



```
RRRRRRRR MM MM 333333 XX XX SSSSSSSS UU UU MM MM 000000
RRRRRRRR MM MM 333333 XX XX SSSSSSSS UU UU MM MM 000000
RR RR RR MMMM MMMM 33 33 XX XX SS SS UU UU MMMM MMMM 00 00
RR RR RR MMMM MMMM 33 33 XX XX SS SS UU UU MMMM MMMM 00 00
RR RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RRRRRRRR MM MM MM 33 33 XX XX SSSSSS UU UU MM MM MM 00 00
RRRRRRRR MM MM MM 33 33 XX XX SSSSSS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SS SS UU UU MM MM MM 00 00
RR RR MM MM MM 33 33 XX XX SSSSSSSS UUUUUUUUUU MM MM 000000
RR RR MM MM MM 33 33 XX XX SSSSSSSS UUUUUUUUUU MM MM 000000
.....
.....
.....
.....
```

```
LL LL I I I I I I SSSSSSSS
LL LL I I I I I I SSSSSSSS
LL LL I I SS
LL LL I I SS
LL LL I I SS
LL LL I I SSSSSS
LL LL I I SSSSSS
LL LL I I SS
LL LL I I SS
LL LL I I SS
LL LL I I SS
LLLLLLLLLLLL I I I I I I SSSSSSSS
LLLLLLLLLLLL I I I I I I SSSSSSSS
```

```

1 0001 0 MODULE RM3XSUMO (LANGUAGE (BLISS32) ,
2 0002 0 IDENT = 'V04-000'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
10 0010 1 * ALL RIGHTS RESERVED. *
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
17 0017 1 * TRANSFERRED. *
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
21 0021 1 * CORPORATION. *
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1 ++
30 0030 1
31 0031 1 FACILITY: RMS32 index sequential file organization
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 Fills in the SUMMARY XAB
36 0036 1
37 0037 1
38 0038 1 ENVIRONMENT:
39 0039 1
40 0040 1 VAX/VMS operating system
41 0041 1
42 0042 1 --
43 0043 1
44 0044 1
45 0045 1 AUTHOR: D. M. BOUSQUET
46 0046 1 CREATION DATE: 17-JUL-78 16:40
47 0047 1
48 0048 1
49 0049 1 MODIFIED BY:
50 0050 1
51 0051 1 V03-003 MCN0002 Maria del C. Nasr 05-Apr-1983
52 0052 1 Routine RM$XSUMO needs its own linkage to preserve R4.
53 0053 1
54 0054 1 V03-002 MCN0001 Maria del C. Nasr 31-Mar-1983
55 0055 1 Reorganize linkages
56 0056 1
57 0057 1 V03-001 KBT0290 Keith B. Thompson 23-Aug-1982

```

```

: 58      0058 1  | Reorganize psects
: 59      0059 1  |
: 60      0060 1  | V02-004 REFORMAT      D M WALP      24-JUL-1980
: 61      0061 1  |
: 62      0062 1  | V02-003 CDS0001      C D SAETHER  13-MAR-1980
: 63      0063 1  |      return OK_NOP alt success if block i/o
: 64      0064 1  |
: 65      0065 1  | REVISION HISTORY:
: 66      0066 1  |
: 67      0067 1  | Wendy Koenig,      24-OCT-78  14:03
: 68      0068 1  | X0002 - make changes caused by sharing conventions
: 69      0069 1  |
: 70      0070 1  | *****
: 71      0071 1  |
: 72      0072 1  | LIBRARY 'RMSLIB:RMS';
: 73      0073 1  |
: 74      0074 1  | REQUIRE 'RMSSRC:RMSIDXDEF';
: 75      0139 1  |
: 76      0140 1  | ! define default psects for code
: 77      0141 1  |
: 78      0142 1  | PSECT
: 79      0143 1  |     CODE = RMSRMS3(PSECT_ATTR),
: 80      0144 1  |     PLIT = RMSRMS3(PSECT_ATTR);
: 81      0145 1  |
: 82      0146 1  | ! Linkages
: 83      0147 1  |
: 84      0148 1  | LINKAGE
: 85      0149 1  |     L_XSUMO;
```

```

: 87 0150 1 GLOBAL ROUTINE RM$XSUMO3 (XAB, FLAGS) : RL$XSUMO =
: 88 0151 1
: 89 0152 1 !++
: 90 0153 1
: 91 0154 1 FUNCTIONAL DESCRIPTION:
: 92 0155 1
: 93 0156 1
: 94 0157 1 This subroutine fills in the various fields in the summary xab
: 95 0158 1 by extracting info from the IFAB.
: 96 0159 1
: 97 0160 1 It also makes certain that there is only one summary XAB and if
: 98 0161 1 not jumps to an error routine in XAB_SCAN via a bliss/mars interface
: 99 0162 1 routine.
100 0163 1
101 0164 1
102 0165 1 CALLING SEQUENCE:
103 0166 1
104 0167 1 BSBW RM$XSUMO3
105 0168 1
106 0169 1 INPUT PARAMETERS:
107 0170 1
108 0171 1 R3 - (XAB), XAB address
109 0172 1 R4 - (flags), to make sure this is the only SUMMARY XAB
110 0173 1
111 0174 1 IMPLICIT INPUTS:
112 0175 1 IFAB -
113 0176 1 [AMAX] - number of areas
114 0177 1 [NUM_KEYS] - number of keys
115 0178 1 [PLG_VER] - prologue version
116 0179 1 [BIO] - open for block i/o if set
117 0180 1
118 0181 1 OUTPUT PARAMETERS:
119 0182 1 NONE
120 0183 1
121 0184 1 IMPLICIT OUTPUTS:
122 0185 1
123 0186 1 SUMMARY XAB fields filled in and RMSERR(IMX) status code
124 0187 1
125 0188 1 ROUTINE VALUE:
126 0189 1 NONE
127 0190 1
128 0191 1 SIDE EFFECTS:
129 0192 1 NONE
130 0193 1
131 0194 1 --
132 0195 1
133 0196 2 BEGIN
134 0197 2
135 0198 2 ! define common registers
136 0199 2 !
137 0200 2 EXTERNAL LITERAL
138 0201 2 XBC$C_OPNSUM3;
139 0202 2
140 0203 2 EXTERNAL REGISTER
141 0204 2 COMMON_FAB_STR;
142 0205 2
143 0206 2 MAP

