

IDENTIFICATION

DECTAPE

PRODUCT CODE: MAINDEC-15-DATCA-A-D  
PRODUCT NAME: TC02/15 BASIC EXERCISER,  
PART 2  
DATE CREATED: MARCH 17, 1972  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHOR: E. STEINBERGER, R. CHRISTOPHER  
REPLACES: MAINDEC-15-D3CB-D

12

COPYRIGHT © 1970, 1971 & 1972  
DIGITAL EQUIPMENT CORPORATION

# ABSTRACT

TCM2/15 BASIC EXERCISER PART 2 IS A SERIES OF TEST PROGRAMS THAT MAY BE USED TO GAIN A HIGH DEGREE OF CONFIDENCE IN THE DATA HANDLING ABILITY OF A TCM2/15 DECTAPE CONTROL AND ANY CONFIGURATION OF 1 TO 8 TU55/56 DECTAPE TRANSPORTS.

THE BASIC EXERCISER CONSISTS OF SEVERAL BASIC ROUTINES THAT MAY BE INDIVIDUALLY SELECTED. THE INSTRUCTION TEST OPERATES DRIVE R/W ONLY, AND REQUIRES THAT DRIVE R/W BE ON LINE AND WRITE ENABLED.

THE ROUTINES ARE:

- ROUTINE 1. TCM2/15 INSTRUCTION TEST
- ROUTINE 2. TCM2/15 API TEST

## 2. REQUIREMENTS

### 2.1 EQUIPMENT

PDP-15  
TCM2/15 DECTAPE CONTROL  
1 TO 8 TU55/56 DECTAPE TRANSPORTS  
API (OPTIONAL)

### 2.2 STORAGE

THE PROGRAM OCCUPIES CORE MEMORY UP TO ADDRESS 2500.

### 2.3 PRELIMINARY PROGRAMS

ALL PDP-15 CENTRAL PROCESSOR MAINTENANCE DIAGNOSTICS SHOULD RUN SUCCESSFULLY BEFORE ATTEMPTING TO RUN THIS PROGRAM.

## 3. LOADING PROCEDURE

PLACE THE BINARY TAPE IN THE READER  
SET THE ADDRESS SWITCHES TO 177MM  
SET THE BANK MODE SWITCH TO A 1  
PRESS I/O RESET  
PRESS READ IN

4. STARTING PROCEDURE  
-----

4.1 CONTROL SWITCH SETTINGS  
-----

THE PARTICULAR ROUTINE TO BE RUN IS SELECTED BY PLACING THE NUMBER OF THE ROUTINE IN AC SWITCH REGISTER BITS 16 AND 17.

AC 16-17 -----	ROUTINE -----	READ SECTION -----
00	NO TEST	
01	TC02/15 INSTRUCTION TEST	7.1
10	TC02/15 API TEST	7.2
11	NO TEST	

4.2 STARTING ADDRESS  
-----

200

4.3 PROGRAM AND/OR OPERATOR ACTION  
-----

- A. SET THE ADDRESS SWITCHES TO 200
- B. SELECT THE ROUTINE TO BE RUN VIA AC SWITCHES 16-17
- C. PRESS I/O RESET
- D. PRESS START

5. OPERATING PROCEDURE  
-----

5.1 OPERATIONAL SWITCH SETTINGS  
-----

AC SWITCH -----	FUNCTION -----
0=0	ENABLE TTY MESSAGES
0=1	DELETE TTY MESSAGES

READ EACH OF THE INDIVIDUAL TEST DESCRIPTIONS TO DETERMINE IF ANY OTHER SWITCH SETTINGS APPLY TO THAT PARTICULAR ROUTINE.

5.1.1 TEST ROUTINES AND APPLICABLE RUN SWITCHES

-----  
TEST ROUTINE

AC SW

FUNCTION  
-----

TC02/15 INSTRUCTION TEST

8=1

HALT AT END OF NON-  
MOTION TESTS (IF TAPE  
MOTION HAS NOT BEEN  
TESTED, ROUTINE 0 (PART 1)  
SHOULD BE UTILIZED BEFORE  
ATTEMPTING TAPE MOTION  
PORTION OF THE INSTRU-  
TION TEST.)

TC02/15 API TEST

9=1

HALT AT END OF TEST.

5.2 SUBROUTINE ABSTRACTS

-----  
N/A

5.3 PROGRAM AND/OR OPERATOR PROCEDURE

-----  
THIS SERIES OF ROUTINES IS DESIGNED FOR INITIAL CHECKOUT OF A  
TC02/15 DECTAPE CONTROL AND ITS ASSOCIATED DRIVES, OR MAINTENANCE  
AND REPAIR OF THE CONTROL AND DRIVES AFTER INSTALLATION.

THE FOLLOWING PROCEDURE COULD BE USED FOR INITIAL CHECKOUT OF  
THE CONTROL AND DRIVES, AND TO AID IN THE REPAIR OF MALFUNCTIONS  
ONCE THE CONTROL AND DRIVES HAVE BEEN OPERATING.

5.3.1 OPERATION CHECK

-----  
THE FIRST ROUTINE UTILIZED IS THE TC02/15 INSTRUCTION TEST. IT IS  
USED TO VERIFY CONTROL OPERATIONS.

### 5.3.1.1 INITIAL CONTROL STATE

-----  
WHEN POWER IS INITIALLY APPLIED TO THE TC02/15 CONTROL, STATUS A, THE ERROR AND DECTAPE FLAGS, AND THE DATA FLAG CAN COME UP IN ANY STATE. A SHORT MANUAL PROCEDURE WILL PREVENT ERASING DECTAPES AND HAVING TO RELOAD PROGRAMS.

PRESS I/O RESET

NOW EXAMINE THE TC02/15 INDICATOR PANEL. THE FOLLOWING INDICATORS SHOULD ALL BE OFF, INDICATING A 0 STATE.

DTF (DECTAPE FLAG)  
DF (DATA FLAG)  
ALL ERROR FLAGS  
W (WREN WRITE ENABLE)  
STATUS A BIT 4 (MOTION)  
US (UP TO SPEED)  
C0 TO C2 CAN BE IN ANY STABLE STATE (NOT COUNTING)  
ALL STATE REGISTER BITS EXCEPT 1 SHOULD BE 0, AND  
BIT 1 SHOULD BE A 1 (STATE IDLE)

### 5.3.1.2 TC02/15 CONTROL TEST

-----  
THE FIRST HALF OF THE TC02/15 INSTRUCTION TEST VERIFIES THE EXISTENCE OF ALL DECTAPE IOTS EXCEPT DIOF (SKIP IF DTF=1), AND ALSO VERIFIES THAT THE STATUS A REGISTER BITS FUNCTION PROPERLY AND WILL HOLD ALL COMBINATIONS OF DATA.

SET THE ADDRESS SWITCHES TO 200  
SET THE AC SWITCHES TO 1001  
PRESS I/O RESET  
SET DRIVE 8/0 ON LINE AND WRITE ENABLED  
PRESS START

THE PROGRAM TESTS AND VERIFIES ALL CONTROLS FUNCTIONS THAT DO NOT REQUIRE TAPE MOTION, AND HALTS AT ADDRESS 522.

A HALT AT ANY OTHER ADDRESS INDICATES A CONTROL ERROR. CONSULT THE PROGRAM LISTING FOR THE ERROR DEFINITION.

15.3.1.2 TC02/15 CONTROL TEST CONTINUED)

AT THIS POINT, IF TAPE MOTION HAS ALREADY BEEN DEFINED, AND IT IS NOT A "RAW" CONTROL THAT IS BEING TESTED:

SET DRIVE 8/0 ON LINE AND WRITE ENABLED  
PRESS CONTINUE

THE PROGRAM WILL VERIFY THE FUNCTIONAL OPERATIONS OF THE TC02/15 CONTROL FOR ALL DECTAPE OPERATIONS EXCEPT WRTM, TYPEOUT "END" ON THE TTY AND HALT AT ADDRESS 1561.

A HALT AT ANY OTHER ADDRESS INDICATES A CONTROL ERROR. CONSULT THE PROGRAM LISTING FOR THE ERROR DEFINITION.

5.3.2 CHECK API OPTION

-----  
RUN TEST ROUTINE 2 TO VERIFY THAT THE API WORKS IN A STATIC CONDITION. ONCE TEST 2 RUNS IN ITS ENTIRETY, ROUTINES 4, 5, 6, AND 7 (PART 1), SHOULD BE RUN WITH THE API OPTION SELECTED (AC SWITCH 12=1 AT START FROM 200).

AFTER ONE PASS THROUGH THE API TEST ROUTINE 2, THE PROGRAM WILL TYPE "API END". IF AC SWITCH 8=1 THE PROGRAM WILL REPEAT, IF AC SWITCH 8=0 THE PROGRAM WILL HALT.

6. ERRORS

-----  
NEITHER THE TC02/15 INSTRUCTION TEST OR THE TC02/15 API TEST HAVE ERROR TYPEOUTS AND ALL HARDWARE MALFUNCTIONS DETECTED RESULT IN A PROCESSOR HALT AT A SPECIFIC ADDRESS.

7. PROGRAM DESCRIPTION

7.1 TC02/15 INSTRUCTION TEST (ROUTINE 1)

-----  
THE TC02/15 INSTRUCTION TEST IS SEGMENTED INTO TWO MAJOR DIVISIONS:

- A. THOSE FUNCTIONAL OPERATIONS THAT CAN BE TESTED WITHOUT TAPE MOTION.
- B. THOSE FUNCTIONS THAT CAN ONLY BE TESTED UTILIZING TAPE MOTION.

IF AC SWITCH 8=1, THE PROCESSOR WILL HALT AFTER COMPLETION OF THE NON-MOTION TESTS AND BEFORE PROCEEDING TO THE TESTS THAT REQUIRE TAPE MOTION.

BOTH SERIES OF TESTS REQUIRE DRIVE 8/0 TO BE ON LINE AND WRITE ENABLED, AND DO NOT USE ANY OTHER DRIVES.

