.REM _

IDENTIFICATION

PRODUCT CODE: AC-E962B-MC

PRODUCT NAME: CXPCSB0 PCS-11 MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

1. ABSTRACT

PCS IS AN IOMOD THAT EXERCISES PCS CONTROLLER AND FILE BOX. IT WILL EXERCISE ALL BITS OF CSR ADDRESS, CHECK FOR MAINTENANCE INTERRUPT CAPABILITIES AND CHECK ALL THE ADDRESSES FROM 171000 TO 171375 IN MAINTENANCE MODE.

2. REQUIREMENTS

HARDWARE: IOCM CONTROL MODULE WITH FILE BOX
STORAGE:: PCS REQUIRES:
1. DECIMAL WORDS: 748
2. OCTAL WORDS: 1354
3. OCTAL BYTES: 2730

3. TEST DESCRIPTION

ONE PASS OF THE MODULE CONSISTS OF CSR CHECKS, INTERRUPT TEST AND IAR TEST IN MAINTENANCE MODE. DURING THE TEST DBIT IS SET SO THE TEST WILL NOT EFFECT I/O MODULES

4. EXECUTION TIME

ONE PASS OF THE TEST TAKES LESS THEN 1 SEC

5. CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS

DEUADR: 171376
VECTOR: 234
DEVCNT: 1

6. MODULE OPERATION

TEST SEQUENCE

A. SET UP THE DEVICE ADDRESSES
B. SET G BIT, READ IT BACK AND CLEAR IT
C. SET D BIT, READ IT BACK AND CLEAR IT
D. SET T BIT, READ IT BACK AND CLEAR IT
E. SET RIF BIT, READ IT BACK AND CLEAR IT
F. SET E BIT, READ IT BACK AND CLEAR IT
G. SET M BIT, READ IT BACK AND CLEAR IT
H. SET MBIT AND READ ALL ADDRESSES FROM
I. SET MAINTENCE INTERRUPT AND READ ALL ONES FROM IAR 171000 TO 171375
J. RESET CSR.

