.REM _

IDENTIFICATION

PRODUCT CODE: AC-E899D-MC

PRODUCT NAME: CXNCAD0 NC-11A MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

SEQ 0002

1. ABSTRACT

"NCA" IS AN "IOMODR" THAT EXERCISES ONE NC-11A INTERFACE.
THE NC-11A INTERFACE DOES MEMORY INCREMENTS VIA NPR UNTIL
A WORD OR BYTE REACHES MAXIMUM CAPACITY AND ATTEMPS TO
OVERFLOW. AT THIS TIME AN INTERRUPT IS GENERATED AT BR LEVEL 7.
THE INTERFACE ALSO DOES TRANSFERS OF DATA TO SERIAL LOCATIONS
IN CORE VIA NPR. THIS MODE IS TERMINATED BY A WORD COUNT OVERFLOW
AND CONSEQUENT INTERRUPT. THE RATE OF INCREMENT OR TRANSFER IS SET
BY A VARIABLE SPEED CLOCK, WHICH IS TURNED ON BY THE SWITCH ON
THE BACK OF THE INTERFACE. THIS SWITCH MUST BE ON TO RUN THIS MODULE.

* NOTE: DATA LATE ERRORS OCCUR WHEN RUNNING ON PDP-11/20 CPU *

2. REQUIREMENTS

HARDWARE: NC-11A INTERFACE WITH MAINTENANCE SWITCH IN THE "ON" POSITION.
STORAGE:: NCA REQUIRES:
1. DECIMAL WORDS: 2519
2. OCTAL WORDS: 04727
3. OCTAL BYTES: 11656

3. PASS DEFINITION

ONE PASS OF NCA MODULE CONSISTS OF FIFTY ITERATIONS OF EACH BASIC TEST SEQUENCE, WHICH RESULTS IN:

200 PROGRAM INTERRUPTS - 3,404,750 NON-PROCESSOR REQUESTS.

4. EXECUTION TIME

NCA RUNNING ALONE ON PDP-11/10 TAKES APPROXIMATELY 30 SECONDS.

5. CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:

DEVADR: 164000, VECTOR: 270, BR1: 7, DEVCNT: 1

REQUIRED PARAMETERS:

ONLY IF PDP-11/20 CPU TYPE (REF. TO 8.)

6. DEVICE/OPTION SETUP

THE NC-11A MUST HAVE THE MAINTENANCE SWITCH IN THE "ON" POSITION.

SEQ 0003

7. MODULE OPERATION

THE FIRST MODE OF OPERATION IS WORD INCREMENT MODE.
THIS MODE IS REPEATED FOR 50 INTERATIONS. UPON COMPLETION,
ODD BYTE OVERFLOW MODE IS ENABLED FOR 50 INTERATIONS.
THIS INTURN ENABLES THE EVEN BYTE MODE FOR 50 INTERATIONS.
THIS INTURN ENABLES THE LIST MODE FOR 50 INTERATIONS.
UPON COMPLETION, AN END OF PASS IS REPORTED AND THE MODULE IS RESTARTED.

8. OPERATION OPTIONS

SR1 IS USED TO INHIBIT TESTING MODES OF OPERATION OF THE NC11A.
BITS 0 THRU 2 SHOULD BE SET TO A ONE IF RUNNING ON PDP-11/20 CPU.

SR1 BIT0 = 1	INHIBIT WORD INCREMENT MODE.
SR1 BIT1 = 1	INHIBIT ODD BYTE INCREMENT MODE
SR1 BIT2 = 1	INHIBIT EVEN BYTE INCREMENT MODE.
SR1 BIT3 = 1	INHIBIT LIST MODE.

9. NON STANDARD PRINTOUTS

NONE. ALL PRINTOUTS HAVE STANDARD MEANINGS AS REPRESENTED IN
DEC/X11 DOCUMENTATION.