[7.1 TC02/15 INSTRUCTION TEST (ROUTINE 1) CONTINUED]

THE CONTROL AND NO TAPE MOTION PORTION OF THE TESTS TEST THE PERFORMANCE OF THE FOLLOWING:

I/O RESET CLEARS ALL TC02/15 FLAGS  
EXISTENCE OF ALL DECTAPE I/O'S EXCEPT DTSE  
BIT AND DATA TEST ON DECTAPE STATUS A  
IORS BIT 10=0 WITH ALL DECTAPE FLAGS=0  
FR 1-3=7 TO GENERATE A SELECT ERROR  
XSA DLY TO BE NOT GREATER THAN SUS  
SELECT ERROR TO GENERATE ERROR FLAG  
IORS BIT 10=1 IF EF=1  
DTEF TO SKIP IF EF IS A 1  
DTDF TO NOT SKIP IF EF IS A 1  
NO PROGRAM INTERRUPT WITH EF=1 AND ENI=0  
DTXA WITH AC BIT 10=1 INHIBITS CLEARING EF  
DTXA WITH AC BIT 10=0 DOES CLEAR EF  
SE GOING TO 1 GENERATES ERR STP AND STATUS A BIT 4 IS  
CLEARED BY ERR STP  
EF AND STATUS A BIT 9 (ENI) ON A 1 ALLOW PROGRAM  
INTERRUPT  
DOES CAF GENERATE I/O POWER CLEAR AND CLR EF

THE TAPE MOTION TESTS VERIFY THE PERFORMANCE OF THE FOLLOWING OPERATIONS:

FIRST TEST  
-----

MOVE FUNCTION  
-----

MOVE TAPE BACKWARDS  
DOES DATA BREAK NOT OCCUR IN MOVE  
WILL EZ BE DETECTED  
DOES EZ GENERATE EF  
DOES EZ GENERATE ERR STP AND CLEAR TAPE MOTION  
(STATUS A BIT 4)  
DTXA WITH BIT 10=0 SHOULD CLEAR EZ  
CAF SHOULD CLEAR EZ

(7.1 TC02/15 INSTRUCTION TEST (ROUTINE 1) CONTINUED)

SECOND TEST      WRITE ALL (BACKWARDS)  
-----

DOES WRITE ALL NOT CAUSE SE  
WILL WRITE ALL CAUSE A DATA BREAK (SHOULD GET AT  
LEAST 9 TPD'S BEFORE EZ. AT LEAST ONE OF  
THEM SHOULD CAUSE Z TO DB.)  
IS WC LOCATION +1 AT DATA BREAK  
IS CA LOCATION +1 AT DATA BREAK IN WRITE ALL  
WILL DECTAPE FLAG (DTF) SET AT WRITE ALL DATA  
BREAK  
IS WRITE ALL TRANSFER DIRECTION OUT AND NOT IN  
WILL DTXA WITH AC BIT 11=1 NOT CLEAR DTF  
DOES DTF=1 MAKE IORS BIT 10 A 1  
WILL DTF=1 AND ENI=1 (STATUS A BIT 9) ALLOW  
PROGRAM INTERRUPT  
WILL DTF=1 AND ENI=0 NOT ALLOW PROGRAM  
PROGRAM INTERRUPT  
WILL DTF SKIP WITH DTF=1  
WILL DTF NOT SKIP WITH DTF=1 AND EF=0  
CAF SHOULD CLEAR THE DTF

THIRD TEST      READ ALL (FORWARD)  
-----

WILL DATA BREAK OCCUR IN READ ALL  
DOES CA LOCATION INCREMENT IN READ ALL  
WILL 1 TO DTF SET TIM WITH DTF ALREADY EQUAL TO 1  
DOES TIM SET EF LEVEL  
DOES TIM GENERATE ERR STP AND CLEAR MOTION  
DTXA WITH BIT 10=1 SHOULD CLEAR DTF BUT NOT  
TIM  
CAF SHOULD CLEAR TIM

FOURTH TEST     READ ALL C MODE  
-----

DOES READ ALL C MODE ALLOW DATA BREAK  
DTF DOES NOT SET AT WCO IN C MODE  
DTF DOES SET AT WCO (WORD COUNT OVERFLOW)  
WC=0 SHOULD NOT ALLOW ANY MORE DATA BREAKS  
CAF SHOULD CLEAR TAPE MOTION

(7.1 TC02/15 INSTRUCTION TEST (ROUTINE 1) CONTINUED)

FIFTH TEST           SEARCH NORMAL MODE  
-----

DOES SEARCH ALLOW DATA BREAK  
DOES DTF=1 AFTER A SEARCH DATA BREAK  
CA LOCATION SHOULD NOT INCREMENT IN SEARCH  
TRANSFER DIRECTION OF DATA SHOULD BE INPUT

SIXTH TEST           SEARCH C MODE  
-----

DOES SEARCH C MODE ALLOW DATA BREAK  
DOES DTF NOT SET TO 1 WITHOUT WCO IN C MODE  
DOES DTF SET TO 1 AT SEARCH C MODE AND WCO

SEVENTH TEST        READ DATA NORMAL MODE  
-----

DOES READ DATA NOT ALLOW DATA BREAKS UNTIL  
STATE DATA  
WHILE IN STATE DATA DO DATA BREAKS OCCUR EVERY  
200MS (APPROXIMATELY)  
DOES STATE CHECK GOING TO 0 SET 1 TO DTF

EIGHTH TEST         READ DATA C MODE  
-----

DOES READ DATA ALLOW 256 DATA BREAKS PER BLOCK  
STATE CHECK GOING TO 0 SHOULD NOT SET 1 TO DTF  
IF WC=1  
WCO (WORD COUNT OVERFLOW) BY ITSELF SHOULD NOT  
SET 1 TO DTF  
STATE CHECK GOING TO 0, WC=0 AND READ DATA  
C MODE SHOULD SET 1 TO DTF

NINTH TEST  
-----

WRITE DATA NORMAL MODE  
-----

WRITE DATA SHOULD NOT ALLOW DATA BREAKS UNTIL  
STATE REVERSE CHECK  
ONE DATA BREAK SHOULD OCCUR DURING STATE REVERSE  
CHECK  
DATA BREAKS SHOULD OCCUR APPROXIMATELY EVERY  
200MS IN STATE DATA  
DOES WREN GO TO A 1 AT STATE DATA (NO DATA  
BREAKS IF IT DOESN'T)  
WRITE DATA SHOULD NOT ALLOW DATA BREAKS DURING  
STATE FINAL  
WRITE DATA SHOULD NOT ALLOW DATA BREAKS DURING  
STATE CHECK  
WRITE DATA NORMAL MODE AND STATE CHECK GOING TO  
0 SHOULD SET 1 TO DTF  
WREN SHOULD GO TO 0 DURING STATE IDLE AND NOT  
ALLOW ANY MORE DATA BREAKS  
AGAIN TEST WRITE DATA TO ALLOW 1 DATA BREAK  
AT STATE REVERSE CHECK  
WREN SHOULD NOT GO TO A 1 AT STATE DATA IF DTF  
IS ALREADY A 1  
DTXA, WRITE DATA AND NOT STATE IDLE OR STATE  
BLOCK MARK SHOULD CAUSE A TIMING ERROR  
TIMING ERROR SHOULD CLEAR US VIA ERR STOP WHICH  
IN TURN CLEARS REGISTER TO IDLE, ANOTHER  
DTXA AND WRITE DATA SHOULD NOT CAUSE A  
SECOND TIMING ERROR

TENTH TEST  
-----

WRITE DATA C MODE  
-----

TEST ONLY 256 DATA BREAKS PER BLOCK WRITE DATA  
STATE CHECK GOING TO 0 SHOULD NOT SET 1 TO DTF  
IF WC=1 (NO WCO)  
STATE CHECK GOING TO 0 SHOULD SET 1 TO DTF IF  
WC=0 (AFTER WCO)

7.2 TC02/15 API TEST (ROUTINE 2)

-----  
THE TC02/15 API TEST IS A STATIC TEST OF THE TC02/15 INTERRUPTS VIA THE API AND SHOULD RUN IN ITS ENTIRETY BEFORE ATTEMPTING TO EXERCISE THE OTHER TEST ROUTINE UTILIZING THE API (AC SWITCH 12=1 AT START FROM 200).

INITIALLY, THE API TEST LOADS ADDRESSES 30 TO 77 WITH A SERIES OF LAW'S (30=LAW 30 TO 77=LAW 77). AFTER EACH DECTAPE API BREAK THE AC SHOULD EQUAL LAW 44. AC SWITCH 8=0 IS HALT AT END OF TEST, AC SWITCH 8=1 IS REPEAT API TEST.

THE SEQUENCE OF TESTS MADE ARE AS FOLLOWS:

WITH ALL DECTAPE FLAGS EQUAL TO 0 API SHOULD NOT BREAK  
WITH EF=1 AND API OFF SHOULD NOT BREAK  
API ON AND EF=1 SHOULD BREAK TO ADDRESS 44  
DBK AND NEW EF=1 SHOULD BREAK TO ADDRESS 44  
API OFF AND DBK WITH NEW EF=1 SHOULD NOT ALLOW BREAK  
DECTAPE SHOULD NOT BREAK WITH PRIORITY 0 ACTIVE  
DBK FROM PRIORITY 0 ACTIVE SHOULD ALLOW BREAK TO ADDRESS  
44  
API REQUEST ON PRIORITY 7 SHOULD NOT INTERFERE WITH  
DECTAPE API REQUEST  
API ON-API OFF SHOULD NOT ALLOW A BREAK  
API ON-API OFF SHOULD NOT CAUSE DECTAPE API REQUEST  
TO BE LAST  
DBK-API OFF SHOULD NOT ALLOW API BREAK  
DBK-API OFF SHOULD NOT CAUSE DECTAPE REQUEST TO BE LAST  
API SHOULD HAVE PRIORITY OVER PROGRAM INTERRUPT  
DBK SHOULD ALLOW SECOND BREAK FROM SAME EF AND PIE  
SHOULD NOT OCCUR  
IF ENI=0 (DECTAPE STATUS A BIT 9) AND EF=1  
NO API BREAK SHOULD OCCUR  
DECTAPE FLAG=1 AND ENI=1 AND API ON SHOULD BREAK TO  
ADDRESS 44  
DECTAPE FLAG=1 AND API ON SHOULD OVERRIDE PROGRAM  
INTERRUPT TO ADDRESS 0

8. PROGRAM LISTING

.TITLE \* MAINDEC-15-DATCA-A \* MARCH 17, 1972 \*

.ABS

/TC02 BASIC EXERCISER PDP-15 - PART2

/COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

/EQUATE DECTAPE IOT'S

```

707541 DTCA=707541
707542 DTRA=707542
707544 DTXA=707544
707545 DTLA=DTCA!DTXA
707561 DTEF=707561
707562 DTRB=707562
707601 DTDF=707601
/EQUATE DECTAPE FUNCTIONS (STAT A)
021000 SRCHFV=21000
061000 SRCHBW=61000
060000 MOVBNW=60000
022000 RDATAF=22000
005000 WRALL=5000
003000 RDALL=3000
/BIT CONSTANTS EQUATED
020000 GOBIT=20000
040000 DIRBIT=40000
100000 EZBIT=100000
010000 MODEBT=10000
020000 PARBIT=20000
000100 DTFBIT=100
000400 ENABLI=400
775000 BLKTIM=LAW -3000 /ABOUT 36 MSEC
777400 BLENTH=LAW -400 /-256 FOR BLOCK WC -400 OCTAL
000030 WCLOC=30 /WORD COUNT
000031 CALOC=31 /CURRENT ADDRESS
006400 BUFFER=6400
007000 BUFFER2=BUFFER+400
000300 INHCLR=300
006377 BLKFND=BUFFER-1
007400 BUFFER3=BUFFER2+400
700004 CLOF=700004
700044 CLON=700044
700301 KSF=700301
700402 TCF=700402
700406 TLS=700406
700401 TSF=700401

```

/

/IN CASE OF CAL

```

00020 .LOC 20
00020 000020 20
00021 700304 IORS=10
00022 700301 KSF
00023 741000 SKP
00024 740040 HLT
00025 700042 ION
00026 703344 DBR
00027 620020 JMP* 20

```

.EJECT

```

/STARTING ADDRESS IS 200
/SET SW REG SW16 TO SW17 SELECT TEST
/=1 IS TC02 INSTRUCTION TEST USES DRIVE 8 ONLY
/= 2 IS API TEST
/ SWITCH W IS DELETE ALL TYPEDUTS
/SELECT TESTS SETUP FOR
/ FIRST DRIVE 0 TO DRIVE TABLES
      .LOC 200
TC02TS  LAS
      AND (3
      TAD (JMP* TSTTBL
      DAC GOTST
      LAC (JMP* 2
      DAC 1
GOTST   JMP* TSTTBL      /LOC CHANGE TO + TEST NUMBER
      HLT
      JMP TC02TS
TSTTBL  GOTST+1
      INSTST
      APITST
      GOTST+1
NOTTBL= .
      .LOC TSTTBL+17
MSBITS  0
UNFUNC  0
      .EJECT

