

IDENTIFICATION

Product Code: MAINDEC 9A-D2DB-D
Product Name: PDP-9 Punch Test
Date Created: November 1, 1967
Maintainer: Diagnostic Group
Author: J. W. Richardson



CONTENTS

<u>Section</u>		<u>Page</u>
1.	ABSTRACT.....	1
2.	REQUIREMENTS.....	1
2.1	Equipment.....	1
2.2	Storage.....	1
2.3	Preliminary Programs.....	1
3.	LOADING PROCEDURE.....	1
4.	STARTING PROCEDURE.....	1
4.1	Control Switch Settings.....	1
4.2	Starting Addresses.....	1
4.2.1	Logic Tests.....	1
4.2.2	Data Checks.....	1
4.2.3	Restarting Addresses.....	1
4.3	Program Action.....	2
5.	OPERATING PROCEDURES.....	2
5.1	Operational Switch Settings.....	2
5.2	Subroutine Abstracts.....	2
5.2.1	Control Logic Tests.....	2
5.2.2	Data Check Test 1 through 9.....	4
5.3	Operating Instructions.....	6
5.4	Data Check Tests.....	6
5.4.1	Error Printouts.....	7
5.4.2	Operating Instructions.....	7
6.	ERRORS.....	8
6.1	Error Halts and Description.....	8
6.2	Error Recovery.....	8
7.	RESTRICTIONS.....	8
7.1	Starting Restrictions.....	8
7.2	Operating Restrictions.....	9
8.	MISCELLANEOUS.....	9
8.1	Execution Time.....	9
8.2	Applications - Control Logic Tests.....	9

CONTENTS (continued)

<u>Section</u>		<u>Page</u>
8.2.1	'Scope Mode	9
8.2.2	Fast 'Scope Mode Loop	10
8.2.3	Looping on Control Logic Tests 2-4	10
8.2.4	Punch Speed Checks	10
8.4	Applications - Data Check Tests	11
8.4.1	Variable Punch Speed Control	11
9.	PROGRAM DESCRIPTION	11

TABLES

<u>Table</u>		
1	Error Halts and Identification	12
2	Restarting Address	14

1. ABSTRACT

The PDP-9 Punch Test is designed to test and verify the operational status of the punch control logic, and the mechanical functions of the punch. A series of six tests is performed on the punch control, followed by nine tests on the punch itself. Provision is made to continuously loop on three of the six punch control tests, and any one of the punch data tests.

2. REQUIREMENTS

2.1 Equipment

Standard PDP-9 computer

2.2 Storage

The program occupies 3403₈ words of core memory from location 0 to 3403.

2.3 Preliminary Programs

(None required)

3. LOADING PROCEDURE

The tape supplied is punched in HRI mode. Set the ADDRESS SWITCHES to 00000. Press I/O RESET, and then READ-IN.

4. STARTING PROCEDURE

4.1 Control Switch Settings

All ACS down.

4.2 Starting Addresses

4.2.1 Logic Tests - 21

4.2.2 Data Checks - 1000

4.2.3 Restarting Addresses - Restarting addresses for individual tests may be found in table 2, appearing at the end of this document.

4.3 Program Action

Unless operating in 'scope mode, the program will first execute tests 1 through 4 in sequence using punch IOT PSA, and then repeat tests 2 through 4 using punch IOT PSB. A halt occurs after testing with PSB with the PC = 324. The punch NO TAPE flag is then tested. If the test is successful, the program halts with C(PC) = 346. The 1-sec delay is then tested by starting test 6 at location 400. A halt occurs with C(PC) = 416, if the test is successful.

The data check tests are then executed by starting at location 1000. There are nine data checks in all, each punching a different pattern. As one test is completed, the next is automatically started. When data test 9 is reached, the program will run until stopped by the operator.

5. OPERATING PROCEDURES

5.1 Operational Switch Settings

All ACS down.

5.2 Subroutine Abstracts

5.2.1 Control Logic Tests

5.2.1.1 Control Logic Test 1 - Illegal Instruction - Starting at location 21, test 1 first turns off the real-time clock, and then issues an I/O power clear (CAF). An illegal IOT of 700110 is then given. A stall of 70 msec is then done, followed immediately by a test of the punch flag. The punch flag should not be set at this point. If no error halts occur, test 2 is automatically executed.

An error halt occurs with PC = 43 (tagged PE01) if the punch flag is set.

5.2.1.2 Control Logic Test 2 - Set Punch Flag with PSA and PSB - The program first deposits a PSA IOT in location 51. Location 51 will be used by tests 2 through 4 by the use of an XCT instruction. Location 51 is changed to contain PSB at the end of test 4 for the second pass through logic tests 2 through 4. Test 2 then executes location 51 (PSA or PSB) and waits approximately 4 seconds for the punch to get up to speed before testing the punch flag. Test 2 is executed twice before going on to test 3.

An error halt occurs with the PC = 70 (tagged PA02) if the flag was not set using PSA. If the flag was not set using PSB, a halt occurs with the PC = 71 (tagged PB02).

