DR11B

DIAGNOSTIC

MD-11-DZDRB-D

EP-DZDRB-D-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA

.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDRB-D

PRODUCT NAME: DR11B - PDP11 GENERAL NPR INTERFACE
DA11B - PDP11 INTERPROCESSER LINK

DATE CREATED: APRIL 1971

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: LARRY CONDON, DICK JONES, STEVE POMFRET

REVISED BY: W.F.KELLICKER 29-JAN-1974
ALAN BOSTICK 20-JUN-1976

1. ABSTRACT

1.1 THIS IS A LOGIC TEST OF THE "NPR GENERAL INTERFACE" -DR11B.
THERE IS A SPECIAL MAINTENANCE FEATURE THAT ALLOWS TESTING
OF NPRS WITHOUT A CUSTOMERS DEVICE ATTACHED.
1.2 THERE IS A SECOND TEST INCLUDED FOR EXERCISING THE DA11B
INTERPROCESSER LINK. THE DR11B TEST SHOULD BE RUN IN
IN EACH COMPUTER BEFOUR TESTING THE DA11B.

2. REQUIREMENTS

2.1 EQUIPMENT

2.1.1 FOR THE DR11B

PDP-11 STANDARD COMPUTER
DR11B
NOTE: WITH OR WITHOUT HARDWARE SWITCH REGISTER
2.1.2 FOR THE DA11B

2 PDP-11 STANDARD COMPUTERS
1 DA11B CONSISTING OF 2 M7229 MODULES AND 2
BC09R CABLES.
NOTE: WITH OR WITHOUT HARDWARE SWITCH REGISTER

2.2 STORAGE

2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY
FROM 0 TO 14000.

3. LORDING PROCEDURE

3.1 METHOD

PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED.

4. DR11B STARTING PROCEDURE

4.1 CONTROL SWITCH SETTING

STARTING AT SA 200 ALL SWITCHES SHOULD BE DOWN OR ZERO.

4.2 STARTING ADDRESS OR ADDRESSES

(A) 200 = TEST OF LOGIC USING MAINTENANCE FEATURE M9&8 IN 004.004

4.3 PROGRAM AND OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY.
LOAD STARTING ADDRESS
PRESS START.
THE PROGRAM WILL LOOP, BELL WILL RING ONCE
PER PASS OF THE PROGRAM.

NOTE: IF SOFTWARE SWITCH REGISTER IS SELECTED THEN THE
FOLLOWING WILL BE PRINTED:
        SWR= XXXXXX NEW=
(REFER TO SECTION 5.1 FOR OPERATOR OPTIONS)

5. DR11B OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

5.1.1 AT SA 200 .. THE INSTRUCTION AND LOGIC TEST.
WITH ALL SWITCHES DOWN THE PROGRAM WILL PRINT
OUT ON ERRORS AND CONTINUE IN TEST. (BELL
WILL RING AT COMPLETION OF A PASS)

5.1.2 SWITCH SETTINGS ARE

SW15 = 1 OR UP ... HALT ON ERROR
SW14 = 1 OR UP ... SCOPE LOOP
SW13 = 1 OR UP ... INHIBIT PRINTOUT
SW12 = 1 OR UP ... INHIBIT TRACE TRAP
SW11 = 1 OR UP ... INHIBIT ITERATIONS

5.1.3
IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH
REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS
THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER.
IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES
AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH
REGISTER (LOC. 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH
REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY
DOING THE FOLLOWING:

1) TYPE CONTROL G (↑G); THIS WILL ALLOW THE TTY TO ENTER DATA INTO
   LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.

2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS
   OF THE SOFTWARE SWITCH REGISTER.)

3) AFTER THE ''NEW='' HAS BEEN TYPED THE: THE OPERATOR CAN DO ONE
   OF THE FOLLOWING AT THE TTY:

   A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>.
      (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS
      WILL BE ALLOWED)

IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.

B) IF A CONTROL U <^U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

5.2. SUBROUTINE ABSTRACTS

BEGIN SA 200

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUBTEST IN THE INSTRUCTION SECTION. IT RECORDS THE STARTING ADDRESS OF EACH SUB-TEST AS IT IS BEING ENTERED. IF A SCOPE LOOP IS REQUESTED, IT WILL JUMP TO THE START OF THE SUBTEST THAT THE SCOPE LOOP IS REQUESTED FOR. IF SCOPE LOOP IS NOT REQUESTED, THERE WILL BE EITHER A FIXED OR RANDOM NUMBER OF ITERATIONS ON THAT SUBTEST BEFORE THE NEXT SUBTEST IS ENTERED. SWITCH 11 ON A 1 INHIBITS ITERATION OF SUBTESTS.
NOTE: SUPPORTS ^G ROUTINE FOR DYNAMIC LOADING OF SOFTWARE SWITCH REGISTER

5.2.2 HALT

IS A ROUTINE THAT PRINTS-OUT AN ADDRESS THAT TAGS THE FAILING SUBTEST, THE CP STATUS REGISTER AND THE DR11B STATUS REGISTER AT THE TIME OF FAILURE.
NOTE: SUPPORTS ^G ROUTINE FOR DYNAMIC LOADING OF SOFTWARE SWITCH REGISTER

5.2.3 LODBUF

THE INBUF BUFFER IS LOADED WITH AN INCREMENTING PATTERN (0,1,2,3,...) BEGINNING AT THE STARTING ADDRESS OF INBUF. THE NUMBER OF WORDS LOADED IS DETERMINED BY THE CONTENTS OF BUFLEN.

5.2.4 CHKBFF

THE CHKBUF BUFFER IS LOADED WITH A MODIFIED INCREMENTING PATTERN (0,0,2,2,4,4,6,6,...) BEGINNING AT THE STARTING ADDRESS OF CHKBUF. THE NUMBER OF WORDS LOADED IS DETERMINED BY THE CONTENTS OF BUFLEN. THIS BUFFER IS LOADED ONLY FOR TESTS WHICH USE THE MAINTENANCE MODE OF THE DR11-B WHICH HAS A SPECIAL ALTERNATING DATI-DATO SEQUENCE OF OPERATION.

5.2.5 INTA

THE IE BIT IS CLEARED IN THE DRST THEN THE DRST IS CHECKED FOR THE ABSENCE OF AN ERROR AND THE PRESENCE OF READY. THE DRWC IS CHECKED TO SEE THAT IT IS EQUAL TO ZERO. THE CORRECT CONTENTS

OF THE DRBA ARE CALCULATED AND CHECKED. THERE IS A JSR TO THE NORMAL SUB-ROUTINE BEFORE THIS ROUTINE IS EXITED. THE PROGRAM WILL HALT IF ERROR IS SET, READY IS CLEAR, OR READY AND ERROR ARE CLEAR.

5.2.6 DATCHK

THIS ROUTINE IS ENTERE. CHECK INBUF AFTER A MAINTENANCE MODE OPERATION. THE CONTENTS OF INBUF AND THE CONTENTS OF CHKBUF ARE CHECKED TO SEE THAT THEY ARE THE SAME. THE NUMBER OF COMPARISONS MADE IS DETERMINED BY THE CONTENTS OF BUFLEN.

5.2.7 NORMAL

THE ROUTINE IS ENTERED FROM INTA AND FROM SOME TESTS WHICH DON'T USE INTA. THE NUMBER OF THE DRINV+2 IS PUT INTO DRINV AND THE DRVS IS CLEARED. IF THE DR11-B INTERRUPTS UNDER THESE CONDITIONS THE PDP-11 WILL HALT AT DRVS. THE PROCESSOR STATUS WORD IS RESTORED TO LEVEL 7 AND THE ROUTINE IS EXITED.

5.2.8 DATOCK

AFTER A STRING OF DATO'S HAS BEEN COMPLETED THIS ROUTINE CHECKS THAT THE CORRECT DATA PATTERN (52525) WAS TRANSFERRED TO INBUF. THE NUMBER OF COMPARISONS MADE IS DETERMINED BY THE CONTENTS OF BUFLEN. AN ADDITIONAL CHECK IS MADE ON BUFLEN+2 TO INSURE THAT TOO MANY WORDS WEREN'T TRANSFERRED.

5.2.9 ERRCHK

THIS ROUTINE CLEARS IE AND HALTS IF ERROR IS SET.

5.2.10 TRAPCATCHER

THIS IS A SERIES OF INSTRUCTIONS STARTING AT LOCATION 0 DESIGNED TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS TO THE TRAP AND INTERRUPT VECTOR AREA OF MEMORY.

EACH VECTOR ENTRANCE ADDRESS IS LOADED WITH THE ADDRESS OF THE NEXT LOCATION. THE NEXT LOCATION IS LOADED WITH A HALT (000000). THUS AN ILLEGAL TRAP OR INTERRUPT WILL CAUSE A HALT AT THE TRAP LOCATION PLUS TWO.

IF A HALT OCCURS IN THE TRAP OR INTERRUPT AREA, EXAMINE REGISTER SIX, IT WILL CONTAIN THE CURRENT STACK ADDRESS. THE CONTENTS OF THE CURRENT STACK ADDRESS IS THE VALUE OF THE LOCATION COUNTER WHEN THE TRAP OR INTERRUPT OCCURRED.

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 LOADING AND STARTING AT 200 ITH ALL SWITCHES DOWN IS THE INSTRUCTION AND LOGIC TEST. IF AN EPROR IS DETECTED HERE, THERE WILL BE A PRINTOUT. WHEN AN ERROR IS DETECTED AND IT IS NECESSARY TO SCOPE ON IT, PLACE SW15 UP TO HALT ON ERROR, THEN SW14 UP TO LOOP ON ERROR, THEN SW13 UP TO DELETE PRINTOUTS.

6. DR11B ERRORS

6.1 ERROR PRINTOUT

THE PC OF THE FAILING TEST AND THE CP STATUS WILL BE PRINTED.

6.2 ERROR RECOVERY

DEPRESS CONTINUE TO RESTART SECTION

7. DR11B RESTRICTIONS

7.1 STARTING RESTRICTION

NONE

7.2 OPERATIONAL RESTRICTION

M968 MUST BE IN SLOTS C04/D04 - FOR DIAGNOSTIC TESTING
SHOULD BE IN A02/B02 FOR NORMAL USER OPERATION.

9. MISCELLANEOUS

9.1 EXECUTION TIME

ABOUT 2 MINUTES

9. PROGRAM DESCRIPTION

THE FOLLOWING IS A GENERAL LIST OF FUNCTIONS TESTED.

CAN ALL REG BE ADDRESSED WITHOUT ERROR
RESET CLEAR DRWC
RESET CLEAR DRBA
CAN ALL DRWC BITS BE SET
CAN 15-1 IN DRBA BE SET
FNCT1 SET & CLEARED
FNCT2 SET & CLEARED
FNCT3 SET & CLEARED
XBA16 SET & CLEARED
XBA17 SET & CLEARED
IE SET & CLEARED
CYCLE SET & CLEARED

MAINT SET & CLEARED
ALL DRST R/W BITS CAN BE SET & CLEARED
ALL DRST R/W SET, RESET TO 0, RDY IS SET, NEX IS CLEARED, GO IS 0
DRWC HOLD ALT. 0'S & 1'S AND ALT. 1'S & 0'S
DRBA HOLD ALT. 0'S & 1'S AND ALT. 1'S & 0'S
INC PATTERN TO WRAP-AROUND IN DRWC
INC PATTERN TO WRAP-AROUND IN DRBA
NO INT. AT LEVEL 7
NO INT. AT LEVEL 6
NO INT. AT LEVEL 5
DOES INT. AT LEVEL 4
NO MAINTBRD A02/B02
MAINTBRD C04/D04
FNCT BITS CONTROL DSTAT BITS
RESET 0'S DRDB
ALL DRDB BITS CAN BE SET
DRDB HOLD ALT. 1'S & 0'S AND ALT. 0'S & 1'S
INC TO WRAP-AROUND IN DRDB
RESET SETS ONLY RDY IN DRST
BADD READS AS 0
1 DATI NPR
1 DATO NPR
BAOF FORCES ERROR & IS CLEARED BY CLEARING DRBA OR RESET
GO CLEARS RDY
DATO TO DIODE MEM CAUSES NEX
DO WITHOUT CLEARING PREVIOUS ERROR CAUSES ANOTHER INT.
10 DATI'S (BURST)
10 DATO'S (BURST)
200 DATI'S (BURST)
200 DATO'S (BURST)
200 DATI'S (NON-BURST)
200 DATO'S (NON-BURST)
FUNCT BITS INC WITH MAINT MODE XFERS
10 MAINT MODE XFERS
200 MAINT MODE XFERS

10. LISTING

11. FLOW CHART(S)

12. DA11B STARTING PROCEDURE

12.1 THERE ARE TWO STARTING LOCATIONS; ONE FOR THE COMUPTER THAT WILL BE THE SLAVE AND ANOTHER FOR THE COMPUTER THAT WILL BE THE MASTER.

12.2 SLAVE COMPUTER, LOAD ADDRESS 1002 WITH THE HALT KEY DOWN AND PRESS START.

12.3 MASTER COMPUTER, LOAD ADDRESS 1000 WITH THE HALT KEY DOWN AND PRESS START.

I01

12.4 SLAVE COMPUTER. PUT THE HALT KEY UP AND PRESS CONTINUE.

12.5 MASTER COMPUTER,PUT THE HALT KEY PU AND PRESS CONTINUE.

13. DA11B OPERATING PROCEDURE

13.1 THE PROGRAM WILL LOOP AFTER STARTING AND PRINT OUT ANY ERRORS. THE PROGRAM WILL HALT AFTER NON RECOVERABLE ERRORS.

13.2 THE MAINT MODULE MUST BE IN A02/B02 AND THE THE DA11B MUST BE IN C04/D04 .

14. DA11B PROGRAM DISCRIPTION

14.1 THE SLAVE COMPLTER STARTS BY ENTERING A BACKGROUND TO WAIT FOR AN INTERJPT WITH THE INTERUPT ENABLED. THE FIRST INTERUPT THAT COMES SHOULD BE THE READY INTERUPT SET UP WHEN THE MASTER HIT THE START KEY. THE INTERUPT CAUSES THE SLAVE TO ENTER THE INTERUPT SERVICE ROUTINE.

14.2 THE INTERUPT SERVICE ROUTINE DETERMINS WHAT INTERUPT CAME UP AND IF IT SHOULD HAVE COME UP. IF THE INTERUPT WAS THE ONE EXPECTED THAN THE THAN THE PROGRAM GOES TO THE TO THE PROPER JOB ROUTINE, SJOBXX FOR SLAVE SERVICE AND JOBXX FOR THE MASTER ROUTINE.

14.3 THE NEXT THING THAT SHOULD HAPPEN IS THE MASTER SHOULD ISSUE AN INTERUPT TO THE SLAVE . THIS IS A SIGNAL FOR THE SLAVE TO ACCEPT THE WORD COUNT,OFFSET,AND TWO CHECK SUM WORDS. THE SLAVE ACCEPTS A WORD AT A TIME FROM THE DATA BUFFER EACH TIME THE MASTER TOGGLES FUNCTION BIT 3. EACH TIME IT READS A WORD THE SLAVE SENDS THE WORD BACK TO THE MASTER FOR VERIFICATION.

14.4 AFTER THE SLAVE HAS RECIEVED ALL THE PARAMETERS IT SETS ITS DIRECTION BIT TO THE OPPOSIT DIRECTION AS THE MASTER AND STARTS THE NPR TRANSFER.

