

```

EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AAAAAA MM MM PPPPPPPP LL EEEEEEEEEE SSSSSSSS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MMMM MMMM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EE XX XX AA AA MM MM MM PP PP LL EE SS
EEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEE SSSSSS
EEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEE SSSSSS
EEEEEEEEE XX XX AA AA MM MM PPPPPPPP LL EEEEEEEEE SSSSSS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AAAAAAAAAA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EE XX XX AA AA MM MM PP LL EE SS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEE SSSSSSSS
EEEEEEEEEE XX XX AA AA MM MM PP LLLLLLLLLL EEEEEEEEE SSSSSSSS

```

```

XX      XX      AAAAAA      MM      MM      EEEEEEEEEE      SSSSSSSS      SSSSSSSS      AAAAAA      GGGGGGGG      EEEEEEEEEE
XX      XX      AAAAAA      MM      MM      EEEEEEEEEE      SSSSSSSS      SSSSSSSS      AAAAAA      GGGGGGGG      EEEEEEEEEE
XX      XX      AA      AA      MMMM      MMMM      EE      SS      SS      AA      AA      GG      GG      EE
XX      XX      AA      AA      MMMM      MMMM      EE      SS      SS      AA      AA      GG      GG      EE
  XX      XX      AA      AA      MM      MM      EE      SS      SS      AA      AA      GG      GG      EE
  XX      XX      AA      AA      MM      MM      EE      SS      SS      AA      AA      GG      GG      EE
    XX      XX      AA      AA      MM      MM      EEEEEEEEEE      SSSSSS      SSSSSS      AA      AA      GG      GG      EEEEEEEEE
    XX      XX      AA      AA      MM      MM      EEEEEEEEEE      SSSSSS      SSSSSS      AA      AA      GG      GG      EEEEEEEEE
      XX      XX      AAAAAAAAAA      MM      MM      EE      SS      SS      AAAAAAAAAA      GG      GGGGGG      EE
      XX      XX      AAAAAAAAAA      MM      MM      EE      SS      SS      AAAAAAAAAA      GG      GGGGGG      EE
XX      XX      AA      AA      MM      MM      EE      SS      SS      AA      AA      GG      GG      EE
XX      XX      AA      AA      MM      MM      EE      SS      SS      AA      AA      GG      GG      EE
XX      XX      AA      AA      MM      MM      EEEEEEEEEE      SSSSSSSS      SSSSSSSS      AA      AA      GGGGGG      EEEEEEEEEE
XX      XX      AA      AA      MM      MM      EEEEEEEEEE      SSSSSSSS      SSSSSSSS      AA      AA      GGGGGG      EEEEEEEEEE

```

```

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 XMESSAGE
.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.
*****

```

.SBTTL LOCAL DEFFINITIONS AND STORAGE

ROUTINE TO TRANSFER A MESSAGE IN LINK MODE BETWEEN TWO DR11-W'S

CALLING SEQUENCE:

CALL (BUFFER_ADDRESS,BUFFER_SIZE,TRANSFER_DIRECTION,CHANNEL,EVENT_FLAG,TIME_OUT,STATUS)

```

BUFFER_ADDRESS = ADDRESS OF DATA BUFFER TO TRANSFER
BUFFER_SIZE = SIZE IN BYTES OF DATA BUFFER TO TRANSFER
TRANSFER_DIRECTION = DIRECTION FOR DATA TO GO
0 = TRANSMITT
1 = RECEIVE
CHANNEL = CHANNEL ASSIGNED TO DR11-W
EVENT_FLAG = EVENT FLAG TO SET WHEN TRANSFER COMPLETE
TIME_OUT = REQUEST TIME OUT VALUE TO USE
STATUS = ADDRESS OF 20 BYTE ARRAY TO RECEIVE FINAL STATUS
IOSB - 8 BYTES
I/O STATUS BLOCK FROM QUEUE I/O REQUEST
ERROR - 4 BYTES
INTERNAL ERROR CODE GENERATED BY THIS ROUTINE
-01 CANNOT ENABLE ATTN AST'S
-02 UNSOLICITED INTERRUPT RECEIVED ON TRANSMITT
-03 QIO ERROR FROM WORD MODE WRITE
-04 QIO ERROR FROM BLOCK MODE WRITE
-05 INCONSISTENT STATE, TRANSMITT ATTN AST
-06 INCONSISTENT STATE, RECEIVE ATTN AST
-09 INCONSISTENT STATE, RECEIVE PROCESSING
-10 QIO ERROR, BLOCK MODE READ
-11 INCONSISTENT STATE, TRANSMITT QIO AST