```

```

: 144      0207      2          FLAGS   : BITVECTOR [32],
: 145      0208      2          XAB     : REF BBLOCK;
: 146      0209      2
: 147      0210      2          ! just to make sure this has indexed file organization
: 148      0211      2
: 149      0212      2          !
: 150      0213      2          IF .IFAB[IFB$B_ORGCASE] EQL IFB$C_IDX
: 151      0214      2          THEN
: 152      0215      2              BEGIN
: 153      0216      2                  !
: 154      0217      2                  !     don't do anything if this is block i/o
: 155      0218      2                  !
: 156      0219      2                  IF .IFAB[IFB$V_BIO]
: 157      0220      2                  THEN
: 158      0221      2                      RETURN RMSSUC(OK_NOP);
: 159      0222      2                  !
: 160      0223      2                  !     have a duplicate SUMMARY XAB - error
: 161      0224      2                  !
: 162      0225      2          IF .FLAGS[XBC$C_OPNSUM3]
: 163      0226      2          THEN
: 164      0227      2              RETURN RMSERR(IMX)
: 165      0228      2          ELSE
: 166      0229      2              BEGIN
: 167      0230      2                  !
: 168      0231      2                  !     else set the flag
: 169      0232      2                  !
: 170      0233      2                  !     FLAGS[XBC$C_OPNSUM3] = 1;
: 171      0234      2                  !
: 172      0235      2                  !     fill in the fields of the xab from the ifab
: 173      0236      2                  !
: 174      0237      2                  XAB[XAB$B_NOA] = .IFAB[IFB$B_AMAX];
: 175      0238      2                  XAB[XAB$B_NOK] = .IFAB[IFB$B_NUM_KEYS];
: 176      0239      2                  XAB[XAB$W_PVN] = .IFAB[IFB$B_PLG_VER];
: 177      0240      2                  END;
: 178      0241      2              END;
: 179      0242      2          END;
: 180      0243      2          RETURN RMSSUC(SUC);
: 181      0244      2
: 182      0245      2
: 183      0246      1          END;

```

```

.TITLE  RM3XSUMO
.IDENT  \V04-000\
.EXTRN  XBC$C_OPNSUM3
.PSECT  RMSRMS3,NOWRT, GBL, PIC,2

```

		02	23	AA	91	0000	RM3XSUMO3::			
							CMPB	35(IFAB), #2		: 0212
							BNEQ	4\$		: :
06	22	AA		05	E1	00006	BBC	#5, 34(IFAB), 1\$		: 0218
		50	8059	8F	3C	0000B	MOVZWL	#32857, R0		: 0220
						05 00010	RSB			: :
06	08	AE	00000000G	8F	E1	00011 1\$:	BBC	#XBC\$C_OPNSUM3, FLAGS, 2\$		: 0225
		50	856C	8F	3C	0001A	MOVZWL	#34156, R0		: 0227

```

00      08 AE 00000000G 8F E2 00020 2$: RSB
      50      04 AE D0 00029 3$: BBSS #XBCSC OPNSUM3, FLAGS, 3$
      08 A0 00B1 CA B0 0002D MOVL XAB, R0
      0A A0 00B7 CA 9B 00033 MOVW 177(IFAB), 8(R0)
      50      01 D0 00039 4$: MOVZBW 183(IFAB), 10(R0)
      05 0003C RSB #1, R0

```

; Routine Size: 61 bytes, Routine Base: RMSRMS3 + 0000

```

: 184      0247 1
: 185      0248 1 END
: 186      0249 1
: 187      0250 0 ELUDOM

```

PSECT SUMMARY

Name	Bytes	Attributes
RMSRMS3	61	NOVEC,NOWRT, RD, EXE,NOSHR, GBL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	Symbols		Pages Mapped	Processing Time
	Total	Loaded Percent		
_\$255\$DUA28:[RMS.OBJ]RMS.L32;1	3109	27 0	154	00:00.4

COMMAND QUALIFIERS

; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:RM3XSUMO/OBJ=OBJ\$:RM3XSUMO MSRC\$:RM3XSUMO/UPDATE=(ENH\$:RM3XSUMO)

```

; Size:          61 code + 0 data bytes
; Run Time:      00:03.0
; Elapsed Time: 00:09.0
; Lines/CPU Min: 5000
; Lexemes/CPU-Min: 13140
; Memory Used:  40 pages
; Compilation Complete

```

