M 8

```
DDDDDDDDDDDD     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  DDDDDDDDDDDD     TTTTTTTTTTTTTTT  RRRRRRRRRRRR
DDDDDDDDDDDD     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  DDDDDDDDDDDD     TTTTTTTTTTTTTTT  RRRRRRRRRRRR
DDDDDDDDDDDD     TTTTTTTTTTTTTTT     SSSSSSSSSSSS  DDDDDDDDDDDD     TTTTTTTTTTTTTTT  RRRRRRRRRRRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT        SSS              DDD         DDD        TTT        RRR         RRR
DDD         DDD        TTT           SSSSSSSSS     DDD         DDD        TTT        RRRRRRRRRRRR
DDD         DDD        TTT           SSSSSSSSS     DDD         DDD        TTT        RRRRRRRRRRRR
DDD         DDD        TTT           SSSSSSSSS     DDD         DDD        TTT        RRRRRRRRRRRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR   RRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR   RRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR   RRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR      RRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR      RRR
DDD         DDD        TTT                    SSS  DDD         DDD        TTT        RRR      RRR
DDDDDDDDDDDD        TTT        SSSSSSSSSSSS     DDDDDDDDDDDD        TTT        RRR         RRR
DDDDDDDDDDDD        TTT        SSSSSSSSSSSS     DDDDDDDDDDDD        TTT        RRR         RRR
DDDDDDDDDDDD        TTT        SSSSSSSSSSSS     DDDDDDDDDDDD        TTT        RRR         RRR
```

```
_$

Vi
St
im
Im
Im
Nu
Nu
Nu
Nu
Nu
Nu
Us
Nu
Im
Ma
Es

Pe
--

To

Us

To

Nu

17

A

LI
DT
```

```
**FILE**ID**DTMACROS
```

```
DDDDDDDD    TTTTTTTTTT  MM      MM    AAAAAA      CCCCCCCC  RRRRRRRR      OOOOOO      SSSSSSSS
DDDDDDDD    TTTTTTTTTT  MM      MM    AAAAAA      CCCCCCCC  RRRRRRRR      OOOOOO      SSSSSSSS
DD      DD      TT      MMMM  MMMM  AA      AA  CC          RR      RR  OO      OO  SS
DD      DD      TT      MMMM  MMMM  AA      AA  CC          RR      RR  OO      OO  SS
DD      DD      TT      MM  MM  MM  AA      AA  CC          RR      RR  OO      OO  SS
DD      DD      TT      MM  MM  MM  AA      AA  CC          RR      RR  OO      OO  SS
DD      DD      TT      MM      MM  AA      AA  CC          RRRRRRRR    OO      OO    SSSSSS
DD      DD      TT      MM      MM  AA      AA  CC          RRRRRRRR    OO      OO    SSSSSS
DD      DD      TT      MM      MM  AAAAAAAAAA  CC          RR  RR      OO      OO          SS
DD      DD      TT      MM      MM  AAAAAAAAAA  CC          RR  RR      OO      OO          SS
DD      DD      TT      MM      MM  AA      AA  CC          RR    RR    OO      OO          SS      ....
DD      DD      TT      MM      MM  AA      AA  CC          RR    RR    OO      OO          SS      ....
DDDDDDDD        TT      MM      MM  AA      AA    CCCCCCCC  RR      RR    OOOOOO    SSSSSSSS      ....
DDDDDDDD        TT      MM      MM  AA      AA    CCCCCCCC  RR      RR    OOOOOO    SSSSSSSS      ....
```

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

```
        .TITLE  TSTSDTMACROS - MACRO DEFINITIONS FOR DTS/DTR
        .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: DTS/DTR DECNET TEST PACKAGE
;
; ABSTRACT: MACRO DEFINITIONS USED BY DTS/DTR MODULES.
;
; ENVIRONMENT: DTS/DTR RUN IN USER MODE AND REQUIRE NETWORK PRIVILEGE.
;
; AUTHOR: JAMES A. KRYCKA,      CREATION DATE: 11-AUG-77
;
; MODIFICATIONS:
;
;--
```