10. MODULE TEST ENVIROMENT

THE NCA MODULE IS KNOWN TO OPERATE UNDER THIS ENVIROMENT:

#1

PDP-11/20	CPU WITH 28K
TC11	2 DRIVES
TM11	1 DRIVE
TA11	2 DRIVES
NC11A	1 UNIT
LP11	1 UNIT
AA11	1 UNIT

#2

PDP-11/40	CPU WITH 28K
RK11-D	1 DRIVE
TM11	1 DRIVE
NC11A	1 UNIT
AA11	1 UNIT

#3

PDP-11/10	CPU WITH 16K
RK11-D	1 DRIVE
TA11	2 DRIVES
NC11A	1 UNIT
AA11	1 UNIT
KW11L	1 UNIT

#4

PDP-11/34	CPU WITH 28K
TC11	2 DRIVES
TM11	1 DRIVE
TA11	2 DRIVES
NC11A	1 UNIT
LP11	1 UNIT
AA11	1 UNIT
VSV01	1 UNIT
RK11-D	1 DRIVE

```
                                        .LIST   SEQ,BIN
                                        IOMODR  <NCAD >,164000,270,7,0,0,2,66
                                        MODULE  152000,'NCAD ',164000,270,7,0,0,2,66
                                        .TITLE  NCAD DEC/X11 SYSTEM EXERCISER MODULE
                                ;       DDXCOM  VERSION  6      23-MAY-78
                                        .LIST   BIN
                                ;*****************************************************************************
000000'                         BEGIN:
000000' 041516  042101      040 MODNAM: .ASCII  /NCAD / ;MODULE NAME.
000005'     000                 XFLAG:  .BYTE   OPEN                ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 164000                  ADDR:   164000+0                    ;1ST DEVICE ADDR.
000010' 000270                  VECTOR: 270+0                       ;1ST DEVICE VECTOR.
000012'     340                 BR1:    .BYTE   PRTY7+0             ;1ST BR LEVEL.
000013'     000                 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' 152000                  STAT:   152000                      ;STATUS WORD.
000030' 000256'                 INIT:   START                       ;MODULE START ADDR.
000032' 000324'                 SPOINT: MODSP                       ;MODULE STACK POINTER.
000034' 000000                  PASCNT: 0                           ;PASS COUNTER.
000036' 000002                  ICONT:  2                           ;# OF ITERATIONS PER PASS=2
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' 000300'                 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' 000066                  IDNUM:  66                          ;MODULE IDENTIFICATION NUMBER=66
```



```
          000040                        .REPT   SPSIZ             ;MODULE STACK STARTS HERE.
                                        .NLIST
                                        .WORD   0
                                        .LIST
                                        .ENDR
000224'                         MODSP:
                                ;*****************************************************************************
```

```
                        ;SUB-SECTION INTERATION EXECUTION COUNT
INTER:  50.

;BUFFER ADDRESS...LOCAL TO PROGRAM

RBUFVA: BUFFER

;SPECIAL FUNCTIONS TO INTERFACE

        CLHLD=1                 ;CLEAR HOLD REGISTER
        CLCOVF=2                ;CLEAR ALL OVERFLOW
        CONVT=4                 ;CONVERT
        CLDP=10                 ;CLEAR JOY STICK DEPRESS
        CLZOVF=20               ;CLEAR Z OVERFLOW FLOP
        STM=40                  ;SET TIMING MARK
        CLALL=102               ;CLEAR ALL

;INTERFACE REGISTERS

CMDCSR: 164000                  ;COMMAND REGISTER
OFFSET: 164002                  ;OFFSET ADDRESS REGISTER
XYHOLD:                         ;WRITE - X/Y HOLD REGISTER
ADDREG: 164004                  ;READ - ADDRESS REGISTER
WCSHI:  164006                  ;WORD COUNT ADDRESS
CAZLO:  164010                  ;BUS ADDRESS
MISCFU: 164012
SFUNC:  164014                  ;SPECIAL FUNCTION ADDRESS

OVFINT: 270                     ;OVERFLOW INTERRUPT VECTOR
OVFIT1: 272
ZOVINT: 274                     ;Z OVERFLOW INTERRUPTS VECTOR
ZOVIT1: 276

;INITIALIZATION OF GENERAL DEVICE ADDRESS AND VECTOR POINTERS

000256' 012767 000230' 177634  START:  MOV     #152.,INTR      ;152. INTERRUPTS/ITERATION
