```
.REM _
```

IDENTIFICATION

PRODUCT CODE: AC-F004A-MC

PRODUCT NAME: CXDRWA0 DR11-W MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

1. ABSTRACT

DRW IS AN IOMOD THAT EXERCISES ONE DR11-W. THE DEVICE IS EXERCISED USING THE MAINTENANCE MODE TO TRANSER A 16 WORD BUFFER.
MAINTENANCE MODE IS INTERNAL WRAP-AROUND LOGIC WHICH ELIMINATES THE NEED FOR A USER DEVICE OR PHYSICAL CABLE.

2. REQUIREMENTS

HARDWARE: ONE DR11-W INTERFACE
STORAGE:: DRW REQUIRES:
1. DECIMAL WORDS: 287
2. OCTAL WORDS: 0437
3. OCTAL BYTES: 1076

3. PASS DEFINITION

ONE PASS OF DRW CONSISTS OF TRANSFERRING ONE 16 WORD BLOCK OF DATA 77000(8) TIMES

4. EXECUTION TIME

ONE PASS OF DRW RUNNING ALONE ON A PDP11/04 PROCESSOR TAKES APPROXIMATELY 85 SECONDS.

5. CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:

DEVADR: 172410, VECTOR: 124, BR1: 5, DEVCNT: 1

REQUIRED PARAMETERS:

NONE

6. DEVICE/OPTION SET-UP

INSTALL DR11-W (NO PHYSICAL CABLE IS NECESSARY)

7. MODULE OPERATION

TEST SEQUENCE:

A. SET UP VECTOR AND DEVICE REGISTERS
B. GENERATE READ-WRITE BUFFER

C. GENERATE CHECK BUFFER
D. TRANSFER 16 WORDS IN MAINT. MODE
E. COMPARE DATA IN:OUT - REPORT ERRORS
F. REPEAT 77000 TIMES
G. SIGNAL END OF PASS, RESTART AT A.

IF DEVICE FAILS TO INTERRUPT,AN ERROR MESSAGE WILL OCCUR AND THE
PROGRAM WILL REQUEST THAT THE MODULE BE DROPPED.

8. OPERATION OPTIONS

NONE

9. NON STANDARD PRINTOUTS

NONE: ALL PRINTOUTS HAVE THE STANDARD FORMATS DESCRIBED IN THE
DEC/X11 DOCUMENT.

10.0 READ-WRITE BUFFER DESCRIPTION

SUBROUTINE LODBUF LOADS DRBUF WITH AN INCREMENTING PATTERN
(0,1,2,3,....15). WHEN XFERS ARE COMPLETE, DRBUF WILL BE
MODIFIED(0,0,2,2,4,4,.....14,14) DUE TO THE ALTERNATING
DATI-DATO SEQUENCE CHARACTERISTIC OF DR11-W MAINTENANCE MODE.

11.0 CHECK BUFFER DESCRIPTION

SUBROUTINE LODCHK LOADS CHKBUF WITH MODIFIED PATTERN
DESCRIBED IN 10.

