, THE PATCHED VERSION OF FORTRAN SHOULD BE COPIED ONTO SOME BACKUP MEDIA FOR POSSIBLE FUTURE PATCHES.

E.G. .R PIP
*XX:FORTRA.SAV=FORTRA.SAV

XX IS THE BACKUP DEVICE.

LISTINGS
=====

FILE: TEST.LST

```
0001      COMMON /A1/ I1
0002      COMMON /A2/ I2
0003      I1=I2+1
          D   A=IFIX(1000**A)
0004      J=J-J
0005      B=1/(A+1)
0006      END
```

FORTRAN IV STORAGE MAP

NAME	OFFSET	ATTRIBUTES
J	000006	INTEGER*2 VARIABLE
B	000010	REAL*4 VARIABLE
A	000014	REAL*4 VARIABLE

COMMON BLOCK /A1/ LENGTH 000002

I1 000000 INTEGER*2 VARIABLE

COMMON BLOCK /A2/ LENGTH 000002

I2 000000 INTEGER*2 VARIABLE

FORTRAN IV GENERATED CODE

```
ISN #0003
000020 LSNS      #000003
000024 MOISMM   A2+#000000      A1+#000000
000032 ICISM    A1+#000000
```

```
ISN #0004
000036 ISNS
000040 MOISOM   000006
```

```
ISN #0005
000044 ISNS
000046 MOISIS   #000001
000052 CFIS
000054 MOISIS   #000001
000060 CFIS
```

000062 ADF\$MS 000014
000066 DIF\$SS
000070 MOF\$SM 000010

ISN #0006
000074 ISN\$
000076 RET\$

FILE: TESTD.LST

0001 COMMON /A1/ I1
0002 COMMON /A2/ I2
0003 I1=I2+1
0004 D A=IFIX(1000**A)
0005 J=J-J
0006 B=1/(A+1)
0007 END

FORTRAN IV DIAGNOSTICS

IN LINE 0004 MSG #090 ILLEGAL TYPE FOR OPERATOR

FORTRAN IV STORAGE MAP

NAME	OFFSET	ATTRIBUTES
A	000006	REAL*4 VARIABLE
IFIX	000000	INTEGER*2 PROCEDURE
J	000012	INTEGER*2 VARIABLE
B	000014	REAL*4 VARIABLE

COMMON BLOCK /A1/ LENGTH 000002

I1 000000 INTEGER*2 VARIABLE

COMMON BLOCK /A2/ LENGTH 000002

I2 000000 INTEGER*2 VARIABLE

FORTRAN IV GENERATED CODE

ISN #0003
000022 LSN\$ #000003
000026 MOISM A2+#000000 A1+#000000
000034 ICISM A1+#000000

ISN #0004
000040 ISN\$
000042 FUD\$

ISN #0005

000044 ISN\$

000046 MOI\$OM 000012

ISN #0006

000052 ISN\$

000054 MOI\$IS #000001

000060 CFIS

000062 MOI\$IS #000001

000066 CFIS

000070 ADF\$MS 000006

000074 DIF\$SS

000076 MOF\$SM 000014

ISN #0007

000102 ISN\$

000104 RET\$

LISTING FILES DIRECTED TO MAGTAPE (GH)

WHEN GENERATING PROGRAM LISTINGS DIRECTLY TO MAGTAPE,
THE FORTRAN COMPILER MAY WRITE EXTRANEEOUS DATA BETWEEN
SECTIONS OF THE LISTING.

THE FOLLOWING PATCH IS OPTIONAL AND SHOULD BE APPLIED AT
THE SYSTEM MANAGERS DISCRETION.

1. VERIFY THAT THE CURRENT VERSION OF THE COMPILER IS V01C-03B. THIS
MAY BE DONE AS FOLLOWS:

```
.R FORTRAN
*,TT:=TT:
^C          (TYPE CTRL C HERF)
```

```
FORTRAN IV V01C-03B SAT 27-MAR-76 19:15:36 PAGE 001
```

IF THE CURRENT VERSION IS NOT V01C-03B, THEN PREVIOUS PATCHES
MAY HAVE BEEN OMITTED.

2. CREATE A COPY OF THE COMPILER. THIS IS THE COPY TO BE PATCHED.

```
.R PIP
*OLDFOR.SAV=FORTRA.SAV
*^C
```

3. IN THE FOLLOWING THE USER TYPES THE UNDERScoreD TEXT; <CR> DENOTES
THE CARRIAGE RETURN KEY; <LF> DENOTES THE LINE FEED KEY. THE
RESULTING VERSION OF FORTRAN WILL BE FORTRAN IV V01C-03B+.

```
.R PATCH
-----
PATCH V01-02

FILE NAME--
*OLDFOR/O<CR>
-----
*2210;1R
-----
*2260;2R
-----
*42/      1000      746<CR>
-----
```

```

*2134/ 1000 746<CR>
-----
*746/ 0 10146<LF>
-----
750/ 0 60001<LF>
-----
752/ 0 60001<LF>
-----
754/ 0 162700<LF>
-----
756/ 0 400<LF>
-----
760/ 0 1403<LF>
-----
762/ 0 5021<LF>
-----
764/ 0 5200<LF>
-----
766/ 0 100775<LF>
-----
770/ 0 12601<LF>
-----
772/ 0 13700<LF>
-----
774/ 0 2156<LF>
-----
776/ 0 207<CR>
-----
*1:2,1060/ 16700 4737<LF>
-----
1:2,1062/ 176604 2210<CR>
-----
*1:1,0/ 0 12704<LF>
-----
1:1,2/ 0 6552<LF>
-----
1:1,4/ 0 5024<LF>
-----
1:1,6/ 0 20427<LF>
-----
1:1,10/ 0 6734<LF>
-----
1:1,12/ 0 101774<LF>
-----
1:1,14/ 0 13700<LF>
-----
1:1,16/ 0 2150<LF>
-----
1:1,20/ 0 207<CR>
-----
*2:2,4222/ 16700 4737<LF>
-----
2:2,4224/ 173450 746<CR>
-----
*15:2,1356/ 16700 4737<LF>
-----

```

```

15:2,1360/      176314  746<CR>
                -----
*17:2,542/      16700   4737<LF>
-----
17:2,544/      177130  746<CR>
                -----
*4664;3R
-----
*22:3,1022/     16700   4737<LF>
-----
22:3,1024/     174244  746<CR>
                -----
*1:2,2507\     40      53<CR>
-----
*E
-
```

4. TEST THE PATCH BY COMPILING ANY FORTRAN PROGRAM, DIRECTING THE LISTING TO A MAGTAPE FILE AS FOLLOWS:

```

.R OLDFOR
*,MT:TEST=TEST/A/L:ALL/O
*^C          (CTRL C IS TYPED HERE)
```

5. IF THE COMPILER ABORTS, LOOPS, OR THE GENERATED LISTING STILL CONTAINS INTERSPERSED EXTRANEIOUS DATA, REAPPLY THE PATCH STARTING AT STEP 2 (ABOVE).
6. ONCE THE COMPILER HAS BEEN SUCCESSFULLY PATCHED, REPLACE THE ORIGINAL WITH THE PATCHED VERSION AS OUTLINED BELOW:

```

E.G.          .R PIP
              *FORTRA.SAV/R=OLDFOR.SAV
              *^C
```

7. NOW THAT THE PATCH IS INSTALLED, THE PATCHED VERSION OF FORTRAN SHOULD BE COPIED ONTO SOME BACKUP MEDIUM FOR POSSIBLE FUTURE PATCHES.

```

E.G.          .R PIP
              *XX:FORTRA.SAV=FORTRA.SAV          XX IS THE BACKUP DEVICE.
              *^C
```

CALL CLOSE ON INACTIVE UNIT (SPR 11-8757 RB)

The documentation of the CALL CLOSE routine in the RT-11/RSTS/E FORTRAN IV User's Guide is inadequate. The following paragraphs explain the actions taken on explicit and implicit CLOSE operations on units which are not open.

The operation of the CALL CLOSE routine differs from that of the implied close operation performed on program termination, under certain conditions.

If a CALL CLOSE is attempted on a logical unit which is not currently open, the specified unit is opened for output and subsequently closed. This will cause any pre-existing file with the same name to be deleted.

If a unit is not open on program termination, no action is taken, and any pre-existing files with same name as may have been associated with the unopened unit (through the CALL ASSIGN routine) are retained.

This allows applications programs which may create several files on each execution to have the previous file (associated with the last execution of the program) deleted if no records are written to the unit in question during the current invocation of the program.

ARITHMETIC STATEMENT FUNCTIONS WITH NO ARGUMENTS (SPR 11-8865 GH)

Compiler terminates prematurely with trap to 4 when a program unit contains an ASF with no arguments; e.g., $AFUN()=A*B+C$

This problem is due to a coding error in the compiler parser. This problem is not patchable but will be considered for the next release of FORTRAN IV.

In the interim the following restriction applies:

All Arithmetic Statement Functions must have at least one argument.

COMPUTED GO TO (SPR 11-8807 GH)

Computed GO TO does not work if the expression is not of type INTEGER*2.

If the expression in a computed GO TO is not of type INTEGER*2, the compiler does not convert it. The flow of control under such a computed GO TO will be as if the expression were less than one or greater than the number of labels in the transfer list.

This problem is due to a coding error in the compiler parser. It is not patchable but will be considered in the next release of FORTRAN IV (Version 2) to be available early next year. It may be avoided in current versions by specifying INTEGER*2 expressions, assigning non-INTEGER*2 expressions to INTEGER*2 temporaries, and using those temporaries or by using the library functions as follows:

```
IFIX (real*4 expression)
IFIX (SNGL(real*8 expression))
IFIX (REAL(complex expression))
```

CLARIFICATION: COMPARING ASCII DATA ITEMS (SPR 11-8572 RB)

When more than one character is stored in an individual FORTRAN variable (as is done with A-FORMAT conversion), comparison between variables may not yield the expected ASCII collating sequence. For example, in the following FORTRAN code, the variable LESS is set to .TRUE.:

```
LOGICAL LESS
DATA A/'SF'/, B/'DP'/, LESS/.FALSE./
IF (A.LT.B) LESS = .TRUE.
```

The FORTRAN language does not specify the number of character data which may be stored in a data item; it merely guarantees that one character (minimum) can be accommodated in each variable. The PDP-11 format for ASCII data stores succeeding characters at ascending byte addresses; however, integer and floating-point comparison operations compare word, not byte, data, hence the items are compared byte-reversed from the ASCII sense. To correctly compare ASCII data in the PDP-11 environment, the LOGICAL*1 datatype should be used (which will cause byte comparisons to take place), or the program should be restricted, as per the FORTRAN standard, to storing only one ASCII character per variable or array element; e.g., A1 FORMAT.

IBEF NOT PROPERLY DECREMENTED (SPR 11-6716 HK)

IBEF is not properly decremented upon filling of the final subbuffer (problem with RTS).

This problem has been considered for the next release of FORTRAN/RT-11 Extensions. Meanwhile, ICMF may be used to detect the filling of the last subbuffer. It will always change.

LPS DEVICE CONFLICT CAUSED BY CALL SETR AFTER CALL RTS (SPR 11-8613 HK)

CALL SETR (-1,,,) executed too quickly after CALL RTS (,,,,,-1,,) causes next CALL RTS to produce LPS DEVICE CONFLICT error.

The problem is caused by the mechanism employed internally for resetting the flag to allow calls to RTS.

This problem is not patchable. It is being considered for the next release of FORTRAN/RT-11 Extensions.

Wide Peaks

PROBLEM:

A software malfunction may occur when the actual number of points processed on a peak becomes larger than 128.

SOLUTION:

Before installing the patch, upgrade the software to include Patch One which is located in Laboratory Applications-11 Release Notes (DEC-11-ALRNA-A-D). The solution consists of the following modifications of the Laboratory Applications-11 Version 3 source code, using the utility program EDIT to modify the file named PEAK.MAC. This updates the system to Patch Two.

```
R EDIT
*EBPEAK.MAC#3F.#C3##
*M.FINCB#-4CMOVB#JI#2,$. $$
*3EM##
*EX##
```

SOFTWARE PRODUCT LA-11	VERSION V3	
COMPONENT PEAK.MAC	VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION LA-11 Release Notes (DEC-11-ALRNA-A-D)	SEQUENCE 1*	PAGE OF 1 1
NEW <input type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/> 1	ORIGINAL DATE February 1976

PEAK PROBLEMS AND CORRECTIONS

Several serious errors exist in the documentation for the Peak Processing Module in the LA-11 Module Manual (DEC-11-ALMMA-A-D).

The first is an error of omission. On page 17-5 of the Module Manual, a required parameter, in the parameter table specification was not described. If an attempt is made to use the module with a parameter table missing this element, the module will most likely function improperly and may also alter locations following the parameter table itself.

The missing parameter should be positioned in the table immediately after the value for PPGATE. The parameter is referred to as PPMINS and as a word which specifies the minimum step to be considered as increase in the data. The parameter aides in the elimination of small or noise peaks. As a consequence, PPGATE*PPMINS sets the minimum detectable peak amplitude.

As a result, the list of parameters for the Peak Processing Module should look as follows.

```
PPDENS:  .WORD XXXXXX
PPWIDT:  .WORD XXXXXX
PPGATE:  .WORD XXXXXX
PPMINS:  .WORD XXXXXX
         .IF  DF,AUTOG$
PPAUTG:  .WORD XXXXXX
         .ENDC
PPINIT:  .BYTE YYY
PPEROR:  .BYTE YYY
PPVTBL:  .IF DF,AUTOG$
         .BLKW 67.
         .IFF
         .BLKW 53.
         .ENDC
```

The other documentation errors involved the length of the variable table which is described as the PPVTBL element in the parameter table for the Peak Processing Module on page 17-6 of the Module Manual.

The description should read as follows:

PPVTBL If AUTOG\$ is defined at assembly time, a variable table containing 67₁₀ words (e.g., .BLKW 67.). If AUTOG\$ is not defined, a variable table containing 53₁₀ words (e.g., .BLKW 53.).

SOURCE CORRECTIONS

A number of obscure errors has been found in the Peak Processing Module all dealing with points near zero when the input data has been auto-gained (PPAUTG=1, and AUTOG\$ is defined at assembly).

These errors should be eliminated by patching the source file named PEAK.MAC provided in the LA-11 V03 package. The RT-11 utility program EDIT should be used to produce the corrected file. Because of previous patches to this file, the file ID should be PEAK.3 before this patch.

The following text illustrates the procedure to be used in making the corrections. In the text, %% represents the device specification (e.g., DK1) of the device where the current version of PEAK.MAC will be found. Underlined characters () represent prompt characters from the system or EDIT program. The Ⓢ represents the altmode (ALT) or escape (ESC) character.

```
.R EDIT
*EB%%:PEAK.MACⓈ3F.ⓈC4ⓈⓈ
*M.FⓈⓈ,MXHⓈⓈAX CLR MXH(R5)
Ⓢ.ⓈⓈ
*EMⓈⓈ
*2F14.Ⓢ2AⓈⓈ
*I BGE DOD ;CHECK FOR NON-NEG POINT
CLR R1 ; IF NEG, THEN ZERO
CLR RⓈ
BR DOND
DOD:ⓈⓈ
*5AIDOND:ⓈⓈ
*EMⓈⓈ
*FAUT(ⓈⓈⓈⓈⓈ
*X MOV AUT(R5),R4
ⓈⓈ
*G4ⓈⓈⓈⓈⓈⓈ
*I BGE 2Ⓢ$ ;IF NON-NEG, BRANCH
CLR R3 ; OTHERWISE, ZERO POINT
BR 21$
2Ⓢ$:ⓈⓈ
*2G:ⓈⓈI21$:ⓈⓈ
*EXⓈⓈ
```

The result is a new file on the same device called PEAK.MAC containing the patch which corrects the problem having a new file ID of PEAK.4. Also, a copy of the old file remains on the device and is given the name PEAK.BAK.

LPS and AR-11 Vector and Status Register

PROBLEM:

The interrupt vectors and status register addresses for the LPS or AR-11 can float; that is, these addresses are jumper selectable. In the program SPARTA they are set as follows:

Vector = 340
Status Register = 170400

SOLUTION:

To run SPARTA in a machine that has these addresses at a different location, the program can be patched via PATCH or the source reassembled.

To patch GSPART.SAV:

Set LOC 2730 to vector address, and
LOC 2732 to status register address.

To patch VSPART.SAV:

Set LOC 1750 to vector address, and
LOC 1752 to status register address.

To reassemble under RT-11, set the following in SPASEM:

LPSV\$ = vector address
LPSS\$ = status register address.

SOFTWARE PRODUCT LA-11		VERSION V03	
COMPONENT SPARTA		VERSION VA02	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 1	PAGE OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE December 1975	

See Sequence 6 for information that was previously included in this article. (Change Bars indicate.) Please note the version number.

Using SPARTA and Floating Point Buffers (SPR 31-4631)

PROBLEM:

The apparent problem: The BSU command of SPARTA used in the context BSU FX may not zero the buffer FX.

The actual problem: Floating point arithmetic operations between two values, when both have an incorrect floating point format, may produce ambiguous results.