5.2.1.3 Control Logic Test 3 - Reset Punch Flag with PSA, PSB and PCF - Test 3 first sets the punch flag by executing the IOT in location 51 (PSA or PSB). After the flag has been set, location 51 is immediately executed again in an attempt to clear the flag. If successful, the same procedure is again performed, and the flag is cleared using the IOT PCF. If no errors, test 4 is executed.

Error halts occur if the flag cannot be cleared. If the PC = 114 (tagged PA03), PSA did not clear the flag. With the PC = 115 (tagged PB03), PSB did not clear the flag. A halt with the PC = 140 (tagged PE04) indicates PCF did not clear the flag.

5.2.1.4 Control Logic Test 4 - Punch Interrupt Test - Test 4 first issues an I/O power clear (CAF), and then enables program interrupt. A stall of approximately 210 msec is performed. No interrupt should occur during this period. If an interrupt was caused by the punch or some other device, a spurious interrupt causes a halt with the PC = 240 (tagged P05A). An interrupt from the punch causes a halt with the PC = 242 (tagged P05B). At either location, C(AC) = the I/O status.

If the above test is successful, a program interrupt using the punch is attempted. An I/O power clear is issued, the interrupt enabled, and location 51 (PSA or PSB) executed. If the test is successful, a check is made to determine if tests 2 through 4 have been using PSA or PSB. If using PSA, the IOT PSB is deposited into location 51, and test 2 is automatically executed. If PSB, test 5 is automatically executed.

If no interrupt occurs, an error halt occurs with the PC = 227 (tagged PA05) when using PSA. A halt with the PC = 230 (tagged PB05) occurs if using PSB. If an interrupt occurred, but not from the punch, a halt occurs with the PC = 254 (tagged P05C). The AC will contain the I/O status word.

5.2.1.5 Control Logic Test 5 - No Tape Test - Test 5 begins by issuing an I/O power clear (CAF), and then testing whether the NO TAPE indicator (bit 9 of I/O status word) is set even though there is tape in the punch. The punch POWER switch is assumed to be ON. If the test is successful, a program halt occurs with the PC = 324 (tagged PHLT10). ←

An error halt occurs if the NO TAPE indicator is set. The PC will equal 315 (tagged FLGNOT).

After the halt at PHLT10, the operator must remove the tape from the punch. Pressing CONTINUE will cause the program to test the no-tape condition. If the NO TAPE indicator is not set, an error halt occurs with the PC = 341 (tagged NOFLG). If the test is successful, a program halt occurs with the PC = 346 (tagged HLT10A).

The punch POWER switch is then tested. Tape is replaced in the punch, and the POWER switch placed to OFF. Test 5 is then re-executed by pressing CONTINUE. If successful, a halt occurs with the PC = 346 (HLT10A). If the NO TAPE indicator is not set, an error halt occurs with the PC = 341 (NOFLG).

The POWER switch is then turned to ON before starting test 6, if the test is successful.

5.2.1.6 Control Logic Test 6 - 1-Second Timing Check - With the punch at a dead halt, the punch flag should be set no less than 1-sec after selecting the punch. This condition is tested by selecting the punch with PSA, and then entering a 1-sec timing loop. The punch flag is monitored during this time, and an error halt occurs with the PC = 413 (tagged EA08), if the flag is set before the exit from the timing loop.

If the test is successful, a halt occurs with the PC = 416 (PRHLT6).

Test 6 is the last test in the series of logic tests. Starting the data check tests is then manually executed from location 1000.

5.2.2 Data Check Test 1 through 9 - The data check tests provide nine different patterns. Each pattern was chosen to check for a specific type of failure, i.e., skew, feed hold registration, ability to punch all channel combinations, etc.

The data punched is also read by the reader. The paper tape is fed directly from the punch to the reader. The reader is always approximately 192 frames (1 eight-bit character = 1 frame) behind the punch. In all tests, except test 1, 5 and 6, 15 blocks (1 block = 64 frames) of one pattern is punched and read before proceeding to the next test. One block is punched, and then the program reads one block of information, checking for errors. Test 1 tests the ability of the punch to punch all 0s. Only 5 blocks of 0s are punched. Tests 5 and 6 both use program interrupt in order to read at punch speed. The total number of frames punched and read is the same as with tests 2 through 4, and tests 7 and 8. Test 9 punches random block lengths with random stalls between blocks. Maximum block length is 64 frames. The data is read only after 64 frames have been punched.

The rate of punch speed may be controlled by ACS 4 through 17 during all tests except tests 5, 6 and 8. ACS 4 through 17 up will give the slowest rate; all ACS down, normal rate (50 characters/sec).

Provisions are provided to halt on error, resync reader and punch after an error, loop on any one of the nine tests, and 'scope mode for any test.

During all punching and reading operations, both the reader and punch NO TAPE indicators are monitored. If either indicator sets, a printout occurs indicating which device is out of tape. This feature may also be regarded as a test of the NO TAPE indicators, but it is included mainly as a guard against tape jams in the punch, resulting in torn tape.

5.2.2.1 Test 1 - Punch All 0s - Five blocks of all 0s are punched and read. Any channel punched will cause an error condition.