12.0 BURST OPERATION

CHARACTERISTIC OF DR11-W MAINTENANCE MODE OPERATION, THE XFERS
ARE MADE IN ALTERNATING SINGLE CYCLE & BURST MODE FASHION.
FOR THE FIRST FOUR XFERS THE DR11-W REQUESTS THE BUS ONLY
ONCE AND DOES NOT RELEASE IT UNTIL THE END OF THE FOURTH XFER.
THE NEXT FOUR XFERS SEES THE DR11-W REQUESTING & RELEASING THE BUS
FOR EACH XFER. THIS ALTERNATING ACTION CONTINUES FOR SUBSEQUENT XFERS.

```
                        ;DR11-W DEC/X11 EXERCISER MODULE
000000'                 IOMOD   <DRWA >,172410,124,5,0,0,77000,165
000000'                 MODULE  140000,DRWA ,172410,124,5,0,0,77000,165
                        .TITLE  DRWA DEC/X11 SYSTEM EXERCISER MODULE
                ;       DDXCOM  VERSION  6      23-MAY-78
                                        .LIST   BIN
                                ;*****************************************************************************
000000'                         BEGIN:
000000' 051104  040527     040  MODNAM: .ASCII  /DRWA / ;MODULE NAME.
000005'    000                  XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 172410                  ADDR:   172410+0                ;1ST DEVICE ADDR.
000010' 000124                  VECTOR: 124+0                   ;1ST DEVICE VECTOR.
```

```
000012  000240                  BR1:    .BYTE   PRTY5+0         ;1ST BR LEVEL.
000013  000000                  BR2:    .BYTE   PRTY0+0         ;2ND BR LEVEL.
000014  000001                  DVID1:  0+1                     ;DEVICE INDICATOR 1.
000016  000000                  SR1:    OPEN                    ;SWITCH REGISTER 1
000020  000000                  SR2:    OPEN                    ;SWITCH REGISTER 2
000022  000000                  SR3:    OPEN                    ;SWITCH REGISTER 3
000024  000000                  SR4:    OPEN                    ;SWITCH REGISTER 4
                                ;*****************************************************************************
000026  140000                  STAT:   140000                  ;STATUS WORD.
000030  000224                  INIT:   START                   ;MODULE START ADDR.
000032  000224                  SPOINT: MODSP                   ;MODULE STACK POINTER.
000034  000000                  PASCNT: 0                       ;PASS COUNTER.
000036  077000                  ICONT:  77000                   ;# OF ITERATIONS PER PASS=77000
000040  000000                  ICOUNT: 0                       ;LOC TO COUNT ITERATIONS
000042  000000                  SOFCNT: 0                       ;LOC TO SAVE TOTAL SOFT ERRORS
000044  000000                  HRDCNT: 0                       ;LOC TO SAVE TOTAL HARD ERRORS
000046  000000                  SOFPAS: 0                       ;LOC TO SAVE SOFT ERRORS PER PASS
000050  000000                  HRDPAS: 0                       ;LOC TO SAVE HARD ERRORS PER PASS
000052  000000                  SYSCNT: 0                       ;# OF SYS ERRORS ACCUMULATED
000054  000000                  RANNUM: 0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056                          CONFIG:                         ;RESERVED FOR MONITOR USE
000056  000000                  RES1:   0                       ;RESERVED FOR MONITOR USE
000060  000000                  RES2:   0                       ;RESERVED FOR MONITOR USE
000062  000000                  SVR0:   OPEN                    ;LOC TO SAVE R0.
000064  000000                  SVR1:   OPEN                    ;LOC TO SAVE R1.
000066  000000                  SVR2:   OPEN                    ;LOC TO SAVE R2.
000070  000000                  SVR3:   OPEN                    ;LOC TO SAVE R3.
000072  000000                  SVR4:   OPEN                    ;LOC TO SAVE R4.
000074  000000                  SVR5:   OPEN                    ;LOC TO SAVE R5.
000076  000000                  SVR6:   OPEN                    ;LOC TO SAVE R6.
000100  000000                  CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
000102                          SBADR:                          ;ADDR OF GOOD DATA, OR
000102  000000                  ACSR:   OPEN                    ;CONTENTS OF CSR.
000104                          WASADR:                         ;ADDR OF BAD DATA, OR
000104  000000                  ASTAT:  OPEN                    ;STATUS REG CONTENTS.
000106                          ERRTYP:                         ;TYPE OF ERROR
000106  000000                  ASB:    OPEN                    ;EXPECTED DATA.
000110  000000                  AWAS:   OPEN                    ;ACTUAL DATA.
000112  000276                  RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
000114  000000                  WDTO:   OPEN                    ;WORDS TO MEMORY PER ITERATION
000116  000000                  WDFR:   OPEN                    ;WORDS FROM MEMORY PER ITERATION