000264' 012767 004226' 177622          MOV     #2198.,WDTO     ;2198. WORDS TO MEM/ITERATION
000272' 012767 004226' 177616          MOV     #2198.,WDFR     ;2198.. WORDS FROM MEM/ITERATION
000300' 016700 177502          RESTRT: MOV     ADDR,R0         ;GET BASE BUS ADDRESS
000304' 012701 000230'                 MOV     #CMDCSR,R1      ;GET ADDRESS POINTER
000310' 010021                 1$:     MOV     R0,(R1)+        ;LOAD INTO BUS POINTER LOCATION
000312' 005720                         TST     (R0)+           ;TEST FOR SLAVE-SYNC
000314' 022701 000246'                 CMP     #OVFINT,R1      ;TEST IF FINISHED THE ADDRESSES
000320' 001373                         BNE     1$              ;BR IF NOT DONE?
000322' 016700 177462                  MOV     VECTOR,R0       ;LOAD VECTOR ADDRESS
000326' 010021                 2$:     MOV     R0,(R1)+        ;LOAD VECTOR POINTER
000330' 005720                         TST     (R0)+           ;BUMP R0 BY 2
000332' 022701 000256'                 CMP     #ZOVIT1+2,R1    ;TEST FOR LAST VECTOR
000336' 001373                         BNE     2$              ;BR IF NOT DONE?

000340' 012777 000102 177676           MOV     #CLALL,@SFUNC   ;CLEAR NC11-A HARDWARE
000346' 016767 177656 177524           MOV     CMDCSR,CSRA     ;SAVE FOR TYPEOUT'S
```



```
;NOW FIND A 2048 WORD BOUNDARY ADDRESS AND SAVE IT IN LOCATION "TARGET"

000354' 016700 177646  BOUND:  MOV     RBUFVA,R0       ;LOAD VIRTUAL BUFFER POINTER
000360' 032700 004000          BIT     #BIT11,R0       ;TEST FOR BIT 11 ON
000364' 001005                 BNE     2$              ;BIT 11 IS ON BRANCH
000366' 005720                 1$:     TST     (R0)+   ;ADJUST ADDRESS
000370' 032700 004000          BIT     #BIT11,R0       ;TEST BIT 11 AGAIN
000374' 001774                 BEQ     1$              ;BR IF NOT SET
000376' 000404                 BR      3$              ;HAVE FOUND A BIT 11 BOUNDARY
000400' 005720                 2$:     TST     (R0)+   ;ADJUST ADDRESS
000402' 032700 004000          BIT     #BIT11,R0       ;TEST BIT 11 AGAIN
000406' 001374                 BNE     2$              ;BR IF SET
000410' 010067 001232          3$:     MOV     R0,TARGET       ;SAVE A BIT 11 BOUNDARY
000414' 010067 001230                  MOV     R0,TARGT1       ;SAVE A BIT 11 BOUNDRY
000420' 005267 001224                  INC     TARGT1          ;MAKE A ODD-BYTE POINTER

;DETERMINE IF THE "INTERFACE MODE SWITCH" HAS BEEN SET
;IF NOT REPORT ERROR AND DROP THE MODULE.

000424' 005077 001216  OVINIT: CLR     @TARGET         ;CLEAR DESTINATION ADDRESS
000430' 016777 001212 177574           MOV     TARGET,@OFFSET  ;LOAD DESTINATION ADDRESS
000436' 012777 001401 177564           MOV     #1401,@CMDCSR   ;ENABLE NC-11A FOR 32X32X16
000444' 012700 004000                  MOV     #BIT11,R0       ;LOAD LOOP COUNTER
000450' 005777 001172          1$:     TST     @TARGET         ;TEST FOR NON-ZERO VALUE
000454' 001021                         BNE     2$              ;BR IF SET
000456' 104407 000000'                 BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
000462' 104407 000000'                 BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
000466' 005300                         DEC     R0              ;DELAY DONE ?
000470' 001367                         BNE     1$              ;BR IF NOT DONE
000472' 012777 000102 177544           MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
000500' 012767 000033 177400           MOV     #33,ERRTYP      ;DEVICE NOT IN MAINT. MODE
                                       ;*****************************************************
000506' 104405 000000' 000000          ERDERS,BEGIN,NULL       ;NC-11A FAILED TO ADDRESS TARGET LOCATION
                                       ;*****************************************************
                                                       ;IS THE MAINTENANCE SWITCH IN THE "ON" POSITION ?????
000514' 104410 000000'                 ENDS,BEGIN              ;