7. OPERATOR OPTIONS:

NONE

8. PRINTOUTS:

ALL PRINTOUTS ARE STANDARD
-

```
                                        IOMOD   <PCSB  >,171376,234,0,0,0,10,147
                                        MODULE  140000,PCSB  ,171376,234,0,0,0,10,147
                                        .TITLE  PCSB DEC/X11 SYSTEM EXERCISER MODULE
                                ;       DDXCOM  VERSION  6      23-MAY-78
                                        .LIST   BIN
                                ;*****************************************************************************
000000'                         BEGIN:
000000' 041520  041123      040 MODNAM: .ASCII  /PCSB / ;MODULE NAME.
000005'     000                 XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 171376                  ADDR:   171376+0                ;1ST DEVICE ADDR.
000010' 000234                  VECTOR: 234+0                   ;1ST DEVICE VECTOR.
000012'     000                 BR1:    .BYTE   PRTY0+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' 140000                  STAT:   140000                  ;STATUS WORD.
000030' 000246'                 INIT:   START                   ;MODULE START ADDR.
000032' 000224'                 SPOINT: MODSP                   ;MODULE STACK POINTER.
000034' 000000                  PASCNT: 0                       ;PASS COUNTER.
000036' 000010                  ICONT:  10                      ;# OF ITERATIONS PER PASS=10
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' 000274'                 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' 000147                  IDNUM:  147                     ;MODULE IDENTIFICATION NUMBER=147
                                        .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
000040
```



```
                                        .NLIST
                                        .WORD   0
                                        .LIST
                                        .ENDR
000224'                         MODSP:
                                ;*****************************************************************************
```

```
                                ;TEMPORARY STORAGE
                                ;TEMPORARY STORAGE
                                ;TEMPORARY STORAGE
                                ;ADDRESS OF CSR = 171377
                                ;ADDRESS OF IAR = 17136
                                ;INTERRUPT OCCURED FLAG
                                ;PASS COUNT
                                ;ADDRESS 171000
                                ;SECEND ADDR OF VECTOR

                                ; ONE INTERRUPT/ITERATION
                                ;8. WORDS TO MEM/ITERATION
                                ;8. WORDS FROM MEM/ITERATION
                                ;CLEAR PASS COUNT

                                ;SET ADDRESS OF IAR = 171376

                                ;SET ADDRESS OF CSR = 171377

                                ;SET ADDRESS OF THE LOWEST ID = 171000

                                ;SET UP WAIT LOOP
                                ;SET CBIT TO CLEAR CSR
                                ;WAIT FOR CLEAR

                                ;ANOTHER WAIT LOOP

                                ;IS CSR CLEAR?
                                ;NO, SET BAD DATA FOR ERROR CALL
                                ;SET GOOD DATA.
                                ;*****************************************************************
                                ;*****************************************************************

                                ;*****************************************************************
                                ;SET RIF BIT
                                ;IS IT SET
                                ;YES, GO TO 3$
                                ;LOAD CONTENTS OF CSR FOR ERROR
                                ;LOAD ADDRESS OF CSR FOR ERROR
                                ;ASCII MESSAGE CALL WITH COMMON HEADER

                                ;BIT STUCK
                                ;*****************************************************************
                                ; RIF BIT IS NOT SETTING
                                ;*****************************************************************

                                ;NOW RIF BIT SHOULD BE CLEAR
                                ;YES GO TO 4$
```



```
                                ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;BIT STUCK
                                ;*****************************************************************
                                ; RIF BIT IS NOT CLEARING
                                ;*****************************************************************

                                ;SET GBIT AT CSR