14.5 THE MASTER SETS UP THE TYPE OF TRANSFERS AND CHECKS THE DATA WHEN IT COMES BACK FROM THE SLAVE.

15. DA11B ERRORS

15.1 THE PC OF THE FAILING TEST, THE CP STATUS AND THE DR11B STATUS REGISTER WILL BE PRINTED AFTER AN ERROR.

15.2 THERE IS NO ERROR RECOVERY FOR THE DA11B TEST BECAUSE THE OTHER COMPUTER WILL GET OUT OF SINC WHEN AN ERROR OCCURS.

```
                                %
                                        .ABS
                                        .MCALL .HEADER,STARS
                                ;;*****************************************************************
                                ;;*****************************************************************
                                .TITLE  MAINDEC-11-DZDRB-D
                                ;*COPYRIGHT (C) 1971,1976
                                ;*DIGITAL EQUIPMENT CORP.
                                ;*MAYNARD, MASS. 01754
                                ;*
                                ;*PROGRAM BY POMFRET,JONES,CONDON
                                ;*
                                ;*THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
                                ;*PACKAGE (MAINDEC-11-DZQAC-C2), SEPT 14, 1976.
                                ;*
000000:                         $TN=1
160000                          $SWR=160000     ;;HALT ON ERROR, LOOP ON TEST, INHIBIT ERROR TYPOUT
                                ;;*****************************************************************
                                ;       REVISED BY ALAN BOSTICK JUNE 1976
                                ;       MODIFIED FOR SOFTWARE SWITCH REGISTER
                                ;       INCLUDING DYNAMIC LOADING OF SWR
                                ;;*****************************************************************
                                ;;*****************************************************************

000240                                  NOP=000240
104400                                  SCOPE=TRAP
104000                                  HLT=EMT
000001                                  BIT0=000001
000002                                  BIT1=000002
000004                                  BIT2=000004
000010                                  BIT3=000010
000020                                  BIT4=000020
000040                                  BIT5=000040
000100                                  BIT6=000100
000200                                  BIT7=000200
000400                                  BIT8=000400
001000                                  BIT9=001000
002000                                  BIT10=002000
004000                                  BIT11=004000
010000                                  BIT12=010000
020000                                  BIT13=020000
040000                                  BIT14=040000
100000                                  BIT15=100000

000004                                  BUSERR=000004

                                ;LOAD TRAP CATCHER INTO 0 THRU 777.

000000                                  .=0
000030                                  .=30
000030  011072                          PRINT
000032  000340                          340
```

```
                    000034                          .=34
000034  011650                                      $COPEC
000036  000340                                      340
                    000176                          .=176
000176  000000                                      SWREG:  0
                    000200                          .=200
000200  012706  013566                              MOV     #BUFF,%6        ;SET UP STACK LIMIT
000204  005077  000604                              CLR     @PSW
000210  005002                                      CLR     %2
000212  012702  012433                              MOV     #$HEAD,%2
000216  004767  012240                              JSR     %7,TTOUT
000222  012702  012403                              MOV     #$MAIN,%2
000226  004767  012230                              JSR     %7,TTOUT
000232  000167  000634                              JMP     SUSWR
                    001000                          .=1000


                                                ;;**************************************************************
                                                ;       START OF BACK-TO-BACK DR11-B
                                                ;;**************************************************************

001000  000402                                  MSTART: BR      MSX             ;MASTER START
001002  000167  006276                          SSTART: JMP     SSI             ;SLAVE START
001006  000167  006212                          MSX:    JMP     MSI
001012  177570                                  SR:     177570
001014  177776                                  PSW:    177776
001016  172410                                  DRWC:   172410
001020  172412                                  DRBA:   172412
001022  172414                                  DRST:   172414
001024  172416                                  DRDB:   172416
001026  000126                                  DRVS:   126
001030  000240                                  DRINL:  240
001032  000124                                  DRINV:  124
001034  052525                                  NPR1:   52525
001036  173000                                  DIOMEM: 173000
001040  013570                                  INBUF:  XINBUF
001042  014572                                  CHKBUF: XCHKBU
001044  000000                                  BUFLEN: HALT
001046  000000                                  LENCHK: HALT
001050  000000                                  BRWAIT: HALT
001052  000000                                  WCLEN:  HALT
001054  000000                                  RDYCHK: HALT
001056  177560                                  TKS:    177560
001060  177562                                  TKB:    177562
001062  177564                                  TPS:    177564
001064  177566                                  TPB:    177566
001066  000000                                  FNCCNT: HALT
001070  000000                                  INBUF1: HALT

001072  013746  000006                          SUSWR:  MOV     @#6,-(SP)       ;SAVE VECTORS
001076  013746  000004                                  MOV     @#4,-(SP)
001102  012737  001122  000004                          MOV     #64$,@#4        ;SET UP FOR TIMEOUT
001110  022777  177777  177574                          CMP     #-1,@SR         ;REFERENCE HARDWARE SWITCH REGISTER
001116  001402                                          BEQ     65$
001120  000404                                          BR      66$
```

```
                                        64$:    CMP     (SP)+,(SP)+     ;ADJUST STACK
                                        65$:    MOV     #SWREG,SR       ;POINT TO SOFTWARE SWITCH REG
                                        66$:    MOV     (SP)+,@#4       ;RESTORE VECTORS
                                                MOV     (SP)+,@#6
                                                CMP     #SWREG,SR       ;IS S.WREG USED
                                                BNE     BEGIN
                                                JSR     PC,CNTLU        ;ALLOW SWREG TO BE LOADED
                                        BEGIN:  MOV     #340,@#PSW      ;PROC. AT LEVEL #7
                                                MOV     #BEGIN,RETURN

                                ;;**************************************************************
                                ;       TEST 0 CAN ALL DR11-B REG BE ADDRESSED WITHOUT ERROR?
                                ;;**************************************************************

                                                SCOPE
                                                MOV     #ERRA,@BUSERR   ;BUS ERROR VECTOR TO ERRA
                                                MOV     %7,%0           ;PC TO R0
                                                INC     @DRWC           ;ADDRESS DRWC
                                                MOV     %7,%0           ;PC TO R0
                                                INC     @DRBA           ;ADDRESS DRBA
                                                MOV     %7,%0           ;PC TO R0
                                                CLR     @DRST           ;ADDRESS DRST
                                                MOV     %7,%0           ;PC TO R0
                                                INC     @DRDB           ;ADDRESS DRDB
                                                BR      .+4             ;MADE IT - BRANCH OVER HALT
                                        ERRA:   HLT                     ;BUS ERROR, R0 HAS PC OF ERROR
                                                MOV     #6,@BUSERR      ;RESTORE #6 TO BUS ERROR VECTOR
                                                SCOPE


                                ;;**************************************************************
                                ;       TEST 1 DOES RESET CLEAR DRWC?
                                ;;**************************************************************


                                                MOV     #10,ICOUNT
                                                MOV     #-1,@DRWC       ;ALL ONES TO DRWC
                                                JSR     %7,CKSWR
                                                RESET                   ;INIT
                                                TST     @DRWC           ;LOOKING FOR Z-BIT TO SET
                                                BEQ     .+4             ;DID DRWC GET CLEARED?
                                                HLT                     ;DRWC NOT CLEAR
                                                SCOPE


                                ;;**************************************************************
                                ;       TEST2 DOES RESET CLEAR DRBA?
                                ;;**************************************************************


                                                SCOPE
                                                MOV     #-1,@DRBA       ;ALL ONES TO DRBA
                                                JSR     %7,CKSWR
```

```
001316  000005                          RESET                   ;INIT
001320  005777  177474                  TST     @DRBA           ;LOOKING FOR Z-BIT TO SET
001324  001401                          BEQ     .+4             ;DID DRBA GET CLEARED?
001326  104000                          HLT                     ;DRBA NOT CLEAR


                                ;;**************************************************************
                                ;       TEST3 CAN ALL DRWC BITS BE SET?
                                ;;**************************************************************

001330  104400                          SCOPE
001332  012767  004000  010370          MOV     #4000,ICOUNT
001340  012777  177777  177450          MOV     #-1,@DRWC       ;SET ALL BITS IN DRWC
001346  022777  177777  177442          CMP     #-1,@DRWC       ;LOOKING FOR Z-BIT TO SET
001354  001401                          BEQ     .+4             ;SEE IF ALL BITS GOT SET
001356  104000                          HLT                     ;ALL BITS AREN'T SET


                                ;;**************************************************************
                                ;       TEST4 CAN BITS 15-01 IN DRBA BE SET?
                                ;;**************************************************************

001360  104400                          SCOPE
001362  012777  177776  177430          MOV     #-2,@DRBA       ;SET BITS 15-01 IN DRBA
001370  022777  177776  177422          CMP     #-2,@DRBA       ;LOOKING FOR Z-BIT TO SET
001376  001401                          BEQ     .+4             ;SEE IF BITS 15-01 GOT SET
001400  104000                          HLT                     ;BITS 15-01 AREN'T SET


                                ;;**************************************************************
                                ;       TEST6 TEST THAT FNCT1 CAN BE SET AND CLEARED
                                ;;**************************************************************

001402  104400                          SCOPE
001404  052777  000002  177410          BIS     #BIT1,@DRST     ;SET FNCT1
001412  032777  000002  177402          BIT     #BIT1,@DRST     ;TEST FNCT1
001420  001001                          BNE     .+4             ;IS IT SET?
001422  104000                          HLT                     ;FNCT1 IS CLEAR
001424  042777  000002  177370          BIC     #BIT1,@DRST     ;CLEAR FNCT1
001432  032777  000002  177362          BIT     #BIT1,@DRST     ;TEST FNCT1
001440  001401                          BEQ     .+4             ;WAS IT CLEAR
001442  104000                          HLT                     ;FNCT1 WAS SET


                                ;;**************************************************************
                                ;       TEST7 TEST THAT FNCT2 CAN BE SET AND CLEARED
                                ;;**************************************************************

001444  104400                          SCOPE
001446  052777  000004  177346          BIS     #BIT2,@DRST     ;SET FNCT2
001454  032777  000004  177340          BIT     #BIT2,@DRST     ;TEST FNCT2
```

```
 658  001462  001001                        BNE     .+4             ;IS IT SET?
 659  001464  104000                        HLT                     ;FNCT2 IS CLEAR
 660  001466  042777  000004  177326        BIC     #BIT2,@DRST     ;CLEAR FNCT2
 661  001474  032777  000004  177320        BIT     #BIT2,@DRST     ;TEST FNCT2
 662  001502  001401                        BEQ     .+4             ;WAS IT CLEAR?
 663  001504  104000                        HLT                     ;FNCT2 WAS SET

                                ;;**************************************************************
                                ;       TEST10 TEST THAT FNCT3 CAN BE SET AND CLEARED
                                ;;**************************************************************

 664  001506  104400                        SCOPE
 665  001510  052777  000010  177304        BIS     #BIT3,@DRST     ;SET FNCT3
 666  001516  032777  000010  177276        BIT     #BIT3,@DRST     ;TEST FNCT3
 667  001524  001001                        BNE     .+4             ;IS IT SET?
 668  001526  104000                        HLT                     ;FNCT3 IS CLEAR
 669  001530  042777  000010  177264        BIC     #BIT3,@DRST     ;CLEAR FNCT3
 670  001536  032777  000010  177256        BIT     #BIT3,@DRST     ;TEST FNCT3
 671  001544  001401                        BEQ     .+4             ;WAS IT CLEAR?
 672  001546  104000                        HLT                     ;FNCT3 WAS SET

                                ;;**************************************************************
                                ;       TEST11 TEST THAT XBA16 CAN BE SET AND CLEARED
                                ;;**************************************************************

 673  001550  104400                        SCOPE
 674  001552  052777  000020  177242        BIS     #BIT4,@DRST     ;SET XBA16
 675  001560  032777  000020  177234        BIT     #BIT4,@DRST     ;TEST XBA16
 676  001566  001001                        BNE     .+4             ;IS IT SET?
 677  001570  104000                        HLT                     ;XBA16 IS CLEAR
 678  001572  042777  000020  177222        BIC     #BIT4,@DRST     ;CLEAR XBA16
 679  001600  032777  000020  177214        BIT     #BIT4,@DRST     ;TEST XBA16
 680  001606  001401                        BEQ     .+4             ;IS IT CLEAR
 681  001610  104000                        HLT                     ;XBA16 WAS SET

                                ;;**************************************************************
                                ;       TEST12 TEST THAT XBA17 CAN BE SET AND CLEARED
                                ;;**************************************************************

 682  001612  104400                        SCOPE
 683  001614  052777  000040  177200        BIS     #BIT5,@DRST     ;SET XBA17
 684  001622  032777  000040  177172        BIT     #BIT5,@DRST     ;TEST XBA17
 685  001630  001001                        BNE     .+4             ;IS IT SET?
 686  001632  104000                        HLT                     ;XBA17 IS CLEAR
 687  001634  042777  000020  177160        BIC     #BIT4,@DRST     ;CLEAR XBA17
 688  001642  032777  000020  177152        BIT     #BIT4,@DRST     ;TEST XBA17
 689  001650  001401                        BEQ     .+4             ;IS IT CLEAR?
 690  001652  104000                        HLT                     ;XBA17 WAS SET

                                ;;**************************************************************
                                ;       TEST13 TEST THAT IE CAN BE SET AND CLEARED
                                ;;**************************************************************

 691  001654  104400                        SCOPE
 692  001656  052777  000100  177136        BIS     #BIT6,@DRST     ;SET IE
 693  001664  032777  000100  177130        BIT     #BIT6,@DRST     ;TEST IE
```

```
        001672  001001                          BNE     .+4             ;IS IT SET?
        001674  104000                          HLT                     ;IE IS CLEAR
        001676  042777  000100  177116          BIC     #BIT6,@DRST     ;CLEAR IE
        001704  032777  000100  177110          BIT     #BIT6,@DRST     ;TEST IE
        001712  001401                          BEQ     .+4             ;IS IT CLEAR?
        001714  104000                          HLT                     ;IE WAS SET

                                        ;;*************************************************************
                                        ;       TEST14 TEST THAT CYCLE CAN BE SET AND CLEARED
                                        ;;*************************************************************
        001716  104400                          SCOPE
        001720  052777  000400  177074          BIS     #BIT8,@DRST     ;SET CYCLE
        001726  032777  000400  177066          BIT     #BIT8,@DRST     ;TEST CYCLE
        001734  001001                          BNE     .+4             ;IS IT SET?
        001736  104000                          HLT                     ;CYCLE WAS CLEAR
        001740  042777  000400  177054          BIC     #BIT8,@DRST     ;CLEAR CYCLE
        001746  032777  000400  177046          BIT     #BIT8,@DRST     ;TEST CYCLE
        001754  001401                          BEQ     .+4             ;IS IT CLEAR?
        001756  104000                          HLT                     ;CYCLE WAS SET

                                        ;;*************************************************************
                                        ;       TEST15 TEST THAT MAINT CAN BE SET AND CLEARED
                                        ;;*************************************************************
        001760  104400                          SCOPE
        001762  052777  010000  177032          BIS     #BIT12,@DRST    ;SET MAINT
        001770  032777  010000  177024          BIT     #BIT12,@DRST    ;TEST MAINT
        001776  001001                          BNE     .+4             ;IS IT SET?
        002000  104000                          HLT                     ;MAINT WAS CLEAR
        002002  042777  010000  177012          BIC     #BIT12,@DRST    ;CLEAR MAINT
        002010  032777  010000  177004          BIT     #BIT12,@DRST    ;TEST MAINT
        002016  001401                          BEQ     .+4             ;IS MAINT CLEAR?
        002020  104000                          HLT                     ;MAINT WAS SET

                                        ;;*************************************************************
                                        ;       TEST 16 TEST THAT ALL DRST R/W BITS CAN BE SET AND CLEARED
                                        ;;*************************************************************
        002022  104400                          SCOPE
        002024  052777  010576  176770          BIS     #10576,@DRST    ;SET FOLLOWING: MAINT(12), CYCLE(08),IE(06),XBA17(05),
                                                                        ;               XBA16(04),FNCT3(03),FNCT2(02),FNCT1(01)
        002032  032777  000002  176762          BIT     #BIT1,@DRST     ;TEST FNCT1
        002040  001001                          BNE     .+4             ;IS IT SET?
        002042  104000                          HLT                     ;FNCT1 IS CLEAR
        002044  032777  000004  176750          BIT     #BIT2,@DRST     ;TEST FNCT2
        002052  001001                          BNE     .+4             ;IS IT SET?
        002054  104000                          HLT                     ;FNCT2 IS CLEAR
        002056  032777  000010  176736          BIT     #BIT3,@DRST     ;TEST FNCT3
        002064  001001                          BNE     .+4             ;IS IT SET?
        002066  104000                          HLT                     ;FNCT3 IS CLEAR
        002070  032777  000020  176724          BIT     #BIT4,@DRST     ;TEST XBA16
        002076  001001                          BNE     .+4             ;IS IT SET?
        002100  104000                          HLT                     ;XBA16 IS CLEAR
        002102  032777  000040  176712          BIT     #BIT5,@DRST     ;TEST XBA17
        002110  001001                          BNE     .+4             ;IS IT SET?
        002112  104000                          HLT                     ;XBA17 IS CLEAR
        002114  032777  000100  176700          BIT     #BIT6,@DRST     ;TEST IE
        002122  001001                          BNE     .+4             ;IS IT SET?
```

```
002312  104000                  HLT                     ;IE IS CLEAR
002314  032777  000400  176666  BIT     #BIT8,@DRST     ;TEST CYCLE
002322  001001                  BNE     .+4             ;IS CYCLE SET?
002324  104000                  HLT                     ;CYCLE IS CLEAR
002326  032777  010000  176654  BIT     #BIT12,@DRST    ;TEST MAINT
002334  001001                  BNE     .+4             ;IS MAINT SET?
002336  104000                  HLT                     ;MAINT IS CLEAR
002340  042777  010576  176642  BIC     #10576,@DRST    ;CLEAR ALL R/W BITS IN DRST
002346  032777  000002  176634  BIT     #BIT1,@DRST     ;TEST FNCT1
002354  001401                  BEQ     .+4             ;IS FNCT1 CLEAR?
002356  104000                  HLT                     ;FNCT1 IS SET
002360  032777  000004  176622  BIT     #BIT2,@DRST     ;TEST FNCT2
002366  001401                  BEQ     .+4             ;IS FNCT2 CLEAR?
002370  104000                  HLT                     ;FNCT2 IS SET
002372  032777  000010  176610  BIT     #BIT3,@DRST     ;TEST FNCT3
002400  001401                  BEQ     .+4             ;IS FNCT3 CLEAR?
002402  104000                  HLT                     ;FNCT3 IS SET
002404  032777  000020  176576  BIT     #BIT4,@DRST     ;TEST XBA16
002412  001401                  BEQ     .+4             ;IS XBA16 CLEAR
002414  104000                  HLT                     ;XBA16 IS SET
002416  032777  000040  176564  BIT     #BIT5,@DRST     ;TEST XBA17
002424  001401                  BEQ     .+4             ;IS XBA17 CLEAR?
002426  104000                  HLT                     ;XBA17 IS SET

002430  032777  000100  176552  BIT     #BIT6,@DRST     ;TEST IE
002436  001401                  BEQ     .+4             ;IS IE CLEAR?
002440  104000                  HLT                     ;IE IS SET
002442  032777  000400  176540  BIT     #BIT8,@DRST     ;TEST CYCLE
002450  001401                  BEQ     .+4             ;IS CYCLE CLEAR?
002452  104000                  HLT                     ;CYCLE IS SET
002454  032777  010000  176526  BIT     #BIT12,@DRST    ;TEST MAINT
002462  001401                  BEQ     .+4             ;IS MAINT CLEAR?
002464  104000                  HLT                     ;MAINT IS SET

                        ;;**********************************************************************
                        ;       TEST17 ALL R/W BITS IN DRST CAN BE SET AND RESET TO ZERO, THAT READY
                        ;       IS SET, NEX IS CLEAR, AND GO IS READ AS A 0.
                        ;;**********************************************************************
002466  104400                  SCOPE
002470  012767  000010  007420  MOV     #10,ICOUNT
002476  052777  010576  176504  BIS     #10576,@DRST    ;SET FOLLOWING: MAINT(12),CYCLE(08),IE(06),XBA17(05),
                                                        ;               XBA16(04),FNCT3(03),FNCT2(02),FNCT1(01)
002504  017701  176500          MOV     @DRST,R1        ;MOVE (DRST) TO R1
002510  052701  167201          BIS     #167201,R1      ;SETS BITS IN R1 THAT WERE NOT SET IN DRST
002514  005201                  INC     R1              ;R1 SHOULD GO FROM -1 TO ZERO
002516  001401                  BEQ     .+4             ;WERE ALL DRST R/W BITS SET?
002520  104000                  HLT                     ;NOT ALL BITS WERE SET
002522  004767  007412          JSR     R7,CKSWR
002526  000005                  RESET                   ;CLEAR ALL DRST R/W BITS
002530  017701  176454          MOV     @DRST,R1        ;MOVE (DRST) TO R1
002534  042701  127200          BIC     #127200,R1      ;CLEAR ALL BITS EXCEPT R/W BITS, NEX, AND GO
002540  001401                  BEQ     .+4             ;SHOULD EQUAL ZERO
002542  104000                  HLT                     ;RESET DIDN'T LEAVE DRST AS IT SHOULD HAVE

                        ;;**********************************************************************
                        ;       TEST20 CAN DRWC HOLD ALTERNATE ONE'S AND ZERO'S
```

```
                                        ;;***************************************************************
        002356  104400                          SCOPE
        002360  012767  004000  007342          MOV     #4000,ICOUNT
        002366  012777  052525  176422          MOV     #052525,@DRWC   ;ALT 0'S AND 1'S TO DRWC
        002374  022777  052525  176414          CMP     #052525,@DRWC   ;LOOKING FOR Z-BIT TO SET
        002402  001401                          BEQ     .+4             ;DOES DRWC HAVE THE CORRECT PATTERN?
        002404  104000                          HLT                     ;DRWC DOESN'T HAVE THE CORRECT PATTERN
        002406  012777  125252  176402          MOV     #125252,@DRWC   ;ALT 1'S AND 0'S TO DRWC
        002414  022777  125252  176374          CMP     #125252,@DRWC   ;LOOKING FOR Z-BIT TO SET
        002422  001401                          BEQ     .+4             ;DOES DRWC HAVE THE CORRECT PATTERN?
        002424  104000                          HLT                     ;DRWC DOESN'T HAVE THE CORRECT PATTERN

                                        ;;***************************************************************
                                        ;       TEST21 CAN DRBA HOLD ALTERNATE ONE'S AND ZERO'S
                                        ;;***************************************************************
        002426  104400                          SCOPE
        002430  012777  052524  176362          MOV     #052524,@DRBA   ;ALT 0'S AND 1'S TO DRBA
        002436  022777  052524  176354          CMP     #052524,@DRBA   ;LOOKING FOR Z-BIT TO SET
        002444  001401                          BEQ     .+4             ;DOES DRBA HAVE THE CORRECT PATTERN?
        002446  104000                          HLT                     ;DRBA DOESN'T HAVE THE CORRECT PATTERN
        002450  012777  125252  176342          MOV     #125252,@DRBA   ;ALT 1'S AND 0'S TO DRBA
        002456  022777  125252  176334          CMP     #125252,@DRBA   ;LOOKING FOR Z-BIT TO SET
        002464  001401                          BEQ     .+4             ;DOES DRBA HAVE THE CORRECT PATTERN?
        002466  104000                          HLT                     ;DRBA DOESN'T HAVE THE CORRECT PATTERN

                                        ;;***************************************************************
                                        ;       TEST22 INCREMENTING PATTERN TO WRAP-AROUND IN DRWC
                                        ;;***************************************************************
        002470  104400                          SCOPE
        002472  005067  007232                  CLR     ICOUNT
        002476  005001                          CLR     %1              ;SET-UP
        002500  005077  176312                  CLR     @DRWC           ;SET-UP
        002504  020177  176306          INCWC:  CMP     %1,@DRWC        ;SEE IF THEY ARE EQUAL
        002510  001401                          BEQ     .+4             ;ARE THEY EQUAL?
        002512  104000                          HLT                     ;THEY'RE NOT EQUAL
        002514  005277  176276                  INC     @DRWC           ;GET NEXT NUMBER
        002520  005201                          INC     %1              ;GET NEXT NUMBER
        002522  001370                          BNE     INCWC           ;DONE WITH TEST?  IF NOT CONTINUE

                                        ;;***************************************************************
                                        ;       TEST23 INCREMENTING PATTERN TO WRAP-AROUND IN DRBA
                                        ;;***************************************************************
        002524  104400                          SCOPE
        002526  005001                          CLR     %1              ;SET-UP
        002530  005077  176264                  CLR     @DRBA           ;SET-UP
        002534  020177  176260          INCBA:  CMP     %1,@DRBA        ;SEE IF THEY ARE EQUAL
        002540  001401                          BEQ     .+4             ;ARE THEY EQUAL?
        002542  104000                          HLT                     ;THEY'RE NOT EQUAL
        002544  062777  000002  176246          ADD     #2,@DRBA        ;GET NEXT NUMBER
        002552  062701  000002                  ADD     #2,%1           ;GET NEXT NUMBER
        002556  001366                          BNE     INCBA           ;DONE WITH TEST?  IF NOT CONTINUE


                                        ;;***************************************************************
                                        ;       TEST25 TEST THAT DR11-B DOES NOT INTERRUPT WITH PROC AT LEVEL 7
                                        ;;***************************************************************
```

```
                                                SCOPE
002560  104400
002562  012767  000340  007140                  MOV     #340,@PSW       ;STATUS AT LEVEL 7
002566  032777  000200  176216                  BIT     #BIT7,@DRST     ;CHECK READY BIT
002574  001010                                  BNE     P7INV           ;IS IT SET
002576  004767  007140                          JSR     %7,CKSWR
002612  000005                                  RESET                   ;INIT TO SET READY
002614  032777  000200  176200                  BIT     #BIT7,@DRST     ;SEE IF READY IS SET NOW
002622  001001                                  BNE     .+4             ;IS READY SET?
002624  104000                                  HLT                     ;READY CAN'T BE SET BY INIT
002626  012777  002654  176176  P7INV:  MOV     #P7ERR,@DRINV   ;SET UP INT VECTOR
002634  052777  000100  176160                  BIS     #BIT6,@DRST     ;SET IE
002642  000240                                  NOP
002644  042777  000100  176150                  BIC     #BIT6,@DRST     ;DR11-B DIDN'T INTERRUPT - CLEAR IE
002652  000401                                  BR      .+4
002654  104000          P7ERR:  HLT                     ;DR11-B INTERRUPTED, BUT IT SHOULDN'T HAVE

                        ;;**************************************************************
                        ;       TEST26 TEST THAT DR11-B DOES NOT INTERRUPT WITH PROC AT LEVEL 6
                        ;;**************************************************************
                                                SCOPE
002656  104400
002660  012777  000300  176126                  MOV     #300,@PSW       ;STATUS AT LEVEL 6
002666  032777  000200  176126                  BIT     #BIT7,@DRST     ;CHECK READY BIT
002674  001010                                  BNE     P6INV           ;IS IT SET?
002676  004767  007050                          JSR     %7,CKSWR
002702  000005                                  RESET                   ;INIT TO SET READY
002704  032777  000200  176110                  BIT     #BIT7,@DRST     ;SEE IF READY IS SET NOW
002712  001001                                  BNE     .+4             ;IS READY SET?
002714  104000                                  HLT                     ;READY CAN'T BE SET BY INIT
002716  012777  002744  176106  P6INV:  MOV     #P6ERR,@DRINV   ;SET UP INT VECTOR
002724  052777  000100  176070                  BIS     #BIT6,@DRST     ;SET IE
002732  000240                                  NOP
002734  042777  000100  176060                  BIC     #BIT6,@DRST     ;DR11-B DIDN'T INTERRUPT - CLEAR IE
002742  000401                                  BR      .+4
002744  104000          P6ERR:  HLT                     ;DR11-B INTERRUPTED, BUT IT SHOULDN'T HAVE


                        ;;**************************************************************
                        ;       TEST27 TEST THAT DR11-B DOES NOT INTERRUPT WITH PROC AT LEVEL 5
                        ;;**************************************************************
                                                SCOPE
002746  104400
002750  012777  000240  176036                  MOV     #240,@PSW       ;STATUS AT LEVEL 5
002756  032777  000200  176036                  BIT     #BIT7,@DRST     ;CHECK READY BIT
002764  001010                                  BNE     P5INV           ;IS IT SET?
002766  004767  006760                          JSR     %7,CKSWR
002772  000005                                  RESET                   ;INIT TO SET READY
002774  032777  000200  176020                  BIT     #BIT7,@DRST     ;SEE IF READY IS SET NOW
003002  001001                                  BNE     .+4             ;IS IT SET?
003004  104000                                  HLT                     ;RDY CAN'T BE SET BY INIT
003006  012777  003034  176016  P5INV:  MOV     #P5ERR,@DRINV   ;SET UP INT VECTOR
003014  052777  000100  176000                  BIS     #BIT6,@DRST     ;SET IE
003022  000240                                  NOP
003024  042777  000100  175770                  BIC     #BIT6,@DRST     ;DR11-B DIDN'T INTERRUPT - CLEAR IE
003032  000401                                  BR      .+4
003034  104000          P5ERR:  HLT                     ;DR11-B INTERRUPTED, BUT IT SHOULDN'T HAVE
```

```
                                        ;;*****************************************************************
                                        ;       TEST30 TEST THAT DR11-B DOES INTERRUPT WITH PROC AT LEVEL 4
                                        ;;*****************************************************************
003036  104400                                  SCOPE
003040  012777  000200  175746                  MOV     #200,@PSW       ;STATUS AT LEVEL 4
003046  032777  000200  175746                  BIT     #BIT7,@DRST     ;CHECK READY BIT
003054  001010                                  BNE     P4INV           ;IS IT SET?
003056  004767  006670                          JSR     %7,CKSWR
003062  000005                                  RESET                   ;INIT TO SET READY
003064  032777  000200  175730                  BIT     #BIT7,@DRST     ;SEE IF READY IS SET NOW
003072  001001                                  BNE     .+4             ;IS IT SET
003074  104000                                  HLT                     ;READY CAN'T BE SET BY INIT
003076  012777  003116  175726  P4INV:         MOV     #P4INT,@DRINV   ;SET UP INT VECTOR
003104  052777  000100  175710                  BIS     #BIT6,@DRST     ;SET IE
003112  000240                                  NOP
003114  104000                                  HLT                     ;DR11-B DIDN'T INTERRUPT
003116  005077  175700          P4INT:         CLR     @DRST           ;CLEAR IE
003122  022626                                  CMP     (%6)+,(%6)+     ;REPOSITION THE STACK AFTER AN INTERRUPT
003124  012777  000126  175700                  MOV     #126,@DRINV     ;RESTORE INTERRUPT VECTOR


                                        ;;*****************************************************************
                                        ;       TEST31 TEST THAT FNCT BITS CONTROL DSTAT BITS (M968 MUST BE USED IN USER SLOTS)
                                        ;;*****************************************************************
003132  104400                                  SCOPE
003134  005077  175662                          CLR     @DRST           ;CLEAR FUNCTION BITS
003140  032777  000016  175654                  BIT     #16,@DRST       ;CHECK FUNCTION BITS
003146  001401                                  BEQ     .+4             ;FUNCTION BITS CLEAR?
003150  104000                                  HLT                     ;FUNCTION BITS NOT CLEAR
003152  052777  000002  175642                  BIS     #BIT1,@DRST     ;SET FNCT1
003160  032777  001000  175634                  BIT     #BIT9,@DRST     ;CHECK DSTAT C
003166  001001                                  BNE     .+4             ;IS IT SET?
003170  104000                                  HLT                     ;DSTAT C IS CLEAR
003172  032777  006000  175622                  BIT     #6000,@DRST     ;CHECK THAT DSTAT A AND DSTAT B ARE CLEAR
003200  001401                                  BEQ     .+4             ;ARE THEY CLEAR?
003202  104000                                  HLT                     ;DSTAT A AND/OR DSTAT B IS SET
003204  005077  175612                          CLR     @DRST           ;CLEAR DRST
003210  052777  000004  175604                  BIS     #BIT2,@DRST     ;SET FNCT2
003216  032777  002000  175576                  BIT     #BIT10,@DRST    ;CHECK DSTAT B
003224  001001                                  BNE     .+4             ;IS IT SET?
003226  104000                                  HLT                     ;DSTAT B IS CLEAR
003230  032777  005000  175564                  BIT     #5000,@DRST     ;CHECK THAT DSTAT A AND DSTAT C ARE CLEAR
003236  001401                                  BEQ     .+4             ;ARE THEY CLEAR?
003240  104000                                  HLT                     ;DSTAT A AND/OR DSTAT B IS SET
003242  005077  175554                          CLR     @DRST           ;CLEAR DRST
003246  052777  000010  175546                  BIS     #BIT3,@DRST     ;SET FNCT3
003254  032777  004000  175540                  BIT     #BIT11,@DRST    ;CHECK DSTAT A
003262  001001                                  BNE     .+4             ;IS IT SET?
003264  104000                                  HLT                     ;DSTAT A IS CLEAR
003266  032777  003000  175526                  BIT     #3000,@DRST     ;CHECK THAT DSTAT B AND DSTAT C ARE CLEAR
003274  001401                                  BEQ     .+4             ;ARE THEY CLEAR?
003276  104000                                  HLT                     ;DSTAT B AND/OR DSTAT C IS SET
003300  005077  175516                          CLR     @DRST           ;CLR DRST
                                        ;;*****************************************************************
                                        ;       TEST 33 TEST FOR 1 DATI NPR TRANSFER (WITH M968 IN USER SLOTS)
                                        ;;*****************************************************************
```

```
                                                SCOPE
003304  104400
003306  005777  175510          TNPR1:  TST     @DRST           ;CHECK ERROR BIT
003312  100027                          BPL     NPRRDY          ;IS IT CLEAR?
003314  032777  020000  175500          BIT     #BIT13,@DRST    ;CHECK ATTN
003322  001401                          BEQ     .+4             ;IS ATTN CLEAR
003324  104000                          HLT                     ;ATTN IS SET
003326  032777  040000  175466          BIT     #BIT14,@DRST    ;CHECK NEX
003334  001410                          BEQ     N1413           ;IS NEX CLEAR?
003336  042777  040000  175456          BIC     #BIT14,@DRST    ;TRY TO CLEAR NEX
003344  032777  040000  175450          BIT     #BIT14,@DRST    ;CHECK AGAIN
003352  001401                          BEQ     .+4             ;NEX STILL SET
003354  104000                          HLT                     ;NEX CAN'T BE CLEARED BY MOVING A 0 TO IT
003356  005077  175436          N1413:  CLR     @DRBA           ;TRY TO CLEAR BAOF
003362  005777  175434                  TST     @DRST           ;CHECK EPROR BIT AGAIN
003366  001401                          BEQ     .+4             ;IS IT CLEAR
003370  104000                          HLT                     ;ERROR CAUSED BY SOMETHING OTHER THAN NEX,ATTN, OR BAOF
003372  012777  177777  175416  NPRRDY: MOV     #-1,@DRWC       ;SET UP FOR 1 TRANSFER
003400  012777  001034  175412          MOV     #NPR1,@DRBA     ;TRANSFER FROM BUS ADDRESS IN NPR1
003406  005077  175412                  CLR     @DRDB           ;GET READY TO RECEIVE DATA
003412  012767  052525  175414          MOV     #52525,NPR1     ;SET UP TRANSFER DATA
003420  012777  003464  175404          MOV     #INTB,@DRINV    ;INTERRUPT VECTOR TC INTB
003426  012777  000005  175372          MOV     #5,@DRVS        ;INTERRUPT PRICRITY TO LEVEL 5
003434  005077  175354                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
003440  012777  000101  175354          MOV     #101,@DRST      ;IE AND DO TO DRST
003446  005067  000002                  CLR     1$+2            ;WAIT FOR NPR AND INTERRUPT
003452  005227  000001          1$:     INC     #1
003456  001375                          BNE     1$
003460  104000                          HLT                     ;NO DR11-B INTERRUPT
003462  000424                          BR      T33CLR          ;CLEAR IE
003464  004767  003312          INTB:   JSR     %7,ERRCHK
003470  005777  175322                  TST     @DRWC           ;TEST DRWC
003474  001401                          BEQ     .+4             ;IS DRWC EQUAL TO ZERO?
003476  104000                          HLT                     ;DRWC NOT EQUAL TO ZERO
003500  022777  001036  175312          CMP     #NPR1+2,@DRBA   ;COMPARE CORRECT DRBA WITH DRBA
003506  001401                          BEQ     .+4             ;IS THE DRBA CORRECT?
003510  104000                          HLT                     ;DRBA IS WRONG
003512  022777  052525  175304          CMP     #52525,@DRDB    ;CHECK FOR CORRECT DATA
003520  001401                          BEQ     .+4             ;DATA GET TRANSFERRED?
003522  104000                          HLT                     ;BAD DATA IN DRDB
003524  004767  003162                  JSR     %7,NORMAL
003530  022626                          CMP     (%6)+,(%6)+     ;RESTORE STACK
003532  000403                          BR      TNPRO           ;GO TO NEXT TEST (NPR OUT)
003534  005077  175262          T33CLR: CLR     @DRST           ;CLEAR IE
003540  000662                          BR      TNPR1           ;TRY TEST AGAIN
                                ;;**************************************************************
                                ;       TEST 34 TEST FOR 1 DATO NPR TRANSFER (WITH M968 IN USER SLOTS)
                                ;;**************************************************************
003542  104400                  TNPRO:  SCOPE
003544  012777  177777  175244          MOV     #-1,@DRWC       ;SET UP FOR 1 TRANSFER
003552  012777  001034  175240          MOV     #NPR1,@DRBA     ;TRANSFER TO BUS ADDRESS IN NPR1
003560  005067  175250                  CLR     NPR1            ;GET READY TO RECEIVE DATA
003564  012777  052525  175232          MOV     #52525,@DRDB    ;SET UP TO TRANSFER DATA
003572  012777  003636  175232          MOV     #INTC,@DRINV    ;INTERRUPT VECTOR TO INTC
003600  016777  175224  175220          MOV     DRINL,@DRVS     ;INTERRUPT STATUS TO LEVEL DRINL
003606  005077  175202                  CLR     @PSW            ;PROC STATUS TO ZERO
003612  012777  000103  175202          MOV     #103,@DRST      ;IE, FNCTION CONTROL, AND DO TO DRST
```

```
1050  003620  005067  000002            CLR     1$+2            ;WAIT FOR NPR AND INTER
1051  003624  005227  000001    1$:     INC     #1
1052  003630  001375                    BNE     1$
1053  003632  104000                    HLT                     ;NO DR11-B INTERRUPT
1054  003634  000424                    BR      T34CLR          ;CLEAR IE
1055  003636  004767  003140    INTC:   JSR     %7,ERRCHK