000520' 012777 000102 177516  2$:      MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
```

```
                                        ;THIS IS THE PRIMING CODE.
                                        ;THE INTERFACE IS INITIALLY SET UP TO INCREMENT WORD
                                        ;TARGET UNTIL OVERFLOW AND THEN TO INTERRUPT.

000526' 012777  000652' 177512  PRIME:  MOV     #OVSR,@OVFINT   ;SETUP INTERRRUPT VECTOR
000534' 116777  177252  177506          MOVB    BR1,@OVFITI     ;LOAD BR LEVEL
000542' 016777  177506  177502          MOV     ZOVITI,@ZOVINT  ;RESET Z OVERFLOW VECTOR
000550' 005077  177500                  CLR     @ZOVITI
000554' 016767  177444  000066          MOV     INTER,PASSCT    ;LOAD PASS COUNT LOC.
000562' 012777  000102  177454          MOV     #CLALL,@SFUNC   ;CLEAR ALL
000570' 032767  000001  177220          BIT     #BIT0,SR1       ;TEST INHIBIT THIS TEST BIT
000576' 001071                          BNE     ODDPRM          ;BR IF SET TO NEXT SECTION
000600' 005077  001042          WORDBK: CLR     @TARGET         ;CLEAR TARGET LOC.
000604' 012777  001400  177416          MOV     #1400,@CMDCSR   ;RESOLUTION SET TO 32X32X16
000612' 016777  001030  177412          MOV     TARGET,@OFFSET  ;BASE AT BUFFER TARGET LOC
000620' 017767  177410  000020          MOV     @ADDREG,JUNK    ;TWO DUMMY READS TO
000626' 052777  000101  177374          BIS     #101,@CMDCSR    ;ENABLE OVERFLOW INTERRUPT
000634' 017767  177374  000004          MOV     @ADDREG,JUNK    ;CLEAR BUFFER CHAIN
000642' 104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

000646' 000000                  JUNK:   0
000650' 000000                  PASSCT: 0


                                        ;OVERFLOW SERVICE ROUTINES
                                        ;THIS ROUTINE SERVICES 20 INTERRUPTS OF WORD OVERFLOW MODE

000652' 012777  000102  177364  OVSR:   MOV     #CLALL,@SFUNC   ;CLEAR THE INTERFACE
                                        ;-----------------------------------------------------------------
000660' 000004  000000' 000666'         PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
                                        ;-----------------------------------------------------------------
000666' 017767  177336  177206  1$:     MOV     @CMDCSR,ACSR    ;READ STATUS
000674' 100010                          BPL     2$
000676' 012767  000003  177202          MOV     #3,ERRTYP       ;DATA LATE
                                        ;*****************************************************************
000704' 104406  000000' 000000          SOFER$,BEGIN,NULL       ;TIME OUT ERROR OR JOY STICK FLAG SET
                                        ;*****************************************************************
000712' 104410  000000'                 END$,BEGIN              ;

000716' 012767  177777  177162  2$:     MOV     #-1,ASR         ;LOAD "SHOULD BE" VALUE
000724' 016767  000716  177152          MOV     TARGET,WASADR   ;LOAD "WAS ADDRESS" VALUE
000732' 017767  000710  177150          MOV     @TARGET,AWAS    ;LOAD "WAS" VALUE
000740' 026767  177142  177142          CMP     ASR,AWAS        ;TEST VALUE
000746' 001402                          BEQ     3$              ;YES - GO ON
                                        ;*****************************************************************
000750' 104404  000000'                 DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************
000754' 005367  177670          3$:     DEC     PASSCT          ;FINISHED ?
000760' 001307                          BNE     WORDBK          ;BR IF NOT DONE
```



```
                                        ;PRIME AND GO FOR TESTING ODD BYTE NPR'S
                                        ;TEST LOCATION TARGET+1

000762' 016767  177236  177660  ODDPRM: MOV     INTER,PASSCT    ;RESET PASS COUNT
000770' 012777  001062' 177250          MOV     #ODDINT,@OVFINT ;ADJUST INT VECTOR
