SEQ 0001

```
.REM _
```

IDENTIFICATION

PRODUCT CODE: AC-E803G-MC

PRODUCT NAME: CXDNAG0 DN11 MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

SEQ 0002

1. ABSTRACT:

DNA IS A IOMOD THAT EXERCISES UP TO 16. POSSIBLE DN11'S.
EACH DN11 IS CHECKED IN MAINTENANCE MODE BY SETTING
UP THE DN CSR TO CAUSE AN INTERRUPT. ALL IMPROPER
INTERRUPTS ARE REPORTED AS ERRORS

2. REQUIREMENTS:

HARDWARE: AT LEAST ONE DN11 CONTROL UNIT

STORAGE:: DNA REQUIRES:
1. DECIMAL WORDS: 275
2. OCTAL WORDS: 0423
3. OCTAL BYTES: 1046

3. PASS DEFINITION:

ONE INTERNAL PASS OF DNA CONSISTS OF GENERATING 4N INTERRUPTS
PER PASS WHERE N= NO. OF SELECTED DN11'S. THIS IS REPEATED
7000(8) TIMES FOR EACH END OF PASS CALL.

4. EXECUTION TIME:

DNA RUNNING ALONE WITH ONE DN11 SELECTED ON A PDP11/20
PROCESSOR TAKES ONE HALF MINUTE TO EXECUTE A PASS.
EACH ADDITIONAL LINE WILL CAUSE AN INCREASE IN TIME.

5. CONFIGURATION REQUIREMENTS:

DEFAULT PARAMETERS:

DEVADR: 175200, VECTOR: 1, BR1: 4, DEVCNT: 1

REQUIRED PARAMETERS:

AT CONFIGURATION TIME "VECTOR" MUST BE MODIFIED

6. DEVICE/OPTION SET-UP

NONE

7. MODULE OPERATION:

TEST SEQUENCE:

A. SET UP THE ASSIGNED VECTOR FOR ALL SELECTED DEVICES
B. GENERATE A DN11 CSR ADDRESS FOR A SELECTED DEVICE

SEQ 0003

C. GENERATE A PRESENT NEXT DIGIT INTERRUPT AND EXIT
D. SERVICE PND INTERRUPT - REPORT ANY ERROR
E. GENERATE A DATA SET STATUS INTERRUPT AND EXIT
F. SERVICE DSS INTERRUPT - REPORT ANY ERROR
G. GENERATE A POWER OFF INTERRUPT AND EXIT
H. SERVICE PO INTERRUPT - REPORT ANY ERROR
I. GENERATE AN ABANDON CALL INTERRUPT AND EXIT
J. SERVICE ABC INTERRUPT - REPORT ANY ERROR
K. IF ANY DN11 LEFT ; REPEAT B THRU J
L. COUNT DOWN INTPSC FROM 2000, WHEN ITS 0 CALL ENDPAS
ELSE GO BACK TO B.

FAILURE TO GENERATE AN INTERRUPT WILL HANG DNA AND
PREVENT END OF PASS PRINTOUT.

8. OPERATION OPTIONS

MODIFYING DIVD1 ALLOWS EXERCISING ANY
COMBINATION OF DN11'S:

DVID1,BIT0=DEV0;DVID1,BIT1;=DEV1.....DVID1,BIT15=DEV15

IF DVID1=0 THEN DNA WILL BE DROPPED FROM THE EXERCISE

9. NON-STANDARD PRINTOUTS:

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

```
                ;DN11 DEC/X11 EXERCISER MODULE

000000'         IOMOD   <DNAG >,175200,1,4,,7000,37
000000'         MODULE  140000,DNAG ,175200,1,4,,7000,37
                .TITLE  DNAG DEC/X11 SYSTEM EXERCISER MODULE
                ;       DDXCOM  VERSION  6      23-MAY-78
                                .LIST   BIN
                                ;*****************************************************************************
000000'                         BEGIN:
000000' 047104  043501      040 MODNAM: .ASCII  /DNAG / ;MODULE NAME.
000005'     000                 XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 175200                  ADDR:   175200+0                ;1ST DEVICE ADDR.
000010' 000001                  VECTOR: 1+0                     ;1ST DEVICE VECTOR.
000012'     200                 BR1:    .BYTE   PRTY4+0         ;1ST BR LEVEL.
000013'     000                 BR2:    .BYTE   PRTY+0          ;2ND BR LEVEL.
000014' 000001                  DVID1:  +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' 007000                  ICONT:  7000                    ;# OF ITERATIONS PER PASS=7000
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:
000056' 000000                  RES1:   0                       ;RESERVED FOR MONITOR USE
000060' 000000                  RES2:   0                       ;RESERVED FOR MONITOR USE
000062' 000000                  RES3:   0                       ;RESERVED FOR MONITOR USE
000064' 000000                  SVR0:   OPEN                    ;LOC TO SAVE R0.
000066' 000000                  SVR1:   OPEN                    ;LOC TO SAVE R1.
000070' 000000                  SVR2:   OPEN                    ;LOC TO SAVE R2.
000072' 000000                  SVR3:   OPEN                    ;LOC TO SAVE R3.
000074' 000000                  SVR4:   OPEN                    ;LOC TO SAVE R4.
000076' 000000                  SVR5:   OPEN                    ;LOC TO SAVE R5.
000100' 000000                  SVR6:   OPEN                    ;LOC TO SAVE R6.
000102'                         CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
000102' 000000                  SBADR:                          ;ADDR OF GOOD DATA, OR
000104'                         ACSR:   OPEN                    ;CONTENTS OF CSR.
000104' 000000                  WASADR:                         ;ADDR OF BAD DATA, OR
000106'                         ASTAT:  OPEN                    ;STATUS REG CONTENTS.
000106' 000000                  ERRTYP:                         ;TYPE OF ERROR
000110' 000000                  ASB:    OPEN                    ;EXPECTED DATA.
000112' 000274'                 AWAS:   OPEN                    ;ACTUAL DATA.
000114' 000000                  RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
000116' 000000                  WDTO:   OPEN                    ;WORDS TO MEMORY PER ITERATION
000120' 000000                  WDFR:   OPEN                    ;WORDS FROM MEMORY PER ITERATION
                                INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
```



```
000122' 000037                  IDNUM:  37                      ;MODULE IDENTIFICATION NUMBER=37
        000040                          .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
                                        .NLIST
                                        .WORD   0
                                        .LIST
                                        .ENDR
000224'                         MODSP:
                                ;*****************************************************************************
                                ;MODULE INITIALIZATION ROUTINES

                                ;ROUTINE TO SET UP VECTORS FOR ALL SELECTED DN11'S

000224' 012767  000002  177664  START:  MOV     #2,WDFR         ;AT LEAST 2 WORDS FROM MEM/ITERATION
000232' 012767  000004  177660          MOV     #4,INTR         ;AT LEAST 4 INTERRUPTS/ITERATION
000240' 016700  177550                  MOV     DVID1,R0        ;SELECTED UNITS TO R0
000244' 001002                          BNE     1$              ;IF SOME SELECTED - GO TO WORK
000246' 104410  000000'                 ENDS$,BEGIN             ;
000252' 006200                  1$:     ASR     R0              ;SHIFT DEV COUNT
000254' 001407                          BEQ     RESTRT          ;IF NO MORE BRANCH OUT
000256' 062767  000004  177634          ADD     #4,INTR         ;DOUBLE INTERRUPTS
000264' 062767  000002  177624          ADD     #2,WDFR         ;DOUBLE WDFR
000272' 000767                          BR      1$              ;GO BACK AND CHECK FOR MORE

000274' 016702  177510  RESTRT: MOV     VECTOR,R2       ;START AT BEGINNING OF VECTOR AREA
