



```

CCCCCCCC RRRRRRRR EEEEEEEEE MM MM AAAAAA PPPPPPPP SSSSSSSS EEEEEEEEE CCCCCCCC
CCCCCCCC RRRRRRRR EEEEEEEEE MM MM AAAAAA PPPPPPPP SSSSSSSS EEEEEEEEE CCCCCCCC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CC RR RR EE MM MM AA AA PP PP SS EEEEEEEEE CC
CCCCCCCC RR RR EEEEEEEEE MM MM AA AA PP SSSSSSS EEEEEEEEE CCCCCCCC
CCCCCCCC RR RR EEEEEEEEE MM MM AA AA PP SSSSSSS EEEEEEEEE CCCCCCCC

```

```

LL IIIIII SSSSSSS
LL IIIIII SSSSSSS
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 IIIIII SSSSSSS
LLLLLLLLLL IIIIII SSSSSSS
LLLLLLLLLL IIIIII SSSSSSS

```

(2) 47  
(3) 88

DECLARATIONS  
CREATE AND MAP SECTION



```
0000 47 .SBTTL DECLARATIONS
0000 48 :
0000 49 : INCLUDE FILES:
0000 50 :
0000 51 $ATRDEF ;FILE ATTRIBUTE BLOCK DEFINITIONS
0000 52 $FABDEF ;FILE ACCESS BLOCK DEFINITIONS
0000 53 $SECDEF ;DEFINE SECTION FLAGS
0000 54 $XABDEF ;EXTENDED ATTRIBUTE BLOCK DEFINITIONS
0000 55 :
0000 56 : MACROS:
0000 57 :
0000 58 :
0000 59 : EQUATED SYMBOLS:
0000 60 :
0000 61 :
0000 62 : OFFSETS FROM AP
0000 63 :
0000 64 $OFFSET 4,POSITIVE,<-
0000 65 INADR,- ;INPUT ADDRESS RANGE
0000 66 RETADR,- ;RETURN ADDRESS RANGE
0000 67 FLAGS,- ;SECTION FLAGS
0000 68 GSDNAM,- ;GLOBAL SECTION NAME DESCRIPTOR
0000 69 RELPAG,- ;RELATIVE PAGE IN SECTION TO START MAPPING
0000 70 FILNAM,- ;FILE NAME DESCRIPTOR
0000 71 PAGCNT,- ;MAX PAGE COUNT TO CREATE
0000 72 VBN,- ;STARTING VIRTUAL BLOCK IN FILE TO CREATE SE
0000 73 ACMODE,- ;ACCESS MODE OF SECTION
0000 74 CHANADR- ;ADDRESS OF CHANNEL TO BE USED/RETURNED
0000 75 >
0004 INADR:
0008 RETADR:
000C FLAGS:
0010 GSDNAM:
0014 RELPAG:
0018 FILNAM:
001C PAGCNT:
0020 VBN:
0024 ACMODE:
0028 CHANADR:
0000 76 :
0000 77 : OFFSETS FROM FP
0000 78 :
0000 79 $OFFSET 0,NEGATIVE,<-
0000 80 SVACMODE,- ;SAVED ACMODE PARAMETER IF PRESENT
0000 81 <FAB,FAB$C,BLN>,- ;FAB
0000 82 <SCRATCHSIZE,0>- ;SIZE OF SCRATCH AREA
0000 83 >
FFFC SVACMODE:
FFAC FAB:
FFAC SCRATCHSIZE:
0000 84 :
0000 85 : OWN STORAGE:
0000 86 :
```

```
0000 88      .SBTTL  CREATE AND MAP SECTION
0000 89      :++
0000 90      : FUNCTIONAL DESCRIPTION:
0000 91      :