000776' 032767  000002  177012          BIT     #BIT1,SR1       ;TEST INHIBIT THIS TEST BIT
001004' 001067                          BNE     FVBYP           ;BR IF SET TO NEXT SECTION
001006' 005077  000634          ODBYOK: CLR     @TARGET         ;CLEAR TARGET LOCATION
001012' 012777  003400  177210          MOV     #3400,@CMDCSR   ;32X32X8 MODE
001020' 012777  000004  177205          MOV     #4,@XVHOLD      ;SET TO BUMP BYTE 1
001026' 016777  000614  177176          MOV     TARGET,@OFFSET  ;LOAD OFFSET REGISTER
001034' 017767  177174  177604          MOV     @ADDREG,JUNK    ;CLEAR OFF BUFFER
001042' 052777  000101  177160          BIS     #101,@CMDCSR    ;INTERRUPT ENABLE + GO!
001050' 017767  177160  177570          MOV     @ADDREG,JUNK
001056' 104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

001062' 012777  000102  177154  ODDINT: MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
001070' 000004  000000' 001076'         ;-----------------------------------------------------------------
                                        ;PIRQ$,BEGIN,1$         ; QUEUE UP TO CONTINUE AT 1$ AND RTI
                                        ;-----------------------------------------------------------------
001076' 017767  177126  176776  1$:     MOV     @CMDCSR,ACSR    ;LOOK FOR NPR ERROR
001104' 100010                          BPL     2$              ;OK
001106' 012767  000032  176772          MOV     #32,ERRTYP      ;NPR ERROR
                                        ;*****************************************************************
001114' 104405  000000' 000000          HRDER$,BEGIN,NULL       ;NPR-TIME OUT ERROR OR JOYSTICK FLAG
                                        ;*****************************************************************
001122' 104410  000000'                 END$,BEGIN              ;

001126' 012767  177400  176752  2$:     MOV     #177400,ASR     ;LOAD "SHOULD BE" VALUE
001134' 017767  000506  176746          MOV     @TARGET,AWAS    ;LOAD "WAS" VALUE
001142' 026767  176740  176740          CMP     ASR,AWAS        ;TEST TWO VALUES
001150' 001402                          BEQ     3$              ;OK
                                        ;*****************************************************************
001152' 104404  000000'                 DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

001156' 005367  177466          3$:     DEC     PASSCT          ;FINISHED ?
001162' 001311                          BNE     ODBYOK          ;BR IF NOT DONE
```

```
                                        ;PRIME + GO FOR TESTING EVEN BYTE NPR'S
                                        ;TEST BYTE TARGET
001164' 016767  177034  177456  EVBYPM: MOV     INTER,PASSCT    ;RESET PASS COUNT
001172' 012777  001256' 177046          MOV     #EVINT,@OVFINT  ;ADJUST VECTOR
001200' 032767  000004  176610          BIT     #BIT2,SR1       ;TEST INHIBIT THIS TEST BIT
001206' 001064                          BNE     LISTPM          ;BR IF SET
001210' 005777  000432          EVBYOK: CLR     @TARGET         ;CLEAR TARGET LOCATION
001214' 012777  003400  177006          MOV     #3400,@CMDCSR   ;22X32X8 MODE
001222' 016777  000420  177002          MOV     TARGET,@OFFSET  ;OFFSET AT 7K
001230' 017767  177000  177410          MOV     @ADDREG,JUNK
001236' 052777  000101  176764          BIS     #101,@CMDCSR    ;ENABLE NC-11
001244' 017767  176764  177374          MOV     @ADDREG,JUNK
001252' 104400  000000'                 EXITS,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

001256' 012777  000102  176760  EVINT:  MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
                                        ;-------------------------------------------------------------------
001264' 000004  000000' 001272'         PIRQS,BEGIN,1S          ; QUEUE UP TO CONTINUE AT 1S AND RTI
                                        ;-------------------------------------------------------------------
