

IDENTIFICATION

PRODUCT CODE: MAINDEC-08-DGDRA-A-D
PRODUCT NAME: DR8-EA DIAGNOSTIC (FOR USE
ON SYSTEMS THAT HAVE A TRADITIONAL
PDP-8 PROCESSOR WITH A DW8/E
OMNIBUS CONVERTER
DATE REVISED: JULY 3, 1973
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: MIKE HARE

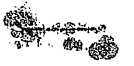
COPYRIGHT (C) 1972, 1973, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED TO PURCHASER UNDER A LICENSE FOR USE
ON A SINGLE SYSTEM AND CAN BE COPIED (WITH INCLUSION
OF DEC'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT
AS MAY OTHERWISE BE PROVIDED IN WRITING BY DEC.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
EQUIPMENT CORPORATION.

DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY
OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.

2/3A



1. ABSTRACT

THIS DIAGNOSTIC HAS BEEN MODIFIED FOR TESTING THE DR8=EA INTERFACE ON SYSTEMS THAT HAVE A TRADITIONAL PDP-8 PROCESSOR WITH A DW8/E BUS CONVERTER.

THIS PROGRAM IS A DIAGNOSTIC AND EXERCISER FOR THE DR8=EA 12 CHANNEL BUFFERED DIGITAL INTERFACE. ALL FUNCTIONS ARE TESTED AND ERRORS ARE REPORTED BY HALTS AND/OR ERROR TYPEOUTS.

2. REQUIREMENTS

2.1 EQUIPMENT

TRADITIONAL PDP-8 COMPUTER WITH 4K OF CORE
ASR-33 TELETYPE (OR EQUIVALENT)
DW8/E BUS CONVERTER
DR8=EA WITH TEST CABLE

2.2 STORAGE

THE PROGRAM USES LOCATIONS 0000-4121

3. LOADING PROCEDURE

THE STANDARD PROCEDURE FOR LOADING BINARY TAPES SHOULD BE USED.

4. STARTING PROCEDURE

4.1 STARTING ADDRESS

200-INPUT DEVICE CONFIGURATION
201-START WITH STANDARD CONFIGURATION

4.2 SWITCH SETTINGS

FOR EITHER STARTING ADDRESS, NORMAL SETTING IS SR0-SR11= 0 (DOWN).

4.3 PROGRAM AND/OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY
SET SWITCH REGISTER TO DESIRED STARTING ADDRESS
LOAD ADDRESS
CLEAR SWITCHES
PRESS CLEAR AND CONTINUE

4.3.1 FOR STARTING ADDRESS 200

THE PROGRAM WILL TYPE "SET SR FOR DEVICE CODE AND CONT"
AND THEN HALT.

SET SWITCHES TO 000X WHERE X IS AN OCTAL
NUMBER CORRESPONDING TO THE 3 LSB OF THE DEVICE SELECTOR CODE,
E.G. IF DEVICE CODE IS 53, SET SR TO 0003,
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING
"SET SR FOR INTERRUPT JUMPERS AND CONT" AND THEN HALT.
SET SWITCHES FOR ALL INPUT REGISTER BITS JUMPED TO INTERRUPT.
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING
"SET SR FOR FLIPFLOP JUMPERS AND CONT" AND THEN HALT.
SET SWITCHES FOR ALL INPUT REGISTER FLIPFLOPS.
PRESS CONTINUE.

PROGRAM WILL RESPOND BY TYPING
"SET SR FOR RUN" AND THEN HALT.
SET SWITCHES AS DESIRED. (REFER TO SECTION 5.1)
PRESS CONTINUE.

PROGRAM WILL BEGIN TEST EXECUTION

4.3.2 FOR STARTING ADDRESS 201

SET SWITCHES AS DESIRED. (REFER TO SECTION 5.1)
DEPRESS CONTINUE.

PROGRAM WILL BEGIN TEST EXECUTION

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

SR0=1, SUPPRESS ERROR HALT
SR1=1, SUPPRESS ERROR TYPEOUT
SR2=1, LOOP ON CURRENT TEST
SR3=1, LOOP WITH CURRENT DATA
SR4=1, SUPPRESS BELL OR TYPEOUT AT END OF PASS
SR5=1, SUPPRESS ITERATIONS
SR6=1, ESCAPE TO NEXT TEST ON ERROR

5.2 PROGRAM AND/OR OPERATOR ACTION

- 5.2.1.1 WITH SWITCHES SET AS IN 4.2, THE PROGRAM WILL RUN ALL TESTS SEQUENTIALLY, EACH IOT TEST WILL BE REPEATED 4096 TIMES, EACH DATA TEST WILL BE REPEATED 50 TIMES. AFTER ALL TESTS HAVE BEEN COMPLETED, THE PROGRAM WILL TYPE "DR" AND START ALL TESTS AGAIN. IF AN ERROR OCCURS, THE PROGRAM WILL HALT AND TYPE AN APPROPRIATE ERROR MESSAGE (SEE SECTION 6 FOR DETAILS).
- 5.2.1.2 WITH SR0=1 (UP), PROGRAM ACTION WILL BE AS IN 5.2.1.1, EXCEPT NO TYPEOUT WILL OCCUR.
- 5.2.1.3 WITH SR2=1 (UP) PROGRAM ACTION WILL BE AS IN 5.2.1.1. EXCEPT NO TYPEOUT WILL OCCUR. THE ADDRESS OF THE FAILING TEST WILL BE DISPLAYED IN THE COMPUTER AC.
- 5.2.1.4 WITH SR4=1 (UP), PROGRAM ACTION WILL BE AS IN 5.2.1.1 EXCEPT NO END OF PASS TYPEOUT WILL OCCUR.
- 5.2.1.5 WITH SR5=1 (UP), EACH TEST WILL BE EXECUTED ONLY ONCE. INSTEAD OF TYPING "DR", THE PROGRAM WILL RING THE TTY BELL.
- 5.2.1.6 WITH SR0=1 AND SR6=1, PROGRAM ACTION WILL BE AS IN 5.2.1.1 IF NO ERRORS OCCUR. IF AN ERROR OCCURS, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE AND IMMEDIATLY TERMINATE ITERATIONS OF THE FAILING TEST. THE PROGRAM WILL THEN START THE NEXT TEST IN SEQUENCE.

6. ERRORS

6.1 NORMAL OPERATION

IF AN ERROR OCCURS WITH SWITCHES SET AS IN 4.2, THE PROGRAM WILL TYPE AN APPROPRIATE ERROR MESSAGE (WITH DATA IF APPLICABLE) AND HALT.

THE FORMAT OF THE ERROR TYPEOUT IS

XXXX MESSAGE
HEADER FOR DATA (IF APPLICABLE)
DATA (IF APPLICABLE)

WHERE XXXX= ADDRESS OF JMS TO ERROR ROUTINE IN TEST THAT FAILED.

6.2 ERROR RECOVERY

SET SR6=1(UP) TO ESCAPE TO NEXT TEST, PRESS CONTINUE.

6.3 ERROR LOOP (IOTS)

SET SR0=1 TO SUPPRESS HALT
SET SR1=1 TO SUPPRESS TYPEOUT
SET SR2=1 TO LOOP ON CURRENT FAILING TEST

6.4 ERROR LOOP (DATA)

SAME AS 6.3 EXCEPT USE SR3 INSTEAD OF SR2 TO LOOP WITH CURRENT DATA.

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

TEST JUMPER CABLE MUST BE INSTALLED.
ANY FLOATING INPUTS TO INPUT REGISTER SHOULD BE GROUNDED, OR ERRORS MAY OCCUR.

7.2 OPERATING RESTRICTIONS

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIME

EXECUTION TIME IS APPROXIMATELY 9 MINUTES FOR FULL ITERATION
AND APPROXIMATELY 10 SECONDS WITH ITERATIONS SUPPRESSED.

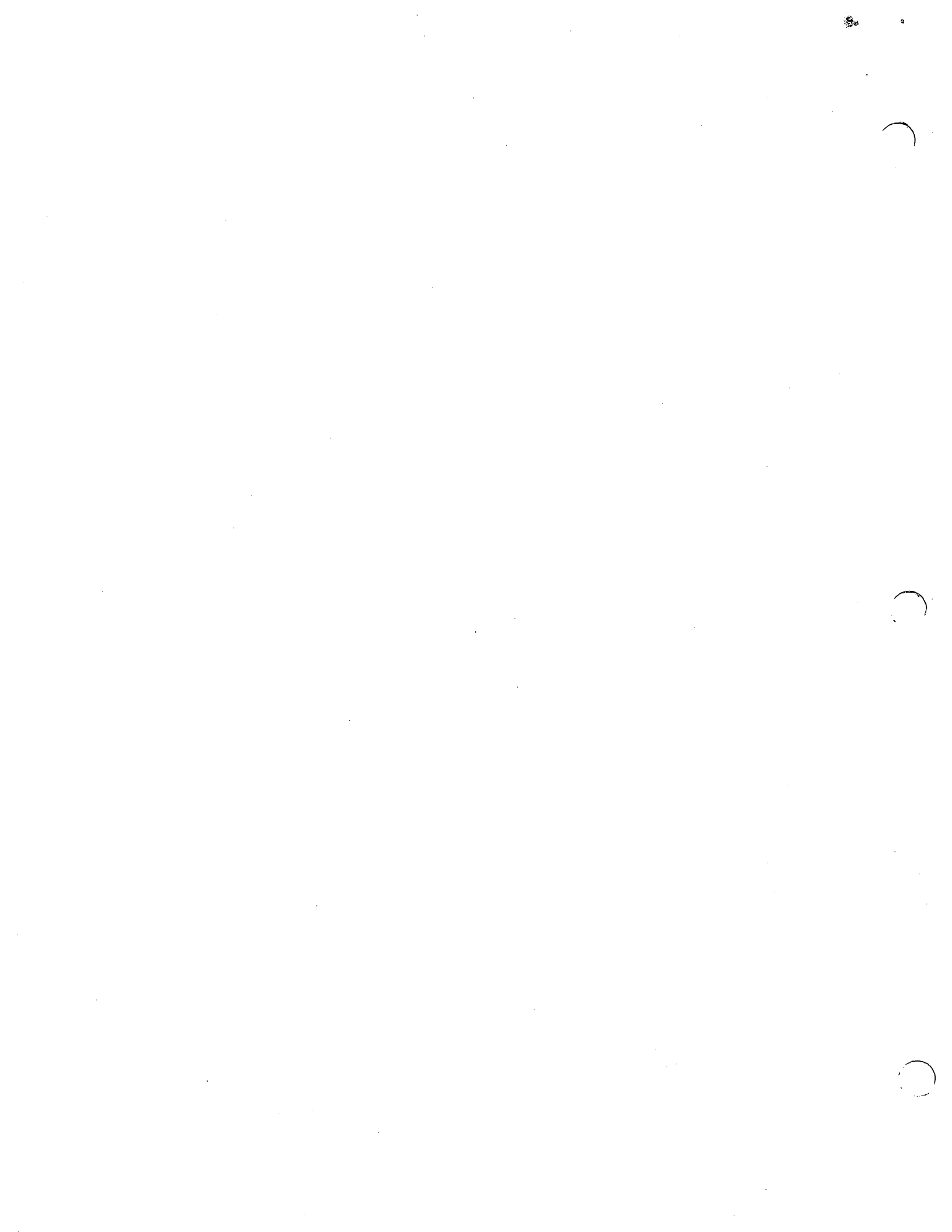
9. PROGRAM DESCRIPTION

THE DR8-EA IS A TEST OF ALL FUNCTIONS OF THE INTERFACE.

THE PROGRAM SEQUENCE IS AS FOLLOWS:

ALL BASIC I/O TESTS ARE EXECUTED 4096 TIMES,
ALL OUTPUT REGISTER FUNCTIONS ARE TESTED WITH BINARY COUNT PATTERNS;
ALL INPUT REGISTER FUNCTIONS ARE TESTED USING BINARY COUNT PATTERNS.
INTERACTION BETWEEN INPUT AND OUTPUT REGISTERS IS TESTED FOR
WITH BINARY COUNT PATTERNS.
ALL SKIPS AND INTERRUPTS ARE TESTED USING BINARY COUNT PATTERNS.