```

```

00200
00200 750004
00201 502300
00202 342301
00203 040206
00204 202302
00205 040001
00206 620211
00207 740040
00210 600200
00211 000207
00212 000232
00213 001563
00214 000207
      000215
00230
00230 000000
00231 000000

```

```

/PART 1 VERIFIES STAT B
/IS CLEARED BY I/O RESET AT START
/PART 2 REQUIRED DRIVE B ON-LINE WRITE EN
00232 707572 INSIST DTRB+10 /ALL FLAGS
00233 740200 SZA /SHOULD BE CLEAR
00234 740040 INER01 HLT /BUT ARE NOT
/ALSO - IORS - BIT 10 SHOULD = 0
00235 700314 IORS /READ STATUS
00236 502303 AND (200 /MASK BIT 10
00237 740200 SZA /SHOULD = 0
00240 740040 INER02 HLT /IORS FAILED
/
/
/SKIP ON DECTAPE FLAG SHOULD NOT SKIP
00241 750000 CLA
00242 707601 DTDF /SHOULD NOT SKIP
00243 740001 CMA /DTF = 0
00244 741200 SNA /SHOULD HAVE EXC CMA
00245 740040 INER03 HLT /AC = 0 IS SKIP ERROR
/
/
/SKIP ON ERROR FLAG SHOULD NOT SKIP
00246 750000 CLA
00247 707561 DTEF /SHOULD NOT SKIP EF = 0
00250 740001 CMA /SHOULD EXECUTE CMA
00251 741200 SNA /OID IT
00252 740040 INER04 HLT /NO AC = 0 IS SKIP ERROR
/
/PROGRAM INTERRUPT SHOULD NOT
/OCCUR ALL DECTAPE FLAGS = 0
00253 202304 LAC (JMP INER05-1 /INT JMP
00254 040001 DAC 1
00255 750000 CLA /MAKE 0 FOR TEST
00256 700042 ION /ENABLE
00257 740000 NOP /SHOULD NOT BREAK
00260 700002 IOF /SHOULD EXECUTE
00261 740001 CMA /SHOULD EXECUTE
00262 741200 SNA /AC = 0 IS INT
00263 740040 INER05 HLT /INTERUPTED IN ERROR
/
/STATUS A CLEAR AND READ 0'S
00264 750000 CLA
00265 707541 DTCA /SHOULD CLEAR STAT A
00266 707552 DTRA+10 /SHOULD READ 0'S
00267 740200 SZA /AC NOT = 0 IS ERROR
00270 740040 INER06 HLT /CLEAR STAT A OR READ ERROR
.EJECT

```

```

/SEE IF ANY STATUS A BITS SET TO 1
/AND READ BACK TO AC (ANY AT ALL)
00271 750000          CLA
00272 707541          DTCA          /CLEAR STAT A
00273 777777          LAW -1
00274 707544          DTXA          /SHOULD READ A FEW 1'S
00275 750000          CLA
00276 707542          DTRA          /SHOULD READ A FEW 1'S
00277 741200          SNA          /ANYTHING BUT ALL W OK
00300 740040          INEK07 HLT          /EITHER DID NOT SET OR READ
/
/
/STATUS A BIT TESTS
/FIRST BIT TEST DOES IT CLEAR
/DO ALL TESTS FOR 1 BIT THEN RPT
/START WITH BIT 9 AND GO TO BIT 0
00301 202305          SBTST LAC (400
00302 040010          DAC 10
/
/BIT CLEAR TEST
00303 777777          LAW -1
00304 040017          DAC 17          /2ND PASS SWITCH
00305 750000          CLA
00306 707541          DTCA          /CLEAR STAT A
00307 707552          DTRA+10        /READ A
00310 500010          AND 10         /MASK BIT
00311 740200          SZA          /IS IT 0
00312 740040          SAER00 HLT          /NO DISPLAY FAILED BIT
/
/NOW TEST THAT THE BIT WILL SET
/TO A 1 THEN LOOP BACK AND TEST CLEAR AGAIN
/BIT SET TEST WILL IT GO TO A 1
/AND READ BACK AS A 1
/
00313 200010          LAC 10
00314 707544          DTXA          /BIT SHOULD SET TO 1
00315 707552          DTRA+10        /AND READ TO AC
00316 500010          AND 10         /BIT SHOULD = 1
00317 240010          XOR 10         /NOW SHOULD = 0
00320 740200          SZA
00321 740040          SAER01 HLT          /DISPLAY FAILED BIT
00322 440017          ISZ 17          /2ND PASS CLEAR AND SET
00323 741000          SKP          /YES
00324 600305          JMP SBTST+4
                     .EJECT

```

/NOW TEST WILL THE BIT GO 1 TO 0  
 /FIRST CLEAR SET TO 1 COMP TO 0  
 /

00325	750000	CLA	
00326	707541	DTCA	/CLEAR A
00327	200010	LAC 10	
00330	707544	DTXA	/SET BIT TO 1
00331	707544	DTXA	/SHOULD MAKE IT 0
00332	707552	DTRA+10	/AND READ 0
00333	500010	AND 10	
00334	740200	SZA	/BIT GO TO 0
00335	740040	SAER02 HLT	/DISPLAY FAILED BIT

/

/NOW TEST WILL THE BIT NOT GO  
 /TO A 1 WITH THE AC BIT = 0  
 /

00336	750000	CLA	
00337	707541	DTCA	/CLEAR ALL
00340	200010	LAC 10	
00341	740001	CMA	/MAKE AC BIT = 0
00342	707544	DTXA	/STAT A BIT SHOULD STAY 0
00343	707552	DTRA+10	/READ IT
00344	500010	AND 10	/MASK IT
00345	740200	SZA	/BIT SHOULD BE 0
00346	740040	SAER03 HLT	/DISPLAY FAILED BIT

/

/NOW TEST WILL THE BIT  
 /REMAIN A 1 WITH THE AC BIT = 0  
 /

00347	750000	CLA	
00350	707541	DTCA	/CLEAR STAT A
00351	200010	LAC 10	
00352	542306	SAD (020000	
00353	600364	JMP SAER04+1	/CAN NOT DO BIT 4
00354	707544	DTXA	/MAKE BIT = 1
00355	740001	CMA	
00356	707544	DTXA	/SHOULD NOT CHANGE IT TO 0
00357	707552	DTRA+10	/READ STAT A
00360	500010	AND 10	/MASK BIT SHOULD = 1
00361	240010	XOR 10	/XOR SHOULD GO TO 0
00362	740200	SZA	
00363	740040	SAER04 HLT	/DISPLAY BIT THAT DID NOT STAY 1

/

00364	200010	LAC 10	/FOR NEXT BIT
00365	744010	CLLIRAL	/POSITION
00366	740200	SZA	/DONE ALL BITS
00367	600362	JMP SBTST+1	/NO .EJECT

```

/STATUS A DATA TEST
/WILL STAT A HOLD ALL
/COMBINATIONS EXCEPT BIT 4 = 0
/
00370 140010 SADATA DZM 10
00371 200010 LAC 10 /GET DATA
00372 502307 AND (757400) /MASK BIT 4
00373 040011 DAC 11 /SAVE FOR COMPARE
00374 707545 DTLA /LOAD A
00375 707552 DTRA+10 /READ A
00376 540011 SAD 11 /CORRECT RESULT
00377 500403 JMP .+4 /YES
00400 740040 SAER06 HLT /DISPLAY FAILED SA
00401 200011 LAC 11 /DISPLAY CORRECT SA
00402 740040 HLT
00403 200010 LAC 10
00404 342305 TAD (400) /DATA INCREMENT
00405 040010 DAC 10
00406 740200 SZA /DONE ALL COMBINATIONS
00407 600371 JMP SADATA+1 /NO
/
/
/NOW TEST STAT B OPERATION
/A LITTLE FARTHER
/
/FIRST TEST DOES CAF
/CLEAR ALL STAT B FLAGS
/
00410 703302 SBTEST CAF /CLEAR ALL
00411 707572 DIRB+10 /READ STAT B
00412 740200 SZA /ANY 1 BITS ARE ERROR
00413 740040 SBER00 HLT /DISPLAY FAILED SB
/
/
/NOW TEST WILL SELECT ERROR
/SET AND READ BACK TO AC
/AS 440000 BIT 0 AND BIT 3 ONLY
/XSA OLY MIGHT BE MORE THAN 5 MICROSEC
00414 202310 LAC (7000)
00415 707545 DTLA
00416 600417 JMP .+1 /1
00417 600420 JMP .+1 /2
00420 600421 JMP .+1 /3
00421 600422 JMP .+1 /4 MICROSEC
00422 707572 DIRB+10 /5+ MICROSEC
00423 242311 XOR (440000) /AC SHOULD = 440000
00424 740200 SZA /NOW SHOULD = 0
00425 740040 SBER01 HLT /DISPLAY FAILED BITS
.EJECT

```

/ERROR FLAG IS AC BIT 10 ON IORS

00426 700314  
 00427 502303  
 00430 242303  
 00431 740200  
 00432 740040

      IGRS  
       AND (200)  
       XOR (200)  
       SZA  
 SBER1A HLT                /IGRS FAILED ON EF = 1  
 /  
 /SE = 1 DTEF SHOULD SKIP EF=1  
 /

00433 750000  
 00434 707561  
 00435 740001  
 00436 740200  
 00437 740040

      CLA  
       DTEF                /SHOULD SKIP EF = 1  
       CMA                /SHOULD NOT BE EXECUTED  
       SZA                /AC NOT 0 THEN NO SKIP  
 SBER02 HLT                /SKIP FAILED TO SKIP  
 /  
 /PROGRAM INTERRUPT SHOULD NOT OCCUR  
 /STAT A BIT 9 = 0  
 /

00440 202312  
 00441 040001  
 00442 750000  
 00443 700042  
 00444 740000  
 00445 700002  
 00446 740001  
 00447 741200  
 00450 740040

      LAC (JMP SBER03-1  
       DAC 1                /IN CASE INT  
       CLA  
       ION                /ENABLE PIE  
       NOP                /SHOULD NOT BREAK  
       IOF                /DISABLE PIE  
       CMA                /SHOULD BE EXECUTED  
       SNA                /IF AC = 0  
 SBER03 HLT                /THEN INTERRUPT IN ERROR  
 /  
 /DTSF SHOULD NOT SKIP DTF = 0  
 /

00451 750000  
 00452 707601  
 00453 740001  
 00454 741200  
 00455 740040

      CLA  
       DTDF                /SHOULD NOT SKIP  
       CMA                /DTF = 0 SHOULD EXECUTE CMA  
       SNA                /AC = 0 IS SKIP ERROR  
 SBER04 HLT  
 /  
 /DTXA WITH BIT 10 = 1 SHOULD  
 /NOT CLEAR STAT B ERRORS  
 /

