.REM _

IDENTIFICATION

PRODUCT CODE: AC-E995C-MC

PRODUCT NAME: CXDTAC0 DTE20 MODULE

PRODUCT DATE: SEPTEMBER 1978

MAINTAINER: DEC/X11 SUPPORT GROUP

1.0 ABSTRACT

"DTA" IS AN IOMOD THAT WILL EXERCISE UP TO FOUR DTE20'S
SEQUENTIALLY. IT USES THE DIAGNOSTIC MODE TO VERIFY THE DTE20
UNIBUS INTERFACE AND THE LOGIC CONTROLLING THIS INTERFACE.
IT PERFORMS SIMPLE RAM READ/WRITE AND ADDRESSING TESTS ALONG
WITH VERIFICATION OF THE VECTORED INTERRUPT AND "NPR" FUNCTIONS.
IT MAKES NO ATTEMPT TO COMMUNICATE WITH THE KL10 SIDE OF THE DEVICE.

2.0 REQUIREMENTS

HARDWARE: A PDP11 COMPUTER SYSTEM WITH AT LEAST ONE
DTE20 KL10 INTERFACE.

STORAGE:: DTA REQUIRES:
1. DECIMAL WORDS: 852
2. OCTAL WORDS: 1524
3. OCTAL BYTES: 3250

3.0 PASS DEFINITION

THE FIRST PASS OF "DXDTA" CONSISTS OF EXECUTING EACH SUB-
TEST ONE TIME. SUBSEQUENT PASSES CONSIST OF 100(8) ITER-
ATIONS OF THE TEST SEQUENCE FOR EACH DTE20 FOUND.

4.0 EXECUTION TIME

PASS TIME WILL VARY DEPENDING UPON THE NO. OF DTE20'S SELECTED
AND THE CONFIGURATION BEING EXERCISED.

5.0 CONFIGURATION PARAMETERS

DEFAULT PARAMETERS:

DVA: 174400 VCT: 774 BR1: 4 BR2: 0 DVC: 1

REQUIRED PARAMETERS:

TO EXERCISE MORE THAN ONE DTE20 "DVC" MUST BE SET UP AS
DESCRIBED IN PARA. 8.0 BELOW.

6.0 DEVICE OPTION SET-UP

NONE REQUIRED

7.0 MODULE OPERATION

BASIC TEST SEQUENCE:

DT01: VERIFY THAT ALL ZEROES CAN BE WRITTEN AND READ FROM THE "DELAY COUNTER" REGISTER

DT02: VERIFY THAT ALL ZEROES CAN BE WRITTEN AND READ FROM ALL ACTIVE "RAM" LOCATIONS

DT03: VERIFY THAT ALL ONES CAN BE WRITTEN AND READ FROM THE "DELAY COUNTER" REGISTER

DT04: VERIFY THAT ALL ONES CAN BE WRITTEN AND READ FROM ALL ACTIVE "RAM" LOCATIONS

DT05: VERIFY THAT EACH "RAM" LOCATION IS UNIQUELY ADDRESSABLE

DT06: VERIFY THAT THE "RMF=0" BIT DOES NOT SET WHEN A FLOATING ONE IS READ OUT OF THE "RAM"

DT07: VERIFY THAT THE "TO10 DONE" BIT CAN CAUSE AN INTERRUPT TO THE PROPER VECTOR

DT10: VERFIY THAT THE "10 REQ INT" BIT CAN CAUSE AN INTERRUPT TO THE PROPER VECTOR

DT11: VERIFY THAT THE "TO11 DONE" BIT CAN CAUSE AN INTERRUPT TO THE PROPER VECTOR

DT12: VERIFY THAT THE "TO10 ER" CAN CAUSE AN INTERRUPT TO THE PROPER VECTOR

DT13: VERIFY THAT THE "TO11 ER" BIT CAN CAUSE AN INTERRUPT TO THE PROPER VECTOR

DT14: VERIFY THAT "MSTR CLR" CAN CLEAR THE "TO11 BC" REG.

DT15: VERIFY THAT THE "ABC" REGISTER INCREMENTS DURING A "TO11 TRANSFER"

DT16: VERIFY THAT THE "TO11 DONE" BIT SETS PROPERLY

DT17: VERIFY THAT THE "STST NULL" FLOP CAN BE SET PROPERLY

DT20: VERIFY THAT THE "ABC" REGISTER INCREMENTS DURING A "TO10" E-BUFF FILL

8.0 OPERATOR OPTIONS

RELATIVE LOCATION "DTA 14" (DVID1) MUST BE MODIFIED TO EXERCISE MORE THAN ONE DTE20 AS SHOWN BELOW:

```
DVID1  BIT00=1  DTE20 #0        (DEFAULT)
DVID1  BIT01=1  DTE20 #1
DVID1  BIT02=1  DTE20 #2
DVID1  BIT03=1  DTE20 #3
```

TO DESELECT A DTE20 THE APPROPRIATE BIT IN "DVID1" MUST BE SET TO A ZERO. IF THE PROGRAM FINDS ALL FOUR BITS = "0" THE MODULE WILL BE DROPPED.