5.2.2.2 Test 2 - Skew Check - Fifteen blocks are punched using channels 1 and 8 only. The pattern consists of 1 block of channel 1 only, followed by 1 block of channel 8 only. The object is to attempt to make the tape skew in the punch tape guides. Error indications will appear as either channel 8 or channel 1 dropping completely as a result of the feed-holes not being aligned correctly (latitudinally).

5.2.2.3 Test 3 - All Channels - Fifteen blocks of all channels are punched. This pattern tests the punch's ability to punch all channels for a prolonged period.

5.2.2.4 Test 4 - Sliding One - Each channel is punched individually starting with channel 1, and punching each channel in sequence up to and including channel 8. Channels 7 through 1 are then punched in sequence.

The type of error expected with this test is two channels (adjacent) being punched. This test may also show errors from tape skew.

5.2.2.5 Tests 5 and 6 - Sliding Zero - Both tests use program interrupt in order to read at punch speed. This provides start/stop type of action with the reader, which will more readily show feed-hole errors. Elongated, or poorly aligned feed holes will probably cause the reader sprocket to skip, which will put the program out of sync and cause an error condition.

Test 5 punches all but one channel, dropping a different channel each frame - similar to test 4 but using 0s instead of 1s.

Test 6, in addition to providing a test of feed-hole alignment, tests the ability of the punch to drop one channel after punching a series of all 1s.

Seven frames of all 1s are punched, and then one channel is dropped. The type of error expected is when the dropped channel continues to punch, even though it was not selected.

The resulting pattern is similar to test 5, but with seven frames of all 1s between each dropped channel.

5.2.2.6 Test 7 - Punch Binary Count With PSA - Testing all combinations of channels using a binary count pattern is done with PSA. Fifteen blocks are punched, the reader checking the data after each block.

5.2.2.7 Test 8 - Binary Count With PSB - Similar to test 7 except channel 7 should never be punched. Fifteen blocks are punched, the reader checking the data after each block.

5.2.2.8 Test 9 - Random Punch and Stall - Random block lengths are punched, with random stall times between blocks. The reader will not check the data until after 64 frames have been punched.

The pattern generated is alternate frames of all 1s and all 0s.

This test runs continuously until stopped by the operator.

5.3 Operating Instructions

- a. Place the HRI tape in the reader.
- b. Place all ACS down, and set the ADDRESS SWITCHES to 00000.
- c. Press I/O RESET, and then READ-IN. The program will halt after loading with PC = 3367, and MB = 740040.
- d. Set the ADDRESS SWITCHES to 21.
- e. Press I/O RESET, and then START. If no errors, the program will halt with PC = 324.
- f. Remove the tape from the punch.
- g. Press CONTINUE. If no errors, the program will halt with PC = 346.
- h. Replace the tape in the punch.
- i. Turn punch POWER OFF.
- j. Press CONTINUE. If no error, the program will halt with PC = 346.
- k. Turn punch POWER ON.
- l. Set the ADDRESS SWITCHES to 400.
- m. Press I/O RESET, and then START. If no error, the program will halt with PC = 416.
- n. The data check tests are performed next.

5.4 Data Check Tests

ACS functions for each of the nine tests are:

<u>ACS</u>	<u>FUNCTION</u>
0 (1)	Halt after error printout
1 (1)	'Scope mode, inhibit reading
2 (1)	Loop on current test
3 (1)	Resync reader and punch after each error
4-17	May be used to vary rate of punching with all tests except 5, 6 and 8

ACS 0 up will cause a halt with the PC = 2752. Press CONTINUE to recover. The program will continue in sequence.

ACS 1 up will cause the current test to be repeated until stopped by pressing PROGRAM STOP. Reading is inhibited during 'scope mode.

ACS 2 up will cause the current test to be repeated until stopped with PROGRAM STOP. Placing ACS 2 down while running will let the program go on to the next test after completing the current one.

ACS 3 up will enable the program to resync the punch and reader after an error, or error halt.

Blank leader is punched, with a sync character at the end of the leader. The reader takes up the slack while the leader is being punched.

After the resyncing operation a halt occurs with PC = 3212. At this point pressing CONTINUE will restart the test which was being performed, or the operator may restart another test.

ACS 0, 2 and 3 may be used together. In this case ACS 0 has priority over 2 and 3, ACS 3 has priority over ACS 2.

5.4.1 Error Printouts - Any error encountered will cause an error printout in the format shown below:

N GOOD XXX BAD XXX

Where: N = Test number (1 through 9)
 GOOD = What the data read was expected to be
 BAD = The data as read

Also, no-tape conditions will give a printout thus:

R NO TAPE (R = Reader)

P NO TAPE (P = Punch)

The ACS have no control over these printouts. The program continues on in sequence after either printout.

5.4.2 Operating Instructions

- a. Set the ADDRESS SWITCHES to 1000.
- b. Press I/O RESET, and then START.
- c. Approximately 1-1/2 ft of leader will be punched. The leader is blank except for one frame of all 1s. This frame is referred to as the MARK character, and has nothing to do with the internal operation of the program.

After punching the leader, a halt occurs with PC = 1004 (PRHLT7).