00456 202303  
 00457 707544  
 00460 707572  
 00461 741200  
 00462 740040

      LAC (200  
       DTXA                /XOR INH CLEAR EF  
       DIRB+10            /READ STATUS B  
       SNA                /SHOULD NOT BE 0  
 SBER05 HLT                /STAT B CLEARED  
       .EJECT

```

/DTXA WITH BIT 10 = 0 SHOULD CLEAR
00463 750000 CLA
00464 707545 DTLA /SHOULD CLEAR BIT 10 = 0
00465 717572 DTRB+10 /READ STAT B
00466 740200 SZA /AC SHOULD = 0
00467 740040 SBER05 HLT /DISPLAY FAILED SE
/
/SE GOING TO 1 SHOULD CLEAR SA BIT 4
/IS ERROR STOP GENERATED
/
00470 202313 LAC (27400 /ALSO ENI BIT
00471 707545 DTLA /SHOULD CAUSE SE
00472 600473 JMP .+1 /WAIT 5
00473 600474 JMP .+1 /MICROSEC
00474 600475 JMP .+1
00475 600476 JMP .+1
00476 707552 DTRA+10
00477 502306 AND (20000
00500 740200 SZA /GO SHOULD BE 0
00501 740040 SBER07 HLT /ERROR STOP NOT GEN
/
/EF=1 SE=1 AND SA BIT 9 = 1
/PIE SHOULD CAUSE PROGRAM INTERRUPT
/
00502 202314 LAC (JMP SBER08-1
00503 040001 DAC 1
00504 750000 CLA
00505 700042 ION /PIE ON
00506 740000 NOP /WAIT SHOULD BREAK
00507 700002 IOF /PIE OFF (NOT EXC)
00510 740001 CMA /SHOULD NOT BE EXC
00511 740200 SZA /INT OCCUR IF AC = 0
00512 740040 SBER08 HLT /INT FAIL AC NOT 0
/
/CAF SHOULD CLEAR SE IN STAT B
/
00513 703302 CAF /SHOULD CLEAR SE
00514 707572 DTRB+10 /SHOULD READ 0'S
00515 740200 SZA
00516 740040 SBER09 HLT /CAF DID NOT CLEAR
/
/END OF TESTS THAT CAN BE MADE
/WITHOUT TAPE MOTION
/MALT IF SWS = 1000
/
00517 750004 LAS
00520 502315 AND (1000
00521 740200 SZA
00522 740040 HLT
.EJECT

```

```

/START TAPE MOTION TESTS
/DO A MOVE BACKWARD INTO EZ
/MONITOR *C LOCATION IN CASE DB
/OCCURS EZ SHOULD OCCUR WITHIN
/45 SECONDS EZ IS ONLY LEGAL STATUS
/

```

00523	202310	TMOIST	LAC (00000)	
00524	707545		DILA	
00525	777777		LAW -1	/SET UP
00526	040030		DAC WCLOC	/CA AND
00527	202317		LAC (BUFFER	/AC TO
00530	040031		DAC CALOC	/PREVENT CLUBBER
00531	202320		LAC (-32	
00532	040010		DAC 10	
00533	140011		DZM 11	
00534	200030		LAC WCLOC	/WATCH OUT
00535	740100		SMA	/FOR DATA BREAK
00536	740040	TMER00	HLT	/AC SHOULD NOT INCREMENT
00537	707572		DTRB+10	
00540	740200		SZA	/STAT 6 ANY YET
00541	600545		JMP .+4	/YES
00542	707552		DTRA+10	
00543	502300		AND (20000	
00544	741200		SNA	/HAS GO CLEARED
00545	600553		JMP .+6	/YES
00546	440011		ISZ 11	
00547	600534		JMP TMER00-2	
00550	440010		ISZ 10	/WAITED 45 SECONDS
00551	600534		JMP TMER00-2	/NO
00552	740040	TMER01	HLT	/NOTHING HAPPENED
00553	707572		DTRB+10	/READ STATUS
00554	242321		XOR (500000	/SHOULD MAKE EZ AND EF = 0
00555	740200		SZA	/ANY OTHER = 1 IS ALSO ERROR
00556	740040	TMER02	HLT	/DISPLAY BITS IN ERROR
00557	707552		DTRA+10	
00560	502300		AND (20000	/ERROR STOP SHOULD
00561	740200		SZA	/CLEAR GO BIT
00562	740040	TMER03	HLT	/EZ DID NOT CLEAR MOT
00563	200030		LAC WCLOC	
00564	740100		SMA	/WAS DATA BREAK MADE
00565	740040	TMER04	HLT	/YES DATA BREAK IN ERROR
		/		
		/		/CAF SHOULD CLEAR EZ AND EF
		/		
00566	703302		CAF	/SHOULD CLEAR EZ
00567	707572		DTRB+10	/READ STATUS
00570	740200		SZA	/AC SHOULD = 0
00571	740040	TMER05	HLT	
			.EJECT	

```

/NO* SET EZ AGAIN
/
00572 202306          LAC (20000
00573 707544          DTXA
00574 707572          DTRB+10      /WAIT FOR EZ
00575 741200          SNA
00576 600574          JMP .-2
/
/DTXA WITH AC BIT 10 = 0 SHOULD CLEAR EZ
/
00577 750000          CLA
00600 707544          DTXA      /SHOULD CLEAR STAT 8
00601 707572          DTRB+10  /READ STATUS
00602 740200          SZA      /BITS SHOULD GO TO 0
00603 740040          TMR006  HLT
/
/WRITE ALL SHOULD NOT CAUSE
/A SELECT ERROR
/USE WRITE ALL NORMAL MODE
00604 202322          LAC (45400      /ALSO ENI
00605 707545          DTLA      /LOAD WRITE ALL
00606 600607          JMP .+1      /WAIT FOR
00607 600610          JMP .+1      /XSA DELAY
00610 600611          JMP .+1      /TO GO AWAY
00611 600612          JMP .+1
00612 707572          DTRB+10  /READ STATUS
00613 740200          SZA      /NO FLAGS SHOULD BE SET
00614 740040          TMR07  HLT
/
/NOW MAKE TAPE GO BKWD
/DATA BREAK SHOULD BE MADE WITHIN
/250 MILLISECONDS AND DTF SHOULD BE SET
/
00615 777777          LAW =1
00616 046400          DAC BUFFER
00617 040030          DAC WCLOC
00620 202323          LAC (BUFFER-1
00621 040031          DAC CALOC
00622 202306          LAC (20000
00623 707544          DTXA
00624 202324          LAC (-55000
00625 040010          DAC 10
00626 200030          LAC WCLOC
00627 740100          SMA
00630 600637          JMP .+7      /WC GO TO 0
00631 707572          DTRB+10  /YES
00632 740200          SZA      /ANY STATUS YET
00633 600637          JMP .+4      /YES
00634 440010          ISZ 10     /WAITED 250 MSEC
00635 600626          JMP .-7      /NO
00636 740040          TMR08  HLT      /NOTHING HAPPENED 250 MSEC
                          .EJECT

```

```

/ WC = 777777 IS NO BREAK
/ N IS 1 BREAK
/ POSITIVE AND NOT = 0 IS MORE THAN 1 BREAK
/ TEST WC TO HAVE INCR
/ FROM WRITE ALL
/

```

```

00637 200030          LAC +CLUC          /WC SHOULD COUNT
00640 740200          SZA              /ONCE AND ONLY ONCE
00641 740040  TMRK9  HLT              /WC = 1'S IS MORE THAN 1 BREAK
/
/

```

```

/ CA SHOULD BE INCREMENTED
/ ONCE AND ONLY ONCE
/

```

```

00642 200031          LAC CALUC
00643 542323  SAD (BUFFER-1  /DID CA CHANGE
00644 740040  HLT              /NO DID NOT +1
00645 542317  SAD (BUFFER  /CA ONLY +1
00646 741000  SKP              /YES
00647 740040  TMR10 HLT          /CA INCREMENTED INCORRECT
/
/

```

```

/ WAIT FOR EZ INTERRUPT
/

```

```

00650 707572          DTRB+10
00651 740100          SMA
00652 600650          JMP .-2
/
/

```

```

/ NOW TEST FLAGS EF EZ AND DTF SHOULD = 1
/ DTF SHOULD SET DURING WA DATA BREAK
/ ALL OTHER FLAGS SHOULD = 0
/

```

```

00653 242325          XOR (500100
00654 740200          SZA
00655 740040  TMR11 HLT          /DISPLAY FLAGS IN ERROR
/
/

```

```

/ WRITE ALL SHOULD NOT HAVE INPUT
/ BUFFER SHOULD STILL BE 777777
/

```

```

00656 206400          LAC BUFFER          /GET WORD WRITTEN
00657 740001          CMA
00660 740200          SZA              /WAS IT ALTERED
00661 740040  TMR12 HLT          /WRITE ALL INPUT
          .EJECT

```

/THE DECTAPE FLAG SHOULD NOT CLEAR  
 /WITH AC BIT 11 = 1 AND DTXA  
 /EZ SHOULD CLEAR AC BIT 10 = 0  
 /

00662	202326	LAC (100	
00663	707544	DTXA	/DTF SHOULD NOT CLEAR
00664	707572	DTRB+10	/READ STATUS
00665	242326	XOR (100	/SHOULD CLEAR DTF POS
00666	740200	SZA	/ALL BITS SHOULD = 0
00667	740040	TMER13 HLT	/DISPLAY BITS IN ERROR

/

/

/DTF = 1 SHOULD READ BACK IORS BIT 10  
 /

00670	700314	IORS	/READ I/O STATUS
00671	502303	AND (200	/MASK BIT 10
00672	242303	XOR (200	/SHOULD MAKE IT 0
00673	740200	SZA	
00674	740040	TMER14 HLT	/IORS FAILED

/

/

/STATUS A BIT 9=1 SHOULD INT WITH PIE = 1  
 /

00675	202327	LAC (JMP TMER15-1	
00676	040001	DAC 1	
00677	750000	CLA	
00700	700042	ION	/PIE = 1
00701	740000	NOP	/SHOULD EXECUTE 7 THE BREAK
00702	700002	IOP	/SHOULD NOT BE EXECUTED
00703	740001	CMA	/SHOULD NOT BE EXECUTED
00704	740200	SZA	/AC NOT 0 DID NOT BREAK
00705	740040	TMER15 HLT	

/

/

/STATUS A BIT 9 = 0 AND DTF = 1  
 /INT SHOULD NOT OCCUR  
 /

00706	202330	LAC (500	/CLEAR SA 9
00707	707544	DTXA	/AND NOT CLEAR DTF
00710	202331	LAC (JMP TMER16-1	
00711	040001	DAC 1	
00712	750000	CLA	
00713	700042	ION	/ENABLE PIE
00714	740000	NOP	/SHOULD NOT BREAK
00715	700002	IOP	/SHOULD BE EXECUTED
00716	740001	CMA	/SHOULD BE EXECUTED
00717	740100	SMA	/AC SHOULD BE NEGATIVE
00720	740040	TMER16 HLT	/AC = 0 IS ILLEGAL INT