10. LISTING



/MAINDEC-08-DGDRA-A DR8-EA DIAGNOSTIC FOR TRADITIONAL PDP8 PROCESSORS
 /WITH A DWS/E OMNIBUS CONVERTER
 /COPYRIGHT 1972,1973 DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

/STARTING ADDRESS: 200=INPUT DEVICE CONFIGURATION
 / 201=USE STANDARD CONFIGURATION

/SWITCH REGISTER OPTIONS

/SR00 =1, SUPPRESS HALT ON ERROR
 /SR01 =1, SUPPRESS ERROR TYPEOUT
 /SR02 =1, LOOP ON CURRENT TEST
 /SR03 =1, LOOP WITH CURRENT DATA
 /SR04 =1, SUPPRESS BELL AT END OF PASS
 /SR05 =1, SUPPRESS ITERATIONS
 /SR06 =1, ESCAPE TO NEXT TEST ON ERROR

/INSTRUCTION DEFINITIONS

3114 MQL=DCA MQ
 4115 MQA=JMS SIMMQA
 4046 BSW=JMS SIMBSW
 4063 CAF=JMS SIMCAF
 4076 SRQ=JMS SIMSRQ
 4432 DBEI=JMS I XDBEI
 4433 OBSK=JMS I XDBSK
 4434 DBCI=JMS I XDBCI
 4435 DBRI=JMS I XDBRI
 4436 DBCO=JMS I XDBCO
 4437 DBSO=JMS I XDBSO
 4440 DBRO=JMS I XDBRO

/LOCATION EQUIVALENCIES

3026 MSTDGT=ERADR+1
 3027 LSTDGT=ERADR+2

/GENERAL VARIABLES

0001 0001 *1
 0001 5402 JMP I 2
 0010 0010 *10
 0010 0000 POINT1, 0
 0020 0020 *20
 0020 0000 CNTR1, 0
 0021 0000 DATA1, 0
 0022 0000 DATA2, 0
 0023 0000 DATA3, 0
 0024 0000 DATA4, 0
 0025 0000 DATA5, 0
 0026 7777 IJUMPE, 7777 /INTERRUPT JUMPER MASK
 0027 7777 FJUMPE, 7777 /FLIPFLOP JUMPER MASK
 0030 0000 TYPFLG, 0

0031 0000 LPCNT, 0

/
/INDIRECT POINTERS
/

0032 0266 XDBEI, DBEIX
 0033 0273 XDBSK, DBSKX
 0034 0300 XDBCI, DBCIX
 0035 0305 XDBRI, DBRIX
 0036 0312 XDBCO, DBCOX
 0037 0317 XDBSO, DBSOX
 0040 0324 XDBRO, DBROX
 0041 3200 XPRINT, PRINT
 0042 3251 XTYPE, TYPE
 0043 2600 XERROR, ERROR
 0044 2667 XLOOP1, LOOP1
 0045 2712 XLOOP2, LOOP2
 0046 0000 SIMBSW, 0
 0047 3113 DCA TYE1
 0050 7012 RTR
 0051 7012 RTR
 0052 7012 RTR
 0053 1113 TAD TYE1
 0054 0062 AND K7700
 0055 1113 TAD TYE1
 0056 7006 RTL
 0057 7006 RTL
 0060 7006 RTL
 0061 5446 JMP I SIMBSW
 0062 7700 K7700, 7700

/
SIMCAF, 0

0063 0000 IOF
 0064 6002 CLA CMA
 0065 7240 DBCI
 0066 4434 SKP
 0067 7410 HLT /DBCI SKIPPED
 0070 7402 DBCO
 0071 4436 SKP
 0072 7410 HLT /DBCO SKIPPED
 0073 7402 CLA CLL
 0074 7300 JMP I SIMCAF
 0075 5463

/
SIMSRQ, 0

0076 0000 DCA TYE1
 0077 3113 TAD (RETURN
 0100 1177 DCA 2
 0101 3002 ION
 0102 6001 NOP
 0103 7000 IOF /NO INTERRUPT
 0104 6002 SKP
 0105 7410 RETURN, ISZ SIMSRQ
 0106 2076 CLA CLL CML RAL
 0107 7324 DCA 2
 0110 3002 TAD TYE1
 0111 1113

```

0112 5476      JMP I SIMSRQ
0113 0000      TYP1, 0
0114 0000      MQ, 0
/
0115 0000      SIMMQA, 0
0116 3113      DCA      TYP1
0117 1113      TAD      TYP1
0120 7040      CMA
0121 0114      AND      MQ
0122 1113      TAD      TYP1
0123 5515      JMP I SIMMQA
/

```

/TEST INITIALIZATION

```

0200 0200      *200
0200 5202      JMP      START1
0201 5244      JMP      START2
0202 4063      START1, CAF      /CLEAR ALL FLAGS
0203 4441      JMS I   XPRINT      /TYPE "SET SR FOR DEVICE
0204 3377      M1=1      /CODE AND CONT"
0205 7402      HLT      /HALT FOR SWITCHES
0206 7604      LAS      /GET SWITCHES
0207 0377      AND      (7      /MASK DEVICE CODE
0210 7106      CLL RTL      /POSITION BITS
0211 7004      RAL
0212 1376      TAD      (6500      /GENERATE BASIC IOT
0213 3247      DCA      IOTS      /SAVE BASIC IOT
0214 4441      JMS I   XPRINT      /TYPE "SET SR FOR JUMPERS
0215 3420      M2=1      /AND CONT"
0216 7402      HLT      /HALT FOR SWITCHES
0217 7604      LAS      /GET SWITCHES
0220 3026      DCA      IJUMPER      /SAVE JUMPER MASK
0221 4441      JMS I   XPRINT
0222 3444      M2A=1
0223 7402      HLT
0224 7604      LAS
0225 3027      DCA      FJUMPER
0226 1375      TAD      (=10      /0 IOTS WILL BE
0227 3020      DCA      CNTR1      /SET UP
0230 1260      TAD      DIOT      /STORE INSTRUCTION FOR
0231 3233      DCA      PNTR1      /IOT SET UP
0232 1247      TAD      IOTS      /GET IOT
0233 0000      PNTR1, 0      /STORE IT
0234 2233      ISZ      PNTR1      /PREPARE TO STORE
0235 2247      ISZ      IOTS      /NEXT IOT
0236 2020      ISZ      CNTR1
0237 5232      JMP      PNTR1=1
0240 4441      JMS I   XPRINT      /TYPE "SET SR FOR RUN
0241 3475      M3=1      /AND CONT
0242 7402      HLT      /HALT FOR SWITCHES
0243 7300      CLA CLL
0244 3030      START2, DCA      TYPFLG      /CLEAR ERROR FLAG
0245 5646      JMP I   .+1      /GO TO FIRST TEST

```

```
0246 0400      INIT1
                /
                /INITIALIZATION CONSTANTS AND VARIABLES
                /
0247 0000      IOTS, 0          /BASIC IOT
0250 0262      TIOT, DBDIX+1
0251 0267      DBEIX+1
0252 0274      DBSKX+1
0253 0301      DBCIX+1
0254 0306      DBRIX+1
0255 0313      DBCOX+1
0256 0320      DBSOX+1
0257 0325      DBROX+1
0260 3650      DIOT, DCA I  TIOT
                /
                /IOT SUBROUTINES
                /
                /DISABLE DATA BUFFER INTERRUPT (DBDI,65X0)
                /
0261 0000      DBDIX, 0
0262 6500      6500
                /
0263 7410      SKP          /TRAP FOR UNDESIRE
0264 7402      HLT          /SKIPS
0265 5661      JMP I  DBDIX
                /
                /ENABLE DATA BUFFER INTERRUPTS (DBEI,65X1)
                /
0266 0000      DBEIX, 0
0267 6501      6501
                /
0270 7410      SKP          /TRAP FOR UNDESIRE
0271 7402      HLT          /SKIPS
0272 5666      JMP I  DBEIX
                /
                /SKIP ON DATA BUFFER INPUT FLAG (DBSK,65X2)
0273 0000      DBSKX, 0
0274 6502      6502
                /
0275 7410      SKP
0276 2273      ISZ  DBSKX
0277 5673      JMP I  DBSKX
                /
                /DS TO INPUT REGISTER CORRESPONDING
                /TO 1S IN AC (DBC1,65X3)
                /
0300 0000      DBCIX, 0
0301 6503      6503
                /
0302 7410      SKP          /TRAP FOR UNDESIRE
0303 7402      HLT          /SKIPS
0304 5700      JMP I  DBCIX
```

```

/
/INPUT REGISTER TO AC (DBRI,65X4)
/
0305 0000 DBRIX, 0
0306 6504          6504

0307 7410 SKP          /TRAP FOR UNDESIRE
0310 7402 HLT          /SKIPS
0311 5705 JMP I DBRIX

/
/ZEROS TO OUTPUT REGISTER CORRESPONDING TO
/ONES IN AC (DBCO,65X5)
0312 0000 DBCOX, 0
0313 6505          6505

0314 7410 SKP          /TRAP FOR UNDESIRE
0315 7402 HLT          /SKIPS
0316 5712 JMP I DBCOX

/
/IS TO OUTPUT REGISTER CORRESPONDING
/TO 1S IN AC (DBSO,65X6)
/
0317 0000 DBSOX, 0
0320 6506          6506

0321 7410 SKP          /TRAP FOR UNDESIRE
0322 7402 HLT          /SKIPS
0323 5717 JMP I DBSOX

/
/JAM TRANSFER OUTPUT REGISTER TO AC (DBRO 65X7)
/
0324 0000 DBROX, 0
0325 6507          6507

0326 7410 SKP          /TRAP FOR UNDESIRE
0327 7402 HLT          /SKIPS
0330 5724 JMP I DBROX
0375 7770
0376 6500
0377 0007
0400 PAGE 0400

/
/IS OUTPUT REGISTER CLEARED BY INITIALIZE?
/
0400 3030 INIT1, DCA TYPFLG /CLEAR ERROR FLAG
0401 3031 DCA LPCNT /SET ITERATION COUNT TO 4096(DECIMAL)
0402 4063 CAF /INITIALIZE INTERFACE
0403 4440 DBRO /READ OUTPUT REGISTER
0404 3021 DCA DATA1 /SAVE REGISTER DATA
0405 1021 TAD DATA1 /GET REGISTER DATA
0406 7650 SNA CLA /WAS REGISTER CLEARED BY INITIALIZE

```

```

0407 5214 JMP      .+5           /DATA CORRECT, CONTINUE
0410 4443 JMS I   XERROR          /NO, ERROR
0411 3645 INIT1E=1              /"OUTPUT REGISTER NOT CLEARED"
0412 3515 DH1=1           /"REGISTER DATA"
0413 7777 =1             /NUMBER OF WORDS TO BE OUTPUT
0414 4444 JMS I   XLOOP1        /CHECK FOR LOOP ON CURRENT TEST
0415 5202 JMP      INIT1+2       /LOOP ON CURRENT TEST
/
/IS INPUT REGISTER CLEARED BY INITIALIZE?
/
0416 3030 INIT2, DCA      TYPFLG          /CLEAR ERROR FLAG
0417 3031 DCA      LPCNT           /SET ITERATION COUNT TO 4096(DECIMAL)
0420 7410 SKP
0421 4063 CAF
0422 4435 DBRI
0423 3021 DCA      DATA1         /INITIALIZE INTERFACE
0424 1021 TAD      DATA1         /READ INPUT REGISTER
0425 7650 SNA CLA          /SAVE REGISTER DATA
0426 5233 JMP      .+5           /GET REGISTER DATA
0427 4443 JMS I   XERROR          /WAS REGISTER CLEARED
0430 3661 INIT2E=1        /DATA CORRECT, CONTINUE
0431 3515 DH1=1           /NO, ERROR
0432 7777 =1             /"INPUT REGISTER NOT CLEARED"
0433 4444 JMS I   XLOOP1        /"REGISTER DATA"
0434 5221 JMP      INIT2+3       /NUMBER OF WORDS TO BE OUTPUT
/                                     /CHECK FOR LOOP ON CURRENT TEST
/                                     /LOOP ON CURRENT TEST
/
/IS SKIP FLAG SET AFTER INITIALIZE
/
0435 3030 INIT3, DCA      TYPFLG          /CLEAR ERROR FLAG
0436 3031 DCA      LPCNT           /SET ITERATION COUNT TO 4096(DECIMAL)
0437 7410 SKP
0440 4063 CAF
0441 4432 DBEI
0442 4433 DBSK
0443 5250 JMP      .+5           /INITIALIZE INTERFACE
0444 4443 JMS I   XERROR          /ENABLE INTERFACE INTERRUPTS
0445 3674 INIT3E=1        /IS INTERFACE FLAG SET
0446 3514 DH0=1          /FLAG NOT SET, CONTINUE
0447 0000 0
0450 4444 JMS I   XLOOP1        /FLAG SET, ERROR
0451 5240 JMP      INIT3+3       /"SKIP FLAG SET"
/                                     /NO DATA HEADER
/                                     /NO DATA
/                                     /CHECK FOR LOOP ON CURRENT TEST
/                                     /LOOP ON CURRENT TEST
/
/DOES OUTPUT REGISTER JAM TRANSFER TO AC?
/
0452 3030 TRAN1, DCA      TYPFLG          /CLEAR ERROR FLAG
0453 3031 DCA      LPCNT           /SET ITERATION COUNT TO 4096(DECIMAL)
0454 4063 CAF
0455 7340 CLA CLL CMA      /INITIALIZE INTERFACE
0456 4440 DBRO
0457 3021 DCA      DATA1         /SET AC =7777
0460 1021 TAD      DATA1         /READ OUTPUT REGISTER
0461 7650 SNA CLA          /SAVE AC CONTENTS
0462 5267 JMP      .+5           /GET AC CONTENTS
0463 4443 JMS I   XERROR          /WAS AC CLEARED BY TRANSFER
/                                     /DATA CORRECT, CONTINUE
/                                     /NO, ERROR