000300' 016700  177510          MOV     DVID1,R0        ;GET 1ST SELECTION PARAMETER

000304' 012701  000017  ST2:    MOV     #17,R1          ;SET UP GROUP OF FOUR MASK
000310' 030100  1$:     BIT     R1,R0           ;TEST FOR ANY DN11 IN GROUP
000312' 001010          BNE     3$              ;GO SET UP VECTOR
000314' 022222          CMP     (R2)+,(R2)+     ;UPDATE VECTOR POINTER
000316' 005701  2$:     TST     R1              ;END OF 1ST GROUP OF 4 VECTORS
000320' 100412          BMI     DNMON           ;BR IF DONE FOUR VECTORS
000322' 006301          ASL     R1              ;SHIFT MASK FOR TESTING NXT. GROUP OF 4
000324' 006301          ASL     R1
000326' 006301          ASL     R1
000330' 006301          ASL     R1
000332' 000766          BR      1$
000334' 012722  000676' 3$:     MOV     #DNISR,(R2)+    ;POINT DN11 INTR. TO DNISR
000340' 012722  177446          MOV     BR1,(R2)+       ;SET UP PRIORITY LEVEL
000344' 000764          BR      2$              ;GO TEST FOR LAST VECTOR

                                ;ROUTINE TO CONTROL EXERCISING ALL SELECTED DEVICES

000346' 016701  177434  DNMON:  MOV     ADDR,R1         ;GET 1ST DN11 CSR ADDRESS
000352' 010167  000466          MOV     R1,BASE         ;SET THE BASE ADDRESS OF A GROUP OF FOUR
000356' 012767  000004  000456  MOV     #4,GROUP        ;SET COUNT FOR NUMBER OF DN'S IN A GROUP
000364' 016700  177424          MOV     DVID1,R0        ;GET  SELECTION PARAMETER
000370' 006200          DN5:    ASR     R0              ;SHIFT SELECT BIT INTO"C"
000372' 100002          BPL     1$              ;DON'T PROPAGATE BAD FLAG
000374' 042700  100000          BIC     #BIT15,R0       ;CLEAR IT OUT INSTEAD
000400' 103435          1$:     BCS     DN4             ;BR IF SELECTED
000402' 001417          BEQ     PASS            ;IF ALL DONE,SIGNAL END OF PASS
000404' 005367  000432          DEC     GROUP           ;REDUCE THE COUNT IN THIS GROUP
000410' 001011          BNE     DN3             ;IF END OF GROUP, RESET THE COUNT
000412' 012767  000004  000422  MOV     #4,GROUP
```

```
000420' 042777  000004  000416          BIC     #BIT2,@BASE     ;DONT LEAVE ANY INTERRUPTS ENABLED
000426' 062767  000010  000410          ADD     #10,BASE        ;UPDATE THE BASE ADDRESS
000434' 062701  000002          DN3:    ADD     #2,R1           ;UPDATE R1 TO POINT TO NEXT DN11
000440' 000753                  PASS:   BR      DN5             ;GO TEST NEXT DN11 IN THIS GROUP
000442' 022767  000004  000372          CMP     #4,GROUP        ;IS THIS ADDRESS ASSIGNED TO DN11?
000450' 001003                          BNE     1$              ;IF YES, GO CLEAR THE INTERRUPT
000452' 162767  000010  000364          SUB     #10,BASE        ;IF NO, POINT TO THE LAST ASSIGNED MASTER DN11
000460' 042777  000004  000356  1$:     BIC     #BIT2,@BASE     ;DISABLE MASTER INTERRUPT
000466' 104413  000000'                 ENDIT$,BEGIN            ;SIGNAL END OF ITERATION.
                                                                ;MONITOR SHALL TEST END OF PASS
000472' 000700                          BR      RESTRT          ;BR IF NOT
000474' 010167  000334          DN4:    MOV     R1,DNCSR        ;SET UP SELECTED DN11 ADDRESS
000500' 052711  000004                  BIS     #4,(R1)         ;TURN ON THE MASTER INTR. ENABLE


                                ;ROUTINES TO EXERCISE A SELECTED DN11

                                        ;TEST TO SEE IF PRESENT NEXT DIGIT CAUSES AN INTERRUPT
000504' 012767  000550' 000324  6$:     MOV     #DN11A,FORK     ;SET UP TO DO DN11A NEXT
000512' 052711  000511                  BIS     #511,(R1)       ;LOAD THE CSR
000516' 005067  000316                  CLR     FLAG            ;CLEAR DEVICE COMPLETION FLAG
000522' 005004                          CLR     R4                      ;CLEAR TIMER COUNTER
000524'                         7$:
000524' 104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR...
000530' 104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
000534' 005767  000300                  TST     FLAG            ;ARE THE INTERRUPTS FINISHED?
000540' 001335                          BNE     DN3             ;IF ALL DONE GO TO NEXT DEVICE
000542' 005304                          DEC     R4              ;IF NOT REDUCE COUNT
000544' 001367                          BNE     7$              ;BREAK AGAIN IF TIME HAS NOT EXPIRED
000546' 000503                          BR      HUNG            ;IF TIMED OUT, GO REPORT BAD DEVICE


                                        ;TEST TO SEE IF DATA SET STATUS CAUSES AN INTERRUPT
000550' 012767  000570' 000260  DN11A:  MOV     #DN11B,FORK     ;SET UP TO DO DN11B NEXT
000556' 052777  001111  000250          BIS     #1111,@DNCSR    ;LOAD THE CSR
000564' 104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.


                                        ;TEST TO SEE IF FORCING POWER OFF CAUSES AN INTERRUPT
