```
                        .SBTTL DOCUMENTATION
                        .REM _
```

IDENTIFICATION

PRODUCT CODE: AC-E881B-MC

PRODUCT NAME: CXBBAB0 KIT-11D MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

```
                        1.      ABSTRACT
                                ********

                                BBA IS AN IOMOD THAT EXERCISES THE KIT11D.  DURING A
                                SINGLE PASS, IT CHECKS THAT THE KIT WILL PERFORM A PROGRAMMED
                                DATA TRANSFER (PDT) FOLLOWED BY A NON PROCESSOR REQUEST
                                (NPR) TRANSFER.  ANY ERRORS DETECTED DURING THE PASS ARE
                                REPORTED ON THE CONSOLE TTY.  THE PROGRAM WILL RUN INDEFINITELY
                                UNTIL HALTED BY THE USER (PROVIDED THAT HARD FAIL SYSTEM
                                ERRORS HAVE NOT BEEN ENCOUNTERED).


                        2.      REQUIREMENTS
                                ************

                                HARDWARE:       KIT11D AND AA SPECIAL WRAP-AROUND
                                                CABLE.
                                STORAGE:: BBA REQUIRES:
                                                1. DECIMAL WORDS: 476
                                                2. OCTAL WORDS: 0734
                                                3. OCTAL BYTES: 1670


                        3.      PASS DEFINITION
                                ***************

                                ONE PASS OF THE KITDQ CONSISTS OF 70000 ITERATIONS 1
                                PROGRAMMED DATA TRANSFER AND ONE 64 WORDS.


                        4.      EXECUTION TIME
                                **************

                                RUNNING ALONE ON A PDP11/05 TAKES APPROXIMATELY 1
                                MINUTE PER PASS.


                        5.      CONFIGURATION REQUIREMENTS
                                **************************

                                DEFAULT PARAMETERS:

                                BR1:    5

                                BR2:    0

                                REQUIRED PARAMETERS:
                                NONE, EXCEPT FOR THE ADDRESS AND VECTOR MUST BE ASSIGNED
                                AT CONFIGURATION TIME(OR RUN TIME VIA THE MOD COMMAND).
                                THE PROGRAM WILL NOT RUN UNLESS YOU SUPPLY T
                                ADDRESS AND FLOATING VECTOR.

                                NONE
```

```
6.      DEVICE/OPTION SETUP
        ********************

        A.  THE KIT11D UNDER TEST MUST CONTAIN THE SPECIAL
            ***** WRAP-AROUND CABLE FOR THIS TEST. *****

        B.  THE VECTOR ADDRESS MAY BE SPECIFIED AT RUN TIME BY USING
            THE FOLLOWING MODIFY COMMAND AT RUN TIME:

                .MODIFY KITD0 10(CR)

            THE SYSTEM WILL ISSUE A CR/LF AND PRINT OUT THE
            PRESENT CONTENTS OF THAT LOCATION (CALLED VECTOR).

            AT THAT TIME, YOU MAY ENTER THE NEW VECTOR ADDRESS AND
            FOLLOW IT BY A CARRIAGE RETURN (CR).


                IF A DEVICE ADDRESS OTHER THAN 160000 IS USED.
                THAT ADDRESS MAY BE UPDATED BY USING THE
                MODIFY COMMAND:

                        .MODIFY KTD0 6 <CR>

        C.  THE MODULE IS SELECTED AT RUN TIME BY TYPING:
                .SEL KITD0 <CR>

        D.  THE MODULE IS RUN BY TYPING:
                .RUN <CR>
            AT RUN TIME.

        E.  A LOAD MAP IS ORTAINED AT RUN TIME BY SAYING:
                .MAP <CR>

        F.  PROGRAMMERS GUIDE TO DECX11 IS AVAILABLE FROM PROGRAM
            LIBRARY.


7.      MODULE OPERATION
        ****************

        A.  INITIALIZE THE PROGRAM TO ALLOW FOR PREVIOUS DEVICE AND/
            OR VECTOR ADDRESS MODIFICATIONS.

        B.  CLEAR THE CSR AND DBR AND TEST FOR DEVICE READY

        C.  BEGIN (PDT) TEST 1.  ENABLE AND CHECK THE INTERRUPT.

        D.  TRANSMIT A WORD (#52525) AND CHECK THAT THE INTERRUPT
            SERVICE ROUTINE WAS ENTERED.  (SWITCH ENTER1 WILL BE
            SET WHEN THE SERVICE ROUTINE IS ENTERED).

        E.  CHECK THE CSR FOR CORRECT STATUS THEN CHECK THAT THE
            CORRECT WORD WAS RETURNED.  END TEST1.