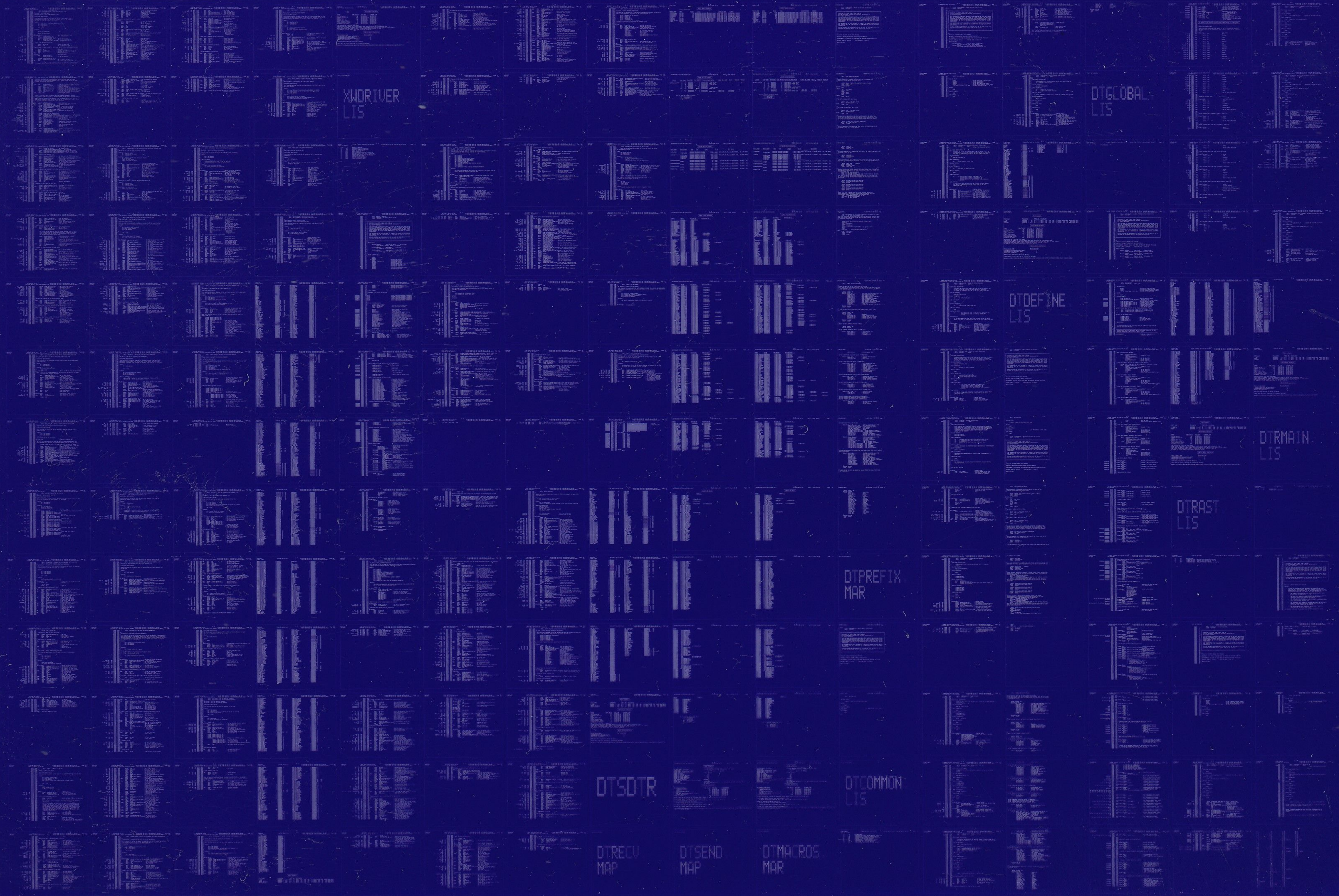
```
        .SBTTL  CODE GENERATION MACROS

;++
; QBLOCK GENERATES A QUADWORD DESCRIPTOR BLOCK FOLLOWED BY THE
; CHARACTER STRING AND/OR ALLOCATED SPACE.
;--

        .MACRO  QBLOCK  TEXT,SPACE=0,BUFADR,?LABEL1,?LABEL2
        .LONG   LABEL2-LABEL1
        .LONG   LABEL1
        .IF NB  BUFADR
BUFADR==.
        .ENDC
LABEL1:
        .IRP    STR,<TEXT>
        .ASCII  \STR\
        .ENDR
        .IF NE  SPACE
        .BLKB   SPACE
        .ENDC
LABEL2:
        .ENDM   QBLOCK

;++
; SSB SETS A SINGLE BIT IN A FIELD.
;--

        .MACRO  SSB     POS,BASE,?DISPL
        BBSS    POS,BASE,DISPL
DISPL:
        .ENDM   SSB

;++
; CSB CLEARS A SINGLE BIT IN A FIELD.
;--

        .MACRO  CSB     POS,BASE,?DISPL
        BBCC    POS,BASE,DISPL
DISPL:
        .ENDM   CSB

;++
; FILLBUF FILLS A BUFFER WITH A SPECIFIED CHARACTER. ON COMPLETION
; R3 CONTAINS THE ADDRESS OF ONE BYTE BEYOND THE FILLED BUFFER. NOTE
; THAT THIS MACRO USES THE MOVC5 INSTRUCTION WHICH DESTROYES R0 - R5!
; THE DEFAULT IS TO ZERO 512 BYTES (1 PAGE) AT THE SPECIFIED ADDRESS.
;--

        .MACRO  FILLBUF DST=,SIZE=#512,CHAR=#^X00
        MOVC5   #0,,CHAR,SIZE,DST
        .ENDM   FILLBUF

;++
; CHECK_SS BRANCHES TO A SUBROUTINE THAT CHECKS THE STATUS CODE IN R0
; FOLLOWING A CALL TO A SYSTEM SERVICE.
;--
```

```
        .MACRO  CHECK_SS
        BSBW    TST$CHECK_SS
        .ENDM   CHECK_SS

;++
; CHECK_RMS BRANCHES TO A SUBROUTINE THAT CHECKS THE COMPLETION CODE IN R0
; FOLLOWING A CALL TO RMS.
;--

        .MACRO  CHECK_RMS
        BSBW    TST$CHECK_RMS
        .ENDM   CHECK_RMS

;++
; CHECK_IOSB BRANCHES TO A SUBROUTINE THAT CHECKS THE STATUS CODE OF THE
; SPECIFIED I/O STATUS BLOCK FOLLOWING A CALL TO THE QIO SYSTEM SERVICE.
;--

        .MACRO  CHECK_IOSB      ADDRESS
        MOVAQ   ADDRESS,R0
        BSBW    TST$CHECK_IOSB
        .ENDM   CHECK_IOSB

;++
; $CASEB, $CASEW, AND $CASEL GENERATE A CASEB, CASEW, CASEL INSTRUCTION,
; RESPECTIVELY, FOLLOWED BY THE CASE DISPLACEMENT 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 THE 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.
; $CASE GENERATES A CASE[B/W/L] INSTRUCTION FOLLOWED BY THE CASE
; DISPLACEMENT TABLE. THE PARAMETERS FOR THE 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