By placing the MARK character directly over the reader drive sprocket, the correct amount of slack will be maintained through the sequence of nine tests.

Starting at address 1000 may be used in lieu of ACS 3, if desired, then manually restarting the desired test.

- d. Press CONTINUE after placing the MARK in position.
- e. The program will do each test in sequence.
- f. When test 9 is reached, the program will run until stopped with PROGRAM STOP.
- g. The program punches more tape than the reader's out-hopper will hold comfortably. It is suggested that the operator let the tape run freely from the reader into a waste basket.
- h. The operator is not restricted to using the MARK character or resync features to position the tape between the punch and reader. The slack should not exceed more than 2 ft between the devices, nor less than that provided by the resync routines or MARK character. A guide would be to never start with leader long enough to touch the bottom of the punch hopper after positioning it in the reader. Also, be sure the leader is blank.

6. ERRORS

6.1 Error Halts and Description

All halts are described in table 1.

6.2 Error Recovery

Recovery from error halts during any of the control logic tests is accomplished by pressing CONTINUE. The current test will be re-executed.

Recovery from error halts during any of the data check tests is done by pressing CONTINUE. The program will continue in sequence.

7. RESTRICTIONS

7.1 Starting Restrictions

MAINDEC 9A-D2CB-PH High Speed Reader Test must show the reader to be completely error-free before attempting to run the data check portion of this program.

7.2 Operating Restrictions

Any device other than the reader, Teletype, or punch must be turned OFF, to avoid unwanted interrupts.

8. MISCELLANEOUS

8.1 Execution Time

If looping on control logic tests 2 through 4, the time per pass is approximately 7 sec.

Data check tests 1 through 8 will require approximately 2 min.

8.2 Applications - Control Logic Tests

8.2.1 'Scope Mode - 'Scope mode for any of the control logic tests may be entered by placing ACS 1 up after an error halt and pressing CONTINUE.

Restarting a test with ACS 1 up will also provide 'scope mode. The starting addresses for each test and the procedure to follow is given below.

<u>Test #</u>	<u>'Scope IOT</u>	<u>Operator Action</u>	<u>Starting Address</u>
1		Press I/O RESET, then START.	21
2	PSA	Press I/O RESET, then START.	52
2	PSB	Place 700244 in location 51.	54
3	PSA	Place 700204 in location 51.	100
3	PSB	Place 700244 in location 51.	100
3A	PCF	Place 700204 or 700244 in location 57.	130
4		'Scope for punch interrupting before it is selected.	200
4	PSA	'Scope for no interrupt. Place 700204 in location 51.	213
4	PSB	'Scope for no interrupt. Place 700244 in location 51.	213
5		'Scope for NO TAPE always set.	300
5		NO TAPE will not set.	324
5		NO TAPE not set with POWER off.	324
6		No 'scope mode available. Punch must be at a complete halt before re-executing the test. S.A. = 400.	400

8.2.2 Fast 'Scope Mode Loop - A separate routine is provided which will punch any character specified by ACS 10-17. The loop uses PSA only, but this may be changed by inserting PSB (700244) into location 3102. The loop consists of six instructions, of which the first is CAF.

The purpose of the routine is to provide a very short loop for 'scoping if the methods previously described are not suitable.

Instructions:

- a. Set the ADDRESS SWITCHES to 3100.
- b. Place any character in ACS 10-17.
- c. Press I/O RESET, and then START.
- d. The loop may be halted with PROGRAM STOP.
- e. ACS 10-17 may be changed while the loop is running.

8.2.3 Looping on Control Logic Tests 2-4 - Tests 2 through 4 may be continuously executed to facilitate testing the control logic with power supply margins.

Looping with either the PSA or PSB IOT is provided.

To loop on tests 2 through 4 using PSA:

- a. Place ACS 2 up.
- b. Set the ADDRESS SWITCHES to 44.
- c. Press I/O RESET, and then START.
- d. The loop may be halted with PROGRAM STOP.

To loop on tests 2 through 4 using PSB:

- a. Place ACS 2 up.
- b. Set the ADDRESS SWITCHES to 270.
- c. Press I/O RESET, and then START.
- d. The loop may be halted with PROGRAM STOP.

8.2.4 Punch Speed Checks - A routine is provided to give decimal printouts of either the punch speed in characters per second, or the acceleration time in milliseconds.

The accuracy of the printout is dependent on the accuracy of the PDP-9's memory cycle time. The accuracy of the loop itself is less than .5% for acceleration time, and the accuracy for speed is ± 1 character per second.

Operating Procedure:

- a. Set the ADDRESS SWITCHES to 500.
- b. Place ACS 6 up for characters/sec, or ACS 7 up for acceleration time.
- c. Press I/O RESET, and then RESTART.
- d. The routine may be halted with PROGRAM STOP.

To compute characters/sec, the program punches tape for 8 seconds. The number of punch flags received is divided by 8, converted to decimal, and printed.

The acceleration time is computed by selecting the punch once and counting the number of passes required through a 100 μ sec loop before the flag is set.