001272' 017767  176732  176602  1S:     MOV     @CMDCSR,ACSR    ;LOOK FOR NPR ERROR
001300' 100410                          BPL     2S              ;OK
001302' 012767  000032  176576          MOV     #32,ERRTYP      ;NPR ERROR
                                        ;*******************************************************************
001310' 104405  000000' 000000          HRDERS,BEGIN,NULL       ;NPR ERROR FLAG SET
                                        ;*******************************************************************
001316' 104410  000000'                 ENDS,BEGIN              ;

001322' 017767  000320  176560  2S:     MOV     @TARGET,AWAS    ;LOAD "WAS" VALUE
001330' 012767  000377  176550          MOV     #377,ASB        ;LOAD "SHOULD BE" VALUE
001336' 026767  176544  176544          CMP     ASB,AWAS        ;COMPARE VALUE
001344' 001402                          BEQ     3S              ;OK
                                        ;*******************************************************************
001346' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*******************************************************************

001352' 005367  177272          3S:     DEC     PASSCT          ;FINISHED ?
001356' 001314                          BNE     EVBYOK          ;BR IF NOT DONE
```



```
                                        ;PRIME FOR LIST MODE + GO
                                        ;DO A TRANSFER OF 2048 WORDS AT THE CLOCK RATE THEN INTERRUPT.
001360' 016767  176640  177262  LISTPM: MOV     INTER,PASSCT
001366' 032767  000010  176422          BIT     #BIT3,SR1       ;TEST INHIBIT TEST BIT
001374' 001117                          BNE     DONE            ;BR IF SET
001376' 016777  176646  176642          MOV     OVFIT1,@OVFINT  ;LOAD WORD OVERFLOW VECTOR
001404' 005077  176640                  CLR     @OVFIT1
001410' 012777  001502' 176634          MOV     #ZOVSR,@ZOVINT  ;LOAD Z OVERFLOW VECTOR
001416' 116777  176370  176630          MOVB    BR1,@ZOVITI     ;LOAD BR LEVEL
001424' 012777  000102  176512          MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
001432' 016777  000206  176574  CONTA:  MOV     XYMIRR,@XYHOLD  ;INSERT FULL WORDS
001440' 012777  000404  176562          MOV     #404,@CMDCSR    ;LIST MODE
001446' 012777  174000  176562          MOV     #-2048.,@WCSHI  ;WORD COUNT = -2048
001454' 016777  176546  176556          MOV     RBUFVA,@CAZLO   ;CURRENT ADDRESS - BUFFER
001462' 017767  176546  177156          MOV     @ADDREG,JUNK    ;CLEAR BUFFERS
001470' 052777  004001  176532          BIS     #4001,@CMDCSR   ;ENABLE INTERRUPT AND GO
001476' 104400  000000'                 EXITS,BEGIN             ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
001502' 012777  000102  176534  ZOVSR:  MOV     #CLALL,@SFUNC   ;CLEAR INTERFACE
                                        ;-------------------------------------------------------------------
001510' 000004  000000' 001516'         PIRQS,BEGIN,1S          ; QUEUE UP TO CONTINUE AT 1S AND RTI
                                        ;-------------------------------------------------------------------
001516' 017767  176506  176356  1S:     MOV     @CMDCSR,ACSR    ;LOOK FOR NPR ERROR
001524' 100010                          BPL     2S
001526' 012767  000032  176352          MOV     #32,ERRTYP      ;NPR ERROR
                                        ;*******************************************************************
001534' 104405  000000' 000000          HRDERS,BEGIN,NULL       ;NPR ERROR FLAG SET
                                        ;*******************************************************************
001542' 104410  000000'                 ENDS,BEGIN              ;
001546' 012767  001652' 176330  2S:     MOV     #BUFFER,WASADR  ;LOAD "WAS ADDRESS"
001554' 016767  000064  176324          MOV     XYMIRR,ASB      ;LOAD "SHOULD BE" DATA
001562' 017767  176316  176320  3S:     MOV     @WASADR,AWAS    ;GET DATA WORD