0000 92      :     THIS ROUTINE PROVIDES A SLIGHTLY SIMPLER INTERFACE TO THE
0000 93      : CREATE AND MAP SECTION SYSTEM SERVICE.  IT'S FUNCTION HOWEVER IS
0000 94      : IDENTICAL, I.E. TO CREATE AND MAP A SECTION.
0000 95      :
0000 96      : CALLING SEQUENCE:
0000 97      :
0000 98      :     CALLG  ARGLIST,CREMAPSEC
0000 99      :
0000 100     : INPUT PARAMETERS:
0000 101     :
0000 102     :     INADR(AP)      =           ; INPUT ADDRESS RANGE TO MAP
0000 103     :     RETADR(AP)   =           ; RETURN ADDRESS RANGE
0000 104     :     FLAGS(AP)   =           ; SECTION CONTROL FLAGS
0000 105     :     GSDNAM(AP)  =           ; GLOBAL SECTION NAME DESCRIPTOR
0000 106     :     RELPAG(AP)  =           ; RELATIVE PAGE IN SECTION TO START MAPPING
0000 107     :     FILNAM(AP) =           ; DESCRIPTOR OF FILE NAME STRING
0000 108     :     PAGCNT(AP) =           ; SIZE OF SECTION TO CREATE
0000 109     :     VBN(AP)    =           ; STARTING VBN OF FILE TO CREATE SECTION FROM
0000 110     :     ; IF THE ABOVE 2 PARAMETERS ARE LEFT OUT
0000 111     :     ; THE SECTION STARTS AT VBN 1 OF THE FILE
0000 112     :     ; AND IS THE FILE LENGTH IN SIZE
0000 113     :     PFN(AP)    =           ; IF PFNMAPPING, IS STARTING PFN TO MAP
0000 114     :     ; TO NOT VBN
0000 115     :
0000 116     :     ACMODE(AP)  =           ; OPTIONAL PARAMETER SPECIFYING ACCESS MODE
0000 117     :     CHANADR(AP) =           ; OPTIONAL PARAMETER SPECIFYING EITHER
0000 118     :     ; ADDRESS OF CHANNEL TO USE OR ADDRESS
0000 119     :     ; TO HOLD RETURNED CHANNEL USED
0000 120     :
0000 121     : IMPLICIT INPUTS:
0000 122     :
0000 123     :     NONE
0000 124     :
0000 125     : OUTPUT PARAMETERS:
0000 126     :
0000 127     :     RO = SYSTEM STATUS CODE
0000 128     :
0000 129     : IMPLICIT OUTPUTS:
0000 130     :
0000 131     :     CHANNEL USED IS RETURNED VIA OPTIONAL PARAMETER, CHANADR.
0000 132     :
0000 133     : COMPLETION CODES:
0000 134     :
0000 135     :     NONE
0000 136     :
0000 137     : SIDE EFFECTS:
0000 138     :
0000 139     :     NONE
0000 140     :
0000 141     :--
```