```

```

0464 3703          TRAN1E=1          /"DBRO DID NOT CLEAR AC"
0465 3525          DH2=1            /"AC CONTENTS"
0466 7777          =1              /NUMBER OF WORDS TO BE OUTPUT
0467 4444          JMS I   XLOOP1    /CHECK FOR LOOP ON CURRENT TEST
0470 5254          JMP     TRAN1+2    /LOOP ON CURRENT TEST
/
/DOES INPUT REGISTER JAM TRANSFER TO AC
/
0471 3030          TRAN2, DCA     TYPFLG    /CLEAR ERROR FLAG
0472 3031          DCA     LPCNT     /SET ITERATION COUNT TO 4096(DECIMAL)
0473 4063          CAF          /INITIALIZE INTERFACE
0474 7340          CLA CLL CMA      /SET AC =7777
0475 4435          DBRI          /READ INPUT REGISTER
0476 3021          DCA     DATA1    /SAVE AC CONTENTS
0477 1021          TAD     DATA1    /GET AC CONTENTS
0500 7650          SNA CLA          /WAS AC CLEARED BY TRANSFER
0501 5306          JMP     ,+5       /DATA CORRECT, CONTINUE
0502 4443          JMS I   XERROR    /NO, ERROR
0503 3716          TRAN2E=1        /"DBRI DID NOT CLEAR AC"
0504 3525          DH2=1            /"AC CONTENTS"
0505 7777          =1              /NUMBER OF WORDS TO BE OUTPUT
0506 4444          JMS I   XLOOP1    /CHECK FOR LOOP ON CURRENT TEST
0507 5273          JMP     TRAN2+2    /LOOP ON CURRENT TEST
/
/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=7777,DBSO)
/
0510 3030          TRAN3, DCA     TYPFLG    /CLEAR ERROR FLAG
0511 3031          DCA     LPCNT     /SET ITERATION COUNT TO 4096(DECIMAL)
0512 4063          CAF          /INITIALIZE INTERFACE
0513 7340          CLA CLL CMA      /SET AC =7777
0514 4437          DBSO          /BIT SET OUTPUT REGISTER
0515 3021          DCA     DATA1    /SAVE AC CONTENTS
0516 1021          TAD     DATA1    /GET AC CONTENTS
0517 7040          CMA          /COMPLIMENT DATA TO TEST FOR 7777
0520 7650          SNA CLA          /DID AC CHANGE
0521 5326          JMP     ,+5       /DATA CORRECT, CONTINUE
0522 4443          JMS I   XERROR    /NO, ERROR
0523 3731          TRAN3E=1        /"DBSO CHANGED AC"
0524 3525          DH2=1            /"AC CONTENTS"
0525 7777          =1              /NUMBER OF WORDS TO BE OUTPUT
0526 4444          JMS I   XLOOP1    /CHECK FOR LOOP ON CURRENT TEST
0527 5312          JMP     TRAN3+2    /LOOP ON CURRENT TEST
/
/DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=7777, DBCO)
/
0530 3030          TRAN4, DCA     TYPFLG    /CLEAR ERROR FLAG
0531 3031          DCA     LPCNT     /SET ITERATION COUNT TO 4096(DECIMAL)
0532 4063          CAF          /INITIALIZE INTERFACE
0533 7340          CLA CLL CMA      /SET AC =7777
0534 4436          DBCO          /BIT CLEAR OUTPUT REGISTER
0535 3021          DCA     DATA1    /SAVE AC CONTENTS
0536 1021          TAD     DATA1    /GET AC CONTENTS
0537 7040          CMA          /COMPLIMENT DATA TO TEST FOR 7777
0540 7650          SNA CLA          /DID AC CHANGE

```

```

0541 5346      JMP      .+5          /DATA CORRECT, CONTINUE
0542 4443      JMS I   XERROR      /NO, ERROR
0543 3741      TRAN4E=1     /"DBCO CHANGED AC"
0544 3525      DH2=1       /"AC CONTENTS"
0545 7777      =1          /NUMBER OF WORDS TO BE OUTPUT
0546 4444      JMS I   XLOOP1     /CHECK FOR LOOP ON CURRENT TEST
0547 5332      JMP      TRAN4+2    /LOOP ON CURRENT TEST

```

/ DOES TRANSFER TO INPUT REGISTER CHANGE AC (WITH AC=7777, DBCI)

```

0550 3030      TRANS, DCA      TYPFLG    /CLEAR ERROR FLAG
0551 3031      DCA      LPCNT     /SET ITERATION COUNT TO 4096(DECIMAL)
0552 4063      CAF          /INITIALIZE INTERFACE
0553 7340      CLA CLL  CMA     /SET AC =7777
0554 4434      DBCI          /BIT CLEAR INPUT REGISTER
0555 3021      DCA      DATA1 /SAVE AC CONTENTS
0556 1021      TAD      DATA1 /GET AC CONTENTS
0557 7040      CMA          /COMPLIMENT DATA TO TEST FOR 7777
0560 7650      SNA CLA     /DID AC CHANGE
0561 5366      JMP      .+5          /DATA CORRECT, CONTINUE
0562 4443      JMS I   XERROR      /NO, ERROR
0563 3751      TRAN5E=1     /"DBCI CHANGED AC"
0564 3525      DH2=1       /"AC CONTENTS"
0565 7777      =1          /NUMBER OF WORDS TO BE OUTPUT
0566 4444      JMS I   XLOOP1     /CHECK FOR LOOP ON CURRENT TEST
0567 5352      JMP      TRAN5+2    /LOOP ON CURRENT TEST
0570 5777      JMP      TRAN6          /GO TO NEXT TEST

```

0577 0600
0600 PAGE

/ DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=0, DBSO)

```

0600 3030      TRANS, DCA      TYPFLG    /CLEAR ERROR FLAG
0601 3031      DCA      LPCNT     /SET ITERATION COUNT TO 4096 (DECIMAL)
0602 4063      CAF          /INITIALIZE INTERFACE
0603 4437      DBSO          /BIT SET OUTPUT REGISTER
0604 3021      DCA      DATA1 /SAVE AC CONTENTS
0605 1021      TAD      DATA1 /GET AC CONTENTS
0606 7650      SNA CLA     /WAS AC CHANGED
0607 5214      JMP      .+5          /DATA CORRECT, CONTINUE
0610 4443      JMS I   XERROR      /AC CHANGED, ERROR
0611 3731      TRAN3E=1     /"DBSO CHANGED AC"
0612 3525      DH2=1       /"AC CONTENTS"
0613 7777      =1          /NUMBER OF WORDS TO BE OUTPUT
0614 4444      JMS I   XLOOP1     /CHECK FOR LOOP ON CURRENT TEST
0615 5202      JMP      TRAN6+2    /LOOP ON CURRENT TEST

```

/ DOES TRANSFER TO OUTPUT REGISTER CHANGE AC (WITH AC=0, DBCO)

```

0616 3030      TRANS, DCA      TYPFLG    /CLEAR ERROR FLAG
0617 3031      DCA      LPCNT     /SET ITERATION COUNT TO 4096 (DECIMAL)
0620 4063      CAF          /INITIALIZE INTERFACE
0621 4436      DBCO          /BIT CLEAR OUTPUT REGISTER

```



```

0622 3021      DCA      DATA1      /SAVE AC CONTENTS
0623 1021      TAD      DATA1      /GET AC CONTENTS
0624 7650      SNA CLA      /IS AC STILL 0
0625 5232      JMP      ,+5      /DATA CORRECT, CONTINYE
0626 4443      JMS I   XERRCR      /NO, ERROR
0627 3741      TRAN4E=1      /"DBCO CHANGED AC"
0630 3525      DH2=1      /"AC CONTENTS"
0631 7777      =1      /NUMBER OF WORDS TO BE OUTPUT
0632 4444      JMS I   XLOOP1      /CHECK FOR LOOP ON CURRENT TEST
0633 5220      JMP      TRAN7+2      /LOOP ON CURRENT TEST
/
/DOES TRANSFER TO INPUT REGISTER CHANGE AC (WITH AC=0, DBCI)
/
0634 3030      TRAN8, DCA      TYPFLG      /CLEAR ERROR FLAG
0635 3031      DCA      LPCNT      /SET ITERATION COUNT TO 4096 (DECIMAL)
0636 4063      CAF      /INITIALIZE INTERFACE
0637 4434      DBCI      /BIT CLEAR INPUT REGISTER
0640 3021      DCA      DATA1      /SAVE AC CONTENTS
0641 1021      TAD      DATA1      /GET AC CONTENTS
0642 7650      SNA CLA      /IS AC STILL 0
0643 5250      JMP      ,+5      /DATA CORRECT, CONTINUE
0644 4443      JMS I   XERROR      /AC CHANGED, ERROR
0645 3751      TRAN5E=1      /"DBCI CHANGED AC"
0646 3525      DH2=1      /"AC CONTENTS"
0647 7777      =1      /NUMBER OF WORDS TO BE OUTPUTED
0650 4444      JMS I   XLOOP1      /CHECK FOR LOOP ON CURRENT TEST
0651 5236      JMP      TRAN8+2      /LOOP ON CURRENT TEST
0652 5777      JMP      OUT1      /GO TO NEXT TEST
0777 1000
1000 1000      PAGE

```

```

/
/CAN ALL BITS IN OUTPUT REGISTER BE SET (DBSO)
/DOES INITIALIZE CLEAR OUTPUT REGISTER
/
1000 3030      OUT1, DCA      TYPFLG      /CLEAR ERROR FLAG
1001 3031      DCA      LPCNT      /SET ITERATION COUNT TO 4096(DECIMAL)
1002 4063      CAF      /INITIALIZE INTERFACE
1003 7340      CLA CLL CMA      /SET AC =7777
1004 4437      DBSO      /BIT SET OUTPUT REGISTER
1005 4440      DBRO      /READ OUTPUT REGISTER
1006 3021      DCA      DATA1      /SAVE REGISTER DATA
1007 1021      TAD      DATA1      /GET REGISTER DATA
1010 7040      CMA      /COMPLIMENT DATA TO TEST FOR 7777
1011 7650      SNA CLA      /IS REGISTER=7777
1012 5217      JMP      ,+5      /DATA CORRECT, CONTINUE
1013 4443      JMS I   XERROR      /NO, ERROR
1014 3761      OUT1E=1      /"DBSO ERROR"
1015 3515      DH1=1      /"REGISTER DATA"
1016 7777      =1      /NUMBER OF WORDS TO BE OUTPUT
1017 4063      CAF      /INITIALIZE INTERFACE
1020 4440      DBRO      /READ OUTPUT REGISTER
1021 3021      DCA      DATA1      /SAVE REGISTER DATA
1022 1021      TAD      DATA1      /GET REGISTER DATA
1023 7650      SNA CLA      /WAS REGISTER CLEARED

```

```

1024 5231      JMP      ,+5          /DATA CORRECT, CONTINUE
1025 4443      JMS I   XERROR      /NO, ERROR
1026 3645      INIT1E-1    /"OUTPUT REG NOT CLEARED"
1027 3515      DH1-1      /"REGISTER DATA"
1030 7777      -1         /NUMBER OF WORDS TO BE OUTPUT
1031 4444      JMS I   XLOOP1  /CHECK FOR LOOP ON CURRENT TEST
1032 5202      JMP      OUT1+2  /LOOP ON CURRENT TEST
/
/CAN ALL BITS OF OUTPUT REGISTER BE CLEARED (DBCO)
/
1033 3030      OUT2,    DCA      TYPFLG  /CLEAR ERROR FLAG
1034 3031      DCA      LPCNT    /SET ITERATION COUNT TO 4096(DECIMAL)
1035 7340      CLA CLL  CMA      /SET AC =7777
1036 4437      DBSO                     /BIT SET OUTPUT REGISTER
1037 4436      DBCO                     /BIT CLEAR OUTPUT REGISTER
1040 7300      CLA CLL
1041 4440      DBRO                     /READ OUTPUT REGISTER
1042 3021      DCA      DATA1  /SAVE REGISTER DATA
1043 1021      TAD      DATA1  /GET REGISTER DATA
1044 7650      SNA CLA                     /WAS OUTPUT REGISTER CLEARED
1045 5252      JMP      ,+5          /DATA CORRECT, CONTINUE
1046 4443      JMS I   XERROR      /NO, ERROR
1047 3645      INIT1E-1    /"OUTPUT REGISTER NOT CLEARED"
1050 3515      DH1-1      /"REGISTER DATA"
1051 7777      -1         /NUMBER OF WORDS TO BE OUTPUT
1052 4444      JMS I   XLOOP1  /CHECK FOR LOOP ON CURRENT TEST
1053 5235      JMP      OUT2+2  /LOOP ON CURRENT TEST
/
/CAN EACH BIT OF OUTPUT REGISTER BE SET
/INDEPENDENTLY (DBSO)
/
1054 3030      OUT3,    DCA      TYPFLG  /CLEAR ERROR FLAG
1055 1176      TAD      L=62    /SET ITERATION COUNT
1056 3031      DCA      LPCNT    /TO 50(DECIMAL)
1057 3021      DCA      DATA1  /CLEAR TEST DATA
1060 4063      OUT3A,  CAF                     /INITIALIZE INTERFACE
1061 1021      TAD      DATA1  /GET TEST DATA
1062 4437      DBSO                     /BIT SET OUTPUT REGISTER
1063 7300      CLA CLL
1064 4440      DBRO                     /READ OUTPUT REGISTER
1065 3022      DCA      DATA2  /SAVE REGISTER DATA
1066 1021      TAD      DATA1  /GET TEST DATA
1067 7041      CIA
1070 1022      TAD      DATA2  /COMPARE TO REGISTER CONTENTS
1071 7650      SNA CLA                     /DO THEY COMPARE
1072 5277      JMP      ,+5          /DATA CORRECT, CONTINUE
1073 4443      JMS I   XERROR      /NO, ERROR
1074 3761      OUT1E-1    /"DBSO ERROR"
1075 3560      DH4-1      /"EXPECTED RECEIVED"
1076 7776      -2         /NUMBER OF WORDS TO BE OUTPUT
1077 4445      JMS I   XLOOP2  /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1100 5260      JMP      OUT3A  /LOOP WITH SAME DATA
1101 2021      ISZ      DATA1  /INCREMENT DATA PATTERN
1102 5260      JMP      OUT3A  /CONTINUE TEST