.EJECT

```

/DTSF SHOULD SKIP DTF = 1
00721 750000 CLA /CLEAR FOR INDICATOR
00722 707601 DTDF /SHOULD SKP CMA
00723 740001 CMA /SHOULD NOT BE EXECUTED
00724 740200 SZA /AC NOT 0 IS NOT SKIP
00725 740040 TMR17 HLT /DTSF FAILED TO SKIP
/
/
/DTDF SHOULD NOT SKIP DF = 0
00726 750000 CLA /CLEAR FOR INDICATOR
00727 707561 DTDF /SHOULD NOT SKIP DF = 0
00730 740001 CMA /SHOULD NOT BE EXECUTED
00731 740100 SMA /AC = 0 IS SKPD
00732 740040 TMR18 HLT /DTDF SKIPPED IN ERROR
/
/
/CAF SHOULD CLEAR DECTAPE FLAG
/
00733 703302 CAF /CLEAR ALL
00734 707572 DTRB+10 /READ STATUS
00735 740200 SZA /AC SHOULD = 0
00736 740040 TMR19 HLT /CAF DID NOT CLEAR
/
/
/DO A READ ALL FORWARD NORMAL MODE
/SHOULD DATA BREAK WITHIN 250 MILLISECONDS
/
00737 202332 LAC (023000 /READ ALL FORWARD
00740 707545 DTLA /LOAD AND GO
00741 777776 LAW -2
00742 040030 DAC WCLOC /2 WORDS
00743 202323 LAC (BUFFER=1
00744 040031 DAC CALOC /INTO BUFFER
00745 202333 LAC (-122000
00746 040010 DAC 10
00747 200030 LAC WCLOC
00750 740001 CMA
00751 741200 SNA /MC +1 YET
00752 600756 JMP .+4 /YES
00753 440010 ISZ 10 /WAITED 250 MILLISECONDS
00754 600747 JMP .-5 /NO
00755 740040 TMR20 HLT /NOTHING IN 250 MSEC
00756 707572 DTRB+10
00757 242326 XOR (100 /DTF SHOULD = 1
00760 740200 SZA /ALL OTHERS 0
00761 740040 TMR21 HLT /DTF DID NOT SET
.EJECT

```

```

00762 200031      /TIME CA SHOULD BE INCR +1
00763 542323      LAC CALOC
00764 740040      SAD (BUFFER=1
                    TMR22 HLT
                    /AND ONLY +1
00765 542317      SAD (BUFFER
00766 741000      SKP
00767 740040      TMR23 HLT
                    /
                    /
                    /NOW A TIMING ERROR SHOULD OCCUR
                    /WITHIN 250 MICROSECONDS
                    /
00770 777700      LAR -100
00771 040010      DAC 10
00772 707572      DTRB+10      /READ STATUS
00773 741100      SPA          /EF YET
00774 600777      JMP .+3      /YES
00775 440010      ISZ 10      /WAITED 250 USEC
00776 600772      JMP .-4      /NO
                    /
                    /
                    /TIMING EF AND DTF SHOULD = 1
00777 242334      XOR (410100      /SHOULD MAKE AC = 0
01000 740200      SZA          /RIGHT FLAGS
01001 740040      TMR24 HLT          /1 BITS ARE ERROR MIGHT ACT = 0
                    /
                    /TIMING ERROR SHOULD CLEAR MOTION
01002 707552      DTRA+10      /READ STAT A
01003 502306      AND (GOBIT      /MASK MOT BIT
01004 740200      SZA          /TIM CLEAR IT
01005 740040      TMR25 HLT          /TIM DID NOT GENERATE ERROR STOP
                    /
                    /
                    /A DATA BREAK SHOULD NOT HAVE
                    /BEEN MADE WCO AT LAST DBRK
                    /
01006 200030      LAC WCLOC          /SHOULD NOT HAVE CHANGED
01007 740200      SZA          /DID IT +1
01010 740040      TMR26 HLT          /DBRKD THRU WC = 0
                    /
                    /
                    /DTXA WITH AC BIT 10 = 1 AND 11 = 0
                    /SHOULD CLEAR DTF AND NOT CLEAR TIMING
                    /
01011 202303      LAC (200
01012 707544      DTXA          /SHOULD CLEAR DTF 11=0
01013 707572      DTRB+10      /READ STAT B
01014 242335      XOR (410000      /MAKE EF AND TIM = 0
01015 740200      SZA          /DTF CLEAR AND NOT TIM
01016 740040      TMR27 HLT          /DISPLAY FLD BITS
                    .EJECT

```

/CAF SHOULD CLEAR TIMING ERROR

/

01017	703302	CAF	/CLEAR ALL
01020	707572	DTRB+10	/READ TIM ERROR
01021	740200	SZA	/STAT B SHOULD = 0
01022	740040	TMER28 HLT	/DISPLAY FLD STAT B

/

/

/NOW DO A READ ALL FORWARD C MODE  
 /SET CA FOR 2 WORDS  
 /1ST DATA BREAK WITHIN 250 MILLISEC  
 /2ND DATA BREAK WITHIN 250 MICROSEC  
 /2ND BRK SETS DTF  
 /A THIRD BREAK SHOULD NOT BE MADE

/

01023	202336	LAC (33000	/READ ALL C MODE
01024	707545	DTLA	/LEAD AND START
01025	777776	LAW =2	/2 WORD
01026	040030	DAC WCLOC	
01027	202323	LAC (BUFFER-1	/INTO BUFFER
01030	040031	DAC CALOC	
01031	202333	LAC (-122000	/COUNT 250 MILLISEC
01032	040010	DAC 10	
01033	200030	LAC WCLOC	
01034	740001	CMA	/WC +1 YET
01035	741200	SNA	
01036	601042	JMP .+4	/YES 1 BREAK MADE
01037	440010	ISZ 10	/WAITED 250 MILLISEC
01040	601033	JMP .-5	/NO
01041	740040	TMER29 HLT	/NO DATA BREAK 250 MILLISEC

/

/

/THE DTF SHOULD NOT HAVE SET

/

01042	707572	DTRB+10	/READ STAT B
01043	740200	SZA	/DTF SHOULD = 0 NOT WCO
01044	740040	TMER30 HLT	/DISPLAY ERROR STAT B

/

/

/A SECOND BREAK SHOULD BE MADE  
 /WITHIN 250 MICROSEC OF LAST  
 /SINCE WCO DTF SHOULD = 1

01045	777700	LAW =100	
01046	040010	DAC 10	
01047	200030	LAC WCLOC	
01050	741200	SNA	
01051	601055	JMP .+4	
01052	440010	ISZ 10	
01053	601047	JMP .-4	
01054	740040	TMER31 HLT	

TMER31

.EJECT

/2ND BREAK SHOULD SET DTF

/

01055	707572	DTRB+10	/READ STAT B
01056	242326	XOR (100	/ONLY DTF SHOULD = 1
01057	740200	SZA	/WAS STAT COR
01060	740040	TMR32 HLT	/DISPLAY FAILED BITS

/

/

/A THIRD BREAK SHOULD NOT BE MADE

/WC = 0

/

01061	777700	LAW =100	
01062	040010	DAC 10	
01063	200030	LAC WCLOC	/GET WC
01064	740200	SZA	/DID IT +1
01065	740040	TMR33 HLT	/DATA BREAK THRU WC = 0
01066	440010	ISZ 10	/WAITED 250 MIC SEC
01067	601063	JMP .-4	/NO

/

/

/I/O POWER CLEAR (CAF) SHOULD CLEAR TAPE MOTION

/

01070	703302	CAF	/CLEAR ALL
01071	707552	DTRA+10	/READ STAT A
01072	502306	AND (GOBIT	/MASK NOT BIT
01073	740200	SZA	/DID IT CLEAR
01074	740040	TMR34 HLT	/NO POWER CLEAR FAILED

/

/

/NOW DO SEARCH FORWARD NORMAL MODE

/DATA BREAK SHOULD BE MADE WITHIN 1 SECOND

/

01075	202337	LAC (21000	/SEARCH N MODE
01076	707545	DTLA	/LOAD AND GO
01077	140030	DZM WCLOC	
01100	202323	LAC (BUFFER-1	
01101	040031	DAC CALOC	
01102	777777	LAW =1	
01103	046377	DAC BUFFER-1	
01104	202340	LAC (172366	
01105	040010	DAC 10	
		.EJECT	

```

/monitor wc for 1 second
01106 200030 LAC WCLUC
01107 740200 SZA /BREAK MADE YET
01110 601114 JMP .+4 /YFS
01111 440010 ISZ 10 /WAITED 1 SEC
01112 601100 JMP .-4 /NO
01113 740040 TMR35 HLT /NO DATA BREAK FOR 1 SEC
/
01114 707572 DTRB+10 /READ STAT 0
01115 242326 XOR (100) /ONLY DTF SHOULD BE SET
01116 740200 SZA
01117 740040 TMR36 HLT /DISPLAY BITS IN ERROR
/
/CA SHOULD NOT INCR IN SEARCH
01120 200031 LAC CALOC
01121 542323 SAD (BUFFER-1) /DID CA +1
01122 741000 SKP /NO
01123 740040 TMR37 HLT /+1 TO CA INH FAILED
/
/WAS TRANSFER DIRECTION INPUT
01124 206377 LAC BUFFER-1 /GET INPUT WORD
01125 740001 CMA /SHOULD NOT = 0
01126 741200 SNA /INPUT MADE
01127 740040 TMR38 HLT /APPARENTLY OUTPUT
.EJECT

```

```

/TAPE SHOULD STILL BE MOVING
/CHANGE TO SEARCH C MODE
01130 202341 LAC (1000Y
01131 707544 DTXA /SET C MODE BIT
01132 777776 LAW -2 /SET UP 2 WOFUS
01133 740030 DAC WCLOC
/
/FIRST DATA BREAK IS APPROX 50 MILLISEC
01134 202342 LAC (-23420
01135 740010 DAC 10
01136 200030 LAC WCLOC /GET WC
01137 740001 CMA /MAKE +
01140 741200 SNA /BREAK YET
01141 601145 JMP .+4 /YES
01142 440010 ISZ 10 /WAITED 70 MILLISEC
01143 601136 JMP .-5 /NO
01144 740040 TMR39 HLT /NO DATA BREAK 70 MILLISEC
/
/NO WCO SHOULD NOT SET DTF AT BREAK1
01145 707572 DTRB+10 /READ STAT B
01146 740200 SZA /NO FLAGS SHOULD = 1
01147 740040 TMR40 HLT /DISPLAY ERROR STAT B
/
/2ND DATA BREAK APPROX. 50 MILLISEC
01150 202342 LAC (-23420
01151 740010 DAC 10
01152 200030 LAC WCLOC /SECOND BREAK
01153 741200 SNA /MADE YET
01154 601160 JMP .+4 /YES
01155 440010 ISZ 10 /WAITED 70 MILLISEC
01156 601152 JMP .-4 /NO
01157 740040 TMR41 HLT /NO DATA BREAK 70 MILLISEC
/
/WCO SHOULD SET DTF IN CMODE
01160 707572 DTRB+10 /READ STAT B
01161 242320 XOR (100 /WCO SHOULD SET DTF
01162 740200 SZA /DTF = 1 ALL OTHERS = 0
01163 740040 TMR42 HLT /DISPLAY BITS IN ERROR
/
/NOW CHANGE TO READ DATA
/NORMAL MODE SET WC TO -257
/
01164 202343 LAC (13000
01165 707544 DTXA /CHANGE TO RDATA NORMAL MODE
01166 777377 BLENTH=1 /-257
01167 040030 DAC WCLOC /FOR WC
.EJECT

```

/FIRST DATA BREAK SHOULD NOT OCCUR  
/FOR AT LEAST 420 MICROSEC

```

01170 777575          LAW -203
01171 040010          DAC 10
01172 440010          ISZ 10          /TIME OUT 420
01173 601172          JMP .-1          /MICRO SEC

/
01174 200030          LAC WCLOC          /BREAK MADE YET
01175 542344          SAD (BLENGTH-1
01176 741000          SKP          /NO BREAK YET OK
01177 740040          TMR43 HLT          /NO DATA BREAK IN STAT BLOCK OR REVCK