1056  003642  005777  175150            TST     @DRWC           ;TEST DRWC
1057  003646  001401                    BEQ     .+4             ;IS DRWC EQUAL TO ZERO?
1058  003650  104000                    HLT                     ;DRWC EQLAL TO ZERO
1059  003652  022777  001036  175140    CMP     #NPR1+2,@DRBA   ;COMPARE CORRECT DR3A WITH DR3A
1060  003660  001401                    BEQ     .+4             ;IS THE DRBA CORRECT?
1061  003662  104000                    HLT                     ;DRBA IS WRONG
1062
1063  003664  026727  175144  052525    CMP     NPR1,#52525     ;CHECK FOR CORRECT DATA
1064  003672  001401                    BEQ     .+4             ;CORRECT DATA TRANSFERRED?
1065  003674  104000                    HLT                     ;BAD DATA
1066  003676  004767  003010            JSR     %7,NORMAL
1067  003702  022626                    CMP     (%6)+,(%6)+     ;RESTORE STACK
1068  003704  000403                    BR      T35             ;GO TO NEXT TEST
1069  003706  005077  175110    T34CLR: CLR     @DRST           ;CLEAR IE
1070  003712  000713                    BR      TNPRO           ;TRY TEST AGAIN
1071
1072                                    ;;**************************************************************
1073                                    ;       TEST 35 STRING OF 10 DATI'S (WITH M958 IN USER SLOTS)
1074                                    ;;**************************************************************
1075  003714  104430            T35:    SCOPE
1076  003716  012767  000020  175120    MOV     #20,BUFLEN      ;LENGTH OF BUFFER=20
1077  003724  004767  002524            JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
1078  003730  006267  175110            ASR     BUFLEN          ;BUFLEN=10
1079  003734  016767  175104  175110    MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
1080  003742  005467  175104            NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
1081  003746  016777  175100  175042    MOV     WCLEN,@DRWC     ;SET UP DRWC
1082  003754  016777  175060  175036    MOV     INBUF,@DRBA     ;SET UP DRBA
1083  003762  012777  177777  175034    MOV     #-1,@DRDB       ;MAINT AIDE
1084  003770  012777  006556  175034    MOV     #INTA,@DRINV    ;INT VECTOR TO INTA
1085  003776  016777  175026  175022    MOV     DRINL,@DRVS     ;INT VECTOR TO PRIORITY DRINL
1086  004004  005077  175004            CLR     @PSW            ;LET THE DR11-B INTERRUPT
1087  004010  012777  000101  175004    MOV     #101,@DRST      ;IE AND DO TO DRST
1088  004016  000777                    BR      .               ;WAIT FOR INTERRUPT
1089  004020  022777  000007  174776    CMP     #7,@DRDB        ;CHECK THAT WORD #10 OF INBUF IS IN DRBA
1090  004026  001401                    BEQ     .+4             ;IS IT?
1091  004030  104000                    HLT                     ;BAD DATA IN DRDB
1092
1093                                    ;;**************************************************************
1094                                    ;       TEST 36 STRING OF 10 DATO'S (WITH M968 IN USER SLOTS)
1095                                    ;;**************************************************************
1096  004032  104400            T36:    SCOPE
1097  004034  012767  000020  175002    MOV     #20,BUFLEN      ;LENGTH OF BUFFER=20
1098  004042  004767  002406            JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
1099  004046  006267  174772            ASR     BUFLEN          ;BUFLEN=10
1100  004052  016767  174766  174772    MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
1101  004060  005467  174766            NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
1102  004064  016777  174762  174724    MOV     WCLEN,@DRWC     ;SET UP DRWC
1103  004072  016777  174742  174720    MOV     INBUF,@DRBA     ;SET UP DRBA
1104  004100  012777  052525  174716    MOV     #52525,@DRDB    ;SET UP DRDB
1105  004106  012777  006556  174716    MOV     #INTA,@DRINV    ;INTERRUPT VECTOR TO INTA
```

```
 1106  004114  016777  174710  174704          MOV     DRINL,@DRVS     ;INTERRUPT VECTOR TO PRIORITY DRINL
 1107  004122  005077  174666                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1108  004126  012777  000103  174666          MOV     #103,@DRST      ;IE, FNCTI(C1 CONTROL), AND DO TO DRST
 1109  004134  000777                          BR      .               ;WAIT FOR INTERRUPT
 1110  004136  004767  002572                  JSR     %7,DATOCK       ;CHECK INBUF
 1111
 1112                                  ;;**************************************************************
 1113                                  ;       TEST 37 STRING OF 200 DATI'S
 1114                                  ;;**************************************************************
 1115  004142  104400                          SCOPE
 1116  004144  012767  000200  174672          MOV     #200,BUFLEN     ;LENGTH OF BUFFER=200
 1117  004152  004767  002276                  JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
 1118  004156  016767  174662  174666          MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
 1119  004164  005467  174662                  NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
 1120  004170  016777  174656  174620          MOV     WCLEN,@DRWC     ;SET UP DRWC
 1121  004176  016777  174636  174614          MOV     INBUF,@DRBA     ;SET UP DRBA
 1122  004204  012777  177777  174612          MOV     #-1,@DRDB       ;MAINT AIDE
 1123  004212  012777  006556  174612          MOV     #INTA,@DRINV    ;INT VECTOR TO INTA
 1124  004220  016777  174604  174600          MOV     DRINL,@DRVS     ;INT VECTOR TO PRIORITY DRINL
 1125  004226  005077  174562                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1126  004232  012777  000101  174562          MOV     #101,@DRST      ;IE AND DO TO DRST
 1127  004240  000777                          BR      .               ;WAIT FOR INTERRUPT
 1128  004242  022777  000177  174554          CMP     #177,@DRDB      ;CHECK THAT WORD #200 OF INBUF IS IN DRBA
 1129  004250  001401                          BEQ     .+4             ;IS IT?
 1130  004252  104000                          HLT                     ;BAD DATA IN DRDB
 1131
 1132                                  ;;**************************************************************
 1133                                  ;       TEST 40 STRING OR 200 DATO'S
 1134                                  ;;**************************************************************
 1135  004254  104400                          SCOPE
 1136  004256  012767  000201  174560          MOV     #201,BUFLEN     ;LENGTH OF BUFFER=201
 1137  004264  004767  002164                  JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
 1138  004270  005367  174550                  DEC     BUFLEN          ;BUFLEN=200
 1139  004274  016767  174544  174550          MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
 1140  004302  005467  174544                  NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
 1141  004306  016777  174540  174502          MOV     WCLEN,@DRWC     ;SET UP DRWC
 1142  004314  016777  174520  174476          MOV     INBUF,@DRBA     ;SET UP DRBA
 1143  004322  012777  052525  174474          MOV     #52525,@DRDB    ;SET UP DRDB
 1144  004330  012777  006556  174474          MOV     #INTA,@DRINV    ;INTERRUPT VECTOR TO INTA
 1145  004336  016777  174466  174462          MOV     DRINL,@DRVS     ;INTERRUPT VECTOR TO PRIORITY DRINL
 1146  004344  005077  174444                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1147  004350  012777  000103  174444          MOV     #103,@DRST      ;IE, FNCTI, AND DO TO DRST
 1148  004356  000777                          BR      .               ;WAIT FOR INTERRUPT
 1149  004360  004767  002350                  JSR     %7,DATOCK       ;CHECK INBUF
 1150
 1151                                  ;;**************************************************************
 1152                                  ;       TEST 42 TEST THAT DOING A DATO TO THE DIODE MEMORY CAUSES NEX
 1153                                  ;;**************************************************************
 1154  004364  104400                          SCOPE
 1155  004366  012777  177776  174422          MOV     #-2,@DRWC       ;SET UP DRWC
 1156  004374  016777  174436  174416          MOV     DIOMEM,@DRBA    ;SET UP DRBA
 1157  004402  012777  004436  174422          MOV     #NEXCHK,@DRINV  ;INTERRUPT VECTOR TO NEXCHK
 1158  004410  016777  174414  174410          MOV     DRINL,@DRVS     ;INTERRUPT STATUS TO LEVEL DRINL
 1159  004416  005077  174372                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1160  004422  012777  000163  174372          MOV     #163,@DRST      ;IE, FNCTI, XBA17, XBA16, AND GO TO DRST
 1161  004430  005237  177560                  INC     @#177560        ;WAIT FOR INTERRUPT
 1162  004434  104000                          HLT                     ;NO DR11-B INTERRUPT
```

```
 1162  004436  005777  174360          NEXCHK: TST     @DRST           ;TEST CRST
 1163  004442  001001                          BNE     .+4             ;ERROR SET?
 1164  004444  104000                          HLT                     ;ERROR NOT SET
 1165  004446  105777  174350                  TSTB    @DRST           ;TEST FOR READY
 1166  004452  001001                          BNE     .+4             ;READY SET?
 1167  004454  104000                          HLT                     ;READY ISN'T SET
 1168  004456  032777  040000  174336          BIT     #BIT14,@DRST    ;CHECK NEX
 1169  004464  001001                          BNE     .+4             ;NEX SET?
 1170  004466  104000                          HLT                     ;NEX IS CLEAR
 1171  004470  022626                          CMP     (%6)+,(%6)+     ;RESTORE THE STACK
 1172  004472  004767  002214                  JSR     %7,NORMAL
 1173
 1174                                  ;;**************************************************************
 1175                                  ;       TEST 43 TEST THAT BAOF FORCES ERROR AND READY AND THAT BAOF IS
 1176                                  ;       CLEARED BY CLEARING THE DRBA OR A RESET
 1177                                  ;;**************************************************************
 1178  004476  104400                          SCOPE
 1179  004500  012767  000010  005222          MOV     #10,ICOUNT
 1180  004506  012777  177760  174302          MOV     #-20,@DRWC      ;SET UP DRWC
 1181  004514  012777  177776  174276          MOV     #-2,@DRBA       ;SET UP DRBA FOR PROC STATUS ADDRESS
 1182  004522  012777  004556  174302          MOV     #BAOFCK,@DRINV  ;INTERRUPT VECTOR TO BAOFCK
 1183  004530  016777  174274  174270          MOV     DRINL,@DRVS     ;INTERRUPT STATUS TO LEVEL DRINL
 1184  004536  005077  174252                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1185  004542  012777  000163  174252          MOV     #163,@DRST      ;IE, FNCT1, XBA17, XBA16, AND GO TO DRST
 1186  004550  005237  177560                  INC     @#177560        ;WAIT FOR INTERRUPT
 1187  004554  104000                          HLT                     ;NO DR11-B INTERRUPT
 1188  004556  022626                  BAOFCK: CMP     (%6)+,(%6)+     ;RESTORE THE STACK
 1189  004560  005777  174236                  TST     @DRST           ;TEST DRST
 1190  004564  100401                          BMI     .+4             ;ERROR SET?
 1191  004566  104000                          HLT                     ;ERROR NOT SET
 1192  004570  105777  174226                  TSTB    @DRST           ;TEST FOR READY
 1193  004574  100401                          BMI     .+4             ;READY SET?
 1194  004576  104000                          HLT                     ;READY ISN'T SET
 1195  004600  042777  040000  174214          BIC     #BIT14,@DRST    ;CLEAR NEX
 1196  004606  032777  060000  174206          BIT     #60000,@DRST    ;CHECK NEX AND ATTN
 1197  004614  001401                          BEQ     .+4             ;ARE THEY CLEAR?
 1198  004616  104000                          HLT                     ;NEX AND/OR ATTN IS SET
 1199  004620  005777  174176                  TST     @DRST           ;TEST FOR ERROR
 1200  004624  100401                          BMI     .+4             ;IS ERROR SET?
 1201  004626  104000                          HLT                     ;ERROR IS CLEAR
 1202  004630  005077  174164                  CLR     @DRBA           ;CLEAR BAOF
 1203  004634  005777  174162                  TST     @DRST           ;CHECK ERROR
 1204  004640  100001                          BPL     .+4             ;SHOULD BE CLEAR
 1205  004642  104000                          HLT                     ;CLEARING DRBA DIDN'T CLEAR BAOF
 1206
 1207  004644  012777  177776  174146          MOV     #-2,@DRBA       ;SET UP DRBA FOR PROC STATUS ADDRESS
 1208  004652  012777  004706  174152          MOV     #BAOCK1,@DRINV  ;INTERRUPT VECTOR TO BAOCK1
 1209  004660  016777  174144  174140          MOV     DRINL,@DRVS     ;INTERRUPT STATUS TO LEVEL DRINL
 1210  004666  005077  174122                  CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1211  004672  012777  000163  174122          MOV     #163,@DRST      ;I.E., XBA17,XBA16,FNCT1, AND GO TO DRST
 1212  004700  005237  177560                  INC     @#177560        ;WAIT FOR INTERRUPT
 1213  004704  104000                          HLT                     ;NO DR11-B INTERRUPT
 1214  004706  022626                  BAOCK1: CMP     (%6)+,(%6)+     ;RESTORE THE STACK
 1215  004710  005777  174106                  TST     @DRST           ;TEST DRST
 1216  004714  100401                          BMI     .+4             ;ERROR SET?
 1217  004716  104000                          HLT                     ;ERROR NOT SET
```

```
1218  004720  105777  174076          TSTB    @DRST           ;TEST FOR READY
1219  004724  100401                  BMI     .+4             ;READY SET?
1220  004726  104000                  HLT                     ;READY ISN'T SET
1221  004730  042777  040000  174064  BIC     #BIT14,@DRST    ;CLEAR NEX
1222  004736  032777  060000  174056  BIT     #60000,@DRST    ;CHECK NEX AND ATTN
1223  004744  001401                  BEQ     .+4             ;ARE THEY CLEAR?
1224  004746  104000                  HLT                     ;NEX AND/OR ATTN IS SET
1225  004750  005777  174046          TST     @DRST           ;TEST FOR ERROR
1226  004754  100401                  BMI     .+4             ;IS ERROR SET?
1227  004756  104000                  HLT                     ;ERROR IS CLEAR
1228  004760  004767  004766          JSR     %7,CKSWR
1229  004764  000005                  RESET                   ;INIT
1230  004766  005777  174030          TST     @DRST           ;CHECK ERROR
1231  004772  100001                  BPL     .+4             ;SHOULD BE CLEAR
1232  004774  104000                  HLT                     ;RESET DIDN'T CLEAR BADF
1233  004776  004767  001710          JSR     %7,NORMAL
1234
1235                                  ;;******************************************************************
1236                                  ;       TEST 44 TEST THAT RESET CLEARS DRDB
1237                                  ;;******************************************************************
1238  005002  104400                  SCOPE
1239  005004  012767  000010  004716  MOV     #10,ICOUNT
1240  005012  012777  177777  174004  MOV     #-1,@DRDB       ;ALL ONES TO DRBA
1241  005020  004767  004726          JSR     %7,CKSWR
1242  005024  000005                  RESET                   ;INIT
1243  005026  005777  173772          TST     @DRDB           ;LOOKING FOR Z-BIT TO SET
1244  005032  001401                  BEQ     .+4             ;DID DRDB GET CLEARED?
1245  005034  104000                  HLT                     ;DRDB NOT CLEAR
1246
1247                                  ;;******************************************************************
1248                                  ;       TEST 45 TEST THAT ALL DRDB BITS CAN BE SET
1249                                  ;;******************************************************************
1250  005036  104400                  SCOPE
1251  005040  012767  004000  004662  MOV     #4000,ICOUNT
1252  005046  012777  177777  173750  MOV     #-1,@DRDB       ;SET ALL BITS IN DRDB
1253  005054  022777  177777  173742  CMP     #-1,@DRDB       ;LOOKING FOR Z-BIT TO SET
1254  005062  001401                  BEQ     .+4             ;SEE IF ALL BITS GOT SET
1255  005064  104000                  HLT                     ;ALL DRDB BITS AREN'T SET
1256
1257                                  ;;******************************************************************
1258                                  ;       TEST 46 TEST THAT DRDB CAN HOLD ALTERNATE ONE'S AND ZERO'S
1259                                  ;;******************************************************************
1260  005066  104400                  SCOPE
1261  005070  012777  052525  173726  MOV     #052525,@DRDB   ;ALT 0'S AND 1'S TO DRDB
1262  005076  022777  052525  173720  CMP     #052525,@DRDB   ;LOOKING FOR Z-BIT TO SET
1263  005104  001401                  BEQ     .+4             ;DOES DRDB HAVE THE CORRECT PATTERN?
1264  005106  104000                  HLT                     ;DRDB IS WRONG
1265  005110  012777  125252  173706  MOV     #125252,@DRDB   ;ALT 1'S AND 0'S TO DRDB
1266  005116  022777  125252  173700  CMP     #125252,@DRDB   ;LOOKING FOR Z-BIT TO SET
1267  005124  001401                  BEQ     .+4             ;DOES DRDB HAVE THE CORRECT PATTERN
1268  005126  104000                  HLT                     ;DRDB IS WRONG
1269
1270                                  ;;******************************************************************
1271                                  ;       TEST 47 INCREMENTING PATTERN TO WRAP-AROUND IN DRDB
1272                                  ;;******************************************************************
1273  005130  104400                  SCOPE
```

```
 1274  005132  005067  004572                  CLR     ICOUNT
 1275  005136  005001                          CLR     %1              ;SET-UP
 1276  005140  005077  173660                  CLR     @DRDB           ;SET-UP
 1277  005144  020177  173654          INCDB:  CMP     %1,@DRDB        ;SEE IF THEY ARE EQUAL
 1278  005150  001401                          BEQ     .+4             ;ARE THEY EQUAL?
 1279  005152  104000                          HLT                     ;THEY'RE NOT EQUAL
 1280  005154  005277  173644                  INC     @DRDB           ;GET NEXT NUMBER
 1281  005160  005201                          INC     %1              ;GET NEXT NUMBER
 1282  005162  001370                          BNE     INCDB           ;DONE WITH TEST? IF NOT CONTINUE
 1283
 1284                                  ;;**************************************************************
 1285                                  ;       TEST 50 TEST THAT RESET SETS READY AND CLEARS ALL OTHER
 1286                                  ;       DRST BITS (WITH M968 INSERTED)
 1287                                  ;;**************************************************************
 1288  005164  104400                          SCOPE
 1289  005166  004767  004560                  JSR     %7,CKSWR
 1290  005172  000005                          RESET                   ;INIT
 1291  005174  032777  000200  173620          BIT     #BIT7,@DRST     ;CHECK DRST
 1292  005202  001001                          BNE     .+4             ;IS READY SET?
 1293  005204  104000                          HLT                     ;READY IS CLEAR
 1294  005206  032777  177577  173606          BIT     #177577,@DRST   ;CHECK DRST
 1295  005214  001401                          BEQ     .+4             ;ARE THEY ALL CLEAR?
 1296  005216  104000                          HLT                     ;A BIT OTHER THAN READY IS SET IN THE DRST
 1297
 1298                                  ;;**************************************************************
 1299                                  ;       TEST 51 TEST THAT BADO READS AS A ZERO WITH MAINT BOARD INSERTED
 1300                                  ;;**************************************************************
 1301  005220  104400                          SCOPE
 1302  005222  012767  004000  004500          MOV     #4000,ICOUNT
 1303  005230  032777  000001  173582          BIT     #BIT0,@DRBA     ;TEST BIT 0 OF DRBA
 1304  005236  001401                          BEQ     .+4             ;IS IT CLEAR?
 1305  005240  104000                          HLT                     ;BADO IS SET
 1306
 1307
 1308
 1309                                  ;;**************************************************************
 1310                                  ;       TEST 52 TEST THAT GO CLEARS READY
 1311                                  ;;**************************************************************
 1312  005242  104400                          SCOPE
 1313  005244  012767  004000  004456          MOV     #4000,ICOUNT
 1314  005252  012777  177600  173536          MOV     #-200,@DRWC     ;SET-UP DRWC
 1315  005260  016777  173554  173532          MOV     INBUF,@DRBA     ;SET-UP DRBA
 1316  005266  105777  173530                  TSTB    @DRST           ;CHECK READY
 1317  005272  100401                          BMI     .+4             ;IS READY SET?
 1318  005274  104000                          HLT                     ;READY IS CLEAR
 1319  005276  012777  000011  173516          MOV     #11,@DRST       ;FNCT3 (NON-BURST) AND GO TO DRST
 1320  005304  105777  173512                  TSTB    @DRST           ;CHECK READY
 1321  005310  100001                          BPL     .+4             ;IS READY CLEAR?
 1322  005312  104000                          HLT                     ;READY IS STILL SET
 1323  005314  005067  173534                  CLR     RDYCHK          ;CLEAR READY CHECK
 1324  005320  105777  173476          TSTRDY: TSTB    @DRST           ;CHECK READY
 1325  005324  100406                          BMI     DONE            ;IF SET GO TO DONE
 1326  005326  062767  000004  173520          ADD     #4,RDYCHK       ;CHECKING TIME FOR READY TO BE SET
 1327  005334  100401                          BMI     .+4             ;IF RDYCHK GETS NEGATIVE IT TOOK TOO LONG
 1328  005336  000770                          BR      TSTRDY          ;CHECK AGAIN
 1329  005340  104000                          HLT                     ;READY GOT CLEARED BUT NEVER SET AGAIN
```

```
1330  005342  000240                    DONE:   NOP                     ;GO TO NEXT TEST
1331                                    ;;**************************************************************
1332                                    ;TEST 55 TEST THAT GIVING A DO WITHOUT CLEARING A PREVIOUS ERROR
1333                                    ;CAUSES ANOTHER INTERRUPT
1334                                    ;;**************************************************************
1335  005344  104400                            SCOPE
1336  005346  012777  177760  173442            MOV     #-20,@DRWC      ;SET-UP DRWC
1337  005354  016777  173456  173436            MOV     DIOMEM,@DRBA    ;SET-UP DRBA
1338  005362  012777  005416  173442            MOV     #ERRD0,@DRINV   ;INTERRUPT VECTOR TO ERRD0
1339  005370  012777  000200  173430            MOV     #200,@DRVS      ;INTERRUPT STATUS TO LEVEL 4
1340  005376  005077  173412                    CLR     @PSW            ;LET THE DR11-B INTERRUPT
1341  005402  012777  000163  173412            MOV     #163,@DRST      ;IE, XBA17,XBA16,FNCT1 AND GO TO DRST
1342  005410  005277  173442                    INC     @TKS            ;WAIT FOR INTERJPT
1343  005414  104000                            HLT                     ;NO DR11-B INTERRUPT
1344  005416  005777  173400            ERRD0:  TST     @DRST           ;TEST DRST
1345  005422  100401                            BMI     .+4             ;ERROR SET?
1346  005424  104000                            HLT                     ;ERROR IS CLEAR - SHOULD HAVE NEX
1347  005426  012777  005474  173376            MOV     #ERRD01,@DRINV  ;INTERRUPT VECTOR TO ERRD01
1348  005434  005077  173360                    CLR     @DRBA           ;PREVENT CAUSING ANOTHER ERROR
1349  005440  042777  000062  173354            BIC     #62,@DRST       ;CLEAR XBA17, XBA16, AND FNCT1
1350  005446  012777  177777  173342            MOV     #-1,@DRWC       ;SET-UP DRWC
1351  005454  005277  173342                    INC     @DRST           ;DO TO DRST
1352  005460  005067  000002                    CLR     1$+2
1353  005464  005227  000001            1$:     INC     #1
1354  005470  001375                            BNE     1$
1355  005472  104000                            HLT                     ;NO DR11-B INTERRUPT
1356  005474  005777  173322            ERRD01: TST     @DRST           ;CHECK ERROR
1357  005500  100401                            BMI     .+4             ;ERROR SET?
1358  005502  104000                            HLT                     ;ERROR IS CLEAR - SHOULD BE SET BECAUSE
1359                                                                    ;PREVIOUS ERROR WAS NOT CLEARED
1360  005504  062706  000010                    ADD     #10,%6          ;REPOSITION THE STACK
1361  005510  004767  001176                    JSR     %7,NORMAL
1362
1363                                    ;;**************************************************************
1364                                    ;       TEST 56 STRING OF 200 DATI'S NON-BURST MODE
1365                                    ;;**************************************************************
1366  005514  104400                            SCOPE
1367  005516  012767  000200  173320            MOV     #200,BUFLEN     ;LENGTH OF BUFFER=200
1368  005524  004767  000724                    JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
1369  005530  016767  173310  173314            MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
1370  005536  005467  173310                    NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
1371  005542  016777  173304  173246            MOV     WCLEN,@DRWC     ;SET-UP DRWC
1372  005550  016777  173264  173242            MOV     INBUF,@DRBA     ;SET-UP DRBA
1373  005556  012777  177777  173240            MOV     #-1,@DRDB       ;MAINT AIDE
1374  005564  012777  006556  173240            MOV     #INTA,@DRINV    ;INT VECTOR TO INTA
1375  005572  016777  173232  173226            MOV     DRINL,@DRVS     ;INT VECTOR TO PRIORITY DRINL
1376  005600  005077  173210                    CLR     @PSW            ;LET THE DR11-B INTERRUPT
1377  005604  012777  000111  173210            MOV     #111,@DRST      ;IE, FNCT3, AND DO TO DRST
1378  005612  005267  173232                    INC     BRWAIT          ;USE A WAIT OR BR. INSTRUCTION
1379  005616  032767  000001  173224            BIT     #BIT0,BRWAIT    ;SEE WHICH ONE
1380  005624  001403                            BEQ     DATINB          ;BIT 0 CLEAR=BR.
1381  005626  000001                            WAIT                    ;WAIT FOR INTERRUPT
1382  005630  000240                            NOP
1383  005632  000401                            BR      .+4
1384  005634  000777            DATINB: BR      .
1385  005636  022777  000177  173160            CMP     #177,@DRDB      ;CHECK THAT WORD #200 OF INBUF IS IN DRBA
```

```
 1396  005644  001401                        BEQ     .+4             ;IS IT?
 1397  005646  104000                        HLT                     ;BAD DATA IN DRDB
 1398
 1399
 1400                                ;;*****************************************************************
 1401                                ;       TEST 57 STRING OF 200 DATO'S NON-BURST MODE
 1402                                ;;*****************************************************************
 1403  005650  104400                        SCOPE
 1404  005652  012767  000201  173164        MOV     #201,BUFLEN     ;LENGTH OF BUFFER=201
 1405  005660  004767  000570                JSR     %7,LODBUF       ;LOAD THE BUFFER WITH INCREMENTING PATTERN
 1406  005664  005367  173154                DEC     BUFLEN          ;BUFLEN=200
 1407  005670  016767  173150  173154        MOV     BUFLEN,WCLEN    ;PREPARE NUMBER FOR DRWC
 1408  005676  005467  173150                NEG     WCLEN           ;2'S COMPLEMENT OF BUFLEN
 1409  005702  016777  173144  173106        MOV     WCLEN,@DRWC     ;SET UP DRWC
 1410  005710  016777  173124  173102        MOV     INBUF,@DRBA     ;SET UP DRBA
 1411  005716  012777  052525  173100        MOV     #52525,@DRDB    ;SET UP DRDB
 1412  005724  012777  006556  173100        MOV     #INTA,@DRINV    ;INTERRUPT VECTOR TO INTA
 1413  005732  016777  173072  173066        MOV     DRINL,@DRVS     ;INTERRUPT VECTOR TO PRIORITY DRINL
 1414  005740  005077  173050                CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1415  005744  012777  000113  173050        MOV     #113,@DRST      ;IE, FNCT3, FNCT1, AND DO TO DRST
 1416  005752  005267  173072                INC     BRWAIT          ;USE A WAIT OR BR. INSTRUCTION
 1417  005756  032767  000001  173064        BIT     #BIT0,BRWAIT    ;BIT 0 CLEAR=BR.
 1418  005764  001403                        BEQ     DATONB
 1419  005766  000001                        WAIT                    ;WAIT FOR INTERRUPT
 1420  005770  000240                        NOP
 1421  005772  000401                        BR      .+4
 1422  005774  000777              DATONB:   BR      .
 1423  005776  004767  000732                JSR     %7,DATOCK       ;CHECK INBUF
 1424
 1425                                ;;*****************************************************************
 1426                                ;       TEST 60 TEST THAT FUNCTION BITS INCREMENT WITH MAINT MODE TRANSFERS
 1427                                ;;*****************************************************************
 1428  006002  104400                        SCOPE
 1429  006004  012767  000010  173032        MOV     #10,BUFLEN      ;SET-UP BUFLEN FOR LODBUF AND CHKBUFF
 1430  006012  016777  173022  173000        MOV     INBUF,@DRBA     ;SET-UP DR2A
 1431  006020  004767  000430                JSR     %7,LODBUF       ;LOAD INBUF
 1432  006024  004767  000462                JSR     %7,CHKBFF       ;LOAD CHKBUF
 1433  006030  005077  172766                CLR     @DRST           ;INIT FOR STARTING
 1434  006034  012767  000001  173024        MOV     #1,FNCONT       ;GET READY FOR CHECKING
 1435  006042  012767  000001  172774        MOV     #1,BUFLEN       ;CHANGE IS NECESSARY FOR INTA ROUTINE
 1436  006050  016767  172764  173012        MOV     INBUF,INBUF1    ;SAVE INBUF
 1437  006056  012777  006556  172746  MFLOOP: MOV     #INTA,@DRINV    ;INTERRUPT VECTOR TO INTA
 1438  006064  016777  172740  172734        MOV     DRINL,@DRVS     ;INTERRUPT VECTOR PRIORITY TO DRINL
 1439  006072  005077  172716                CLR     @PSW            ;LET THE DR11-B INTERRUPT
 1440  006076  012777  177777  172712        MOV     #-1,@DRWC       ;SET-UP FOR 1 TRANSFER
 1441  006104  052777  010101  172710        BIS     #10101,@DRST    ;MAINT, IE, AND DO TO DRST
 1442  006112  000001                        WAIT                    ;WAIT FOR INTERRUPT
 1443  006114  000240                        NOP                     ;FAKE-OUT RETURN ADDRESS CHANGING
 1444  006116  117701  172700                MOVB    @DRST,R1        ;LOWER BYTE OF DRST TO R1
 1445  006122  042701  000600                BIC     #600,R1         ;GET RID OF READY AND CYCLE BECAUSE OF MAINT MODE
 1446  006126  006201                        ASR     R1              ;MOVE IT RIGHT ONE PLACE
 1447  006130  126701  172732                CMPB    FNCONT,R1       ;CHECK AGAINST FNCONT
 1448  006134  001401                        BEQ     .+4             ;SHOULD BE EQUAL
 1449  006136  104000                        HLT                     ;FUNCTION BITS DIDN'T INCREMENT IN MAINT MODE
 1450  006140  005267  172722                INC     FNCONT          ;GET READY FOR NEXT PASS
 1451  006144  022767  000010  172714        CMP     #10,FNCONT      ;ONLY 10 BECAUSE FNCT3-1 GO TO ZERO
 1452  006152  001404                        BEQ     MFCHK           ;IF ITS EQUAL GO CHECK DATA
```

```
        006154  062767  000002  172656            ADD     #2,INBUF          ;FAKE-OUT INTA ROUTINE
        006162  000735                            BR      MFLOOP            ;DO IT AGAIN
        006164  012767  000007  172652  MFCHK:    MOV     #7,BUFLEN         ;SET UP FOR DATCHK (10 FNCT CHECKS, 7 TRANSFERS
        006172  016767  172672  172640            MOV     INBUF1,INBUF      ;RESTORE INBUF
        006200  004767  000446                    JSR     %7,DATCHK         ;CHECK DATA

                                          ;;*******************************************************************
                                          ;       TEST 61 TEST FOR 10 MAINT MODE TRANSFERS
                                          ;;*******************************************************************
        006204  104400                            SCOPE
        006206  012767  000010  172630            MOV     #10,BUFLEN        ;BUFLEN=10
        006214  016767  172624  172630            MOV     BUFLEN,WCLEN      ;PREPARE NUMBER FOR DRWC
        006222  005467  172624                    NEG     WCLEN             ;2'S COMPLEMENT OF BUFLEN
        006226  004767  000222                    JSR     %7,LODBUF         ;LOAD IN BUFFER WITH INCREMENTING PATTERN
        006232  004767  000254                    JSR     %7,CHKBFF         ;LOAD CHECK BUFFER WITH MODIFIED INCREMENTING PATTERN
        006236  016777  172610  172552            MOV     WCLEN,@DRWC       ;SET UP DRWC
        006244  016777  172572  172546            MOV     INBUF,@DRBA       ;SET UP DRBA
        006252  012777  177777  172544            MOV     #-1,@DRDB         ;MAINT AIDE
        006260  012777  006556  172544            MOV     #INTA,@DRINV      ;INTERRUPT VECTOR TO INTA
        006266  016777  172536  172532            MOV     DRINL,@DRVS       ;INTERRUPT STATUS AT PRIORITY DRINL
        006274  005077  172514                    CLR     @PSW              ;LET DR11-B INTERRUPT
        006300  012777  010101  172514            MOV     #10101,@DRST      ;MAINT, IE, AND DO TO DRST
        006306  000777                            BR      .                 ;WAIT FOR INTERRUPT
        006310  004767  000335                    JSR     %7,DATCHK

                                          ;;*******************************************************************
                                          ;       TEST 62 TEST FOR 200 NPR TRANSFERS IN MAINT MODE
                                          ;;*******************************************************************
        006314  104400                            SCOPE
        006316  012767  000200  172520            MOV     #200,BUFLEN       ;LENGTH OF BUFFER = 200
        006324  016767  172514  172520            MOV     BUFLEN,WCLEN      ;PREPARE NUMBER FOR DRWC
        006332  005467  172514                    NEG     WCLEN             ;2'S COMPLEMENT OF BUFLEN
        006336  004767  000112                    JSR     %7,LODBUF         ;LOAD INBUF WITH INCREMENTING PATTERN
        006342  004767  000144                    JSR     %7,CHKBFF         ;LOAD CHKBUF WITH MODIFIED INCREMENTED PATTERN
        006346  016777  172500  172442            MOV     WCLEN,@DRWC       ;SET UP DRWC
        006354  016777  172460  172436            MOV     INBUF,@DRBA       ;SET UP DRBA
        006362  012777  000001  172434            MOV     #1,@DRDB          ;MAINT AIDE
        006370  012777  006556  172434            MOV     #INTA,@DRINV      ;INT VECTOR TO INTA
        006376  016777  172426  172422            MOV     DRINL,@DRVS       ;INT VECTOR AT PRIORITY DRINL
        006404  005077  172404                    CLR     @PSW              ;LET THE DR11-B INTERRUPT
        006410  012777  010101  172404            MOV     #010101,@DRST     ;FOLLOWING TO DRST:  MAINT(12),IE(06),DO(00)
        006416  005267  172426                    INC     BRWAIT            ;USE A WAIT OR BR. INSTRUCTION
        006422  032767  000001  172420            BIT     #BIT0,BRWAIT      ;SEE WHICH ONE
        006430  001403                            BEQ     BRANCH            ;BIT 0 CLEAR = BR.
        006432  000001                            WAIT
        006434  000240                            NOP
        006436  000401                            BR      .+4
        006440  000777                  BRANCH:   BR      .
        006442  004767  000204                    JSR     %7,DATCHK         ;CHECK THAT CORRECT DATA WAS TRANSFERRED
        006446  104400                            SCOPE
        006450  000167  000356                    JMP     BELL              ;DO IT ALL AGAIN.
        006454  016702  172360          LODBUF:   MOV     INBUF,%2          ;MOVE STARTING ADDRESS OF INBUF TO R2
        006460  005067  172362                    CLR     LENCHK            ;CLEAR LENGTH CHECK
        006464  005022                            CLR     (%2)+             ;CLEAR STARTING ADDRESS OF INBUFF AND INC BY 2
        006466  005267  172354          LOADA:    INC     LENCHK            ;INC LENGTH CHECK BY 1
        006472  026767  172350  172344            CMP     LENCHK,BUFLEN     ;CHECK FOR DONE
```

```
006500  001403                                BEQ    LDEXIT          ;IS INBUF FILLED?
006502  016722  172340                        MOV    LENCHK,(%2)+    ;LOAD NEXT BUFFER WORD
006506  000767                                BR     LOADA           ;CONTINUE CHECKING
006510  000207                       LDEXIT:  RTS    %7              ;EXIT
006512  016702  172324               CHKBFF:  MOV    CHKBUF,%2       ;STARTING ADDRESS OF CHECK-BUFFER TO R2
006516  005067  172324                        CLR    LENCHK          ;CLEAR LENGTH CHECK
006522  005003                                CLR    %3              ;CLEAR R3
006524  010322                       CHKA:    MOV    %3,(%2)+        ;MOVE R3 TO CHKBUF ADDRESS AND INC BY 2
006526  010322                                MOV    %3,(%2)+        ;MOVE R3 TO NEXT CHKBUF ADDRESS AND INC BY 2
006530  062767  000002  172310                ADD    #2,LENCHK       ;ADD 2 TO LENGTH CHECK
006536  026767  172304  172300                CMP    LENCHK,BUFLEN   ;CHECK FOR DONE
006544  100003                                BPL    .+10            ;IS CHECK-BUFFER FILLED?
006546  062703  000002                        ADD    #2,%3           ;NEXT NUMBER FOR BUFFER
006552  000764                                BR     CHKA            ;CONTINUE FILLING
006554  000207                                RTS    %7              ;EXIT
006556  042777  000100  172236       INTA:    BIC    #BIT6,@DRST     ;CLEAR IE
006564  005777  172232                        TST    @DRST           ;CHECKING FOR ERROR
006570  100001                                BPL    .+4             ;ERROR SET?
006572  104000                                HLT                    ;ERROR BIT IS SET
006574  105777  172222                        TSTB   @DRST           ;CHECKING READY BIT
006600  100401                                BMI    .+4             ;IS READY SET
006602  104000                                HLT                    ;FALSE INTERRUPT - ERROR AND READY ARE CLEAR
006604  005777  172206                        TST    @DRWC           ;TESTI FOR DRWC=0
006610  001401                                BEQ    .+4             ;WAS IT EQUAL?
006612  104000                                HLT                    ;DRWC NOT =0
006614  016702  172224                        MOV    BUFLEN,%2       ;BUFFER LENGTH TO R2
006620  066702  172220                        ADD    BUFLEN,%2       ;NUMBER OF TRANSFERS TIMES 2
006624  066702  172210                        ADD    INBUF,%2        ;CORRECT DRBA
006630  027702  172164                        CMP    @DRBA,%2        ;CHECKING DRBA
006634  001401                                BEQ    .+4             ;IS DRBA CORRECT?
006636  104000                                HLT                    ;DRBA NOT CORRECT
006640  062716  000002                        ADD    #2,(%6)         ;RETURN ADDRESS TO RETURN ADDRESS +2
006644  004767  000042                        JSR    %7,NORMAL
006650  000002                                RTI                    ;EXIT

006652  016702  172164               DATCHK:  MOV    CHKBUF,%2       ;STARTING ADDRESS OF CHECK BUFFER TO R2
006656  016703  172156                        MOV    INBUF,%3        ;STARTING ADDRESS OF IN BUFFER TO R3
006662  005067  172150                        CLR    LENCHK          ;CLEAR LENGTH CHECK
006666  005267  172154               COMPAR:  INC    LENCHK          ;MAKE A COMPARISON
006672  022223                                CMP    (%2)+,(%3)+     ;IS THE DATA CORRECT?
006674  001401                                BEQ    .+4             ;BRANCH IF OK
006676  104000                                HLT                    ;BAD DATA
006700  026767  172142  172136                CMP    LENCHK,BUFLEN   ;SEE IF THE BUFFER HAS BEEN CHECKED
006706  001367                                BNE    COMPAR          ;BUFFER CHECKED?
006710  000207                                RTS    %7
006712  012777  001026  172112       NORMAL:  MOV    #DRVS,@DRINV    ;RESTORE DR11-B INTERRUPT VECTOR
006720  005077  172102                        CLR    @DRVS           ;RESTORE DR11-B INTERRUPT STATUS
006724  012777  000340  172062                MOV    #340,@PSW       ;RESTORE PROC TO PRIORITY LEVEL 7
006732  000207                                RTS    %7              ;EXIT
006734  012702  052525               DATOCK:  MOV    #52525,%2       ;DATO NUMBER TO R2
006740  016703  172074                        MOV    INBUF,%3        ;STARTING ADDRESS OF IN BUFFER TO R3
006744  005067  172076                        CLR    LENCHK          ;CLEAR LENGTH CHECK
006750  005267  172072               COMPRR:  INC    LENCHK          ;MAKE A COMPARISON
006754  020223                                CMP    %2,(%3)+        ;IS THE DATA CORRECT?
006756  001401                                BEQ    .+4             ;BRANCH IF OK
006760  104000                                HLT                    ;BAD DATA
```

```
006762  026767  172060  172054              CMP     LENCHK,BUFLEN   ;SEE IF THE BUFFER HAS BEEN CHECKED
006770  001367                              BNE     COMPRR          ;BUFFER CHECKED?
006772  022323                              CMP     (2)+,(3)+       ;CHECK END OF BUFFER + 1
006774  001001                              BNE     .+4             ;SEE IF TOO MANY WORDS WERE TRANSFERRED
006776  104000                              HLT                     ;TOO MANY
007000  000207                              RTS     %7              ;EXIT
007002  042777  000100  172012  ERRCHK:     BIC     #BIT6,@DRST     ;CLEAR IE
007010  005777  172006                      TST     @DRST           ;CHECKING FOR ERROR
007014  100001                              BPL     .+4             ;ERROR SET?
007016  104000                              HLT                     ;ERROR BIT IS SET
007020  105777  171776                      TSTB    @DRST           ;CHECKING READY BIT
007024  100401                              BMI     .+4             ;IS RDY SET
007026  104000                              HLT                     ;FALSE ENTRY - ERROR AND READY ARE CLEAR
007030  000207                              RTS     %7              ;EXIT

                                        ;;*******************************************************
                                        ;       BELL ON PASS COMPLETE
                                        ;;*******************************************************
007032  012737  000207  177566  BELL:       MOV     #207,@#177566
007040  105737  177564                      TSTB    @#177564
007044  100375                              BPL     .-4
007046  012702  012365                      MOV     #$DR11,%2
007052  004767  003404                      JSR     %7,TTOUT
                                        ;;*******************************************************
                                        ;       ROUTINE TO CHECK FOR TRACE TRAP TO BE RUN WITH PROGRAM
                                        ;;*******************************************************

007056  004767  002670          TRTRAP:     JSR     %7,CKSWR        ;CHECK FO CONT G
007062  032777  010000  171722              BIT     #10000,@SR      ;SHOULD WE RUN WITH TRACE TRAP
007070  001417                              BEQ     YESTR           ;YES
007072  005767  000104                      TST     YESTR1          ;NO  HAVE WE RAN WITH TRACE TRAP ON
007076  001411                              BEQ     TRPA            ;IF SO RESTORE PREVIOUS CONTENTS
007100  016767  000076  170706              MOV     YESTR1,14
007106  016767  000072  170702              MOV     YESTR2,16
007114  042777  000020  171672              BIC     #20,@PSW        ;CLEAR TRACE TRAP