SOLUTION:

The solution consists of an explanation and a restriction. When floating point buffers are defined in SPARTA, only space is allocated. Thus, random data in these buffers may not be in the correct floating point format. To assure that the format of the values in a floating point buffer containing random data is correct, it should be acted upon by a scaler or a buffer of values known to be correctly formatted. (Note that "acted upon" implies the results of an operation should be left in the buffer to be corrected.) Examples of assuring correct format in F1:

- a. SMU F1 Ø
- b. SMU F1 1
- c. BAD F2 F1, where values in F2 are known to have the correct floating point format.

SOFTWARE PRODUCT LA-11		VERSION V3	
COMPONENT SPARTA		VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 2	PAGE 1 OF 1
NEW <input type="checkbox"/>	REPLACEMENT ARTICLE <input checked="" type="checkbox"/>	ORIGINAL DATE February 1976	

AR-11 Timing Problems With ADSAM and SPARTA.

PROBLEM:

A fatal software malfunction may occur when the Laboratory Applications-11 A/D sampling module (ADSAM) is used to sample an AR-11 Analog-to-Digital converter. In other cases the malfunction may not be apparent, but the user chosen sample period may be implemented with random inaccuracies.

NOTE: SPARTA is subject to this problem.

SOLUTION:

Before installing, upgrade the software to include Patch One. The solution consists of the following modifications:

- Using the utility program EDIT, modify the file named ADSAM.MAC. This updates the system to Patch Two.

```
.R EDIT
*EBADSAM.MAC$3F.$C5$$
*FADDL3:$0A$$
*I      BR      AD39
$$
*FADCRS:$A$$
*I      IF      DF,AR11$
          CLR      R5          ; CLEAR REMAINDER
          ENDC
$$
*2AX    BNE     ADCRS0          ; IF HIGH PART NON-ZERO, BRANCH
```

SOFTWARE PRODUCT LA-11	VERSION V3	
COMPONENT SPARTA & ADSAM	VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 3*	PAGE 1 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976

AR-11 Timing Problems With ADSAM and SPARTA

```

BIT      #177400, R0      ; OTHERWISE, TEST FOR ONLY 8 BITS OF INFO
BNE      ADCRS0          ; IF MORE, BRANCH
CMP      #5, R5          ; OTHERWISE, CHECK FOR ROUNDING UP
BGT      ADCRS1          ; IF NOT, BRANCH AND EXIT
INC      R0              ; OTHERWISE, ROUND UP
BR       ADCRS1          ; AND EXIT
      .IFF
  
```

```

$$
*ASK-AU$$
*AIADCRS0:$$
*EX$$
  
```

RESTRICTION:

This patch cures a serious problem but introduces a restriction in the choice of sampling rates with an AR11. To understand the restriction, one must first understand the manner in which the clock functions.

The clock monitors time via a control over the rate at which the clock actually TICKS and a clock tick counter. The clock done flag is set and possibly an interrupt occurs when the number of clock ticks since the GO pulse, at any rate, equals the number in the clock tick counter. The clock tick counter for the AR11 is a byte, i.e., it can only count modulo 256 decimal.

The rates at which the clock can tick are at 1, 10, or 100 microsecond and 1 or 10 millisecond intervals. If the hardware modification noted in the AR11 User's Manual has been made,

SOFTWARE PRODUCT LA-11		VERSION V3	
COMPONENT SPARTA & ADSAM		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 3*	PAGE OF 2 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

AR-11 Timing Problems With ADSAM and SPARTA

clock tick intervals of line frequency may also be chosen. This ends the description of the clock function.

The program receives its sampling rates in microseconds (ADRATE). The program selects the fastest clock ticking rate so that the sample period may be measured by 255 (decimal) or less clock ticks, in order that this count fits in the clock tick counter. The selection of this rate may, in some cases, cause the sample period to be somewhat different from what was originally selected.

For example:

- a. If a sampling rate of 250 (decimal) microseconds were selected via ADRATE, the resulting sampling rate would be exactly as chosen since 250 is less than 255 and clock tick rate of one microsecond intervals would be chosen.
- b. If a sampling rate of 500 (decimal) microseconds were selected, the resulting rate for this selection also would be exact, since a clock tick rate of 10 microseconds would be chosen with a clock tick counter of 50 (decimal) which is less than 255.
- c. If a sampling rate of 258 microseconds were selected, the resulting sampling rate would not be exact. In this case a clock tick rate of 10 microsecond intervals would be forced and the clock tick counter would be set to 26 (decimal). Thus the resulting sampling rate would be every 260 microseconds instead of the intended 258 microseconds.

SOFTWARE PRODUCT LA-11		VERSION V3	
COMPONENT SPARTA & ADSAM		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 3*	PAGE OF 3 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

AR-11 Timing Problems With ADSAM and SPARTA

2. After the changes are made to the sources, SPARTA should be updated by assembling and linking according to the instructions in the Laboratory Applications Program Manual.

NOTE: Always after patches are made to sources, these modules must be reassembled and relinked to any programs containing them. (SPARTA is a case in point for Patch Two.)

SOFTWARE PRODUCT LA-11	VERSION V3
COMPONENT SPARTA & ADSAM	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE PAGE 3* 4 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>
ORIGINAL DATE February 1976	

FFT Scaling Correction

PROBLEM:

Documentation concerning the resulting scale factor of an operation using the FFT MODULE (Module Manual, page 5-3) is inconsistent with the definitions of the algorithms used to perform the transforms (Program Manual, page 5-103).

SOLUTION:

The FFT algorithm described in the program manual on page 5-103 (additional note below) is the definition, though not the implementation, of the algorithm employed by the module. Thus, in all cases, to obtain the coefficients described on page 5-103 of the Program Manual, the resulting output should be multiplied by $2^{**}F$, where F represents the scale factor which is indicated.

1. In SPARTA, if the FFT MODULE is used, the scale factor F is indicated after an FFT, FAC, or FFC command by the program printing the line:

$$SF = F$$

(Program Manual, page 5-102)

2. When using the FFT MODULE directly, the scale factor F is stored at location SOLF.

(Module Manual, Page 5-2)

SOFTWARE PRODUCT LA-11		VERSION V3	
COMPONENT SPARTA		VERSION VA02	
SUBPROGRAM OR ADDITIONAL INFORMATION LA-11 PROGRAM MANUAL LA-11 MODULE MANUAL (DEC-11-ALPMA-A-D) (DEC-11-ALMMA-A-D)		SEQUENCE 4	PAGE 1 OF 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

FFT Scaling Correction

In addition, after all scaling is performed using the indicated scale factor, the output values from an inverse FFT operation are still a factor of N, where N is the size of the array, larger than the coefficients indicated by the algorithm on page 5-103 of the Program Manual. This is because, as is the custom with FFT routines, the module does not multiply the output from the inverse FFT by the leading 1/N factor. To obtain the exact coefficients indicated by the inverse transform definition, simply divide the output arrays by N.

Additional NOTE: The algorithm provided in the Program Manual for the inverse FFT is in error. It should read.

$$X_r = (1/N) \sum_{K=0}^{N-1} A_K \exp (2\pi jrK/N), r=0, \dots, N-1$$

The /N factor in the exponential term has been previously omitted in the documentation.

SOFTWARE PRODUCT LA-11		VERSION V3	
COMPONENT SPARTA		VERSION VA02	
SUBPROGRAM OR ADDITIONAL INFORMATION LA-11 PROGRAM MANUAL LA-11 MODULE MANUAL (DEC-11-ALPMA-A-D) (DEC-11-ALMMA-A-D)		SEQUENCE 4	PAGE OF 2 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

Scale Factor Correction for SPARTA Commands FAC and FCC

PROBLEM:

An error exists in the code for the FCC and FAC commands involving the scale factor which results from the associated operations.

SOLUTION:

The following patch corrects the problem. Using the RT-11 utility program EDIT, modify the file supplied in the IA-11 package called, SPCOR.MAC in the manner described below. Note that the underlined symbols are system prompt characters, and the \$ symbol represents the altmode (ALT key) or escape (ESC key) character.

```
.R EDIT
#EBSPCOR.MAC$3F.$C2$$
#F11$-C2$$
#FF.R$2AK$$
#M.F2*$-2D$. $$
#3EM$$
#FSCLF$0AJ$$
#XSUB C.SCLF,C.SCL1 ;SCL1=SCL1+SCL2+SCL1-LOG2(#POINTS)
$$
#3GLF$-C1$$
#EX$$
```

SOFTWARE PRODUCT LA-11		VERSION V03	
COMPONENT SPARTA, SPCOR.MAC, SPINIT.MAC		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION LA-11 PROGRAM MANUAL (DEC-11-ALPMA-A-D)		SEQUENCE 5*	PAGE OF 1 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976	

Scale Factor Correction for SPARTA Commands FAC and FCC

The resulting module FILE ID will be SPCOR.2.

After the changes are made to the source, SPARTA should be updated by assembling and linking according to the instructions in the Laboratory Applications-11 Program Manual (DEC-11-ALPMA-A-D).

SOFTWARE PRODUCT LA-11	VERSION V03	
COMPONENT SPARTA, SPCOR.MAC, SPINIT.MAC	VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION LA-11 PROGRAM MANUAL (DEC-11-ALPMA-A-D)	SEQUENCE 5*	PAGE 2 OF 2
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE February 1976

Data Displays Using LA-11

PROBLEM:

If a display (e.g., Tektronix 603) is jumpered to allow bipolar displays and A/D output values in the range ± 5 volts are displayed, the negative values appear on the upper half of the screen. The positive values will appear on the lower half of the screen if wraparound has been enabled. If not enabled, the positive values will not appear.

SOLUTION:

The LA-11 module ADSAM, and in general SPARTA, do not attempt to display acquired data as anything but A/D converted values in the range of 0 through 4095. To display data in a specific fashion, some user manipulation of the data may be required. To display the data described in the problem properly, the scalar 2047₁₀ should be subtracted from the buffer.

SOFTWARE PRODUCT LA-11	VERSION V03	
COMPONENT SPARTA	VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 6	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE March 1976

Data Preparation for SPARTA Commands FAC and FCC

This article contains general information concerning data handling techniques involved in the use of the FCC and FAC commands of SPARTA.

The FCC and FAC commands of SPARTA assume that the input buffers are in the correct format for the FFT module and command, i.e., the data is normalized and considered a 15 place, signed binary number. Thus, to get accurate results, the following procedure should be performed on arbitrary input data for the FCC and similarly for the FAC commands.

Assume, for FCC command, the data to be correlated is in buffers S1 and S2. First, the values in each buffer should be moved to a floating point buffer.

1. BMO S1 F1
2. BMO S2 F2

Then, the largest absolute value of the quantities in each buffer should be found.

3. BMM F1
 MAX VALUE = X1
 MIN VALUE = X2
4. BMM F2
 MAX VALUE = Y1
 MIN VALUE = Y2

Now, normalize each buffer by dividing it by the largest absolute quantity found in each buffer. Assume, for example:

$$|x1| > |x2|$$

AND

$$|y2| > |y1|$$

The buffers F1 and F2 will be normalized by dividing all values in the buffers by $|x1|$ and $|y2|$ respectively as follows:

5. SDI F1 $|x1|$

SOFTWARE PRODUCT IA-11		VERSION V03	
COMPONENT SPARTA		VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 7	PAGE 1 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

Data Preparation for SPARTA Commands FAC and FCC

6. SDI F2 |Y2|

To obtain maximum accuracy as 15 bit, signed twos complement fractions, multiply each buffer F1 and F2 by $(2^{15} - 1)$ or 32767. Move this data back into single precision buffers.

7. SMU F1 32767

8. SMU F2 32767

Move back to integer buffers for FCC command.

9. BMO F1 S1

10. BMO F2 S2

Do a cross correlation.

11. FCC S1

SF=L

Reconvert data or, remembering that the data is represented as binary fractions, use as normalized between + and - 1. To reconvert, put back in floating point buffer.

12. BMO S1 F1

Divide by 32767

13. SDI F1 32767

Multiply by |X1|, |Y2| and 2^{**L}

14. SMU F1 |X1|

15. SMU F1 |Y2|

16. SMU F1 N, where $N=2^{**L}$

The results are now correctly scaled.

SOFTWARE PRODUCT LA-11		VERSION V03	
COMPONENT SPARTA		VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 7	PAGE 2 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

Data Preparation for SPARTA Commands FAC and FCC

The differences for the FAC command are as follows:

1. Omit operations 2, 4, 6, 8, and 10.
2. Operation 11 should be changed to

FAC S1

3. Operation 15 changes to

SMU F1 |x1|

Note that this operation is changed and not omitted unless operation 14 is changed to reflect a |x1|**2 term as follows:

SMU F1 W , where W = |x1|**2

SOFTWARE PRODUCT LA-11	VERSION V03	
COMPONENT SPARTA	VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 7	PAGE 3 OF 3
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976

SPARTA Corrections for Point-Plot Display (VR14)

PROBLEM:

A fatal error occurs when SPARTA assembled and linked to be used with a point plot display (VR14) attempts to display a floating point buffer after the required DAS command has been executed.

SOLUTION:

The solution consists of patching a source file named SPCOM2.MAC provided in the LA-11 V03 package through the use of the RT-11 Utility Program, EDIT. The following text illustrates the procedure to be used in making the corrections. In the text, %%, represents the device specifications (e.g., DK1) of the device where the current version of SPCOM2.MAC will be found. Underlined characters () in the text represent prompt characters from the system or EDIT program. The \$ symbol represents the altmode (ALT) or escape (ESC) character.

```

R EDIT
*EB%%:SPCOM2.MAC#2F.#C2##
*FICT:~-J-3CIRO##
*GIRO:~-J-3CICT##
*EX##
  
```

The result is a new file SPCOM2.MAC containing the patch which corrects the error and a new file ID of 2. Also a copy of the old file remains on the device and is given the name SPCOM2.BAK. After the changes are made to the sources, SPARTA should be updated by assembling and linking according to the LA-11 Program Manual (DEC-11-ALPMA-A-D).

SOFTWARE PRODUCT LA-11		VERSION V03	
COMPONENT SPARTA		VERSION	
SUBPROGRAM OR ADDITIONAL INFORMATION SPCOM2.MAC		SEQUENCE 8*	PAGE OF 1 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

ADDING COMMANDS TO SPARTA (SPR 11-6704 JG)

PROBLEM: In SPARTA, source program SPCQA.MAC has an error such that more than one user added extension command cannot be implemented.

DISPOSITION: The solution consists of patching a source file named SPCQA.MAC provided in the LA-11 V03 package through the use of the RT-11 utility program EDIT. The following text illustrates the procedure to be used in making the corrections.

In the text, %% represents the device specifications (e.g., DK1) of the device where the current version of SPCQA.MAC will be found.

Underlined characters () in the text represent prompt characters from the system or EDIT program. The (\$) symbol represents the altmode (ALT) or escape (ESC) character.

```
.R EDIT
*EB%%:SPCQA.MAC ($) 2F. ($) C2 ($$)
*FDE57: ($) G@ ($) -DJD ($$)
*AI      MOV      (R1),R1
($$)
*EX ($$)
```

.

The result is a new file SPCQA.MAC containing the patch which corrects the error and a new file ID of 2. Also a copy of the old file remains on the device and is given the name SPCQA.BAK.

After the changes are made to the source file, SPARTA should be updated by assembling and linking according to the instructions in the Laboratory Applications-11 Program Manual (DEC-11-ALPMA-A-D).

CORRECTION FOR THE DFV COMMAND WITH POINT PLOT DISPLAY (VR14\$) (JG)

PROBLEM: A fatal error occurs in SPARTA assembled with VR14\$ defined, when the user attempts to issue a DFV display free cursor values command.

DISPOSITION: To eliminate this problem, the source file SPOV7.MAC, which was provided in the LA-11 V03 package, should be patched using the RT-11 utility program EDIT. The following text illustrates the procedure to be used in making the corrections. In the text, \$\$\$ represents the device specification, e.g., DK1, of the device where the current version of SPOV7.MAC will be found. Underlined characters () in the text represent prompt characters from the system or EDIT program. The © symbol represents the altmode (ALT) or escape (ESC) character.

```
  R EDIT  
  ERXXX: SPOV7 MAC©3F. ©C2©©  
  3F$ADR©I, $RI©©  
  EX©©
```

The result is a new file on the same device called SPOV7.MAC containing the patch which corrects the error having a new file ID of SPOV7.2. A copy of the old file remains on the device of origin and is given the name SPOV7.BAK.

After these changes have been made to the source, the new SPOV7.MAC should be assembled and then SPARTA should be updated by linking according to the instructions in the LA-11 Program Manual (DEC-11-ALPMA-A-D).

GENERAL SUBROUTINE MODULE FOR EAE (JG)

PROBLEM: An assembly error occurs when the General Subroutine module (GENS.MAC) is assembled with the conditional assembly parameter EAE defined.

DISPOSITION: The solution consists of patching the source file named GENS.MAC provided in the LA-11 V03 package through the use of the RT-11 utility program EDIT. The following text illustrates the procedure to be used in making the corrections.

In the text, %% represents the device specification, e.g., DK1, of the device where the current version of GENS.MAC can be found. The underlined characters () in the text represent prompt characters from the system or EDIT program. The Ⓢ symbol represents the alt-mode (ALT) or escape (ESC) character.

```
.R EDIT
*EB%%:GENS.MAC Ⓢ 3F. Ⓢ C5 Ⓢ Ⓢ
*F1 $ Ⓢ 2A11$: Ⓢ Ⓢ
*EX Ⓢ Ⓢ
```

.