```

/FIRST DATA BREAK SHOULD OCCUR  
/WITHIN NEXT 300 MICROSEC

```

01200 777700          LAW -100
01201 040010          DAC 10
01202 200030          LAC WCLOC          /GET WC
01203 542345          SAD (BLENGTH          /BREAK YET
01204 601210          JMP .+4          /YES
01205 440010          ISZ 10          /TIMED OUT 300 MICROSEC
01206 601202          JMP .-4          /NO
01207 740040          TMR44 HLT          /NO RDATA BREAK

```

/1 TO DF SHOULD NOT SET DTF

```

01210 707572          DTRB+10          /READ STAT 0
01211 740200          SZA          /NO FLAGS SHOULD = 1
01212 740040          TMR45 HLT

```

/EACH SUCCEEDING DATA BREAK  
/SHOULD OCCUR WITHIN 300 MICROSEC  
/OF LAST DATA BREAK

```

01213 202346          SVWCP1 LAC (1
01214 340030          TAD WCLOC          /CURRENT WC +1
01215 040011          DAC 11          /SAVE FOR COMPARE
01216 777720          LAW -60
01217 040010          DAC 10
01220 200030          LAC WCLOC          /BREAK MADE YET
01221 540011          SAD 11
01222 601226          JMP .+4          /YES DATA BREAK
01223 440010          ISZ 10          /WAITED 300 MICROSEC
01224 601220          JMP .-4          /NO
01225 740040          TMR46 HLT          /NO DBRK 300 MICROSEC

```

```

01226 740001          CMA
01227 740200          SZA          /250 WORDS YET
01230 601213          JMP SVWCP1          /NO
                          .EJECT

```

```

/ THE DTF SHOULD SET WITHIN 300 MICROSEC
/ STAT CHECK GOING TO 0 AND KDATA NORMAL
01231 777634          LAW -144
01232 040010          DAC 10
01233 440010          ISZ 10          / TIME OUT 300 MICROSEC
01234 601233          JMP .-1
01235 707572          DIRB+10
01236 502347          AND (357777      / MASK PAR
01237 242326          XOR (100          / ONLY DTF SHOULD = 1
01240 740200          SZA
01241 740040          TMR47 HLT          / DISPLAY FAILED BITS
/
/ WAIT ANOTHER 300 MICROSEC
/ WCLOC SHOULD NOT CHANGE
01242 777634          LAW -144
01243 040010          DAC 10
01244 440010          ISZ 10          / TIME OUT 300
01245 601244          JMP .-1          / MICRO SEC
01246 200030          LAC WCLOC          / WC GET +1
01247 740001          CMA
01250 740200          SZA
01251 740040          TMR48 HLT          / DATA BREAK DURING IDLE
/
/
/ NOW CHANGE TO READ DATA
/ C MODE SET WC FOR 2 BLOCKS
/ DTF SHOULD NOT SET AT END OF
/ FIRST BLOCK NO WCO, DTF SHOULD SET
/ AT END OF SECOND BLOCK WC WILL = 0
/
01252 202341          LAC (10000
01253 707544          DTXA          / SET CMODE
01254 777000          LAW -1000          / =512
01255 040030          DAC WCLOC          / FOR 2 BLOCKS
01256 202323          LAC (BUFFER-1
01257 040031          DAC CALOC
/
/ SHOULD MAKE 256 DATA BREAKS
/ WITHIN NEXT 56 MILLISEC
/
01260 202350          LAC (756700
01261 040010          DAC 10
01262 200031          LAC CALOC          / GET CA
01263 542351          SAD (BUFFER2-1      / END OF BUFFER
01264 601270          JMP .+4          / YES MADE 256 BRKS
01265 440010          ISZ 10          / WAITED 1 BLOCK TIME
01266 601262          JMP .-4          / NO
01267 740040          TMR49 HLT          / DISPLAY CA
.EJECT

```

/TIME OUT 600 MICROSEC  
/NO MORE DATA BRKS SHOULD BE MADE  
/

01270	777470	LAW -310	
01271	040010	DAC 10	
01272	440010	ISZ 10	/TIME OUT
01273	601272	JMP .-1	/000 MICROSEC
01274	200030	LAC WCLOC	
01275	542345	SAD (BLENGTH	/MAKE ANY MORE BREAKS
01276	741000	SKP	/NO
01277	740040	TIMER50 HLT	/EXTRA DATA BREAKS

/THE DTF SHOULD NOT BE SET  
/RDATA CMODE AND WC = 1  
/

01300	707572	DTRB+10	/READ STAT B
01301	502347	AND (357777	/MASK OFF PARITY
01302	740200	SZA	/ANY OTHERS SET
01303	740040	TIMER51 HLT	/DISPLAY FLAGS IN ERROR

/NOW SHOULD MAKE ANOTHER 250  
/DATA BREAKS WITHIN NEXT 56 MILLISEC  
/

01304	202352	LAC (-34000	
01305	040010	DAC 10	
01306	200030	LAC WCLOC	
01307	741200	SNA	/WC GO TO 0 YET
01310	001314	JMP .+4	/YES
01311	440010	ISZ 10	/5 BLOCK TIME ELAPSED
01312	601306	JMP .-4	/NO
01313	740040	TIMER52 HLT	

/WCO SHOULD NOT SET DTF BY ITSELF  
/

01314	707572	DTRB+10	/READ STAT B
01315	502347	AND (357777	/MASK PARITY
01316	740200	SZA	/SHOULD HAVE NO OTHER FLAGS
01317	740040	TIMER53 HLT	

/TIME OUT 300 MICROSEC THEN  
/THE DTF SHOULD BE SET ST CHECK TO 0  
/RDATA C MODE AND WC = 0

01320	777634	LAW -144	
01321	040010	DAC 10	
01322	440010	ISZ 10	/TIME OUT 300
01323	601322	JMP .-1	/MICRO SEC
01324	707572	DTRB+10	/READ STAT B
01325	502347	AND (357777	/MASK PARITY
01326	242326	XOR (100	/ONLY DTF SHOULD = 1
01327	740200	SZA	/DISPLAY FAILED BITS
01330	740040	TIMER54 HLT	

.EJECT

/NOW CHANGE BACK TO SEARCH NORMAL MODE  
/AND TEST WRITE DATA OVER NEXT BLOCK  
/

01331	202343	LAC (13000	
01332	707544	DTXA	/CHANGE TO SEARCH N MODE
01333	140030	D2M WCLOC	
01334	202323	LAC (0LKFND	
01335	040031	DAC CALOC	
01336	707572	DTRB+10	/FIND BLOCK YET
01337	741200	SNA	
01340	601336	JMP .-2	/WAIT FOR ST BLK
01341	202353	LAC (5000	
01342	707544	DTXA	/CHANGE TO WRITE DATA
01343	777377	BLENTN=1	
01344	040030	DAC WCLOC	/-257

/SHOULD NOT MAKE ANY BREAKS FOR AT LEAST 200 USEC

01345	777702	LAW -76	
01346	040010	DAC 10	
01347	440010	ISZ 10	/TIME OUT
01350	601347	JMP .-1	/200 MICROSEC
01351	200030	LAC WCLOC	
01352	542344	SAD (BLENTN=1	/MAKE ANY BREAKS
01353	741000	SKP	/NO
01354	740040	TIMER55 HLT	/PREMATURE WDATA BREAK

/THE FIRST DATA BREAK SHOULD  
/OCCUR DURING STATE REVERSE CHECK  
/WITHIN NEXT 300 MICROSEC

01355	777634	LAW -144	
01356	040010	DAC 10	
01357	200031	LAC CALOC	
01360	542317	SAD (BUFFER	/MAKE 1ST BREAK
01361	601365	JMP .+4	/YES
01362	440010	ISZ 10	/WAITED 300 MICSEC
01363	601357	JMP .-4	/NO
01364	740040	TIMER56 HLT	/NO BREAK STATE REVERSE CHECK

/NEXT 255 DATA BREAKS  
/SHOULD OCCUR WITHIN 300  
/MICROSEC OF EACH OTHER

01365	202346	WDWCP1 LAC (1	
01366	340030	TAD WCLOC	/CURRENT WC +1
01367	040011	DAC 11	/SAVE IT
01370	777720	LAW -60	
01371	040010	DAC 10	
01372	200030	LAC WCLOC	
01373	540011	SAD 11	/MAKE NEXT BREAK
01374	601400	JMP .+4	/YES
01375	440010	ISZ 10	/WAITED 300 MICSEC
01376	601372	JMP .-4	/NO
01377	740040	TIMER57 HLT	/NO DBRK 300 MICSEC

.EJECT

```

/SEE IF 256 DATA BREAKS YET
01400 740001 CMA
01401 740200 SZA /MADE 256 BREAKS
01402 601365 JMP WDXCP1 /NO
/
/
/TIME OUT 200 MICRO SEC
/WRITE DATA SHOULD NOT DATA BREAK DURING STATE FINAL
/
01403 777700 LAW -100
01404 040010 DAC 10
01405 440010 ISZ 10 /TIME OUT
01406 601405 JMP .-1 /200 MIC SEC
01407 200030 LAC WCLOC /GET WC
01410 740001 CMA
01411 740200 SZA /EXTRA BREAK
01412 740040 TMR58 HLT /WDATA BREAK STATE FINAL
/
/
/TIME OUT ANOTHER 200 USEC
/DTF SHOULD GO TO 1 AND NO DATA BREAK
/DURING STATE CHECK
/
01413 777800 LAW -200 /* A LITTLE XTRA
01414 040010 DAC 10
01415 440010 ISZ 10 /TIME OUT ABOUT
01416 601415 JMP .-1 /200 MICROSEC
01417 707572 DTRB+10 /READ STAT 8
01420 242326 XOR (100 /DTF SHOULD = 1
01421 740200 SZA
01422 740040 TMR59 HLT /DISPLAY ERROR BITS
/
/WRITE DATA DATA BREAK SHOULD
/BE INHIBITED DURING STATE CHECK
01423 200030 LAC WCLOC /GET WC
01424 740001 CMA
01425 740200 SZA /EXTRA BREAK
01426 740040 TMR60 HLT /WDATA BREAK STATE CHECK
/
/TIME OUT 600 MICROSEC
/WREN SHOULD GO TO 0 DURING STATE IDLE
/AND NO MORE DATA BREAKS
/
01427 777470 LAW -310
01430 040010 DAC 10
01431 440010 ISZ 10 /TIME OUT 600 MICSEC
01432 601431 JMP .-1
01433 200030 LAC WCLOC /GET WC
01434 740001 CMA
01435 740200 SZA /DID IT +1
01436 740040 TMR61 HLT /WREN PROB DID NOT GO TO 0
.EJECT

```

```

/WRITE DATA WILL MAKE 1 DATA BREAK AT STATE REVERSE CHECK
01437 777776      LAW -2
01440 040030      DAC WCLUC
01441 200030      LAC WCLUC
01442 740001      CMA          /WAIT FOR DBRK
01443 740200      SZA          /AT NEXT REVERSE CHECK
01444 601441      JMP .-3
01445 777470      LAW -310
01446 040010      DAC 10
01447 440010      ISZ 10
01450 601447      JMP .-1

/
/
/A DTXA AT THIS POINT WITH
/WRITE DATA WILL CAUSE TIMING ERROR
/STBLKMRK AND ST IDLE = 0
01451 707554      DTXA+10      /XOR 0'S
01452 442100      ISZ DLY      /NDATA ALREADY = 1
01453 601452      JMP .-1      /WAIT
01454 707572      DTRB+10     /TIMING SHOULD BE = 1
01455 242335      XOR (410000) /SHOULD CLEAR TIM AND EF
01456 740200      SZA
01457 740040      TMR63 HLT      /DISPLAY FAILED BIT

/
/NEXT DTXA SHOULD CLEAR TIMING
/LAST ERROR MAKES US = 0 WHICH SHOULD
/CLEAR STATE REG TO STATE IDLE
01460 707554      DTXA+10      /CLEAR AGAIN
01461 601462      JMP .+1      /WAIT 5
01462 601463      JMP .+1
01463 601464      JMP .+1
01464 601465      JMP .+1
01465 707572      DTRB+10     /READ SHOULD ALL = 0
01466 740200      SZA
01467 740040      TMR64 HLT      /DISPLAY ERROR STATUS
.EJECT