9.0 NON STANDARD PRINTOUTS

DTA USES THE DATA ERROR PRINTOUT IN SOME CASES TO REPORT OTHER THAN NORMAL DATA ERRORS. REFER TO THE ACTUAL ERROR CALL "APC" TO LOCATE THE CALL IN THE LISTING AND ANALYZE THE INSTRUCTIONS PRECEDING THE "DATERS" CALL TO OBTAIN THE INTERPRETION OF THE INFORMATION PRINTED.

```
000000'  TOMOD   <DTAC >,174400,774,4,0,0,4,157
000000'           MODULE  140000,DTAC ,174400,774,4,0,0,4,157
                  .TITLE  DTAC DEC/X11 SYSTEM EXERCISER MODULE
                ;       DDXCOM  VERSION  6      23-MAY-78
                                .LIST   BIN
                                ;*****************************************************************************
000000'                         BEGIN:
000000' 052104  041501      040 MODNAM: .ASCII  /DTAC / ;MODULE NAME.
000005'     000                 XFLAG:  .BYTE   OPEN                ;USED TO KEEP TRACK OF WBUFF USAGE
000006' 174400                  ADDR:   174400+0                    ;1ST DEVICE ADDR.
000010' 000774                  VECTOR: 774+0                       ;1ST DEVICE VECTOR.
000012'     200                 BR1:    .BYTE   PRTY4+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' 000224'                 INIT:   START                       ;MODULE START ADDR.
000032' 000224'                 SPOINT: MODSP                       ;MODULE STACK POINTER.
000034' 000000                  PASCNT: 0                           ;PASS COUNTER.
000036' 000004                  ICONT:  4                           ;# OF ITERATIONS PER PASS=4
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, OP
000102' 000000                  ACSR:   OPEN                        ;CONTENTS OF CSR.
000104'                         WASADR:                             ;ADDR OF BAD DATA, OP
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' 000246'                 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' 000157                  IDNUM:  157                         ;MODULE IDENTIFICATION NUMBER=157
        000040                          .REPT   SPSIZ               ;MODULE STACK STARTS HERE.
```



```
                                .NLIST
                                .WORD   0
                                .LIST
                                .ENDR
000224'                         MODSP:
                                ;*****************************************************************************
  239
  240
  241           000040          DTESIZ= 000040                      ;16 REGISTERS PER DTE20
  242           000004          DTEMAX= 000004                      ;UP TO FOUR DTE'S CAN BE TESTED
  243           020000          TO11BM= BIT13                       ;TO-11 BYTE MODE
  244           002000          DS05=   BIT10                       ;DIAGNOSTIC STATUS
  245           000060          PULSE=  BIT4!BIT5                   ;SINGLE PULSE THE 10/11 CLOCK
  246           000040          D1011=  BIT5                        ;10/11 INTERFACE DIAGNOSTIC MODE
  247           000200          TO10=   BIT7                        ;INTERFACE MAJOR STATE - TO10 TRANSFER
  248           040000          EDONES= BIT14                       ;SET E-BUS DONE
  249           000100          DRESET= BIT6                        ;PERFORM DIAGNOSTIC CLEAR
  250           000040          INTRON= BIT5                        ;ENABLE DTE20 TO INTR. THE 11
  251           000002          ERR11S= BIT1                        ;SET TO11 ERROR
  252           100000          TO10DN= BIT15                       ;TO10 DONE
  253           010000          RAMIS0= BIT12                       ;OUTPUT READ FROM RAM IS ALL ZEROES
  254           000400          TO10DB= BIT8                        ;REQ 10 INTERRUPT - DOORBELL FROM 11
  255           000040          NULSTP= BIT5                        ;NULL STOP
  256           020000          ERR10S= BIT13                       ;SET TO 10 ERROR
  257           100000          DON10S= BIT15                       ;SET TO 10 DONE
  258           000200          DON11S= BIT7                        ;SET TO 11 DONE
  259           000100          TO11=   BIT6                        ;INTERFACE MAJOR STATE - TO11 XFR
  260           004000          TO11DB= BIT11
  261           000200          TO11DN= BIT7
  262           040000          ZSTOP=  BIT14           ;STOP ON NULL (ZERO) CHAR
  263
  264 000224' 016767  177564  003000  START:  MOV     DVID1,TDVD1             ;GET DEVICE SELECT PARAMETER
  265 000232' 042767  177760  002772          BIC     #177760,TDVD1           ;CLEAR OUT UNUSED BITS
  266 000240' 001002                          BNE     RESTRT                  ;BR IF ANY DTE20'S SELECTED
  267 000242' 104410  000000'                 END$,BEGIN                      ;
  268
  269 000246' 016767  002760  002760  RESTRT: MOV     TDVD1,TDVD2             ;SAVE THE SELECT BITS
  270 000254' 016705  177526                  MOV     ADDR,R5                 ;GET THE FIRST DTE20 ADDRESS
  271 000260' 016700  177524                  MOV     VECTOR,R0               ;GET THE FIRST VECTOR ADDRESS
  272 000264' 000257                  1$:     CCC                             ;CLEAR OUT THE "C" BIT
  273 000266' 006067  002742                  ROR     TDVD2                   ;"C" WILL SET IF A DTE IS SELECTED
  274 000272' 103002                          BCC     2$                      ;BR IF SELECT BIT = 0
  275 000274' 004767  000016                  JSR     PC,GOEXDT               ;GO EXERCISE SELECTED DTE
  276 000300' 062705  000040          2$:     ADD     #DTESIZ,R5              ;GENERATE NEXT DTE START ADDR.
  277 000304' 162700  000004                  SUB     #DTEMAX,R0              ;GENERATE NEXT DTE VECTOR ADDR.
  278 000310' 104413  000000'                 ENDIT$,BEGIN                    ;SIGNAL END OF ITERATION.
  279                                                                         ;MONITOR SHALL TEST END OF PASS
  280 000314' 000763                          BR      1$
  281
  282 000316' 012702  003172'         GOEXDT: MOV     #DLYCNT,R2              ;POINT TO FIRST TABLE ENTRY
  283 000322' 005003                          CLR     R3                      ;INIT R3 TO COUNT BY +2
  284 000324' 010512                  1$:     MOV     R5,(R2)                 ;STORE A DTE ADDRESS
  285 000326' 060322                          ADD     R3,(R2)+                ;MAKE IT THE RIGHT ADDRESS
  286 000330' 005723                          TST     (R3)+                   ;ADD +2 TO R3
  287 000332' 022703  000040                  CMP     #DTESIZ,R3              ;STOPED ALL DTE ADDRESSES ??
  288 000336' 001372                          BNE     1$                      ;BR IF NOT
```

```
                                        MOV     ITCNT,ICOUN             ;INITIALIZE ITERATION COUNTER
                                DOAGIN: DEC     ICOUN                   ;COUNT ONE ITERATION

                                        ;ZEROES TEST TO DELAY COUNTER
                                        ;----------------------------

                                DT01:   CLR     ASB                     ;RESULT S/B = 000000
                                        JSR     PC,DIAGRT               ;GO DO DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET UP DIAGNOSTIC MODE
                                        MOV     #0,@DLYCNT              ;ZERO THE DELAY COUNT REG.
                                        MOV     @DLYCNT,AWAS            ;GET THE DELAY COUNT REG.
                                        BEQ     DT02                    ;BR IF DATA CORRECT
                                        MOV     R5,CSRA                 ;SAVE THE ERROR INFO