000120  000000                  INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
000122  000165                  IDNUM:  165                     ;MODULE IDENTIFICATION NUMBER=165
        000040                          .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
                                        .NLIST
                                        .WORD   0
                                        .LIST
                                        .ENDR
        000224                  MODSP:
                                ;*****************************************************************************
                                ;THIS MODULE TESTS THE DR-11W DIRECT MEMORY ACCESS INTERFACE

                                        ;INITIALIZATION FOR (DMA) DR-11W


000224                          START:
```



```
000224  016705  177556                MOV     ADDR,R5         ;GET 1ST ADDRESS
000230  010567  000526                MOV     R5,WCR          ;SET UP WORD COUNT REG.POINTER
000234  005725                        TST     (R5)+           ;MAKE CERTAIN REG. CAN BE ACCESSED
000236  010567  000522                MOV     R5,BAR          ;SET UP BUS ADDRESS REG. POINTER
000242  005725                        TST     (R5)+           ;ACCESS REG.
000244  010567  000516                MOV     R5,CSR          ;SET UP CONTROL STATUS REG. POINTER
000250  005725                        TST     (R5)+           ;ACCESS REGISTER
000252  010567  000512                MOV     R5,BDR          ;SET UP DATA BUFFER REG. POINTER
000256  005715                        TST     (R5)            ;ACCESS REG.
000260  016705  177524                MOV     VECTOR,R5       ;LOAD DEVICE VECTOR
000264  010567  000502                MOV     R5,DRVEC0       ;SETUP DEV VECTOR POINTER
000270  005725                        TST     (R5)+           ;ACCESS REGISTER
000272  010567  000476                MOV     R5,DRVEC2       ;SETUP PRIORITY POINTER

                                ; DEVICE SERVICE CODE

000276  012767  077000  000440  RESTRT: MOV     #77000,INTPSC   ;TIMER FOR WHEN TO END PASS
000304                          CONT:
000304  104407  000000                  BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
000310  104407  000000                  BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
000314  012777  177760  000440        MOV     #-20,@WCR       ;SIXTEEN WORD TRANSFER
000322  012767  000776  000424        MOV     #DRBUF,VA       ;SETUP BASE ADDRESS
000330  104415  000000  000754        GETPA$,BEGIN,VA         ;GET PHYSICAL ADDRESS FROM 16-BIT VA
000336  016777  000414  000420        MOV     PA,@BAR ;SET UP REAL ADDR.
000344  005077  000416                CLR     @CSR            ;CLEAR THE CSR
000350  012777  177777  000412        MOV     #-1,@BDR        ;SET UP DATA
000356  056777  000376  000402        BIS     PA,@CSR ;SET EXTENDED MEM. BITS
000364  012777  000512  000400        MOV     #DRIR,@DRVEC0   ;SET VECTOR TO SERVICE ROUTINE
000372  116777  177414  000374        MOVB    BR1,@DRVEC2     ;SET PRIORITY
000400  005067  000342                CLR     INTFLG          ;CLEAR INTERRUPT FLAG
000404  004767  000244                JSR     PC,LODBUF       ;GENERATE READ AND WRITE BUFFER
000410  004767  000202                JSR     PC,LODCHK       ;GENERATE CHECKING BUFFER
000414  012767  000010  000330        MOV     #10,TMRCNT      ;SET UP INTERRUPT LOOP
000422  005067  000322        TIME:   CLR     TMR             ;SETUP TIMER COUNTER
000426  052777  010000  000332        BIS     #10000,@CSR     ;MAINTENANCE MODE
000434  052777  000501  000324        BIS     #501,@CSR       ;IE,CYCLE,AND GO

                                ; WAIT FOR INTERRUPT CODE

000442                          TIMER:
000442  104407  000000                  BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
000446  104407  000000                  BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
000452  005767  000270                  TST     INTFLG          ;DID INTERRUPT OCCUR?
000456  001402                          BEQ     1$              ;IF NOT,CONTINUE IN TIMER
000460  000167  000042                  JMP     CHECK           ;IF YES,GO CHEC DATA
000464  005367  000260          1$:     DEC     TMR             ;COUNT MORE TIME;IS TMR FINISHED
000470  001364                          BNE     TIMER           ;IF NOT,BREAK AGAIN
000472  005367  000254                  DEC     TMRCNT          ;DECREASE OVERALL COUNT
000476  100361                          BPL     TIMER           ;GO AGAIN IF MORE TIME
000500  104403  000000  000702          MSGN$,BEGIN,HUNG        ;ASCII MESSAGE CALL WITH COMMON HEADER
000506  104410  000000                  END$,BEGIN              ;

                                ;INTERRUPT SERVICE CODE

000512  005267  000230        DRIR:   INC     INTFLG          ;INTERRUPT HAS OCCURRED