007122  000167  172030          TRPA:       JMP     BEGIN           ;START OF TEST WITH TRACE OFF
007126  000000                  TRPB:       0

                                        ;;*******************************************************
                                        ;       SAVE OLD CONTENTS, SET UP FOR TRACE TRAP
                                        ;;*******************************************************
007130  016767  170660  000044  YESTR:      MOV     14,YESTR1       ;SAVE ODT PC
007136  016767  170654  000040              MOV     16,YESTR2       ;SAVE ODT STATUS
007144  012767  007206  170642              MOV     #YESRT,14       ;NEW TRAP VECTOR
007152  005067  170640                      CLR     16              ;NEW CONDITION CODES
007156  005077  171632                      CLR     @PSW
007162  005167  177740                      COM     TRPB
007166  100403                              BMI     .+10
007170  052777  000020  171616              BIS     #20,@PSW        ;SET TRACE TRAP
007176  000167  171754                      JMP     BEGIN           ;START OF TEST WITH TRACE ON

007202  000000                  YESTR1:     0                       ;STORAGE FOR ODT PC
007204  000000                  YESTR2:     0                       ;STORAG FOR ODT STATUS
007206  000002                  YESRT:      RTI                     ;RETURN TO PROGRAM FROM TRAP
007210  000000                              HALT                    ;RTI FAILED
```

```
1610
1611
1612                            ;;*****************************************************************
1613                            ;       BUS TO BUS TEST (DR11-B TO DR11-B)
1614                            ;;*****************************************************************
1615
1616
1617   000000                   R0=%0
1618   000001                   R1=%1
1619   000002                   R2=%2
1620   000003                   R3=%3
1621   000004                   R4=%4
1622   000005                   R5=%5
1623   000006                   R6=%6
1624   000007                   PC=%7
1625   000006                   SP=%6

1626   000001                   GO=1
1627   000002                   FNCT1=2          ;OUTPUT MODE
1630   000004                   FNCT2=4          ;OUTPUT DIRECTION
1631   000010                   FNCT3=10         ;OUTPUT INTER REQ
1632   000020                   XBA16=20
1633   000040                   XBA17=40
1634   000100                   IE=100
1635   000200                   READY=200
1636   000400                   CYCLE=400
1637   001000                   DSTATA=1000      ;INPUT MODE
1640   002000                   DSTATB=2000      ;INPUT DIRECTION
1641   004000                   DSTATC=4000      ;INPUT INTR REQ
1642   010000                   MAINT=10000
1643   020000                   ATTN=20000
1644   040000                   NEX=40000
1645   100000                   ERROR=100000
1646
1647
1650   007312  000004           NWRDXF: 4                ;# OF WORDS TRANSFERRED UNDER FLAG CONTROL PRIOR TO NPR
1651   007314  000000           NEXJOB: 0                ;HOLDS ADDRESS OF READY INTERRUPT ROUTINE
1652   007316  000000                   0                ;HOLDS ADDRESS OF ERROR INTERRUPT ROUTINE
1653   007320  000000           JBFLAG: 0                ;JOB FLAG
1654   007322  000000           JBCNT:  0                ;JOB COUNT
1655                            ;;*****************************************************************
1656                            ;MASTER START
1657                            ;;*****************************************************************
1660   007324  000240           MS1:    NOP
1661   007326  005067  177770          CLR     JBCNT            ;CLEAR JOB COUNT
1662   007332  012706  013566          MOV     #BLFF,R6         ;SETUP STACK
1663   007336  004767  000256          JSR     R7,SETVEC        ;SET UP INTERRUPT VECTORS
1664   007342  005067  177752          CLR     JBFLAG           ;CLEAR JOB FLAG
1665   007346  012757  007700  177740  MOV     #JOBAD,NEXJOB    ;DO JOBAD FIRST
1666   007354  012767  000000C 177734  MOV     #0,NEXJOB+2      ;NO ERROR RECOVERY
1667   007362  012777  000340  171524  MOV     #340,@PSW        ;LOCK OUT INTERRUPTS
1670   007370  012777  000100  171524  MOV     #IE,@DRST        ;SET INTERRUPT ENABLE
1671   007376  005077  171512          CLR     @PSW             ;DROP PRIOTIRY TO ZERO
1672   007402  000425                  BR      BACKGD           ;WAIT FOR JOBS IN BACKGROUND
```

```
                                        ;;*****************************************************************
                                        ;       SLAVE START
                                        ;;*****************************************************************

007302  000240                  SS1:    NOP
007304  012706  013566                  MOV     #BUFF,R6
007310  004767  000002                  JSR     R7,SETVEC       ;SET UP INTERRUPT VECTORS
007314  005067  177676                  CLR     JBFLAG
007320  012767  010122  177664          MOV     #SJOB1,NEXJOB   ;FOR READY INTERRUPT
007326  012767  010140  177660          MOV     #SJOB2,NEXJOB+2 ;FOR ERROR INTERRUPT
007334  012777  000340  171455          MOV     #340,@PSW       ;RAISE CP PRIORITY TO 7
007342  012777  000100  171450          MOV     #IE,@DRST       ;SET INTERRUPT ENABLE
007350  005077  171436                  CLR     @PSW            ;DROP CP PRIORITY TO 0 AND ENTER BACKGROUND

                                        ;;*****************************************************************
                                        ;       BACKGROUND PROGRAM; WAITS FOR JBFLAG TO SET
                                        ;;*****************************************************************

007356  005200                  BACKGD: INC     R0
007360  005201                          INC     R1
007362  005202                          INC     R2
007364  005203                          INC     R3
007366  005204                          INC     R4
007370  005205                          INC     R5
007372  020005                          CMP     R0,R5
007374  001402                          BEQ     .+6
007376  104000                          HLT
007400  000000                          HALT                    ;BACKGROUND TEST FAILED
007402  020104                          CMP     R1,R4
007404  001402                          BEQ     .+6
007406  104000                          HLT
007410  000000                          HALT                    ;BACKGROUND TEST FAILED
007412  020203                          CMP     R2,R3
007414  001402                          BEQ     .+6
007416  104000                          HLT
007420  000000                          HALT                    ;BACKGROUND TEST FAILED
007422  005767  177572                  TST     JBFLAG          ;ANY JOBS?
007426  001753                          BEQ     BACKGD          ;BRANCH IF NONE
007430  004567  000032                  JSR     R5,SAVALL       ;YES ; EXECUTE JOB WHOSE ADDRESS IS IN JBFLAG
007434  005067  177554                  CLR     NEXJOB
007440  005067  177552                  CLR     NEXJOB+2
007444  016767  177553  170362          MOV     JBFLAG,34
007452  005067  177542                  CLR     JBFLAG
007456  104400                          TRAP                    ;TRAP THROUGH JBFLAG AT 34
007460  004567  000C16                  JSR     R5,RESALL
007464  000734                          BR      BACKGD

                                    ;;*****************************************************************
                                    ;       SUBROUTINE TO PUSH ALL REGISTERS ONTO THE STACK
                                    ;;*****************************************************************

007466  010446              SAVALL: MOV     R4,-(R6)        ;R5 WAS PUSHED BY JSR
007470  010346                      MOV     R3,-(R6)
007472  010246                      MOV     R2,-(R6)
```

```
                                        MOV     R1,-(R6)
                                        MOV     R0,-(R6)
                                        JMP     @R5)            ;R5 HOLDS RETURN ADDRESS

                                ;;**************************************************************
                                ;       SUBROUTINE TO POP ALL REGISTERS OFF THE STACK
                                ;;**************************************************************

                                RESALL: TST     (R6)+
                                        MOV     @R6)+,R0
                                        MOV     (R6)+,R1
                                        MOV     (R6)+,R2
                                        MOV     @R6)+,R3
                                        MOV     @R6)+,R4
                                        RTS     R5

                                ;;**************************************************************
                                ;       ROUTINE TO SET UP INTERRUPT VECTORS
                                ;;**************************************************************

                                SETVEC: MOV     @DRINV,R0       ;R0 IS VECTOR ADDRESS
                                        MOV     #@DRINS,@R0)+   ;PUT SERVICE ADDRESS INTO VECTOR
                                        MOV     @DRINL,(R0)     ;PUT PRIORITY INTO VECTOR+2
                                        MOV     #PRINT,30       ;SET UP EMT ADDRESS
                                        MOV     @DRINL,32       ;SET UP EMT PRIORITY LEVEL
                                        CLR     34
                                        MOV     @DRINL,36       ;SET UP TRAP ADDRESS
                                        CLR     R0              ;INITIALIZE REGISTERS
                                        CLR     R1
                                        CLR     R2
                                        CLR     R3
                                        CLR     R4
                                        CLR     R5
                                        R*S     R7
                                ;;**************************************************************
                                ;       PRIMARY INTERRUPT SERVICE ROUTINE.
                                ;       SETS UP JBLFAG WITH ADDRESS OF JOB TO BE RUNSTARS
                                ;;**************************************************************

                                DRINS:  TST     JBFLAG          ;HAS THE PREVIOUS INTERRUPT BEEN SERVICED?
                                        BEQ     DRINO
                                        HLT
                                        HALT                    ;NO
                                DRINC:  BIT     #@DSTATC,@DRST  ;CHECK FOR ERROR
                                        BEQ     DRIN3           ;BRANCH IF NO ERROR
                                        TST     NEXJOB+2        ;IS THERE AN ERROR SERVICE ROUTINE?
                                        BNE     DRIN1           ;BRANCH IF THERE IS.
                                        HLT
                                        HALT                    ;ERROR INTERRUPT, NO ERROR SERVICE.
                                DRIN1:  MOV     NEXJOB+2,JBFLAG ;SET UP JOBFLAG WITH ADDRESS OF SERVICE ROUTINE
                                        RTI
                                DRIN3:  TSTB    @DRST           ;CHECK READY
                                        BMI     DRIN2           ;BRANCH IF SET
                                        HLT
                                        HALT                    ;INTERRUPT WITHOUT ERROR OR READY
                                DRIN2:  TST     NEXJOB          ;IS THERE A READY SERVICE ROUTINE
```

```
 1778  007662  001002                        BNE     .+6             ;BRANCH IF THERE IS.
 1779  007664  104000                        HLT
 1780  007666  000000                        HALT                    ;READY INTERRUPT, NO READY SERVICE
 1781  007670  016767  177322  177322        MOV     NEXJOB,JBFLAG   ;SET UP JOBFLAG WITH SERVICE ROUTINE ADDRESS
 1782  007676  000002                        RTI
 1783
 1784                                  ;;**********************************************************
 1785                                  ;       MASTER'S INTERRUPT SERVICE ROUTINES
 1786                                  ;       ROUTINE A, SEGMENT 0
 1787                                  ;       FILL BUFFER AND TRANSMIT
 1788                                  ;;**********************************************************
 1789  007700  012700  011034        JOBA0:  MOV     #LISTA,R0       ;R0 IS XMIT LIST ADDRESS
 1790  007704  012701  011046                MOV     #LISTA1,R1      ;LISTA1 WILL BE REC LIST
 1791  007710  012021                        MOV     (R0)+,(R1)+     ;START WITH BUS ADDRESSES EQUAL
 1792  007712  012002                        MOV     (R0)+,R2        ;R2 HOLDS WORD COUNT OF XMIT
 1793  007714  010211                        MOV     R2,(R1)         ;MAKE REC WORD COUNT THE SAME
 1794  007716  005402                        NEG     R2              ;MAKE WORD COUNT POSITIVE
 1795  007720  006302                        ASL     R2              ;TRANSFORM INTO BYTE COUNT
 1796  007722  060241                        ADD     R2,-(R1)        ;ADD TO REC BUS ADDRESS
 1797  007724  005010                        CLR     (R0)            ;CLEAR OFFSET IN XMIT LIST
 1798  007726  024040                        CMP     -(R0),-(R0)     ;LEAVE R0=LISTA=XMIT LIST
 1799
 1800  007730  022767  000100  177264        CMP     #100,JBCNT      ;ENOUGH PASSES FOR BELL?
 1801  007736  003010                        BGT     JOBA0A          ;BRANCH IF NOT ENOUGH
 1802  007740  105737  177564        JOBA0B: TSTB    @#177564        ;TTY READY?
 1803  007744  100375                        BPL     JOBA0B
 1804  007746  012737  000207  177566        MOV     #207,@#177566   ;RING BELL
 1805  007754  005067  177242                CLR     JBCNT           ;RESET JOB COUNT
 1806  007760  004767  000704        JOBA0A: JSR     R7,SETBUF       ;FILL UP XMIT BUFFER WITH SPECIAL BINARY COUNT
 1807  007764  012700  011034                MOV     #LISTA,R0
 1808  007770  004767  000370                JSR     R7,MXMIT        ;TRANSMIT DATA TO SLAVE
 1809  007774  012767  010012  177212        MOV     #JOBA1,NEXJOB   ;JOBA1 IS NEXT
 1810  010002  012767  000000  177206        MOV     #0,NEXJOB+2     ;NO ERROR RECOVERY
 1811  010010  000002                        RTI                     ;RETURN TO BACKGROUND VIA TRAP
 1812
 1813                                  ;;**********************************************************
 1814                                  ;       ROUTINE A, SEGMENT 1
 1815                                  ;       FLUSH A BUFFER AND RECEIVE DATA
 1816                                  ;;**********************************************************
 1817
 1818  010012  012700  011046        JOBA1:  MOV     #LISTA1,R0      ;PUT REC LIST ADDRESS INTO R0
 1819  010016  004767  000574                JSR     R7,FLUSH        ;FLUSH BUFFER
 1820  010022  012700  011046                MOV     #LISTA1,R0
 1821  010026  004767  000406                JSR     R7,MREC         ;RECEIVE DATA FROM SLAVE
 1822  010032  012767  010050  177154        MOV     #JOBA2,NEXJOB   ;JOBA2 IS NEXT
 1823  010040  012767  000000  177150        MOV     #0,NEXJOB+2     ;NO ERROR RECOVERY
 1824  010046  000002                        RTI
 1825
 1826
 1827                                  ;;**********************************************************
 1828                                  ;       ROUTINE A, SEGMENT 2
 1829                                  ;       CHECKS TRANSMITTED DATA WITH RECEIVED
 1830                                  ;;**********************************************************
 1831
 1832  010050  012700  011034        JOBA2:  MOV     #LISTA,R0       ;XMIT BUFFER LIST
 1833  010054  012701  011046                MOV     #LISTA1,R1      ;REC BUFFER LIST
 1834  010060  004767  000642                JSR     R7,BUFCHK       ;COMPARE THE TWO BUFFERS
```

```
 1834  010064  042777  000100  170730          BIC     #IE.2DRST       ;GLITCH INTERRUPT
 1835  010072  012777  000100  170722          MOV     #IE.2DRST
 1836  010100  012767  000000  177106          MOV     #JOBA0,NEXJOB   ;REPEAT JOBA0
 1837  010106  012767  000000  177102          MOV     #0,NEXJOB+2
 1838  010114  005267  177102                  INC     JBCNT           ;ADVANCE COUNT
 1839  010120  000002                          RTI
 1840
 1841                                  ;;*****************************************************************
 1842                                  ;       SLAVE'S INTERRUPT SERVICE ROUTINES
 1843                                  ;
 1844                                  ;       JOB1:  IGNORE FIRST READY INTERRUPT
 1845                                  ;;*****************************************************************
 1846
 1847  010122  012767  000000  177064  SJOB1:  MOV     #0,NEXJOB       ;NO MORE READY INTERRUPTS
 1848  010130  012767  010140  177060          MOV     #SJOB2,NEXJOB+2 ;UNTIL ATTN INTERRUPT
 1849  010136  000002                          RTI
 1850
 1851                                  ;;*****************************************************************
 1852                                  ;       JOB2:  WAIT FOR COMMAND
 1853                                  ;;*****************************************************************
 1854
 1855  010140  032777  004000  170654  SJOB2:  BIT     #DSTATC,2DRST   ;TEST FOR INTER
 1856  010146  001002                          BNE     SJOB2A
 1857  010150  104000                          HLT
 1858  010152  000000                          HALT                    ;ERROR OTHER THAN DSTATC
 1859  010154  005001                  SJOB2A: CLR     R1              ;SET UP FOR PARAMETERS
 1860  010156  016702  170640                  MOV     DRST,R2         ;R2 IS STATUS ADDRESS
 1861  010162  012703  000010                  MOV     #FNCT3,R3       ;R3 IS FUNCTION BIT 3
 1862  010166  012704  004000                  MOV     #DSTATC,R4      ;R4 IS INTERUPT BIT
 1863  010172  016705  170626                  MOV     DRDB,R5         ;R5 IS DATA BUFFER ADDRESS
 1864  010176  012700  011062                  MOV     #LISTB+2,R0     ;STORE PARAMETERS HERE STARTING WITH WORD COUNT
 1865  010202  004767  000122                  JSR     R7,HNDSHK       ;GET PARAMETERS
 1866  010206  012700  011060                  MOV     #LISTB,R0       ;R0 IS TOP OF LIST
 1867  010212  016010  000004                  MOV     4(R0),(R0)      ;MOVE OFFSET TO TOP
 1868  010216  066710  170616                  ADD     INBUF,(R0)      ;TOP OF LIST IS BUFFER START + OFFSET
 1869  010222  012077  170572                  MOV     (R0)+,2DRBA     ;SET UP BUS ADDRESS
 1870  010226  012077  170564                  MOV     (R0)+,2DRWC     ;SET UP WORD COUNT
 1871  010232  005077  170564                  CLR     2DRST           ;CLEAR ALL FUNCTION BITS
 1872  010236  032777  002000  170556          BIT     #DSTATB,2DRST   ;WHICH DIRECTION
 1873  010244  001405                          BEQ     SJOB2C          ;BRANCH IF RECIEVE (LEAVE FNCT1 CLEAR FOR DATI'S)
 1874  010246  032777  000400  170546  SJOB2B: BIT     #CYCLE,2DRST    ;WAIT FOR MASTER TO SET CYCLE
 1875  010254  001774                          BEQ     SJOB2B          ;BRANCH IF NOT SET
 1876  010256  000412                          BR      SJOB2D          ;GO DO THE COMAND
 1877  010260  012700  011060          SJOB2C: MOV     #LISTB,R0
 1878  010264  004767  000356                  JSR     R7,BLUSH        ;BLUSH THE BUFFER
 1879  010270  052777  000004  170524          BIS     #FNCT2,2DRST    ;SET FNCT2 FOR DATO'S
 1880  010276  042777  000400  170516          BIC     #CYCLE,2DRST    ;CLEAR CYCLE
 1881  010304  052777  000101  170510  SJOB2D: BIS     #IE!GO,2DRST    ;EXECUTE COMMAND AND INTERRUPT WHEN DONE
 1882  010312  012767  010122  176674          MOV     #SJOB1,NEXJOB   ;IGNORE READY INTERRUPT
 1883  010320  012767  010140  176670          MOV     #SJOB2,NEXJOB+2 ;WAIT FOR ATTN INTERRUPT
 1884  010326  000002                          RTI
 1885
 1886                                  ;;*****************************************************************
 1887                                  ;       SLAVE ROUTINE TO ACCEPT PARAMETERS FROM MASTER
 1888                                  ;;*****************************************************************
 1889
```

```
                                                HNDSHK: MOV     (R5)+,R5        ;ECHO PARAMETER
                                                        MOV     (R5)+,(R0)+     ;STORE PARAMETER
                                                        BIS     R3,(R2)         ;REPLY WITH FNCT3
                                                HNDSH1: BIT     R4,(R2)         ;WAIT FOR ATTN TO DROP
                                                        BNE     HNDSH1
                                                        BIC     R3,(R2)         ;DROP FNCT3
                                                        INC     R1              ;CHECK NUMBER
                                                        CMP     R1,NWRDXF
                                                        BLT     HNDSH2          ;BRANCH IF NOT DONE YET
                                                        RTS     R7
                                                HNDSH2: BIT     R4,(R2)         ;WAIT FOR NEXT WORD
                                                        BEQ     HNDSH2          ;BRANCH IF ATTN CLEAR
                                                        BR      HNDSHK          ;GET ANOTHER PARAMETER

                                                ;;**************************************************************
                                                ;       MASTER TRANSMIT ROUTINE
                                                ;       ENTER WITH TRANSFER LIST IN R0
                                                ;;**************************************************************

010364  005777  170432                        MXMIT:  TST     @DRST           ;MAKE SURE ERROR IS CLEAR
010370  100002                                        BPL     MXMIT1          ;AND READY IS SET
010372  104000                                        HLT
010374  000000                                        HALT                    ;ERROR IS SET
010376  105777  170420                        MXMIT1: TSTB    @DRST
010402  100402                                        BMI     MXMIT2
010404  104000                                        HLT
010406  000000                                        HALT                    ;READY NOT SET
010410  012777  000000  170404                MXMIT2: MOV     #0,@DRST        ;SET UP FUNCTION FOR DATI'S
010416  004767  000102                              JSR     R7,PRMXFR       ;TRANSFER PARAMETERS OF LIST WHOSE ADDRESS IS IN R0
010422  042777  000400  170372                      BIC     #CYCLE,@DRST    ;MAKE SURE CYCLE IS CLEAR
010430  052777  000101  170364                      BIS     #IE!GO,@DRST    ;EXECUTE COMMAND AND INTERRUPT WHEN DONE
010436  000207                                      RTS     R7

                                                ;;**************************************************************
                                                ;       MASTER RECEIVE ROUTINE
                                                ;       ENTER WITH TRANSFER LIST IN R0
                                                ;;**************************************************************

010440  005777  170356                        MREC:   TST     @DRST           ;MAKE SURE ERROR IS CLEAR
010444  100002                                        BPL     MREC1           ;AND READY SET.
010446  104000                                        HLT
010450  000000                                        HALT                    ;ERROR SET
010452  105777  170344                        MREC1:  TSTB    @DRST
010456  100402                                        BMI     MREC2
010460  104000                                        HLT
010462  000000                                        HALT                    ;READY CLEAR
010464  012777  000004  170330                MREC2:  MOV     #FNCT2,@DRST    ;SET UP FUNCTION FOR DATO'S
010472  004767  000026                              JSR     R7,PRMXFR       ;TRANSFER PARAMETERS OF LIST WHOSE ADDRESS IS IN R0
010476  032777  002000  170316                MREC3:  BIT     #DSTATB,@DRST   ;WAIT FOR SLAVE TO CLEAR DIRECTION
010504  001374                                      BNE     MREC3           ;BRANCH IF SET
010506  042777  000400  170306                      BIC     #CYCLE,@DRST    ;CLEAR CYCLE
010514  052777  000101  170300                      BIS     #IE!GO,@DRST    ;EXECUTE COMMAND AND INTERRUPT WHEN DONE.
010522  000207                                      RTS     R7

                                                ;;**************************************************************
                                                ;       ROUTINE TO TRANSFER AND CHECK PARAMETERS UNDER FLAG CONTROL
                                                ;       ENTER WITH R0 POINTING TO TRANSFER LIST
```

```
                                        ;;*****************************************************************
                                        PRMXFR: MOV     (R0)+,@DRBA     ;FIRST WORD IN LIST IS ADDRESS
                                                MOV     (R0),@DRWC      ;SECOND WORD IN LIST IS WORD COUNT
                                                CLR     R1              ;R1 COUNTS PARAMETERS TRANSFERRED
                                                MOV     DRST,R2         ;R2 IS THE STATUS ADDRESS
                                                MOV     #FNCT3,R3       ;R3 USED FOR FUNCTION BIT 3
                                                MOV     #DSTATC,R4      ;R4 USED FOR INTERUPT BIT
                                                MOV     DRDB,R5         ;R5 IS THE DATA BUFFER ADDRESS
                                        PRMXF1: MOV     (R0),(R5)       ;SET UP DRDB WITH PARAMETER
                                                BIS     R3,(R2)         ;CALL SLAVE'S ATTN
                                        PRMXF2: BIT     R4,(R2)         ;WAIT FOR REPLY
                                                BEQ     PRMXF2          ;BRANCH IF ATTN CLEAR
                                                CMP     (R0)+,(R5)      ;COMPARE PARAMETER SENT WITH SLAVE'S ECHO
                                                BEQ     PRMXF3          ;BRANCH IF EQUAL
                                                HLT
                                                HALT                    ;PARAMETER DID NOT ECHO
                                        PRMXF3: BIC     R3,(R2)         ;DROP SLAVE'S ATTN
                                        PRMXF4: BIT     R4,(R2)         ;WAIT FOR REPLY
                                                BNE     PRMXF4          ;BRANCH IF ATTN SET
                                                INC     R1              ;ADVANCE PARAMETER COUNT
                                                CMP     R1,NUMDXF       ;ALL PARAMETER XFERRED?
                                                BLT     PRMXF1          ;BRANCH IF NOT DONE
                                                RTS     R7              ;RETURN WHEN ALL PARAMETERS TRANSFERRED AND CHECKED.

                                        ;;*****************************************************************
                                        ;       ROUTINE TO CLEAR BUFFER
                                        ;       ENTER WITH R0 POINTING TO TRANSFER LIST
                                        ;;*****************************************************************

                                        FLUSH:  CLR     R5              ;SET R5 TO ZIP
                                                JSR     R7,BSETUP       ;SET UP REGISTERS
                                        FLUSH1: JSR     R7,BUFPUT       ;STORE ITEM IN BUFFER
                                                BLT     FLUSH1
                                                BR      BUFOUT          ;STOP WHEN BUFFER FULL

                                        BUFPUT: MOV     R5,(R1)+        ;PUT R5 INTO BUFFER
                                                ADD     R5,R3           ;INCLUDE IN CHECKSUM
                                                ADC     R4
                                                INC     R2              ;ADVANCE WORD COUNT
                                                RTS     R7              ;RETURN WITH STATUS SET

                                        ;;*****************************************************************
                                        ;       ROUTINE TO FILL BUFFER WITH ALL ONE'S
                                        ;       ENTER WITH R0 POINTING TO TRANSFER LIST
                                        ;;*****************************************************************
                                        BLUSH:  MOV     #-1,R5          ;SET R5 TO ALL ONE'S
                                                BR      FLUSH+2

                                        ;REGISTER SETUP ROUTINE
                                        BSETUP: MOV     (R0)+,R1        ;R1 IS BUS ADDRESS
                                                MOV     (R0)+,R2        ;R2 IS WORD COUNT
                                                TST     (R0)+           ;SKIP OVER OFFSET
                                                CLR     R3              ;CLEAR LOW CHECKSUM
                                                CLR     R4              ;CLEAR HIGH CHECKSUM
                                                RTS     R7              ;RETURN
```

```
                                        ;;*****************************************************************
                                        ;       ROUTINE TO FILL BUFFER WITH SPECIAL BINARY COUNT
                                        ;       ENTER WITH R0 POINTING TO TRANSFER LIST
                                        ;;*****************************************************************
010670  004767  177760          SETBUF: JSR     R7,BSETUP       ;SET UP REGISTERS
010674  005005                  CLR     R5
010676  004767  177732          SETBF1: JSR     R7,BUFPUT       ;NUMBER TO BUFFER
010702  002006                  BGE     BUFOUT
010704  005105                  COM     R5
010706  004767  177722          JSR     R7,BUFPUT       ;COMPLEMENT OF NUMBER TO BUFFER
010712  002002                  BGE     BUFOUT
010714  005405                  NEG     R5              ;INCREMENT NUMBER
010716  000767                  BR      SETBF1
010720  010320          BUFOUT: MOV     R3,(R0)+        ;MOVE LOW CHECK TO LIST
010722  010420                  MOV     R4,(R0)+        ;MOVE HIGH CHECK TO LIST
010724  000207                  RTS     R7              ;RETURN

                                        ;;*****************************************************************
                                        ;       ROUTINE TO COMPARE TWO BUFFERS
                                        ;       BUFFER LIST #1 IS IN R1
                                        ;       BUFFER LIST #2 IS IN R0.  #2 CHECKED FOR PROPER CHECKSUM
                                        ;;*****************************************************************

010726  010046          BUFCHK: MOV     R0,-(R6)        ;SAVE LIST#2 ON STACK
010730  010146                  MOV     R1,-(R6)        ;SAVE LIST#1 ON STACK
010732  004767  000036          JSR     R7,CHKSUM       ;VERIFY INTEGRITY OF CHECK BUFFER (#2)
010736  012600                  MOV     (R6)+,R0        ;RECOVER 1ST LIST POINTER
010740  012001                  MOV     (R0)+,R1        ;BUS ADDRESS #1
010742  012002                  MOV     (R0)+,R2        ;WORD COUNT #1
010744  012600                  MOV     (R6)+,R0        ;RECOVER 2ND LIST POINTER
010746  012003                  MOV     (R0)+,R3        ;BUS ADDRESS #2
010750  020220                  CMP     R2,(R0)+        ;COMPARE WORD COUNTS
010752  003402                  BLE     BUFCK1          ;BRANCH IF EQUAL OR LESS THAN
010754  104000                  HLT                     ;BUFFER 2 IS LONGER THAN 1
010756  000000                  HALT

010760  022123          BUFCK1: CMP     (R1)+,(R3)+     ;COMPARE BUFFERS
010762  001401                  BEQ     BUFCK2
010764  104000                  HLT                     ;DATA ERROR
010766  005202          BUFCK2: INC     R2              ;ADVANCE COUNT
010770  002773                  BLT     BUFCK1          ;BRANCH IF NOT DONE
010772  000207                  RTS     R7              ;RETURN; STACK IS CLEAR

                                        ;;*****************************************************************
                                        ;       ROUTINE TO CHECK SUM OF BUFFER
                                        ;       ENTER WITH R0 POINTING TO TRANSFER LIST
                                        ;;*****************************************************************
010774  004767  177654          CHKSUM: JSR     R7,BSETUP       ;SET UP REGISTERS
011000  004767  000016          CHKSM1: JSR     R7,GETBUF       ;GET ITEM FROM BUFFER
011004  002775                  BLT     CHKSM1          ;BRANCH IF NOT DONE
011006  020320                  CMP     R3,(R0)+        ;COMPARE LOW ORDER CHECKS
011010  001003                  BNE     CHKSM2
011012  020420                  CMP     R4,(R0)+        ;COMPARE HIGH ORDER CHECKS
011014  001001                  BNE     CHKSM2
```

```
 2058  011016  000207                    RTS     R7              ;RETURN IF CHECKSUM OK.
 2059  011020  104000              CHKSM2: HLT                     ;ORIGINAL BUFFER CHECKSUM DOES NOT AGREE WITH PRESENT
 2060
 2061  011022  012105              GETBUF: MOV     (R1)+,R5        ;GET ITEM OUT OF BUFFER
 2062  011024  060503                      ADD     R5,R3           ;ADD TO CHECKSUM
 2063  011026  005504                      ADC     R4
 2064  011030  005202                      INC     R2              ;ADVANCE WORD COUNT
 2065  011032  000207                      RTS     R7
 2066
 2067  011034  013570              LISTA:  XINBUF                  ;START OF XMIT BUFFER
 2068  011036  177605                      -123.                   ;WORD COUNT
 2069  011040  000000                      0                       ;OFFSET
 2070  011042  000000                      0                       ;CHECKSUM LOW
 2071  011044  000000                      0                       ;CHECKSUM HIGH
 2072
 2073  011046  000000              LISTA1: 0                       ;START OF REC BUFFER
 2074  011050  000000                      0                       ;WORD COUNT
 2075  011052  000000                      0                       ;OFFSET
 2076  011054  000000                      0                       ;CHECKSUM LOW
 2077  011056  000000                      0                       ;CHECKSUM HIGH
 2078
 2079  011060  013570              LISTB:  XINBUF                  ;SLAVE'S ECHO BUFFER
 2080  011062  000000                      0
 2081  011064  000000                      0
 2082  011066  000000                      0
 2083  011070  000000                      0
 2084
 2085
 2086                              ;;*****************************************************************
 2087                              ;       ENTERED WITH SYSTEM TRAP CALL(HLT)
 2088                              ;       PRINT OUT THE ERROR PC AND STATUS REGISTER
 2089                              ;;*****************************************************************
 2090
 2091
 2092  011072  004767  000654      PRINT:  JSR     %7,CKSWR
 2093  011076  037727  167710  020000        BIT     @SR,#20000      ;TEST FOR INHIBIT PRINT OU
 2094  011104  001401                      BEQ     .+4             ;BRANCH TO PRINT
 2095  011106  000002                      RTI                     ;INHIBIT, RETURN TO MAIN STREAM
 2096  011110  012667  000172              MOV     (6)+,SAVPC      ;PC OF FAILING ROUTINE
 2097  011114  012667  000170              MOV     (6)+,SAVCC      ;CC OF ERROR CONDITION
 2098  011120  024646                      CMP     -(6),-(6)       ;REPOSITION THE STACK
 2099  011122  012777  000215  167734      MOV     #215,@TPB       ;CR
 2100  011130  105777  167726              TSTB    @TPS
 2101  011134  100375                      BPL     .-4
 2102  011136  012777  000212  167720      MOV     #212,@TPB       ;LINE FEED
 2103  011144  105777  167712              TSTB    @TPS
 2104  011150  100375                      BPL     .-4
 2105  011152  010267  000122              MOV     %2,SAVR2        ;SAVE R2
 2106  011156  010367  000120              MOV     %3,SAVR3        ;SAVE R3
 2107  011162  010467  000116              MOV     %4,SAVR4        ;SAVE R4
 2108  011166  016702  000114              MOV     SAVPC,%2
 2109  011172  004767  000114              JSR     %7,PRTAB        ;PRINT OCTAL NUMBER
 2110  011176  012777  000240  167660      MOV     #240,@TPB
 2111  011204  105777  167652              TSTB    @TPS            ;SPACE BETWEEN WORDS
 2112  011210  100375                      BPL     .-4
 2113  011212  016702  000072              MOV     SAVCC,%2
```

```
2114  011216  004767  000070                    JSR     %7,PRTAB        ;PRINT OCTAL NUMBER
2115  011222  012777  000240  167634            MOV     #240,@TPB       ;PRINT SPACE
2116  011230  105777  167626                    TSTB    @TPS            ;PRINTER DONE
2117  011234  100375                            BPL     .-4             ;BRANCH WHEN NOT DONE
2118  011236  017702  167560                    MOV     @DRST,%2        ;GET DR11B STATUS
2119  011242  004767  000044                    JSR     %7,PRTAB        ;PRINT OCTAL NUMBER
2120  011246  016702  000026                    MOV     SAVR2,%2
2121  011252  016703  000024                    MOV     SAVR3,%3
2122  011256  016704  000022                    MOV     SAVR4,%4
2123  011262  004767  000464                    JSR     %7,CKSWR
2124  011266  005777  167520                    TST     @SR             ;CHECK SR FOR HALT SWITCH
2125  011272  100001                            BPL     .+4
2126  011274  000000                            HALT                    ;HALT ON ERROR UP
2127  011276  000002                            RTI                     ;RETURN TO MAINLINE
2128  011300  000000                SAVR2:      0
2129  011302  000000                SAVR3:      0
2130  011304  000000                SAVR4:      0
2131  011306  000000                SAVPC:      0
2132  011310  000000                SAVCC:      0
2133
2134  011312  005067  000260        PRTAB:      CLR     BINCT
2135  011316  005067  000252                    CLR     WGTCT
2136  011322  012704  011602                    MOV     #LIST,%4        ;GET LIST ADDRESS
2137  011326  142777  000177  167526            BICB    #177,@TPS       ;CLR INT FLAG
2138  011334  012767  000005  000236            MOV     #5,ASCNT
2139  011342  012767  000007  000220            MOV     #7,SEVEN
2140  011350  012767  000001  000214            MOV     #1,DECML
2141  011356  105777  167500          WAIT1:    TSTB    @TPS
2142  011362  100375                            BPL     WAIT1
2143  011364  005702                            TST     %2
2144  011366  100404                            BMI     MINUS           ;NEG SIGN PRINT 1
2145  011370  012777  000260  167466            MOV     #260,@TPB       ;POS SIGN PRINT 0
2146  011376  000403                            BR      START
2147  011400  012777  000261  167456  MINUS:    MOV     #261,@TPB
2148  011406  016703  000156          START:    MOV     SEVEN,%3        ;PUT MASK IN R3
2149  011412  010267  000150                    MOV     %2,TOODLE       ;GET READY TO DOODLE NUMBER IN TOODLE
2150  011416  005167  000144                    COM     TOODLE          ;COMPENSATES FOR COMPLEMENT DURING BIC
2151  011422  046703  000140                    BIC     TOODLE,%3       ;AND IN OCTAL CHARACTER
2152  011426  001410                            BEQ     WRTOC           ;ZERO, WRITE 0 IN LIST
2153  011430  066767  000136  000136  MKNUM:    ADD     DECML,WGTCT     ;COUNT UP TO
2154  011436  005267  000134                    INC     BINCT           ;AND RECORD
2155  011442  026703  000126                    CMP     WGTCT,%3        ;SAME BINARY WEIGHT
2156  011446  001370                            BNE     MKNUM           ;KEEP COUNTN
2157  011450  062767  000260  000120  WRTOC:    ADD     #260,BINCT      ;ADD ASCII PREFIX
2158  011456  016724  000114                    MOV     BINCT,(4)+      ;WRITE ASCII CHAR IN LIST
2159  011462  066767  000102  000102            ADD     SEVEN,DECML     ;EXPAND BINARY WEIGHT
2160  011470  005067  000100                    CLR     WGTCT
2161  011474  005067  000076                    CLR     BINCT
2162  011500  005367  000074                    DEC     ASCNT
2163  011504  001410                            BEQ     XLIST           ;5 CHAR IN LIST
2164  011506  012703  000003                    MOV     #3,%3           ;SET X3 FOR ADD LOOP
2165  011512  066767  000052  000050  MORDD:    ADD     SEVEN,SEVEN     ;MAKING SEVENTY BY SEVEN
2166  011520  005303                            DEC     %3
2167  011522  001373                            BNE     MORDD
2168  011524  000730                            BR      START           ;NX SEVEN SET GET NX OCTAL
2169  011526  012767  000005  000044  XLIST:    MOV     #5,ASCNT        ;SEND 5 CHAR TO TTY
```

```
011534  [illegible]  167322          WAIT2:  TSTB    @TPS
[illegible]  [illegible]                       BPL     WAIT2
[illegible]  [illegible]  167316                  MOV     -(4),@TPB
[illegible]  005367  000026                  DEC     ASCNT
[illegible]  [illegible]                       BEQ     HOFHM           ;FINISH PRINTING GET NXT NUM
[illegible]  [illegible]                       BR      WAIT2
[illegible]  [illegible]  167300          HOFHM:  TSTB @TPS
[illegible]  [illegible]                       BPL     .-4
[illegible]  [illegible]                       RTS     %7              ;HEAD FOR HOME
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                  [illegible]: 0
[illegible]  [illegible]                          0
[illegible]  [illegible]                          0
[illegible]  [illegible]                          0
[illegible]  [illegible]                          0
011612  000000                          0

                                ;;*****************************************************************
                                ;       SCOPE LOOP ROUTINE ENTERED BY USER TRAP
                                ;;*****************************************************************

011614  004767  000132          SCOPEA: JSR     %7,CKSWR
011620  032777  040000  167164          BIT     #40000,@SR
011626  001003                          BNE     SCOPEB          ;SCOPE, BIT IS A ONE
011630  011667  000100                  MOV     @%6,RETURN      ;NO - SAVE PC FOR NEXT TIME
011634  000002                          RTI                     ;RETURN IN SEQUENCE
011636  022626                  SCOPEB: CMP     (6)+,%6         ;REPOSITION THE STACK
011640  012677  167150                  MOV     (6)+,@PSW
011644  000177  000064                  JMP     @RETURN         ;SCOPE RETURN

                                ;;*****************************************************************
                                ;       SCOPE OR AND ITERATION LOOP FOR EACH TEST 4000 TIMES
                                ;;*****************************************************************

011650  004767  000076          SCOPEC: JSR     %7,CKSWR
011654  032777  040000  167130          BIT     #40000,@SR      ;TEST SR FOR SCOPE
011662  001365                          BNE     SCOPEB          ;YES SCOPE
011664  004767  000062                  JSR     %7,CKSWR
011670  032777  004000  167114          BIT     #4000,@SR       ;NO - TEST FOR ITERATION
011676  001007                          BNE     SCOPEG          ;INHIBIT ITERATION
011700  026767  000026  000022          CMP     SCOPEF,ICOUNT
011706  001403                          BEQ     SCOPEG          ;EXIT - DONE
011710  005267  000016                  INC     SCOPEF          ;INCREMENT COUNT
011714  000750                          BR      SCOPEB          ;LOOP SOME MORE
011716  005067  000010          SCOPEG: CLR     SCOPEF          ;CLEAR COUNT
011722  011667  000006                  MOV     @%6,RETURN      ;SAVE SCOPE RETURN POINTER
011726  000002                          RTI                     ;RETURN INLINE-NEXT TEST
011730  004000                  ICOUNT: 4000
011732  000000                  SCOPEF: 0                       ;COUNT LOCATION FOR ITERATION LOOP
011734  001156                  RETURN: BEGIN                   ;ADDRESS OF LAST TEST
                                        .EVEN
```