                                        MOV     DLYCNT,WASADR
                                        MOV     #ASB,SBADR
                                        ;*****************************************************************
                                        DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

                                        ;TEST TO WRITE ZEROES IN ALL ACTIVE RAM LOCATIONS
                                        ;-----------------------------------------------

                                DT02:   CLR     ASB                     ;RESULT S/B = 000000
                                        MOV     DLYCNT,R3               ;POINT R3 TO FIRST RAM LOC.
                                1$:     JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET IT IN DIAGNOSTIC MODE
                                        CLR     (R3)+                   ;ZERO A RAM LOCATION
                                        CMP     R3,DIAG1                ;DONE ALL ACTIVE LOCATIONS ??
                                        BNE     1$                      ;BR IF NOT
                                2$:     JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET DIAGNOSTIC MODE
                                        MOV     -(R3),AWAS              ;GET CONTENTS OF RAM
                                        BEQ     3$                      ;BR IF IT WAS 000000
                                        MOV     R3,WASADR               ;SAVE THE ERROR INFO
                                        MOV     #ASB,SBADR
                                        ;*****************************************************************
                                        DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************
                                3$:     BIT     #RAMISO,@STATUS         ;DID RMF=0 BIT SET
                                        BNE     4$                      ;BR IF IT DID
                                        MOV     R5,CSRA                 ;SAVE THE ERROR INFO
                                        MOV     @STATUS,ASTAT
                                        CLR     ERRTYP
                                        ;*****************************************************************
                                        ARDER$,BEGIN,NULL       ;RMF=0 FAILED TO SET
                                        ;*****************************************************************
                                4$:     CMP     R3,DLYCNT               ;CHECKED ALL LOCATIONS ??
                                        BGT     2$                      ;BR IF NOT

                                        ;TEST TO WRITE ALL 1'S INTO DELAY COUNTER
                                        ;---------------------------------------

                                DT03:   MOV     #-1,ASB                 ;RESULT S/B = 177777
                                        JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET UP DIAGNOSTIC MODE
```



```
                                        MOV     #-1,@DLYCNT             ;ALL 1'S TO DELAY COUNT REG.
                                        MOV     @DLYCNT,AWAS            ;GET THE CONTENTS OF DELAY COUNT
                                        CMP     ASB,AWAS                ;WAS IT ALL ONES ??
                                        BEQ     DT04                    ;BR IF YES
                                        MOV     DLYCNT,WASADR           ;SAVE THE ERROR INFO
                                        MOV     #ASB,SBADR
                                        MOV     R5,CSRA
                                        ;*****************************************************************
                                        DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

                                        ;TEST TO WRITE ALL 1'S INTO ACTIVE RAM LOCATIONS
                                        ;----------------------------------------------

                                DT04:   MOV     DLYCNT,R3               ;R3 POINTS TO 1ST ADDRESS
                                1$:     JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET DIAGNOSTIC MODE
                                        MOV     #-1,(R3)+               ;LOAD ALL 1'S INTO RAM
                                        CMP     R3,DIAG1                ;LOADED ALL LOCATIONS ??
                                        BNE     1$                      ;BR IF NOT
                                2$:     MOV     #-1,ASB                 ;RESULT S/B = 177777
                                        JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET DIAGNOSTIC MODE
                                        MOV     -(R3),AWAS              ;READ A RAM LOCATION
                                        CMP     ASB,AWAS                ;WAS IT 177777 ??
                                        BEQ     3$                      ;BR IF YES
                                        MOV     R3,WASADR               ;SAVE THE ERROR INFO
                                        MOV     #ASB,SBADR
                                        MOV     R5,CSRA
                                        ;*****************************************************************
                                        DATER$,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************
                                3$:     CMP     R3,DLYCNT               ;CHECKED THEM ALL YET ??
                                        BGT     2$                      ;BR IF NOT

                                        ;RAM MEMORY ADDRESS TEST
                                        ;----------------------

                                DT05:   MOV     DLYCNT,R3               ;START WITH 1ST RAM LOC
                                        CLR     R4                      ;R4 CONTAINS FOUR ZEROBYTES (4 BITS EACH)
                                        JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
                                        JSR     PC,DIAGME               ;GO SET DIAGNOSTIC MODE
                                1$:     MOV     R4,(R3)+                ;LOAD A RAM LOCATION-UPDATE POINTER
                                        ADD     #21042,R4               ;ADD +2 TO EACH DATA BYTE
                                        CMP     R3,DIAG1                ;LOADED THEM ALL ??
                                        BNE     1$                      ;BR IF NOT
                                2$:     SUB     #21042,R4               ;INIT R4 TO START CHECKING DATA
                                        MOV     R4,ASB                  ;SAVE S/B DATA
                                        MOV     -2(R3),AWAS             ;GET CONTENTS OF RAM
                                        CMP     R4,-(R3)                ;CORRECT CONTENTS ??
                                        BEQ     3$                      ;BR IF YES
                                        MOV     R3,WASADR               ;SAVE THE ERROR DATA
                                        MOV     #ASB,SBADR
                                        MOV     R5,CSRA
                                        ;*****************************************************************
                                        DATER$,BEGIN            ;DATA ERROR!!!
```

```
                                ;*****************************************************************
001056' 005704                  3$:     TST     R4                      ;BACK TO 000000 DATA
001060' 001354                          BNE     2$                      ;BR IF NOT

                                ;TEST THAT RMF=0 DOES NOT SET WITH FLOATING ONE OUT OF RAM
