


```

RRRRRRRR      MM      MM      11      DDDDDDDD      IIIIII      SSSSSSSS      CCCCCCCC      000000      NN      NN
RRRRRRRR      MM      MM      11      DDDDDDDD      IIIIII      SSSSSSSS      CCCCCCCC      000000      NN      NN
RR      RR      MMMM      MMMM      1111      DD      DD      SS      SSSSSSSS      CCCCCCCC      00      00      NN      NN
RR      RR      MMMM      MMMM      1111      DD      DD      SS      SSSSSSSS      CCCCCCCC      00      00      NN      NN
RR      RR      MM      MM      11      DD      DD      SS      SSSSSS      CCCCCC      00      00      NNNN      NN
RR      RR      MM      MM      11      DD      DD      SS      SSSSSS      CCCCCC      00      00      NNNN      NN
RRRRRRRR      MM      MM      11      DD      DD      SS      SSSSSS      CCCCCC      00      00      NN      NN
RRRRRRRR      MM      MM      11      DD      DD      SS      SSSSSS      CCCCCC      00      00      NN      NN
RR      RR      MM      MM      11      DD      DD      SS      SS      CC      00      00      NN      NNNN
RR      RR      MM      MM      11      DD      DD      SS      SS      CC      00      00      NN      NNNN
RR      RR      MM      MM      11      DD      DD      SS      SS      CC      00      00      NN      NN
RR      RR      MM      MM      11      DD      DD      SS      SS      CC      00      00      NN      NN
RR      RR      MM      MM      111111      DDDDDDDD      IIIIII      SSSSSSSS      CCCCCCCC      000000      NN      NN
RR      RR      MM      MM      111111      DDDDDDDD      IIIIII      SSSSSSSS      CCCCCCCC      000000      NN      NN

```

```

LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLLL      IIIIII      SSSSSSSS

```

(2) 53
(3) 75

DECLARATIONS
RMSDISCONNECT1 - SEQ. FILE ORG. SPECIFIC DISCONNECT CODE

```

0000 1          $BEGIN RM1DISCON,000,RM$RMS1,<DISCONNECT FOR SEQ. ORG.>
0000 2
0000 3 :*****
0000 4 :*
0000 5 :*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 6 :*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 7 :*  ALL RIGHTS RESERVED.
0000 8 :*
0000 9 :*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 10 :*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 11 :*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 12 :*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 13 :*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 14 :*  TRANSFERRED.
0000 15 :*
0000 16 :*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 17 :*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 18 :*  CORPORATION.
0000 19 :*
0000 20 :*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 21 :*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 22 :*
0000 23 :*
0000 24 :*****
0000 25
0000 26 :++
0000 27 : FACILITY: RMS32
0000 28
0000 29 : ABSTRACT:
0000 30 :           Module to give back all storage associated with IRAB.
0000 31 :
0000 32 :
0000 33 : ENVIRONMENT:
0000 34 :           STAR processor running STARLET EXEC.
0000 35 :
0000 36 : AUTHOR: L F Laverdure,           CREATION DATE: 31-Mar-1977
0000 37 :
0000 38 : MODIFIED BY:
0000 39 :
0000 40 :           V03-001 KBT0138           Keith B. Thompson           20-Aug-1982
0000 41 :           Reorganize psects
0000 42 :
0000 43 :           V02-008 REFORMAT           Frederick E. Deen, Jr.           25-Jul-1980
0000 44 :           This code was reformatted to adhere to RMS standards
0000 45 :
0000 46 :           V007 PSK001           P S Knibbe           05-Dec-1979
0000 47 :           On foreign magtapes the IRBSV_EOF bit is moved into
0000 48 :           the IFBSV_EOF bit.
0000 49 :
0000 50 :--
0000 51