```

/NOW START UP IN SEARCH AGAIN

/AND TEST WRITE DATA CMODE

/

01470	202337	LAC (21000	/SEARCH N MODE
01471	707545	DTLA	/LOAD AND GO
01472	140030	DZM WCLOC	
01473	202323	LAC (BLKFND	
01474	040031	DAC CALOC	
01475	200030	LAC WCLOC	/WAIT FOR BLOCK
01476	741200	SNA	
01477	601475	JMP .-2	
01500	202354	LAC (15000	
01501	707544	DTXA	/MAKE W DATA C MODE
01502	777001	LAW -777	/-511
01503	040030	DAC WCLOC	
01504	202323	LAC (BUFFER-1	/1 WORD SHORT 2 BLOCKS
01505	040031	DAC CALOC	
01506	202352	LAC (-34000	
01507	040010	DAC 10	
01510	200031	LAC CALOC	
01511	542351	SAD (BUFFER2-1	/WAIT FIRST
01512	601516	JMP .+4	/256 DATA BRAKS
01513	440010	ISZ 10	
01514	601510	JMP .-4	
01515	740040	TMER65 HLT	/
/			
/NOW TIME OUT 600 MICROSEC DTF SHOULD = 0			
01516	777470	LAW -310	
01517	040010	DAC 10	
01520	440010	ISZ 10	/WAIT 600 MICSEC
01521	601520	JMP .-1	
01522	707572	DTRB+10	/READ STAT B
01523	740200	SZA	/DTF SHOULD = 0
01524	740040	TMER66 HLT	/CMODE AND WC = 1 INH DTF
/			
/ANY BREAKS MADE ST FINAL AND STATE CHECK			
01525	200031	LAC CALOC	/GET CA
01526	542351	SAD (BUFFER2-1	/SHOULD NOT HAVE INCREMENTED
01527	741000	SKP	/OK
01530	740040	TMER67 HLT	/EXTRA DB WRITE DATA CMODE
/			
/SHOULD MADE NEXT 255 BREAKS 56 MILLISEC			
01531	202352	LAC (-34000	
01532	040010	DAC 10	
01533	200030	LAC WCLOC	/WAIT FOR WC
01534	741200	SNA	/TO GO TO 0
01535	601541	JMP .+4	/AC = 0
01536	440010	ISZ 10	/WAITED 56 MILLISEC
01537	601533	JMP .-4	/NO
01540	740040	TMER68 HLT	/DID NOT MAKE ENOUGH BREAKS
.EJECT			

```

01541 777320 /NOW WAIT 900 MICROSEC AND DTF SHOULD = 1
01542 740010 LAW -400
01543 440010 DAC 10
01544 601543 ISZ 10 /WAIT 900 MICSEC
01545 707572 JMP .-1
01546 242326 DTRB+10 /READ STAT 6
01547 740200 XOR (100)
01550 740040 SZA /DTF SHOULD = 1
TMER69 HLT /CMODE WC=2 AND STCHK TO W FAILED
/WITH WC = 0 SHOULD NOT MADE ANY MORE BREAKS
01551 200030 LAC WCLOC
01552 740200 SZA
01553 740040 TMER70 HLT /WC = 0 DID NOT INH OF
01554 703302 CAF
01555 102077 JMS TYPTX
01556 777777 777777
01557 455644 455644
01560 770000 770000
01561 740040 TMER71 HLT
01562 600523 JMP TMOTST
/LOAD ADDRESSES 30 TO 77 WITH LAW .15
/TEST FOR NO API BREAK AT START
01563 202355 APITST LAC (27) /FOR START STORE IN 30
01564 040010 DAC 10
01565 760030 LAW 30 /FIRST LAW
01566 060010 DAC+ 10
01567 542356 SAD (LAW 77) /DONE TO LAST CHANNEL
01570 601573 JMP .+3 /YES
01571 342346 TAD (1) /LAW +1
01572 601560 JMP .-4 /STORE NEXT
/NOW CLEAR ALL FLAGS AND TURN API ON
/NONE OF THE CHANNELS SHOULD BREAK
01573 703302 CAF /CLEAR ALL FLAGS
01574 202357 LAC (400000)
01575 705504 ISA /TURN ON API ENABLE
01576 402074 XCT LAC10 /STALL 4 CYCLES
01577 740200 SZA /AC SHOULD STILL = 0
01600 740040 HLT /DISPLAY FAILED CHAN IN AC
/NOW SET SELECT ERROR IN THE TC02 API IS OFF CAF
/TC02 SHOULD NOT API BREAK TO ADDRESS 44
01601 703302 CAF /SHOULD TURN API OFF
01602 202360 LAC (7400) /ALSO ENABLE I
01603 707545 DTLA /WILL SET SELECT ERROR
01604 750000 CLA
01605 601606 JMP .+1
01606 601607 JMP .+1 /API SHOULD NOT
01607 601610 JMP .+1 /BREAK IT IS TURNED
01610 601611 JMP .+1 /OFF
01611 601612 JMP .+1
01612 740200 SZA /API BREAK IN ERROR
01613 740040 HLT /DISPLAY FAILED CHANNEL
.EJECT

```

```

/NOV TC02 SHOULD BREAK TO ADDRESS 44
/WHEN API IS TURNED ON
01614 202357 LAC (400000) /TC02 SE = 1
01615 705504 ISA /SET API ENABLE
01616 402074 XCT LACIO /STALL 4 CYCLES API BREAK
01617 542361 SAD (LAW 44) /TC ADDRESS 44
01620 741000 SKP /YES OK IF AC = 0 NO BREAK
01621 740040 HLT /API NOT = 0 WRONG ADDRESS

/DBK INSTRUCTION SHOULD ALLOW 2ND BREAK
/SE STILL = 1 AND API STILL ENABLED
01622 707554 DTXA+10
01623 703304 DBK /CLEAR PRIORITY ACTIVE
01624 401610 XCT .-6 /SHOULD ALLOW 2ND BREAK AFTER 5 CYCLES
01625 542361 SAD (LAW 44) /RIGHT ADDRESS
01626 741000 SKP /YES, BROKE TO 44
01627 740040 HLT /AC = 0 NO BREAK OTHER WRONG

/NOV TURN API OFF
/THEN DBK SHOULD NOT ALLOW A BREAK
01630 707554 DTXA+10
01631 705514 ISA+10 /CLEAR API ENABLE
01632 703304 DBK /CLEAR DECTAPE PR ACT
01633 402074 XCT LACIO
01634 740200 SZA /AC SHOULD STILL = 0 NO BREAK
01635 740040 HLT /API DID NOT TURN OFF

/SET A HIGHER PRIORITY ACTIVE
/AND ENABLE API SHOULD NOT BREAK
01636 707554 DTXA+10
01637 202362 LAC (400200) /SET PR0 ACTIVE API ON
01640 705504 ISA /DO IT
01641 402074 XCT LACIO
01642 740200 SZA /SHOULD NOT ALLOW BREAK
01643 740040 HLT /0 ACTIVE YET API BROKE

/NOV DBK SHOULD ALLOW TC02 BREAK
/CLEAR PR0 ACTIVE THEN BREAKS
01644 703304 DBK /CLEAR PR0 ACTIVE
01645 401641 XCT .-4 /STALL 5 CYCLES SHOULD TEN API
01646 542361 SAD (LAW 44) /IF AC = 0 NO BREAK
01647 741000 SKP /AC = LAW 44 IS OK
01650 740040 HLT /AC NOT 0 AND NOT LAW 44 OTHER CHAN

/PROGRAM LEVEL REQUESTS SHOULD NOT INTERFERE WITH DECTAPE
01651 703302 CAF
01652 707554 DTXA+10 /WILL CAUSE SELECT ERROR
01653 202363 LAC (400400) /TC TURN API AND 4 PROG REQ ON
01654 705504 ISA /ENABLE
01655 402074 XCT LACIO /STALL 4 CYCLES AND API BREAK
01656 542361 SAD (LAW 44) /GET TO 44 ONLY
01657 741000 SKP /YES
01660 740040 HLT
.EJECT

```

/TURN API ON THEN OFF  
/SHOULD NOT ALLOW TC02 TO BREAK

```

01661 703302          CAP          /CLEAR API ENABLE
01662 707554          DTXA+10      /AND DECTAPE ACTIVE
01663 402074          XCT LACIO    /WAIT FOR SE
01664 202357          LAC (400000
01665 705504          ISA          /TURN API ON
01666 705514          ISA+10      /TURN IT OFF
01667 402074          XCT LACIO
01670 740200          SZA          /SHOULD NOT HAVE BROKEN
01671 740040          HLT          /BREAK AFTER API OFF

```

/ALSO DBK FOLLOWED BY API OFF  
/SHOULD NOT ALLOW A BREAK BUT LAST API OFF  
/SHOULD NOT LOOSE SE INTERRUPT

```

01672 202357          LAC (400000
01673 705504          ISA          /ENABLE API
01674 402074          XCT LACIO    /SHOULD NOP THEN BREAK
01675 542361          SAD (LAW 44
01676 741000          SKP          /API DID NOT LOOSE SE
01677 740040          HLT          /API ON API OFF CLRD SE
01700 707554          DTXA+10
01701 402074          XCT LACIO
01702 703304          DBK          /CLEAR DECTAPE ACTIVE
01703 705514          ISA+10      /TURN API OFF
01704 402074          XCT LACIO    /SHOULD NOT ALLOW BREAK
01705 740200          SZA
01706 740040          HLT          /BREAK AFTER API OFF

```

/DBK FOLLOWED BY API OFF SHOULD  
/NOT HAVE LOST THE SELECT ERROR INTERRUPT

```

01707 202357          LAC (400000
01710 705504          ISA
01711 402074          XCT LACIO
01712 542361          SAD (LAW 44
01713 741000          SKP          /LAST ISA 10 LOST AN ACTIVE INTERRUPT
01714 740040          HLT
                .EJECT

```

/API SHOULD OVERRIDE PROGRAM INTERRUPT

```

01715 1400000 DZM P
01716 202364 LAC (.+10
01717 040002 DAC 2
01720 703302 CAF
01721 700042 ION /ENABLE PI
01722 202357 LAC (400000 /ENABLE API
01723 705504 ISA /ENABLE API
01724 202360 LAC (7400 /CAUSE A SELECT ERROR
01725 707545 DTLA /CAUSE A SELECT ERROR
01726 402074 XLT LACI0
01727 601730 JMP .+1 /WAIT
01730 601731 JMP .+1 /FOR PI OR API
01731 601732 JMP .+1 /SHOULD ONLY GET
01732 601733 JMP .+1 /THE API BREAK
01733 601734 JMP .+1 /AND NOT THE PI
01734 601735 JMP .+1
01735 542361 SAD (LAW 44 /API OK
01736 741000 SKP /YES
01737 740040 HLT /API DID NOT BREAK
01740 200000 LAC 0 /SHOULD STILL BE 0
01741 740200 SZA /DID PI OCCUR
01742 740040 HLT /YES API DID NOT INH PI