                                ;-------------------------------------------------------

001062' 004767  001652          DT06:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001066' 005067  177014                  CLR     ASR                     ;RESULT S/B = 000000
001072' 012703  000001                  MOV     #BIT0,R3                ;INIT R3 TO LOAD DLYCNT
001076' 010377  002070          1$:     MOV     R3,@DLYCNT              ;LOAD FLOATING ONE INTO DLYCNT
001102' 017767  002064  177000          MOV     @DLYCNT,AWAS            ;READ IT BACK OUT
001110' 032777  010000  002110          BIT     #RAMISO,@STATUS         ;DID RMF=0 GET SET ??
001116' 001415                          BEQ     2$                      ;BR IF NOT -- ITS OK
001120' 016767  176764  176754          MOV     AWAS,ACSR               ;(CSRC) = DATA READ
001126' 010567  176746                  MOV     R5,CSRA                 ;SAVE ERROR INFO
001132' 017767  002070  176744          MOV     @STATUS,ASTAT
001140' 005067  176742                  CLR     ERRTYP
                                ;*****************************************************************
001144' 104405  000000' 000000          ERDER$  BEGIN,NULL      ;RMF=0 BIT FAILED TO GET CLEARED
                                ;*****************************************************************
001152' 000241                  2$:     CLC                             ;PUT 0'S IN ON RIGHT SIDE
001154' 006103                          ROL     R3                      ;ROTATE THE FLOATING ONE
001156' 001347                          BNE     1$                      ;BR TILL IT COMES OUT ON LEFT

                                ;TEST THAT "TO10 DONE" CAUSES A VECTORED INTERRUPT
                                ;-------------------------------------------------

001160' 004767  001554          DT07:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001164' 005067  002050                  CLR     INTFLG                  ;CLEAR SOFTWARE FLAG
001170' 012770  003164'                 MOV     #DTINT,(R0)+            ;GO TO DTINT ON INTERRUPT
001174' 116710  176612                  MOVB    BR1,(R0)
001200' 005740                          TST     -(R0)                   ;RESET VECTOR POINTER
001202' 012777  100040  002016          MOV     #DON10S!INTRON,@STATUS  ;ENABLE "TO10" TO CAUSE INTR.
001210' 000240                          NOP                             ;DELAY A LITTLE
001212' 000240                          NOP                             ;TO ALLOW INTERRUPT
001214' 000240                          NOP
001216' 005767  002016                  TST     INTFLG                  ;DID THE INTERRUPT OCCUR ??
001222' 001015                          BNE     DT10                    ;BR IF IT DID
001224' 010567  176650                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
001230' 012767  100040  176646          MOV     #DON10S!INTRON,ASTAT
001236' 004767  001476                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001242' 012767  000023  176636          MOV     #23,ERRTYP
                                ;*****************************************************************
001250' 104405  000000' 000000          ERDER$  BEGIN,NULL      ;"TO10 DONE" FAILED TO CAUSE INTERRUPT
                                ;*****************************************************************

                                ;TEST THAT "10 REQ INT" CAN GENERATE VECTORED INTERRUPT
                                ;-----------------------------------------------------

001256' 004767  001456          DT10:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001262' 005067  001752                  CLR     INTFLG                  ;CLEAR SOFTWARE FLAG
001266' 012777  004040  001732          MOV     #TO11D8!INTRON,@STATUS  ;ENABLE "10 REQ INT" TO CAUSE INTR
001274' 000240                          NOP                             ;A LITTLE STALL
001276' 000240                          NOP
001300' 000240                          NOP
```



```
001302' 005767  001732                  TST     INTFLG                  ;DID THE INTR. OCCUR ??
001306' 001015                          BNE     DT11                    ;BR IF YES
001310' 010567  176564                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
001314' 012767  004040  176562          MOV     #TO11D8!INTRON,ASTAT
001322' 004767  001412                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001326' 012767  000023  176552          MOV     #23,ERRTYP
                                ;*****************************************************************
001334' 104405  000000' 000000          ERDER$  BEGIN,NULL      ;"10 REQ INT" FAILED TO GENERATE INTERRUPT
                                ;*****************************************************************

                                ;TEST THAT "TO11 DONE" CAN GENERATE A VECTURED INTERRUPT
                                ;------------------------------------------------------

001342' 004767  001372          DT11:   JSR     PC,DIAGRT               ;GO DO DIAGNOSTIC RESET
001346' 005067  001666                  CLR INTFLG                      ;CLEAR SOFTWARE INTR. FLAG
001352' 012777  000240  001646          MOV     #DON11S!INTRON,@STATUS  ;ENABLE "TO11 DONE" TO CAUSE INTR
001360' 000240                          NOP                             ;WAIT A LITTLE
001362' 000240                          NOP
001364' 000240                          NOP
001366' 005767  001646                  TST     INTFLG                  ;DID INTR. OCCUR ??
001372' 001015                          BNE     DT12                    ;BR IF IT DID
001374' 010567  176500                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
001400' 012767  000240  176476          MOV     #DON11S!INTRON,ASTAT    ;
001406' 004767  001326                  JSR     PC,DIAGRT               ;GO DO DIAGNOSTIC RESET
001412' 012767  000023  176466          MOV     #23,ERRTYP
                                ;*****************************************************************
001420' 104405  000000' 000000          ERDER$  BEGIN,NULL      ;"TO11 DONE FAILED TO CAUSE INTERRUPT
                                ;*****************************************************************

                                ;TEST THAT "TO10 ER" CAN CAUSE A VECTORED INTERRUPT
                                ;--------------------------------------------------

001426' 004767  001306          DT12:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001432' 005067  001602                  CLR     INTFLG                  ;INIT SOFTWARE FLAG
001436' 012777  020040  001562          MOV     #ERR10S!INTRON,@STATUS  ;ENABLE "TO10ER" TO CAUSE INTR.
001444' 000240                          NOP                             ;A LITTLE STALL
001446' 000240                          NOP
001450' 000240                          NOP
001452' 005767  001562                  TST     INTFLG                  ;DID INTR. OCCUR ??
001456' 001015                          BNE     DT13                    ;BR IF IT DID
001460' 010567  176414                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
001464' 012767  020040  176412          MOV     #ERR10S!INTRON,ASTAT
001472' 004767  001242                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001476' 012767  000023  176402          MOV     #23,ERRTYP
                                ;*****************************************************************
001504' 104405  000000' 000000          ERDER$  BEGIN,NULL      ;"TO10ER" FAILED TO CAUSE AN INTR.