```

```
0000 53      .SBTTL  DECLARATIONS
0000 54
0000 55  :
0000 56  : INCLUDE FILES:
0000 57  :
0000 58
0000 59  :
0000 60  : MACROS:
0000 61  :
0000 62
0000 63      $IFBDEF
0000 64      $IRBDEF
0000 65      $DEVDEF
0000 66
0000 67  :
0000 68  : EQUATED SYMBOLS:
0000 69  :
0000 70  :
0000 71  : OWN STORAGE:
0000 72  :
0000 73
```

```
0000 75 .SBTTL RMSDISCONNECT1 - SEQ. FILE ORG. SPECIFIC DISCONNECT CODE
0000 76
0000 77 :++
0000 78 : RMSDISCONNECT1 - Sequential file organization specific DISCONNECT code
0000 79 :
0000 80 : 1.0 If not doing BLOCK I/O, call RMSWTLST1 to write out last
0000 81 : block with padding and extending as required.
0000 82 : 2.0 If disk reset IFBSV_EOF.
0000 83 : 2.5 If foreign magtape, move IRBSV_EOF bit to IFBSV_EOF bit
0000 84 : 3.0 Jump to RMSDISCOMMON
0000 85 :
0000 86 : CALLING SEQUENCE:
0000 87 :
0000 88 : BSBW RMSDISCONNECT1
0000 89 :
0000 90 : (entered at RMSDISCONNECT1 via case branch from
0000 91 : RMSDISCONNECT with return PC on STACK)
0000 92 :
0000 93 : INPUT PARAMETERS:
0000 94 :
0000 95 : R11 IMPURE AREA address
0000 96 : R10 IFAB address
0000 97 : R9 IRAB address
0000 98 : R8 RAB address
0000 99 :
0000 100 : IMPLICIT INPUTS:
0000 101 :
0000 102 : The contents of the various RMS internal structures
0000 103 :
0000 104 : OUTPUT PARAMETERS:
0000 105 :
0000 106 : R0 STATUS CODE
0000 107 : R1-R7,AP destroyed
0000 108 :
0000 109 : IMPLICIT OUTPUTS:
0000 110 :
0000 111 : IFBSV_EOF cleared
0000 112 : The implicit outputs of RMSWTLST1
0000 113 :
0000 114 : COMPLETION CODES:
0000 115 :
0000 116 : Standard RMS
0000 117 :
0000 118 : SIDE EFFECTS:
0000 119 :
0000 120 : May be running at AST level.
0000 121 :--
0000 122
```

```

0000 124 RMSDISCONNECT1::
0000 125      $TSTPT DISCON1
0C 50 01 D0 0006 126      MOVL #1,R0 ; anticipate success
0C 69 22 E0 0009 127      BBS #IRBSV_PPF_IMAGE,(R9),10$ ; branch if indirect PPF
; to avoid write
07 22 AA 05 E0 000D 128      ;
03 69 27 E0 000D 129      BBS #IFBSV_BIO,IFBSB_FAC(R10),10$ ; branch if BLOCK I/O
0012 130      BBS #IRBSV_BIO_LAST,(R9),10$ ; branch if last operation
; was a BLOCK I/O operation
0016 131      ; (mixed block and rec. ops)
0016 132      ; write last block if needed
FFE7' 30 0016 133      BSBW RMSWTLST1 ; branch if not disk
1C E1 0019 134 10$: BBC #DEVSV_RND,-
04 6A 001B 135      IFBSL PRIM_DEV(R10),15$
001D 136      CSB #IFBSV_EOF,(R10) ; clear EOF flag
05 E1 0021 137 15$: BBC #DEVSV_SQD,- ; branch if not magtape
10 6A 0023 138      IFBSL PRIM_DEV(R10),20$
0C 6A E1 0025 139      BBC #DEVSV_FOR,- ; branch if not foreign
0027 140      IFBSL PRIM_DEV(R10),20$
04 69 21 E1 0029 141      CSB #IFBSV_EOF,(R10) ; assume it's not at EOF
002D 142      BBC #IRBSV_EOF,(R9),20$ ; that's right
0031 143      SSB #IFBSV_EOF,(R10) ; nope - set IFAB bit
09 22 AA FFC8' 30 0035 144 20$: BSBW RMSDISCOMMON ; go finish up
05 E1 0038 145      BBC #IFBSV_BIO,IFBSB_FAC(R10),30$ ; branch if not BLOCK I/O
003D 146      ;
003D 147      ; This connect was for BLOCK I/O.
003D 148      ; Reset to BRO if also set.
003D 149      ;
003D 150      ;
003D 151      ;
04 22 AA 06 E1 003D 152      BBC #IFBSV_BRO,IFBSB_FAC(R10),30$ ; branch if BRO not also set
22 AA 20 8A 0042 153      BICB2 #IFBSM_BIO,IFBSB_FAC(R10) ; clear BIO
05 0046 154 30$: RSB ; return to caller
0047 155      .END

```

```

$$PSECT EP          = 00000000
$$RMSTEST          = 0000001A
$$RMS_PBUGCHK      = 00000010
$$F45_TBUGCHK      = 00000008
$$RMS_UMODE        = 00000004
DEVSV_FOR          = 00000018
DEVSV_RND          = 0000001C
DEVSV_SQD          = 00000005
IFBSB_FAC          = 00000022
IFBSL_PRIM_DEV     = 00000000
IFBSM_BIO          = 00000020
IFBSV_BIO          = 00000005
IFBSV_BRO          = 00000006
IFBSV_EOF          = 00000021
IRBSV_BIO_LAST     = 00000027
IRBSV_EOF          = 00000021
IRBSV_PPF_IMAGE   = 00000022
PIO$A_TRACE        ***** X 01
RMSDISCOMMON       ***** X 01
RMSDISCONNECT1    00000000 RG 01
RMSWTLST1          ***** X 01
TPTSL_DISCON1     ***** X 01
    
```

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
RMSRMS1	00000047 (71.)	01 (1.)	PIC USR CON REL GBL NOSHR EXE RD NOWRT NOVEC BYTE
\$ABSS	00000000 (0.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	36	00:00:00.07	00:00:00.92
Command processing	137	00:00:00.68	00:00:04.70
Pass 1	201	00:00:04.38	00:00:15.95
Symbol table sort	0	00:00:00.54	00:00:01.01
Pass 2	43	00:00:00.82	00:00:02.89
Symbol table output	4	00:00:00.02	00:00:00.11
Psect synopsis output	1	00:00:00.02	00:00:00.08
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	424	00:00:06.56	00:00:25.67

The working set limit was 1200 pages.
 22790 bytes (45 pages) of virtual memory were used to buffer the intermediate code.
 There were 30 pages of symbol table space allocated to hold 444 non-local and 7 local symbols.
 155 source lines were read in Pass 1, producing 13 object records in Pass 2.
 15 pages of virtual memory were used to define 14 macros.

↑-----↑
! Macro library statistics !
↑-----↑

Macro library name	Macros defined
-----	-----
-\$255\$DUA28:[RMS.OBJ]RMS.MLB;1	6
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	0
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	4
TOTALS (all libraries)	10

531 GETS were required to define 10 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:RM1DISCON/OBJ=OBJ\$:RM1DISCON MSRC\$:RM1DISCON/UPDATE=(ENH\$:RM1DISCON)+EXECMLS/LIB+LIB\$:RMS/LIB

RM1CONN
LIS

RM1GET
LIS

RM1INPSON
LIS

RM1DISCON
LIS

RM1GETINT
LIS

RM1CREATE
LIS

RM1JOURNL
LIS