C04


```
011736  000167  166236                      JMP     200

                                    ;;*****************************************************
                                    ;       CHECK SWITCH REGISTER ROUTINE. CHECKS FOR ^G TO ALLOW CHANGING
                                    ;       OF LOC. 176.
                                    ;;*****************************************************
011742  000000                              TEMPST: .WORD   0
011744  000000                              COUNT:  .WORD   0
011746  000000                              RDSW:   .WORD   0
011750  000000                              TIB:    .WORD   0

011752  022767  000176  167032  CKSWR:      CMP     #SWREG,SR          ;SOFTWARE SWITCH REGISTER PRESENT
011760  001133                              BNE     OUT
011762  105777  167070                      TSTB    @TKS               ;YES,WAIT FOR
011766  100130                              BPL     OUT                ;READY, GET CHARACTER
011770  017767  167064  177752              MOV     @TKB,TIB           ;AND STRIP OFF
011776  042767  177600  177744              BIC     #177600,TIB        ;THE GARBAGE
012004  022767  000007  177736              CMP     #7,TIB             ;IS IT A (^G)
012012  001116                              BNE     OUT
012014  012702  012324                      MOV     #$CNTG,%2
012020  004767  000436                      JSR     PC,TTOUT
012024  012702  012336          CNTLU:      MOV     #$MSWR,%2
012030  004767  000426                      JSR     PC,TTOUT
012034  017702  166752                      MOV     @SR,%2
012040  004767  177246                      JSR     %7,PRTAB
012044  012702  012346                      MOV     #$MNEW,%2
012050  004767  000406                      JSR     PC,TTOUT
012054  005037  011742                      CLR     @#TEMPST
012060  005067  177656          $READ:      CLR     TEMPST
012064  012767  000007  177652              MOV     #7,COUNT
012072  004767  000154          1$:         JSR     PC,TTIN            ;GO  READ A CHARACTER
012076  042767  177600  177644              BIC     #177600,TIB        ;STRIP OFF GARBAGE
012104  122767  000025  177636              CMPB    #25,TIB            ;IS IT A ^U?
012112  001001                              BNE     2$                 ;BRANCH IF NOT
012114  000743                  3$:         BR CNTLU                   ;START OVER
012116  122767  000015  177624  2$:         CMPB    #15,TIB            ;IS IT A (CR)?
012124  001011                              BNE     4$                 ;BRANCH IF NOT
012126  012702  012332                      MOV     #$CRLF,%2
012132  004767  000324                      JSR     %7,TTOUT
012136  022767  000007  177600              CMP     #7,COUNT           ;WAS IT FIRST CHARACTER
012144  001036                              BNE     7$                 ;CHANGE SWR IF NOT FIRST ONE
012146  000440                  8$:         BR      OUT                ;GET OUT
012150  122767  000060  177572  4$:         CMPB    #60,TIB
012156  003004                              BGT     5$
012160  122767  000067  177562              CMPB    #67,TIB
012166  002005                              BGE     6$
012170  012702  012357          5$:         MOV     #$QUEST,%2
012174  004767  000262                      JSR     PC,TTOUT
012200  000745                              BR      3$                 ;START OVER IF NOT LEGAL CHARACTER
012202  006367  177534          6$:         ASL     TEMPST
012206  006367  177530                      ASL     TEMPST
012212  006367  177524                      ASL     TEMPST
012216  142767  000060  177524              BICB    #60,TIB            ;GET NITTY-GRITTY
012224  156767  177520  177510              BISB    TIB,TEMPST
012232  005367  177506                      DEC     COUNT              ;ONLY WANT 6   DIGITS
```