        .SBTTL  SYMBOL DEFINITION MACROS

;++
; EFNDEF DEFINES THE USE OF EVENT FLAGS BY DTS/DTR.
; NOTE: MANY OF THE FLAG VALUES SERVE A DUAL PURPOSE; THEY ARE ALSO USED
; A FUNCTION/INDEX CODES THAT ARE MAPPED INTO THE APPROPRIATE QIO REQUEST
; SYSTEM SERVICE CALLS.
;--

        .MACRO  EFNDEF  GBL
        $DEFINI EFN,GBL
        $EQULST EFN_K_,GBL,,,<-         ; EFN [AND FUNCTION/INDEX CODE] FOR:
                <READ_MAIL,0>-          ; READ ASSOCIATED MAILBOX
                <CONN_INIT,1>-          ; NSP CONNECT INITIATE
                <CONN_ACCE,1>-          ; NSP CONNECT ACCEPT (CONFIRM)
                <CONN_REJE,2>-          ; NSP CONNECT REJECT
                <DISC_SYNC,3>-          ; NSP SYNCHRONOUS DISCONNECT
                <DISC_ABRT,4>-          ; NSP DISCONNECT ABORT
                <XMIT_DATA,5>-          ; NSP TRANSMIT DATA MESSAGE
                <XMIT_INTE,6>-          ; NSP TRANSMIT INTERRUPT MESSAGE
                <RECV_DATA,7>-          ; NSP RECEIVE DATA MESSAGE
                <TIMER,8>-              ; TIMER AST
                <SIGNAL,9>-             ; SIGNALLING AN EVENT FROM AN AST
        >                               ;
        $DEFEND EFN,GBL
        .ENDM   EFNDEF

;++
; FLGDEF DEFINES OFFSETS AND MASKS FOR COMMAND PARSE STATUS FLAGS.
;--

        .MACRO  FLGDEF  GBL
        $DEFINI FLG,GBL
        _VIELD  FLG,0,<-                ; MEANING:
                <PARSERROR,,M>-         ; PARSE ERROR DETECTED
                <MULTILINE,,M>-         ; COMMAND LINE IS CONTINUED
                <PARAMETER,,M>-         ; COMMAND PARAMETER FOUND
                <DELIMITER,,M>-         ; COMMAND DELIMITER FOUND
        >                               ;
        $DEFEND FLG,GBL
        .ENDM   FLGDEF

;++
; CMDDEF DEFINES COMMAND LANGUAGE SYMBOLS.
;--

        .MACRO  CMDDEF  GBL
        $DEFINI CMD,GBL
;
; DEFINE COMMAND PARAMETER VALUES (TST$GB_TEST).
;
        $EQULST VAL_K_,GBL,,,<-         ; TEST FUNCTION CODE:
                <TEST_CONN,0>-          ; CONNECT TEST
                <TEST_DATA,1>-          ; DATA TEST
                <TEST_DISC,2>-          ; DISCONNECT TEST
                <TEST_INTE,3>-          ; INTERRUPT TEST
                <TEST_MISC,4>-          ; MISCELLANEOUS TEST
        >                               ;
;
; DEFINE /[NO]PRINT QUALIFIER VALUES (TST$GB_PRINT).