000570' 012767  000616' 000240  DN11B:  MOV     #DN11C,FORK     ;SET UP TO DO DN11C NEXT
000576' 052777  000111  000230          BIS     #111,@DNCSR     ;LOAD THE CSR
000604' 052777  002000  000222          BIS     #2000,@DNCSR    ;GO
000612' 104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.


                                        ;TEST TO SEE IF FORCING ABANDON CALL CAUSES INTERRUPT
000616' 012767  000636' 000212  DN11C:  MOV     #DN11D,FORK     ;SET UP TO RETURN TO EXERCISE NXT DN11
000624' 052777  004111  000202          BIS     #4111,@DNCSR    ;LOAD THE CSR
000632' 104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.


000636' 005367  000200          DN11D:  DEC     GROUP           ;REDUCE THE COUNT OF DEVICES IN THIS GROUP
000642' 001011                          BNE     1$              ;IF COUNT NOT EXCEEDED , CONTINUE
000644' 012767  000004  000170          MOV     #4,GROUP        ;OTHERWISE, RESET IT
000652' 042777  000004  000164          BIC     #BIT2,@BASE     ;CLEAR MASTER ENABLE
000660' 062767  000010  000156          ADD     #10,BASE        ;UPDATE THE BASE ADDRESS
```



```
000666' 005167  000146  1$:     COM     FLAG            ;SET DEVICE COMPLETION FLAG
000672' 104400  000000'          EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.


                                ;INTERRUPT SERVICE ROUTINES

000676'                         DNISR:
                                ;-----------------------------------------------------------------
000676' 000004  000000' 000704' ;IRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
                                ;-----------------------------------------------------------------


000704' 105777  000124  1$:     TSTB    @DNCSR          ;DONE SET ??
000710' 100005                  BPL     3$              ;BR IF NOT
000712' 042777  177773  000114  2$:     BIC     #177773,@DNCSR  ;CLEAR ALL BITS IN THE CSR EXCEPT BIT02
                                                        ;IN CASE IT IS THE FIRST ONE IN A GROUP
                                                        ;OF 4 (MASTER INTR. ENAB. MUST STAY ON)
000720' 000177  000112          JMP     @FORK           ;RETURN TO NEXT TEST VIA BRANCH FORK
000724' 016767  000104  177146  3$:     MOV     DNCSR,CSRA      ;SAVE THE ADDRESS OF THE CSR
000732' 017767  000076  177142          MOV     @DNCSR,ACSR     ;SAVE THE CONTENTS OF THE CSR
000740' 012767  000011  177140          MOV     #11,ERRTYP      ;ILLEGAL INTERRUPT
                                ;*****************************************************************
000746' 104405  000000' 000000          ORDER$,BEGIN,NULL       ;FALSE INTERRUPT
                                ;*****************************************************************
000754' 000756                  BR      2$              ;RETURN TO NEXT TEST


000756' 010167  000024  HUNG:   MOV     R1,TEMP         ;LOAD DEVICE ADDRESS FOR OCTAL-ASCII CONVRT.
                                ;*****************************************************************
                                                        ;CONVERT TEMP TO ASCII AND
                                                        ;STORE AT M2
000762' 104420  000000' 001006'         OTOA$,BEGIN,TEMP,M2

                                ;*****************************************************************
000772' 104403  000000' 001002'         MSGN$,BEGIN,FAIL        ;ASCII MESSAGE CALL WITH COMMON HEADER
001000' 000615                  BR      DN3                     ;GO DO NEXT ONE
001002' 001010'                 FAIL:   M1
001004' 177777                          177777


001006' 000000                  TEMP:   OPEN
001010' 042045  053105  041511  M1:     .ASCII  '%DEVICE '
001016' 020105
001020' 000000                  M2:     OPEN
001022' 000000                          OPEN
001024' 000000                          OPEN
001026' 044040  047125  000107  M3:     .ASCIZ  ' HUNG'


                                ;SOME DN MODULE VARIABLES

001034' 000000                  DNCSR:  OPEN            ;CONTAINS THE ADDRESS OF THE CONTROL
                                                        ;REG. OF THE DN11 UNDER TEST
001036' 000000                  FORK:   OPEN            ;STEERS INTR. SERVICE TO NEXT TEST
001040' 000000                  FLAG:   OPEN            ;DEVICE COMPLETION FLAG
001042' 000000                  GROUP:  OPEN            ;COUNT OF DN11'S IN A GROUP
001044' 000000                  BASE:   OPEN            ;BASE ADDRESS OF A GROUP OF FOUR DN11'S
```

```
   354
   355          000001                  .END
```



```
ACSR     000102R          176#    315*
ADDR     000006R          142#    230
ADDR22=  001000           194#
ASB      000106R          180#
ASTAT    000104R          178#
AWAS     000110R          181#
BASE     001044R          231*    242*    243*    248*    249*    296*    297*    353#