        F.  BEGIN (NPR) TEST2, CHECK THECSR FOR CORRECT STATUS.
```

```
153                         SET UP THE WORST CASE DATA PATTERN AND LOAD IT INTO
154                         TRANSMIT BUFFER.
155
156                     G.  SET UP THE NPR PARAMETERS AND START THE NPR TRANSFER.
157                         CHECK THAT THE INTERRUPT ROUTINE WAS ENTERED.
158                         CHECK THE CSR FOR CORRECT STATUS.
159                         CHECK THAT ALL 64 WORDS WERE TRANSMITTED AND RECEIVED.
160                         CHECK THAT TRANSMITTED BUFFER MATCHES RECEIVED
161                         BUFFER.
162
163                     H.  IF TEST IS SUCCESSFUL, PRINT OUT "ENDPASS" AFTER 130000
164                         ITERARIONS OF THE TEST, AND LOOP TO BEGINNING OF TEST.
165
166
167             8.      OPERATION OPTIONS
168                     **********
169
170                     NONE
171
172             9.      NON-STANDARD PRINTOUTS
173                     **********
174
175                     MOST PRINTOUTS HAVE STANDARD FORMATS AS DESCRIBED IN THE
176                     DEC/X11 DOCUMENT.
177
178                     THE ONLY EXCEPTION IS THE KIT11D PROGRAM ERROR PRINTOUTS.
179                     ON THESE PRINTOUTS:
180
181                     THE CSRA WILL INDICATE THE CONTENTS OF THE WORD COUNT REGISTER.
182
183                     THE ACSR WILL INDICATE THE CONTENTS OF THE CSR AND THE
184                     ASTAT WILL BE THE KIT11D PROGRAM ERROR NUMBER.
185
186             10.     LOADING THE BINARY (IE.,CONFIGURED EXERCISER) PROGRAM
187
188                     THE PROGRAM IS LOADED VIA THE ABSOLUTE LOADER.
189
190                     START ADDRESS=200
191
192                     RESTART ADDRESS=1000
193
194
```

```
                        ;*****************************************************
                        ;******************** KIT 11D ************************
                        ;******************** ERROR MESSAGES *****************
                        ;*****************************************************
                        ;
                        ;
                        ;THE FORMAT OF KIT11D ERROR PRINTOUTS IS:
                        ;
                        ;NAME   PC      ASSEMBLED PC     PASS #          SYSTEM ERROR COUNT
                        ;WC REGISTER    CSR CONTENTS     PROGRAM ERROR #
                        ;
                        ;
                        ;THE POSSIBLE PROGRAM ERRORS ARE:
                        ;
                        ;********* ERROR ************************** EXPLANATION ********
                        ;
                        ;       1               CSR NOT 610 OR 210 (PDT) DEVICE NOT READY
                        ;       2               CSR NOT 710 OR 310 (PDT) INT NOT ENABLED
                        ;       3               CSR NOT 126510 OR 126110 (PDT)
                        ;       4               DATA MISMATCH, DBR NOT =52525 (PDT)
                        ;
                        ;       5               CSR NOT 710 OR 210 (NPR) CSR NTO READY
                        ;       6               CSR NOT 310 (NPR) INT OK BUT CSR NOT READY.
                        ;       7               WORD COUNT NOT ZERO (NPR)
                        ;       10              ALL WORDS NOT XMITTED (NPR)
                        ;       11              DATA MISMATCH (NPR)
                        ;
                        ;       12              TIMEOUT ERROR (NO INT OCCURRED) (NPR/PDT)
                        ;       13              SLAVE SYNCH ERROR (NPR/PDT)
                        ;
                        ;
                        ;
                        ;KIT/11D EXERCISER MODULE
                        IOMOD   <BBAB >,1,1,5,,70000,62
                        MODULE  140000,BBAB ,1,1,5,,,70000,62
                                .TITLE  BBAB DEC/X11 SYSTEM EXERCISER MODULE
                        ;       DDXCOM  VERSION  6              23-MAY-78
                                .LIST   BIN
                        ;*****************************************************************************
000000'                 BEGIN:
000000' 041102  041101      040 MODNAM: .ASCII  /BBAB / ;MODULE NAME.
000005'     000         XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 000001          ADDR:   1+0                     ;1ST DEVICE ADDR.
000010' 000001          VECTOR: 1+0                     ;1ST DEVICE VECTOR.
000012'     240         BR1:    .BYTE   PRTY5+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' 000254'         INIT:   START                   ;MODULE START ADDR.
```



```
000032' 000224'         SPOINT: MODSP                   ;MODULE STACK POINTER.
000034' 000000          PASCNT: 0                       ;PASS COUNTER.
000036' 070000          ICONT:  70000                   ;# OF ITERATIONS PER PASS=70000
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'                 SRADR:                          ;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          AS8:    OPEN                    ;EXPECTED DATA.
000110' 000000          AWAS:   OPEN                    ;ACTUAL DATA.
000112' 000344'         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' 000062          IDNUM:  62                      ;MODULE IDENTIFICATION NUMBER=62
        000040                  .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
                                .NLIST
                                .WORD   0
                                .LIST
                                .ENDR
000224'                 MODSP:
                        ;*****************************************************************************
```

```
                                ;******************** INITIALIZATION ********************
                                ;
                                ;LOCATIONS LOCAL TO THIS MODULE.
                                ;
000224' 160006                  DBR:    160006          ;DATA BUFFER REGISTER.
000226' 160004                  CSR:    160004          ;CONTROL STATUS REGISTER.
000230' 160002                  BAR:    160002          ;BUS ADDRESS REGISTER.