|    |      |    |    |    |    |    |    |          |     |        |   |   |
|----|------|----|----|----|----|----|----|----------|-----|--------|---|---|
| 6E | 0054 | 8F | 00 | SE | AC | AE | DE | 0000     | 143 | .LIST  | MEB   |   |
|    |      |    |    |    |    |    |    | 0000     | 144 |        |   |   |
|    |      |    |    |    |    |    |    | 00000000 | 145 | .PSECT | _LIB\$CODE PIC,SHR,BYTE,RD,EXE,NOWRT                                |   |
|    |      |    |    |    |    |    |    | 0000     | 146 |        |   |   |
|    |      |    |    |    |    |    |    | 006C     | 147 | .ENTRY | LIB\$CREMAPSEC,^M<R2,R3,R5,R5,R6>                                   |   |
|    |      |    | 02 |    |    |    | 11 | 0002     | 148 | BRB    | START   |   |
|    |      |    |    |    |    |    |    | 0004     | 149 |        |   |   |
|    |      |    |    |    |    |    |    | 007C     | 150 | .ENTRY | LIB\$ CREMAPSEC,^M<R2,R3,R4,R5,R6>                                  |   |
|    |      |    |    |    |    |    |    | 0006     | 151 | START: | MOVAL   | SCRATCHSIZE(SP),SP ;RESERVE SCRATCH STORAGE             |
|    |      |    |    |    |    |    |    | 000A     | 152 |        | MOVCS   | #0,(SP),#0,#-SCRATCHSIZE,(SP) ;ZERO THE SCRATCH STORAGE |
|    |      |    |    |    |    |    |    | 0012     | 153 |        | CMPL  | (AP),#<ACMODE/4> ;IS ACCESS MODE PARAMETER PRESENT?     |
|    |      |    |    |    |    |    |    | 0015     | 154 |        | BLSS  | 20\$ ;BRANCH IF NOT                                     |
|    |      |    |    |    |    |    |    | 0017     | 155 |        | MOVL  | ACMODE(AP),SVACMODE(FP) ;YES, SAVE IT IN A TEMP         |
|    |      |    |    |    |    |    |    | 001C     | 156 | 20\$:  | MOVL  | FLAGS(AP),R4 ;GET CONTROL FLAGS                         |
|    |      |    |    |    |    |    |    | 0020     | 157 |        | BBC   | #SECSV GBL,R4,40\$ ;BRANCH IF PROCESS SECTION           |
|    |      |    |    |    |    |    |    | 0024     | 158 |        | MOVQ  | @INADR(AP),R2 ;SAVE INPUT ADDRESS RANGE                 |
|    |      |    |    |    |    |    |    | 0028     | 159 |        |   | ;COULD BE OVERWRITTEN BY RETADR IN                      |
|    |      |    |    |    |    |    |    | 0028     | 160 |        |   | ;AN UNSUCCESSFUL MGBLSC ATTEMPT                         |
|    |      |    |    |    |    |    |    | 0028     | 161 | CLRL   | R1 ;ASSUME NO RELATIVE PAGE OFF. SPECIFIED                          |   |
|    |      |    |    |    |    |    |    | 002A     | 162 | BBC    | #SECSV PFNMAP,R4,25\$ ;BRANCH IF NOT PFNMAP-ING                     |   |
|    |      |    |    |    |    |    |    | 002E     | 163 | MOVL   | RELPA@AP),R1 ;GET RELATIVE PAGE OFFSET                              |   |
|    |      |    |    |    |    |    |    | 0032     | 164 | 25\$:  | \$MGBLSC  | S -   |
|    |      |    |    |    |    |    |    | 0032     | 165 |        |   | INADR=@INADR(AP),- ;ADDRESS OF INPUT ADDRESS RANGE      |
|    |      |    |    |    |    |    |    | 0032     | 166 |        |   | RETADR=@RETADR(AP),- ;ADDRESS OF RETURN ADDRESS RANGE   |
|    |      |    |    |    |    |    |    | 0032     | 167 |        |   | FLAGS=R4,- ;SECTION FLAGS                               |