                                ;*****************************************************************

                                ;TEST THAT "TO11ER" CAN CAUSE A VECTURED INTERRUPT
                                ;-------------------------------------------------

001512' 005067  001522          DT13:   CLR     INTFLG
001516' 012777  000042  001502          MOV     #ERR11S!INTRON,@STATUS  ;ENAR "TO11ER" TO CAUSE INTR.
001524' 000240                          NOP                             ;A LITTLE STALL
001526' 000240                          NOP
001530' 000240                          NOP
```

```
001532' 005767  001502                  TST     INTFLG                  ;DID THE INTR OCCUR ??
001536' 001015                          BNE     1$                      ;BR IF IT DID
001540' 010567  176334                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
001544' 012767  000042  176332          MOV     #ERR11S!INTRON,ASTAT
001552' 004767  001162                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001556' 012767  000023  176322          MOV     #23,ERRTYP
                                        ;*****************************************************************
001564' 104405  000000' 000000          $RDERS,BEGIN,NULL       ;"TO11ER" FAILED TO CAUSE AN INTERRUPT
                                        ;*****************************************************************
001572' 010010                  1$:     MOV     R0,(R0)                 ;RESTORE TRAP CATCHER IN DTE
001574' 062720  000002                  ADD     #2,(R0)+                ;BEFORE LEAVING
001600' 005010                          CLR     (R0)
001602' 005740                          TST     -(R0)

                                ;TEST TO VERIFY "MSTR CLR" CAN CLEAR "TO11 BC"
                                ;---------------------------------------------

001604' 004767  001130          DT14:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001610' 016767  001432  176270          MOV     NPRLOC,ASB              ;GET NPR ADDRESS
001616' 012777  007777  001364          MOV     #7777,@TO11BC           ;LOAD TO11 BYTE COUNT
001624' 004767  001110                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTID RESET
001630' 016777  001412  001356          MOV     NPRLOC,@TO11AD          ;LOAD TO 11 ADDRESS
001636' 000240                          NOP                             ;STALL A LITTLE
001640' 000240                          NOP
001642' 000240                          NOP
001644' 017767  001344  176236          MOV     @TO11AD,AWAS            ;GET THE TO 11 ADDRESS
001652' 026767  176230  176230          CMP     ASB,AWAS                ;DID IT GET MODIFIED ??
001660' 001412                          BEQ     DT15                    ;BR IF NOT
001662' 016767  001326  176214          MOV     TO11AD,WASADR           ;SAVE THE ERROR INFO
001670' 012767  000106' 176204          MOV     #ASB,SBADR
001676' 010567  176176                  MOV     R5,CSRA
                                        ;*****************************************************************
001702' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

                                ;TEST TO VERIFY ABC REG INCREMENTS DURING TO11 TRANSFER
                                ;-----------------------------------------------------

001706' 016767  001334  176172  DT15:   MOV     NPRLOC,ASB              ;GET THE NPR ADDRESS
001714' 005267  176166                  INC     ASB                     ;INCREMENT IT
001720' 004767  001014                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
001724' 016703  001316                  MOV     NPRLOC,R3               ;GET NPR ADDRESS AGAIN
001730' 062703  000002                  ADD     #2,R3                   ;ADD +2 TO IT
001734' 004767  001034                  JSR     PC,T11LOC               ;GO LOCK IN TO11 MAJOR STATE
001740' 005067  176142                  CLR     ERRTYP
                                        ;*****************************************************************
001744' 104405  000000' 000000          $RDERS,BEGIN,NULL       ;FAILED TO LOCK IN "TO11" MAJOR STATE
                                        ;*****************************************************************
001752' 000450                          BR      DT16                    ;GO TO NEXT TEST
001754' 012777  027777  001226          MOV     #TO11BM!7777,@TO11BC    ;LOAD TO 11 BYTE COUNT REG.
001762' 016777  001260  001224          MOV     NPRLOC,@TO11AD          ;LOAD TO 11 ADDR. REG.
001770' 012777  040010  001226          MOV     #10!IEDONES,@DIAG2      ;GO SET UP DIAG REG.
001776' 004767  000722                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002002' 012777  000014  001214          MOV     #14,@DIAG2              ;CHANGE DIAG REG.
002010' 004767  000710                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002014' 012777  000020  001202          MOV     #20,@DIAG2              ;CHANGE THE DIAG REG.
```



```
002022' 004767  000676                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002026' 017767  001162  176054          MOV     @TO11AD,AWAS            ;GET THE TO 11 ADDRESS
002034' 026767  176046  176046          CMP     ASB,AWAS                ;DID IT INCREMENT PROPERLY ??
002042' 001414                          BEQ     DT16                    ;BR IF IT DID
002044' 016767  001144  176032          MOV     TO11AD,WASADR           ;SAVE THE ERROR INFO
002052' 012767  000106' 176022          MOV     #ASB,SBADR
002060' 010567  176014                  MOV     R5,CSRA
002064' 004767  000650                  JSR     PC,DIAGRT               ;GO DO DIAGNOSTIC RESET
                                        ;*****************************************************************
002070' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************

                                ;TEST TO CHECK THE "TO11 DONE" FLAG GETS SET
                                ;-------------------------------------------

002074' 012767  000200  176004  DT16:   MOV     #TO11DN,ASB             ;TO11DN BIT SHOULD GET SET
002102' 004767  000632                  JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
002106' 005077  001076                  CLR     @TO11BC                 ;CLEAR TO 11 BYTE COUNT REG.
002112' 004767  000622                  JSR     PC,DIAGRT               ;GO DO A DIAGRT
002116' 016777  001124  001070          MOV     NPRLOC,@TO11AD          ;LOAD THE TO 11 ADDR REG.
002124' 004767  000644                  JSR     PC,T11LOC               ;GO LOCK IN TO 11 MAJOR STATE
002130' 005067  175752                  CLR     ERRTYP
                                        ;*****************************************************************
002134' 104405  000000' 000000          $RDERS,BEGIN,NULL       ;FAILED TO LOCK IN TO 11 STATE
                                        ;*****************************************************************
002142' 000504                          BR      DT17                    ;GO TO NEXT TEST
002144' 005077  001040                  CLR     @TO11BC                 ;LOAD TO 11 BYTE COUNT REG.
002150' 005077  001050                  CLR     @DIAG2                  ;START AT MINOR STATE "TO11 DLV RD"
002154' 005077  001042                  CLR     @DIAG1                  ;TURN CLOCK BACK ON
002160' 005004                          CLR     R4                      ;INIT BREAK TIMER
002162'                         1$:
002162' 104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
002166' 104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002172' 032777  000100  001022          BIT     #TO11,@DIAG1            ;DID WE LEAVE THE TO11 STATE ??