000232' 160000                  WC:     160000          ;WORD COUNT REGISTER.
000234' 000001                  NPRMAX: .WORD   1       ;NPR WORST CASE DATA LOOP LIMIT.
000236' 177700                  NPRLNT: .WORD   -100    ;(NPR), #OF WORDS IN WORST CASE PATTERN
                                                        ;64=32X2 TRANSFERS FOR WORD COUNT.
000240' 000000                  NPRCTR: .WORD   0       ;UP COUNTER FOR NPR PATTERN.
000242' 000000                  ERRON:  .WORD   0       ;KIT11D ERROR NUMBER.
                                                        ;THIS NUMBER IS PRINTED DURING KIT11
                                                        ;PROGRAM GENERATED ERROR MESSAGES.
000244' 000200                  TLIMIT: .WORD   200     ;TIME-OUT LIMIT.
000246' 000000                  TTEMP:  .WORD   0       ;WORKING LOCATION.
000250' 000000                  ENTERI: .WORD   0       ;INTERRUPT ENTERED FLAG.
000252' 000000                  SLVSYN: .WORD   0       ;SLAVE SYNCH ERROR FLAG.
                                                        ;FLAG IS SET UPON ENTERING INTERRUPT SERVICE ROUTINE.
                                                        ;(NORMALLY CLEARED).
                                                        ;WILL EQUAL ONE ENDPASS.
000254' 012767  000002  177636  START:  MOV     #2,INTR         ;2 INTERRUPTS/ITERATION
000262' 012767  000100  177624          MOV     #100,WDTQ       ;100 WORDS TO MEM/ITERATION
000270' 012767  000100  177620          MOV     #100,WDFR       ;100 WORDS FROM MEM/ITERATION
000276' 016701  177506                  MOV     VECTOR,R1       ;SET UP INTERRUPT
000302' 012721  001432'                 MOV     #KITSRV,(R1)+   ;INFORMATION.
000306' 116721  177500                  MOVB    BR1,(R1)+
                                ;
                                ;
                                ;SET UP THE REGISTER ADDRESSES FOR THE KIT-11 D.
                                ;
                                ;
000312' 016702  177470                  MOV     ADDR,R2         ;GET BASE ADDRESS.
000316' 010267  177710                  MOV     R2,WC
000322' 005722                          TST     (R2)+
000324' 010267  177700                  MOV     R2,BAR          ;BUSS ADDRESS.
000330' 005722                          TST     (R2)+
000332' 010267  177670                  MOV     R2,CSR          ;CONTROL STATUS
000336' 005722                          TST     (R2)+
000340' 010267  177660                  MOV     R2,DBR          ;DATA BUFFER REG.
000344' 005077  177654          RESTRT: CLR     @DBR            ;START OFF BY CLEARING
000350' 005077  177652                  CLR     @CSR            ;THE CONTROL STATUS REGISTER.
000354' 005067  177670                  CLR     ENTERI          ;ZERO THE "INTERRUPT ENTERED" FLAG.
000360' 005067  177666                  CLR     SLVSYN          ;ZERO THE "SLAVE SYNCH" ERROR FLAG.
000364' 012777  177777  177640          MOV     #-1,@WC         ;SET UP FOR PDT.
000372' 012777  001466' 177630          MOV     #BUFFER,@BAR    ;SAME AS ABOVE.
000400' 022777  000210  177620          CMP     #210,@CSR       ;CHECK TO SEE IF BIT7
                                                        ;CAME UP (READY BIT).
```



```
000406' 001420                  BEQ     PDT1
000410' 022777  000610  177610          CMP     #610,@CSR       ;OPTIONAL CASE.
000416' 001414                  BFQ     PDT1            ;YES, CSR=610 (FIRST TIME THRU THIS IS TRUE)
000420' 004767  000756          ERR:    JSR     PC,SETUP        ;LOAD CSRA, ASSR,ASTAT
                                                        ;PRIOR TO ERROR CALL.
000424' 012767  000001  000630          MOV     #1,ERRORN       ;ERROR NO. 1
000432' 012767  000003  177446          MOV     #3,ERRTYP       ;CONTROLLER NOT READY
                                ;*************************************************************
000440' 104405  000000' 000000          HRDRRS,BEGIN,NULL       ;CSR NOT READY ---- ERROR #1----
                                ;*************************************************************
000446' 000702                  BR      START           ;GO TRY AGAIN.
                                .SBTTL  TEST 1 (PDT)
```

```
                                ;********************** TEST 1 (PDT) ******************
                                ;------------------------ PDT ---------------------------
                                ;(PDT MEANS "PROGRAMMED DATA TRANSFER")
                                ;THIS PART ATTEMPTS A SINGLE DATA TRANSFER UNDER
                                ;PROGRAM CONTROL.  A CHARACTER IS WRITTEN INTO
                                ;OUTPUT BUFFER AND KEPT IN A TIMING LOOP UNTIL
                                ;THE CHARACTER EVOKES AN "INPUT" INTERRUPT.  IF NO
                                ;INTERRUPT IS GENERATED, AN ERROR IS PRINTED AND A
                                ;RESTART IS INITIATED.
                                ;
                                ;
                                ;THIS PART OF THE EXERCISER USES A MOVE INSTRUCTION TO
                                ;LOAD THE DBR.