001570' 026767  176314  176310          CMP     AWAS,ASB        ;CHECK FOR PROPER DATA
001576' 001402                          BEQ     4S              ;BR IF CORRECT
                                        ;*******************************************************************
001600' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*******************************************************************
001604' 062767  000002  176272  4S:     ADD     #2,WASADR       ;GO TO NEXT LOCATION
001612' 026727  176266  002052'         CMP     WASADR,#BUFFER+200      ;BUT NOT TOO FAR
001620' 001360                          BNE     3S
001622' 005167  000016                  COM     XYMIRR          ;CHANGE DATA PATTERN
001626' 005367  177016                  DEC     PASSCT
001632' 001277                          BNE     CONTA           ;PRIME WORD NPR AGAIN
001634'                         DONE:
001634' 104413  000000'                 ENDITS,BEGIN            ;SIGNAL END OF ITERATION.
                                                                ;MONITOR SHALL TEST END OF PASS
001640' 000167  176434                  JMP     RESTRT

001644' 052525          XYMIRR: 52525                   ;MIRROR OF X-Y HOLD
001646' 001653'         TARGET: BUFFER                  ;2048 WORD BOUNDARY POINTER INTO BUFFER AREA
001650' 001653'         TARGT1: BUFFER+1                ;2048 ODD-BYTE BOUNDARY POINTER INTO BUFFER
001652' 000000          BUFFER: 0
001654' 004000                  .BLKW   2048.
011654' 000000          LAST:   0
        000001          .END
```

```
ACSR    000102R          194#  331*  369*  407*  446*
ADDR    000006R          160#  249
ADDREG  000234R          233#  316   318   360   362   398   400   439
ADDR22= 001000           212#
ASB     000106R          198#  339*  342   377*  379   416*  417   454*  456
ASTAT   000104R          196#
AWAS    000110R          199#  341*  342   378*  379   415*  417   455*  456
BEGIN   000000R          157#  288   289   295   298   319   329   335   337   345   363   367   373
                         175   392   401   405   411   413   420   441   444   450   452   459   468
BIT0    = 000001         212#  311
BIT1    = 000002         212#  354
BIT10   = 002000         212#
BIT11   = 004000         212#  266   269   273   285
BIT12   = 010000         212#
BIT13   = 020000         212#
BIT14   = 040000         212#
BIT15   = 100000         212#
BIT2    = 000004         212#  393
BIT3    = 000010         212#  428
BIT4    = 000020         212#
BIT5    = 000040         212#
BIT6    = 000100         212#
BIT7    = 000200         212#
BIT8    = 000400         212#
BIT9    = 001000         212#
BOUND   000354R          265#
BREAKS= 104407           212#  288   289
BR1     000012R          162#  306   433
BR2     000013R          163#
BTOD$ = 104421           212#
BUFFER  001652R          216   453   462   472   473   474#
CAZLO   000240R          235#  438*
CDATA$= 104412           212#
CLALL   = 000163         226#  261   292   300   310   327   365   403   434   442
CLCOVF= 000002           221#
CLDP    = 000010         223#
CLHLD   = 000001         220#
CLZOVF= 000020           224#
CMDCSR  000230R          230#  250   262   284*  314*  317*  331   357*  361*  369   396*  399*  407
                         436*  440*  446
CONFIG  000056R          182#
CONTA   001432R          436#  466
COVT    = 000004         222#
CSRA    000100R          192#  262*
DATCK$= 104411           212#
DATER$= 104404           212#  345   382   420   459
DONE    001634R          429   467#
DVID1   000014R          164#
ENDIT$= 104413           212#  468
END$    = 104410         212#  298   337   375   413   452
ERRTYP  000106R          197#  293*  333*  371*  409*  448*
EVBYOK  001210R          395#  424
EVBYPM  001164R          355   391#
EVINT   001256R          392   403#
EXIT$ = 104400           212#  319   363   401   441
GETPA$= 104415           212#
```



```
GWBUF$= 104414           212#
HRDCNT  000044R          177#