```

```

-12 ERROR FROM WORD MODE WRITE
-13 ERROR FROM BLOCK MODE WRITE
-14 INCONSISTENT STATE, RECEIVE QIO AST
-15 ERROR FROM BLOCK MODE READ
STATE - 4 BYTES
STATE VARIABLE USED FOR INTERNAL DISPATCHING
00 INITIAL STATE
01 WORD MODE WRITE MESSAGE SENT
02 UNSOLICITED INPUT FROM BUDDY DURING XMIT
03 BLOCK MODE WRITE MESSAGE SENT
04 WAITING FOR BUDDY, SOLICITED RECIEVE
05 SOLICITED BLOCK MODE RECEIVE
06 UNSOLICITED BLOCK MODE RECEIVE
DSW - 4 BYTES
SYSTEM SERVICE STATUS
VALUE OF R0 RETURNED FROM SYSTEM SERVICES

```

```

: LOCAL DATA

```

```

ADDRESS=4
SIZE=8
DIRECTION=12
CHANNEL=16
EFN=20
TIME=24
STATUS=28

```

```

.PSECT XADATA, LONG

```

```

BUFFER: .LONG 0 ; SAVED BUFFER ADDRESS
LENGTH: .LONG 0 ; SAVED BUFFER LENGTH
XAFUNC: .LONG 0 ; CALCULATED DR11-W CSR FNCT BITS
CHAN: .LONG 0 ; SAVED CHANNEL ASSIGNED TO DR11-W
TIMER: .LONG 0 ; SAVED TIME-OUT VALUE
EVENTFLG: .LONG 0 ; SAVED EVENT FLAG NUMBER
IOSB1: .QUAD 0 ; IOSB FOR WORD MODE WRITE

```

```

: NOTE - ORDER IS ASSUMED FOR NEXT FOUR VARIABLES

```

```

IOSB: .QUAD 0 ; QIO IOSB
ERROR: .LONG 0 ; ERROR VALUE PARAMATER
STATE: .LONG 0 ; STATE VARIABLE
DSW: .LONG 0 ; SYSTEM SERVICE STATUS

```

```

:
RETURN_STATUS:

```

```

ZERO: .LONG 0 ; ADDRESS OF CALLERS STATUS VARIABLE
ZERO: .WORD 0 ; ZERO CONSTANT FOR ERROR REPORTING

```

```

.PAGE
.SBTTL ENTRY AND PARAMATER PROCESSING

```

```

: MAIN PROGRAM

```

```
.PSECT XACODE,NOWRT
.ENTRY XAMESSAGE,^M<R2,R3,R4,R5>
```

```
: GATHER CALL PARAMATERS
```

```

      CLRQ    W^ERROR                : CLEAR ERROR FLAG AND STATE VARIABLE
      CMPW    (AP),#7                : MUST PASS 7 PARAMATERS
      BNEQ    ERROR1                 : IF NOT, THEN ERROR
      MOVL    ADDRESS(AP),W^BUFFER    : GET BUFFER ADDRESS
      MOVL    @SIZE(AP),W^LENGTH      : GET BUFFER SIZE
      BEQL    ERROR1                 : XFER SIZE MUST BE NON ZERO
      MOVZBL  @DIRECTION(AP),W^XAFUNC : GET TRANSFER DIRECTION FLAG
      MOVL    @CHANNEL(AP),W^CHAN     : FETCH CHANNEL
      MOVL    @EFN(AP),W^EVENTFLG    : AND EVENT FLAG
      BEQL    ERROR1                 : MUST SPECIFY EVENT FLAG
      $CLREF_S EFN=W^EVENTFLG        : ELSE, MAKE SURE IT'S CLEAR
20$:  BLBC    R0,ERROR1               : CHECK FOR SERVICE ERROR
      MOVL    @TIME(AP),W^TIMER       : FETCH TIME-OUT VALUE
      BNEQ    30$                    : CHECK FOR ZERO
30$:  MOVZBL  #5,W^TIMER              : USER SOME "REASONABLE" VALUE
      MOVL    STATUS(AP),W^RETURN_STATUS : GET ADDRESS OF STATUS ARRAY
      BEQL    ERROR1                 : IF NOT SPECIFIED, ERROR
      CLRL   @W^RETURN_STATUS        : INITIALIZE STATUS VALUE
      BRB    START

ERROR1: MOVZWL #SS$_BADPARAM,R0      : ERROR IN CALL
      RET

      .PAGE
      .SBTTL START MESSAGE PROCESSOR
      .ENABL LSB

START:                                : ENABLE AST'S, REST DONE AT AST LEVEL

      $QIO_S CHAN=W^CHAN,-           : QIO TO ENABLE AST'S
          FUNC=#<IOS$ SETMODE!IOSM_ATTNAST>,-
          IOSB=W^IOSB,-
          P1=W^ATTN_AST

      BLBS   R0,10$                  : BRANCH IF DONE OK
      CVTBL  #-1,W^ERROR              : ERROR - CANNOT ENABLE ATTN AST'S
      MOVL   R0,W^DSW                 : SAVE SYSTEM SERVICE STATUS

RETURN_ERROR:                          : RETURN ERROR STATUS TO CALLER
      $CANCEL_S CHAN=W^CHAN           : CANCEL ALL I/O AND FLUSH ATTN AST'S
      BRB    5$

RETURN_SUCCESS:                         : RETURN FINAL STATUS TO CALLER
      MOVL   R0,W^DSW                 : SAVE SYSTEM SERVICE STATUS
5$:  $SETEF_S EFN=W^EVENTFLG         : SET CALLER'S EVENT FLAG
      MOVCS  #20,W^IOSB,@W^RETURN_STATUS : PUT FINAL STATUS IN CALLER'S BUFFER
      RET

10$:  $SETAST_S #0                    : DISABLE AST RECOGNITION
      BLBS   W^XAFUNC,RECEIVE         : BRANCH ON TRANSMITT OR RECEIVE
```

```

TRANSMITT:          : SEND A MESSAGE
TSTL      W^STATE   : DOES STATE INDICATE UNSOLICITED INPUT?
BEQL      20$      : NO, OK
CVTBL    #-2,W^ERROR : YES, UNSOLICITED INPUT FOR TRANSMITT NOT VALID
15$: $SETAST_S #1   : RE-ENABLE AST RECOGNITION
BRB      RETURN_ERROR : TELL THE CALLER

20$: MOVZBL #1,W^STATE : CHANGE STATE TO WORD MODE MESSAGE SENT
      $QIO_S CHAN=W^CHAN,- : SEND WORD MODE QIO TO BUDDY
          FUNC=#<IOS WRITELBLK!IOSM_TIMED!IOSM_WORD!IOSM_SETFNCT>,- ; WORD+SETFNCT
          IOSB=W^IOSB1,-
          ASTPRM=W^STATE,-
          ASTADR=QIO AST,-
          P1=W^LENGTR,- : LENGTH OF BUFFER TO WRITE
          P2=#2,- : TWO BYTES
          P3=W^TIMER,- : TIME-OUT VALUE
          P4=#2 : INTERRUPT BUDDY + TRANSMITT
      MOVL  R0,W^DSW   : SAVE SYSTEM SERVIC STATUS
      BLBS  R0,50$    : BRANCH IF STARTED OK
      CVTBL #-3,W^ERROR : QIO ERROR FOR WORD MODE WRITE
      BRW   15$      : TELL CALLER ABOUT IT ALL

RECEIVE:           : RECEIVE MESSAGE FROM BUDDY
CASE      W^STATE,<- : CHECK RESULT STATE
          40$,- : 0 - NO UNSOLICITED MESSAGE, WAIT
          35$,- : 1 - BAD STATE
          35$,- : 2 - BAD STATE
          35$,- : 3 - BAD STATE
          35$,- : 4 - BAD STATE
          35$,- : 5 - BAD STATE
          50$>,TYPE=W : 6 - UNSOLICITED MESSAGE PROCESSING
35$: CVTBL #-9,W^ERROR : BAD STATE FOR RECEIVE FUNCTION
      BRW   15$      : END IT ALL

40$: MOVZBL #4,W^STATE : UPDATE STATE, WAITING FOR MESSAGE
50$: $SETAST_S #1     : RE-ENABLE AST RECOGNITION
      RET
      .DSABL LSB
      .PAGE
      .SBTTL ATTN_AST - ATTENTION AST ROUTINE

: PROCESS ATTENTION AST'S FOR DR11W
:
: .ENTRY ATTN_AST,^M<R2,R3,R4,R5>
: .ENABL LSB
MOVL  W^BUFFER,R4 : GET ADDRESS OF CALLER'S DATA BUFFER
BLBS  W^XAFUNC,ATTN_RECEIVE : CALLER'S FUNCTION IS RECEIVE MESSAGE

ATTN_TRANSMITT:    : SEND A MESSAGE TO BUDDY
CASE      W^STATE,<- : DISPATCH ON CURRENT STATE VARIABLE
          3$,- : 0 - UNSOLICITED INPUT - ERROR
          10$>,TYPE=W : 1 - NORMAL PROCESSING, TRANSMITT BUFFER
          CVTBL #-5,W^ERROR : BAD STATE VARIABLE - INCONSISTENT
          BRB  15$ : RETURN ERROR
3$: MOVL #2,W^STATE : UNSOLICITED INPUT - EITHER ERROR FROM
: BUDDY OR BOTH WANT TO TRANSMITT AT ONCE.