```

```

1103 4444      JMS I   XLOOP1      /CHECK FOR LOOP ON CURRENT TEST
1104 5257      JMP     OUT3+3      /LOOP ON CURRENT TEST
/
/CAN EACH BIT OF OUTPUT REGISTER BE CLEARED
/INDEPENDENTLY (DBCO)
/
1105 3030      OUT4:  DCA     TYPFLG      /CLEAR ERROR FLAG
1106 1176      TAD     [-62        /SET ITERATION COUNT
1107 3031      DCA     LPCNT      /TO 50 (DECIMAL)
1110 3021      DCA     DATA1     /CLEAR TEST DATA
1111 4063      OUT4A: CAF      /INITIALIZE INTERFACE
1112 1021      TAD     DATA1     /GET MASK
1113 7040      CMA     /COMPLIMENT TO GET EXPECTED RESULT
1114 3022      DCA     DATA2     /SAVE EXPECTED RESULT
1115 7040      CMA     /SET OUTPUT REGISTER TO 7777
1116 4437      DBSO    /BIT SET OUTPUT REGISTER
1117 7300      CLA CLL
1120 1021      TAD     DATA1     /GET PATTERN TO CLEAR OUTPUT REGISTER
1121 4436      DBCO    /BIT CLEAR OUTPUT REGISTER
1122 4440      DBRO    /READ OUTPUT REGISTER
1123 3023      DCA     DATA3     /SAVE REGISTER DATA
1124 1022      TAD     DATA2     /GET EXPECTED RESULT
1125 7041      CIA
1126 1023      TAD     DATA3     /COMPARE TO RECEIVED DATA
1127 7650      SNA CLA /WERE CORRECT BITS IN OUTPUT REGISTER CLEARED
1130 5335      JMP     ,+5        /DATA CORRECT, CONTINUE
1131 4443      JMS I   XERROR     /NO, ERROR
1132 3767      OUT4E=1 /"DBCO ERROR"
1133 3534      DH3=1 /MASK EXPECTED RECEIVED
1134 7775      =3 /NUMBER OF WORDS TO BE OUTPUT
1135 4445      JMS I   XLOOP2     /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1136 5311      JMP     OUT4A      /LOOP WITH SAME DATA
1137 2021      ISZ    DATA1     /INCREMENT DATA PATTERN
1140 5311      JMP     OUT4A      /CONTINUE TEST
1141 4444      JMS I   XLOOP1     /CHECK FOR LOOP ON CURRENT TEST
1142 5310      JMP     OUT4+3     /LOOP ON CURRENT TEST
1143 5777      JMP     OUT5
1177 1200
1200 1200

```

PAGE

```

/
/WITH OUTPUT REGISTER CLEARED, DOES CLEARING
/OUTPUT REGISTER CHANGE ANY BIT IN OUTPUT
/

```

```

1200 3030      OUT5:  DCA     TYPFLG      /CLEAR ERROR FLAG
1201 1176      TAD     [-62        /SET ITERATION COUNT
1202 3031      DCA     LPCNT      /TO 50 (DECIMAL)
1203 3021      DCA     DATA1     /CLEAR TEST DATA
1204 3022      DCA     DATA2     /CLEAR EXPECTED RESULT
1205 4063      OUT5A: CAF      /INITIALIZE INTERFACE
1206 1021      TAD     DATA1     /GET TEST DATA
1207 4436      DBCO    /BIT CLEAR OUTPUT REGISTER
1210 7300      CLA CLL
1211 4440      DBRO    /READ OUTPUT REGISTER
1212 3023      DCA     DATA3     /SAVE REGISTER DATA
1213 1023      TAD     DATA3     /GET REGISTER DATA

```

```

1214 7650      SNA CLA      /IS OUTPUT REGISTER 0
1215 5222      JMP      .+5      /DATA CORRECT, CONTINUE
1216 4443      JMS I    XERROR   /NO, ERROR
1217 3767      OUT4E-1     /"DBCO ERROR"
1220 3534      DH3-1      /"MASK EXPECTED RECEIVED"
1221 7775      -3          /NUMBER OF DATA WORDS
1222 4445      JMS I    XLOOP2   /TEST FOR LOOP ON SAME DATA
1223 5205      JMP      OUT5A    /LOOP WITH SAME DATA
1224 2021      ISZ     DATA1  /INCREMENT DATA PATTERN
1225 5205      JMP      OUT5A    /CONTINUE
1226 4444      JMS I    XLOOP1   /CHECK FOR LOOP ON CURRENT TEST
1227 5203      JMP      OUT5+3   /LOOP ON CURRENT TEST

```

/

/DOES SETTING OUTPUT REGISTER TWICE WITH SAME
/DATA CHANGE OUTPUT REGISTER

/

```

1230 3030      OUT6, DCA     TYPFLG   /CLEAR ERROR FLAG
1231 1176      TAD     [-62     /SET ITERATION COUNT
1232 3031      DCA     LPCNT   /TO 50 (DECIMAL)
1233 3021      DCA     DATA1  /CLEAR TEST DATA
1234 4063      OUT6A, CAF     /INITIALIZE INTERFACE
1235 1021      TAD     DATA1  /GET TEST DATA
1236 4437      DBSO                    /BIT SET OUTPUT REGISTER
1237 4437      DBSO                    /BIT SET OUTPUT REGISTER
1240 7300      CLA CLL
1241 4440      DBRO                    /READ OUTPUT REGISTER
1242 3022      DCA     DATA2  /SAVE REGISTER DATA
1243 1021      TAD     DATA1  /GET TEST DATA
1244 7041      CIA
1245 1022      TAD     DATA2  /COMPARE TO REGISTER DATA
1246 7650      SNA CLA      /ARE THEY THE SAME
1247 5254      JMP      .+5      /DATA CORRECT, CONTINUE
1250 4443      JMS I    XERROR   /NO, ERROR
1251 3767      OUT4E-1     /"DBSO ERROR"
1252 3560      DH4-1      /"EXPECTED RECEIVED"
1253 7776      -2          /NUMBER OF DATA WORDS
1254 4445      JMS I    XLOOP2   /TEST FOR LOOP ON SAME DATA
1255 5234      JMP      OUT6A    /LOOP WITH SAME DATA
1256 2021      ISZ     DATA1  /INCREMENT DATA PATTERN
1257 5234      JMP      OUT6A    /CONTINUE
1260 4444      JMS I    XLOOP1   /CHECK FOR LOOP ON CURRENT TEST
1261 5233      JMP      OUT6+3   /LOOP ON CURRENT TEST

```

/

/DOES READING OUTPUT REGISTER TWICE CHANGE
/OUTPUT REGISTER

/

```

1262 3030      OUT7, DCA     TYPFLG   /CLEAR ERROR FLAG
1263 1176      TAD     [-62     /SET ITERATION COUNT
1264 3031      DCA     LPCNT   /TO 50 (DECIMAL)
1265 3021      DCA     DATA1  /CLEAR TEST DATA
1266 4063      OUT7A, CAF     /INITIALIZE INTERFACE
1267 1021      TAD     DATA1  /GET TEST DATA
1270 4437      DBSO                    /BIT SET OUTPUT REGISTER

```

```

1271 7300      CLA CLL
1272 4440      DBRO          /READ OUTPUT REGISTER
1273 7300      CLA CLL
1274 4440      DBRO          /READ OUTPUT REGISTER
1275 3022      DCA      DATA2 /SAVE REGISTER DATA
1276 1021      TAD      DATA1 /GET TEST DATA
1277 7041      CIA
1300 1022      TAD      DATA2 /COMPARE TO REGISTER DATA
1301 7650      SNA CLA /ARE THEY THE SAME
1302 5307      JMP      ,+5 /DATA CORRECT, CONTINUE
1303 4443      JMS I   XERROR /NO, ERROR
1304 3775      OUT7E=1 /"DBRO ERROR"
1305 3560      DH4=1 /"EXPECTED RECEIVED"
1306 7776      =2 /NUMBER OF DATA WORDS
1307 4445      JMS I   XLOOP2 /TEST FOR LOOP WITH SAME DATA
1310 5266      JMP      OUT7A /LOOP WITH SAME DATA
1311 2021      ISZ      DATA1 /INCREMENT DATA PATTERN
1312 5266      JMP      OUT7A /CONTINUE
1313 4444      JMS I   XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1314 5265      JMP      OUT7+3 /LOOP ON CURRENT TEST

```

```

/
/DOES CLEARING OUTPUT REGISTER TWICE
/CHANGE ANY BIT IN OUTPUT REGISTER
/

```

```

1315 3030      OUT8, DCA      TYPFLG /CLEAR ERROR FLAG
1316 1176      TAD      C=62 /SET ITERATION COUNT
1317 3031      DCA      LPCNT /TO 50 (DECIMAL)
1320 3021      DCA      DATA1 /CLEAR TEST DATA
1321 3022      DCA      DATA2 /CLEAR EXPECTED RESULT
1322 4063      OUT8A, CAF /INITIALIZE INTERFACE
1323 1021      TAD      DATA1 /GET TEST DATA
1324 4437      DBSO /BIT SET OUTPUT REGISTER
1325 4436      DBCO /BIT CLEAR OUTPUT REGISTER
1326 4436      DBCO /BIT CLEAR OUTPUT REGISTER
1327 7300      CLA CLL
1330 4440      DBRO          /READ OUTPUT REGISTER
1331 3023      DCA      DATA3 /SAVE REGISTER DATA
1332 1023      TAD      DATA3 /GET REGISTER DATA
1333 7650      SNA CLA /IS REGISTER 0
1334 5341      JMP      ,+5 /DATA CORRECT, CONTINUE
1335 4443      JMS I   XERROR /NO, ERROR
1336 3767      OUT4E=1 /"DBCO ERROR"
1337 3534      DH3=1 /"MASK EXPECTED RECEIVED"
1340 7775      =3 /NUMBER OF DATA WORDS
1341 4445      JMS I   XLOOP2 /TEST FOR LOOP WITH SAME DATA
1342 5322      JMP      OUT8A /LOOP WITH SAME DATA
1343 2021      ISZ      DATA1 /INCREMENT DATA PATTERN
1344 5322      JMP      OUT8A /CONTINUE
1345 4444      JMS I   XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1346 5320      JMP      OUT6+3 /LOOP ON CURRENT TEST
1347 5777      JMP      IN1
1377 1400
1400

```

```

/
/CAN ALL BITS IN INPUT REGISTER BE SET
/DOES INITIALIZE CLEAR INPUT REGISTER
/
1400 3030 IN1, DCA TYPFLG /CLEAR ERROR FLAG
1401 3031 DCA LPCNT /SET ITERATION COUNT TO 4096(DECIMAL)
1402 4063 CAF /INITIALIZE INTERFACE
1403 7340 IN1A, CLA CLL CMA /SET AC =7777
1404 4437 DBSO /BIT SET OUTPUT REGISTER
1405 7300 CLA CLL
1406 4435 DBRI /READ INPUT REGISTER
1407 3021 DCA DATA1 /SAVE REGISTER DATA
1410 1021 TAD DATA1 /GET REGISTER DATA
1411 7040 CMA /COMPLIMENT TO TEST FOR 7777
1412 7650 SNA CLA /WAS INPUT REGISTER SET TO 7777
1413 5220 JMP .+5 /DATA CORRECT, CONTINUE
1414 4443 JMS I XERROR /NO, ERROR
1415 4011 IN3E=1 /"INPUT REGISTER NOT CORRECT"
1416 3515 DH1=1 /"REGISTER DATA"
1417 7777 =1 /NUMBER OF WORDS TO BE OUTPUT
1420 4063 IN1B, CAF /INITIALIZE INTERFACE
1421 4435 DBRI /READ INPUT REGISTER
1422 3021 DCA DATA1 /SAVE REGISTER DATA
1423 1021 TAD DATA1
1424 7650 SNA CLA
1425 5232 JMP .+5 /DATA CORRECT, CONTINUE
1426 4443 JMS I XERROR /NO, ERROR
1427 3661 INIT2E=1 /"INPUT REGISTER NOT CLEARED"
1430 3515 DH1=1 /"REGISTER DATA"
1431 7777 =1 /NUMBER OF WORDS TO BE OUTPUT
1432 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1433 5203 JMP IN1A /LOOP ON CURRENT TEST