HRDERS= 104405           212#  295   373   411   450
HRDPAS  000050R          179#
ICONT   000036R          174#
ICOUNT  000040R          175#
IDNUM   000122R          204#
INIT    000030R          171#
INTER   000224R          212#  309   352   391   427
INTR    000120R          203#  246*
JUNK    000646R          316*  318*  321#  360*  362*  398*  400*  439*
LAST    011654R          476#  427#
LISTEM  001360R          394
MAP22S= 104415           212#
MISCFU  000242R          236#
MODNAM  000006R          158#
MODSP   000224R          152   210#
MSGN$ = 104403           212#
MSGS$ = 104402           212#
MSG$  = 104401           212#
NULL  = 000000           212#  295   335   373   411   450
ODBYOK  001006R          356#  386
ODDINT  001062R          353   365#
ODDPRM  000762R          312   352#
OFFSET  000232R          231#  283*  315*  359*  397*
OPEN    = 000000         159   165   166   167   168   185   186   187   188   189   190   191   192
                         194   196   198   199   201   202   203   212#
OTOA$ = 104420           212#
OVFINT  000246R          239#  253   305*  353*  392*  430*
OVFIT1  000250R          240#  306*  430   431*
OVINIT  000424R          282#
OVSR    000652R          305   327#
PASCNT  000034R          173#
PASSCT  000650R          309*  322#  347*  352*  385*  391*  423*  427*  465*
PIRQ$ = 000004           212#  329   367   405   444
POPSP = 005726           212#
POPSP2= 022626           212#
PRIME   000526R          305#
PRTY    = 000000         212#
PRTY0   = 000000         163   212#
PRTY1   = 000040         212#
PRTY2   = 000100         212#
PRTY3   = 000140         212#
PRTY4   = 000200         212#
PRTY5   = 000240         212#
PRTY6   = 000300         212#
PRTY7   = 000340         162   212#
PS      = 177776         212#
PSW     = 177776         212#
PUSH    = 005746         212#
PUSH2   = 024646         212#
RAND$ = 104417           212#
RANNUM  000054R          181#
RBUFVA  000226R          216#  265   438
RESTRT  000300R          200   249#  470
RES1    000056R          183#
```

```
RES2    000060R          184#
RSTRT   000112R          200#
SBADR   000102R          193#
SFUNC   000244R          237#    261*    292*    300*    310*    327*    365*    403*    434*    442*
SOFCNT  000042R          176#
SOFERS= 104406           212#    335
SOFPAS  000046R          178#
SPOINT  000032R          172#
SPSIZ = 000040             1#    205
SR1     000016R          165#    311     354     393     428
SR2     000020R          166#
SR3     000022R          167#
SR4     000024R          168#
START   000256R          171     246#
STAT    000026R          170#
STW   = 000040           225#
SVR0    000062R          185#
SVR1    000064R          186#
SVR2    000066R          187#
SVR3    000070R          188#
SVR4    000072R          189#
SVR5    000074R          190#
SVR6    000076R          191#
SYSCNT  000052R          180#
TARGET  001646R          275*    282*    283     286     313*    315     340     341     356*    359     378     395*    397
                         415     472#
TARGT1  001650R          276*    277*    473#
TRPDFD= 000022           212#
VECTOR  000010R          161#    255
WASADR  000104R          195#    340*    453*    455     461*    462
WCSHI   000236R          234#    437*
WDFR    000116R          202#    248*
WDTO    000114R          201#    247*
WORDBK  000600R          313#    348
XFLAG   000005R          159#
XYHOLD  000234R          232#    358*    435*
XYMIRR  001644R          435     454     464*    471#
ZOVINT  000252R          241#    307*    432*
ZOVIT1  000254R          242#    258     307     308*    433*
ZOVSR   001502R          432     442#
.     = 011656R          475#

. ABS.  000000    000
        011656    001

ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

XNCAD0,XNCAD0/SQL/CRF:SYM=DDXCOM,XNCAD0
RUN-TIME:  1  1  .3 SECONDS
RUN-TIME RATIO: 21/3=6.0
CORE USED:  7K  (13 PAGES)
```