                                ;TEST IF SET
                                ;YES,GO TO 5$

                                ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;BIT STUCK
                                ;*****************************************************************
                                ; GENERIC BIT IS NOT SETTING
                                ;*****************************************************************

                                ;CLEAR GBIT AT CSR
                                ;TEST IF CLEAR
                                ;YES, GO TO 6$

                                ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;BIT STUCK
                                ;*****************************************************************
                                ;GENERIC BIT IS NOT CLEARING
                                ;*****************************************************************

                                ;SET DBIT AT CSR
                                ;TEST IF SET
                                ;YES,GO TO 7$

                                ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;BIT STUCK
                                ;*****************************************************************
                                ;DBIT IS NOT SETTING
                                ;*****************************************************************

                                ;SET TBIT @CSR
                                ;IS IT SET
                                ;YES, CONT.

                                ;ASCII MESSAGE CALL WITH COMMON HEADER
                                ;BIT STUCK
                                ;*****************************************************************
                                ;TBIT NOT SETTING
                                ;*****************************************************************

                                ;CLEAR TBIT
                                ;IS IT CLEAR
                                ;YES, CONT
```

```
001034' 016767  177172  177036'        MOV     CSR,CSRA
001042' 104403  000000' 002106'        MSGNS,BEGIN,TBITC       ;ASCII MESSAGE CALL WITH COMMON HEADER
001050' 012767  000025  177030         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001056' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;TBIT NOT CLEARING
                                       ;*****************************************************************

001064' 012703  000025         9$:     MOV     #7.*3.,R3               ;SET UP WAIT LOOP
001070' 142777  000020  177134         BICB    #20,@CSR                ;CLEAR DBIT
001076' 000240                 50$:    NOP                             ;WAIT FOR CSR
001100' 005303                         DEC     R3
001102' 001375                         BNE     50$
001104' 132777  000020  177120         BITB    #20,@CSR
001112' 001420                         BEQ     10$
001114' 001417                         BEQ     10$
001116' 117767  177110  176756         MOVB    @CSR,ACSR
001124' 016767  177102  176746         MOV     CSR,CSRA
001132' 104403  000000' 002112'        MSGNS,BEGIN,DBITC       ;ASCII MESSAGE CALL WITH COMMON HEADER
001140' 012767  000025  176740         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001146' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;DBIT NOT CLEARING
                                       ;*****************************************************************

001154' 152777  000040  177050  10$:    BISB    #40,@CSR                ;SET MBIT
001162' 132777  000040  177042         BITB    #40,@CSR                ;IS IT SET
001170' 001017                         BNE     11$
001172' 117767  177034  176702         MOVB    @CSR,ACSR               ;NO
001200' 016767  177026  176672         MOV     CSR,CSRA
001206' 104403  000000' 002114'        MSGNS,BEGIN,MBITS       ;ASCII MESSAGE CALL WITH COMMON HEADER
001214' 012767  000025  176664         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001222' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;MAINTENANCE BIT IS NOT SETTING
                                       ;*****************************************************************

001230' 142777  000040  176774  11$:    BICB    #40,@CSR                ;CLEAR MBIT
001236' 132777  000040  176766         BITB    #40,@CSR                ;IS IT CLEAR
001244' 001417                         BEQ     12$
001246' 117767  176760  176626         MOVB    @CSR,ACSR               ;NO
001254' 016767  176752  176616         MOV     CSR,CSRA
001262' 104403  000000' 002120'        MSGNS,BEGIN,MBITC       ;ASCII MESSAGE CALL WITH COMMON HEADER
001270' 012767  000025  176610         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001276' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;MAINTENANCE BIT IS NOT CLEARING