```

```

/
/CAN ALL BITS IN INPUT REGISTER BE CLEARED (DBCI)
/
1434 3030 IN2, DCA TYPFLG /CLEAR ERROR FLAG
1435 3031 DCA LPCNT /SET ITERATION COUNT TO 4096(DECIMAL)
1436 4063 CAF /INITIALIZE INTERFACE
1437 7340 CLA CLL CMA /SET AC =7777
1440 0027 AND FJUMPER /MASK TO TEST ONLY FLIPFLOP BITS
1441 3021 DCA DATA1 /SAVE MASK
1442 3022 DCA DATA2 /SAVE EXPECTED RESULT
1443 1021 TAD DATA1 /GET MASK
1444 4437 DBSO /BIT SET OUTPUT REGISTER
1445 4434 DBCI /BIT CLEAR INPUT REGISTER
1446 7300 CLA CLL
1447 4435 DBRI /READ INPUT REGISTER
1450 3023 DCA DATA3 /SAVE REGISTER DATA
1451 1023 TAD DATA3 /COMPARE TO REGISTER DATA
1452 7650 SNA CLA /WERE CORRECT BITS CLEARED
1453 5260 JMP .+5 /DATA CORRECT, CONTINUE
1454 4443 JMS I XERROR /NO, ERROR
1455 4003 IN2E=1 /"DBCI ERROR"
1456 3534 DH3=1 /"MASK EXPECTED RECEIVED"

```

```

1457 7775          -3          /NUMBER OF WORDS TO BE OUTPUT
1460 4444          JMS I   XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1461 5236          JMP     IN2+2  /LOOP ON CURRENT TEST

/
/CAN EACH BIT OF INPUT REGISTER BE SET INDEPENDENTLY
/
1462 3030          IN3,   DCA     TYPFLG /CLEAR ERROR FLAG
1463 1176          TAD     [-62  /SET ITERATION COUNT
1464 3031          DCA     LPCNT  /TO 50(DECIMAL)
1465 3021          DCA     DATA1 /CLEAR TEST DATA
1466 4063          IN3A,  CAF     /INITIALIZE INTERFACE
1467 1021          TAD     DATA1 /GET TEST DATA
1470 4437          DBSO   /BIT SET OUTPUT REGISTER
1471 7300          CLA CLL
1472 4435          DBRI   /READ INPUT REGISTER
1473 3022          DCA     DATA2 /SAVE REGISTER DATA
1474 1021          TAD     DATA1 /GET TEST DATA
1475 7041          CIA
1476 1022          TAD     DATA2 /COMPARE TO RECEIVED DATA
1477 7650          SNA CLA /ARE THEY THE SAME
1500 5305          JMP     ,+5  /DATA CORRECT, CONTINUE
1501 4443          JMS I   XERROR /NO, ERROR
1502 4011          IN3E=1 /"INPUT REGISTER DATA ERROR"
1503 3560          DH4=1  /"EXPECTED RECEIVED"
1504 7776          -2          /NUMBER OF WORDS TO BE OUTPUT
1505 4445          JMS I   XLOOP2 /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1506 5266          JMP     IN3A  /LOOP WITH SAME DATA
1507 2021          ISZ     DATA1 /INCREMENT DATA PATTERN
1510 5266          JMP     IN3A  /CONTINUE TEST
1511 4444          JMS I   XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1512 5265          JMP     IN3+3 /LOOP ON CURRENT TEST
1513 5777          JMP     IN5
1577 1600          PAGE
1600 1600

```

```

/
/VERIFY THAT ALL LATCHING INPUT LINES HOLD DATA
/

```

```

1600 3030          IN5,   DCA     TYPFLG /CLEAR ERROR FLAG
1601 1176          TAD     [-62  /SET ITERATION COUNT
1602 3031          DCA     LPCNT  /TO 50(DECIMAL)
1603 3023          DCA     DATA3 /CLEAR TEST DATA
1604 4063          IN5A,  CAF     /INITIALIZE INTERFACE
1605 1023          TAD     DATA3 /GET TEST DATA
1606 0027          AND     FJUMPER /MASK OFF NON LATCHING BITS
1607 3021          DCA     DATA1 /SAVE AS EXPECTED RESULT
1610 1021          TAD     DATA1 /GET TEST DATA
1611 7450          SNA     /ARE ANY BITS TO BE TESTED
1612 5233          JMP     IN5C  /NO, GET NEXT DATA WORD
1613 4437          DBSO   /BIT SET OUTPUT REGISTER
1614 4436          DBCO   /BIT CLEAR OUTPUT REGISTER
1615 7300          CLA CLL
1616 4435          DBRI   /READ INPUT REGISTER

```

```

1617 3022      DCA      DATA2      /SAVE REGISTER DATA
1620 1021      TAD      DATA1      /GET EXPECTED RESULT
1621 7041      CIA
1622 1022      TAD      DATA2      /COMPARE TO RECEIVED DATA
1623 7650      SNA CLA      /ARE THEY THE SAME
1624 5231      JMP      ,+5          /DATA CORRECT, CONTINUE
1625 4443      JMS I    XERROR      /NO, ERROR
1626 4026      IN4E=1          /"LATCH ERROR"
1627 3560      DH4=1          /"EXPECTED RECEIVED"
1630 7776      =2            /NUMBER OF WORDS TO BE OUTPUT
1631 4445      JMS I    XLOOP2     /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1632 5204      JMP      IN5A        /LOOP WITH SAME DATA
1633 2023      IN5C,  ISZ      DATA3  /INCREMENT DATA PATTERN
1634 5204      JMP      IN5A        /CONTINUE TEST
1635 4444      JMS I    XLOOP1     /CHECK FOR LOOP ON CURRENT TEST
1636 5203      JMP      IN5+3      /LOOP ON CURRENT TEST
/

```

/VERIFY ALL NON LATCHING DATA BITS DO NOT HOLD DATA

```

1637 3030      IN6,  DCA      TYPFLG      /CLEAR ERROR FLAG
1640 1176      TAD      C=62        /SET ITERATION COUNT
1641 3031      DCA      LPCNT      /TO 50(DECIMAL)
1642 3024      DCA      DATA4     /CLEAR TEST DATA
1643 3022      DCA      DATA2     /CLEAR EXPECTED RESULT
1644 4063      IN6A, CAF          /INITIALIZE INTERFACE
1645 1027      TAD      FJUMPER    /GET MASK FOR NON LATCHING BITS
1646 7040      CMA
1647 0024      AND      DATA4     /CHANGE TO MASK OFF LATCHING BITS
1650 3021      DCA      DATA1     /SAVE FOR TRANSMISSION
1651 1021      TAD      DATA1     /GET TEST DATA
1652 7450      SNA
1653 5272      JMP      IN6C        /NO GET NEXT DATA WORD
1654 4437      DBSO
1655 4436      DBCO
1656 7300      CLA CLL
1657 4435      DBRI
1660 3023      DCA      DATA3     /READ INPUT REGISTER
1661 1023      TAD      DATA3     /SAVE REGISTER DATA
1662 7650      SNA CLA      /GET RECEIVED DATA
1663 5270      JMP      ,+5          /DID ANY BITS HOLD DATA
1664 4443      JMS I    XERROR      /DATA CORRECT, CONTINUE
1665 4026      IN4E=1          /YES, ERROR
1666 3534      DH3=1          /"LATCH ERROR"
1667 7775      =3            /"MASK EXPECTED RECEIVED"
1670 4445      JMS I    XLOOP2     /NUMBER OF WORDS TO BE OUTPUT
1671 5244      JMP      IN6A        /TEST FOR LOOP ON SAME DATA,ESCAPE ON DATA ERROR
1672 2024      IN6C,  ISZ      DATA4  /LOOP WITH SAME DATA
1673 5244      JMP      IN6A        /INCREMENT DATA PATTERN
1674 4444      JMS I    XLOOP1     /CONTINUE TEST
1675 5242      JMP      IN6+3      /CHECK FOR LOOP ON CURRENT TEST
/LOOP ON CURRENT TEST
/

```

/VERIFY THAT ALL LATCHING LINES CAN BE CLEARED INDEPENDENTLY


```

1676 3030 IN7, DCA TYPFLG /CLEAR ERROR FLAG
1677 1176 TAD [=62 /SET ITERATION COUNT
1700 3031 DCA LPCNT /TO 50(DECIMAL)
1701 3024 DCA DATA4 /CLEAR TEST
1702 4063 IN7A, CAF /INITIALIZE INTERFACE
1703 1027 TAD FJUMPER /GET MASK FOR LATCHING BITS
1704 0024 AND DATA4 /MASK OFF NON LATCHING BITS
1705 3021 DCA DATA1 /SAVE FOR TRANSMISSION
1706 3022 DCA DATA2 /EXPECTED RESULT
1707 1021 TAD DATA1 /SET OUTPUT REGISTER=7777
1710 4437 DBSO /BIT SET OUTPUT REGISTER
1711 4436 DBCO /BIT CLEAR OUTPUT REGISTER
1712 7300 CLA CLL
1713 1021 TAD DATA1 /GET TEST DATA
1714 4434 DBCI /BIT CLEAR INPUT REGISTER
1715 7300 CLA CLL
1716 4435 DBRI /READ INPUT REGISTER
1717 3023 DCA DATA3 /SAVE REGISTER DATA
1720 1023 TAD DATA3 /COMPARE TO RECEIVED DATA
1721 7650 SNA CLA /ARE THEY THE SAME
1722 5327 JMP ,+5 /DATA CORRECT, CONTINUE
1723 4443 JMS I XERROR /NO, ERROR
1724 4026 IN4E=1 /"LATCH ERROR"
1725 3534 DH3=1 /"MASK EXPECTED RECEIVED"
1726 7775 -3 /NUMBER OF WORDS TO BE OUTPUT
1727 4445 JMS I XLOOP2 /TEST FOR LOOP ON SAME DATA, ESCAPE ON DATA ERROR
1730 5302 JMP IN7A /LOOP WITH SAME DATA
1731 2024 IN7C, ISZ DATA4 /INCREMENT DATA PATTERN
1732 5302 JMP IN7A /CONTINUE TEST
1733 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
1734 5301 JMP IN7+3 /LOOP ON CURRENT TEST
1735 5777 JMP IN8 /GO TO NEXT TEST
1777 2000
2000 PAGE

```

```

/
/WITH THE INPUT REGISTER CLEARED, DOES CLEARING
/THE INPUT REGISTER SET ANY BIT IN INPUT
/

```

```

2000 3030 IN8, DCA TYPFLG /CLEAR ERROR FLAG
2001 1176 TAD [=62 /SET ITERATION COUNT
2002 3031 DCA LPCNT /TO 50 (DECIMAL)
2003 3021 DCA DATA1 /CLEAR TEST DATA
2004 3022 DCA DATA2 /CLEAR EXPECTED RESULT
2005 4063 IN8A, CAF /INITIALIZE INTERFACE
2006 1021 TAD DATA1 /GET TEST DATA
2007 4434 DBCI /BIT CLEAR INPUT REGISTER
2010 7300 CLA CLL
2011 4435 DBRI /READ INPUT REGISTER
2012 3023 DCA DATA3 /SAVE REGISTER DATA
2013 1023 TAD DATA3 /GET REGISTER DATA
2014 7650 SNA CLA /IS INPUT REGISTER 0
2015 5222 JMP ,+5 /DATA CORRECT, CONTINUE
2016 4443 JMS I XERROR /NO, ERROR

```

```

2017 4003      IN2E-1      /"DBCI ERROR"
2020 3534      DH3-1        /"MASK EXPECTED RECEIVED"
2021 7775      =3           /NUMBER OF DATA WORDS
2022 4445      JMS I  XLOOP2 /TEST FOR LOOP WITH SAME DATA
2023 5205      JMP      IN8A  /LOOP WITH SAME DATA
2024 2021      ISZ     DATA1 /INCREMENT DATA PATTERN
2025 5205      JMP      IN8A  /CONTINUE
2026 4444      JMS I  XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
2027 5203      JMP      IN8+3 /LOOP ON CURRENT TEST