002200' 001415                          BEQ     2$                      ;BR IF YES
002202' 005304                          DEC     R4                      ;COUNT THE TIMER
002204' 001366                          BNE     1$                      ;BR IF NO TIME OUT
002206' 010567  175666                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
002212' 017767  001004  175664          MOV     @DIAG1,ASTAT
002220' 005067  175662                  CLR     ERRTYP
                                        ;*****************************************************************
002224' 104405  000000' 000000          $RDERS,BEGIN,NULL       ;FAILED TO LEAVE THE TO11 STATE
                                        ;*****************************************************************
002232' 000450                          BR      DT17                    ;GO TO NEXT TEST
002234' 005004                  2$:     CLR     R4                      ;INIT THE TIMER AGAIN
002236'                         3$:
002236' 104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
002242' 104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002246' 032777  000100  000746          BIT     #TO11,@DIAG1            ;BACK IN TO11 STATE YET
002254' 001015                          BNE     4$                      ;BR IF YES
002256' 005304                          DEC     R4                      ;COUNT THE BREAK TIMER
002260' 001366                          BNE     3$                      ;BR IF NO TIMEOUT
002262' 017767  000734  175614          MOV     @DIAG1,ASTAT            ;SAVE THE ERROR INFO
002270' 010567  175604                  MOV     R5,CSRA
002274' 005067  175606                  CLR     ERRTYP
```

```
                                        ;*****************************************************************
002300' 104405  000000' 000000          HRDERS,BEGIN,NULL       ;FAILED TO GET BACK IN TO11 STATE
                                        ;*****************************************************************
002306' 000422                          BR      DT17                    ;GO TO NEXT TEST
002310' 000240                  4$:     NOP
002312' 017767  000710  175570          MOV     @STATUS,AWAS            ;GET THE STATUS REG.
002320' 032767  000200  175562          BIT     #TO11DN,AWAS            ;TO11 DONE BIT SET ?
002326' 001012                          BNE     DT17                    ;BR IF YES
002330' 010567  175544                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
002334' 012767  000110' 175542          MOV     #AWAS,WASADR
002342' 012767  000106' 175532          MOV     #ASB,SBADR
                                        ;*****************************************************************
002350' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************


                                ;CHECK THAT STST NULL FLOP CAN BE SET
                                ;-------------------------------------

002354' 012767  000040  175524  DT17:   MOV     #NULSTP,ASB             ;NULSTP SHOULD GET SET
002362' 004767  000402          JSR     PC,T11LOK               ;LOCK IN TO 11 STATE
002366' 005067  175514          CLR     ERRTYP
                                        ;*****************************************************************
002372' 104405  000000' 000000          HRDERS,BEGIN,NULL       ;FAILED TO LOCK IN TO 11 STATE
                                        ;*****************************************************************
002400' 000446                          BR      DT20                    ;GO TO NEXT TEST
002402' 012777  077777  000600          MOV     #ZSTOP!37777,@TO11BC    ;SET THE NULL STOP BIT
002410' 005077  000604                  CLR     @TO11DT                 ;CLEAR TO 11 DATA REG.
002414' 016777  000626  000572          MOV     NPRLOC,@TO11AD          ;LOAD THE TO11 ADDR REG.
002422' 012777  040010  000574          MOV     #10!EDONES,@DIAG2       ;LOAD DIAG2 REG
002430' 004767  000270                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002434' 012777  000014  000562          MOV     #14,@DIAG2              ;SET UP DIAG2
002442' 004767  000256                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002446' 017767  000554  175434          MOV     @STATUS,AWAS            ;GET THE STATUS REG.
002454' 042767  177737  175426          BIC     #^C<NULSTP>,AWAS        ;CLEAR OUT JUNK BITS
002462' 026767  175420  175420          CMP     ASB,AWAS                ;DID NULSTP SET ??
002470' 001412                          BEQ     DT20                    ;BR IF YES
002472' 010567  175402                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
002476' 012767  000110' 175400          MOV     #AWAS,WASADR
002504' 012767  000106' 175370          MOV     #ASB,SBADR
                                        ;*****************************************************************
002512' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************


                                ;CHECK ABC INCREMENTATION DURING TO 10 E-BUFF FILL
                                ;-------------------------------------------------

002516' 004767  000216          DT20:   JSR     PC,DIAGRT               ;GO DO A DIAGNOSTIC RESET
002522' 016767  000520  175356          MOV     NPRLOC,ASB              ;SAVE NPR ADDRESS
002530' 062767  000001  175350          ADD     #1,ASB                  ;INCREMENT IT
002536' 016777  000504  000446          MOV     NPRLOC,@TO10AD          ;LOAD THE TO 10 ADDR REG.
002544' 004767  000322                  JSR     PC,T10LOC               ;GO LOCK IN TO 10 STATE
002550' 005067  175332                  CLR     ERRTYP
                                        ;*****************************************************************
002554' 104405  000000' 000000          HRDERS,BEGIN,NULL       ;FAILED TO LOCK IN TO 10 STATE
                                        ;*****************************************************************
002562' 000441                          BR      DTEXIT                  ;EXIT TESTS
```



```
002564' 012777  000010  000432          MOV     #10,@DIAG2              ;SET UP TO SHIFT STATES
002572' 004767  000126                  JSR     PC,DIAG9P               ;GO PULSE THE CLOCK
002576' 012777  040000  000420          MOV     #EDONES,@DIAG2          ;SET EDONES BIT
002604' 000240                          NOP
002606' 012777  000014  000410          MOV     #14,@DIAG2              ;SHIFT STATES
002614' 004767  000104                  JSR     PC,DIAG9P               ;PULSE THE CLOCK TWICE
002620' 004767  000100                  JSR     PC,DIAG9P
002624' 017767  000362  175256          MOV     @TO10AD,AWAS            ;GET THE TO 10 ADDRESS
002632' 026767  175250  175250          CMP     ASB,AWAS                ;DID IT GET INCREMENTED ??
002640' 001412                          BEQ     DTEXIT                  ;BR IF YES
002642' 010567  175232                  MOV     R5,CSRA                 ;SAVE THE ERROR INFO
002646' 016767  000340  175230          MOV     TO10AD,WASADR
002654' 012767  000106' 175220          MOV     #ASB,SBADR
                                        ;*****************************************************************
002662' 104404  000000'                 DATERS,BEGIN            ;DATA ERROR!!!
                                        ;*****************************************************************


002666' 005767  175142          DTEXIT: TST     PASCNT                  ;FIRST TIME THROUGH ?
002672' 001405                          BEQ     1$                      ;BR IF YES-QUICK PASS