8.4 Applications - Data Check Tests

8.4.1 Variable Punch Speed Control - ACS 4 through 17 will provide speed control with data check tests 1, 2, 3, 4, 7 and 9. With all 14 ACS up, the speed will be the slowest. Normal speed is restored by placing ACS 4 - 17 down.

Any combination of the 14 switches may be used, any may be raised or lowered while the program is running.

ACS 4 through 17 have no effect on the program while test 5, 6 or 8 is running.

9. PROGRAM DESCRIPTION

The PDP-9 Punch Test is designed to test the functions of the punch control logic with minimum operator intervention. One timing test is performed, and a 'scope loop is included in the program to aid in making any other mechanical or timing adjustments necessary. If the control logic tests run on normal and nominal power supply voltages, any remaining errors will be in the data paths.

The data check tests are designed to aid in finding mechanical failures in the punch itself. There are nine data check tests in all, each punching a different pattern.

TABLE 1 ERROR HALTS AND IDENTIFICATION

Test No.	C(PC)	Tag	Identification	Suggested Module
1	43	PE01	Punch flag set after IOT of 700110	1. S202 C24 2. S602 D24
2	70	PA02	Flag not set after PSA IOT.	
3	71	PB02	Flag not set after PSB IOT. If punch turned ON:	1. S202 C24 2. R302 C25 3. W520 D25 4. R111 D26 5. S107 D23
			If punch did not turn ON:	1. S202 C24 2. R303 CD27 3. S107 D23 4. S602 D24
3	114	PA03	Flag did not clear using PSA.	1. S202 C24
3	115	PB03	Flag did not clear using PSB.	2. S602 D24
4	240	P05A	Punch caused illegal interrupt.	1. R111 C26
4	242	P05B	Spurious interrupt. Turn off I/O devices other than TTY, reader and punch.	2. R111 D21
4	227	PA05	No interrupt using PSA.	1. R111 C26
	230	PB05	No interrupt using PSB.	2. R111 D21
5	315	FLGNOT	NO TAPE set with tape in punch. Check POWER switch wiring.	1. R111 D21
5	341	NOFLG	NO TAPE not set with no tape in punch. Check wiring on punch NO TAPE switch.	
5	341	NOFLG	No TAPE not set with POWER off. Check wiring.	
5	324	PHLT10	Remove tape from punch and press CONTINUE.	
5	346	HLT10A	End of no tape test.	
6	413	EA08	Flag is set sooner than 1-sec after punch is selected.	1. R302 C25
6	416	PRHLT6	End of 1-sec timing tests.	

TABLE 1 ERROR HALTS AND IDENTIFICATION (continued)

Test No.	C (PC)	Tag	Identification	Suggested Module
	1004	PRHLT7	Leader and MARK character have been punched. Place leader in reader with MARK character over drive sprocket. Adjust tape between reader and punch for minimum binding. Press CONTINUE to begin testing.	
	2040	SYHLT	Illegal sync character read during test 5 or 6. Press CONTINUE to ignore or restart test 5 or 6. Sync character should equal 377.	
	2461	NOTSYC	Illegal sync character read during tests 1, 2, 3, 4, 7, 8, or 9. Proceed as in above paragraph. Sync should equal 377.	
	2752	PHLT11	Halt as a result of ACS 0 up to halt on error. Press CONTINUE to go on.	
	3212		Halt which occurs after resyncing the program as specified by ACS 3. Press CONTINUE to resume current test, or restart any desired test.	
	3367	PHLT12	C(PC) immediately after loading.	

TABLE 2 RESTARTING ADDRESS

Control Logic Tests

Test 1	-	21	Illegal Instruction
Test 2	-	44	Set Flag With PSA and PSB
Test 3	-	100	Clear Flag With PSA and PSB
Test 3A	-	130	Clear Flag With PCF
Test 4	-	200	Interrupt Test
Test 5	-	300	No Tape Test
Test 6	-	400	1 Second Timing Test

Fast 'Scope Mode - 3100

Printout of Punch Speeds - 500

Data Check Tests

Test 1	-	1004
Test 2	-	1100
Test 3	-	1200
Test 4	-	1300
Test 5	-	1500
Test 6	-	1700
Test 7	-	2200
Test 8	-	2300
Test 9	-	2400

Resync Routine - 3175

10. LISTING

```

                .TITLE PTPD-9
/PDP-9 PAPER TAPE PUNCH DIAGNOSTIC
/INTERRUPT ROUTINE
                .ARS
                .LOC 0
00000          000000
00000          0
00001          740040      HLT
00002          620000      JMP* 0          /INTERRUPT ON ILLEGALLY
00003          740040      XX
00011          .LOC 11
00011          000000      0          /INDEX 11
00016          .LOC 16
00016          000000      0          /INDEX 16
00017          000000      0          /INDEX 17
/PUNCH TEST 1 - ILLEGAL INSTUCTION
                .LOC 21
00021          700004      RGNPCH CLOF
00021          703302      CAF
00022          103152      JMS CRLF
00023          777776      LAW -2
00024          040013      DAC 13
00025          700110      700110
00026          102670      JMS STAL70
00027          700201      PSF
00030          741000      SKP          /TEST FOR SET FLAG
00031          600041      JMP PFERR
00032          102767      JMS SCOPE
00033          741000      SKP          /ERROR
00034          600026      JMP RGNPCH+5
00035          440013      ISZ 13
00036          600026      JMP RGNPCH+5
00037          600044      JMP PTST2-2
00040          102767      PFERR JMS SCOPE
00041          740040      PE01  HLT          /ILLEGAL INSTRUCTION EXECUTED
00042          600026      JMP RGNPCH+5
00043          .EJECT