|    |      |    |    |    |    |    |    | 0032     | 168 |        |   | ACMODE=SVACMODE(FP),- ;ACCESS MODE                      |
|    |      |    |    |    |    |    |    | 0032     | 169 |        |   | GSDNAM=@GSDNAM(AP),- ;ADDRESS OF GLOBAL SECTION NAME    |
|    |      |    |    |    |    |    |    | 0032     | 170 |        |   | RELPA@R1 ;RELATIVE PAGE IN SECTION                      |
|    |      |    |    |    |    |    |    | 0032     |     |        | PUSHL   | R1  |
|    |      |    |    |    |    |    |    | 0034     |     |        | PUSHL   | #0  |
|    |      |    |    |    |    |    |    | 0036     |     |        | PUSHAQ  | @GSDNAM(AP)   |
|    |      |    |    |    |    |    |    | 0039     |     |        | PUSHL   | R4  |
|    |      |    |    |    |    |    |    | 003B     |     |        | PUSHL   | SVACMODE(FP)  |
|    |      |    |    |    |    |    |    | 003E     |     |        | PUSHAQ  | @RETADR(AP)   |
|    |      |    |    |    |    |    |    | 0041     |     |        | PUSHAQ  | @INADR(AP)  |
|    |      |    |    |    |    |    |    | 0044     |     |        | CALLS   | #7,G^SYSS\$MGBLSC                                       |
|    |      |    |    |    |    |    |    | 004B     | 171 | BLBC   | RO,35\$ ;BRANCH ON ERROR MAPPING FILE                               |   |
|    |      |    |    |    |    |    |    | 004E     | 172 | BRW    | EXIT ;BRANCH IF SUCCESSFUL, ALL DONE                                |   |
|    |      |    |    |    |    |    |    | 0051     | 173 | 35\$:  | IFNOWRT   | #8,@INADR(AP),40\$ ;UNLESS INPUT RANGE IS READ ONLY     |
|    |      |    |    |    |    |    |    | 0051     |     |        | PROBEW  | #0,#8,@INADR(AP)  |
|    |      |    |    |    |    |    |    | 0056     |     |        | BEQL  | 40\$  |
|    |      |    |    |    |    |    |    | 0058     | 174 | MOVQ   | R2,@INADR(AP) ;RESTORE INPUT ADDRESS RANGE                          |   |
|    |      |    |    |    |    |    |    | 005C     | 175 | 40\$:  | CMPL  | (AP),#<CHANADR/4> ;WAS CHANADR SPECIFIED?               |
|    |      |    |    |    |    |    |    | 005F     | 176 |        | BLSS  | 50\$ ;BR IF NOT SPECIFIED                               |
|    |      |    |    |    |    |    |    | 0061     | 177 |        | IFNGRD  | #4,@CHANADR(AP),50\$ ;CAN THE LOCATION BE READ?         |
|    |      |    |    |    |    |    |    | 0068     | 178 |        | PROBER  | #0,#4,@CHANADR(AP)                                      |
|    |      |    |    |    |    |    |    | 0066     |     |        | BEQL  | 50\$  |
|    |      |    |    |    |    |    |    | 0068     | 178 | MOVL   | @CHANADR(AP),R6 ;GET CHANNEL SPECIFIED BY USER                      |   |
|    |      |    |    |    |    |    |    | 006C     | 179 | BNEQ   | 100\$ ;DON'T OPEN FILE IF CHAN PROVIDED                             |   |
|    |      |    |    |    |    |    |    | 006E     | 180 | 50\$:  | MOVAL   | FAB(FP),R5 ;ADDRESS OF FAB                              |
|    |      |    |    |    |    |    |    | 0072     | 181 |        |   |   |
|    |      |    |    |    |    |    |    | 0072     | 182 | ASSUME | FAB\$B BLN EQ FAB\$B BID+1 ;ASSUME THESE ARE ADJACENT               |   |
