```
RRRRRRRRRRR      MMM          MMM      SSSSSSSSSSS
RRRRRRRRRRRR     MMM          MMM      SSSSSSSSSSSS
RRRRRRRRRRRR     MMM          MMM      SSSSSSSSSSSS
RRR          RRR MMMMMMM    MMMMMMM   SSS
RRR          RRR MMMMMMM    MMMMMMM   SSS
RRR          RRR MMMMMMM    MMMMMMM   SSS
RRR          RRR MMM   MMM   MMM   MMM SSS
RRR          RRR MMM   MMM   MMM   MMM SSS
RRR          RRR MMM   MMM   MMM   MMM SSS
RRRRRRRRRRR      MMM          MMM      SSSSSSSSS
RRRRRRRRRRRR     MMM          MMM      SSSSSSSSS
RRRRRRRRRRRR     MMM          MMM      SSSSSSSSS
RRR    RRR       MMM          MMM               SSS
RRR    RRR       MMM          MMM               SSS
RRR    RRR       MMM          MMM               SSS
RRR       RRR    MMM          MMM               SSS
RRR       RRR    MMM          MMM               SSS
RRR       RRR    MMM          MMM               SSS
RRR          RRR MMM          MMM  SSSSSSSSSSS
RRR       RRR    MMM          MMM               SSS
RRR          RRR MMM          MMM  SSSSSSSSSSS
RRR          RRR MMM          MMM  SSSSSSSSSSS
RRR          RRR MMM          MMM  SSSSSSSSSSS
```

```
NN       NN  TTTTTTTTTT    000000    MM       MM    AAAAAA      CCCCCCCC  RRRRRRRR      000000      SSSSSSSS
NN       NN  TTTTTTTTTT    000000    MM       MM    AAAAAA      CCCCCCCC  RRRRRRRR      000000      SSSSSSSS
NN       NN      TT       00    00  MMMM   MMMM   AA    AA    CC          RR      RR  00    00  SS
NN       NN      TT       00    00  MMMM   MMMM   AA    AA    CC          RR      RR  00    00  SS
NNNN     NN      TT       00    00  MM MM MM MM   AA    AA    CC          RR      RR  00    00  SS
NNNN     NN      TT       00    00  MM MM MM MM   AA    AA    CC          RR      RR  00    00  SS
NN NN    NN      TT       00  00 00  MM    MM MM   AA    AA    CC          RRRRRRRR    00    00    SSSSSS
NN NN    NN      TT       00  00 00  MM    MM MM   AA    AA    CC          RRRRRRRR    00    00    SSSSSS
NN   NNNN        TT       0000   00  MM    MM MM   AAAAAAAAAA  CC          RR   RR     00    00          SS
NN   NNNN        TT       0000   00  MM    MM MM   AAAAAAAAAA  CC          RR   RR     00    00          SS
NN     NN        TT       00     00  MM    MM MM   AA    AA    CC          RR    RR    00    00          SS      ....
NN     NN        TT       00     00  MM    MM MM   AA    AA    CC          RR    RR    00    00          SS      ....
NN     NN        TT         000000    MM    MM MM   AA    AA      CCCCCCCC  RR     RR     000000    SSSSSSSS     ....
NN     NN        TT         000000    MM    MM MM   AA    AA      CCCCCCCC  RR     RR     000000    SSSSSSSS     ....

MM       MM     AAAAAA     RRRRRRRR
MM       MM     AAAAAA     RRRRRRRR
MMMM   MMMM   AA    AA    RR      RR
MMMM   MMMM   AA    AA    RR      RR
MM MM MM MM   AA    AA    RR      RR
MM MM MM MM   AA    AA    RR      RR
MM    MM MM   AA    AA    RRRRRRRR
MM    MM MM   AA    AA    RRRRRRRR
MM    MM MM   AAAAAAAAAA  RR   RR
MM    MM MM   AAAAAAAAAA  RR   RR
MM    MM MM   AA    AA    RR    RR
MM    MM MM   AA    AA    RR    RR
MM    MM MM   AA    AA    RR     RR
MM    MM MM   AA    AA    RR     RR
```

```
        .TITLE  NTOMACROS - RMS NETWORK MACRO DEFINITIONS
        .IDENT  'V04-000'

;*****************************************************************************
;*                                                                           *
;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;*  ALL RIGHTS RESERVED.                                                     *
;*                                                                           *
;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
;*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE           *
;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER      *
;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY       *
;*  TRANSFERRED.                                                             *
;*                                                                           *