The result is a new file GENS.MAC containing the patch which corrects the error and a new file ID of 5. Also, a copy of the old file remains on the device and is given the name GENS.BAK.

HOW TO START DATA ACQUISITION WHEN CSTART EQUALS ZERO (JG)

PROBLEM: The documentation for the program THRU in the Laboratory Applications-11 Program Manual does not explain how to produce a call to CSAMPL when CSTART equals zero.

DISPOSITION: The following paragraph will supply the necessary information.

If CSTART is set equal to zero, the program THRU uses terminal input as a prompt to start and stop data acquisition, via calls to CSAMPL. The program continually monitors terminal input in its idle time.

NOTE: Terminal input is recognized only when terminated by a carriage return.

In mode one, any terminal input, including just a carriage return, will affect the start of the acquisition process. Any subsequent terminal input will cause data acquisition to stop. In modes two and three, the values entered through the terminal will be passed to the Continuous Sampling module, which in turn will produce the corresponding effect based on the values given for CSSIN, CSHIN, and CSSTOP.

Building MU BASIC/RT-11 Under RT-11 V02C

PROBLEM:

When building MU BASIC/RT-11 under the RT-11 V02C monitor, use of the LUSER configuration file as described on page 1-26 of the MU BASIC System Installation Guide (DEC-11-LIBMA-A-D) will produce the message:

ILLEGAL DEVICE/UNIT SPECIFICATION

This will occur for MU BASIC/RT-11 binary kits distributed on all media except floppy disk.

SOLUTION:

This problem is due to the fact that the available device list in the LUSER configuration file contains the designator for the card reader handler. However, in the RT-11 V02C monitor, CR: has been removed from the monitor tables and thus is not recognized.

If it is desired to use CR: with BASIC, the RT-11 monitor must be patched as described in the RT-11 System Generation Manual (DEC-11-ORGMA-A-D), section 4.6.7.

Alternatively, if CR: is not to be used with BASIC, the following procedure should be followed to remove CR: from the LUSER configuration file; the underlined portions are typed by the user, <CR> represents the return key, and \$ represents the key labeled ESC or ALT MODE.

SOFTWARE PRODUCT MU BASIC/RT-11	VERSION V01-01
COMPONENT N/A	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION MU BASIC System Installation Guide (DEC-11-LIBMA-A-D)	SEQUENCE PAGE 1* 1 OF 2
NEW <input type="checkbox"/> REPLACEMENT ARTICLE <input type="checkbox"/> 1	ORIGINAL DATE February 1976

Building MU BASIC/RT-11 Under RT-11 V02C

.R EDIT <CR>
*ERUSER.DØØ\$\$
*EWUSER.DØØ\$\$
*FCR\$ØKKEKX\$\$

SOFTWARE PRODUCT MU BASIC/RT-11	VERSION V01-01
COMPONENT N/A	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION MU BASIC System Installation Guide (DEC-11-LIBMA-A-D)	SEQUENCE PAGE 1* 2 2 OF
NEW <input type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/> 1
ORIGINAL DATE February 1976	

REMOTE TERMINAL SUPPORT ON MODEMS (SPR 11-6939 SJ)

PROBLEM: The CE ON option available on certain modems (notably the BELL 103A3) will cause the MU BASIC remote user to be disconnected 30 seconds after dialing in. This will occur on versions of MU BASIC/RT-11 distributed on all media, except floppy disk.

DISPOSITION: The following patch will allow MU BASIC to operate correctly on the BELL 103A3 or an equivalent modem. The underlined portions of the patch are typed by the user. <CR> indicates the RETURN key. System dependent information is represented by a field of Xs.

NOTE: Do not make the following patches, if the distribution media for MU BASIC/RT-11 was floppy disk.

```
.R PATCH<CR>
PATCH VXX-XX
FILE NAME--
*MUBAS.SAV<CR>
*14012#0R
*0,2020/ 100701      100705<CR>
*1464#1R
*1,6\   XXX          102<CR>
*E
```

The file MUBAS.REL cannot be patched with the preceding input. The following patch will effect the equivalent change in MUBAS.REL, or if BASIC has been relinked by the user.

```
.R PATCH0<CR>
*OPEN<CR>
ENTER INPUT FILE *MUBS2.OBJ<CR>
ENTER OUTPUT FILE *MUBS2.TMP<CR>
*BYTE MUBS2+2020=#305<CR>
*EXIT<CR>
ENTER CHECKSUM: 125111<CR>
STOP --

.R PATCH0<CR>
*OPEN<CR>
ENTER INPUT FILE *MUBR.OBJ<CR>
ENTER OUTPUT FILE *MUBR.TMP<CR>
*BYTE BASICR+6=#102<CR>
*EXIT<CR>
ENTER CHECKSUM: 105645<CR>
STOP --
```

REMOTE TERMINAL SUPPORT ON MODEMS

```
.R PATCHO<CR>  
*OPEN<CR>  
ENTER INPUT FILE *MUBRNS.OBJ<CR>  
ENTER OUTPUT FILE *MUBRNS.TMP<CR>  
*BYTE BASICR+6=#102<CR>  
*EXIT<CR>  
ENTER CHECKSUM: 105645<CR>  
STOP --
```

```
.R FIP<CR>  
*MUBS2.OBJ=MUBS2.TMP/R<CR>  
*MUBR.OBJ=MUBR.TMP/R<CR>  
*MUBRNS.OBJ=MUBRNS.TMP/R<CR>  
*<CTRL/C>
```

The new object files can be used to relink MUBAS.REL as described in Section 4.1 of the MU BASIC System Installation Guide (DEC-11-LIBMA-A-D). The resulting version number of MU BASIC/RT-11 is V01-01B.

OVERLAY... LINE WORKS INCORRECTLY

PROBLEM: The OVERLAY with a line number specified closes open files, redimensions arrays, and tries to redefine DEF FNs.

DISPOSITION: There is no practical patch for this problem.

The problem can be overcome at the BASIC source level by replacing OVERLAY... LINE with two statements: OVERLAY (without line) and GOTO line number. The GOTO must be on the line following the OVERLAY statement, and it must not be replaced during the overlay process; for example,

```
1000 OVERLAY "OVRLY1" LINE 1200
```

is replaced by

```
1000 OVERLAY "OVRLY1"  
1001 GOTO 1200
```


SCHEDULER DOES NOT PROPERLY SET PROCESSOR PRIORITY

PROBLEM:

REMOTE-11 may hang when a CTRL/C is issued to the foreground. The scheduler does not properly set the processor priority when polling users for runability.

SOLUTION:

Do the following edit to REMCOM.MAC. Rebuild the system. This edit assumes that REMCOM.MAC is on the system device and that sufficient space exists on the device to copy the file.

```
FILE: REMCOM.MAC
ENOT ENOUGH BUFFER ROOM FOR BLOCK-ON MODE]
*F SCHED:
SCHED:
*L 300
      .MTPS    300           #DO THIS PROTECTED
*C/3/#3/
      .MTPS    #300         #DO THIS PROTECTED
*KCL
FILE:
```

NOEDIT- Ø HALTS (HK)

PROBLEM: If NOEDIT is defined, the generated system halts immediately when run.

DISPOSITION: In order to provide overlays and sizing, the largest overlay, REMED2, must be called to determine the maximum size. The NOEDIT conditional section must be corrected to handle this call properly.

Edit REMED2.MAC as follows. Rebuild REMOTE.

```
FILE: REMED2.MAC
[PAGE 11]
*L02
    .TITLE REMOTE V01-X02
*C/2/3/
    .TITLE REMOTE V01-X03
*L02
EDIT=002.
*C/2/3/
EDIT=003.
*L02
‡ EDITOR SECTION PART 2: VERSION LEVEL V01-X02: PATCH LEVEL A
*C/02/03/
‡ EDITOR SECTION PART 2: VERSION LEVEL V01-X03: PATCH LEVEL A
*PL ENDP
    .GLOBL ENDP
*L
    ENDP=.
*
    MOV    ‡ENDP,%2
    RTS    ‡7
*CLOSE
FILE:
```

NUSERS=1 STAYS IN A FILE MESSAGE LOOP (DA)

PROBLEM: FILE message is given no matter what the user types for a file name.

DISPOSITION: The REMOTE-11 routine CKFILE is called to ensure that there are no FILE name conflicts between users. With only one user, this is unnecessary.

Edit REMEDC.MAC as follows. Rebuild the REMOTE system. This edit assumes that REMEDC.MAC is on the system device.

```
FILE: REMEDC.MAC
[PAGE 1]
*L 2
      .TITLE REMOTE V01-X02
*C/2/3
      .TITLE REMOTE V01-X03
*L 2
EDIT=002.
*C/2/3
EDIT=003.
*L X01
; EDITOR COMMON SECTION:  VERSION LEVEL V01-X01:  PATCH LEVEL A
*C/X01/X03
; EDITOR COMMON SECTION:  VERSION LEVEL V01-X03:  PATCH LEVEL A
*PF TESTNM:
L00055 LINES READ IN]
TESTNM:
*PL 1*:
1*:
*
      .IFF
      ADD      #2,(SP)          ;SHOW A SUCCESSFUL RETURN FOR 1 USER

*CLOSE
FILE:
```

INCORRECT SWAP AREA ALLOCATION FOR FOUR OR MORE USERS (HK)

PROBLEM: Swap area references for user numbers greater than three are offset by four.

DISPOSITION:

Perform the following edit on REMEDC.MAC. The edit shown assumes that the file is on the system device. Rebuild the system.

```
FILE: REMEDC.MAC
LPAGE 11
*L X01
        .TITLE  REMOTE V01-X01
*C/X01/X02/
        .TITLE  REMOTE V01-X02
*F EDIT
EDIT=001.
*C/1/2/
EDIT=002.
*PF FSWAP
FSWAP:
*F2
2#:    TST      (SP)+          ;CLEAN UP STACK
*
        .IF LE  NUSERS-3

*L
        MOV     #11*400+<NUSERS*4>,R0 ;ASSUME A .WRITW OPERATION
*
        .ENDC
        .IF GT  NUSERS-3
        MOV     #11*400+<NUSERS*4+4>,R0
        .ENDC

*CLOSE
FILE:
```


REBOOT FROM SATELLITE DURING EDIT HANGS HOST

PROBLEM: Should a power fail occur in a satellite during editing,
it is not possible to reboot.

DISPOSITION: Edit REMTTY.MAC as shown below. This edit assumes that
there is sufficient space for BLOCK-ON mode and that there is room
on the system disk for the newly edited file. Rebuild your system.

```
FILE: REMTTY.MAC
[PAGE 1]
*L 03
      .TITLE REMOTE V01-X03
*C/3/4/
      .TITLE REMOTE V01-X04
*L 03
EDIT=003.
*C/3/4/
EDIT=004.
*L 03
; TTY SECTION:  VERSION LEVEL V01-X03:  PATCH LEVEL A
*C/3/4/
; TTY SECTION:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*PF ,TTIN
.TTIN:
*L
      TSTB   DDCMP(R5)      ;IN DDCMP MODE?
*C/TST/CLR/
      CLRB   DDCMP(R5)      ;IN DDCMP MODE?
*C/IN/CLR/
      CLRB   DDCMP(R5)      ;CLR DDCMP MODE?
*C/?//
      CLRB   DDCMP(R5)      ;CLR DDCMP MODE
*L
      BNE    3#             ;YES:  BEAT IT OUT OF HERE
*D
*L 3
3#:   RETURN                ;RETURN TO CALLER
*C/3#//
      RETURN                ;RETURN TO CALLER
*PF CTRLC:
[00055 LINES READ IN]
CTRLC:
*L
      BIS    #4000,SW2(R5)  ;SET USER'S GO BIT
*
      CLR    TTOCLO(R5)     ;CLEAR THE CTRL-O FLAG

*CLOSE
FILE:
```

REMOTE/RT-11, V01
REMTTY.MAC

Seq 5
1 of 1

HARD ERROR ON LOOKUP IS FATAL (HK)

PROBLEM: Lookups on devices that are not present or not enabled, cause fatal system errors.

DISPOSITION: A .SERR programmed request must precede each instance of .LOOKUP. One is missing. The following patch will rectify the situation. The patch assumes that Block-on made is enabled, that REMEDI.MAC is on the system device, and that there is sufficient space on the system to perform the EDIT. When the EDIT is complete, rebuild your system.

```
FILE: REMEDI.MAC
CPAGE 11
*L 02
    .TITLE REMOTE V01-X02
*C/OTE/ED1/
    .TITLE REMEDI V01-X02
*C/2/3/
    .TITLE REMEDI V01-X03
*L 02
EDIT=002.
*C/2/3/
EDIT=003.
*L 02
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X02:  PATCH LEVEL A
*C/2/3/
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X03:  PATCH LEVEL A
*PF EDIT1
10055 LINES READ INJ
EDIT1: CLR B  OVLYSW(R5)      ;SHOW OVERLAY #1 ACTIVE
*L RENA
; CAN RENAME IT.
*1
XX:
;DUMMY TAG TO PREVENT MACRO ERRORS
*L .HERR
.HERR
;ALLOW MONITOR ERRORS NOW
*DP
.CLOSE 1
;CLOSE FILE
*L ALR
; ALREADY EXISTS.
*1
8*: .HERR
;ALLOW MONITOR ERRORS NOW
```

REMOTE/RT-11, V01

Seq 6*
1 of 2

HARD ERROR ON LOOKUP IS FATAL

```
*L
8#:      JMP      START          †RESTART EDITOR
*C/8#://
        JMP      START          †RESTART EDITOR
*F 7
7#:      .HERR                    †ALLOW MONITOR ERRORS NOW
*DF
        .PRINT  †DIRERR          †YES:  TELL USER
*C//7#://
7#:      .PRINT  †DIRERR          †YES:  TELL USER
*2L BR
        BR      8#              †NO:  RESTART THEN
*I
        .SERR                    †CATCH MONITOR ERRORS

*L BCC
        BCC     5#              †YES
*I
        TSTB   0#52            †ERROR OCCURRED:  WAS IT HARD?
        BMI    7#              †YES:  TELL USER AND QUIT

*PL MOV
        MOV     R0,FSIZE(R5)    †    INPUT FILE SIZE PLUS 50 BLOCKS.
*I
6#:      .HERR                    †ALLOW MONITOR ERRORS NOW

*L
6#:      .SCHED                    †SCHEDULE SOMEONE ELSE
*C/6#://
        .SCHED                    †SCHEDULE SOMEONE ELSE
*CL
```

SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL (HK)

PROBLEM: If the satellite has been in terminal mode, the secondary load feature does not work. If the secondary file does not exist when the request occurs, the system will hang.

DISPOSITION: In order to completely fix this problem, edits must be made to four REMOTE modules: REMCOM, REMDDC, REMTTY, and REMED1. The edits, using LETTER, follow. After completing the edits, the four modules should be assembled. Finally, the system should be rebuilt.