D04


```
012236  001754                           BEQ     5$
012240  000714                           BR      1$
012242  016767  177474  166542   7$:     MOV     TEMPST,@SR          ;CHANGE SWITCH REGISTER CONTENTS
012250  000207                   OUT:    RTS     %7                  ;RETURN TO PROGRAM
                                 ;;*************************************************************
                                 ;       TTY READ SUBROUTINE**********
                                 ;;*************************************************************



012252  005077  166600           TTIN:   CLR     @TKS
012256  005077  166576                   CLR     @TKB
012262  005067  177462                   CLR     TIB
012266  005277  166564                   INC     @TKS
012272  105777  166560           TTIN1:  TSTB    @TKS
012276  100375                           BPL     TTIN1
012300  017767  166554  177442           MOV     @TKB,TIB
012306  105777  166550           TTIN2:  TSTB    @TPS
012312  100375                           BPL     TTIN2
012314  116777  177430  166542           MOVB    TIB,@TPB

012322  000207                           RTS     %7
012324  057137  020107  000046   $CNTG:  .ASCIZ  '←↑G $'
012332  020137  000046           $CRLF:  .ASCIZ  '← $'
012336  051537  051127  020075   $MSWR:  .ASCIZ  '←SWR= $'
012344  000046
012346  020040  042516  036527   $MNEW:  .ASCIZ  '  NEW= $'
012354  023040     000
012357     137  020077  023137   $QUEST: .ASCIZ  '←? ←$'
012364     000
012365     137  051104  030461   $DR11:  .ASCIZ  '←DR11B OK   $'
012372  020102  045517  020040
012400  020044     000
012403     137  046440  044501   $MAIN:  .ASCIZ  '← MAINDEC-11-DZDRB-D ←$'
012410  042116  041505  030455
012416  026461  055104  051104
012424  026502  020104  023137
012432     000
012433     137  042040  030522   $HEAD:  .ASCIZ  '← DR11B LOGIC TEST $'
012440  041061  046040  043517
012446  041511  052040  051505
012454  020124  000046
                                         .EVEN

012460  000000                   CFL:    0                           ;FIRST CHAR FLAG


                                 ;;*************************************************************
                                 ;       TTY ASCII OUTPUT ROUTINE
                                 ;;*************************************************************


012462  105712                   TTOUT:  TSTB    (2)                 ;CHECK FOR NULL CHARACTER
012464  001403                           BEQ     1$                  ;IF NOT, TYPE THE CHARACTER
012466  122712  000046                   CMPB    #'$,(2)             ;CHECK FOR TERMINATOR
012472  001005                           BNE     .EMPTY
```

```
012474  042777  000100  166360  1$:      BIC     #100,@TPS
012502  005002                           CLR     %2                  ;CLEAR POINTER TO CHARACTER
012504  000207                           RTS     %7                  ;RETURN
012506  122712  000137          .EMPTY:  CMPB    #'↑,(2)             ;CRLF CHAR?
012512  001411                           BEQ     .RET
012514  122712  000041                   CMPB    #'!,(2)             ;CHECK FOR RETURN TERMINATOR
012520  001414                           BEQ     .REST
012522  105737  166334          1$:      TSTB    @TPS
012526  100375                           BPL     1$
012530  112237  166330                   MOVB    (2)+,@TPB           ;TYPE CHARACTER
012534  000752                           BR      TTOUT
012536  005202                  .RET:    INC     %2
012540  010267  000020                   MOV     %2,.SAV             ;SET UP NEW POINTER
012544  012702  012560                   MOV     #.RETR,%2
012550  000767                           BR      .RET-6
012552  016702  000006          .REST:   MOV     .SAV,%2
012556  000741                           BR      TTOUT

012560     015     012     041  .RETR:   .BYTE   15,12,'!
        012564                           .EVEN
012564  000000                  .SAV:    0
        013566                           .=.+1000
013566  000000                  BUFF:    0                           ;FOR STACK POINTER 100 LOCATIONS
013570  013570                  XINBUF:  .
        014572                           .=.+1000
014572  014572                  XCHKBU:  .
        000001                           .END
```