                                ;AN INTERRUPT IS NOT EXPECTED UNTIL THE DATA BUFFER
                                ;REGISTER HAS BEEN LOADED.

000450' 052777  000100  177550  PDT1:   BIS     #100,@CSR       ;ENABLE THE INTERRUPT.
000456' 022777  000310  177542          CMP     #310,@CSR       ;CHECK THE BIT.

000464' 001420                          BEQ     2$              ;BIT 6 IS UP.
000466' 022777  000710  177532          CMP     #710,@CSR       ;SECOND CASE TO CHECK.
000474' 001414                          BEQ     2$
000476' 004767  000700                  JSR     PC,SETUP
000502' 012767  000002  000552          MOV     #2,ERRORN       ;ERROR #2.
000510' 012767  000027  177370          MOV     #27,ERRTYP      ;INTERRUPT ENABLE NOT WORKING
                                        ;*****************************************************
000516' 104405  000000' 000000          HRDERR,BEGIN,NULL       ;INTERRUPT ENABLE DIDN'T COME UP ---- ERROR #2 ----
                                        ;*****************************************************
000524' 000653                          BR      START           ;RESTART.
```



```
                                ;THE INTERRUPT HAS BEEN ENABLED, XMIT A WORD.  START
                                ;THE TIMER, AND SEE IF THE WORD IS WRAPPED AROUND.
                                ;
000526' 012777  052525  177470  2$:     MOV     #52525,@DBR     ;XMIT A WORD.
000534' 004767  000524          JSR     PC,TIMOUT       ;GO TO THE TIME-OUT
                                                        ;DELAY ROUTINE.
                                ;RETURNS HERE IF THE INTERRUPT OCCURRED.

000540' 027727  177462  126010  1$:     CMP @CSR,#126010                ;CASE FOR LAST DIRECTION=IN.
000546' 001420                          BEQ     3$              ;MATCH
000550' 022777  126410  177450          CMP     #126410,@CSR    ;CASE FOR LAST DIR=OUT.
000556' 001414                          BEQ     3$
000560' 012767  000003  000474          MOV     #3,ERRORN       ;ERROR 3
000566' 004767  000610                  JSR     PC,SETUP
000572' 012767  000025  177306          MOV     #25,ERRTYP      ;BIT STUCK IN REG.
                                        ;*****************************************************
000600' 104405  000000' 000000          HRDERR,BEGIN,NULL       ;CSR NOT 126010 OR 126410 ---- ERROR #3----
                                        ;*****************************************************
000606' 000622                          BR      START           ;RESTART
000610' 022777  052525  177406  3$:     CMP     #52525,@DBR     ;COMPARE XMITTED
                                                                ;DATA TO RCVD DATA.
                                                                ;

000616' 001414                          BEQ     PDT2            ;YES, DATA MATCHED.
000620' 012767  000004  000434          MOV     #4,ERRORN       ;ERROR #4
000625' 004767  000550                  JSR     PC,SETUP
                                ;
000632' 012767  000025  177246          MOV     #25,ERRTYP      ;BIT STUCK IN REG
                                        ;*****************************************************
000640' 104405  000000' 000000          HRDERR,BEGIN,NULL       ;DATA MISMATCH DURING PDT ---- ERROR #5----
                                        ;*****************************************************
000646' 000602          RTN1:   BR      START           ;RESTART.
                                ;
                                .SBTTL  TEST 2 NPR
```

```
                                        ;********************** TEST 2 (NPR) ******************
                                        ;AT THIS POINT:
                                        ;A PROGRAMMED DATA XFER WAS SUCCESSFUL.  DATA
                                        ;XMITTED WAS EQUAL TO DATA RCVD.
                                        ;INTERRUPTS ARE WORKING CORRECTLY.
                                        ;
                                        ;NOW DO A NPR TRANSFER.
                                        ;
000650' 042777  002000  177350  PDT2:   BIC     #2000,@CSR      ;CLEAR BIT 10. (INPUT BIT)
000656' 027727  177344  000210          CMP     @CSR,#210       ;READY BIT UP?
000664' 001421                          BEQ     NPR1
000666' 027727  177334  000610          CMP     @CSR,#610       ;CHECK FOR 2ND CASE.
000674' 001415                          BEQ     NPR1            ;YES, BIT READY.
000676' 012767  000005  000356          MOV     #5,ERRORN       ;ERROR #5.
000704' 004767  000472                  JSR     PC,SETUP        ;GET READY FORERROR MESSAGE.
000710' 012767  000003  177170          MOV     #3,ERRTYP       ;CONTROLLER NOT READY
                                        ;*************************************************************
000716' 104405  000000' 000000          ERRORS,BEGIN,NULL       ;CSR NOT READY AFTER PDT ----- ERROR #3-----
                                        ;*************************************************************
000724' 000167  177324                  JMP     START           ;AND RETURN.
```



```
                                        ;
                                        ;SET UP THE WORST CASE DATA PATTERN AND
                                        ;XMIT IT UNDER NPR CONTROL.