```

/
/DOES READING THE INPUT REGISTER TWICE
/CHANGE THE INPUT REGISTER
/

```

2030 3030      IN9,   DCA      TYPFLG      /CLEAR ERROR FLAG
2031 1176      TAD      L=62      /SET ITERATION COUNT
2032 3031      DCA      LPCNT     /TO 50 (DECIMAL)
2033 3021      DCA      DATA1    /CLEAR TEST DATA
2034 4063      IN9A,  CAF      /INITIALIZE INTERFACE
2035 1021      TAD      DATA1    /GET TEST DATA
2036 4437      DBSO     /BIT SET OUTPUT REGISTER
2037 7300      CLA CLL
2040 4435      DBRI     /READ INPUT REGISTER
2041 7300      CLA CLL
2042 4435      DBRI     /READ INPUT REGISTER
2043 3022      DCA      DATA2    /SAVE REGISTER DATA
2044 1021      TAD      DATA1    /GET TEST DATA
2045 7041      CIA
2046 1022      TAD      DATA2    /COMPARE TO REGISTER DATA
2047 7650      SNA CLA  /ARE THEY THE SAME
2050 5255      JMP      .+5       /DATA CORRECT, CONTINUE
2051 4443      JMS I  XERROR     /NO, ERROR
2052 4034      IN9E-1 /"DBRI ERROR"
2053 3560      DH4-1 /"EXPECTED RECEIVED"
2054 7776      =2           /NUMBER OF DAT WORDS
2055 4445      JMS I  XLOOP2 /TEST FOR LOOP WITH SAME DATA
2056 5234      JMP      IN9A  /LOOP WITH SAME DATA
2057 2021      ISZ     DATA1 /INCREMENT DATA PATTERN
2060 5234      JMP      IN9A  /CONTINUE
2061 4444      JMS I  XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
2062 5233      JMP      IN9+3 /LOOP ON CURRENT TEST

```

/
/DOES CLEARING INPUT REGISTER TWICE SET ANY BIT
/IN INPUT REGISTER
/

```

2063 3030      IN10,  DCA      TYPFLG      /CLEAR ERROR FLAG
2064 1176      TAD      L=62      /SET ITERATION COUNT
2065 3031      DCA      LPCNT     /TO 50 (DECIMAL)
2066 3022      DCA      DATA2    /CLEAR EXPECTED RESULT
2067 3021      DCA      DATA1    /CLEAR TEST DATA
2070 4063      IN10A, CAF     /INITIALIZE INTERFACE
2071 1021      TAD      DATA1    /GET TEST DATA
2072 4437      DBSO     /BIT SET OUTPUT REGISTER
2073 4436      DBCO     /BIT CLEAR OUTPUT REGISTER

```

2074	4434	DBCI		/BIT CLEAR INPUT REGISTER
2075	4434	DBCI		/BIT CLEAR INPUT REGISTER
2076	7300	CLA CLL		
2077	4435	DBRI		/READ INPUT REGISTER
2100	3023	DCA	DATA3	/SAVE REGISTER DATA
2101	1023	TAD	DATA3	/GET REGISTER DATA
2102	7650	SNA CLA		/IS INPUT REGISTER 0
2103	5310	JMP	,+5	/DATA CORRECT, CONTINUE
2104	4443	JMS I	XERROR	/NO, ERROR
2105	4003	IN2E=1		/"DBCI ERROR"
2106	3534	DH3=1		/"MASK EXPECTED RECEIVED"
2107	7775	=3		/NUMBER OF DATA WORDS
2110	4445	JMS I	XLOOP2	/TEST FOR LOOP WITH SAME DATA
2111	5270	JMP	IN10A	/LOOP WITH SAME DATA
2112	2021	ISZ	DATA1	/INCREMENT DATA PATTERN
2113	5270	JMP	IN10A	/CONTINUE
2114	4444	JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
2115	5266	JMP	IN10+3	/LOOP ON CURRENT TEST
2116	5777	JMP	INOU1	/GO TO NEXT TEST
2177	2200			

PAGE

/

/WITH BOTH INPUT AND OUTPUT REGISTERS CLEARED

/DOES CLEARING OUTPUT SET

/ANY BIT IN INPUT

/

2200	3030	INOU1,	DCA	TYPFLG	/CLEAR ERROR FLAG
2201	1176		TAD	[=62	/SET ITERATION COUNT
2202	3031		DCA	LPCNT	/TO 50 (DECIMAL)
2203	3021		DCA	DATA1	/CLEAR TEST DATA
2204	3022		DCA	DATA2	/CLEAR EXPECTED RESULT
2205	4063	INOU1A,	CAF		/INITIALIZE INTERFACE
2206	1021		TAD	DATA1	/GET TEST DATA
2207	4436		DBCO		/BIT CLEAR OUTPUT REGISTER
2210	7300		CLA CLL		
2211	4435		DBRI		/READ INPUT REGISTER
2212	3023		DCA	DATA3	/SAVE REGISTER DATA
2213	1023		TAD	DATA3	/GET REGISTER DATA
2214	7650		SNA CLA		/IS OUTPUT REGISTER 0
2215	5222		JMP	,+5	/DATA CORRECT, CONTINUE
2216	4443		JMS I	XERROR	/NO, ERROR
2217	3767		OUT4E=1		/"DBCO ERROR"
2220	3534		DH3=1		/"MASK EXPECTED RECEIVED"
2221	7775		=3		/NUMBER OF DATA WORDS
2222	4445		JMS I	XLOOP2	/TEST FOR LOOP WITH SAME DATA
2223	5205		JMP	INOU1A	/LOOP WITH SAME DATA
2224	2021		ISZ	DATA1	/INCREMENT DATA PATTERN
2225	5205		JMP	INOU1A	/CONTINUE
2226	4444		JMS I	XLOOP1	/CHECK FOR LOOP ON CURRENT TEST
2227	5203		JMP	INOU1+3	/LOOP ON CURRENT TEST

/

/WITH BOTH INPUT AND OUTPUT REGISTERS CLEARED

/DOES CLEARING INPUT SET ANY BIT IN OUTPUT

```
2230 3030      /
                INOU2, DCA      TYPFLG      /CLEAR ERROR FLAG
2231 1176      TAD      [=62      /SET ITERATION COUNT
2232 3031      DCA      LPCNT      /TO 50 (DECIMAL)
2233 3021      DCA      DATA1     /CLEAR TEST DATA
2234 3022      DCA      DATA2     /CLEAR EXPECTED RESULT
2235 4063      INOU2A, CAF      /INITIALIZE INTERFACE
2236 1021      TAD      DATA1     /GET TEST DATA
2237 4434      DBCI      /BIT CLEAR INPUT REGISTER
2240 7300      CLA CLL
2241 4440      DBRO      /READ OUTPUT REGISTER
2242 3023      DCA      DATA3     /SAVE REGISTER DATA
2243 1023      TAD      DATA3     /GET REGISTER DATA
2244 7650      SNA CLA      /IS OUTPUT REGISTER 0
2245 5252      JMP      ,+5        /DATA CORRECT, CONTINUE
2246 4443      JMS I      XERROR    /NO, ERROR
2247 4003      IN2E=1      /"DBCI ERROR"
2250 3534      DH3=1      /"MASK EXPECTED RECEIVED"
2251 7775      =3        /NUMBER OF DATA WORDS
2252 4445      JMS I      XLOOP2    /TEST FOR LOOP WITH CURRENT DATA
2253 5235      JMP      INOU2A     /LOOP WITH SAME DATA
2254 2021      ISZ      DATA1     /INCREMENT DATA PATTERN
2255 5235      JMP      INOU2A     /CONTINUE
2256 4444      JMS I      XLOOP1    /CHECK FOR LOOP ON CURRENT TEST
2257 5233      JMP      INOU2+3    /LOOP ON CURRENT TEST
                /
                /WITH THE OUTPUT REGISTER SET TO ALL 1S, AND
                /THE INPUT REGISTER CLEARED, DOES SELECTIVELY
                /CLEARING THE OUTPUT REGISTER SET ANY BIT IN
                /THE INPUT REGISTER
                /
2260 3030      INOU3, DCA      TYPFLG      /CLEAR ERROR FLAG
2261 1176      TAD      [=62      /SET ITERATION COUNT
2262 3031      DCA      LPCNT      /TO 50 (DECIMAL)
2263 3021      DCA      DATA1     /CLEAR TEST DATA
2264 3022      DCA      DATA2     /CLEAR EXPECTED RESULT
2265 4063      INOU3A, CAF      /INITIALIZE INTERFACE
2266 7040      CMA      /SET AC=7777
2267 4437      DBSO      /BIT SET OUTPUT REGISTER
2270 4434      DBCI      /BIT CLEAR INPUT REGISTER
2271 7300      CLA CLL
2272 1027      TAD      FJUMPER    /GET FLIPFLOP JUMPER MASK
2273 7040      CMA
2274 3022      DCA      DATA2     /GET TEST DATA2
2275 1021      TAD      DATA1     /COMPLEMENT
2276 7040      CMA      /AND WITHN COMPLEMENT OF JUMPER MASK
2277 0022      AND      DATA2     /TO GET EXPECTED RESULT
2300 3022      DCA      DATA2     /GET TEST DATA
2301 1021      TAD      DATA1     /BIT CLEAR OUTPUT REGISTER
2302 4436      DBCO      /READ INPUT REGISTER
2303 7300      CLA CLL
2304 4435      DBRI      /SAVE REGISTER DATA
2305 3023      DCA      DATA3     /GET REGISTER DATA
2306 1023      TAD      DATA3
2307 7041      CIA
```

```

2310 1022      TAD      DATA2      /COMPARE TO EXPECTED RESULT
2311 7650      SNA CLA      /ARE THEY THE SAME
2312 5317      JMP      .+5          /DATA CORRECT, CONTINUE
2313 4443      JMS I    XERROR      /NO, ERROR
2314 3767      OUT4E-1      /"DBCO ERROR"
2315 3534      DH3-1      /"MASK EXPECTED RECEIVED"
2316 7775      -3          /NUMBER OF DATA WORDS
2317 4445      JMS I    XLOOP2      /TEST FOR LOOP WITH SAME DATA
2320 5265      JMP      INOU3A      /LOOP WITH SAME DATA
2321 2021      ISZ      DATA1      /INCREMENT DATA PATTERN
2322 5265      JMP      INOU3A      /CONTINUE
2323 4444      JMS I    XLOOP1      /CHECK FOR LOOP ON CURRENT TEST
2324 5263      JMP      INOU3+3      /LOOP ON CURRENT TEST
/
/ WITH THE INPUT REGISTER SET TO ALL 1S, DOES SELECTIVELY
/ CLEARING THE OUTPUT REGISTER CLEAR ANY BITS IN THE INPUT
/ REGISTER (EXCEPT THOSE NOT FLIPFLOPS)
/
2325 3030      INOU4, DCA      TYPFLG      /CLEAR ERROR FLAG
2326 1176      TAD      C=62          /SET ITERATION COUNT
2327 3031      DCA      LPCNT          /TO 50 (DECIMAL)
2330 3021      DCA      DATA1      /CLEAR TEST DATA
2331 4063      INOU4A, CAF      /INITIALIZE INTERFACE
2332 7040      CMA          /SET AC TO 7777
2333 4437      DBSO         /BIT SET OUTPUT REGISTER
2334 7300      CLA CLL
2335 1027      TAD      FJUMPER      /GET FLIPFLOP JUMPER MASK
2336 7040      CMA
2337 0021      AND      DATA1      /COMBINE WITH MASK
2340 7040      CMA
2341 3022      DCA      DATA2      /TO GET EXPECTED RESULT
2342 1021      TAD      DATA1      /GET TEST DATA
2343 4436      DBCO         /BIT CLEAR OUTPUT REGISTER
2344 7300      CLA CLL
2345 4435      DBRI         /READ INPUT REGISTER
2346 3023      DCA      DATA3      /SAVE REGISTER DATA
2347 1022      TAD      DATA2      /GET EXPECTED RESULT
2350 7041      CIA
2351 1023      TAD      DATA3      /COMPARE TO RECEIVED DATA
2352 7650      SNA CLA      /ARE THEY THE SAME
2353 5360      JMP      .+5          /DATA CORRECT, CONTINUE
2354 4443      JMS I    XERROR      /NO, ERROR
2355 3767      OUT4E-1      /"DBCO ERROR"
2356 3534      DH3-1      /"MASK EXPECTED RECEIVED"
2357 7775      -3          /NUMBER OF DATA WORDS
2360 4445      JMS I    XLOOP2      /TEST FOR LOOP WITH CURRENT DATA
2361 5331      JMP      INOU4A      /LOOP WITH SAME DATA
2362 2021      ISZ      DATA1      /INCREMENT DATA PATTERN
2363 5331      JMP      INOU4A      /CONTINUE
2364 4444      JMS I    XLOOP1      /CHECK FOR LOOP ON CURRENT TEST
2365 5330      JMP      INOU4+3      /LOOP ON CURRENT TEST
2366 5777      JMP
2377 2400