NOTE: It is strongly recommended that these edits be performed even if you do not expect to use the secondary load facility since they affect other portions of REMOTE.

```
FILE: REMCOM.MAC
CPAGE 11
*L 01
      .TITLE REMOTE V01-X00
*C/OTE/COM/
      .TITLE REMCOM V01-X00
*C/00/03/
      .TITLE REMCOM V01-X03
*L 02
EDIT=002.
*C/2/3/
EDIT=003.
*L 00
; COMMON SECTION:  VERSION LEVEL V01-X00:  PATCH LEVEL A
*C/00/02/
; COMMON SECTION:  VERSION LEVEL V01-X02:  PATCH LEVEL A
*PL GO
      .GLOBL QUIT,STACK,SAVFIL,GO
*AP ,WAIT
      .GLOBL QUIT,STACK,SAVFIL,GO,WAIT
*CL
FILE:
```

SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL

```
FILE: REMDDC.MAC
IPAGE 13
*L 01
    .TITLE REMOTE V01-X01
*C/OTE/DDC/
    .TITLE REMDDC V01-X01
*C/X01/X02/
    .TITLE REMDDC V01-X02
*L 01
EDIT=001.
*C/1/2/
EDIT=002.
*L 01
; DOWN-LINE LOADING SECTION:  VERSION LEVEL V01-X01:  PATCH LEVEL A
*C/X01/X02/
; DOWN-LINE LOADING SECTION:  VERSION LEVEL V01-X02:  PATCH LEVEL A
*PL DWAIT
    .GLOBL GETLIN,ERR,ILLLDA,DDCMP0,TTOPUT,DWAIT
*AP ,WAIT
    .GLOBL GETLIN,ERR,ILLLDA,DDCMP0,TTOPUT,DWAIT,WAIT
*PL /LDA/
[00055 LINES READ IN]
    .RAD50 /LDA/
*L
    MOV    R0,INFIL+6(R5)
*I
6$:    TSTB   DDCMP(R5)           ;IN DDCMP MODE?
      BEQ    8$                  ;NO:  TREAT AS NORMAL REQUEST
      CLR    DDCMP(R5)          ;YES:  CLEAR MODE AND DONE FLAGS
      ;ASSUME SEC MODE LOAD REQUEST
10$:   .SERR
      .LOOKUP 0,#INFIL          ;SEE IF FILE EXISTS
      BCC    2$                  ;IT DOES
      .HERR
      CALL   WAIT                ;ALLOW MONITOR ERRORS NOW
      TSTB   DDCMP(R5)          ;WAIT A FEW SECS BEFORE TRYING AGAIN
      BEQ    10$                 ;HAS A NEW MSG COME IN?
      JMP    DDCMPF             ;NO:  SEE IF FILE EXISTS NOW
      ;YES:  HANDLE IT
*L
6$:    .LOOKUP 0,#INFIL          ;SEE IF FILE EXISTS AND OPEN IT IF IT DOES
*C/6/8/
8$:    .LOOKUP 0,#INFIL          ;SEE IF FILE EXISTS AND OPEN IT IF IT DOES
*CL
FILE:
```

SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL

```
FILE: REMTTY.MAC
IPAGE 11
*L 04
      .TITLE  REMOTE V01-X04
*C/4/5/
      .TITLE  REMOTE V01-X05
*L 04
EDIT=004.
*C/4/5/
EDIT=005.
*L 04
; TTY SECTION:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*C/4/5/
; TTY SECTION:  VERSION LEVEL V01-X05:  PATCH LEVEL A
*PL TKS
      .GLOBL  TKS
*AF ,GO
      .GLOBL  TKS,GO
*PF .TTIN
.TTIN:
*L
*DP  CLRB  DDCMP(R5)      ;CLR DDCMP MODE
      .MTPS  #200        ;PROTECT WHILE LOOKING AT DYNAMIC FLAGS
*I   TSTB  GO(R5)        ;IS THIS USER EDITING?
      BEQ   3#           ;NO
      CLRB  DDCMP(R5)    ;YES:  NO DDCMP MSGS ALLOWED HERE
*L
      TST   TTILCT(R5)   ;GOT ANY DATA
*C//3#:/
3#:#  TST   TTILCT(R5)   ;GOT ANY DATA
*PF NOINPT
NOINPT: INCB  NEEDLN(R5) ;INDICATE A LINE IS NEEDED
*NP 2
      .WAIT              ;WAIT TILL LINE GETS IN
*I   TSTB  DDCMP(R5)    ;DID WE END UP IN DDCMP MODE?
      BEQ   .TTIN        ;NO
      TSTB  GO(R5)      ;MAKE SURE WE'RE NOT EDITING
      BNE   .TTIN        ;WE ARE
      CLRB  NEEDLN(R5)  ;GET RID OF NEED FOR LINE
      RETURN            ;RETURN TO REMCOM
```

REMOTE/RT-11, V01

Seq 7*
3 of 4

SECONDARY MODE PROGRAM LOAD FEATURE NOT COMPLETELY FUNCTIONAL

```
*L
      BR      .TTIN          ;PROCESS LINE NOW
*D
*CL
FILE:

FILE:  REMED1.MAC
(PAGE  11
*L 03      .TITLE  REMED1 V01-X03
*C/3/4/   .TITLE  REMED1 V01-X04
*L 03
EDIT=003.
*C/3/4/
EDIT=004.
*L 03
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X03:  PATCH LEVEL A
*C/3/4/
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*PF EDIT1
[00055 LINES READ IN]
EDIT1:  CLRB  OVLYSW(R5)    ;SHOW OVERLAY #1 ACTIVE
*NP 2
      .CLOSE  1
*I
      INCB   GO(R5)         ;INDICATE THIS USER HAS BEGUN EDITING
*PL INCB
      INCB   GO(R5)         ;INDICATE THIS USER HAS BEGUN EDITING
*DP
      BITB   #1,SWITCH(R5)  ;IF THERE WAS NO INPUT FILE OPEN, GO
*CL
FILE:
```


ONE SECOND TIMER FOR LINE TIMEOUTS IS SET INCORRECTLY (HK)

The value given to the .MRKT request does not provide the one second interval required by DDCMP protocol. There is no provision for line frequencies other than 60 Hz.

This problem will not cause failures of the system. However, the following edit will provide more accurate conformance to DDCMP protocol. Edit REMMSG.MAC and reassemble it with PARAM.MAC. An additional parameter, HZ, is available. If not defined, 60 Hz is assumed. If defined, it must be set to the line frequency in Hz. It should be in PARAM.MAC. Rebuild the system.

```
FILE: REMMSG.MAC
IPAGE 11
*L 03
      .TITLE  REMOTE V05-X03
*C/3/4/
      .TITLE  REMOTE V05-X04
*L 03
EDIT=003.
*C/3/4/
EDIT=004.
*L 03
; MESSAGES AND COMMANDS SECTION:  VERSION LEVEL V05-X03:  PATCH LEVEL A
*C/3/4/
; MESSAGES AND COMMANDS SECTION:  VERSION LEVEL V05-X04:  PATCH LEVEL A
*PL MACRO
;      MACRO SUPPORT IN THE EDITOR
*I
;      60 HZ LINE FREQUENCY

*PL NOMAC
;      IF DEFINED "NOMAC", THE MACRO SUPPORT IN THE EDITOR IS
```

```
*L
;
;          REMOVED.
*1
;          IF DEFINED "HZ", THE LINE FREQUENCY IS THE VALUE OF HZ,
;          E.G. HZ=50.
;          .IIF      NDF      HZ,      HZ=60.

*PF INTERV
INTERV: .WORD  0,60          ;MARK TIME INTERVAL (1 SECOND)
*C/60/HZ/
INTERV: .WORD  0,HZ        ;MARK TIME INTERVAL (1 SECOND)
*CL
FILE:
```

REMOTE/RT-11 V01
REMMSG.MAC

Seq 8*
2 of 2
August 1976

LINE FEEDS MAY CAUSE SYSTEM ERRORS--ASSEMBLY ERROR WITH DIAL AND
NODDC (HK)

On terminals with high baud rates, line feeds may cause
an error in maintaining the separate user states.

When both DIAL and NODDC are defined, an assembly
error occurs in REMTTY.

The following edit to REMTTY.MAC corrects both problems.
Block on mode is assumed. The files are assumed to be
on the system device. When the edit is complete,
reassemble REMTTY.MAC with PARAM.MAC and rebuild the
system.

```
FILE: REMTTY.MAC
[PAGE 1]
*L 05
        .TITLE  REMOTE V01-X05
*C/OTE/TTY/
        .TITLE  REMTTY V01-X05
*C/5/6/
        .TITLE  REMTTY V01-X06
*L 05
EDIT=005.
*C/5/6/
EDIT=006.
*L 05
; TTY SECTION:  VERSION LEVEL V01-X05:  PATCH LEVEL A
*C/5/6/
; TTY SECTION:  VERSION LEVEL V01-X06:  PATCH LEVEL A
*PF OPUT
[00055 LINES READ IN]
OPUT:
*PF 3
3#:      INC      2(R2)          ;INCREMENT COUNT IN BUFFER
*L
        .CLRFS
```

REMOTE/RT-11 V01
REMTTY.MAC

Seq 9*
1 of 2

```
*DF      MOV      #100,@TPS(R5)    ;ENABLE TELEPRINTER
*PF 8
I00055 LINES READ INJ
B#:      BIS      R3,@TKS(R5)     ;NOW RE-ENABLE INTRPTS
*I
        .IF NDF NODDC

*L JMP
        JMP      DMEXIT           ;NO, JUST RESTORE REGS AND RTI
*I
        .ENDC                    ;NODDC

*CL
FILE:
```

REMOTE/RT-11 V01
REMTTY.MAC

Seq 9*
2 of 2
August 1976

PROPER GENERATION OF REMOTE IS DEPENDENT ON MODULE ORDER (HK)

REMED2.MAC must be the last module when linking.
Since it may be omitted if editing is not required,
this may cause problems.

Use of the .LIMIT directive will solve the above
problem. Edit REMUSR.MAC and REMED2.MAC as shown
below. Note that these edits assume block-on
mode, files on system device, and enough room on
the system device for the edit. When the editing
is complete, reassemble the two modules and relink
the system.

```
FILE: REMUSR.MAC
IPAGE 11
*L 03
      .TITLE  REMOTE V01-X03
*C/3/4/
      .TITLE  REMOTE V01-X04
*L 03
EDIT=003.
*C/3/4/
EDIT=004.
*L 03
; USER IMPURE AREAS:  VERSION LEVEL V01-X03:  PATCH LEVEL A
*C/3/4/
; USER IMPURE AREAS:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*PL ENDP
[00055 LINES READ IN]
      .GLOBL  MSGO,ENDP,SCHED
*C/,ENDP//
      .GLOBL  MSGO,SCHED
*PL ENDP
[00055 LINES READ IN]
      CALL   ENDP           ;CALCULATE SIZE OF EACH USERS AREA
*C/CALL/MOV/
      MOV    ENDP           ;CALCULATE SIZE OF EACH USERS AREA
```

REMOTE/RT-11 V01
REMUSR.MAC, REMED2.MAC

Seq 10*
1 of 2

```

*C/F/P,R2/
      MOV      ENDP,R2          ;CALCULATE SIZE OF EACH USERS AREA
*L
      BR       Z#
*I
      .LIMIT
      ENDP=    .-2

*CL
FILE:

```

```

FILE:  REMED2.MAC
[PAGE  1]
*L 03

```

```

      .TITLE  REMOTE V01-X03
*C/3/4/
      .TITLE  REMOTE V01-X04

```

```

*L 03
EDIT=003.

```

```

*C/3/4/
EDIT=004.

```

```

*L 03
; EDITOR SECTION PART 2:  VERSION LEVEL V01-X03:  PATCH LEVEL A

```

```

*C/3/4/
; EDITOR SECTION PART 2:  VERSION LEVEL V01-X04:  PATCH LEVEL A

```

```

*PL .IF
      .IF DF NOEDIT

```

```

*DP7

```

```

*PL ENDP
      .GLOBL  CWD,CWDS,ILLCD,.NOL,EOF,ILLMOD,ENDP

```

```

*C/,ENDP//
      .GLOBL  CWD,CWDS,ILLCD,.NOL,EOF,ILLMOD

```

```

*PF ENDP
ENDP:          ;END OF REMOTE-11

```

```

*DP4
      .END

```

```

*CL
FILE:

```

ASCII CODES 173 AND 174 DO NOT PRINT (HK)

These codes are converted to "lower case" by the terminal output routine. Thus, a vertical bar (174) becomes a backslash (134).

The following edit to REMTTY.MAC will fix this problem. The edit assumes block-on mode and that the file is on the system disk. When the edit is complete, assemble REMTTY.MAC and rebuild your system.

```
FILE: REMTTY.MAC
CPAGE 11
*L 06      .TITLE  REMTTY V01-X06
*C/6/7    .TITLE  REMTTY V01-X07
*L 06
EDIT=006.
*C/6/7/
EDIT=007.
*L 06
; TTY SECTION:  VERSION LEVEL V01-X06:  PATCH LEVEL A
*C/6/7/
; TTY SECTION:  VERSION LEVEL V01-X07:  PATCH LEVEL A
*PL 174
I00055 LINES READ INJ
      CMPB      R0,#174          ;ADD ESCAPES AND RUBOUTS?
*C/4/2/
      CMPB      R0,#172          ;ADD ESCAPES AND RUBOUTS?
*CL
FILE:
```

IMPROPER FILLER HANDLING FOR VT05 (HK)

The command FILL only provides filler characters after carriage return. VT05 terminals require filler after line feed.

The following edit will allow specification of filler after line feed. A negative argument to the FILL command will invoke this feature. A positive argument will cause filler after carriage return. The edit assumes block-on mode (always attainable in a one-user system). The file, REMED1.MAC, is assumed to be on the system device. After completing the edit, assemble REMED1.MAC and rebuild REMOTE.

```

FILE:  REMED1.MAC
LPAGE   13
*L 04
      .TITLE  REMED1 V01-X04
*C/4/5/
      .TITLE  REMED1 V01-X05
*L 04
EDIT=004.
*C/4/5/
EDIT=005.
*L 04
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*C/4/5/
; EDITOR SECTION PART 1:  VERSION LEVEL V01-X05:  PATCH LEVEL A
*PL FILLCT
      .GLOBL  FILLCT,SW2,MACARG,CURMAC
*C/,/,FILLIT,/
      .GLOBL  FILLCT,FILLIT,SW2,MACARG,CURMAC
*PF FILL
FILL:  CLRB      OVLYSW(R5)          ;SHOW OVERLAY #1 ACTIVE
*NP2
      BMI      1$                    ;NEGATIVE ARG IS ILLEGAL
*C/IS ILLEGAL/FOR LINE FEED/
      BMI      1$                    ;NEGATIVE ARG FOR LINE FEED
    
```

REMOTE/RT-11 V01
 REMED1.MAC

Seq 12*
 1 of 2


```
*I      MOVB      #15,FILLIT(R5)  ;RESTORE FILL CHAR TO CR

*NP
      MOVB      R1,FILLCT(R5)    ;SAVE ARGUMENT
*0//2#://
2#:#    MOVB      R1,FILLCT(R5)    ;SAVE ARGUMENT
*F 1
1#:#    JMP      PRTO              ;NEGATIVE ARG IS ILLEGAL
*0
1#:#    NEG      R1                ;GET COUNT FOR LF
      MOVB      #12,FILLIT(R5)    ;MAKE LF THE FILL CHAR
      BR       2#

*CL.
FILE:
```

SYSTEM CRASHES IF RUN IN FOREGROUND WITHOUT /N (HK)

Buffer space allocation algorithm is in error. This problem only occurs in a foreground system when the /N switch is not used.

Perform the following edit to REMUSR.MAC. The edit shown assumes block-on mode and the file is on the system device. Assembly REMUSR.MAC and rebuild REMOTE.

```
FILE: REMUSR.MAC
IPAGE 11
*L 04
      .TITLE  REMOTE V01-X04
*C/4/5/
      .TITLE  REMOTE V01-X05
*L 04
EDIT=004.
*C/4/5/
EDIT=005.
*L 04
; USER IMPURE AREAS:  VERSION LEVEL V01-X04:  PATCH LEVEL A
*C/4/5/
; USER IMPURE AREAS:  VERSION LEVEL V01-X05:  PATCH LEVEL A
*PF X#
I00055 LINES READ INJ
X#:   MOV      TEMPO,R1          ;USE LOW CORE ON NON MMT MACHINE
*I
      TST      (R1)+

*CL
FILE:
```

REMOTE/RT-11 V01
REMUSR.MAC

Seq 13*
1 of 1

EDIT ERRORS OCCUR WHEN THE FIRST CHARACTER IN THE TEXT BUFFER IS A
LINE FEED (SPR 11-6336,11-6670 JD)

PROBLEM: When the following conditions are met simultaneously,
EDIT may lose characters:

1. The first character in the text buffer is a <LF>, and
2. The pointer is within the first line of the buffer, and
3. A line-oriented command with a negative or zero argument is executed.

When the VT11 display is used, and a <LF> is the first character of
the text buffer, the pointer becomes maladjusted. This will
occasionally produce erroneous characters.

DISPOSITION: EDIT presupposes lines to be delimited by <CR><LF>
pairs. These problems will occur if a search for the pair is
unsuccessful, placing the pointer outside the text buffer. The
following patch will correct this:

Patch to EDIT V02-12.

R PATCH <CR>

PATCH V01-02

FILE NAME--

*EDIT.SAV <CR>

*2402;R

*0,13010\ *0,10330/ *0,10334/ 0,10336/ 0,10340/ 0,10342/ 0,10344/ 0,10346/ 0,10350/ 0,10352/ 0,10354/ 0,10356/ 0,10360/ 0,10362/ 0,10364/	40 5767 1003 12703 1 406 105767 167564 1420 16703 167614 5203 20467 167640 101411	101 <CR> 16703 <CR> 1403 <LF> 105767 <LF> 167572 <LF> 1425 <LF> 5203 <LF> 20467 <LF> 167652 <LF> 101414 <LF> 124427 <LF> 12 <LF> 1372 <LF> 20467 <LF> 167636 <LF>
---	---	---

EDIT ERRORS OCCUR WHEN THE FIRST CHARACTER IN THE TEXT BUFFER IS A
LINE FEED

0,10366/	124427	101406 <LF>
0,10370/	12	124427 <LF>
0,10372/	1372	15 <LF>
0,10374/	124427	1364 <LF>
0,10376/	15	5303 <LF>
0,10400/	1367	1362 <LF>
0,10402/	5303	122424 <LF>
0,10404/	1365	207 <LF>
0,10406/	122424	240 <LF>
0,10410/	207	240 <LF>
0,10412/	16703	240 <LF>
0,10414/	167554	240 <CR>>
*E		

The new EDIT version will be EDIT V02-12A.

CHARACTER IS LOST WHEN EXECUTING A READ COMMAND (SPR 11-8302 JD)

PROBLEM: When EDIT is reading a file which is exactly n blocks long, the last character is lost.

DISPOSITION: A check for the end of file is made before the last character is read into the buffer. If the end of the file is encountered, EDIT will immediately return to command mode without inputting that character. The following patch will correct this problem:

Patch to EDIT V02-12A.

```
.R PATCH <CR>

PATCH V01-02

FILE NAME--
*EDIT.SAV <CR>
*2402;R
*0,13010\      101      102 <CR>
*0,10630/      4242      4243 <CR>
*0,10666/      5242      5243 <CR>
*E
```

The new EDIT version will be EDIT V02-12B.

EXTRA TEXT APPENDED TO EDIT OUTPUT (SPR 11-8455, 11-8430 JD)

PROBLEM: EDIT does not properly reset its output buffer pointer after writing out a partially filled buffer.

DISPOSITION: The following patch to EDIT V02-12B will correct this problem.