```

```

5$: RET ; PICK THIS UP AT MAINLINE LEVEL
10$: MOVZBL #3,W^STATE ; XMITT BLOCK OF DATA TO BUDDY
    $QIO_S CHAN=W^CHAN,- ; BLOCK MODE WRITE
    FUNC=#<IOS WRITELBLK!IOSM_TIMED!IOSM_SETFNCT!IOSM_CYCLE>,- ; SETFNCT+CYCLE
    IOSB=W^IOSB,-
    ASTPRM=W^STATE,-
    ASTADR=QIO_AST,-
    P1=(R4),- ; ADDRESS OF CALLER'S DATA BUFFER
    P2=W^LENGTH,- ; LENGTH OF BUFFER
    P3=W^TIMER,- ; TIMEOUT VALUE
    P4=#4 ; FNCT BITS FOR DR11W CSR
    BLBS R0,5$ ; RETURN IF QIO STARTED OK
    CVTBL #-4,W^ERROR ; BLOCK MODE WRITE QIO FAILED
    MOVL R0,W^DSW ; SAVE QIO STATUS
15$: BRW RETURN_ERROR ; TELL CALLER

ATTN_RECEIVE: ; READ A DATA BLOCK FROM BUDDY
CASE W^STATE,<- ; DISPATCH ON STATE VARIABLE
    40$,- ; 0 - UNSOLICITED INPUT
    30$,- ; 1 - BAD STATE
    30$,- ; 2 - BAD STATE
    30$,- ; 3 - BAD STATE
    45$>,TYPE=W ; 4 - SOLICITED READ
    CVTBL #-6,W^ERROR ; BAD STATE VARIABLE - INCONSISTENT
    BRW RETURN_ERROR ; THATS ALL
40$: MOVZBL #6,W^STATE ; FLAG NEXT STATE - UNSOLICITED READ
    BRB 50$

45$: MOVZBL #5,W^STATE ; FLAG STATE - SOLICITED READ
50$: $QIO_S CHAN=W^CHAN,- ; QIO - BLOCK MODE READ
    FUNC=#<IOS READLBLK!IOSM_TIMED!IOSM_SETFNCT>,- ; SETFNCT
    IOSB=W^IOSB,-
    ASTPRM=W^STATE,-
    ASTADR=W^QIO_AST,-
    P1=(R4),- ; ADDRESS OF CALLER'S DATA BUFFER
    P2=W^LENGTH,- ; LENGTH OF DATA BUFFER
    P3=W^TIMER,- ; TIMEOUT VALUE
    P4=#7 ; INTERRUPT+READ
    BLBS R0,55$ ; BRANCH IF QIO STARTED OK
    MOVL R0,W^DSW ; SAVE STATUS
    CVTBL #-10,W^ERROR ; BLOCK MODE READ QIO ERROR
    BRB 15$ ; RETURN_ERROR
55$: RET ; RETURN
    .DSABL LSB
    .PAGE
    .SBTTL QIO_AST - AST ROUTINE FOR QIO'S

; COMMON AST ROUTINE TO PROCESS END QIO ACTIONS
;
; .ENTRY QIO_AST,^M<R2,R3,R4,R5>
; .ENABL LSB

CMPW #SS$_CANCEL,W^IOSB ; IS THIS THE RESULT OF A CANCEL?
BEQL 25$ ; YES, DO NOTHING
BLBS W^XAFUNC,AST_RECEIVE ; BRANCH IF READ FUNCTION