                                        ;
                                        ;
                                        ;
                                        ;
                                        ;SET UP THE WORST CASE DATA PATTERN BELOW:
                                        ;(THIS PATTERN WILL BE LOADED INTO ODD LOCATIONS)
                                        ;
                                        ;       WORD:           PATTERN:
                                        ;       1               177776
                                        ;       3                    1
                                        ;
                                        ;       5               177775
                                        ;       7                    2
                                        ;
                                        ;       11              177763
                                        ;       13                   4
                                        ;
                                        ;       15              177767
                                        ;       17                  10
                                        ;
                                        ;       21              177757
                                        ;       23                  20
                                        ;       :                    :
                                        ;       :                    :
                                        ;
                                        ;       75              077777
                                        ;       77              100000
                                        ;
000730' 012701  001466'  NPR1:  MOV     #BUFFER,R1      ;BUFFER ADDRESS FOR
                                                        ;XMITTED/RCVD DATA.
000734' 005067  177300          CLR     NPRCTR          ;START WITH FRESH COUNTER
000740' 012700  000001  NPR1A:  MOV     #1,R0           ;START FORMING DATA ITS
                                                        ;COMPLEMENT.
                                        ;
000744' 005100          NPR2:   COM     R0              ;INVERT IT(NEG #).
000746' 010021                  MOV     R0,(R1)+        ;AND PUT IT INTO ADDRESS AT R1
000750' 005021                  CLR     (R1)+           ;OPEN A SPOT IN BUFFER FOR
                                                        ;RETURNED DATA (FREE LOCATION).
000752' 005100                  COM     R0              ;INVERT IT AGAIN (POS #) AND
000754' 010021                  MOV     R0,(R1)+        ;OPEN A FREE SPOT FOR
                                                        ;RETURNED DATA (TO BE USED LATER).
000756' 005021                  CLR     (R1)+           ;(FREE LOCATION)
                                        ;
                                        ;WE HAVE FORMED A NEGATIVE NUMBER AND ITS
                                        ;POSITIVE COMPLEMENT IN THE BUFFER.  WE HAVE ALSO
                                        ;STAGGERED THE BUFFER TO ALLOW EACH NUMBER FORMED
                                        ;TO BE FOLLOWED BY A FREE LOCATION, DUE TO THE
                                        ;NATURE OF THE "WRAP-AROUND" CABLE USED IN THIS
                                        ;TEST.
                                        ;
                                        ;THUS OUR BUFFER WILL CONTAIN DATA TRANSMITTED INTERLEAVED
```

```
                                        ;WITH THE DATA THAT WAS RECEIVED.
                                        ;
000760' 006300                  ASL     R0              ;START SHIFTING R0 TO FORM
                                                        ;THE NEXT SET OF DATA TO BE
                                                        ;LOADED.
000762' 001370                  BNE     NPR2
000764' 005267  177250          INC     NPRCTR          ;NP COUNTER TO
                                                        ;LOOP X TIMES TO
                                                        ;OBTAIN 32X
                                                        ;WORDS TO XFER.
                                                        ;
000770' 026767  177240  177242  CMP     NPRMAX,NPRCTR   ;X TIMES THRU LOOP?
                                                        ;(HERE, X=1.)
000776' 001360                  BNE     NPR1A           ;NO, SO DO IT AGAIN.
                                ;
                                ;YES, THE XMIT BUFFER HAS BEEN SET UP.
                                ;
```



```
                                        ;NOW SETUP THE TRANSFER (NPR) PARAMETERS.
                                        ;
                                        ;
001000' 016777  177232  177224          MOV     NPRLMT,@WC      ;2X WORDS
001006' 012777  001466' 177214          MOV     #BUFFER,@BAR    ;SET UP THE ADDRESS
                                                                ;OF THE BUFFER.
                                        ;START THE ACTUAL TRANSFER HERE.
001014' 052777  000103  177204          BIS     #103,@CSR       ;SET IN AND OUT.
                                                                ;ENABLE THE INTERRUPT.
                                                                ;
001022' 004767  000236                  JSR     PC,TIMOUT       ;CHECK THE INTERRUPT SERVICE
                                                                ;ROUTINE VIA TIME-
                                                                ;OUT LOOP.
                                                                ;
                                        ;
                                        ;AT THIS POINT, WE HAVE VERIFIED THAT AN INTERRUPT HAS
                                        ;OCCURRED.  CHECK THE CSR FOR ERRORS.
                                        ;
001026' 027727  177174  000210  1$:     CMP     @CSR,#210       ;CSR=310?
001034' 001415                  BEQ     NPR4A           ;YES, GOTO WORD COUNT OVERFLOW CHECK
001036' 012767  000006  000216          MOV     #6,ERRORN       ;ERROR #6
001044' 004767  000332                  JSR     PC,SETUP        ;NO, GET READY TO REPORT ERROR.
001050' 012767  000025  177030          MOV     #25,ERRTYP      ;BIT STUCK IN REG.
                                        ;*****************************************************
001056' 104405  000000' 000000          ERDERS,BEGIN,NULL       ;CSR NOT 310 ----- ERROR #6----
                                        ;*****************************************************
001064' 000167  177164                  JMP     START           ;RESTART
                                        ;
                                        ;NOW CHECK THE BAR TO MAKE SURE THAT THE
                                        ;INTERRUPT WAS NOT DUE TO A PREMATURE WC OVERFLOW.