|    |      |    |    |    |    |    |    | 0072     | 183 | MOVW   | #<FAB\$C BLN@B + FAB\$C BID>,FAB\$B BID(R5) ;SET ID AND SIZE OF FAB |   |
|    |      |    |    |    |    |    |    | 0077     | 184 | MOVL   | FILNAM(XP),RO ;GET ADDRESS OF FILNAM DESC                           |   |
|    |      |    |    |    |    |    |    | 007B     | 185 | BEQL   | 100\$ ;BR ON SECTION NOT MAPPED TO A FILE                           |   |
|    |      |    |    |    |    |    |    | 007D     | 186 | MOVQ   | @FILNAM(AP),RO ;GET SIZE AND ADDRESS OF FILE NAME STRING            |   |
|    |      |    |    |    |    |    |    | 0081     | 187 | MOVQ   | RO,FAB\$B_FNS(R5) ;SIZE OF FILE NAME STRING                         |   |

```

2C A5 51 DO 0085 188 MOVL R1,FAB$FNA(R5) ;ADDRESS OF FILE NAME STRING
08 54 03 E1 0089 189 BBC #SECSV_WRT,R4,60$ ;BRANCH IF NOT MAPPING FOR WRITE
04 54 01 E0 008D 190 BBS #SECSV_CRF,R4,60$ ;COPY ON REF IS STILL READ ONLY
16 A5 01 90 0091 191 MOVB #FAB$M_PUT,FAB$B_FAC(R5) ;ACCESS FOR WRITING
00 04 A5 11 E2 0095 192 60$: BBSS #FAB$V_UFO,FAB$FOP(R5),80$ ;SET USER FILE OPEN
65 DF 009A 193 80$: $OPEN (R5) ;OPEN THE FILE
00000000'GF 67 50 E9 00A3 194 PUSHAL (R5)
56 0C A5 DO 00A6 195 CALLS $$,TMP1,G^SYSSOPEN
0A 6C D1 00AA 196 BLBC R0,EXIT ;BRANCH IF FAILED
08 19 00AD 197 MOVL FAB$STV(R5),R6 ;GET CHANNEL FROM FAB
28 BC 04 00 OD 00AF 198 Cmpl (AP),#<CHANADR/4> ;DOES CALLER WANT THE CHANNEL RETURNED?
28 BC 04 00 13 00B4 199 BLSS 100$ ;BR IF DOES NOT WANT CHANNEL
06 54 10 E0 00BA 200 100$: MOVL R6,@CHANADR(AP) ;CHECK CHAN ADR CAN BE WRITTEN
51 14 AC DO 00BE 201 BRB 200$ ;SET RETURN CHANNEL NUMBER
04 51 D4 00C4 202 110$: BRB 200$ ;BRANCH IF PFNMAP-ING
56 D4 00C6 203 110$: CLRL R1 ;GET RELATIVE PAGE TO MAP
00C8 204 110$: CLRL R6 ;GO TRY TO CREATE SECTION
00C8 205 200$: $CRMPSC_S - ;SET REL PAG AS 0
00C8 206 ;SET NO FILE TO MAP TO
00C8 207 ;CREATE AND MAP SECTION
00C8 208 ;INPUT RANGE DESCRIPTOR
00C8 209 ;RETURN RANGE DESCRIPTOR
00C8 210 ;SECTION FLAGS
00C8 211 ;ACCESS MODE OF SECTION
00C8 212 ;GLOBAL SECTION NAME DESCRIPTOR
00C8 213 ;RELATIVE PAGE IN SECTION TO MAP
00C8 214 ;CHANNEL TO FILE
7E 7C 00C8 ;NO. OF SECTION PAGES TO CREATE
20 AC DD 00CA ;STARTING VIRTUAL BLOCK IN SECTION
1C AC DD 00CD CLRQ -(SP)
7E 56 3C 00D0 PUSHL VBN(AP)
51 DD 00D3 PUSHL PAGCNT(AP)
00 DD 00D5 MOVZWL R6,-(SP)
10 BC 7F 00D7 PUSHL R1
54 DD 00DA PUSHL #0
FC AD DD 00DC PUSHAQ @GSDNAM(AP)
08 BC 7F 00DF PUSHL R4
04 BC 7F 00E2 PUSHL SVACMODE(FP)
00000000'GF 0C FB 00E5 PUSHAQ @RETADR(AP)
18 54 00 DD 00EC 215 CALLS #12,G^SYSS$CRMPSC ;SAVE CRMPSC STATUS CODE
51 18 AC DO 00E2 216 BBC #SECSV_GBL,R4,EXIT ;BRANCH IF NOT GLOBAL SECTION
15 13 00F6 217 MOVL FILNAM(AP),R1 ;GET ADR OF FILE NAME DESC
0A 6C D1 00F8 218 BEQL EXIT ;BR ON NO CHANNEL TO DEASSIGN
08 OD 18 00FB 219 Cmpl (AP),#<CHANADR/4> ;DOES CALLER WANT THE CHANNEL RETURNED?
7E 56 3C 00FD 220 BGEQ 300$ ;BR IF DOES WANT CHANNEL
00000000'GF 03 50 E9 0107 221 $DASSGN_S CHAN=R6 ;DEASSIGN THE CHANNEL FOR GLOBAL SECTION
03 50 8ED0 010A 222 MOVZWL R6,-(SP)
50 04 010D 223 300$: CALLS #1,G^SYSS$DASSGN ;RETURN DEASSIGN CHANNEL ERROR
010E 224 EXIT: RET ;RETURN CRMPSC STATUS CODE
010E 225 ;AND RETURN TO CALLER
010E 226 .END