.R PATCH <CR>

PATCH V01-02

FILE NAME--

*EDIT.SAV <CR>

*2402;R

*0,13010\ *0,14674/ 0,14676/ *0,14716/ 0,14720/ 0,14722/ 0,14724/ 0,14726/ 0,14730/ 0,14732/ 0,14734/ 0,14736/ *0,14742/ 0,14744/ 0,14746/ 0,14750/ 0,14752/ 0,14754/ 0,14756/ 0,14760/ 0,14762/ 0,14764/ 0,14766/ 0,14770/ 0,14772/ 0,14774/ 0,14776/ *0,15032/ 0,15034/ 0,15036/ *E	102 416 1026 5267 163430 110022 20227 6243 1006 12702 5242 12746 167 32 20227 7243 103407 12702 6242 4767 10 12767 5242 163360 207 12746 5242 12602 4767 173750	103 <CR> 414 <LF> 1025 <CR> 110022 <LF> 20227 <LF> 6242 <LF> 1007 <LF> 12702 <LF> 5242 <LF> 4767 <LF> 42 <LF> 12702 <CR> 411 <LF> 20227 <LF> 7242 <LF> 1010 <LF> 12702 <LF> 6242 <LF> 4767 <LF> 16 <LF> 12702 <LF> 5242 <LF> 4767 <LF> 174016 <LF> 10267 <LF> 163354 <LF> 207 <CR> 240 <LF> 240 <LF> 240 <CR>
---	--	--

The new EDIT version will be EDIT V02-12C.

RT-11, V02C-02
EDIT, V02-12B

Seq 7*
1 of 1

Patching LP Vector

PROBLEM:

In some installations the line printer may be installed at a vector and/or device address differing from the standard UNIBUS assignments. In this case, it is desirable to be able to patch the LP handler to function with the non-standard printer installation.

SOLUTION:

A simple patch is possible to the V02C LP handler. The locations to patch are:

Address	Contents	Normal Value
1000	vector	200
1052	status register	177514
1154	buffer register	177516

For example, to patch the handler for an LA35 installed at vector address 204, but using the standard UNIBUS printer address, the following procedure may be used:

```
.R PATCH

FILE NAME --
*LP.SYS <CR>
*1000/ 200      204 <CR>
*E
.R PIP <CR>
*SY:/O <CR>
```

The monitor must be rebooted after a handler is patched.

SOFTWARE PRODUCT RT-11		VERSION V02C-02	
COMPONENT HANDLER		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION LP.SYS		SEQUENCE 2*	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

LP HANGS SYSTEM (SPR 11-6879, 11-6881 JD)

PROBLEM: On recently shipped systems, RT-11 exhibits a problem with the line printer. If the line printer is off-line when a transfer is started or is taken off-line during a transfer, the system will hang, regardless of the SET LP NOHANG option. The problem does not occur if the printer goes off-line because of an error, such as a paper fault.

DISPOSITION: The problem is due to a change in the new printer controller module, M7258, for an LP05 printer. No interrupt is generated when an off-line condition is removed, unless an error condition (e.g., paper fault) was also present. This is known to affect the LP05 printer and may affect other printers.

We are currently addressing the problem. Until a solution is generated to correct the controller, the following procedures should be applied:

1. Make sure printer is on-line before a transfer is started, and avoid taking it off-line during a transfer.
2. If a system hang does occur, the following manual procedure should correct the situation without aborting the program.
 - a. Correct any printer problem and place printer on-line.
 - b. If printing does not resume, place the HALT/ENABLE switch in HALT position.
 - c. Load Address 777514.
 - d. Deposit 0
 - e. Load Address 777514
 - f. Deposit 100
 - g. Raise HALT/ENABLE switch to ENABLE position.
 - h. Press CONTINUE switch.

This procedure will generate a printer interrupt and should cause the transfer to proceed.

PERFORMANCE IMPROVEMENT IN LINKER

An enhancement can be made to the linker which will improve the execution time when linking with libraries. The most dramatic change in linking time will be noticed when using a slower CPU (e.g., 11/05 or 11/10). The following patch is to LINK V04-04.

```
.R PATCH <CR>

PATCH V01-02

FILE NAME--
*LINK.SAV <CR>
*616/ 1043 1077<CR>
*E
.R PATCH <CR>

PATCH V01-02

FILE NAME--
*LINK.SAV/O <CR>

?BOTTOM ADDR WRONG?
*500;B
*3242/ 12700 105767 <LF>
3244/ 1636 176531 <LF>
3246/ 10146 1404 <LF>
3250/ 10003 167 <LF>
3252/ 12702 6040 <LF>
3254/ 10 62704 <LF>
3256/ 112420 10 <LF>
3260/ 105767 11400 <LF>
3262/ 176513 1416 <LF>
3264/ 1412 21100 <LF>
3266/ 20405 1372 <LF>
3270/ 103410 26164 <LF>
3272/ 10046 2 <LF>
3274/ 105067 2 <LF>
3276/ 176475 1366 <LF>
3300/ 4767 12703 <LF>
3302/ 1054 1636 <LF>
3304/ 12600 10300 <LF>
3306/ 105167 12420 <LF>
```

3310/	176463	12420	<LF>
3312/	5302	12420	<LF>
3314/	1360	12420	<LF>
3316/	10300	207	<LF>
3320/	5202	261	<LF>
3322/	22123	207	<CR>
*7210;OR			
*2:0,2104/	XXX	12702	<LF>
2:0,2106/	XXX	4	<LF>
2:0,2110/	XXX	12703	<LF>
2:0,2112/	XXX	1636	<LF>
2:0,2114/	XXX	5714	<LF>
2:0,2116/	XXX	1424	<LF>
2:0,2120/	XXX	20405	<LF>
2:0,2122/	XXX	103406	<LF>
2:0,2124/	XXX	105067	<LF>
2:0,2126/	XXX	170435	<LF>
2:0,2130/	XXX	4767	<LF>
2:0,2132/	XXX	173014	<LF>
2:0,2134/	XXX	105167	<LF>
2:0,2136/	XXX	170425	<LF>
2:0,2140/	XXX	12423	<LF>
2:0,2142/	XXX	5302	<LF>
2:0,2144/	XXX	1365	<LF>
2:0,2146/	XXX	12703	<LF>
2:0,2150/	XXX	1636	<LF>
2:0,2152/	XXX	21113	<LF>
2:0,2154/	XXX	1353	<LF>
2:0,2156/	XXX	26163	<LF>
2:0,2160/	XXX	2	<LF>
2:0,2162/	XXX	2	<LF>
2:0,2164/	XXX	1347	<LF>
2:0,2166/	XXX	207	<LF>
2:0,2170/	XXX	261	<LF>
2:0,2172/	XXX	207	<CR>
*7216;OR			
*4:0,24\	40	101	<CR>

The new LINK version will be LINK V04-04A.

Note: The ?BOTOM ADDR WRONG? error is produced for the following reason. Patch expects LINK to be loaded at location 1000 (default), but is actually loaded starting in location 500. LINK's true base address must be set to its actual value (500;B).

?ERR 61 MESSAGE FROM PATCHO

PROBLEM: The version of PATCHO supplied with RT-11 V02C will generate an ?ERR 61 message after the user types a number in response to the ENTER CHECKSUM: message.

DISPOSITION: A way around the problem is to enter patches always including the DEC command, but this eliminates PATCHO checksum verification and could result in errors.

The following patch corrects this problem and allows checksums to be entered by the user for verification. The resulting version of PATCHO is V01-3A and will identify itself when the HELP command is used.

```
.R PATCH <CR>

PATCH V01-02

FILE NAME--
*PATCHO.SAV/O <CR>
*12606;0R
*23732;1R
*6032/ 50011 12666 <CR>
*3:1,0/ 30055 31455 <LF>
3:1,2/ 20063 20101 <CR>
*0,54/ 1434 434 <LF>
0,56/ 16402 0 <LF>
0,60/ 2 12601 <LF>
0,62/ 12705 12600 <LF>
0,64/ 400 12603 <LF>
0,66/ 5022 10146 <LF>
0,70/ 5305 5001 <LF>
0,72/ 1375 5300 <LF>
0,74/ 32714 100423 <LF>
0,76/ 400 112302 <LF>
0,100/ 1007 42702 <LF>
0,102/ 26464 177600 <LF>
0,104/ 26 162702 <LF>
0,106/ 32 60 <LF>
0,110/ 101016 103413 <LF>
0,112/ 4767 120227 <LF>
0,114/ 175226 7 <LF>
0,116/ 771 101010 <LF>
```

11 Digital Software News
May 1976

?ERR 61 MESSAGE FROM PATCHO

```
0,120/ 12763 6301 <LF>  
0,122/ 12746 103406 <LF>  
0,124/ 160 6301 <LF>  
0,126/ 10663 103404 <LF>  
0,130/ 144 6301 <LF>  
0,132/ 4767 103402 <LF>  
0,134/ 175206 60201 <LF>  
0,136/ 775 103355 <LF>  
0,140/ 12750 261 <LF>  
0,142/ 16304 40101 <LF>  
0,144/ 24 4136 <CR>  
*E
```

RT-11, V02C-02
PATCHO, V01-03

Seq 2
2 of 2

Error in F/B HRESET Code

PROBLEM:

A .HRESET done from a F/B job will cause channel 17 to appear to be closed.

SOLUTION:

The following patch corrects the problem.

```
.R PATCH <CR>  
  
FILE NAME --  
*MONITR.SYS/M <CR>  
*20000;OR  
*0,33620/      376      410 <CR>  
*324\  0      101 <CR>  
*E  
.R PIP <CR>  
*A=MONITR.SYS/U <CR>  
*SY:/O <CR>
```

The new version will be F/B V02C-02A.

SOFTWARE PRODUCT RT-11	VERSION V02C-02	
COMPONENT Monitor	VERSION V02C-02	
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE 13	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE March 1976

FIS Exception Error (SPR 11-6520 JD)

PROBLEM:

The S/J monitor does not correctly handle FIS exception errors.

SOLUTION:

The following patch applied to the V02C-02B S/J monitor will correct the problem.

```
.R PATCH <CR>

FILE NAME --
*MONITR.SYS/M <CR>
*17000;OR
*0,17510/      1001      1402 <LF>
0,17512/      2         240 <CR>
*324\  102      103 <CR>
*E
.R PIP <CR>
*A=MONITR.SYS/U <CR>
*SY:/O <CR>
```

The new version number will be SJ V02C-02C.

SOFTWARE PRODUCT RT-11		VERSION V02C-02	
COMPONENT Monitor		VERSION SJ V02C-02B	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 17*	PAGE 1 OF 1
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1976	

MIDNIGHT ROLLOVER FOR F/B MONITOR MALFUNCTIONS (SPR 11-6823 JD)

PROBLEM: There is an error in the code which supports midnight rollover in the F/B monitor.

DISPOSITION: This patch is mandatory for installations using 50 CPS power. This patch will not affect installations using 60 CPS power and should be installed to maintain the correct patch level.

For the F/B monitor, V02C-02A, use the following patch.

```
.R PATCH <CR>

PATCH V01-02

FILE NAME--
*MONITR.SYS/M <CR>
*20000;1R
*324\ 101      102 <CR>
*1,31356/     21027  21067 <LF>
1,31360/     15000  10 <CR>
*E
.R PIP <CR>
*SY:A=SY:MONITR.SYS/U <CR>
*SY:/O <CR>
```

The new patch level is V02C-02B.



Low Speed Reader Support

This RT-11 program allows the user to PIP a file from the low speed paper tape reader to a file-structured device.

First, use the editor to create the source file LSRPIP.MAC. Then, assemble and link the program as follows:

```
.R MACRO
*LSRPIP=LSRPIP
*↑C
.R LINK
*LSRPIP=LSRPIP
*↑C
.
```

Then, use the operating instructions given on the following pages.

SOFTWARE PRODUCT RT-11	VERSION V02C
COMPONENT SYSTEM INFORMATION	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE PAGE 1 1 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>
ORIGINAL DATE April 1974	

Low Speed Reader Support

;LSRPIP
;PROGRAM TO TRANSFER FILE FROM LOW SPEED PAPER
;TAPE READER TO RT-11 FILE NAMED IN COMMAND STRING.

;OPERATING INSTRUCTIONS

;CALL PROGRAM BY TYPING "R LSRPIP"
;RESPOND TO CSI "*" WITH NAME OF FILE TO
;BE CREATED, FOLLOWED BY AN "=" OR "<". LSRPIP
;WILL ASK YOU TO PREPARE THE TAPE, THEN STRIKE A
;KEY WHEN TAPE IS READY. TAPE WILL BE READ INTO
;THE FILE, AND AN "*" WILL APPEAR TO
;INDICATE READINESS FOR THE NEXT CYCLE.
;THE TAPE WILL PAUSE OCCASIONALLY DURING THE READ PROCESS, BUT THE
;OPERATION IS NOT COMPLETE UNTIL THE "*" FOR THE NEXT COMMAND IS
;PRINTED.

;THE KEYBOARD IS DISABLED DURING THE TAPE TRANSFER.
;TO ABORT AN UNDESIRABLE OPERATION, SET THE LOW SPEED READER CONTROL
;SWITCH TO "STOP", WHICH WILL TERMINATE THE READ AND RETURN WITH
;AN "*". A CTRL/C CAN THEN BE TYPED.

;ANY LEGAL OUTPUT FILE OR DEVICE MAY BE USED TO TRANSFER TO; AN
;ASCII TAPE MAY EVEN BE "LISTED" BY USING "TT;" AS THE OUTPUT FILE.

;SAMPLE USAGE: ;R LSRPIP<CR>
; ;TAPE1,BIN=<CR>
; PLACE TAPE IN READER, SET SWITCH TO START,
; THEN STRIKE ANY KEY TO BEGIN TRANSFER.
; ;TAPE2,BIN=<CR>
; PLACE TAPE.....(ETC)

;CREATE THE PROGRAM WITH EDIT, CALLING THE OUTPUT FILE LSRPIP.MAC
;ASSEMBLE IT WITH MACRO (*LSRPIP=LSRPIP), THEN LINK IT WITH LINK
;(*LSRPIP=LSRPIP). IT IS THEN READY TO GO.

SOFTWARE PRODUCT RT-11	VERSION V02C
COMPONENT SYSTEM INFORMATION	VERSION N/A
SUBPROGRAM OR ADDITIONAL INFORMATION	SEQUENCE PAGE 1 2 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>
	ORIGINAL DATE April 1974

Low Speed Reader Support

```

R1=X1
.MCALL .TTYIN,.WRITW,.PRINT,.CSIGEN,.CLOSE, ..V1..
..V1..
START: BIC      #10000,44      ;USE TELETYPE IN GENERAL MODE FOR CSI
        .CSIGEN #DEVSPC,#DEFEXT,#0      ;USE CSI TO GET AND OPEN
                                           ;OUTPUT FILE
        BIS      #10000,44      ;SET TELETYPE TO SPECIAL MODE FOR PROMPT
        .PRINT  #MSG           ;PRINT SETUP MESSAGE
        .TTYIN  ;WAIT FOR HIS KEYBOARD STROKE
        CLR     BLOCK         ;NEW FILE-ZERO BLOCK NUMBER
BUFCLR: MOV     #BUFFER,R1     ;POINT R1 TO BUFFER
CLRLP:  CLR     (R1)+         ;CLEAR THE BUFFER
        CMP     R1,#BUFEND    ;DONE?
        BLO    CLRLP         ;LOOP IF NOT
        MOV     #BUFFER,R1     ;YES-RESET R1 TO POINT TO BUFFER
TTINLP: MOVB   #1,177560      ;DISABLE TTY INT,SET READER RUN
WAIT:   TSTB   177560         ;BYTE IN YET?
        BMI    BYTEIN        ;BRANCH IF YES
        INC    COUNT         ;NO-BUMP TIMEOUT COUNTER
        BNE    WAIT          ;IF TIMEOUT NOT ZERO,LOOP
        .WRITW 0,#BUFFER,#400,BLOCK ;WE TIMED OUT-WRITE LAST BLOCK
        .CLOSE 0              ;CLOSE OUTPUT FILE
        MOV    #100,177560    ;RE-ENABLE KEYBOARD INTERRUPT
        BR     START         ;AND CYCLE

BYTEIN: CLR     COUNT         ;RESET TIMEOUT COUNTER
        MOVB   177562,(R1)+   ;PUT BYTE IN BUFFER
        CMP    R1,#BUFEND    ;BUFFER FULL?
        BLO    TTINLP        ;GO GET NEXT BYTE IF NOT
        .WRITW 0,#BUFFER,#400,BLOCK ;YES-WRITE IT OUT
        INC    BLOCK         ;BUMP BLOCK NUMBER
        BR     BUFCLR        ;AND ZERO BUFFER
  
```

SOFTWARE PRODUCT RT-11		VERSION V02C	
COMPONENT SYSTEM INFORMATION		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 1	PAGE 3 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1974	

Low Speed Reader Support

```

BUFFER: .*,+1000
BUFEND:
DEFEXT: 0
          0 ;DEFAULT EXTENSION BLOCK FOR CSIGEN
COUNT: 0 ;TIMEOUT COUNTER
BLOCK: 0 ;FILE BLOCK NUMBER
MSG: .ASCII /PLACE TAPE IN READER,SET SWITCH TO START,/
      .BYTE 15,12
      .ASCIZ /THEN STRIKE ANY KEY TO BEGIN TRANSFER,/
      .EVEN
DEVSPC:
          .END START
  
```

SOFTWARE PRODUCT RT-11		VERSION V02C	
COMPONENT SYSTEM INFORMATION		VERSION N/A	
SUBPROGRAM OR ADDITIONAL INFORMATION		SEQUENCE 1	PAGE 4 OF 4
NEW <input checked="" type="checkbox"/>	REPLACEMENT ARTICLE <input type="checkbox"/>	ORIGINAL DATE April 1974	

PROGRAM EXAMPLE ERROR IN AR11 USER'S GUIDE (SPR-11-6714)

PROBLEM: The program example on page 3-6 of the AR-11 User's Guide is in error. After the first time the clock overflow is tested, it does not get reset and therefore satisfies the test every time thereafter.