```

```

/TEST 2. SET PUNCH FLAG WITH PSA AND PSB
/
00044 200272 LAC CKPSA /PSA IOT
00045 040051 DAC PSAB
00046 777776 PTST2 LAW -2
00047 040013 DAC 13
00050 750000 CLA
/
00051 000000 PSAB 0 /WILL = PSA OR PSB
00052 100121 JMS STAL4 /STALL 4 SFC
00053 700201 PSF /MUST B SET
00054 600062 JMP PER3 /ERROR PATH
00055 440013 IS7 13
00056 600051 JMP PSAB
00057 102767 JMS SCOPE /SUCCESS PATH
00060 600100 JMP PTST3 /ONLY EXIT
00061 600051 JMP PSAB /SCOPE
00062 102767 PER3 JMS SCOPE /CHECK FOR SCOPE MODE
00063 741000 SKP
00064 600051 JMP PSAB /SCOPE
00065 200051 LAC PSAB
00066 540272 SAD CKPSA
00067 740040 PA02 HLT /FLAG NOT SETTING WITH PSA
00070 740040 PB02 HLT /FLAG NOT SETTING WITH PSB
00071 600051 JMP PSAB /RE-EXECUTE
.EJECT

```

		/TEST 3. RESET PUNCH FLAG WITH PSA, PSB AND PCF	
00100		.LOC 100	
00100	400051	PTST3 XCT PSAB	/FIRST SET THE FLAG
00101	700201	PSF	
00102	600101	JMP .-1	
00103	400051	XCT PSAB	/SET. NOW CLEAR IT
00104	700201	PSF	
00105	600116	JMP OK3	/SUCCESS PATH
00106	102767	JMS SCOPE	/ERROR PATH. FLAG STILL SET
00107	741000	SKP	
00110	600100	JMP PTST3	/SCOPE
00111	200272	LAC CKPSA	
00112	540051	SAD PSAB	
00113	740040	PA03 HLT	/FLAG DID NOT CLEAR USING PSA
00114	740040	PB03 HLT	/FLAG DID NOT CLEAR USING PSB
00115	600100	JMP PTST3	/RE-EXECUTE
		/	
00116	102767	OK3 JMS SCOPE	/SUCCESS PATH
00117	600130	JMP PTST3A	/EXIT
00120	600100	JMP PTST3	/SCOPE
		/	
00121	000000	STAL4 0	
00122	777703	LAW -75	/STALL 4 SEC
00123	042514	DAC RLKLMT	
00124	102670	JMS STAL70	
00125	442514	ISZ RLKLMT	
00126	600124	JMP .-2	
00127	620121	JMP* STAL4	
		.EJECT	

00130	400051	/CLEAR FLAG WITH PCF	
00131	700201	PTST3A XCT PSAB	/SET
00132	600131	PSF	
00133	700202	JMP -1	/CLEAR
00134	700201	PCF	
00135	600141	PSF	
00136	602767	JMP OK3A	/SUCCESS PATH
		JMP SCOPE	
		/	
00137	740040	PE04 HLT	/ERROR. PCF DID NOT CLEAR FLAG
00140	600130	JMP PTST3A	
00141	102767	OK3A JMS SCOPF	
00142	600200	JMP PTST4	/EXIT
00143	600130	JMP PTST3A	/SCOPE
		/	
		/PUNCH INTERRUPT TEST	
00200		.LOC 200	
		/	
00200	200274	PTST4 LAC JMP5	
00201	040001	DAC 1	
00202	700004	CLOF	
00203	703302	CAF	
00204	700042	ION	
00205	102670	JMS STAL70	/WAIT 210 MS FOR ILLEGAL INT.
00206	102670	JMS STAL70	
00207	102670	JMS STAL70	
00210	102767	JMS SCOPE	/OK, CHECK FOR SCOPE MODE
00211	741000	SKP	
00212	600200	JMP PTST4	
00213	200275	LAC JMP6	
00214	040001	DAC 1	
00215	703302	CAF	
00216	700042	ION	
00217	400051	XCT PSAB	
00220	100121	JMS STAL4	/GET PUNCH GOING
00221	102767	JMS SCOPE	/ERROR. NO INTERRUPT
00222	741000	SKP	
00223	600213	JMP PSRINT	/SCOPE
00224	200272	LAC CKPSA	
00225	540051	SAD PSAB	
00226	740040	PA05 HLT	/NO INTERRUPT USING PSA
00227	740040	PB05 HLT	/NO INTERRUPT USING PSB
00230	600213	JMP PSRINT	
		.EJECT	

```

/INTERRUPT SERVICE ROUTINE
/
00231 102767 ILLEG JMS SCOPE
00232 741000 SKP
00233 600200 JMP PTST4 /SCOPE
00234 700314 IORS /READ I/O STATUS
00235 742010 RTL
00236 741100 SPA /SEE IF PUNCH FLAG IS SET
00237 740040 P05A HLT /ERROR, PUNCH CAUSED ILLEGAL INTERRUPT
/CAF DIDN'T CLEAR FLAG,
/I/O STATUS
/SPURIOUS INTERRUPT AC = I/O STATUS

00240 700314 IORS
00241 740040 P05B HLT
00242 600200 JMP PTST4
00243 102767 TSPFLG JMS SCOPE /RETURN FROM LEGAL INTERRUPT
00244 741000 SKP
00245 600213 JMP PSRINT /SCOPE
00246 700314 IORS /I/O STATUS
00247 503370 AND (100000
00250 740200 SZA /MAKE SURE PUNCH CAUSED INTERRUPT
00251 600255 JMP OK4 /SUCCESS PATH
00252 700314 IORS /I/O STATUS
00253 740040 P05C HLT /SPURIOUS INTERT AC =I/O STATUS
00254 600213 JMP PSRINT
00255 200252 OK4 LAC P05C-1
00256 040001 DAC 1
00257 700002 IOF
00260 750004 LAS
00261 742010 RTL
00262 741100 SPA /CHECK FOR LOOP ON TESTS 2-4
00263 600046 JMP PTST2 /LOOP
00264 200273 LAC CKPSR
00265 540051 SAD PSAB
00266 600300 JMP NOTAPE /EXIT
00267 600045 JMP PTST2-1 /DO PSR IOT

/
00270 200273 PSRSRT LAC CKPSP
00271 600045 JMP PTST2-1 /RESTART HFRE IF LOOPING WITH PSR

/
00272 700204 CKPSA 700204
00273 700244 CKPSR 700244
00274 600231 JMP5 JMP ILLEG
00275 600243 JMP6 JMP TSPFLG
.EJECT