000516  042777  000100  000242        BIC     #100,@CSR       ;CLEAR INTERRUPT ENABLE
```

```
000524' 000002                          RTI                     ;RETURN TO CODE WHERE INTERRUPT OCCURRED

                                        ;DATA CHECK CODE

000526' 012701  000776'         CHECK:  MOV     #DRBUF,R1       ; GET THE BUFFER ADDRESS
000532' 012702  000010                  MOV     #8.,R2          ; LOAD THE COUNTER
000536' 012703  001036'                 MOV     #CHKBUF,R3      ;GET CHECK BUFFER ADDRESS
000542' 022123                  1$:     CMP     (R1)+,(R3)+     ; ARE THE TWO SEQUENTIAL WORDS EQUAL ?
000544' 001003                          BNE     2$              ; NO, REPORT AN ERROR
000546' 005302                          DEC     R2              ; DONE THE WHOLE BUFFER ?
000550' 003374                          BGT     1$              ; NO, KEEP CHECKING
000552' 000415                          BR      DONE            ;YES;CHECK FOR ENDPASS
000554' 016767  000206  177320  2$:     MOV     CSR,ACSR        ;ADDRESS OF CONTROL STATUS REGISTER
000562' 014167  177322                  MOV     -(R1),AWAS      ;ACTUAL DATA
000566' 010167  177312                  MOV     R1,WASADR       ;ADDRESS OF ACTUAL DATA
000572' 014167  177310                  MOV     -(R1),ASB       ;CORRECT DATA
000576' 010167  177300                  MOV     R1,SBADR        ;ADDRESS OF CORRECT DATA
                                        ;*****************************************************************
000602' 104404  000000'                 DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************
000606'                         DONE:
000606' 104413  000000'                 ENDIT$,BEGIN            ;SIGNAL END OF ITERATION.
                                                                ;MONITOR SHALL TEST END OF PASS
000612' 000167  177466                  JMP     CONT            ;ELSE DO IT AGAIN


000616' 012702  001036'         LDDCHK: MOV     #CHKBUF,R2      ;LOAD CHKBUF WITH INCREMENTING
000622' 005001                          CLR     R1              ;MODIFIED PATTERN
000624' 005003                          CLR     R3
000626' 010322                  1$:     MOV     R3,(R2)+
000630' 010322                          MOV     R3,(R2)+
000632' 062701  000002                  ADD     #2,R1
000636' 020127  000020                  CMP     R1,#20          ;ARE WE FINISHED?
000642' 001001                          BNE     2$              ;NO
000644' 000207                          RTS     PC              ;YES
000646' 062703  000002          2$:     ADD     #2,R3
000652' 000765                          BR      1$


000654' 012701  000776'         LDDBUF: MOV     #DRBUF,R1       ;LOAD DRBUF WITH INCREMENTING PATTERN
000660' 005002                          CLR     R2
000662' 005021                          CLR     (R1)+
000664' 005202                  1$:     INC     R2
000666' 020227  000020                  CMP     R2,#20          ;ARE WE DONE?
000672' 001402                          BEQ     LDEXIT          ;YES
000674' 010221                          MOV     R2,(R1)+        ;NO,CONTINUE
000676' 000772                          BR      1$
000700' 000207                  LDEXIT: RTS     PC

000702' 000706'                 HUNG:   FAIL
000704' 177777                          177777
000706' 042045  053105  041511  FAIL:   .ASCIZ  /%DEVICE FAILED TO INTERRUPT%/
000714' 020105  040506  046111
000722' 042105  052040  020117
000730' 047111  042524  051122
000736' 050125  022524     000
```



```
        000744'                         .EVEN
000744' 000000                  INTPSC: OPEN                    ;COUNTS FOR WHEN TO ENDPASS
000746' 000000                  INTFLG: OPEN
000750' 000000                  TMR:    OPEN
000752' 000000                  TMRCNT: OPEN
000754' 000000                  VA:     OPEN
000756' 000000                  PA:     OPEN
000760' 000000                  EA:     OPEN
000762' 000000                  WCR:    OPEN
000764' 000000                  BAR:    OPEN
000766' 000000                  CSR:    OPEN
000770' 000000                  BDR:    OPEN
000772' 000000                  DRVEC0: OPEN
000774' 000000                  DRVEC2: OPEN
000776' 000020                          DRBUF:  .BLKW 20
001036' 000020                          CHKBUF: .BLKW  20
        000001                          .END
```

```
ACSR     000102R          192#    284*
ADDR     000006R          158#    216
ADDR22=  001000           210#
ASB      000106R          196#    287*
ASTAT    000104R          194#
AWAS     000110R          197#    285*
BAR      000764R          219*    239*    337#
BDR      000770R          223*    241*    339#
BEGIN    000000R          155#    234     235     238     256     257     265     266     290     293