DISPOSITION: The clock overflow flag must be cleared after each clock overflow. This can be done one of two ways:

1. Insert a BIC #200,@CKSR before the BELL: TSTB @TPS instruction.
or
2. Change the BR LOOP to a BB AGAIN. The AGAIN is a label added to the MOV #413,@CKSR instruction. This will not cause any loss of clock ticks.

The next printing of the manual will correct this programming example as such:

Location	Instruction	Comment
INIT:	MOV #70,@CKBR MOV #40,@ADSR	;Set to -200 ₁₀ ;Enable overflow to start A/D ;conversion
AGAIN:	MOV #413,@CKSR	;Start count at 100 Hz with mode 1
LOOP:	TSTB @CKSR BPL LOOP	;Check for Overflow flag ;No, try again
BELL:	TSTB @TPS BPL BELL MOV #7,@TPB	;Test for ready on TTY ;No, try again ;Ring the bell
CONV:	TSTB @ADSR BPL CONV JSR PC, READ BR AGAIN	;Check for Done flag ;No, try again ;Go to subroutine which reads ;results of A/D conversion ;Do it again
CKBR:	770406	;Buffer/Preset register address
CKSR:	770404	;Clock Status register address
ADSR:	770400	;A/D Status register address
TPS:	777564	;TTY Status register
TPB:	777566	;TTY Buffer register

REPLACEMENT PAGES

The following pages are page replaceable updates to the RT-11 System Generation Manual for V02C.

In all cases, these pages replace the existing pages in this manual or in any addendum to it. Change bars indicate where changes were made.

SYSTEM BUILD INSTRUCTIONS

6. Type: DKI*.*=RKI*,*/X<CR>
 Response: ?NO ,SYS/,BAD ACTION?
 *

 Type: DKI/K<CR>
 Response: *

Each block on the disk being built will be checked for errors; the process takes about 4 minutes on an RK11, 1 minute on a single platter RF11 or RJS03/4, or about 30 minutes on an RP02.

NOTE

If the response is anything but the above, see the RT-11 System Reference Manual, Section 4.2.12.

 Type: DKIA=DKIMONITR,SYS/U<CR>
 Response: *

7. Remove the master disk from Unit 0 and store it in a safe place. If the new system is an RK system, remove the copy from Unit 1, label it RT-11 V02C SYSTEM DISK, then mount it in Unit 0, WRITE-ENABLED.

If the system has a hardware bootstrap capable of bootstrapping the new system disk, boot the disk; otherwise, perform the bootstrapping procedures in the appropriate section of Appendix A.

RT-11 should bootstrap from the new system disk and type its identifying message,

RT-11SJ V02C-xx
.

If it does not, repeat this section.

8. Proceed to Section 2.2.2 if building a FORTRAN system from DECpack, Section 2.2.3 if building a BASIC system from DECpack, or consult the Table of Contents for the appropriate section if building FORTRAN or BASIC from another media. If neither FORTRAN nor BASIC will be used on the system, go to Chapter 3.

2.2.2 FORTTRAN IV

This section contains instructions for those who received RT-11 FORTRAN on DECpack disk. The instructions involve the transfer of the necessary FORTRAN-related files to the system disk and the creation of the FORTRAN library.

1. Bootstrap an RT-11 V02C system (if you have completed Section 2.2.1, the system is already booted and running).

To the running RT-11 monitor,

SYSTEM BUILD INSTRUCTIONS

Type: DATE dd-mmm-yy<CR>

where dd-mmm-yy is the current date in the form 27-NOV-75.

Response: .

Type: TIME hh:mm:ss<CR>

where hh:mm:ss is the current 24-hour time in hours:minutes:seconds.

Response: .

2. If an RK11 system is being built, mount the RT-11 FORTRAN IV system disk on Unit 1, WRITE-PROTECTED.

Type: ASSIGN RK11DIS<CR>

Response: .

Go to Step 3.

If an RF11, RSJ03/4, or RP02 system is being built, mount the RT-11 FORTRAN IV system disk on Unit 0, WRITE-PROTECTED.

Type: ASSIGN RK01DIS<CR>

Response: .

3. Type: R PIP<CR>
Response: *

Type: *,*=DIS!*,*/X<CR>

Response: *

Dismount the RT-11 FORTRAN master disk and store it in a safe place.

Type: CTRL C

RESPONSE: ^C

Type: R LIBR<CR>

Response: *

4. If the FORTRAN system will be running on a configuration that includes an EAE, proceed. Otherwise, go to Step 5.

Type: FORLIB=UNI,OTSCOM,V2NS,EAE/G<CR>

Response: ENTRY POINT!

Type: SERRTB<CR>

SERRS<CR>

<CR>

Response: SEND2 ILL INS
SEND2 ILL INS
SF102 ILL INS
SCL02 ILL INS
SF102 ILL INS
SF102 ILL INS

*

SYSTEM BUILD INSTRUCTIONS

Type: CTRL C
Response: ␣
.

If the system has a hardware bootstrap capable of bootstrapping the new system DEctape, bootstrap the DEctape; otherwise, perform the bootstrapping procedures in Section A.2.

1. Bootstrap an RT-11 V02C system. Ensure that the system device is WRITE-ENABLED.

Type: DATE dd-mmm-yy<CR>

where dd-mmm-yy is the current date in the form 27-NOV-75.

Response: .

2. Mount the BASIC/RT-11 System DEctape 1 (DEC-11-LBACA-C-UC1) on Unit 1, WRITE LOCKed.

3. Type: R PIP<CR>
Response: *

Type: *,*DT1:BASIC,SAV,BAS8K,SAV,DEMO,BAS/X<CR>
Response: *

Dismount the master DEctape and store it in a safe place.

Type: CTRL C
Response: ␣
.

Proceed to Chapter 3.

2.4 BUILDING AND STARTING ON DISKETTE

2.4.1 RT-11

This section contains instructions for those who received RT-11 on diskette. If diskette is to be the system device, the user is instructed how to copy the master diskette. If another disk (RK11, RF11, or RP02) is to be the system device, the user is instructed how to make the larger disk system from the master diskette. It is important (for the user's protection) to build the system as instructed, then store the master in a safe place.

1. Mount the RT-11 master diskette (Disk 1 of 2, DEC-11-ORTSA-E-YC1) in the left-hand drive (Unit 0). Set the ENABLE/HALT switch to HALT to stop any previous program that may be running. Set the ENABLE/HALT switch to ENABLE.
2. If the system has a hardware bootstrap capable of bootstrapping the diskette, boot the diskette and proceed to Step 3; otherwise, see Section A.3, then proceed to Step 3.
3. There is a slight pause after which the following message is displayed on the terminal:

SYSTEM BUILD INSTRUCTIONS

RT-11SJ V02C-xx

If not, repeat this section from Step 2.

Type: DATE dd-mmm-yy<CR>

where dd-mmm-yy is the current date in the form 27-NOV-75.

Users of diskette as the system device, proceed to Step 4.
Users of other disks as the system device, proceed to Step 6.

4. Type: R PIP<CR>
Response: *

Mount a blank diskette in Unit 1.

Type: DX11/Z<CR>
Response: DX11 /Z ARE YOU SURE?

Type: Y<CR>
Response: *

Type: DX11*,*=DX01*,*/X/Y<CR>

(The system will be copied onto the blank diskette. The process takes about one minute.)

Response: *
Type: DX11A=DX11MONITR;SYS/U<CR>

(The system bootstrap will be copied onto the diskette on Unit 1.)

Response: *

Type: DX11/K<CR>
Response: *

Each block of the diskette being built will be checked for errors. This process takes about 30 seconds.

NOTE

If the response is anything but the above, see the RT-11 System Reference Manual, Section 4.2.12.

Remove the master RT-11 System Disk and store in a safe place. Dismount the new disk from Unit 1, label it as RT-11 SYSTEM DISK 1.

5. Mount the RT-11 System Disk (Disk 2 of 2, DEC-11-ORTSA-E-YC2) on Unit 0. Mount a blank diskette in Unit 1.

Type: DX11/Z<CR>
Response: DX11 /Z ARE YOU SURE?

Type: Y<CR>
Response: *

SYSTEM BUILD INSTRUCTIONS

Response: \$END2 ILL INS
\$END2 ILL INS
\$F102 ILL INS
\$CLO2 ILL INS
\$F102 ILL INS
\$F102 ILL INS

*

Go to Step 7.

6. If the configuration contains none of the above options:

Type: SYIFORLIB[140]=SYIUNI,OTSCOM,V2NS,NHD/G<CR>
Response: ENTRY POINTI

Type: \$ERRTB<CR>
\$ERRS<CR>
<CR>

Response: \$END2 ILL INS
\$END2 ILL INS
\$F102 ILL INS
\$CLO2 ILL INS
\$F102 ILL INS
\$F102 ILL INS

*

Type: SYIFORLIB,V2S[140]=SYIUNI,OTSCOM,V2S,NHD/G<CR>
Response: ENTRY POINTI

Type: \$ERRTB<CR>
\$ERRS<CR>
<CR>

Response: \$END2 ILL INS
\$END2 ILL INS
\$F102 ILL INS
\$CLO2 ILL INS
\$F102 ILL INS
\$F102 ILL INS

*

7. Type: CTRL C
Response: ^C

,

Proceed to Section 2.5.3 if building a BASIC system from magtape or consult the Table of Contents for the appropriate section if building BASIC from another media. If BASIC will not be used on the system, go to Chapter 3.

2.5.3 BASIC/RT-11

This section contains instructions for those who received BASIC/RT-11 on magtape. The instructions involve placing running versions of BASIC on the system device.

SYSTEM BUILD INSTRUCTIONS

The system device onto which BASIC is being built must have at least 80 free blocks. The monitor, the appropriate magtape handler (MM.SYS or MT.SYS), and PIP.SAV are the only system components necessary on this system device.

1. Bootstrap an RT-11 V02C system (if you have completed Section 2.5.1 or 2.5.2, the system is already booted and running).
2. Mount the BASIC/RT-11 master magtape (DEC-11-LBACA-C-MC7 for 7-track tape or DEC-11-LBACA-C-MC9 for 9-track tape) on Unit 0.

Type: DATE dd-mmm-yy<CR>

where dd-mmm-yy is the current date in the form 27-NOV-75.

Response: .

3. Type: R PIP<CR>
Response: *

If the master magtape is on MT11,

Type: *,*=MT0|BASIC,SAV,BAS8K,SAV,DEMO,BAS/X/M,1000<CR>

If the master magtape is on TJU16,

Type: *,*=MM0|BASIC,SAV,BAS8K,SAV,DEMO,BAS/X/M,1000<CR>

In either case,

Response: *

Dismount the master magtape and store it in a safe place.

Type: CTRL C

Response: ^C

.

Proceed to Chapter 3.

2.6 BUILDING AND STARTING FROM CASSETTE

2.6.1 RT-11

This section contains instructions for those who received RT-11 on cassettes. The instructions describe the building of an RK11 disk system from cassette. RK11 users will need a blank, formatted disk for this procedure. See Appendix B for formatting procedures.

Prior to inserting any master cassette during this procedure, WRITE PROTECT the data by placing the orange tabs so that the holes are uncovered. Cassette Unit 0 is on the left and cassette Unit 1 is on the right.

1. Set the ENABLE/HALT switch to HALT to stop any previous program that may be running. Set the ENABLE/HALT switch to ENABLE.

.R PATCH<CR>

PATCH Version number

FILE NAME--

```
*MONITR.SYS/M<CR>
*56\0_____15<LF>      (Fill after <CR>
57\0_____4<CR>      with 4 nulls)
*E
```

Once the change is made, the bootstrap recopied with the PIP /U switch, and the monitor rebooted with the PIP /O switch, all programs that use the monitor for console I/O will operate correctly.

4.6.2 Magtape Parity, Density, Number of Tracks

4.6.2.1 TM11 (MT.SYS) -- The RT-11 TM11 magtape handler is distributed such that it will correctly handle both 7- and 9-track drives without modification. It does so at 800 bpi, using the TM11 dump mode for 7-track drives. Seven-track drives can also be written (in hardware mode only; see the RT-11 System Reference Manual, Appendix H) at 200, 556, and 800 bpi (non-dump mode) by modifying the handler as described below.

To alter the magtape density used by the handler, the following patches must be made:

1. Patch the handler density word (location PARDEN in MT.SYS).
2. When changing from 800 bpi 9- or 7-track dump mode to 200, 556, or 800 bpi 7-track, you must patch the handler to default to hardware mode (location HW in MT.SYS) and patch the \$STAT entry in the monitor for MT.SYS (location MTSTAT in MONITR.SYS) so that the special device bit (bit 12) is off (zero).
3. When changing to 800 bpi 9- or 7-track dump mode from 200, 556, or 800 bpi 7-track, you must patch the handler to default to software mode and patch the \$STAT entry in the monitor for MT.SYS so that the special device bit is on (one).

NOTE

In the 200, 556, and 800 bpi 7-track modes, with the \$STAT table patched so that bit 12 is off, the MT handler will not get control when a .LOOKUP, .ENTER, .CLOSE, or .DELETE is issued for MT. Doing a .LOOKUP, .ENTER, or .CLOSE in this state simply opens or closes a channel associated with MT. The magtape is not rewound for any of these functions in this state; the only operations that are passed to the magtape handler are .READ, .WRITE, and special functions (.SPFUN).

RT-11 SYSTEM CUSTOMIZATION

The following table describes the patches for the various magtape densities:

<u>Location</u>	<u>200 bpi 7-track</u>	<u>556 bpi 7-track</u>	<u>800 bpi 7-track</u>	<u>7-track dump mode (800 bpi) or 800 bpi 9-track</u>
PARDEN (MT.SYS)	0	20000	40000	60000
HW (MT.SYS)	377	377	377	0
MTSTAT (MONITR.SYS)	2011	2011	2011	12011

For example, to cause the MT handler to write 7-track tapes at 556 bpi:

```

.R PATCH<CR>

PATCH Version number

FILE NAME--
*MT.SYS<CR>
*PARDEN/60000 20000<CR>
*HW /00000 377<CR>
*F

FILE NAME--
*MONITR.SYS/M<CR>
*MTSTAT/12011 2011<CR>
*E

.R PIP<CR>
*SY:A=MONITR.SYS/U<CR>
*SY:/O<CR>

```

NOTE

See RT-11 System Release Notes, Table 2, for the exact addresses of PARDEN, HW, and MTSTAT.

4.6.2.2 TJU16 (MM.SYS) -- The TJU16 allows five possible tape modes in file-structured operation. Since the user may wish to transfer between tapes written in various modes, the TJU16 handler includes a table that defines the default mode for each magtape unit (MM0 to MM7). The user can change the default mode for a particular unit simply by patching that unit's entry in the mode table. The table is located at UNIMOD in the magtape handler and is eight words long. Each word contains the mode for a given unit (0-7).

The default mode for a particular unit may also be changed dynamically under program control. A nonfile-structured LOOKUP with a file count between 1 and 5 causes a mode change for the unit accessed. The following list shows the possible modes and corresponding table value for patching and file count for dynamic modification:

<u>Mode</u>	<u>Table Value</u>	<u>File Count</u>
200 bpi	0	1
556 bpi	400	2
800 bpi (odd parity)	1000	3
800 bpi (even parity)	1400	4
1600 bpi (phase-encoded)	2000	5

RT-11 SYSTEM CUSTOMIZATION

```
.R PIP<CR>
*SY:A=MONITR.SYS/U<CR>
*SY:/O<CR>
```

4.6.5 Interfacing RJS03/4 DISKS TO RT-11

RT-11 is distributed with the monitor device tables for RJS03/4 disk initialized for RJS03. To allow complete use of all the space available on an RJS04 disk, modify the device size table in the monitor as follows:

<u>Location</u>	<u>RJS03 Value</u>	<u>RJS04 Value</u>
DSSIZ	2000	4000

For example, to modify the RT11 DS monitor to support an RJS04 disk rather than RJS03, type:

```
.R PATCH
PATCH Version number
FILE NAME--
*MONITR.SYS/M<CR>
*DSSIZ/2000 4000<CR>
*E
.R PIP<CR>
*SY:A=MONITR.SYS/U<CR>
*SY:/O<CR>
```

NOTE

See RT-11 System Release Notes, Table 2, for the exact address of DSSIZ in the current monitor.

Once the above change has been made, zeroing the disk (using the PIP /Z switch) will adjust the directory to the appropriate size. If the system is already running from fixed-head disk as the system device and the disk cannot be zeroed without destroying the system, compressing the disk (with the PIP /S switch) will automatically re-adjust the directory size.