```

```

/TFST 5. NO TAPE TEST
/REMOVE ALL TAPE FROM PUNCH AFTER HALT AT LOCATION TAGGED PHLT10
/
00300 .LOC 300
/
00300 703302 NOTAPE CAF
00301 777766 LAW -12
00302 040534 DAC WORK
00303 750000 CLA
00304 700204 PSA
00305 700201 PSF
00306 600305 JMP .-1
00307 700314 IORS
00310 501362 AND K400
00311 741200 SNA
00312 600316 JMP OKNO /SEE IF FLAG IS SET
00313 102767 JMS SCOPE /SUCCESS PATH
00314 740040 FLGNOT HLT /ERROR. NO TAPE FLAG SET WITH
/TAPE IN PUNCH.
00315 600300 JMP NOTAPE
00316 440534 OKNO ISZ WORK
00317 600303 JMP NOTAPE+3 /TEST 10 TIMES
00320 102767 JMS SCOPE /CHECK FOR SCOPE MODE
00321 741000 SKP
00322 600300 JMP NOTAPE /SCOPE
00323 740040 PHLT10 HLT /REMOVE TAPE FROM PUNCH AT
/THIS POINT. PRESS CONTINUE
/TO TEST NO TAPE FLAG.
00324 777766 LAW -12
00325 040534 DAC WORK
00326 703302 TSFLG CAF
00327 750000 CLA
00330 700204 PSA
00331 700201 PSF
00332 600331 JMP .-1
00333 700314 IORS /I/O STATUS
00334 501362 AND K400
00335 740200 SZA /NO TAPE FLAG SHOULD BE SET
00336 600342 JMP OKFLG /SUCCESS PATH
00337 102767 JMS SCOPE
00340 740040 NOFLG HLT /ERROR. NO TAPE FLAG NOT SET
00341 600326 JMP TSFLG
00342 440534 OKFLG ISZ WORK
00343 600326 JMP TSFLG /TEST 10 TIMES
00344 102767 JMS SCOPE
00345 740040 HLT10A HLT /END OF NO TAPE TEST
00346 600324 JMP TSFLG-2 /SCOPE
.EJECT