                                       ;*****************************************************************

001304' 152777  000100  176720  12$:    BISB    #100,@CSR               ;SET EBIT
001312' 132777  000100  176712         BITB    #100,@CSR
001320' 001017                         BNE     13$
001322' 117767  176704  176552         MOVB    @CSR,ACSR
001330' 016767  176676  176542         MOV     CSR,CSRA
001336' 104403  000000' 002124'        MSGNS,BEGIN,EBITS       ;ASCII MESSAGE CALL WITH COMMON HEADER
001344' 012767  000025  176534         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001352' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;INTERRUPT ENABLE BIT IS NOT SETTING
                                       ;*****************************************************************
```



```
001360' 142777  000100  176644  13$:    BICB    #100,@CSR               ;CLEAR EBIT
001366' 132777  000100  176636         BITB    #100,@CSR
001374' 001417                         BEQ     14$
001376' 117767  176630  176476         MOVB    @CSR,ACSR
001404' 016767  176622  176466         MOV     CSR,CSRA
001412' 104403  000000' 002130'        MSGNS,BEGIN,EBITC       ;ASCII MESSAGE CALL WITH COMMON HEADER
001420' 012767  000025  176460         MOV     #25,ERRTYP      ;BIT STUCK
                                       ;*****************************************************************
001426' 104405  000000' 000000         HRDERS,BEGIN,NULL       ;INTERRUPT ENABLE BIT IS NOT CLEARING
                                       ;*****************************************************************

                                       ;THIS TEST WILL CHECK ALL ADDRESSES WITH MBIT SET

001434' 112777  000040  176570  14$:    MOVB    #40,@CSR                ;SET MBIT
001442' 016702  176574                 MOV     BASE,R2
001446' 005000                         CLR     R0
001450' 005001                         CLR     R1
001452' 152777  000001  176552  15$:    BISB    #1,@CSR                 ;SET RIF BIT
001460' 112201                         MOVB    (R2)+,R1                ;READ ADDRESS 171000+R0 AND
001462' 042701  177400                 BIC     #177400,R1
                                                                       ;LOAD CONTENTS INTO R1

001466' 020001                         CMP     R0,R1                   ;R1 SHOULD BE EQUAL R2
001470' 001416                         BEQ     16$                     ;YES
001472' 010067  176410                 MOV     R0,ASB                  ;NO, SAVE GOOD DATA
001476' 010167  176406                 MOV     R1,AWAS                 ;    SAVE BAD DATA
001502' 016767  176334  176372         MOV     BASE,ACSR               ;    SAVE ADDRESS
001510' 150067  176366                 BISB    R0,ACSR
001514' 016767  176512  176356         MOV     CSR,CSRA
                                       ;*****************************************************************
001522' 104404  000000'                DATERS,BEGIN            ;DATA ERROR!!!
                                       ;*****************************************************************
001526' 005200                 16$:    INC     R0                      ;GO TO NEXT ADDRESS
001530' 122700  000376                 CMPB    #376,R0                 ;IS IT LAST ONE
001534' 001346                         BNE     15$                     ;NO, DO IT AGAIN

                                       ;THIS TEST CHECKS MAINTENANCE INTERRUPT
                                       ;IF MBIT & EBIT ARE SET 10CM GENERATES
                                       ;INTERRUPT AT ADDRESS 234 AND IAR HAS
                                       ;UPPER BYTE OF CSR ADDRESS (377)

001536' 012777  001672' 176244  17$:    MOV     #20$,@VECTOR            ;SET INTERRUPT VECTOR
001544' 016767  176240  176472         MOV     VECTOR,VECT2            ;SET VECTOR + 2
001552' 062767  000002  176464         ADD     #2,VECT2
001560' 116777  176226  176456         MOVB    BR1,@VECT2              ;SET PRIORITY LEVEL ON INTERRUPT
001566' 005067  176444                 CLR     INTFLG                  ;CLEAR INTERRUPT FLAG
001572' 012767  000005  176424         MOV     #5,TEMP1                ;SET COUNT FOR TIMEOUT
001600' 152777  000140  176424         BISB    #140,@CSR               ;ENABLE INTERRUPT

001606'                        18$:
001606' 104407  000000'                BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
001612' 104407  000000'                BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.

001616' 005767  176414                 TST     INTFLG                  ;INTERRUPT OCCURR?
001622' 001031                         BNE     21$                     ;YES, 21$
001624' 005367  176374                 DEC     TEMP1                   ;NO, IS IT TIMEOUT
```

```