BIT0   = 000001           210#
BIT1   = 000002           210#
BIT10  = 002000           210#
BIT11  = 004000           210#
BIT12  = 010000           210#
BIT13  = 020000           210#
BIT14  = 040000           210#
BIT15  = 100000           210#
BIT2   = 000004           210#
BIT3   = 000010           210#
BIT4   = 000020           210#
BIT5   = 000040           210#
BIT6   = 000100           210#
BIT7   = 000200           210#
BIT8   = 000400           210#
BIT9   = 001000           210#
BREAK$=  104407           210#    234     235     256     257
BR1      000012R          160#    244
BR2      000013R          161#
BTOD$  = 104421           210#
CDATA$=  104412           210#
CHECK    000526R          260     276#
CHKBUF   001036R          278     298     343#
CONFIG   000056R          180#
CONT     000304R          233#    295
CSR      000766R          221*    240*    242*    250*    251*    271*    284     338#
CSRA     000100R          190#
DATCK$=  104411           210#
DATER$=  104404           210#    290
DONE     000606R          283     292#
DRBUF    000776R          237     276     311     342#
DRIR     000512R          243     270#
DRVEC0   000772R          226*    243*    340#
DRVEC2   000774R          228*    244*    341#
DVID1    000014R          162#
EA       000760R          242     335#
ENDIT$=  104413           210#    293
END$   = 104410           210#    266
ERRTYP   000106R          195#
EXIT$  = 104400           210#
FAIL     000706R          321     323#
GETPA$=  104415           210#    238
GWBUF$=  104414           210#
HRDCNT   000044R          175#
HRDER$=  104405           210#
HRDPAS   000050R          177#
HUNG     000702R          265     321#
```



```
ICONT    000036R          172#
ICOUNT   000040R          173#
IDNUM    000122R          202#
INIT     000030R          169#
INTFLG   000746R          245*    258     270*    330#
INTPSC   000744R          232*    329#
INTR     000120R          201#
LDEXIT   000700R          316     319#
LDDBUF   000654R          246     311#
LDDCHK   000616R          247     298#
MAP22$=  104416           210#
MODNAM   000000R          156#
MODSP    000224R          170     208#
MSGN$  = 104403           210#    265
MSGS$  = 104402           210#
MSG$   = 104401           210#
NULL   = 000000           210#
OPEN   = 000000           157     163     164     165     166     183     184     185     186     187     188     189     190
                          192     194     196     197     199     200     201     210#    329     330     331     332     333
                          334     335     336     337     338     339     340     341
OTOA$  = 104420           210#
PA       000756R          239     334#
PASCNT   000034R          171#
PIRQ$  = 000004           210#
POPSP  = 005726           210#
POPSP2=  022626           210#
PRTY   = 000000           210#
PRTY0  = 000000           161     210#
PRTY1  = 000040           210#
PRTY2  = 000100           210#
PRTY3  = 000140           210#
PRTY4  = 000200           210#
PRTY5  = 000240           160     210#
PRTY6  = 000300           210#
PRTY7  = 000340           210#
PS     = 177776           210#
PSW    = 177776           210#
PUSH   = 005746           210#
PUSH2  = 024646           210#
RAND$  = 104417           210#
RANNUM   000054R          179#
RESTRT   000276R          198     232#
RES1     000056R          181#
RES2     000060R          182#
RSTRT    000112R          198#
SBADR    000102R          191#    288*
SOFCNT   000042R          174#
SOFER$=  104406           210#
SOFPAS   000046R          176#
SPOINT   000032R          170#
SPSIZ  = 000040           174#    203
SR1      000016R          163#
SR2      000020R          164#
SR3      000022R          165#
SR4      000024R          166#
START    000224R          169     215#
```

```
STAT    000026R           168#
SVR0    000062R           183#
SVR1    000064R           184#
SVR2    000066R           185#
SVR3    000070R           186#
SVR4    000072R           187#
SVR5    000074R           188#
SVR6    000076R           189#
SYSCNT  000052R           178#
TIME    000422R           249#
TIMER   000442R           255#    262     264
TMR     000750R           249*    261*    331#
TMRCNT  000752R           248*    263*    332#
TRPDFD= 000022            210#
VA      000754R           237#    238     333#
VECTOR  000010R           159#    225
WASADR  000104R           193#    286*
WCR     000762R           217*    236*    336#
WDFR    000116R           200#
WDTO    000114R           199#
XFLAG   000005R           157#
.       = 001076R         328#    342#    343#

. ABS.  000000    000
        001076    001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XDRWA0,XDRWA0/SOL/CRF:SYM=DDXCOM,XDRWA0
 RUN-TIME: 1 1 .2 SECONDS
 RUN-TIME RATIO: 11/3=3.7
 CORE USED:  7K  (13 PAGES)
```