4.6.6 Interfacing RP03 Disks to RT-11

The RP02 support provided in the distribution kit is initialized for RP02 only. The RT-11 file structure can accommodate a maximum of 64000(10) blocks. The 40000(10)-block RP02 cartridge, therefore, can be accommodated as a single logical unit, while an RP03 cannot.

The RT-11 RP02 support can easily be altered, however to accommodate RP03s as follows:

RT-11 SYSTEM CUSTOMIZATION

1. Each RP03 drive must be considered as two logical units of 40000 blocks each; in essence, a single RP03 drive looks like two RP02 drives to the system. The cartridge on physical unit n is accessed as logical DPn and DPn+4; thus, drive 0 is referenced as DP0: and DP4:, drive 1 is DP1: and DP5:, etc. Note that although an RP03 is physically one device, it is two separate devices to the system. Each logical unit has its own complete directory and data space.
2. The DP.SYS handler must be patched to change location RP23 from 404 (octal) to 1404 (octal).
3. The DP monitors must be patched to alter location RP23 from 404 (octal) to 1404 (octal).

For example, to allow RT-11 to support RP03s:

```
.R PATCH<CR>
PATCH Version number
FILE NAME--
*DP.SYS<CR>
*RP23/ 404 1404<CR>
*_E
.R PATCH<CR>
PATCH Version number
FILE NAME--
*DPMNFB.SYS/M <CR>
*RP23/ 404 1404<CR>
*_E
.R PATCH<CR>
PATCH Version number
FILE NAME--
*DPMNSJ.SYS/M<CR>
*RP23/ 404 1404<CR>
*_E
.R PIP<CR>
*_SY:/O<CR>
```

NOTE

See RT-11 System Release Notes, Table 2, for the actual location of RP23 in each monitor and in DP.SYS.

Note that a maximum of four RP03s can be supported on the system. RP02s and RP03s can be mixed as long as the total number of units (physical drives) on the system does not exceed four. If the system contains only RP02s, the above changes must not be made and the system can support as many as eight units.

4.6.15 Reassigning Device Names for RK11 and RF11

Users of other DIGITAL operating systems will notice that RT-11 uses the controller names (RK and RF) rather than the more common user-level names (DK and DF) for these devices. This is due to the fact that RT-11 uses the name DK to refer to the default storage device, which may not necessarily be the RK11.

If you find this situation annoying, the device names can be reassigned with the monitor ASSIGN command, as follows:

```

.ASSIGN RK|DK<CR>
.ASSIGN RF|DF<CR>

```

Note, however, that when DK is reassigned in this manner, all default storage goes to the device name DK, and you may not wish to use the physical device RK as the default storage device.

4.6.16 Interfacing a Foreground Terminal

Applications for the F/B Monitor frequently require that the foreground program dialogue appear on a separate terminal, independent of the console terminal.

To facilitate development of these applications, or any others requiring multiple terminals, a source (KB.MAC) for a device-independent terminal handler has been included in the distribution kit. The source is provided for user convenience and can be used as a model for handler development, modified to meet specific needs, or assembled and used as is to provide support for a second terminal.

Documentation for the use of KB.MAC is contained in the comments at the beginning of the source; a listing appears in Appendix B of the RT-11 Software Support Manual.

4.6.17 Modifying the Line Count in MACRO and CREF

RT-11 MACRO and CREF set the number of lines printed per listing page at 60. This line count is satisfactory for applications with line printers that use paper 10.5 inches long. Applications that use paper of a different size (e.g., 8.5 inches long) and applications without line printers should modify MACRO and CREF as follows.

For MACRO:

```

.R PATCH<CR>

PATCH Version number

FILE NAME--
*MACRO.SAV<CR>
*MACR1;0R<CR>
*0,12366/____ 74 ____ n<CR>
*E

.R PIP<CR>
*SY:/O<CR>

```

where n is the new line count specified in octal.

For CREF:

```
.R PATCH<CR>
PATCH Version number
FILE NAME--
*CREF.SAV<CR>
*CREFL;0R<CR>
*0,3122/____ 74 ____ n<CR>
*E

.R PIP<CR>
*SY:/O<CR>
```

where n is the new line count specified in octal.

NOTE

See RT-11 System Release Notes, Table 2, for the actual addresses of MACR1 and CREFL.

4.6.18 Changing the DUMP Default Output Device

The DUMP utility program uses LP as its default output device. Systems that do not have line printers will want to change the default device, normally to TT.

DUMP.SAV can be easily altered to change the default output device as follows:

1. Patch to change location LP to the .RAD50 code for the new default output device.
2. Patch to change location MSGO+1 to the .ASCII code for the first letter of the new default output device.
3. Patch to change location MSGO+2 to the .ASCII code for the second letter of the new default output device.

For example, to change the DUMP default output device to TT (the console terminal),

```
.R PATCH<CR>
PATCH Version number
FILE NAME--
*DUMP.SAV<CR>
*LP/____ 46600 ____ 10040<CR>      [.RAD50 for TT]
*MSGO+1\114 ____ 124<CR>          [.ASCII for T]
*MSGO+2\120 ____ 124<CR>          [.ASCII for 1]
*E

.R PIP<CR>
*SY:/O<CR>
```

REPLACEMENT PAGES

The following pages are page replaceable updates to the RT-11 System Message Manual for V02C.

In all cases, these pages replace existing pages in this manual or in any addendum to it. Change bars indicate where changes were made.

BAD VARIABLE

BATCH F The variable specified is not one of the characters A-Z. Enter the variable as an alphabetic character.

BAD VID

BATCH F The volume identification specified in a \$MOUNT command was not in the correct form. Ensure that the equal sign and the name of the volume are included in the command.

BATCH FATAL ERROR

BATCH F A nonrecoverable error has occurred. This may indicate a problem in the software. Check the procedures for hard error conditions listed in Section 1.0 at the beginning of this manual. The system may have to be rebootsrapped. If the problem continues to occur, report the problem to Software Communications using an SPR (Software Performance Report). Include the contents of memory location 6012 (octal), along with a program listing and a machine readable source program, if possible.

BATCH HANDLER NOT RESIDENT

BATCH F The BATCH run-time handler was not loaded with the RT-11 LOAD command. LOAD the BATCH handler before attempting to run BATCH.

BATCH STACK OVERFLOW

BATCH F There were too many nested \$CALL commands in the BATCH stream. Ensure that there are no more than 31 nested \$CALL commands in the stream.

BC

BATCH F A Bad Code was found in the control file by the BATCH handler. This can happen when a .CTL file has been garbled or if an editing mistake was made by the programmer when altering or creating the file with EDIT.

?BDR AT LINE ccccccccc

BASIC F Item input from a DATA statement list by a READ statement was bad.

BEcccccccc

ODT IN A bad breakpoint entry occurred from location xxxxxx. This may indicate that a breakpoint routine was entered for no known breakpoint, that an illegal trace trap instruction occurred, that the T-bit was set in the status register, or that a jump to the middle of ODT occurred.

?B-I/O ERROR

MBOOT F An I/O error occurred during system MONITOR F (or magtape) boot.

?B-NO BOOT ON VOLUME

MONITOR F No bootstrap was written on the volume.

?B-NO MONITOR.SYS

MONITOR F No monitor existed on the volume being booted.

The name of the monitor being booted must be MONITR.SYS. Verify that it is correctly named. If MONITR.SYS has somehow been deleted from the system device, use a backup copy of the system device to restore the missing monitor file.

REPLACEMENT PAGES

The following pages are page replaceable updates to the RT-11 System Reference Manual for V02C.

In all cases, these pages replace existing pages in this manual or in any addendum to it. Change bars indicate where changes were made.

PREFACE

This manual describes the use of the RT-11 Operating System. It assumes the reader is familiar with computer software fundamentals and has had some exposure to assembly language programs. The section "Additional and Reference Material" later in this Preface lists documents that may prove helpful in reviewing those areas. The Glossary provides definitions of technical terms used in the manual.

The user who is unfamiliar with RT-11 should first read those chapters of interest (see "Chapter Summary" below) to become familiar with system conventions. Having gained familiarity with RT-11, the user can then reread the manual for specific information.

Chapter Summary

Chapter 1 discusses system hardware and software requirements. It describes general system operations and lists specific components available under RT-11.

Chapter 2 introduces the user to system conventions and monitor/memory layout. It describes in detail the keyboard commands for controlling jobs and implementing user programs.

Chapters 3 through 8 describe the system utility programs EDIT, PIP, MACRO, LINK, LIBR, and ODT, respectively. These programs (a text editor, file transfer program, assembler, linker, librarian, and debugging program) aid the user in creating text files and producing assembly-language programs.

Chapter 9, which describes programmed requests, is of particular interest to the experienced programmer. It describes call sequences that allow the user to access system monitor services from within assembly-language programs.

Chapters 10 and 11 describe the 8K Assembler and EXPAND programs, respectively. These programs are useful in RT-11 installations with minimum memory configurations.

Chapter 12 describes the BATCH command language for RT-11. In BATCH mode, the RT-11 system can be left to run unattended for long periods of time.

The appendixes summarize the contents of the manual and describe additional system utility programs that can be used for extended system operations. These programs include SRCCOM (a source file comparison program); FILEX (a file translation program that allows

Preface

transfer of files between RT-11 and other DIGITAL operating systems); PATCH and PATCHO (patching programs); DUMP (a file dump program); and SYSLIB (a library of programmed requests for FORTRAN users).

Version History

The current RT-11 system (monitor) is Version 2C (V2C). Each system component (monitors and utilities) is assigned a software identification number in the form Vxx-xx. Current identification numbers for V2C are listed in the RT-11 System Release Notes (DEC-11-ORNRA-A-D). To determine whether the correct version of a component is in use, examine its identification number and compare it with the list. (The procedure for examining the version number varies. Most system programs provide a special command; others print the version number when an output listing is requested. Consult the appropriate chapter or appendix of this manual for each component.)

NOTE

Throughout this manual, any references to V2 or V2B of RT-11 will pertain also to V2C. The RT-11 System Release Notes contain a comprehensive list of differences between V2C and previous versions of RT-11 (V2B, V2, V1).

Change bars and asterisks in the outermost margins of the manual are used to denote changes made to the text since the Version 2 release (DEC-11-ORUGA-B-D). The date July 1975 in the lower outside corner of a page indicates that the page was changed as a result of a release-independent update that occurred in July, 1975. The date January 1976 in the lower outside corner of the page indicates that the page was changed specifically as a result of the V2C update.

The user who is already familiar with the Version 2B RT-11 System Reference Manual (DEC-11-ORUGA-C-D, DN1) should first read the RT-11 System Release Notes document to note the major differences between V2B and V2C, and then read those pages of the RT-11 System Reference Manual that have changed as a result of the V2C update (identified by the date January 1976). The RT-11 System Generation Manual (DEC-11-ORGMA-A-D) should also be read if customization for special devices and features is required.

The user who is familiar with only the Version 2 RT-11 System Reference Manual (DEC-11-ORUGA-B-D) should read the following in addition to those items mentioned in the preceding paragraph:

Chapter 2	(System Communication)	- Tables 2-2, 2-3, and 2-5
Chapter 3	(Text Editor)	- Section 3.6.5.6
Chapter 9	(Programmed Requests)	- Sections 9.1 and 9.1.3.6
Chapter 12	(BATCHE)	- Entire Chapter
Appendix H	(F/B Programming And Device Handlers)	- Sections H.2.4 and H.2.5
Appendix O	(SYSLIB)	- Entire Appendix

Finally, the user familiar with only the Version 1 RT-11 System Reference Manual (DEC-11-ORUGA-A-D) should read this entire manual with these exceptions:

Table 2-5 (Cont.)
SET Command Options

Device	Option	Alteration																																																
CR	IMAGE	<p>Causes each card column to be stored as a 12-bit binary number, one column per word. The CODE option has no effect in IMAGE mode. The format of the 12-bit binary number is:</p> <p>PDP-11 WORD</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> <tr> <td>UNUSED (ALWAYS 0)</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td><td>ZONE</td> </tr> <tr> <td></td><td>12</td><td>11</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td></td><td></td><td></td> </tr> </table> <p>This format allows binary card images to be read and is especially useful if a special encoding of punch combinations is to be used. Mark-sense cards may be read in IMAGE mode.</p>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	UNUSED (ALWAYS 0)	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE		12	11	0	1	2	3	4	5	6	7	8	9			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																			
UNUSED (ALWAYS 0)	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE																																			
	12	11	0	1	2	3	4	5	6	7	8	9																																						
CR	NOIMAGE	<p>Allows the normal translation (as specified by the CODE option) to take place; data is packed one column per byte. Invalid punch combinations are translated into the error character, ASCII "\" (backslash), which is octal code 134. This is the normal mode.</p>																																																
CR	TRIM	<p>Causes trailing blanks to be removed from each card read. It is not recommended that TRIM and NOCRLF be used together since card boundaries will be difficult to find. This is the normal mode.</p>																																																
CR	NOTRIM	<p>Transfers a full 80 characters per card.</p>																																																
CT	RAW	<p>Causes the cassette handler to perform a read-after-write check for every record written, and retry if an output error occurred. If three retries fail, an output error is detected.</p>																																																
CT	NORAW	<p>Causes the cassette handler to write every record directly without reading it back for verification. This significantly increases transfer rates at the risk of increased error rates. Normal mode is NORAW.</p>																																																
<p>The following options, with the exception of HOLD/NOHOLD and COPY/NOCOPY, are available in the Foreground/Background System only; HOLD/NOHOLD and COPY/NOCOPY are available in both systems. These options are not permanent, and must be reissued whenever the monitor is re-bootstrapped. They can be made permanent by modifying the monitor as described in Chapter 2 of the <u>RT-11 Software Support Manual</u>. (Note that the device specification is TTY, not TT, because the handler itself is not changed.)</p>																																																		
TTY	COPY	<p>Enables use of the auto-print mode of the VT50 copier option, if present. The command is a no-op for any terminal other than the VT50, but a "]" character may be printed on the terminal. Consult the <u>VT50 Video Terminal Programmer's Manual</u> for more information.</p>																																																
TTY	NOCOPY	<p>Disables use of the auto-print mode of the VT50 copier option, if present. The command is a no-op for any terminal other than the VT50, but a "^" character may be printed on the terminal. This is the normal mode.</p>																																																

Table 2-5 (Cont.)
SET Command Options

Device	Option	Alteration
TTY	CRLF	Causes the monitor to issue a carriage return/line feed on the console terminal whenever it attempts to type past the right margin (as set by the WIDTH option). This is the normal mode.
TTY	NOCRLF	Causes no special action to be taken at the right margin.
TTY	FB	Causes the monitor to treat CTRL B and CTRL F characters as background and foreground program control characters and does not transmit them to the user program. This is the normal mode.
TTY	NOFB	Causes CTRL B and CTRL F to have no special meaning.
		NOTE
		SET TTY NOFB is issued to KMON, (which runs as a background job) and disables all communication with the foreground job. To enable communication with the foreground job, issue the command SET TTY FB.
TTY	FORM	Indicates that the console terminal is capable of executing hardware form feeds.
TTY	NOFORM	Causes the monitor to simulate form feeds by typing eight line feeds. This is the normal mode.
TTY	HOLD	Enables use of the hold screen mode of operation for the VT50 terminal. The command is a no-op for any terminal other than the VT50, but a "[" character may be printed on on the terminal. The command is valid for F/B and Single-Job Monitors. Consult the <u>DECscope User's Manual</u> for more information.
TTY	NOHOLD	Disables use of the hold screen mode of operation for the VT50 terminal. The command is a no-op for any terminal other than the VT50, but a "\" character may be printed on the terminal. This is the normal mode.
TTY	PAGE	Causes the monitor to treat CTRL S and CTRL Q characters as terminal output hold and unhold flags, and does not transmit them to the user program. This is the normal mode.

(continued on next page)

Peripheral Interchange Program

```

< UNUSED >      2
TRIG .OBJ      2  6-SEP-74

STP .OBJ      2  6-SEP-74
BAC .OBJ      2  6-SEP-74
< UNUSED >     20
.
.
LIBR1 .OBJ    137  6-SEP-74
DIRECT      1  9-SEP-74
< UNUSED >   230
254 FILES, 4280 BLOCKS
498 FREE BLOCKS

```

*LP:=CT1:/E

```

11-SEP-74
A .MAC      0 11-SEP-74
A .MAC      0 11-SEP-74
B .MAC      0 11-SEP-74
3 FILES, 0 BLOCKS

```

Outputs to the line printer a complete directory of cassette drive 1. 0's represent segment numbers.

The /F switch lists only filenames, omitting the file lengths and associated dates.

Examples:

```

*DT0:/F
TRACE .MAC
CARGO .REL
BMAP .OBJ
AAA

```

Outputs a filename directory of the device DT0 to the terminal.

*LP:=CT1:/F

```

A .MAC
A .MAC
B .MAC

```

Outputs a filename directory of the device CT1 to the line printer.

The /L, /E and /F commands have no effect on the files of the specified device. If the /W switch is used in conjunction with the /L or /E switches, the absolute starting block of the file and extra words (in octal) will be included in the listing (for all but cassette and magtape). For example:

```

*RK1:/L/W
10-SEP-74
DSQRT .OBJ      1 10-SEP-74      16      0
MAIN .OBJ       1 10-SEP-74      17      0
BASICR.OBJ     11 10-SEP-74      20      0
OTSV2 .OBJ      3 10-SEP-74      33      0

```

The first three columns indicate the filename and extension, block length, and date. The fourth column shows the absolute starting block (in octal), and the fifth column shows the contents of each extra word per directory entry (in octal). (This is allocated using the /Z:n switch; see Section 4.2.7.)