                                        BNE     18$                     ;NO, LOOP TO BREAK
001632' 117767  176374  176242          MOVB    @CSR,ACSR
001640' 016767  176366  176232          MOV     CSR,CSRA
001646' 104403  000000' 002134'         MSGN$,BEGIN,NOINT       ;ASCII MESSAGE CALL WITH COMMON HEADER
001654' 012767  000023  176224          MOV     #23,ERRTYP              ;NO INTERRUPT
                                        ;*****************************************************************
001662' 104405  000000' 000000          ERRORS,BEGIN,NULL       ;NO INTERRUPT
                                        ;*****************************************************************

001670' 000444                          BR      22$

001672' 142777  000100  176332  20$:    BICB    #100,@CSR               ;CLEAR INTERRUPT ENABLE
001700' 005267  176332                  INC     INTFLG                  ;GET INTERRUPT FLAG
001704' 000002                          RTI

001706' 152777  000001  176316  21$:    BISB    #1,@CSR                 ;SET RIF BIT
001714' 117700  176314                  MOVB    @IAR,R0                 ;CHECK IF IAR = 377
001720' 122700  000377                  CMPB    #377,R0
001724' 001426                          BEQ     22$                     ;YES, GO TO END
001726' 110067  176156                  MOVB    R0,AWAS                 ;NO, SAVE BAD DATA
001732' 012767  000377  176146          MOV     #377,ASB                ;    SAVE GOOD DATA
001740' 117767  176270  176134          MOVB    @IAR,@CSR               ;    SAVE ADDRESS
001746' 016767  176262  176124          MOV     IAR,CSRA
001754' 104403  000000' 002140'         MSGN$,BEGIN,IARERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
001762' 012767  000001  176116          MOV     #1,ERRTYP               ;DATA ERROR
                                        ;*****************************************************************
001770' 104405  000000' 000000          ERRORS,BEGIN,NULL       ;WRONG DATA IN IAR AFTER INTERRUPT
                                        ;*****************************************************************
                                        ;*****************************************************************
001776' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

002002' 012703  000041          22$:    MOV     #11.*3.,R3              ;SET UP WAIT LOOP
002006' 152777  000002  176216          BISB    #2,@CSR                 ;CLEAR CSR
002014' 000240          60$:    NOP                             ;WAIT
002016' 005303                  DEC     R3
002020' 001375                  BNE     60$                     ;WAIT SOME MORE
002022' 012703  000036                  MOV     #10.*3.,R3              ;SET UP ANOTHER LOOP
002026' 152777  000002  176176          BISB    #2,@CSR                 ;CLEAR AGAIN
002034' 000240          65$:    NOP
002036' 005303                  DEC     R3
002040' 001375                  BNE     65$
002042'                 23$:
002042' 104413  000000'                 ENDIT$,BEGIN                    ;SIGNAL END OF ITERATION.
                                                                        ;MONITOR SHALL TEST END OF PASS
002046' 000167  176222                  JMP     RESTRT
002052'                 FINI:
002052' 104410  000000'                 END$,BEGIN              ;

                                        .EVEN

002056' 002144'                 RIFNOT: MES1
002060' 177777                          177777
```



```
002062' 002173'                 RIFCLR: MES2
002064' 177777                          177777

002066' 002223'                 GBITS:  MES3
002070' 177777                          177777

002072' 002263'                 GBITC:  MES4
002074' 177777                          177777

002076' 002324'                 DBITS:  MES5
002100' 177777                          177777

002102' 002350'                 TBITS:  MES6
002104' 177777                          177777

002106' 002374'                 TBITC:  MES7
002110' 177777                          177777

002112' 002421'                 DBITC:  MES8

002114' 002446'                 MBITS:  MES9
002116' 177777                          177777

002120' 002472'                 MBITC:  MES10
002122' 177777                          177777

002124' 002517'                 EBITS:  MES11
002126' 177777                          177777

002130' 002563'                 EBITC:  MES12
002132' 177777                          177777

002134' 002630'                 NOINT:  MES13
002136' 177777                          177777

002140' 002666'                 IARERR: MES14
002142' 177777                          177777

002144' 044522  020106  044502  MES1:   .ASCIZ  "RIF BIT IS NOT SETTING"
002152' 020124  051511  047040
002160' 052117  051440  052105
002166' 044524  043516      000