F04


```
ASSNT    011620        2139*  2162*  2169*  2173*  2184#
ATEN   = XXXXXX        1641#
BACKGD   007356        1665   1696#  1705   1713
BADCNT   004736        1208   1214#
BADFCK   004556        1192   1199#
BEGIN    001156         551    553#   554   1590   1604   2224
BEL      007732        1492   1572#
BINIT    011576        2134*  2154*  2157*  2158   2161*  2163#
BIT0   = 000001         465#  1303   1379   1406   1484
BIT1   = 000002         466#   639    640    643    644    753    779    967
BIT10  = 002000         475#   975
BIT11  = 004000         476#   994
BIT12  = 010000         477#   738    739    742    743    774    800
BIT13  = 020000         478#   997
BIT14  = 040000         479#  1000   1002   1003   1168   1195   1221
BIT15  = 100000         480#
BIT2   = 000004         467#   656    657    660    661    756    781    975
BIT3   = 000010         468#   673    674    677    679    759    784    983
BIT4   = 000020         469#   685    687    690    691    703    704    762    787
BIT5   = 000040         470#   699    700    765    790
BIT6   = 000100         471#   712    713    716    717    768    794    893    895    912    914    932    934
                        951   1513   1560
BIT7   = 000200         472#   885    899    904    909    924    928    943    947   1291
BIT8   = 000400         473#   725    726    729    730    771    797
BIT9   = 001000         474#   968
BLUSH    010646        1878   1992#
BRANCH   006440        1485   1489#
BRWAIT   001050         530#  1378*  1379   1405*  1406   1483*  1484
BSETJP   010654        1977   1996#  2008   2051
BUFCHK   010726        1833   2027#
BUFCK1   010760        2036   2040#  2044
BUFCK2   010766        2041   2043#
BUFF     012566         496   1657   1673   2360#
BUFLEN   001044         528#  1076*  1078*  1079   1097*  1099*  1100   1116*  1118   1135*  1137*  1138   1367*
                       1369   1393*  1395*  1396   1418*  1424*  1444*  1452*  1453   1471*  1472   1497   1509
                       1523   1524   1540   1554
BUFOUT   010720        1980   2011   2014   2017#
BUFPUT   010634        1979   1992#  2010   2013
BUSERR=  000004         482#   563*   574*
CHKA     006524        1505#  1511
CHKBFF   006512        1421   1456   1475   1502#
CHKBUF   001042         527#  1502   1533
CHKSM1   011000        2052#  2053
CHKSM2   011020        2055   2057   2059#
CHKSUM   010774        2029   2051#
CKSWR    011752         585    601    817    887    906    926    945   1228   1241   1289   1582   2092   2123
                       2195   2209   2212   2238#
CNTLU    012024         552   2248#  2261
COMPAR   006666        1536#  1541
COMPRR   006750        1550#  1555
COUNT    011744        2234#  2256*  2266   2281*
CYCLE  = 000400        1636#  1874   1880   1919   1939
DATCHK   006652        1446   1465   1490   1533#
DATINB   005634        1380   1394#
DATOCK   006734        1110   1148   1412   1547#
DATONB   005774        1407   1411#
```