```



LIBSCREMAPSEC  
Symbol table

- CREATE AND MAP A SECTION

F 4

16-SEP-1984 02:15:27  
5-SEP-1984 04:38:59

VAX/VMS Macro V04-00  
[VMSLIB.SRC]CREMAPSEC.MAR;1

Page 6  
(4)

```

$$TMP1          = 00000001
$$TMP2          = 00000065
$$I1            = 00000000
ACMODE          = 00000024
CHANADR         = 00000028
DIR...         = FFFFFFFF
EXIT            = 000010D  R    02
FAB             = FFFFFFFAC
FABS$B_BID      = 00000000
FABS$B_BLN      = 00000001
FABS$B_FAC      = 00000016
FABS$B_FNS      = 00000034
FABS$C_BID      = 00000003
FABS$C_BLN      = 00000050
FABS$L_FNA      = 0000002C
FABS$L_FOP      = 00000004
FABS$L_STV      = 0000000C
FABS$M_PUT      = 00000001
FABS$V_UFO      = 00000011
FILNAM         = 00000018
FLAGS          = 0000000C
GSDNAM         = 00000010
INADR          = 00000004
LIBSCREMAPSEC  = 00000000  RG    02
LIB$ CREMAPSEC = 00000004  RG    02
PAGCNT         = 0000001C
RELPAG         = 00000014
RETADR         = 00000008
SAVABS...      = FFFFFFFAC
SCRATCHSIZE    = FFFFFFFAC
SECSV_CRF      = 00000001
SECSV_GBL      = 00000000
SECSV_PFNMAP   = 00000010
SECSV_WRT      = 00000003
START          = 00000006  R    02
SVACMODE       = FFFFFFFFC
SYS$CRMPSC     = *****  GX    02
SYS$DASSGN     = *****  GX    02
SYS$MGBLSC     = *****  GX    02
SYS$OPEN       = *****  GX    02
VBN            = 00000020
    
```

-----  
! Psect synopsis !  
-----

| PSECT name | Allocation       | PSECT No. | Attributes  |
|------------|------------------|-----------|---|
| . ABS .    | 00000000 ( 0.)   | 00 ( 0.)  | NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE |
| \$ABS\$    | FFFFFFFFC ( 0.)  | 01 ( 1.)  | NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE       |
| _LIB\$CODE | 0000010E ( 270.) | 02 ( 2.)  | PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC BYTE         |

-----  
! Performance indicators !  
-----

| Phase                  | Page faults | CPU Time    | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization         | 48          | 00:00:00.11 | 00:00:00.62  |
| Command processing     | 159         | 00:00:00.83 | 00:00:05.86  |
| Pass 1                 | 206         | 00:00:04.94 | 00:00:14.24  |
| Symbol table sort      | 0           | 00:00:00.41 | 00:00:00.82  |
| Pass 2                 | 53          | 00:00:01.02 | 00:00:02.45  |
| Symbol table output    | 5           | 00:00:00.06 | 00:00:00.06  |
| Psect synopsis output  | 2           | 00:00:00.03 | 00:00:00.03  |
| Cross-reference output | 0           | 00:00:00.00 | 00:00:00.00  |
| Assembler run totals   | 476         | 00:00:07.40 | 00:00:24.08  |

The working set limit was 1350 pages.  
28075 bytes (55 pages) of virtual memory were used to buffer the intermediate code.  
There were 20 pages of symbol table space allocated to hold 373 non-local and 11 local symbols.  
226 source lines were read in Pass 1, producing 16 object records in Pass 2.  
26 pages of virtual memory were used to define 23 macros.

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

| Macro library name                  | Macros defined |
|-------------------------------------|----------------|
| _\$255\$DUA28:[SYS.OBJ]LIB.MLB;1    | 2              |
| -\$255\$DUA28:[SYSLIB]STARLET.MLB;2 | 18             |
| TOTALS (all libraries)              | 20             |

590 GETS were required to define 20 macros.

There were no errors, warnings or information messages.

MACRO/DISA=TRACE/LIS=LISS:CREMAPSEC/OBJ=OBJ\$:CREMAPSEC MSRC\$:CREMAPSEC/UPDATE=(ENHS:CREMAPSEC)+EXECMLS/LIB