                                        ;
001070' 005777  177136  NPR4A:  TST     @WC
001074' 001413                  BEQ     NPR4
001076' 012767  000007  000156          MOV     #7,ERRORN       ;ERROR 7.
001104' 012767  000011  176774          MOV     #11,ERRTYP      ;ILLEGAL INTERRUPT
                                        ;*****************************************************
001112' 104405  000000' 000000          ERDERS,BEGIN,NULL       ;WORD COUNT NOT ZERO ----ERROR #7----
                                        ;*****************************************************
001120' 000167  177130                  JMP     START           ;RESTART.
                                        ;
                                        ;NOW CHECK TO MAKE SURE THAT 100 WORDS WERE TRANSFERRED.
                                        ;THE BAR SHOULD BE EQUAL TO BUFFER START ADDRESS PLUS 200.
                                        ;
001124' 017767  177100  177114  NPR4:   MOV     @BAR,TTEMP      ;SAVE BAR.
001132' 162767  001466' 177106          SUB     #BUFFER,TTEMP   ;GET THE OFFSET.
001140' 026727  177102  000200          CMP     TTEMP,#200      ;ALL WORDS XFERRED?
001146' 001415                  BEQ     NPR5            ;YES.
001150' 012767  000010  000104          MOV     #10,ERRORN      ;ERROR #10
001156' 004767  000220                  JSR     PC,SETUP        ;NO
001162' 012767  000011  176716          MOV     #11,ERRTYP      ;ILLEGAL INTERRUPT
                                        ;*****************************************************
001170' 104405  000000' 000000          ERDERS,BEGIN,NULL       ;ALL WORDS NOT TRANSFERRED ----ERROR #10----
```

```
                        ;*************************************************************
001176' 000167  177052          JMP     START           ;RESTART.
                        ;
                        ;ALL X WORDS HAVE BEEN XFERRED.
                        ;CHECK THE XMITTED/RCVD DATA.
                        ;CONTENTS OF BUFFER LOCATION X SHOULD EQUAL
                        ;CONTENTS OF X+2.
                        ;
001202' 012701  001466'  NPR5:   MOV     #BUFFER,R1      ;GET THE STARTING
                                                         ;ADDRESS OF
001206' 021121           NPR5A:  CMP     (R1),(R1)+      ;BUFFER.
001210' 001415                   BEQ     NPR5B           ;MATCH.
001212' 012767  000011  000042           MOV     #11,ERRORN      ;ERROR #11.
001220' 004767  000156           JSR     PC,SETUP        ;NO MATCH.
001224' 012767  000001  176654           MOV     #1,ERRTYP       ;DATA ERROR
                                         ;*************************************************************
001232' 104405  000000' 000000           ERDER$,BEGIN,NULL       ;DATA MISMATCH ;NPR CHECK ----ERROR #11----
                                         ;*************************************************************
001240' 000167  177010           JMP     START           ;RESTART.
                        ;LOOP UNTIL ENTIRE BUFFER HAS BEEN CHECKED.
001244' 020127  001666'  NPR5B:  CMP     R1,#BUFFER+200  ;HAVE WE CHECKED THE ENTIRE BUFFER?
001250' 100756           BMI     NPR5A                   ;GET PAIRS AND
                                                         ;CHECK THEM UNTIL
                                                         ;ALL HAVE BEEN CHECKED.
                        ;TEST RAN SUCCESSFULLY.  TYPE OUT A MESSAGE
                        ;OF "ENDPASS" AND THEN RE-RUN THE TEST.
                        ;
                        ;LET 70000 PASSES THROUGH THE KIT EXERCISER
                        ;BE EXECUTED BEFORE PRINTING ENDPASS MESSAGE.
                        ;ELSE PRINT OUT THE MESSAGE AND RESTART THE EXERCISER.
                        ;A PASS HAS BEEN COMPLETED.
                        ;*************************************************************
                        ;********************* END OF TEST ***************************
                        ;*************************************************************
001252'                  NPR6:
001252' 104413  000000'          ENDIT$,BEGIN            ;SIGNAL END OF ITERATION.
                                                         ;MONITOR SHALL TEST END OF PASS
001256' 000167  177062   RESTRA: JMP     RESTRT          ;COMES HERE IF NOT YET 70000 ITERATIONS OF THE EXERCISER
001262' 000000           ERRORN: .WORD   0               ;ERROR NUMBER.
                        .SBTTL  SUBROUTINES
```



```
                        ;*************************************************************
                        ;********************* TIME OUT ROUTINE **********************
                        ;*************************************************************
                        ;THIS IS A CLOSE (TIGHT) LOOP TO SERVICE OTHERE DEVICES.
                        ;THE ONLY WAY THIS LOOP CAN BE BROKEN OUT OF IS VIA
                        ;AN INTERRUPT FROM THE KIT11D MODULE.
                        ;AN ERROR IS PRODUCED IF THE INTERRUPT DOES NOT OCCUR
                        ;WITHIN LIMITS.
001264' 016767  176754  176754   TIMOUT: MOV     TLIMIT,TTEMP    ;SET THE TIMEOUT WAIT LIMIT.