Peripheral Interchange Program

Using the /L, /E, or /F switch in conjunction with a device and filename causes the filename, and optionally the date and file length, to be output rather than a directory of the entire device. For example:

```
*F1.SAV/L
```

causes:

```
4-JUN-74
F1 SAV 18 4-JUN-74
3710 FREE BLOCKS
*
```

to be output, providing the file exists on device DK.

Directories are made up of segments which are two blocks long. Full directory listings with multiple segments contain blank lines as segment boundaries.

4.2.7 The Directory Initialization Operation

The /Z switch clears and initializes the directory of an RT-11 directory-structured device and writes logical end-of-file to a cassette or magtape device. The /Z operation must always be the first operation performed on a new (that is, previously unused) device. The form of the switch is:

```
/Z:n
```

where n is an optional octal number to increase the size of each directory entry on a directory-structured device. If n is not specified, each entry is 7 words long (for filename and file length information) and 72 entries can be made in a directory segment. When extra words are allocated, the number of entries per directory segment decreases. The formula for determining the number of entries per directory segment is:

$$507 / ((\# \text{ of extra words}) + 7)$$

For example, if the switch /Z:1 is used, 63 entries can be made per segment.

More information concerning the format of directory entries is supplied in Chapter 3 of the RT-11 Software Support Manual.

When /Z is used, PIP responds as follows:

```
device/Z ARE YOU SURE ?
```

For example:

```
*DT1:/Z
DT1:/Z ARE YOU SURE ?
```

Answer Y and a carriage return to perform the initialization. An answer beginning with a character other than Y is considered to be no.

Example:

```
*DT1:/Z
DT1:/Z ARE YOU SURE ?Y<CR>
*
Zeroes the directory on device DT1 and
allocates no extra words for the
directory.
```


.CHAIN

9.4.2 .CHAIN

This request allows a background program to pass control directly to another background program without operator intervention. Since this process may be repeated, a large "chain" of programs can be strung together.

The area from locations 500-507 contains the device name and file name (in RAD50) to be chained to, and the area from locations 510-777 is used to pass information between the chained programs.

Macro Call: .CHAIN

Notes:

1. No assumptions should be made concerning which areas of memory will remain intact across a .CHAIN. In general, 500-777 is the only area guaranteed to be preserved across a .CHAIN.
2. I/O channels are left open across a .CHAIN for use by the new program. However, I/O channels opened via a .CDFN request are not available in this way. Since the monitor reverts to to the original 16 channels during a .CHAIN, programs which leave files open across a .CHAIN should not use .CDFN. Furthermore, non-resident device handlers are released during a .CHAIN, and must be .FETChed again by the new program.
3. A program can determine whether it was CHAINED to or RUN from the keyboard by examining bit 8 of the JSW. This bit is on during program execution only if the program was entered via CHAIN. If a program normally loads into area 500-777, bit 8 of the JSW should be set during program assembly. This causes the monitor to load the area properly. If the bit is not set, locations 500-777 are preserved from the chaining program, causing the new program to malfunction.

Errors:

.CHAIN is implemented by simulating the monitor RUN command (described in Chapter 2), and can produce any errors which RUN can produce. If an error occurs, the .CHAIN is abandoned and the Keyboard Monitor is entered.

When using .CHAIN, care should be taken for initial stack placement, since the program being "chained to" is started. The Linker normally defaults the initial stack to 1000(octal); if caution is not observed, the stack may destroy chain data before it can be used (see Chapter 2, the RUN command).

Programmed Requests

Example:

```
      .MCALL  .,V2,..,REGDEF
      .,V2..
      .REGDEF
      .MCALL  .CHAIN,.TTYIN
START:
      MOV     #500,R1           ;SET UP TO CHAIN
      MOV     #CHPTR,R2        ;DEVICE, FILE NAME TO 500-511
      .REPT   4
      MOV     (R2)+,(R1)+
      .ENDR
LOOP:  .TTYIN                   ;NOW GET A COMMAND LINE
      MOV     R0,(R1)+         ;AND PASS IT TO THE JOB
      CMP     R0,#12           ;IN LOCATIONS 512 AND UP
      BNE     LOOP            ;LOOP UNTIL LINE FEED
      CLRB   (R1)+           ;PUT IN A NULL BYTE
      .CHAIN
CHPTR: .RAD50  /DK /
      .RAD50  /TECO /
      .RAD50  /SAV/
      .END    START
```

.CHCOPY

9.4.3 .CHCOPY (F/B only)

The .CHCOPY request opens a channel for input, logically connecting it to a file which is currently open by the other job for either input or output. This request may be used by either the foreground or the background. .CHCOPY must be done before the first .READ or .WRITE.

Macro Call: .CHCOPY .area, .chan, .ochan

where: .chan is the channel which the job will use to read the data.

.ochan is the channel number of the other job which is to be copied

Request Format:

R0 ⇒ .area:

13	.chan
	.ochan

.CHCOPY is legal only on files which are on disk or DECTape; however, no errors are detected by the system if another device is used. (To close a channel following use of .CHCOPY, use either the .CLOSE or .PURGE request.)

Notes:

1. If the other job's channel was opened via an .ENTER in order to create a file, the copier's channel indicates a file which extends to the highest block that the creator of the file had written at the time the .CHCOPY was executed.

REPLACEMENT PAGES

The following pages are page replaceable updates to the RT-11 System Release Notes for V02C.

In all cases, these pages replace the existing pages in this manual or in any addendum to it. Change bars indicate where changes were made.

RT-11 V02C SYSTEM RELEASE NOTES

Table 1
RT-11 V02C Version Identifications

System Component	V02C Identification	How Obtained
Monitors (S/J & F/B)	V02C-02	Automatic
ASEMBL	VS02-12	Listing
BATCH	V01-04	/V
CBUILD	V01-02	Automatic
CREP	V01-04	Listing
DUMP	V02-02	PATCH
EDIT	V02-12	EV\$\$
EXPAND	V02-02	Listing
FILEX	V02-02	/V
LIBR	V03-03	Map
LINK	V04-04	Map
MACRO	VM02-12	Listing
MBUILD	V02-03	Automatic
MSBOOT	V01-05	Automatic
MTINIT	V01-01	Automatic
ODT	V01-02	Automatic
PATCH	V01-02	Automatic
PATCHO	None	
PIP	V04-06	/V
PTBUILD	V02-01	Automatic
SRCCOM	V01-03	/H
SYSLIB	V6	SYSLBV

3.1 Important Memory Locations in V02C

Table 2 lists the V02C specific memory locations referenced in RT-11 documentation. The locations are documented for the purposes of modification; therefore they do not represent actual memory locations, but addresses in the disk file as accessed by PATCH.

3.2 System Device Handler Information

Table 3 lists pertinent information concerning RT-11 system device handlers. This information is useful when replacing one device with another in the system, modifying device handlers, adding new devices to the system, etc.

Table 2
Memory Locations in V02C

Location	S/J Monitor	F/B Monitor	Other
ALTTST	-	-	12106 (EDIT.SAV)
BASE	17000	20000	
CONFIG	35300*	37300*	
CREFL	-	-	1000 (CREF.SAV)
DSARG	-	-	14040 (EDIT.SAV)
DSSIZ	32662*	34614*	
\$DVSIZ	13660	14612	
DXIOP	-	-	1214 (DY.SYS)
EBASE	-	-	2402 (EDIT.SAV)
\$ENTRY	16564	17724	
GTVECT	35354*	37354*	
\$HSIZE	13624	14556	
HW	-	-	1366 (MT.SYS)
LISTFB	-	26032	
LOWMAP	16326	17326	
LP	-	-	2626 (DUMP.SAV)
MACR1	-	-	7662 (MACRO.SAV)
MAXBLK	16314	17314	
MSGO	-	-	2520 (DUMP.SAV)
MMHSIZ	32650*	34602*	
MMPNAM	35514*	37654*	
MMSTAT	35550*	37710*	
MTHSIZ	32640*	34572*	
MTPNAM	35504*	37644*	
MTSTAT	35540*	37700*	
PARDEN	-	-	1104 (MT.SYS)
\$PNAME	16470	17630	
RFSIZ	32666*	34620*	
RP23	43260*	54556*	1076 (DP.SYS)
\$STAT	16524	17664	
STATIN	340	340	
STATOUT	340	340	
TTCNFG	-	21416	
TTWIDT	-	21410	
UNIMOD	-	-	1312 (MM.SYS)
VECTIN	37572	47314	
VECTOUT	41026	50264	

*BASE has already been added to these numbers.

Location	Monitor (Same for S/J and F/B Versions)					
	RK	DT	DX	DP	DS	RF
BHALT	570	634	606	652	574	562
RELLST	1610	1556	1564	1622	1542	1504

Table 3
System Device Handler Information

Entry #	Octal Offset	Device Name		Device Code	Handler Size (Bytes)	Device Size ₈	Status Word
		ASCII	RAD5Ø				
1	0	TT	100040	4	500	0	4
2	2	DS	15770	16	320	2000*	100016
3	4	RK	71070	0	342	11300	100000
4	6	RF	70560	12	270	2000	100012
5	10	PR	63320	7	176	0	40007
6	12	PP	63200	10	132	0	20010
7	14	MT	52140	11	4300	0	12011
8	16	LP	46600	3	306	0	20003
9	20	DT	16040	1	322	1102	100001
10	22	CT	12740	13	3710	0	12013
11	24	MM	51510	20	4700	0	12020
12	26	BA	6250	4	4076	0	4
13	30	DP	15600	21	434	116110	100021
14	32	DX	16300	22	670	756	102022
-	-	CR		14	1326	0	40014

*RJS03. The RJS04 has 4000 blocks.

3.3 Linker

3.3.1 Changes

1. The V04 Linker (RT-11 V02C and V02B) is the result of dramatic performance improvements to the V03 Linker (RT-11 V02). One of the optimizations makes use of resident library directories; the library directories are read into memory and kept there until no longer needed.

The Linker needs approximately 10.5K of user space for this optimization to take effect. The Linker will function in less space, but performance deteriorates as memory decreases; the 10.5K point marks a sharp drop in the performance curve.

Users with 16K of memory, therefore, should be careful when loading handlers and options if concerned about link times. The F/B Monitor uses 3.5K, GT ON uses 1.25K, and handlers use from 100 (decimal) to 1000 (decimal) words each. If many of these options are invoked, the Linker will be left with less than 10.5K and will be unable to benefit from the major optimizations.

For the same reasons, the /S switch should not be used unless absolutely necessary. It disables several optimizations for the accompanying gain in symbol table space.

The user may combine object modules using the librarian, such as combining the modules in FORLIB and SYSLIB. Doing so makes it easier to specify the command string when linking some FORTRAN programs but can also cause the link time to be more than twice as long. This is because the FORTRAN library is almost at the maximum entry point limit that the Linker can keep resident in memory in 16K or larger memory configurations.

2. The CHAIN\$ bit in the job status word for LINK.SAV is set.

3.3.2 Restrictions

1. If a file of zero length is accidentally specified as input to the Linker, the Linker may malfunction when generating the link map. If this happens, type two CTRL C's to return to the monitor.
2. If relocatable code is to be linked for the foreground, no location may be filled more than once (using location counter arithmetic); any such location may be improperly relocated during the FRUN and may cause program or system failure. (See the RT-11 System Reference Manual, Section 6.3, for further information.)
3. If two or more libraries are used in a link, they should be specified on the same command line.
4. The Linker may print the message ?HARD I/O ERROR? when it exceeds memory.

3.4 Librarian

3.4.1 Changes

1. The V02 Librarian (RT-11 V02) had a bug that caused it to produce faulty library directories. Although these libraries were acceptable to the V03 Linker (RT-11 V02), the improved V04 Linker (RT-11 V02C and V02B) is sensitive to the bug and will not operate correctly on libraries built with the V02 Librarian. To correct the problem, libraries built with the V02 Librarian need only be run through the V03 Librarian; the result will be a correct library.

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.)

<u>Areas Covered</u>	<u>SPR Center</u>
Australia/New Zealand	Digital Equipment Australia Pty. Ltd. 123-125 Willoughby Road, P.O. Box 491 Crows Nest New South Wales, Australia 2065
Brazil	Digital Equipment Comercio E Industria LTDA Rua Batatais, 429 (Esq. Al. Campinas) 01423-Jardim Paulista São Paulo-SP-Brazil
Canada	Digital Equipment of Canada, Ltd. Software Services P.O. Box 11500, K2H 8K8 Ottawa, Ontario, Canada
Caribbean	Digital Equipment Latin America, Inc. 407 del Parque Street Santurce, Puerto Rico 00912
France	Digital Equipment France 18, rue Saarinen Centre Silic - CIDEX L225 F-94533 Rungis, France
Israel	DEC-sys Computers Ltd. 7 Habakuk Street IL-Tel Aviv 63505, Israel
Italy	Digital Equipment S.P.A. Corso Garibaldi 49 I-20121 Milano, Italy
Japan	Digital Equipment Corp. Intl. Kowa Building 25 (3d Floor) 8-7 Sunban-Cho Chiyoda-ku, Tokyo 102, Japan
Mexico	Equipo Digital, S.A. de C.V. 109 Concepcion Beistegui Mexico 12, D.F.
The Netherlands	Digital Equipment B.V. Kaap Hoorndreef 38, P.O. Box 9064 NL-Utrecht - Overvecht, The Netherlands
Nordic	Digital Equipment AB Englundavägen 7 S-17141 Solna Sweden
Switzerland	Digital Equipment Corp. SA 20, Quai Ernest Ansermet Case Postale 23, CH-1211 Geneva 8 Switzerland
Spain Portugal	
Greece Bulgaria	
Romania Yugoslavia	
United Kingdom	Digital Equipment Co. Ltd. Fountain House, Butts Centre GB-Reading RG1 7QN, England
West Germany Austria	
East Germany Russia	
Hungary Poland	
Czechoslovakia	
United States; remainder of Far East, Middle East, Africa, Latin America	Software Communications P.O. Box F Maynard, MA 01754

DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS 01754
European Headquarters: 81 route de l'Aire, 1211 Geneva 26. Switzerland
Digital Equipment of Canada Ltd., P.O. Box 11500 Ottawa, Ontario K2H8K8.

DIGITAL EQUIPMENT CORPORATION, Component Group Headquarters: 1 Iron Way,
Marlborough, Mass. 01752, Telephone: (617) 481-7400

DIGITAL EQUIPMENT CORPORATION, Corporate Headquarters: Maynard,
Massachusetts 01754, Telephone: (617) 897-5111

SALES AND SERVICE OFFICES

DOMESTIC — ARIZONA, Phoenix and Tucson • CALIFORNIA, Los Angeles, Monrovia,
Oakland, Ridgecrest, San Diego, San Francisco (Mountain View), Santa Ana, Sunnyvale,
and Woodland Hills • COLORADO, Englewood • CONNECTICUT, Fairfield and Meriden
• DISTRICT OF COLUMBIA, Washington (Latham, Md.) • FLORIDA, Orlando • GEORGIA,
Atlanta • ILLINOIS, Chicago (Rolling Meadows) • INDIANA, Indianapolis • IOWA,
Bettendorf • KENTUCKY, Louisville • LOUISIANA, Metairie (New Orleans)
• MASSACHUSETTS, Marlborough and Waltham • MICHIGAN, Detroit (Farmington
Hills) • MINNESOTA, Minneapolis • MISSOURI, Kansas City and St. Louis • NEW
HAMPSHIRE, Manchester • NEW JERSEY, Fairfield, Metuchen and Princeton • NEW
MEXICO, Albuquerque • NEW YORK, Albany, Huntington Station, Manhattan, Rochester
and Syracuse • NORTH CAROLINA, Durham/Chapel Hill • OHIO, Cleveland, Columbus
and Dayton • OKLAHOMA, Tulsa • OREGON, Portland • PENNSYLVANIA, Philadelphia
(Bluebell) and Pittsburgh • TENNESSEE, Knoxville • TEXAS, Austin, Dallas and Houston
• UTAH, Salt Lake City • 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, Puerto Alegre, Rio de Janeiro and São Paulo • CANADA,
Calgary, Halifax, Montreal, Ottawa, Toronto and Vancouver • CHILE, Santiago
• DENMARK, Copenhagen • FINLAND, Helsinki • FRANCE, Grenoble and Paris
• GERMANY, Berlin, Cologne, Hannover, Hamburg, Frankfurt, Munich and Stuttgart
• HONG KONG • INDIA, Bombay • INDONESIA, Djakarta • ISRAEL, Tel Aviv • ITALY,
Milan and Turin • JAPAN, Osaka and Tokyo • MALAYSIA, Kuala Lumpur • MEXICO,
Mexico City • NETHERLANDS, Utrecht • NEW ZEALAND, Auckland
• NORWAY, Oslo • PHILIPPINES, Manila • PUERTO RICO, Santurce • SINGAPORE
• SPAIN, Barcelona and Madrid • SWEDEN, Gothenburg and Stockholm
• SWITZERLAND, Geneva and Zurich • TAIWAN, Taipei and Taoyuan • UNITED
KINGDOM, Birmingham, Bristol, Dublin, Edinburgh, Leeds, London, Manchester
and Reading • VENEZUELA, Caracas • YUGOSLAVIA, Ljubljana •

digital

digital equipment corporation