```

```

/DBK SHOULD NOT ALLOW 2ND BREAK FROM SAME FLAG
/AND PIE SHOULD NOT OCCUR ADRS 0 STILL = 0

```

```

01743 202365 LAC (.+6
01744 040002 DAC 2
01745 703304 DBK /DBK
01746 402016 XCT SRCHL0 /STALL 5 CYCLES
01747 601750 JMP .+1 /SHOULD NOT PIE SHOULD APE
01750 601751 JMP .+1 /SELECT ERROR HAS ALREADY
01751 542361 SAD (LAW 44 /BEEN SERVICED SHD BE
01752 741000 SKP /SERVICED TWICE
01753 740040 HLT
01754 200000 LAC 0 /API BROKE
01755 740200 SZA
01756 740040 HLT /PROGRAM INTERRUPT WENT TO 0
01757 700002 IOF
01760 760044 LAW 44
01761 040044 DAC 44 /RESTORE 44 TO LAW
01762 202302 LAC (JMP+ 2) /RESTORE 1 FOR NEXT
01763 040001 DAC 1 /INTERRUPT

```

.EJECT

/STATUS A BIT 9 = 0 SE = 1  
/SHOULD NOT ALLOW API BREAK  
/

01764	703302	CAF	
01765	202357	LAC (400000)	
01766	705504	ISA	
01767	202310	LAC (7000)	/SE AGAIN ONLY NO-ENI
01770	707545	DTLA	/CAUSE SE
01771	402074	XCT LACI0	
01772	601773	JMP .+1	/WAIT SHOULD NOT API
01773	601774	JMP .+1	
01774	601775	JMP .+1	
01775	601776	JMP .+1	
01776	740200	SZA	/DID API BREAK
01777	740040	HLT	/API BROKE NOT DT ENI
02000	703302	CAF	
02001	102260	JMS REWDV	/PUT DRIVE IN EZ
02002	202366	LAC (SRCHF+ENABLI	
02003	340231	TAD UNFUNC	
02004	707545	DTLA	/SEARCH FORWARD GO
02005	202323	LAC (BLKFND	
02006	040031	DAC CALOC	/SET UP CA
02007	707572	DTRB+10	
02010	741200	SNA	/WAIT FOR DTF
02011	602007	JMP .-2	
02012	741100	SPA	/ERROR STATUS
02013	602002	JMP .-11	/DO IT AGAIN
02014	202357	LAC (400000)	
02015	705504	ISA	/ENABLE API
02016	402074	SRCHL0 XCT LACI0	
02017	542361	SAD (LAW 44	/DID BREAK OCCUR TO 44
02020	741000	SKP	/YES API ON DTF
02021	740040	HLT	
		.EJECT	

/NOW START SEARCH FORWARD WITH API ON  
 /WHEN SEARCH DATA BREAKS SHOULD API  
 /ALSO USE CLOCK INTERRUPT FOR TIMER  
 /

02022	703302	CAF	
02023	200231	LAC UNFUNC	
02024	342300	TAD (SRCHF+ENABLI	
02025	707545	DTLA	
02026	777324	LAW -454	/5 SECOND CLOCK
02027	040007	UAC 7	/JUST IN CASE HANGS
02030	140000	OZM 0	
02031	700044	CLON	
02032	202367	LAC (.+12	
02033	040002	DAC 2	
02034	202357	LAC (400000	
02035	705504	ISA	
02036	700042	IUN	
02037	750000	CLA	
02040	741200	SNA	
02041	602040	JMP .-1	
02042	700004	CLOF	
02043	700002	IOF	
02044	542361	SAD (LAW 44	/DID API FOR DECTAPE
02045	741000	SKP	/YES DT BREAK OK
02046	740040	HLT	
02047	200000	LAC 0	
02050	740200	SZA	/DID PI OCCUR
02051	740040	HLT	/PI IN ERROR
02052	206377	LAC BLKFND	/WAS SEARCH
02053	741200	SNA	/DATA BREAK MADE
02054	740040	HLT	/SEARCH DID NOT DB
02055	703302	CAF	
02056	102077	JMS TYPTX	
02057	777777	LAW -1	
02060	416051	416051	/API
02061	004556	4556	/ EN
02062	447700	447700	/D EOM
02063	750004	LAS	
02064	502315	AND (1000	
02065	741200	SNA	
02066	740040	HLT	
02067	601563	JMP APITST	
02070	402071	XCTXCT XCT .+1	/SO THAT
02071	402072	XCT .+1	/PROGRAM WILL STALL
02072	402016	XCT SRCHL0	/FOR 7 CYCLES
02073	777777	EXIFLG LAW -1	/MADE 0 BY OZM IN 1
		.EJECT	

02074	222075	LACI0	LAC* .+1	/LAC I NEXT ADDRESS
02075	002076		.+1	/WHICH POINTS AT NEXT
02076	000000		0	/WHICH CONTAINS ZERUS
		/TYPE TEXT ROUTINE		
		/MESSAGE PACKED 6 BIT CODES		
		/ASCII -240 STORED AFTER JMS		
		/777777 IS CAR RET LINE FEED		
		/77 CHAR IS EOM		
02077	602077	TYPTX	JMP .	
02100	222077		LAC+ TYPTX	
02101	442077		ISZ TYPTX	
02102	042162		DAC TYP5AV	
02103	740001		CMA	
02104	740200		SZA	
02105	602110		JMP .+3	
02106	102252		JMS TYCRLF	
02107	602100		JMP TYPTX+1	
02110	740001		CMA	
02111	742020		RTR	
02112	742020		RTR	
02113	742020		RTR	
02114	042163		DAC TYP5AV+1	
02115	742020		RTR	
02116	742020		RTR	
02117	742020		RTR	
02120	102136		JMS TYPCHR	
02121	202163		LAC TYP5AV+1	
02122	102136		JMS TYPCHR	
02123	202162		LAC TYP5AV	
02124	102136		JMS TYPCHR	
02125	602100		JMP TYPTX+1	
		/STOP ON ERROR IF DELETE SNS NOT SET		
02126	602126	ERRHLT	JMP .	
02127	750004		LAS	
02130	502370		AND (600000	
02131	740200		SZA	
02132	622126		JMP+ ERRHLT	
02133	202126		LAC ERRHLT	
02134	740040		HLT	
02135	622126		JMP+ ERRHLT	
02136	602136	TYPCHR	JMP .	
02137	502371		AND (77	
02140	542371		SAD (77	
02141	622077		JMP+ TYPTX	
02142	342372		TAD (240	
02143	102145		JMS TYPCHA	
02144	622136		JMP+ TYPCHR	
			.EJECT	

02145	602145	TYPCHA	JMP .
02146	042161		DAC SAVCHA
02147	750004		LAS
02150	741100		SPA
02151	622145		JMP* TYPCHA
02152	202161		LAC SAVCHA
02153	700400		TLS
02154	700401		TSF
02155	602154		JMP .-1
02156	700402		TCF
02157	622145		JMP* TYPCHA
02160	000000	OLY	0
02161	000000	SAVCHA	0
02162	000000	TYPSAV	0
02163	000000		0
02164	000000		0
02165	000000		0
02166	000000		0

/TYPE CONTENTS OF THE  
/AC IN OCTAL  
TYPCON

02167	602167	JMP .
02170	102220	JMS DECONT
02171	102241	JMS TYPOCT
02172	202166	LAC TYPSAV+4
02173	102241	JMS TYPOCT
02174	202165	LAC TYPSAV+3
02175	102241	JMS TYPOCT
02176	202164	LAC TYPSAV+2
02177	102241	JMS TYPOCT
02200	202163	LAC TYPSAV+1
02201	102241	JMS TYPOCT
02202	202162	LAC TYPSAV
02203	102241	JMS TYPOCT
02204	102246	JMS SPACE2
02205	622167	JMP* TYPCON

/TYPE OUT LOWEST 3 CHAR  
/IN OCTAL

02206	602206	TYPC03	JMP .
02207	102220		JMS DECONT
02210	202164		LAC TYPSAV+2
02211	102241		JMS TYPOCT
02212	202163		LAC TYPSAV+1
02213	102241		JMS TYPOCT
02214	202162		LAC TYPSAV
02215	102241		JMS TYPOCT
02216	102246		JMS SPACE2
02217	622206		JMP* TYPC03
			.EJECT

```

02220 602220 DECONT JMP .
02221 742102 DAC TYP5AV
02222 742020 RTR
02223 740020 RAR
02224 042103 DAC TYP5AV+1
02225 742020 RTR
02226 740020 RAR
02227 042104 DAC TYP5AV+2
02230 742020 RTR
02231 740020 RAR
02232 042105 DAC TYP5AV+3
02233 742020 RTR
02234 740020 RAR
02235 042106 DAC TYP5AV+4
02236 742020 RTR
02237 740020 RAR
02240 622220 JMP* DECONT
02241 602241 TYPOCT JMP .
02242 502373 AND (7
02243 342374 TAD (260
02244 102145 JMS TYPCHA
02245 622241 JMP* TYPOCT
02246 602246 SPACE2 JMP .
02247 102077 JMS TYPTX
02250 000077 77
02251 602246 JMP* SPACE2
02252 602252 TYCRLF JMP .
02253 202375 LAC (215
02254 102145 JMS TYPCHA
02255 202376 LAC (212
02256 102145 JMS TYPCHA
02257 622252 JMP* TYCRLF

/REWIND DRIVE TO REVERSE ENDZONE
REWDRV 0
LAC (GOBIT+DIRBIT
TAD UNFUNC
DTLA /MOVE TAPE BKWD
DTRB+10 /WAIT FOR FLAGS
SNA
JMP .-2
OTRB
SAU (500000 /CAN ONLY BE ENDZONE
SKP /OK
XX /ERROR
IDRS
AND (200
SZA /ERROR FLAG?
JMP* REWDRV /YES
JMP .-5 /NO
.END

02300 000003 *L
02301 620211 *L
02302 620002 *L

```

02303	000200	*L
02304	600262	*L
02305	000400	*L
02306	020000	*L
02307	757400	*L
02310	007000	*L
02311	440000	*L
02312	600447	*L
02313	027400	*L
02314	600511	*L
02315	001000	*L
02316	060000	*L
02317	006400	*L
02320	777746	*L
02321	500000	*L
02322	045400	*L
02323	006377	*L
02324	723000	*L
02325	500100	*L
02326	000100	*L
02327	600704	*L
02330	000500	*L
02331	600717	*L
02332	023000	*L
02333	656000	*L
02334	410100	*L
02335	410000	*L
02336	033000	*L
02337	021000	*L
02340	172366	*L
02341	010000	*L
02342	754360	*L
02343	013000	*L
02344	777377	*L
02345	777400	*L
02346	000001	*L
02347	357777	*L
02350	756700	*L
02351	006777	*L
02352	744000	*L
02353	005000	*L
02354	015000	*L
02355	000027	*L
02356	760077	*L
02357	400000	*L
02360	007400	*L
02361	760044	*L
02362	400200	*L
02363	400400	*L
02364	001726	*L
02365	001751	*L
02366	021400	*L
02367	002044	*L

02370	000000	*L
02371	000077	*L
02372	000240	*L
02373	000007	*L
02374	000260	*L
02375	000215	*L
02376	000212	*L

SIZE=02377

NO ERROR LINES