                        ;WHEN AN INTERRUPT IS SERVICED, "ENTERI"
                        ;WILL BE SET.  THIS SOFTWARE SWITCH WILL EXIT US OUT OF
                        ;THE BREAK LOOP
001272'                  TIM1:
001272' 104407  000000'          BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
001276' 104407  000000'          BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
001302' 005767  176742           TST     ENTERI          ;CHECK TO SEE IF AN INTERRUPT
                                                         ;HAS OCCURRED.
001306' 001021                   BNE     TIMOK           ;AN INTERRUPT DID OCCUR--BRANCH
001310' 005367  176732           DEC     TTEMP           ;DOWN THE WAIT COUNTER.
001314' 001366                   BNE     TIM1            ;NO, CONTINUE ON.
001316' 012767  000012  177736           MOV     #12,ERRORN      ;ERROR #12.
001324' 004767  000052           JSR     PC,SETUP        ;TAKE AN RTS TO CLEAN UP THE STACK
001330' 005726                   TST     (SP)+           ;POP THE STACK
001332' 012767  000023  176546           MOV     #23,ERRTYP      ;MISSING INTERRUPT
                                         ;*************************************************************
001340' 104405  000000' 000000           ERDER$,BEGIN,NULL       ;TIMEOUT ERROR ---- ERROR #12----
                                         ;*************************************************************
001346' 000167  176702           JMP     START           ;RESTART
001352' 005067  176672   TIMOK:  CLR     ENTERI          ;RESET THE FLAG TO ZERO.
001356' 005767  176670           TST     SLVSYN          ;TEST FOR A POSSIBLE SLAVE SYNCH
                                                         ;ERROR.
                                                         ;IF SLAVE SYNCH DOESN'T COME BACK
                                                         ;WITHIN 20 US, BIT 14 WILL GENERATE AN INTERRUPT.
001362' 001406                   BEQ     1$              ;BRANCH IF NOT SLAVE SYNCH.
001364' 012767  000013  177670           MOV     #13,ERRORN
001372' 005726                   TST     (SP)+           ;TAKE AN RTS TO CLEANUP STACK
001374' 000167  176654           JMP     START           ;RESTART.
001400' 000207           1$:     RTS     PC              ;RETURN TO MAIN PROGRAM.
                                                         ;(NORMAL RETURN.)
```

```
                                        ;*************************************************************
                                        ;************************ SETUP ROUTINE **********************
                                        ;
                                        ;ROUTINE TO LOAD THE CSRA,ACSR,ASTAT
                                        ;FOR DEC/X11 ERROR ROUTINES.
                                        ;PRINTOUT INCLUDES:  WC,CONTENTS OF CSR, ERROR #
                                        ;

001402' 017767  176624  176470  SETUP:  MOV     @WC,CSRA        ;SAVETHE CSR.
001410' 017767  176612  176464          MOV     @CSR,ACSR       ;SAVE CONTENTS OF CSR.
001416' 016767  177640  176460          MOV     ERRORN,ASTAT    ;GET ERROR#.
001424' 005077  176576                  CLR     @CSR            ;CLEAR THE (INTERRUPT ENABLE)CSR
001430' 000207                          RTS     PC              ;AND RETURN.

                                        .SBTTL  INTERRUPT SERVICE ROUTINE
```



```
                                        ;
                                        ;*************************************************************
                                        ;INTERRUPT SERVICE ROUTINE.
                                        ;*************************************************************
                                        ;
                                        ;SET THE INTERRUPT ENTERED FLAG (ENTERI)AND
                                        ;RETURN.
                                        ;
001432' 042777  000100  176566  KITSRV: BIC     #100,@CSR       ;DISABLE THE INTERRUPT.
001440' 012767  000001  176602          MOV     #1,ENTERI       ;SET THE "INTERRUPT ENTERED" FLAG.
001446' 032777  040000  176552          BIT     #40000,@CSR     ;20 US SLAVE SYNCH FAILURE??.
001454' 001403                          BEQ     1$              ;BRANCH IF NOT.
001456' 012767  000001  176566          MOV     #1,SLVSYN       ;ELSE SET THE SLAVE ERROR FLAG.

001464' 000002          1$:     RTI                             ;RETURN TO THE TIMOUT ROUTINE.