002674' 005767  000336                  TST     ICOUN                   ;DO IT AGAIN ??
002700' 001402                          BEQ     1$                      ;BR IF NO
002702' 000167  175440                  JMP     DOAGIN                  ;GO DO IT
002706' 000207                  1$:     RTS     PC                      ;GO TO NEXT DTE20


                                ;COMMON SUBROUTINES
                                ;-------------------

002710' 012777  000060  000304  DIAGPU: MOV     #PULSE!D1011,@DIAG1     ;PULSE THE CLOCK
002716' 000240                          NOP
002720' 000240                          NOP
002722' 000207                          RTS     PC

002724' 012777  002060  000270  DIAG9P: MOV     #DSO5!PULSE!D1011,@DIAG1
002732' 000240                          NOP
002734' 000240                          NOP
002736' 000207                          RTS     PC

002740' 012777  000100  000256  DIAGRT: MOV     #DRESET,@DIAG2          ;DIAGNOSTIC RESET
002746' 000240                          NOP
002750' 000240                          NOP
002752' 000207                          RTS     PC

002754' 012777  000040  000240  DIAGME: MOV     #D1011,@DIAG1           ;SET DIAGNOSTIC MODE
002762' 000240                          NOP
002764' 000240                          NOP
002766' 000207                          RTS     PC

002770' 004767  177744          T11LOK: JSR     PC,DIAGRT
002774' 004767  177754          T11LOC: JSR     PC,DIAGME
003000' 005004                  T11LKA: CLR     R4
003002' 032777  000100  000212  1$:     BIT     #TO11,@DIAG1            ;IN TO11 STATE YET ??
003010' 001017                          BNE     T11KB                   ;BR IF YES
003012' 004767  177672                  JSR     PC,DIAGPU               ;GO PULSE THE CLOCK
003016' 104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
```

```
003022' 104407  000000'          BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
003026' 005304                   DEC     R4              ;COUN THE TIMER
003030' 001364                   BNE     1$              ;BR IF NO TIMEOUT
003032' 016767  000164  175040   MOV     DIAG1,CSRA      ;SAVE THE ERROR INFO
003040' 017767  000156  175034   MOV     @DIAG1,ACSR
003046' 000207                   RTS     PC              ;ERROR RETURN
003050' 012777  002040  000144   T11KB:  MOV     #D1011!DS05,@DIAG1      ;SET STATE HOLD
003056' 062716  000006           ADD     #6,(SP)         ;MOVE PC AROUND ERROR CALL
003062' 000240                   NOP
003064' 000207                   RTS     PC              ;NOW RETURN

003066' 004767  177646           T10LOK: JSR     PC,DIAGRT       ;GO DO A DIAGNOSTIC RESET
003072' 004767  177656           T10LOC: JSR     PC,DIAGME       ;GO SET DIAG MODE
003076' 005004                   CLR     R4              ;INIT TIME OUT COUNTER
003100'                          1$:
003100' 104407  000000'          BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
003104' 104407  000000'          BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
003110' 032777  000200  000104   BIT     #T010,@DIAG1    ;SHIFT TO TO 10 STATE YET ?
003116' 001013                   BNE     T10LKB          ;BR IF YES
003120' 004767  177564           JSR     PC,DIAGPU       ;PULSE THE CLOCK
003124' 005304                   DEC     R4              ;COUNT THE TIMER
003126' 001364                   BNE     1$              ;BR IF NO TIMEOUT
003130' 016767  000066  174742   MOV     DIAG1,CSRA      ;SAVE THE ERROR INFO
003136' 017767  000060  174736   MOV     @DIAG1,ACSR
003144' 000207                   RTS     PC              ;ERROR RETURN
003146' 012777  002040  000046   T10LKB: MOV     #D1011!DS05,@DIAG1      ;LOCK IN TO 10 STATE
003154' 062716  000006           ADD     #6,(SP)         ;MOVE PC AROUND ERROR CALL
003160' 000240                   NOP
003162' 000207                   RTS     PC              ;NORMAL RETURN


003164' 005167  000050           DTINT:  COM     INTFLG          ;SET SOFTWARE FLAG
003170' 000002                   RTI

                                 ;CONSTANTS,ADDRESS TABLE, AND VARIABLES
                                 ;--------------------------------------

                                 ;THIS TABLE GETS LOADED WITH THE ADDRESSES OF THE DTE REGISTERS

003172' 000000                   DLYCNT: 0
003174' 000000                   DEXWD3: 0
003176' 000000                   DEXWD2: 0
003200' 000000                   DEXWD1: 0
003202' 000000                   TENAD1: 0
003204' 000000                   TENAD2: 0
003206' 000000                   T010BC: 0
003210' 000000                   T011BC: 0
003212' 000000                   T010AD: 0
003214' 000000                   T011AD: 0
003216' 000000                   T010DT: 0
003220' 000000                   T011DT: 0
003222' 000000                   DIAG1:  0
003224' 000000                   DIAG2:  0
003226' 000000                   STATUS: 0
003230' 000000                   DIAG3:  0
```



```
                                 ;VARIABLES AND FLAGS

003232' 000000                   TDVD1:  0               ;DEVICE SELECT BITS BUFFERS
003234' 000000                   TDVD2:  0
003236' 000000                   ICOUN:  0               ;PASS ITERATION COUNTER
003240' 000000                   INTFLG: 0               ;SOFTWARE INTERRUPT FLAG
003242' 000100                   ITCNT:  100             ;NO. OF TEST ITERATIONS FOR EA. DTE20
003244' 000000                   NPRBUF: 0               ;NPR XFER BUFFER

                                 ;CONSTANTS

003246' 003244'                  NPRLOC: NPRBUF          ;ADDRESS POINTER TO NPR BUFFER

          000001                 .END
```

```
ACSR     000102R          221#   415*   741*   760*
ADDR     000006R          187#   270
ADDR22=  001000           239#
ASB      000106R          225#   295*   303    311*   323    342*   347    350    365*   369    372    392*   397
                          409*   541*   539    542    551*   552*   571    574    585*   635    643*   659    663
                          672*   673*   689    693
ASTAT    000104R          223#   331*   417*   441*   460*   479*   498*   516*   608*   622*
AWAS     000110R          226#   299*   320*   346*   347    368*   369    393*   412*   415    538*   539    570*
                          571    630*   631    634    657*   658*   659    662    686*   689
BEGIN    000000R          184#   267    278    305    326    334    353    375    400    420    445    464    483
                          502    520    545    559    578    593    601    602    611    616    617    626    637
                          647    665    678    695    736    737    752    753
BIT0   = 000001           239#   410
BIT1   = 000002           239#   251
BIT10  = 002000           239#   244
BIT11  = 004000           239#   260
BIT12  = 010000           239#   253
BIT13  = 020000           239#   243    256
BIT14  = 040000           239#   248    264
BIT15  = 100000           239#   252    257
BIT2   = 000004           239#
BIT3   = 000010           239#
BIT4   = 000020           239#   245
BIT5   = 000040           239#   245    246    250    255
BIT6   = 000100           239#   249    259
BIT7   = 000200           239#   247    258    261
BIT8   = 000400           239#   254
BIT9   = 001000           239#
BREAK$= 104407            239#   601    602    616    617    736    737    752    753
BR1      000012R          189#   432
BR2      000013R          190#
BTOD$  = 104421           239#
CDATA$= 104412            239#
CONFIG   000056R          209#
CSRA     000100R          219#   301*   324*   330*   351*   373*   398*   416*   440*   459*   478*   497*   515*