002173'     122 043111  041040  MES2:   .ASCIZ  "RIF BIT IS NOT CLEARING"
002200' 052111  044440  020123
002206' 041516  020124  046103
002214' 040505  044522  043516
002222'     000
002223'     107 047105  051105  MES3:   .ASCIZ  "GENERIC CODE BIT IS NOT SETTING"
002230' 041511  041440  042117
002236' 020105  044502  020124
002244' 051511  047040  052117
002252' 051440  052105  044524
002260' 043516      000
002263'     107 047105  051105  MES4:   .ASCIZ  "GENERIC CODE BIT IS NOT CLEARING"
002270' 041511  041440  042117
002276' 020105  044502  020124
```

```
  503  002304- 051511  047040  052117
  504  002312- 041440  042514  051101
  505  002320- 047111  000107
  506  002324- 041104  052111  044440  MES5:   .ASCIZ  "DBIT IS NOT SETTING"
  507  002332- 020123  047516  020124
  508  002340- 042523  052124  047111
  509  002346- 000107
  510  002350- 041124  052111  044440  MES6:   .ASCIZ  "TBIT IS NOT SETTING"
  511  002356- 020123  047516  020124
  512  002364- 042523  052124  047111
  513  002372- 000107
  514  002374- 041124  052111  044440  MES7:   .ASCIZ  "TBIT IS NOT CLEARING"
  515  002402- 020123  047516  020124
  516  002410- 046103  040505  044522
  517  002416- 043516      000
  518  002421-    104  044502  020124  MES8:   .ASCIZ  "DBIT IS NOT CLEARING"
  519  002426- 051511  047040  052117
  520  002434- 041440  042514  051101
  521  002442- 047111  000107
  522  002446- 041115  052111  044440  MES9:   .ASCIZ  "MBIT IS NOT SETTING"
  523  002454- 020123  047516  020124
  524  002462- 042523  052124  047111
  525  002470- 000107
  526  002472- 041115  052111  044440  MES10:  .ASCIZ  "MBIT IS NOT CLEARING"
  527  002500- 020123  047516  020124
  528  002506- 046103  040505  044522
  529  002514- 043516      000
  530  002517-    111  052116  051105  MES11:  .ASCIZ  "INTERRUPT ENABLE BIT IS NOT SETTING"
  531  002524- 052522  052120  042440
  532  002532- 040516  046102  020105
  533  002540- 044502  020124  051511
  534  002546- 047040  052117  051440
  535  002554- 052105  044524  043516
  536  002562-    000
  537  002563-    111  052116  051105  MES12:  .ASCIZ  "INTERRUPT ENABLE BIT IS NOT CLEARING"
  538  002570- 052522  052120  042440
  539  002576- 040516  046102  020105
  540  002604- 044502  020124  051511
  541  002612- 047040  052117  041440
  542  002620- 042514  051101  047111
  543  002626- 000107
  544  002630- 047516  046440  044501  MES13:  .ASCIZ  "NO MAINTENANCE MODE INTERRUPT"
  545  002636- 052116  047105  047101
  546  002644- 042503  046440  042117
  547  002652- 020105  047111  042524
  548  002660- 051122  050125  000124
  549  002666- 051127  047117  020107  MES14:  .ASCIZ  "WRONG DATA IN IAR AFTER INTERRUPT"
  550  002674- 040504  040524  044440
  551  002702- 020116  040511  020122
  552  002710- 043101  042524  020122
  553  002716- 047111  042524  051122
  554  002724- 050125  000124
  555
  556                                   .EVEN
  557
  558          000001                   .END
```



```
ACSR     000102R          148#  212*  223*  234*  245*  256*  267*  278*  294*  305*  316*  327*  338*
                          361*  392*  393*  413*
ADDR     000006R          114#  182
ADDR22=  001000           166#
ASB      000106R          152#  202*  359*  412*
ASTAT    000104R          150#
AWAIS    000110R          153#  201*  360*  411*
BASE     000242R          175#  187*  349   361
BEGIN    000000R          111#  205   214   218   225   228   236   239   247   250   258   261   269
                          272   280   283   296   299   307   310   318   321   329   332   340   343
                          365   385   386   395   398   415   418   421   435   439

BIT0   = 000001           166#
BIT1   = 000002           166#
BIT10  = 002000           166#
BIT11  = 004000           166#
BIT12  = 010000           166#
BIT13  = 020000           166#
BIT14  = 040000           166#
BIT15  = 100000           166#
BIT2   = 000004           166#
BIT3   = 000010           166#
BIT4   = 000020           166#
BIT5   = 000040           166#
BIT6   = 000100           166#
BIT7   = 000200           166#
BIT8   = 000400           166#
BIT9   = 001000           166#
BREAK$= 104407            166#  385   386
BR1      000012R          116#  379
BR2      000013R          117#
BTOD$  = 104421           166#
CDATA$= 104412            166#
CNT      000240R          174#  181*
CONFIG   000056R          136#
CSR      000232R          171#  185*  190*  195*  199   201   209*  210   212   213   221   223   224
                          231*  232   234   235   242*  243   245   246   253*  254   256   257   264*
                          265   267   268   275*  276   278   279   287*  291   294   295   302*  303
                          305   306   313*  314   316   317   324*  325   327   328   335*  336   338
                          339   348*  352*  363   382*  393   394   403*  407*  425*  430*