;
        $EQULST VAL_K_,GBL,,,<-         ; FUNCTION MODIFIER CODE:
                <PRIN_NO,0>-            ; NOPRINT
                <PRIN_YES,128>-         ; PRINT (BIT7 = 1)
        >                               ;
;
; DEFINE /TYPE QUALIFIER VALUES (TST$GB_TYPE).
;
        $EQULST VAL_K_,GBL,,,<-         ; TEST SUBFUNCTION CODE:
                <TYPE_REJE,0>-          ; CONNECT REJECT
                <TYPE_ACCE,1>-          ; CONNECT ACCEPT (CONFIRM)
                <TYPE_SINK,0>-          ; SINK (NO CHECKING)
                <TYPE_SEQU,1>-          ; SEQUENCE CHECK
                <TYPE_PATT,2>-          ; SEQUENCE AND PATTERN CHECK
                <TYPE_ECHO,3>-          ; ECHO MESSAGE
                <TYPE_SYNC,0>-          ; SYNCHRONOUS DISCONNECT
                <TYPE_ABRT,1>-          ; DISCONNECT ABORT
                <TYPE_NAME,0>-          ; INVALID NODENAME
        >                               ;
;
; DEFINE /[NO]RETURN QUALIFIER VALUES (TST$GB_RETURN).
;
        $EQULST VAL_K_,GBL,,,<-         ; SUBFUNCTION MODIFIER CODE:
                <RETU_NO,0>-            ; NORETURN USERDATA
                <RETU_STAN,2>-          ; RETURN STANDARD USERDATA
                <RETU_RECE,4>-          ; RETURN RECEIVED USERDATA
        >                               ;
;
; DEFINE /[NO]FLOW QUALIFIER VALUES (TST$GB_FLOW).
;
        $EQULST VAL_K_,GBL,,,<-         ; FLOW CONTROL VALUE:
                <FLOW_NO,0>-            ; NOFLOW CONTROL
                <FLOW_SEGM,1>-          ; SEGMENT FLOW CONTROL
                <FLOW_MESS,2>-          ; MESSAGE FLOW CONTROL
        >                               ;
;
; DEFINE /[NO]STATISTICS QUALIFIER VALUES (TST$GB_STAT).
;
        $EQULST VAL_K_,GBL,,,<-         ; STATISTICS VALUE:
                <STAT_NO,0>-            ; NOSTATISTICS
                <STAT_YES,1>-           ; STATISTICS
        >                               ;
;
; DEFINE /[NO]BACK QUALIFIER VALUES (TST$GB_BACK).
; DEFINE /[NO]DISPLAY QUALIFIER VALUES (TST$GB_DISPLAY).
; DEFINE /[NO]NAK QUALIFIER VALUES (TST$GB_NAK).
; EACH OF THESE ALSO TAKE EXPLICIT NUMERIC VALUES.
;
        $EQULST VAL_K_,GBL,,,<-         ;
                <BACK_NO,0>-            ; NO BACK PRESSURE CONTROL
                <DISP_NO,0>-            ; NO DISPLAY