                          543*   575*   607*   623*   633*   661*   691*   740*   759*
DATCK$= 104411            239#
DATER$= 104404            239#   305    326    353    375    400    545    578    637    665    695
DEXWD1   003200R          779#
DEXWD2   003176R          778#
DEXWD3   003174R          777#
DIAGME   002754R          297    314    319    344    361    367    386    725#   731    749
DIAGPU   002710R          710#   735    756
DIAGRT   002740R          296    313    318    343    360    366    385    408    429    442    451    461    470
                          480    489    499    517    530    533    553    576    586    588    671    720#   730
                          748
DIAG1    003222R          316    363    389    598*   603    608    618    622    710*   715*   725*   733    740
                          741    743*   754    759    760    762*   788#   683*   685*   720*   789#
DIAG2    003224R          564*   566*   568*   597*   653*   655*   681*
DIAG3    003230R          791#
DIAG9P   002724R          565    567    569    654    656    682    686    687    715#
DLYCNT   003172R          282    298*   299    302    312    336    345*   346    349    359    377    383    411*
                          412    776#
DOAGIN   000346R          290#   703
DON10S=  100000           257#   434    441
DON11S=  000200           258#   472    479
```



```
DRESET=  000100           249#   720
DS05   = 002000           244#   715    743    762
DTEMAX=  000004           242#   277
DTESIZ=  000040           241#   276    287
DTEXIT   002666R          680    690    699#
DTINT    003164R          431    768#
DT01     000352R          295#
DT02     000430R          300    311#
DT03     000562R          342#
DT04     000650R          348    359#
DT05     000760R          383#
DT06     001062R          408#
DT07     001160R          429#
DT10     001256R          439    451#
DT11     001342R          458    470#
DT12     001426R          477    489#
DT13     001512R          496    508#
DT14     001604R          530#
DT15     001706R          540    551#
DT16     002074R          561    572    585#
DT17     002354R          595    613    628    632    643#
DT20     002516R          649    660    671#
DVID1    000014R          191#   264
D1011  = 000040           246#   710    715    725    743    762
EDONE$=  040000           248#   564    653    683
ENDIT$=  104413           239#   278
END$   = 104410           239#   267
ERRTYP   000106R          224#   332*   418*   443*   462*   481*   500*   518*   557*   591*   609*   624*   645*
                          676*
ERR10S=  020000           256#   491    498
ERR11S=  000002           251#   509    516
EXIT$  = 104400           239#
GETPA$=  104415           239#
GOEXDT   000316R          275    282#
GWBUF$=  104414           239#
HRDCNT   000044R          204#
HRDER$=  104405           239#   334    420    445    464    483    502    520    559    593    611    626    647
                          678
HRDPAS   000050R          206#
ICONT    000036R          201#
ICOUN    000236R          289*   290*   701    797#
ICOUNT   000040R          202#
IDNUM    000122R          231#
INIT     000030R          198#
INTFLG   003240R          430*   438    452*   457    471*   476    490*   495    508*   513    768*   798#
INTR     000120R          230#
INTRON=  000040           250#   434    441    453    460    472    479    491    498    509    516
ITCNT    003242R          289*   799#
MAP22$=  104416           239#
MODNAM   000000R          185#
MODSP    000224R          199    237#
MSGN$  = 104403           239#
MSGS$  = 104402           239#
MSG$   = 104401           239#
NPRBUF   003244R          804
NPRLOC   003246R          531    534    551    554    563    589    652    672    674    804#
```

```
NULL   = 000000            239#    334     420     445     464     483     502     520     559     593     611     626     647
                           678
NULSTP= 000040             255#    643     658
OPEN   = 000000            186     192     193     194     195     212     213     214     215     216     217     218     219
                           221     223     225     226     228     229     230     239#
OTOAS  = 104420            239#
PASCNT  000034R            200#    699
PIRQS  = 000004            239#
POPSP  = 005726            239#
POPSP2= 022626             239#
PRTY   = 000000            239#
PRTY0  = 000000            190     239#
PRTY1  = 000040            239#
PRTY2  = 000100            239#
PRTY3  = 000140            239#
PRTY4  = 000200            189     239#
PRTY5  = 000240            239#
PRTY6  = 000300            239#
PRTY7  = 000340            239#
PS     = 177776            239#
PSW    = 177776            239#
PULSE  = 000060            245#    710     715
PUSH   = 005746            239#
PUSH2 = 024646             239#
RAMISO= 010000             253#    328     413
RAND$  = 104417            239#
RANNUM  000054R            208#
RESTRT  000246R            227     266     269#
RES1    000056R            210#
RES2    000060R            211#
RSTRT   000112R            227#
SBADR   000102R            220#    303*    323*    350*    372*    397*    542*    574*    635*    663*    693*
SOFCNT  000042R            203#
SOFERS= 104406             239#
SOFPAS  000046R            205#
SPOINT  000032R            199#
SPSIZ  = 000040              1#    232
SR1     000016R            192#
SR2     000020R            193#
SR3     000022R            194#
SR4     000024R            195#
START   000224R            198     264#
STAT    000026R            197#
STATUS  003226R            328     331     413     417     434*    453*    472*    491*    509*    630     657     790#
SVR0    000062R            212#
SVR1    000064R            213#
SVR2    000066R            214#
SVR3    000070R            215#
SVR4    000072R            216#
SVR5    000074R            217#
SVR6    000076R            218#
SYSCNT  000052R            207#
TDVD1   003232R            264*    265*    269     795#
TDVD2   003234R            269*    273*    796#
TENAD1  003202R            780#
TENAD2  003204R            781#
```



```
TO10   = 000200            247#    754
TO10AD  003212R            674*    688     692     784#
TO10BC  003206R            782#
TO10DB= 000400             254#
TO10DN= 100000             252#
TO10DT  003216R            786#
TO11   = 000100            259#    603     618     733
TO11AD  003214R            534*    538     541     563*    570     573     589*    652*    785#
TO11BC  003210R            532*    562*    587*    596*    650*    783#
TO11BM= 020000             243#    562
TO11DB= 004000             260#    453     460
TO11DN= 000200             261#    585     631
TO11DT  003220R            651*    787#
TRPDFD= 000022             239#
T10LKB  003146R            755     762#
T10LOC  003072R            675     749#
T10LOK  003066R            748#
T11KB   003050R            734     743#
T11LKA  003000R            732#
T11LOC  002774R            556     590     731#
T11LOK  002770R            644     730#
VECTOR  000010R            188#    271
WASADR  000104R            222#    302*    322*    349*    371*    396*    541*    573*    634*    662*    692*
WDFR    000116R            229#
WDTO    000114R            228#
XFLAG   000005R            186#
ZSTOP  = 040000            262#    650

. ABS.  000000      000
        003250      001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XDTAC0,XDTAC0/SOL/CRF:SYM=DDXCOM,XDTAC0
 RUN-TIME: 1 2 .4 SECONDS
 RUN-TIME RATIO: 16/5=3.2
 CORE USED:  7K  (13 PAGES)
```