CSRA     000100R          146#  213*  224*  235*  246*  257*  268*  279*  295*  306*  317*  328*  339*
                          363*  394*  414*
DATCK$= 104411            166#
DATER$= 104404            166#  205   365   421
DBITC    002112R          296   465#
DBITS    002076R          258   456#
DVTD1    000014R          118#
EBITC    002130R          340   476#
EBITS    002124R          329   473#
ENDITS= 104413            166#  435
END$   = 104410           166#  439
ERRTYP   000106R          151#  216*  226*  237*  248*  259*  270*  281*  297*  308*  319*  330*  341*
                          396*  416*
EXIT$  = 104400           166#
FINI     002052R          438#
GBITC    002072R          247   453#
GBITS    002066R          236   450#
```

```
GETPA$= 104415           166#
GWBUF$= 104414           166#
HRDCNT  000044R          131#
HRDER$= 104405           166#    218     228     239     250     261     272     283     299     310     321     332     343
                         398     418
HRDPAS  000050R          133#
IAR     000234R          172#    183*    408     413     414
IARERR  002140R          415     462#
ICONT   000036R          128#
ICOUNT  000040R          129#
IDNUM   000122R          158#
INIT    000030R          125#
INTFLG  000236R          173#    380*    388     404*
INTR    000220R          157#    178*
MAP22$= 104416           166#
MBITC   002120R          318     470#
MBITS   002114R          307     467#
MES1    002144R          443     495#
MES10   002472R          470     526#
MES11   002517R          473     530#
MES12   002563R          476     537#
MES13   002630R          479     544#
MES14   002666R          482     549#
MES2    002173R          447     489#
MES3    002223R          450     494#
MES4    002263R          453     500#
MES5    002324R          456     506#
MES6    002350R          459     510#
MES7    002374R          462     514#
MES8    002411R          465     518#
MES9    002446R          467     522#
MODNAM  000000R          112#
MODSP   000224R          126     164#
MSGN$ = 104403           166#    214     225     236     247     258     269     280     296     307     318     329     340
                         395     415
MSGS$ = 104402           166#
MSGS  = 104401           166#
NOINT   002134R          395     479#
NULL  = 000000           166#    218     228     239     250     261     272     283     299     310     321     332     343
                         398     418
OPEN  = 000000           113     119     120     121     122     139     140     141     142     143     144     145     146
                         148     150     152     153     155     156     157     166#
OTOAS = 104420           166#
PASCNT  000034R          127#
PIRQS = 000004           166#
POPSP = 005726           166#
POPSP2= 022626           166#
PRTV  = 000000           166#
PRTV0 = 000000           116     117     166#
PRTV1 = 000040           166#
PRTV2 = 000100           166#
PRTV3 = 000140           166#
PRTV4 = 000200           166#
PRTV5 = 000240           166#
PRTV6 = 000300           166#
PRTV7 = 000340           166#
```



```
PS    = 177776           166#
PSW   = 177776           166#
PUSH  = 005746           166#
PUSH2 = 024546           166#
RANDS = 104417           166#
RANNUM  000054R          135#
RESTRT  000274R          154     182#    437
RES1    000056R          137#
RES2    000060R          138#
RIPCLR  002062R          225     447#
RIPNDT  002056R          214     443#
RSTRT   000112R          154#
SBADR   000102R          147#
SOFCNT  000042R          130#
SOFER$= 104406           166#
SOFPAS  000046R          132#
SPOINT  000032R          126#
SPSIZ = 000040             1#    159
SR1     000016R          119#
SR2     000020R          120#
SR3     000022R          121#
SR4     000024R          122#
START   000246R          125     178#
STAT    000026R          124#
SVR0    000062R          139#
SVR1    000064R          140#
SVR2    000066R          141#
SVR3    000070R          142#
SVR4    000072R          143#
SVR5    000074R          144#
SVR6    000076R          145#
SYSCNT  000052R          134#
TBITC   002106R          280     462#
TBITS   002102R          269     459#
TEMP1   000224R          168#    381*    390*
TEMP2   000226R          169#
TEMP3   000230R          170#
TRPDFD= 000022           166#
VECTOR  000010R          115#    376*    377
VECT2   000244R          176#    377*    378*    379*
WASADR  000104R          148#
WDFR    000116R          156#    180*
WDTO    000114R          155#    179*
XFLAG   000005R          113#

. ABS.  000000      000
        002730      001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XPCSB0,XPCSB0/SOL/CRF:SYM=DDXCOM,XPCSB0
 RUN-TIME:  1 1  .2 SECONDS
 RUN-TIME RATIO: 26/3=6.7
```

CORE USED: 7K (13 PAGES)