```

XI
:
SDE
SDE
SDE
SDE
:
UCE

```
AST_TRANSMITT:
CASE    4(AP),<-
        10$, -
        20$, -
        10$, -
        30$, >,TYPE=W
10$:    CVTBL # -11,W^ERROR
        BRB   65$
20$:    BLBS  W^IOSB1,25$
        CVTBL # -12,W^ERROR
        MOVL  W^IOSB1,W^DSW
        BRB   65$
25$:    FET
30$:    BLBS  W^IOSB,35$
        CVTBL # -13,W^ERROR
        BRB   65$
35$:    MOVZWL #SS$ NORMAL,RO
        BRW   RETURN_SUCCESS

AST_RECEIVE:
CASE    4(AP),<-
        40$, -
        40$, -
        40$, -
        40$, -
        40$, -
        40$, -
        45$, -
        45$, >,TYPE=W
40$:    CVTBL # -14,W^ERROR
        BRB   65$
45$:    BLBS  W^IOSB,35$
        CVTBL # -15,W^ERROR
65$:    BRW   RETURN_ERROR
        .DSABL LSB
        .END
```

```
: END OF TRANSMITT FUNCTION
: DISPATCH ON STATE
: 0 - BAD STATE
: 1 - END OF WORD MODE WRITE
: 2 - BAD STATE
: 3 - END OF BLOCK MODE WRITE
: ERROR - INCONSITENT STATE VARIABLE
: RETURN_ERROR
: END OF WORD MODE WRITE - BRANCH IF FINISHED OK
: ERROR FROM WORD MODE WRITE
: SAVE IOSB1 FOR CALIER
: RETURN_ERROR

: CHECK FOR QIO ERRORS
: OOPS
: RETURN_ERROR

: RETURN NORMAL STATUS
: RETURN - EVERYTHING OK

: END ACTION FOR RECEIVE QIO'S
: DISPATCH ON STATE
: 0 - BAD STATE
: 1 - BAD STATE
: 2 - BAD STATE
: 3 - BAD STATE
: 4 - BAD STATE
: 5 - END OF SOLICITED READ
: 6 - END OF UNSOLICITED READ
: ERROR - INCONSISTENT STATE VARIABLE
: RETURN_ERROR

: BRANCH IF NO ERROR
: ERROR FROM BLOCK MODE READ QIO
```

XIC
:
:
:
SDE
:
E

XI
SDE
SDE
SDE
:
E

SDE

XALINK
MAR

LABIOLINK
COM

DRMASTER
FOR

LPATEST
FOR

LABIOSTR
COM

LABTOPEAK
FOR

XATEST
FOR

XMESSAGE
MAR

LABDEMO
COM

LABTOCOM
FOR

LABMBXDEF
FOR

LABTOSAMP
FOR

MAILCOMPRESS
COM

LABCHNDEF
FOR

CONNECT
COM

LABTOCON
FOR

LABDEMO
FOR

PEAK
FOR

DRCOPYBLD
COM

XIDRIVER
MAR

LABTOSEC
FOR

DRSLAVE
FOR

LABTOACQ
FOR

LABTOCOMP
COM

LABIOSTAT
FOR

TESTLABIO
FOR