BEGIN    000000R          139#    203     250     266     267     278     285     291     299     305     318     329     331
BIT0  =  000001           194#
BIT1  =  000002           194#
BIT10 =  002000           194#
BIT11 =  004000           194#
BIT12 =  010000           194#
BIT13 =  020000           194#
BIT14 =  040000           194#
BIT15 =  100000           194#    236
BIT2  =  000004           194#    242     249     296
BIT3  =  000010           194#
BIT4  =  000020           194#
BIT5  =  000040           194#
BIT6  =  000100           194#
BIT7  =  000200           194#
BIT8  =  000400           194#
BIT9  =  001000           194#
BREAK$=  104407           194#    266     267
BR1      000012R          145#    225
BR2      000013R          145#
BTOD$ =  104421           194#
CDATA$=  104412           194#
CONFIG   000056R          164#
CSRA     000100R          174#    314*
DATCK$=  104411           194#
DATER$=  104404           194#
DNCSR    001034R          153#    277*    283*    284*    290*    308     310*    314     315     348#
DNISR    000676R          224     303#
DNMON    000346R          218     230#
DN11A    000550R          261     276#
DN11B    000570R          276     282#
DN11C    000616R          282     289#
DN11D    000636R          289     293#
DN3      000434R          240     244#    269     332
DN4      000474R          237     253#
DN5      000370R          234#    245
DVID1    000014R          146#    201     211     233
ENDIT$=  104413           194#    250
END$  =  104410           194#    203
ERRTYP   000106R          179#    316*
EXIT$ =  104400           194#    278     285     291     299
FAIL     001002R          331     333#
FLAG     001040R          263*    268     298*    351#
FORK     001036R          261*    276*    282*    289*    313     350#
GETPA$=  104415           194#
GROUP    001042R          232*    239*    241*    246     293*    295*    352#
GWBUF$=  104414           194#
HRDCNT   000044R          159#
HRDER$=  104405           194#    318
```

```
HRDPAS   000950R          161#
HUNG     000756R          272     324#
ICONT    000036R          156#
ICOUNT   000040R          157#
IDNUM    000122R          186#
INIT     000030R          153#
INTR     000120R          185#    200*    206*
MAP22$=  104416           194#
MODNAM   000000R          140#
MODSP    000224R          154     192#
MSGN$  = 104403           194#    331
MSGS$  = 104402           194#
MSG$   = 104401           194#
M1       001010R          333     337#
M2       001020R          338     339#
M3       001026R          342#
NULL   = 000000           194#    318
OPEN   = 000000           141     147     148     149     150     167     168     169     170     171     172     173     174
                          176     178     180     181     183     184     185     194#    336     339     340     341     348
                          350     351     352     353
OTOA$  = 104420           194#    328
PASCNT   000034R          155#
PASS     000442R          238     246#
PIRQ$  = 000004           194#    305
POPSP  = 005726           194#
POPSP2 = 022626           194#
PRTY   = 000000           145     194#
PRTY0  = 000000           194#
PRTY1  = 000040           194#
PRTY2  = 000100           194#
PRTY3  = 000140           194#
PRTY4  = 000200           142     194#
PRTY5  = 000240           194#
PRTY6  = 000300           194#
PRTY7  = 000340           194#
PS     = 177776           194#
PSW    = 177776           194#
PUSH   = 005746           194#
PUSH2  = 024646           194#
RAND$  = 104417           194#
RANNUM   000054R          163#
RESTRT   000274R          182     205     210#    252
RES1     000056R          165#
RES2     000060R          166#
RSTRT    000112R          182#
SBADR    000102R          175#
SOFCNT   000042R          158#
SOFER$=  104406           194#
SOFPAS   000046R          160#
SPOINT   000032R          154#
SPSIZ  = 000040             1#    187
SR1      000016R          147#
SR2      000020R          148#
SR3      000022R          149#
SR4      000024R          150#
START    000224R          153     199#
```



```
STAT     000026R          152#
ST2      000304R          213#
SVR0     000062R          167#
SVR1     000064R          168#
SVR2     000066R          169#
SVR3     000070R          170#
SVR4     000072R          171#
SVR5     000074R          172#
SVR6     000076R          173#
SYSCNT   000052R          162#
TEMP     001006R          324#    328     336#
TRPDFD=  000022           194#
VECTOR   000010R          143#    210
WASADR   000104R          177#
WDFR     000116R          184#    199*    207*
WDTO     000114R          183#
XFLAG    000005R          141#

. ABS.   000000      000
         001046      001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XDNAG0,XDNAG0/SOL/CRF:SYM=DDXCOM,XDNAG0
 RUN-TIME:  1  1  .2 SECONDS
 RUN-TIME RATIO: 10/3=3.3
 CORE USED:  7K  (13 PAGES)
```