                <NAK_NO,0>-             ; NO NAK CONTROL
        >                               ;
;
; DEFINE DEFAULT QUALIFIER VALUES.
;
        $EQULST DFT_K_,GBL,,,<-         ; DEFAULT QUALIFIER VALUE FOR:
                <BACK,VAL_K_BACK_NO>-   ; BACK PRESSURE CONTROL
                <DISPLAY,VAL_K_DISP_NO>- ; DISPLAY SIZE IN BYTES
                <FLOW,VAL_K_FLOW_MESS>- ; FLOW CONTROL
                <NAK,VAL_K_NAK_NO>-     ; NAK CONTROL
                <PRINT,VAL_K_PRIN_NO>-  ; PRINT
                <RETURN_CO,VAL_K_RETU_NO>- ; RETURN USERDATA (CONNECT)
                <RETURN_DI,VAL_K_RETU_NO>- ; RETURN USERDATA (DISCONNECT)
                <RQUEUE_DA,1>-          ; DTR QUEUE (DATA)
                <RQUEUE_IN,1>-          ; DTR QUEUE (INTERRUPT)
                <SIZE_DA,128>-          ; MESSAGE SIZE IN BYTES (DATA)
                <SIZE_IN,16>-           ; MESSAGE SIZE IN BYTES (INTERRUPT)
        >                               ;
        $EQULST DFT_K_,GBL,,,<-         ; DEFAULT QUALIFIER VALUE FOR:
                <SPEED,1000000>-        ; LINE SPEED IN BAUD
                <SQUEUE_DA,1>-          ; DTS QUEUE (DATA)
                <SQUEUE_IN,1>-          ; DTS QUEUE (INTERRUPT)
                <STAT,VAL_K_STAT_YES>-  ; STATISTICS
                <TIME_DA,030>-          ; TIME IN SECONDS (DATA)
                <TIME_IN,030>-          ; TIME IN SECONDS (INTERRUPT)
                <TYPE_CO,VAL_K_TYPE_ACCE>- ; TYPE (CONNECT)
                <TYPE_DA,VAL_K_TYPE_SINK>- ; TYPE (DATA)
                <TYPE_DI,VAL_K_TYPE_ABRT>- ; TYPE (DISCONNECT)
                <TYPE_IN,VAL_K_TYPE_SINK>- ; TYPE (INTERRUPT)
                <TYPE_MI,VAL_K_TYPE_NAME>- ; TYPE (MISCELLANEOUS)
        >                               ;
;
; DEFINE MAXIMUM QUALIFIER VALUES FOR THOSE QUALIFIERS THAT ACCEPT NUMERIC
; QUALIFIER VALUES.
;
        $EQULST MAX_K_,GBL,,,<-         ; MAXIMUM QUALIFIER VALUE FOR:
                <BACK,128>-             ; BACK PRESSURE CONTROL
                <DISPLAY,38>-           ; DISPLAY SIZE IN BYTES
                <NAK,128>-              ; NAK CONTROL
                <RQUEUE_DA,8>-          ; DTR QUEUE (DATA)
                <RQUEUE_IN,8>-          ; DTR QUEUE (INTERRUPT)
                <SIZE_DA,4096>-         ; MESSAGE SIZE IN BYTES (DATA)