;*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE     *
;*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT         *
;*  CORPORATION.                                                             *
;*                                                                           *
;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS      *
;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
;*                                                                           *
;*                                                                           *
;*****************************************************************************


;++
; Facility: RMS
;
; Abstract:
;
;       This module contains MACRO definitions used by RMS network modules.
;
; Environment: VAX/VMS, executive mode
;
; Author: James A. Krycka,      Creation Date:  17-MAY-1978
;
; Modified By:
;
;       V02-004 REFORMAT        J A Krycka      26-JUL-1980
;--
```

```
        .SBTTL  CODE GENERATION MACROS

;++
; $SETBIT sets a single bit in a field.
;--

        .MACRO  $SETBIT POS,BASE,?DISPL
        BBSS    POS,BASE,DISPL
DISPL:
        .ENDM   $SETBIT

;++
; $CLRBIT clears a single bit in a field.
;--

        .MACRO  $CLRBIT POS,BASE,?DISPL
        BBCC    POS,BASE,DISPL
DISPL:
        .ENDM   $CLRBIT

;++
; $MAPBIT maps the designated bit from R1 into the designated bit in R2.
; The bit is set in R2 only if the corresponding bit is set in R1.
;--

        .MACRO  $MAPBIT SRCBIT,DSTBIT,?LABEL
        BBC     #SRCBIT,R1,LABEL
        BBCS    #DSTBIT,R2,LABEL
LABEL:
        .ENDM   $MAPBIT

;++
; $ZERO_FILL writes zeroes into the specified buffer. On completion R0-R5 are
; destroyed (with R3 containing the address of one byte beyond the buffer).
; The default is to zero 512 bytes (1 page) at the specified address.
;--

        .MACRO  $ZERO_FILL DST=,SIZE=#512
        MOVC5   #0,DST,#0,SIZE,DST
        .ENDM   $ZERO_FILL

;++
; $CASEB, $CASEW, and $CASEL generate a CASEB, CASEW, CASEL instruction,
; respectively, followed by the case disolacement table. The parameters for
; each macro are:
;
;       SELECTOR = the selector operand
;       BASE     = the base operand
;       (The limit operand is calculated from the # of entries in DISPL.)
;       DISPL    = the case displacement list
;
; Note that these macro definitions place BASE after SELECTOR and DISPL so that
; BASE can be omitted when keywords are not used in the macro invocation.
;--

        .MACRO  $CASEB  SELECTOR,DISPL,BASE=#0
```

```
        $CASE   SELECTOR,<DISPL>,BASE,TYPE=B
        .ENDM   $CASEB

        .MACRO  $CASEW  SELECTOR,DISPL,BASE=#0
        $CASE   SELECTOR,<DISPL>,BASE,TYPE=W
        .ENDM   $CASEW

        .MACRO  $CASEL  SELECTOR,DISPL,BASE=#0
        $CASE   SELECTOR,<DISPL>,BASE,TYPE=L
        .ENDM   $CASEL

;++
; $CASE is a level 2 macro used by $CASEB, $CASEW, and $CASEL. It generates a
; CASE[B/W/L] instruction followed by the case displacement table. The
; parameters for this macro are:
;
;       TYPE     = operand datatype of b, w, or l
;       SELECTOR = the selector operand
;       BASE     = the base operand
;       (The limit operand is calculated from the # of entries in DISPL.)
;       DISPL    = the case displacement list
;
; Note that the macro definition places SELECTOR and DISPL ahead of BASE and
; TYPE so that the latter can be omitted when keywords are not used in the
; macro invocation.
;--

        .MACRO  $CASE   SELECTOR,DISPL,BASE=#0,TYPE=B,?TABLE
        $$COUNT=0
        .IRP    EP,<DISPL>
        $$COUNT=$$COUNT+1
        .ENDR
        .IF     EQ,$$COUNT
        .ERROR  ; ***** case displacement list is null ***** ;
        .MEXIT
        .ENDC
        CASE'TYPE       SELECTOR,BASE,#<$$COUNT-1>
TABLE:
        .IRP    EP,<DISPL>
        .WORD   EP-TABLE
        .ENDR
        .ENDM   $CASE

        .END                            ; End of module
```

RMSCALLS
MAR

RMSIDXLNK
R32

RMS
LST

RMS32MAC
MAR

RMSMSGMAC
MAR

UTLDEF
R32

RMSIDXMAC
R32

UTLDEFUND
R32

RMSINTDEF
LST

RMSIDXDEF
R32

NT@MACROS
MAR