DIAGNOSTIC ENGINEERING

digital **DECO☐ DEPO☒ SUBMISSION☐**

FOR RELEASE ENG. USE ☐ NEW ☐ CHANGE ☐ DELETE

PRODUCT IDENTIFICATION

MD	LIBRARY	PRODUCT NUMBER	REV	PATCH	ECO TALLY	PRODUCT DATE (DD MMM YY)	STATUS	DISTRIBUTION	1ST COPY-RIGHT YEAR	LAST COPY-RIGHT YEAR
	ZZ	CXDNA	G	1	Ø1	5, JAN 79	☐ OBSOLETE	XX G ☐ R	1973	1979

TITLE: CXDNAG1 DN11 MODULE

AUTHOR	MAINTAINING GROUP	MAINTAINER	SUBMITTING ENGINEER
D.BUTENHOF	DEC/X11 SPT GRP	D.BUTENHOF	D.BUTENHOF

PRODUCT COMPONENTS

CK	DESCRIPTION	PRODUCT NO.	REV	CK	DESCRIPTION	PRODUCT NO	REV
	DOCUMENT				INDEX		
	LISTING				SOURCE MEDIA		
	OBJECT MEDIA				TEST MEDIA		
X		AF-E8Ø3G-M1					

PRODUCTS OBSOLETED (other than previous version)

MD	LIBRARY	PRODUCT NUMBER	REV	MD	LIBRARY	PRODUCT NUMBER	REV	MD	LIBRARY	PRODUCT NUMBER	REV

PRODUCT CHARACTERISTICS

PROCESSORS PRODUCT OPERATES WITH (Enter all applicable 2 digit codes representing the Processor the product operates with. See separate instructions.)

03 04 05 10 20 2134 35 40 45 5055 60 70

OPERATIONAL CODES (Enter all applicable 2-digit codes that describe the product. See separate instructions.)

02 03 0406 50

ACT/APT/XXDP INFORMATION FIELD		EXT	ACT SEQ NUMBER	ACT/XXDP COMPATIBLE?	APT COMPATIBLE?	1ST PASS RUN TIME	SUBSEQUENT PASS RUN TIME
				☒ Y ☐ N	☒ Y ☐ N	30 SECONDS	30 SECONDS

DECO/DEPO INFORMATION

PROBLEM REPORTS CLOSED:

ICE AFFECTED: DEC/X11 MULTIMEDIA AFFECTED? ☐ YES ☐ NO

KIT NUMBERS				
~~ZJ129-RZ,FR~~	ZJ239-RZ,PB	ZJ240-RB,RE	ZJ240-FR
ZJ239-RB,RY	ZJ239-RB,FR	ZJ240-RZ,RB	ZJ130-RB

PROBLEM:

ASR of device count word prevented completion of test (replication of bit 15 if 16th DN11 selected)

SOLUTION:

due to program logic, at shifts "C" bit is Ø, so it is safe to patch ASR's to ROR's.

DEPO PATCH AREA

CHANGE LOC	FROM	TO	CHANGE LOC	FROM	TO
252	62ØØ	6ØØØ			
37Ø	62ØØ	6ØØØ			

SUBMITTING ENGINEER	MANUFACTURING ENGINEER	SUPPORT ENGINEER	CHARGE DECO/DEPO TO DISCRETE PROJECT NUMBER
DATE	DATE	DATE	Q98-C-[illegible]
[illegible]	ELD SERVICE	MATERIALS MANAGER	COORDINATION NO. mc# 2807
22-Dec 78	DATE	DATE	