G04

```
MAINDEC-11 DZDRB-D      MACY11 27(732)  20-SEP-76  10:15  PAGE 47
DZDRB.P11       CROSS REFERENCE TABLE -- USER SYMBOLS

DECML    011572        2140*   2153    2159*   2191*
DIDMEM   001036         525*   1155    1337
DONE     005342        1325    1330*
DRBA     001020         518*    567*    600*    603     626*    627     842*    843     846*    847     870*    871     874*
                       1006*   1011*   1027    1043*   1059    1082*   1103*   1121*   1141*   1155*   1181*   1202*   1207*
                       1303    1315*   1337*   1348*   1372*   1399*   1419*   1458*   1477*   1526    1869*   1948*
DRDB     001024         520*    571*   1012*   1030    1045*   1083*   1089    1104*   1122*   1128    1142*   1240*   1243
                       1252*   1253    1261*   1262    1265*   1266    1276*   1277    1280*   1373*   1395    1400*   1459*
                       1478*   1863    1954
DRINL    001030         522*   1047    1085    1106    1124    1144    1157    1183    1209    1375    1402    1427    1461
                       1480    1744    1746    1748
DRINS    007630        1743    1751*
DRINV    001032         523*    892*    911*    931*    950*    956*   1014*   1046*   1084*   1105*   1123*   1143*   1156*
                       1192*   1209*   1338*   1347*   1374*   1401*   1426*   1460*   1479*   1543*   1742
DRIN0    007612        1762    1765*
DRIN1    007634        1768    1771*
DRIN2    007656        1774    1777*
DRIN3    007644        1766    1773*
DRST     001022         519*    569*    639*    640     643*    644     656*    657     660*    661     673*    674     677*
                        678     695*    697     690*    691     699*    700     703*    704     712*    713     716*    717
                        725*    726     729*    730     738*    739     742*    743     751*    753     756     759     762
                        765     768     771     774     777*    778     781     784     787     790     794     797     800
                        810*    812     819     885     889     893*    895*    904     908     912*    914*    924     929
                        932*    934*    943     947     951*    954*    963*    964     967*    969     971     974*    975*
                        976     979     982*    983*    984     987     990*    995     997    1000    1002*   1003    1007
                       1017*   1036*   1049*   1069*   1087*   1108*   1126*   1146*   1159*   1162    1165    1168    1185*
                       1189    1192    1195*   1196    1199    1203    1211*   1215    1218    1221*   1222    1225    1230
                       1291    1294    1316    1319*   1320    1324    1341*   1344    1349*   1351*   1356    1377*   1404*
                       1422*   1430*   1433    1463*   1482*   1513*   1514    1517    1560*   1561    1564    1663*   1679*
                       1765    1773    1834*   1835*   1855    1860    1871*   1872    1874    1879*   1880*   1881*   1909
                       1913    1917*   1919*   1920    1927    1931    1935*   1937    1939*   1940*   1951    2118
DRVS     001026         521*   1015*   1047*   1095*   1106*   1124*   1144*   1157*   1183*   1209*   1339*   1375*   1402*
                       1427*   1461*   1480*   1543    1544*
DRWC     001016         517*    565*    594*    597     615*    616     829*    830     833*    834     857*    858     861*
                       1010*   1024    1042*   1056    1081*   1102*   1120*   1140*   1154*   1180*   1314*   1336*   1350*
                       1371*   1398*   1429*   1457*   1476*   1520    1870*   1949*
DSTATA=  001000        1637*
DSTATB=  002000        1638*   1872    1937
DSTATC=  004000        1639*   1765    1855    1862    1953
ERRA     001234         563     573*
ERRCHK   007002        1023    1055    1560*
ERRDO    005416        1339    1344*
ERRDO1   005474        1347    1356*
ERROR =  100000        1643*
FLUSH    010616        1813    1976*   1993
FLUSH1   010624        1978*   1979
FNCCNT   001066         537*   1423*   1436    1439*   1440
FNCT1 =  000002        1629*
FNCT2 =  000004        1630*   1879    1935
FNCT3 =  000010        1631*   1961    1952
GETBUF   011022        2052    2061*
GO    =  000001        1628*   1881    1920    1940
HDFHM    011556        2174    2176*
HLT   =  104000         464*    573     589     605     618     629     642     646     659     663     676     680     689
                        693     702     706     715     719     728     732     741     745     755     758     761     764
                        767     770     773     776     780     783     786     789     792     796     799     802     816
```

```
                         822    832    836    845    849    860    873    891    897    910    916    930    936
                         949    953    956    970    973    979    981    986    993    999   1005   1009   1021
                        1026   1029   1032   1053   1058   1061   1065   1091   1130   1161   1164   1167   1170
                        1187   1191   1194   1198   1201   1205   1213   1217   1220   1224   1227   1232   1245
                        1255   1264   1268   1279   1293   1296   1305   1318   1322   1329   1343   1346   1355
                        1358   1387   1438   1516   1519   1522   1528   1539   1553   1558   1563   1566   1694
                        1698   1702   1763   1769   1775   1779   1857   1911   1915   1929   1933   1961   2037
                        2042   2059
HNDSHK   010330         1865   1890*  1902
HNDSH1   010336         1893*  1894
HNDSH2   010356         1898   1900*  1901
ICOUNT   011730          593*   614*   809*   829*   855*   883*  1179*  1239*  1251*  1274*  1302*  1313*  2215
                        2222*
IE     = 000100         1634*  1663   1679   1834   1835   1881   1920   1940
INBUF    001040          526*  1082   1103   1121   1141   1315   1372   1399   1419   1425   1442*  1445*  1458
                        1477   1493   1525   1534   1548   1868
INBUF1   001070          538*  1425*  1445
INCBA    002534          871*   876
INCDB    005144         1277*  1282
INCWC    002504          858*   863
INTA     006556         1084   1105   1123   1143   1374   1401   1426   1460   1479   1513*
INTB     003464         1014   1023*
INTC     003636         1046   1055*
JBCNT    007222         1651*  1656*  1800   1805*  1838*
JBFLAG   007220         1650*  1659*  1675*  1704   1709   1710*  1761   1771*  1791*
JOBAD    007700         1660   1799*  1836
JOBADA   007760         1801   1806*
JOBADB   007740         1802*  1803
JOBA1    010012         1809   1819*
JOBA2    010050         1822   1831*
LDEXIT   006510         1498   1501*
LENCHK   001046          529*  1494*  1496*  1497   1499   1503*  1507*  1508   1535*  1536*  1540   1549*  1550*
                        1554
LIST     011602         2136   2185*
LISTA    011034         1789   1807   1831   2067*
LISTA1   011046         1790   1818   1820   1832   2073*
LISTB    011060         1864   1866   1877   2079*
LOADA    006466         1496*  1500
LODBUF   006454         1077   1098   1117   1136   1368   1394   1420   1455   1474   1493*
MAINT  = 010000         1640*
MFCHK    006164         1441   1444*
MFLOOP   006056         1425*  1443
MINUS    011400         2144   2147*
MKNUM    011430         2153*  2156
MOADD    011512         2165*  2167
MREC     010440         1821   1927*
MREC1    010452         1928   1931*
MREC2    010464         1932   1935*
MREC3    010476         1937*  1938
MSTART   001000          512*
MSX      001006          512    514*
MS1      007224          514   1555*
MXMIT    010364         1808   1909*
MXMIT1   010376         1910   1913*
MXMIT2   010410         1914   1917*
NEX    = 040000         1642*
```

```
NEXCHK  004436  1156  1162#
NEXJOB  007214  1648#  1660*  1661*  1676*  1677*  1707*  1708*  1767  1771  1777  1781  1809*  1810*
                1822*  1823*  1836*  1837*  1947*  1848*  1882*  1883*
NOP     = 000240  462#
NORMAL  006713  1033  1066  1172  1233  1361  1530  1543#
NPRDY   003372  996  1010#
NPR1    001034  524#  1011  1013*  1027  1043  1044*  1059  1063
NXROXF  007212  1647#  1697  1967
N1413   003356  1001  1006#
OFL     012460  2326#
OUT     012250  2239  2241  2245  2268  2285#
PC      =%000007  552*  1625#  2247*  2249*  2253*  2257*  2274*
PRINT   011072  488  1745  2092#
PRMXFR  010524  1918  1936  1948#
PRMXF1  010556  1955#  1969
PRMXF2  010562  1957#  1959
PRMXF3  010576  1960  1963#
PRMXF4  010600  1964#  1965
PRTAB   011312  2109  2114  2119  2134#  2251
PSW     001014  497*  516#  553*  884*  903*  923*  942*  1016*  1048*  1086*  1107*  1125*  1145*
                1158*  1184*  1210*  1340*  1376*  1403*  1428*  1462*  1481*  1545*  1589*  1600*  1603*
                1662*  1664*  1678*  1680*  2201*
P4INT   003116  950  954#
P4INV   003076  944  950#
P6ERR   003034  931  936#
P6INV   003006  925  931#
P6ERR   002744  911  916#
P6INV   002716  905  911#
P7ERR   003654  892  897#
P7INV   002626  886  892#
RDSW    011746  2235#
RDYCHK  001054  532#  1323*  1326*
READY   = 000200  1635#
RESALL  007502  1712  1730#
RETURN  011734  554*  2198*  2202  2220*  2224#
R0      =%000000  1617#  1686*  1692  1723  1731*  1742*  1743*  1744*  1749*  1789*  1791  1792  1797*
                1798  1807*  1818*  1820*  1831*  1864*  1866*  1867*  1868*  1969  1970  1977*  1991*
                1948  1949  1955  1959  1996  1997  1998  2017*  2018*  2027  2030*  2031  2032
                2033*  2034  2035  2054  2056
R1      =%000001  1618#  1687*  1696  1722  1732*  1750*  1790*  1791*  1793*  1796*  1832*  1859*  1996*
                1897  1950*  1966*  1967  1982*  1996*  2028  2031*  2040  2061
R2      =%000002  1619#  1688*  1700  1721  1733*  1751*  1792*  1793  1794*  1795*  1796  1860*  1892*
                1893  1895*  1900  1951*  1956*  1957  1963*  1964  1985*  1997*  2032*  2035  2043*
                2064*
R3      =%000003  1620#  1689*  1700  1720  1734*  1752*  1861*  1892  1895  1952*  1956  1963  1983*
                1999*  2017  2034*  2040  2054  2062*
R4      =%000004  1621#  1690*  1696  1719  1735*  1753*  1862*  1893  1900  1953*  1957  1964  1984*
                2000*  2018  2056  2063*
R5      =%000005  1622#  1691*  1692  1706*  1712*  1724  1736*  1754*  1863*  1890*  1891  1954*  1955*
                1959  1976*  1982  1983  1992*  2009*  2012*  2015*  2061*  2062
R6      =%000006  1623#  1657*  1673*  1719*  1720*  1721*  1722*  1723*  1730  1731  1732  1733  1734
                1735  2027*  2028*  2030  2033
R7      =%000007  1624#  1658*  1674*  1755*  1806*  1808*  1819*  1821*  1833*  1865*  1878*  1999*  1919*
                1921*  1936*  1941*  1969*  1977*  1978*  1986*  2001*  2008*  2010*  2013*  2019*  2029*
                2045*  2051*  2052*  2058*  2065*
SAVALL  007466  1706  1719#
```

```
SAVCC   011310          2097*   2113    2132*
SAVPC   011306          2096*   2108    2131*
SAVR2   011300          2105*   2120    2128*
SAVR3   011302          2106*   2121    2129*
SAVR4   011304          2107*   2122    2130*
SCOPE = 104405           463*    562     575     590     599     613     625     638     655     672     685     698     711
                         724     737     750     808     827     841     854     868     922     938     952     941     963
                         994    1041    1075    1096    1115    1134    1153    1178    1238    1250    1260    1273    1293
                        1301    1312    1335    1366    1392    1417    1451    1470    1491
SCOPEA  011614          2196*
SCOPEB  011636          2197    2200*   2211    2218
SCOPEC  011650           491    2209*
SCOPEF  011732          2215    2217*   2219*   2223*
SCOPEG  011716          2214    2216    2219*
SETBPT  010426          2010*   2016
SETBUF  010470          1806    2009*
SETVEC  007520          1658    1674    1742*
SEVEN   011570          2139*   2148    2159    2165*   2180*
SJOB1   010122          1676    1847*   1522
SJOB2   010140          1677    1848    1855*   1833
SJOB2A  010154          1856    1859*
SJOB3B  010246          1874*   1875
SJOB3C  010260          1873    1877*
SJOB2D  010304          1876    1891*
SP      =%000006         540*    541*    546     548     549    1626*
SR      001012           515*    543     547*    550    1583    2093    2124    2196    2210    2213    2238    2250    2294*
SSTART  001002           513*
SS1     007304           513    1672*
START   011436          2146    2148*   2168
SUSWR   001072           503     540*
SWREG   000176           494*    547     550    2238
TEMPST  011742          2233*   2254*   2255*   2275*   2277*   2278*   2280*   2284
TIB     011750          2236*   2242*   2243*   2244    2258*   2259    2262    2269    2271    2279*   2280    2294*   2298*
                        2301
TKB     001060           534*   2242    2293*   2298
TKS     001056           533*   1342*   2240    2292*   2295*   2296
TNPRC   003542          1035    1041*   1070
TNPR1   003306           995*   1037
TOGGLE  011566          2149*   2150*   2151    2179*
TPB     001064           536*   2099*   2102*   2110*   2115*   2145*   2147*   2172*   2301*   2347*
TPS     001062           535*   2100    2103    2111    2116    2137*   2141    2170    2176    2299    2338*   2345
TRPA    007122          1586    1590*
TRPB    007126          1591*   1601*
TRTRAP  007056          1582*
TSTRDY  005320          1324*   1328
TTIN    012252          2257    2292*
TTIN1   012272          2296*   2297
TTIN2   012306          2299*   2300
TTOUT   012462           500     502    1576    2247    2249    2253    2265    2274    2334*   2348    2354
T33CLR  003534          1022    1036*
T34CLR  003706          1054    1069*
T35     003714          1068    1075*
WAIT1   011356          2141*   2142
WAIT2   011534          2170*   2171    2175
WCLEN   001052           531*   1079*   1080*   1081    1100*   1101*   1102    1118*   1119*   1120    1138*   1139*   1140
                        1369*   1370*   1371    1396*   1397*   1398    1453*   1454*   1457    1472*   1473*   1476
```

K04


```
WRTCT    011574        2135*   2153*   2155    2160*   2192*
WRTOC    011450        2152    2157*
XBA16  = 000020        1632*
XBA17  = 000040        1630*
XCHKBU   014572         527    2363*
XINBJF   013570         526    2067    2079    2361*
XLIST    011526        2163    2169*
YESRT    007206        1598    1608*
YESTR    007130        1594    1596*
YESTR1   007202        1585    1587    1596*   1606*
YESTR2   007204        1598    1597*   1607*
$CNTG    012324        2246    2304*
$CRLF    012332        2264    2305*
$CRI1    012365        1575    2312*
$HD    = 000003         453     454
$HEAD    012433         499    2320*
$MAIN    012403         501    2315*
$MNEW    012346        2252    2309*
$MSWR    012336        2248    2306*
$QUEST   012357        2273    2310*
$READ    012360        2255*
$SWR   = 160000         453     454*
$TN    = 000001         453*
.      = 014574         486*    487*    490*    493*    495*    504*    572     589     604     617     629     641     645
                        659     662     675     679     688     692     701     705     714     718     727     731     740
                        744     754     757     760     763     766     769     772     775     779     782     795     799
                        791     795     798     801     815     821     831     835     844     848     859     872     890
                        996     909     915     929     935     948     965     969     972     977     980     985     998
                        998    1004    1008    1025    1028    1031    1057    1060    1064    1088    1090    1109    1127
                       1129    1147    1163    1166    1169    1190    1193    1197    1200    1204    1216    1219    1223
                       1226    1231    1244    1254    1263    1267    1278    1292    1295    1304    1317    1321    1327
                       1345    1357    1383    1384    1396    1410    1411    1437    1464    1488    1499    1509    1515
                       1518    1521    1527    1538    1552    1557    1562    1565    1574    1602    1693    1697    1701
                       1778    2094    2101    2104    2112    2117    2125    2177    2357*   2359*   2361    2362*   2363
.EMPTY   012506        2337    2341*
.REST    012552        2344    2353*
.RET     012536        2342    2349*   2352
.RETR    012560        2351    2356*
.SAV     012564        2350*   2353    2359*
```

```
COMMEN         1#
ENDCOM         1#
ESCAPE         1#
GETPRI         1#
GETSWR         1#
[illegible]    1#
NEWTST         1#
POP            1#
PUSH           1#
REPORT         1#
SETPRI         1#
SETUP          1#
SKIP           1#
[illegible]    1#
[illegible]    1#     440#    441     455     459     507     509     557     559     579     580     594     596     608     610
             620     622     633     635     650     652     667     669     692     694     695     697     708     710     721
             723     724     736     747     749     804     807     824     826     838     840     851     853     855     867
             879     881     899     901     919     921     939     940     959     961     991     993    1038    1040    1072
            1074    1093    1095    1112    1114    1131    1133    1150    1152    1174    1177    1235    1237    1247    1249
            1257    1259    1270    1272    1284    1287    1295    1300    1309    1311    1331    1334    1353    1355    1389
            1391    1414    1416    1448    1450    1467    1469    1569    1571    1577    1579    1593    1595    1612    1614
            1652    1654    1669    1670    1682    1684    1715    1717    1726    1728    1738    1740    1756    1759    1784
            1788    1813    1816    1826    1829    1841    1845    1851    1853    1886    1888    1904    1907    1923    1925
            1942    1946    1971    1974    1988    1991    2003    2006    2021    2025    2047    2050    2086    2099    2190
            2192    2204    2206    2229    2232    2296    2298    2329    2331
SWRSU          1#
TYPBIN         1#
TYPDEC         1#
TYPNAM         1#
TYPNUM         1#
TYPOCS         1#
TYPOCT         1#
TYPTXT         1#
$$ESCA         1#
$$NEWT         1#
$$SKIP         1#
.EQUAT         1#
.HEADE         1#     440#    443
.KT11          1#
.SETUP         1#
.SWRHI         1#
.$ACT1         1#
.$APTB         1#
.$APTH         1#
.$APTY         1#
.$ASTA         1#
.$CATC         1#
.$CMTA         1#
.$DB2D         1#
.$DB2O         1#
.$DIV          1#
.$EOP          1#
.$ERRC         1#
.$ERRT         1#
.$MULT         1#
.$POWE         1#
```

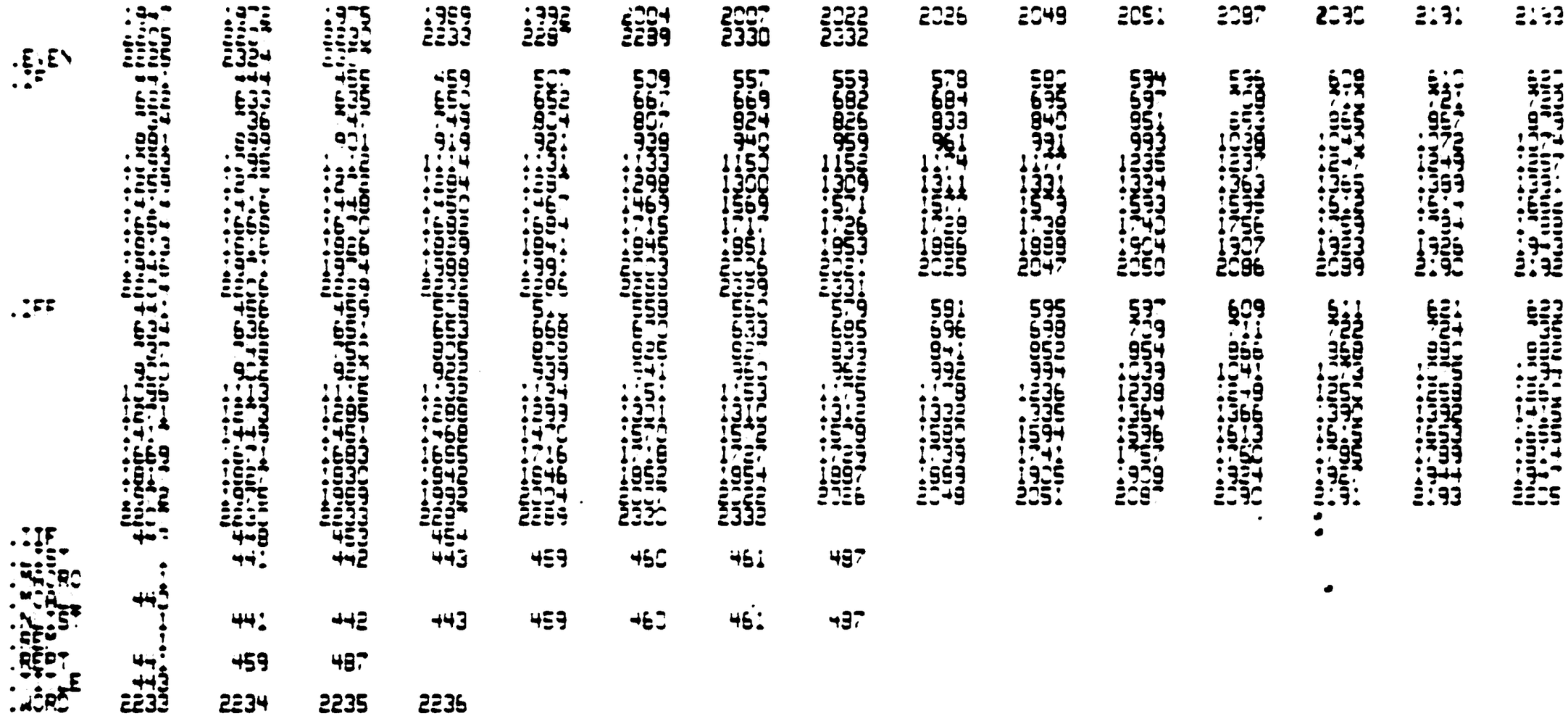
M04


```
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.$[illegible]     18
.[illegible]      18
```

N04

ADC	1984	2063													
ADD	874	875	1326	1360	1442	1507	1510	1524	1525	1529	1796	1968	1982	2062	2153
	2157	2159	2165												
ASL	1795	2276	2277	2279											
ASR	1078	1099	1435												
BEQ	544	598	604	617	629	645	662	679	692	705	719	731	744	779	792
	785	789	791	795	798	801	815	821	831	835	844	849	859	872	965
	972	980	988	998	1001	1004	1009	1025	1028	1031	1057	1060	1064	1090	1129
	1197	1220	1244	1254	1263	1267	1278	1295	1304	1380	1396	1407	1437	1441	1493
	1498	1521	1527	1539	1552	1594	1596	1693	1697	1701	1705	1752	1766	1873	1875
	1901	1959	1960	2041	2094	2152	2163	2174	2216	2292	2335	2342	2344		
BGE	2011	2014	2272												
BGT	1801	2270													
BIC	643	660	677	690	703	716	729	742	777	820	895	914	934	1002	1195
	1221	1349	1434	1513	1560	1589	1834	1890	1895	1919	1939	1963	2151	2243	2259
	2238														
BICB	2137	2273													
BIS	639	656	673	686	699	712	725	738	751	810	813	893	912	932	951
	967	975	983	1430	1603	1879	1981	1952	1920	1940	1956				
BISB	2280														
BIT	640	644	657	661	674	678	687	691	700	704	713	717	726	730	739
	743	753	756	759	762	765	768	771	774	778	781	784	787	790	794
	797	800	885	889	904	908	924	928	943	947	964	969	971	975	979
	984	987	997	1000	1003	1168	1196	1222	1291	1294	1303	1379	1406	1484	1593
	1765	1855	1872	1874	1893	1900	1937	1957	1964	2093	2136	2210	2213		
BLE	2036														
BLT	1898	1969	1979	2044	2053										
BMI	1190	1193	1200	1216	1219	1226	1317	1325	1327	1345	1357	1519	1565	1602	1774
	1914	1932	2144												
BNE	551	641	658	675	688	701	714	727	740	754	757	760	763	766	769
	772	775	863	876	886	890	905	909	925	929	944	948	969	977	995
	1020	1052	1163	1166	1169	1282	1292	1354	1541	1555	1557	1768	1779	1956	1894
	1938	1965	2055	2057	2156	2167	2197	2211	2214	2239	2245	2260	2263	2267	2337
BPL	996	1204	1231	1321	1509	1515	1562	1574	1803	1910	1928	2101	2104	2112	2117
	2125	2142	2171	2177	2241	2297	2300	2346							
BR	512	545	572	896	915	935	1022	1035	1037	1054	1068	1070	1099	1109	1127
	1147	1328	1383	1384	1410	1411	1443	1464	1488	1499	1500	1511	1565	1713	1976
	1902	1980	1993	2016	2146	2168	2175	2218	2261	2268	2275	2293	2348	2352	2354
CLR	497	498	569	855	856	857	869	870	954	963	974	982	990	1006	1012
	1016	1018	1036	1044	1048	1050	1069	1086	1107	1125	1145	1159	1184	1202	1210
	1274	1275	1276	1323	1340	1348	1352	1376	1403	1422	1428	1462	1491	1494	1495
	1503	1504	1535	1544	1549	1599	1600	1656	1659	1664	1675	1690	[illegible]	1709	[illegible]
	1747	1749	1750	1751	1752	1753	1754	1797	1805	1959	1971	1950	1976	1999	2000
	2009	2134	2135	2160	2161	2219	2254	2255	2292	2293	2294	2339			
CMP	543	546	550	616	627	830	834	843	847	858	871	955	1027	1030	1034
	1059	1063	1067	1099	1129	1171	1198	1214	1253	1262	1266	1277	1395	1440	1497
	1508	1526	1537	1540	1551	1554	1556	1692	1696	1700	1799	1900	1997	1959	1967
	2035	[illegible]	2054	2056	2098	2155	2200	2215	2238	2244	2266				
CMPB	1436	[illegible]	[illegible]	2269	2271	2335	2341	2343							
COM	1601	[illegible]	[illegible]												
DEC	1137	1395	2162	2166	2173	2261									
EMT	464														
HALT	487	528	529	530	531	532	537	538	1609	1695	1699	1703	1764	1770	1776
	1780	[illegible]	1912	1916	1930	1934	1562	2038	2126						
INC	555	[illegible]	571	914	961	962	1019	1051	1160	1186	1212	1280	1281	1342	1351
	1353	1373	1405	1439	1483	1496	1536	1550	1656	1697	1698	1699	1690	1691	1938

CROSS REFERENCE TABLE -- PERMANENT SYMBOLS



ERRORS DETECTED: 0
DEFAULT ALLOCATES GENERATED: 0

*,DZORB.SEQ/SOL/CRF/PAGNUM=LIB:SYSMAC.SML,DSK:DZORB
RUN-TIME: 30.37 4 SECONDS
RUN-TIME RATIO: 17177=2.3
CORE USED: 33K (65 PAGES)

DOS

Spooler runtime 9 Seconds, 41 KCS, 246 disk reads, 3 disk writes, 54 pages