                                        ;
                                        ;KIT INTERRUPT VECTOR ADDRESS
                                        ;
                                        ;
                                        .SBTTL  DATA BUFFER
001466' 000100          BUFFER: .BLKW   100
001666' 000000                  .WORD   0
        000001          .END
```

```
ACSR     000102R           273#    692*
ADDR     000006R           239#    331
ADDR22=  001000            291#
ASB      000106R           277#
ASTAT    000104R           275#    693*
AWAS     000110R           274#
BAR      000230R           299#    334*    346*    528*    570
BEGIN    000000R           236#    358     391     411     424     446     551     563     578     599     625     649     650
                           665
BIT0   = 000001            291#
BIT1   = 000002            291#
BIT10  = 002000            291#
BIT11  = 004000            291#
BIT12  = 010000            291#
BIT13  = 020000            291#
BIT14  = 040000            291#
BIT15  = 100000            291#
BIT2   = 000004            291#
BIT3   = 000010            291#
BIT4   = 000020            291#
BIT5   = 000040            291#
BIT6   = 000100            291#
BIT7   = 000200            291#
BIT8   = 000400            291#
BIT9   = 001000            291#
BREAK$= 104407             291#    649     650
BR1      000012R           241#    323
BR2      000013R           242#
BTOD$  = 104421            291#
BUFFER   001466R           346     481     528     571     589     605     718#
CDATA$= 104412             291#
CONFIG   000056R           261#
CSR      000226R           298#    336*    341*    347     351     381*    382     385     403     405     437*    438     440
                           532*    544     692     694*    706*    708
CSRA     000100R           271#    691*
DATCK$= 104411             291#
DATER$= 104404             291#
DBR      000224R           297#    338*    340*    398*    414
DVID1    000014R           243#
ENDITS= 104413             291#    625
END$   = 104410            291#
ENTERI   000250R           312#    342*    652     669*    707*
ERR      000420R           353#
ERROR    000242R           306#
ERRORN   001262R           355*    388*    407*    419*    442*    546*    560*    574*    593*    629#    659*    676*    693
ERRTYP   000106R           276#    356*    389*    409*    422*    444*    549*    561*    576*    597*    663*
EXIT$  = 104400            291#
GETPA$= 104415             291#
GWBUF$= 104414             291#
HRDCNT   000044R           256#
HRDERS= 104405             291#    358     391     411     424     446     551     563     578     599     665
HRDPAS   000050R           258#
ICONT    000036R           253#
ICOUNT   000040R           254#
IDNUM    000122R           283#
INIT     000030R           250#
```



```
INTR     000120R           282#    318*
KITSRV   001432R           322     706#
MAP22$= 104416             291#
MODNAM   000000R           237#
MODSP    000224R           251     289#
MSGN$  = 104403            291#
MSGS$  = 104402            291#
MSG$   = 104401            291#
NPRCTR   000240R           305#    483*    511*    517
NPRLMT   000236R           302#    527
NPRMAX   000234R           301#    517
NPR1     000730R           439     441     491#
NPR1A    000740R           484#    519
NPR2     000744R           487#    510
NPR4     001124R           559     570#
NPR4A    001070R           545     558#
NPR5     001202R           573     589#
NPR5A    001206R           591#    606
NPR5B    001244R           592     605#
NPR6     001252R           624#
NULL   = 000000            291#    358     391     411     424     446     551     563     578     599     665
OPEN   = 000000            238     244     245     246     247     264     265     266     267     268     269     270     271
                           273     275     277     278     280     281     282     291#
OTOA$  = 104420            291#
PASCNT   000034R           252#
PDT1     000450R           350     352     381#
PDT2     000650R           418     437#
PIRQ$  = 000004            291#
POPSP  = 005726            291#
POPSP2= 022626             291#
PRTY   = 000000            242     291#
PRTY0  = 000000            291#
PRTY1  = 000040            291#
PRTY2  = 000100            291#
PRTY3  = 000140            291#
PRTY4  = 000200            291#
PRTY5  = 000240            241     291#
PRTY6  = 000300            291#
PRTY7  = 000340            291#
PS     = 177776            291#
PSW    = 177776            291#
PUSH   = 005746            291#
PUSH2  = 024646            291#
RAND$  = 104417            291#
RANNUM   000054R           260#
RESTRA   001256R           628#
RESTRT   000344R           279     340#    628
RES1     000056R           262#
RES2     000060R           263#
RSTRT    000112R           279#
RTN1     000646R           426#
SBADR    000102R           272#
SETUP    001402R           353     387     408     420     443     547     575     595     661     691#
SLVSYN   000252R           313#    343*    670     710*
SOFCNT   000042R           255#
SOFERS= 104406             291#
```

```
SOFPAS  000046R          257#
SPOINT  000032R          251#
SPSIZ =  000040            1#    284
SR1     000016R          244#
SR2     000020R          245#
SR3     000022R          246#
SR4     000024R          247#
START   000254R          250     318#   361    393    413    426    448    553    565    580    601    667    678
STAT    000026R          249#
SVR0    000062R          264#
SVR1    000064R          265#
SVR2    000066R          266#
SVR3    000070R          267#
SVR4    000072R          268#
SVR5    000074R          269#
SVR6    000076R          270#
SYSCNT  000052R          259#
TIMOK   001352R          655     669#
TIMOUT  001264R          399     535    642#
TIMI    001272R          648#    658
TLIMIT  000244R          310#    642
TRPDFD= 000022           291#
TTEMP   000246R          311#    570*   571*   572    642*   657*
VECTOR  000010R          240#    321
WASADR  000104R          274#
WC      000232R          300#    332*   345*   527*   558    691
WDFR    000116R          281#    320*
WDTO    000114R          280#    319*
XFLAG   000005R          239#
.     = 001670R          718#

. ABS.  000000     000
        001670     001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XBBABO,XBBABO/SOL/CRF:SYM=DDXCOM,XBBABO
 RUN-TIME:  1  2  .3 SECONDS
 RUN-TIME RATIO: 19/4=4.4
 CORE USED:  7K  (13 PAGES)
```