```

```

/
/VERIFY THAT EACH BIT SET UP TO SKIP DOES
/
2400 3030 INT1, DCA TYPFLG /CLEAR ERROR FLAG
2401 1176 TAD [-62 /SET ITERATION COUNT
2402 3031 DCA LPCNT /TO 50(DECIMAL)
2403 3022 DCA DATA2
2404 4063 INT1A, CAF /INITIALIZE INTERFACE
2405 1022 TAD DATA2
2406 0026 AND IJUMPER
2407 7450 SNA
2410 5233 JMP INT1D
2411 3021 DCA DATA1 /SAVE TEST DATA
2412 1021 TAD DATA1
2413 4437 DBSO /BIT SET OUTPUT REGISTER
2414 7300 CLA CLL
2415 4432 DBEI /ENABLE INTERFACE
2416 4076 SRQ /IS INTERRUPT ACTIVE
2417 5223 JMP INT1AE /NO, ERROR
2420 4433 DBSK /IS FLAG SET
2421 5240 JMP INT1BE /NO, ERROR
2422 5231 JMP INT10K /INTERRUPT ACTIVE, FLAG SET
2423 4433 INT1AE, DBSK /IS INTERFACE FLAG SET
2424 5245 JMP INT1CE
2425 4443 JMS I XERROR /NO, ERROR
2426 4042 INT1E-1
2427 3515 DH1-1
2430 7777 -1 /NUMBER OF WORDS TO BE OUTPUT
2431 4445 INT10K, JMS I XLOOP2 /TEST FOR LOOP ON SAME DATA, ESCAPE ON DATA ERROR
2432 5204 JMP INT1A /LOOP WITH SAME DATA
2433 2022 INT1D, ISZ DATA2 /INCREMENT DATA PATTERN
2434 5204 JMP INT1A /CONTINUE
2435 4444 JMS I XLOOP1 /CHECK FOR LOOP ON CURRENT TEST
2436 5203 JMP INT1+3 /LOOP ON CURRENT TEST
2437 5252 JMP INT3 /GO TO NEXT TEST
2440 4443 INT1BE, JMS I XERROR /NO, ERROR
2441 4065 INT3E-1
2442 3515 DH1-1
2443 7777 -1 /NUMBER OF WORDS TO BE OUTPUT
2444 5231 JMP INT10K
2445 4443 INT1CE, JMS I XERROR /NO, ERROR
2446 4077 INT4E-1
2447 3515 DH1-1
2450 7777 -1 /NUMBER OF WORDS TO BE OUTPUT
2451 5231 JMP INT10K

```

```

/
/VERIFY THAT EACH BIT NOT JUMPERD TO SKIP DOES NOT
/

```

```

2452 3030 INT3, DCA TYPFLG /CLEAR ERROR FLAG
2453 1176 TAD [-62 /SET ITERATION COUNT
2454 3031 DCA LPCNT /TO 50 (DECIMAL)
2455 3022 DCA DATA2 /CLEAR TEST DATA
2456 4063 INT3A, CAF /INITIALIZE INTERFACE

```

```

2457 1026      TAD      IJUMPER      /GET JUMPER MASK
2460 7040      CMA                      /COMPLIMENT FOR NO SKIP BITS
2461 0022      AND      DATA2      /GET BITS TO BE TESTED
2462 7450      SNA                      /ARE ANY BITS TO BE TESTED
2463 5300      JMP      INT3C      /NO, GET NEXT DATA PATTERN
2464 3021      DCA      DATA1      /SAVE FOR OUTPUT
2465 1021      TAD      DATA1      /GET TEST DATA
2466 4437      DBSO                      /BIT SET OUTPUT REGISTER
2467 7300      CLA CLL
2470 4433      DBSK                      /IS FLAG SET
2471 5276      JMP      ,+5      /NO, CONTINUE
2472 4443      JMS I   XERROR      /YES, ERROR
2473 3674      INITSE=-1      /"SKIP FLAG SET"
2474 3515      DH1=-1      /"REGISTER DATA"
2475 7777      -1
2476 4445      JMS I   XLOOP2
2477 5256      JMP      INT3A
2500 2022      INT3C, ISZ   DATA2
2501 5256      JMP      INT3A
2502 4444      JMS I   XLOOP1
2503 5255      JMP      INT3+3
2504 5777      JMP      EPASS

/
/ERROR HANDLER
/

2577 3257
2600 2600      PAGE
2600 0000      ERROR, 0
2601 7300      CLA CLL
2602 1600      TAD I   ERROR      /GET POINTER TO ERROR MESSAGE
2603 3234      DCA   MESSG      /SAVE POINTER
2604 2200      ISZ   ERROR
2605 1600      TAD I   ERROR      /GET POINTER TO DATA HEADER
2606 3236      DCA   DHDER      /SAVE HEADER
2607 2200      ISZ   ERROR      /NUMBER OF WORDS TO BE OUTPUT
2610 1600      TAD I   ERROR      /GET NUMBER OF DATA WORDS TO BE TYPED
2611 3264      DCA   DATCNT      /SAVE
2612 1200      TAD   ERROR      /GET ADDRESS OF TEST THAT FAILED
2613 1377      TAD   (=3
2614 3776      DCA   LSTDGT
2615 1776      TAD   LSTDGT
2616 3266      DCA   ERRAD
2617 7604      LAS
2620 0334      AND   SR01
2621 7640      SZA CLA
2622 5254      JMP   EHALT
2623 1030      TAD   TYPFLG
2624 7640      SZA CLA
2625 5241      JMP   DATOUT
2626 7040      CMA
2627 3030      DCA   TYPFLG      /CLEAR ERROR FLAG
2630 4775      JMS   OCTASC
2631 4441      JMS I   XPRINT
2632 3024      ERADR=-1

```

```

2633 4441 JMS I XPRINT
2634 0000 MSG, 0
2635 4441 JMS I XPRINT
2636 0000 DHDR, 0
2637 4441 JMS I XPRINT
2640 3512 CRLF-1
2641 1264 DATOUT, TAD DATCNT
2642 7650 SNA CLA
2643 5254 JMP EHALT
2644 1265 TAD DATAP
2645 3010 DCA POINT1
2646 1410 BITS, TAD I POINT1
2647 4774 JMS BITOUT
2650 2264 ISZ DATCNT
2651 5246 JMP BITS
2652 4441 JMS I XPRINT
2653 3512 CRLF-1
2654 7604 EHALT, LAS
2655 0333 AND SR00
2656 7640 SZA CLA
2657 5262 JMP +3
2660 1266 TAD ERRAD
2661 7402 HLT
2662 2200 ISZ ERROR
2663 5600 JMP I ERROR
2664 0000 DATCNT, 0
2665 0020 DATAP, DATA1-1
2666 0000 ERRAD, 0
/
/TEST FOR LOOP ON CURRENT TEST
/
2667 0000 LOOP1, 0
2670 1030 TAD TYPFLG
2671 7650 SNA CLA
2672 5277 JMP LP1EXA
2673 7604 LAS
2674 0341 AND SR06
2675 7640 SZA CLA
2676 5310 JMP LP1EXX-1
2677 7604 LP1EXA, LAS
2700 0340 AND SR05
2701 7640 SZA CLA
2702 5305 JMP LP1EXT
2703 2031 ISZ LPCNT
2704 5311 JMP LP1EXX
2705 7604 LP1EXT, LAS
2706 0335 AND SR02
2707 7650 SNA CLA
2710 2267 ISZ LOOP1
2711 5667 LP1EXX, JMP I LOOP1
/
/TEST FOR LOOP ON CURRENT DATA
/
2712 0000 LOOP2, 0
2713 1030 TAD TYPFLG

```



```

2714 7650      SNA CLA
2715 5326      JMP      LP2EXT
2716 7604      LAS
2717 0341      AND      SR06
2720 7650      SNA CLA
2721 5326      JMP      :+5      /DATA CORRECT, CONTINUE
2722 1312      TAD      LOOP2
2723 1373      TAD      (5
2724 3312      DCA      LOOP2
2725 5712      JMP I    LOOP2
2726 7604      LP2EXT, LAS
2727 0336      AND      SR03
2730 7650      SNA CLA
2731 2312      ISZ      LOOP2
2732 5712      JMP I    LOOP2
2733 4000      SR00, 4000
2734 2000      SR01, 2000
2735 1000      SR02, 1000
2736 0400      SR03, 400
2737 0200      SR04, 200
2740 0100      SR05, 100
2741 0040      SR06, 40
    
```

/OCTAL TO PACKED ASCII CONVERSION

```

2773 0005
2774 3031
2775 3000
2776 3027
2777 7775
3000 0000      PAGE /
3001 7300      OCTASC, 0
3002 1227      CLA CLL
3003 4046      TAD      LSTDGT      /GET WORD TO BE CONVERTED
3004 4212      RSW
3005 3226      JMS      SPLIT      /SWAP HALVES, SEPARATE DIGITS,
3006 1227      DCA      MSTDGT      /CONVERT MOST SIGNIFICANT
3007 4212      TAD      LSTDGT      /DIGITS TO ASCII
3008 3227      JMS      SPLIT      /CONVERT LEAST SIGNIFICANT
3009 3227      DCA      LSTDGT      /DIGITS TO ASCII
3010 5600      JMP I    OCTASC      /RETURN
3011 0000      SPLIT, 0
3012 0377      AND      (77
3013 3114      MQL
3014 4115      MQA
3015 7106      CLL RTL
3016 7004      RAL
3017 0376      AND      (707
3018 4115      MQA
3019 0376      AND      (707
3020 1375      TAD      (6060
3021 5612      JMP I    SPLIT
3022 3736      ERADR, TEXT /+? /
3023 4040
    
```

3027 4040
3030 4000

/
/OUTPUT 12 BIT BINARY WORD
/

3031	3000	BITOUT, 2		
3032	3114		MQL	/SAVE DATA IN MQ
3033	1374		TAD (-14	/SET UP TO OURPUT
3034	3020		DCA CNTR1	/12 BITS
3035	4115	BIT01,	MQA	/GET DATA
3036	7104		CLL RAL	/GET MSB INTO LINK
3037	3114		MQL	/SAVE REST OF WORD
3040	1373		TAD ("1	/GET ASCII 1 INTO AC
3041	7420		SNL	
3042	0372		AND ("0	/IS BIT=1
3043	4442		JMS I XTYPE	/NO, CHANGE TO ASCII 0
3044	2020		ISZ CNTR1	/OUTPUT BIT
3045	5235		JMP BIT01	/CONTINUE
3046	1371		TAD (240	/TYPE 2 SPACES
3047	4442		JMS I XTYPE	/AFTER LAST BIT HAS BEEN
3050	1371		TAD (240	/OUTPUTTED
3051	4442		JMS I XTYPE	
3052	5631		JMP I BITOUT	/RETURN

/

/CHARACTER STRING OUTPUT ROUTINE

3171 0240
3172 0260
3173 0261
3174 7764
3175 6060
3176 0707
3177 0077

3200	0000	PAGE PRINT, 0		
3201	7300		CLA CLL	
3202	1600		TAD I PRINT	/GET POINTER TO MESSAGE
3203	3010		DCA POINT1	/SET UP AUTO-INDEX REGISTER
3204	2200		ISZ PRINT	/SET UP RETURN
3205	1410		TAD I POINT1	/GET PACKED WORD
3206	3114		MQL	/SAVE IN MQ
3207	4115		MQA	/GET WORD
3210	4046		BSW	/SWAP HALVES
3211	4215		JMS TYPSET	/DECODE AND OUTPUT
3212	4115		MQA	/GET WORD
3213	4215		JMS TYPSET	/DECODE AND OUTPUT
3214	5205		JMP PRINT*5	/CONTINUE

/

/UNPACK, DECCDE, OUTPUT

3215	0000	TYPSET, 0		
3216	0243		AND K0077	/MASK UNWANTED BITS
3217	7450		SNA	/IS AC=0
3220	5600		JMP I PRINT	/YES, END OF MESSAGE, EXIT