                <SIZE_IN,16>-           ; MESSAGE SIZE IN BYTES (INTERRUPT)
                <SPEED,1000000>-        ; LINE SPEED IN BAUD
                <SQUEUE_DA,8>-          ; DTS QUEUE (DATA)
                <SQUEUE_IN,8>-          ; DTS QUEUE (INTERRUPT)
                <TIME_DA,360000>-       ; TIME IN SECONDS (DATA)
                <TIME_IN,360000>-       ; TIME IN SECONDS (INTERRUPT)
        >                               ;
        $DEFEND CMD,GBL
        .ENDM   CMDDEF

;++
; VLDDEF DEFINE OFFSETS AND MASKS FOR VALID (PERMITTED) QUALIFIER FLAGS
; IN TST$GL_VALID.
;--
```

```
        .MACRO  VLDDEF  GBL
        $DEFINI VLD,GBL
        _VIELD  VLD,0,<-                        ; QUALIFIER:
                <BACK,,M>-                      ; BACK
                <DISPLAY,,M>-                   ; DISPLAY
                <FLOW,,M>-                      ; FLOW
                <HOURS,,M>-                     ; HOURS
                <MINUTES,,M>-                   ; MINUTES
                <NAK,,M>-                       ; NAK
                <NOBACK,,M>-                    ; NOBACK
                <NODENAME,,M>-                  ; NODENAME
                <NODISPLAY,,M>-                 ; NODISPLAY
                <NOFLOW,,M>-                    ; NOFLOW
                <NONAK,,M>-                     ; NONAK
                <NOPRINT,,M>-                   ; NOPRINT
                <NORETURN,,M>-                  ; NORETURN
                <NOSTAT,,M>-                    ; NOSTATISTICS
                <PRINT,,M>-                     ; PRINT
                <RETURN,,M>-                    ; RETURN
                <RQUEUE,,M>-                    ; DTR QUEUE
                <SECONDS,,M>-                   ; SECONDS
                <SIZE,,M>-                      ; SIZE
                <SPEED,,M>-                     ; SPEED
                <SQUEUE,,M>-                    ; DTS QUEUE
                <STAT,,M>-                      ; STATISTICS
                <TYPE,,M>-                      ; TYPE
        >                                       ;
        $DEFEND VLD,GBL
        .ENDM   VLDDEF
        .END
```

0122 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



XWDRIVER
LIS

DTGLOBAL
LIS

DTDEFINE
LIS

DTRMAIN
LIS

DTRAST
LIS

DTPREFIX
MAR

DTSDTR

DTCOMMON
LIS

DTRECV
MAP

DTSEND
MAP

DTMACROS
MAR