3221	1244		TAD	M40	/SUBTRACT 40
3222	7510		SPA		/IS PACKED CHARACTER >40
3223	5226		JMP	,+3	/NO
3224	1250		TAD	K240	/YES, CONVERT TO ASCII
3225	5241		JMP	MTP	/OUTPUT
3226	7001		IAC		/ADD 1 TO AC
3227	7440		SZA		/IS CHARACTER=37
3230	5233		JMP	,+3	/NO,
3231	1245		TAD	K215	/GET CODE FOR CARRIAGE RETURN
3232	5241		JMP	MTP	/OUTPUT
3233	7001		IAC		/ADD 1 TO AC
3234	7440		SZA		/IS CHARACTER=37
3235	5240		JMP	,+3	/NO
3236	1246		TAD	K212	/GET CODE FOR LINE FEED
3237	5241		JMP	MTP	/OUTPUT
3240	1247		TAD	K336	/PACKED CHARACTER >40, CONVERT TO ASCII
3241	4442	MTP,	JMS I	XTYPE	/OUTPUT
3242	5615		JMP I	TYPSET	
3243	0077	K0077,		77	
3244	7740	M40,		-40	
3245	0215	K215,		215	
3246	0212	K212,		212	
3247	0336	K336,		336	
3250	0240	K240,		240	
				/	
				/OUTPUT ONE CHARACTER TO TTY	
				/	
3251	0000	TYPE,		?	
3252	6046		TLS		
3253	6041		TSF		
3254	5253		JMP	,-1	
3255	7200		CLA		
3256	5651		JMP I	TYPE	
				/	
				/	
				/	
3257	7604	EPASS,	LAS		
3260	0777		AND	SR04	
3261	7640		SZA	CLA	
3262	5776		JMP	INIT1	
3263	7604		LAS		
3264	0775		AND	SR05	
3265	7640		SZA	CLA	
3266	5272		JMP	EPAS1	
3267	4441		JMS I	XPRINT	
3270	3274		MEP-1		
3271	5776		JMP	INIT1	
3272	1374	EPAS1,	TAD	(007	
3273	4251		JMS	TYPE	
3274	5776		JMP	INIT1	
3275	3736	MEP,	TEXT	/+DR/	
3276	0422				
3277	0000				

/TELETYPE MESSAGES

3374 0207
3375 2740
3376 0400
3377 2737
3400 3400
3401 3736
3402 2305
3403 2440
3404 2322
3405 4006
3406 1722
3407 4004
3410 0526
3411 1103
3412 0540
3413 0317
3414 0405
3415 4001
3416 1604
3417 4003
3420 1716
3421 2400
3422 3736
3423 2305
3424 2440
3425 2322
3426 4006
3427 1722
3430 4011
3431 1624
3432 0522
3433 2225
3434 2024
3435 4012
3436 2515
3437 2005
3440 2223
3441 4001
3442 1604
3443 4003
3444 1716
3445 2400
3446 3736
3447 2305
3450 2440
3451 2327
3452 1124
3453 0310
3454 0523
3455 4006
3456 1722
3457 4006

PAGE
M1.

TEXT /->SET SR FOR DEVICE CODE AND CONT/

M2.

TEXT /->SET SR FOR INTERRUPT JUMPERS AND CONT/

M2A.

TEXT /->SET SWITCHES FOR FLIPFLOP JUMPERS AND CONTINUE/

3457 1411
3460 2006
3461 1417
3462 2040
3463 1225
3464 1520
3465 0522
3466 2340
3467 0116
3470 0440
3471 0317
3472 1624
3473 1116
3474 2505
3475 0000
3476 3736
3477 2305
3500 2440
3501 2322
3502 4006
3503 1722
3504 4022
3505 2516
3506 4001
3507 1604
3510 4003
3511 1716
3512 2400
3513 3736
3514 0000

3515 0000
3516 3736
3517 2205
3520 0711
3521 2324
3522 0522
3523 4004
3524 2124
3525 0100
3526 3736
3527 0103
3530 4003
3531 1716
3532 2405
3533 1624
3534 2300
3535 3736
3536 1501
3537 2313
3540 4040
3541 4040
3542 4040

M3, TEXT /*SET SR FOR RUN AND CONT/

CRLF, TEXT /*?/

/
/DATA HEADERS
/

DH0, 2
DH1, TEXT /*REGISTER DATA/

DH2, TEXT /*AC CONTENTS/

DH3, TEXT /*MASK EXPECTED RECEIVED/

3543 4040
3544 4040
3545 0530
3546 2005
3547 0324
3550 0504
3551 4040
3552 4040
3553 4040
3554 2205
3555 0305
3556 1126
3557 0504
3560 0000
3561 3736
3562 0530
3563 2005
3564 0324
3565 0504
3566 4040
3567 4040
3570 4040
3571 2205
3572 0305
3573 1126
3574 0504
3575 0000
3576 3736
3577 2205
3600 0711
3601 2324
3602 0522
3603 4040
3604 4040
3605 4040
3606 4004
3607 0124
3610 0140
3611 1725
3612 2440
3613 4040
3614 4040
3615 4040
3616 0401
3617 2401
3620 4011
3621 1600
3622 3736
3623 0103
3624 4003
3625 1716
3626 2405
3627 1624
3630 2340
3631 4040

DH4, TEXT /*-EXPECTED RECEIVED/

DH5, TEXT /*-REGISTER DATA OUT DATA IN/

DH6, TEXT /*-AC CONTENTS DATA OUT DATA IN/

3632 4004
 3633 0124
 3634 0140
 3635 1725
 3636 2440
 3637 4040
 3640 4040
 3641 4040
 3642 0401
 3643 2401
 3644 4011
 3645 1600

/
 /ERROR MESSAGE
 /

3646 1725 INIT1E, TEXT /OUTPUT REG NOT CLEARED/
 3647 2420

3650 2524
 3651 4022
 3652 0507
 3653 4016
 3654 1724
 3655 4003
 3656 1405
 3657 0122
 3660 0504

3661 0000
 3662 1116 INIT2E, TEXT /INPUT REG NOT CLEARED/
 3663 2025
 3664 2440

3665 2205
 3666 0740
 3667 1617
 3670 2440
 3671 0314
 3672 0501
 3673 2205
 3674 0400

3675 2313 INIT3E, TEXT /SKIP FLAG SET/
 3676 1120
 3677 4006

3700 1401
 3701 0740
 3702 2305
 3703 2400

3704 0402 TRAN1E, TEXT /DBRD DID NOT CLEAR AC/
 3705 2217
 3706 4004
 3707 1104
 3710 4016
 3711 1724
 3712 4003
 3713 1405
 3714 0122
 3715 4001

3716	0300		
3717	0402	TRAN2E, TEXT	/DBRI DID NOT CLEAR AC/
3720	2211		
3721	4004		
3722	1104		
3723	4016		
3724	1724		
3725	4003		
3726	1405		
3727	0122		
3730	4001		
3731	0300		
3732	0402	TRAN3E, TEXT	/DBSO CHANGED AC/
3733	2317		
3734	4003		
3735	1001		
3736	1607		
3737	0504		
3740	4001		
3741	0300		
3742	0402	TRAN4E, TEXT	/DBCO CHANGED AC/
3743	0317		
3744	4003		
3745	1001		
3746	1607		
3747	0504		
3750	4001		
3751	0300		
3752	0402	TRAN5E, TEXT	/DBCI CHANGED AC/
3753	0311		
3754	4003		
3755	1001		
3756	1607		
3757	0504		
3760	4001		
3761	0300		
3762	0402	OUT1E, TEXT	/DBSO ERROR/
3763	2317		
3764	4005		
3765	2222		
3766	1722		
3767	0000		
3770	0402	OUT4E, TEXT	/DBCO ERROR/
3771	0317		
3772	4005		
3773	2222		
3774	1722		
3775	0000		
3776	0402	OUT7E, TEXT	/DBRO ERROR/
3777	2217		
4000	4005		
4001	2222		
4002	1722		
4003	0000		
4004	0402	IN2E, TEXT	/DBCI ERROR/

4005 0311
4006 4005
4007 2222
4010 1722
4011 0000
4012 1116
4013 2025
4014 2440
4015 2205
4016 0711
4017 2324
4020 0522
4021 4004
4022 0124
4023 0140
4024 0522
4025 2217
4026 2200
4027 1401
4030 2403
4031 1040
4032 0522
4033 2217
4034 2200
4035 0402
4036 2211
4037 4005
4040 2222
4041 1722
4042 0000
4043 1116
4044 2405
4045 2222
4046 2520
4047 2440
4050 0103
4051 2411
4052 2605
4053 0000
4054 1617
4055 4011
4056 1624
4057 0522
4060 2225
4061 2024
4062 5440
4063 2313
4064 1120
4065 0000
4066 1116
4067 2405
4070 2222
4071 2520
4072 2454
4073 4016

IN3E, TEXT /INPUT REGISTER DATA ERROR/

IN4E, TEXT /LATCH ERROR/

IN9E, TEXT /DBRI ERROR/

INT1E, TEXT /INTERRUPT ACTIVE/

INT2E, TEXT /NO INTERRUPT, SKIP/

INT3E, TEXT /INTERRUPT, NO SKIP/

4074 1740
4075 2313
4076 1120
4077 0000
4100 1617
4101 4011
4102 1624
4103 0522
4104 2225
4105 2024
4106 5440
4107 1617
4110 4023
4111 1311
4112 2000
4113 1617
4114 4023
4115 1311
4116 2000
4117 2313
4120 1120
4121 0000

INT4E, TEXT /NO INTERRUPT, NO SKIP/

INT5E, TEXT /NO SKIP/

INT6E, TEXT /SKIP/

\$

0176 7716
0177 0106

4000	11111111	11111111	11111111	11111111	11111111	11111111	11111111	11111111
4100	11111111	11111111	11000000	00000000	00000000	00000000	00000000	00000000

4200
4300

4400
4500

4600
4700

5000
5100

5200
5300

5400
5500

5600
5700

6000
6100

6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

BIT01	3035	IN2	1434	K212	3246	SR00	2733
BITOUT	3031	IN2E	4004	K215	3245	SR01	2734
BITS	2646	IN3	1462	K240	3250	SR02	2735
BSW	4046	IN3A	1466	K336	3247	SR03	2736
CAF	4063	IN3E	4012	K7700	0062	SR04	2737
CNTR1	0020	IN4E	4027	LOOP1	2667	SR05	2740
CRLF	3513	IN5	1600	LOCP2	2712	SR06	2741
DATA1	0021	IN5A	1604	LP1EXA	2677	SRQ	4076
DATA2	0022	IN5C	1633	LP1EXT	2705	START1	0202
DATA3	0023	IN6	1637	LP1EXX	2711	START2	0244
DATA4	0024	IN6A	1644	LP2EXT	2726	TLOT	0250
DATA5	0025	IN6C	1672	LPCNT	0031	TRAN1	0452
DATAP	2665	IN7	1676	LSTDGT	3027	TRAN1E	3704
DATCNT	2664	IN7A	1702	M1	3400	TRAN2	0471
DATOUT	2641	IN7C	1731	M2	3421	TRAN2E	3717
DBCi	4434	IN8	2000	M2A	3445	TRAN3	0510
DBCIX	0300	IN8A	2005	M3	3476	TRAN3E	3732
DBCO	4436	IN9	2030	M40	3244	TRAN4	0530
DBCOX	0312	IN9A	2034	MEP	3275	TRAN4E	3742
DBDIX	0261	IN9E	4035	MESG	2634	TRAN5	0550
DBEI	4432	INIT1	0400	MQ	0114	TRAN5E	3752
DBEIX	0266	INIT1E	3646	MQA	4115	TRAN6	0600
DBRI	4435	INIT2	0416	ML	3114	TRAN7	0616
DBRIX	0305	INIT2E	3662	MSTDGT	3026	TRAN8	0634
DBRO	4440	INIT3	0435	MTP	3241	TYE1	0113
DBROX	0324	INIT3E	3675	OCTASC	3000	TYPE	3251
DBSK	4433	INO1	2200	OUT1	1000	TYPFLG	0030
DBSKX	0273	INO1A	2205	OUT1E	3762	TYPSET	3215
DBSO	4437	INO2	2230	OUT2	1033	XBCi	0034
DBSOX	0317	INO2A	2235	OUT3	1054	XBCO	0036
DH0	3515	INO3	2260	OUT3A	1060	XBEI	0032
DH1	3516	INO3A	2265	OUT4	1105	XBRI	0035
DH2	3526	INO4	2325	OUT4A	1111	XBRO	0040
DH3	3535	INO4A	2331	OUT4E	3770	XBSK	0033
DH4	3561	INT1	2400	OUT5	1200	XBSO	0037
DH5	3576	INT1A	2404	OUT5A	1205	XERROR	0043
DH6	3622	INT1AE	2423	OUT6	1230	XLOOP1	0044
DHDER	2636	INT1BE	2440	OUT6A	1234	XLOOP2	0045
DIOT	0260	INT1CE	2445	OUT7	1262	XPRINT	0041
EHALT	2654	INT1D	2433	OUT7A	1266	XTYPE	0042
EPAS1	3272	INT1E	4043	OUT7E	3776		
EPASS	3257	INT1QK	2431	OUT8	1315		
ERADR	3025	INT2E	4054	OUT8A	1322		
ERRAD	2666	INT3	2452	PNTR1	0233		
ERROR	2600	INT3A	2456	POINT1	0010		
FJUMPE	0027	INT3C	2500	PRINT	3200		
IJUMPE	0026	INT3E	4066	RETURN	0106		
IN1	1400	INT4E	4100	SIMBSW	0046		
IN10	2063	INT5E	4113	SIMCAF	0063		
IN10A	2070	INT6E	4117	SIMMQA	0115		
IN1A	1403	IOTS	0247	SIMSRQ	0076		
IN1B	1420	K0077	3243	SPLIT	3012		

ERRORS DETECTED: 0

LINKS GENERATED: 18

RUN-TIME: 10 SECONDS

3K CORE USED