```

00400		/TEST 6. 1 SEC. TIMING CHECK	
00400	200417	.LOC 400	
00401	040534	PTST6 LAC PDLY1S	/POWER SHOULD BE OFF
00402	750000	DAC WORK	
00403	700204	CLA	
00404	440534	PSA	/SELECT PUNCH
00405	741000	TST1S ISZ WORK	/START 1 SEC. TIMER
00406	600415	SKP	
00407	700201	JMP PRHLT6	/SUCCESS PATH
00410	600404	PSF	/MUST NOT SET
00411	102767	JMP TST1S	
00412	740040	JMS SCOPE	
00413	600400	EA08 HLT	/ERROR. DELAY < 1 SFC.
00414	102767	JMP PTST6	
00415	740040	JMS SCOPE	/SUCCESS PATH
00416	600400	PRHLT6 HLT	/END OF LOGIC TEST
00417	413667	JMP PTST6	/DO ALL AGAIN
		PDLY1S -364111	/1 SFC. DELAY CONSTANT
		/	
		/PUNCH SPEED CHECK	
00500		.LOC 500	
00500	750004	PNSPED LAS	
00501	503371	AND (400	
00502	741200	SNA	
00503	600616	JMP FLTST	/ACCEL. TIME
00504	141361	DZM CHACNT	
00505	765400	LAW -12400	
00506	040534	DAC WORK	
00507	777507	LAW -271	
00510	040535	DAC WORK1	
00511	700204	PSA	/GET UP TO SPEED
00512	700201	PSF	
00513	600512	JMP .-1	
00514	700204	PSA1 PSA	/SELECT PUNCH
00515	441361	ISZ CHACNT	/CHARACTER COUNTER
00516	600520	JMP TIME	
00517	740040	EA09 HLT	/SHOULD NEVER GET HERE
		.EJECT	

```

00520 440534 /8 SEC. TIME LOOP
00521 741000 TIME ISZ WORK
00522 600526 SKP
00523 700201 JMP TIMEX /CHARACTER PUNCHED IF SET
00524 600520 PSF
00525 600514 JMP TIME
JMP PSA1

/
TIMEX LAW -12400
DAC WORK
ISZ WORK1
JMP TIME /PRINT TOTAL PUNCHED/SEC.
JMS DEC
JMP PRSEC

/
WORK 0
WORK1 0

/
DEC 0
LAC CHACNT /CONVERT TO DECIMAL
RTR; RAR

NXTPWR AND (777
DAC CHACNT
DZM PCW /PCW = PARTIAL CONVERSION WORD
ISZ TABLE /TABLE USED TO STORE DIGITS
ISZ PWRTEN
LAC CHACNT
DAC COUNTB

SUBAGN CLL
TAD* PWRTEN
SNL /TRUE = CONVERTED ONE POWER X 10
JMP DONCON
DAC COUNTB
ISZ PCW /INCREMENT FOR EACH - RESULT
JMP SURAGN /SUBTRACT UNTIL RESULT IS +
LAC PCW
TAD N260 /MAKE ASCII
DAC* TABLE
LAC COUNTB
DAC CHACNT
LAC* PWRTEN
SAD PWRTEN+1
JMP* DEC
JMP NXTPWR /DO NEXT POWER OF TEN
,EJECT
    
```

00571	200606
00572	342761
00573	043237
00574	200611
00575	043236
00576	203235
00577	040011
00600	103163
00601	200615
00602	040613
00603	200612
00604	040610
00605	600500
00606	000000
00607	000000
00610	000610
00611	000000
00612	000610
00613	000613
00614	777766
00615	000613

PRSEC	LAC COUNTB	
	TAD N260	/MAKE ASCII
	DAC INF5+2	
	LAC TABLE+1	
	DAC INF5+1	
	LAC INF5	
	DAC 11	
	JMS MSG1	/PRINT CHARS./SEC.
	LAC PWRTEN+2	
	DAC PWRTEN	
	LAC TABLE+2	
	DAC TABLE	
	JMP PNSPED	
/		
COUNTB	0	
PCW	0	
TABLE	.	
	0	
	.-2	
PWRTEN	.	
	777766	/-10
	.-2	
	.EJECT	

00616 750004
 00617 503373
 00620 741200
 00621 600500

 00622 202761
 00623 040701
 00624 040702
 00625 040704
 00626 700204
 00627 700201
 00630 600627
 00631 700204
 00632 700201
 00633 741000
 00634 600714
 00635 440704
 00636 200704
 00637 740001
 00640 342760
 00641 340711
 00642 740200
 00643 600661
 00644 202761
 00645 040704
 00646 440702
 00647 200702
 00650 740001
 00651 342760
 00652 340711
 00653 740200
 00654 600670
 00655 202761
 00656 040702
 00657 440701
 00660 600673
 00661 500000
 00662 500000
 00663 500000
 00664 500000
 00665 500000
 00666 500000
 00667 500000

FLTST LAS
 AND (2000
 SNA
 JMP PNSPED

 /
 DUL LAC N260
 DAC 00
 DAC 01
 DAC 02
 PSA
 PSF
 JMP .-1
 PSA
 BIN PSF
 SKP
 JMP TOUT
 ISZ 02
 LAC 02
 CMA
 TAD ONE
 TAD N272
 SZA
 JMP FINUP
 LAC N260
 DAC 02
 ISZ 01
 LAC 01
 CMA
 TAD ONE
 TAD N272
 SZA
 JMP FINUP1
 LAC N260
 DAC 01
 ISZ 00
 JMP FINUP2
 FINUP AND 0
 AND 0
 AND 0
 AND 0
 AND 0
 AND 0
 AND 0
 .EJECT

/WAIT FOR FLAG
 /TRUF = ACCELERATION TIME

 /SELECT PUNCH

 /GOT FLAG, COMPUTE TIME

```

****
00670 500000 FINUP1 AND 0
00671 500000 AND 0
00672 500000 AND 0 /33 US TO HERE
00673 200713 FINUP2 LAC TUSE
00674 740000 NOP /3 US
00675 040534 DAC WORK /2 MORE = 38 US
00676 440534 ISZ WORK /00 20 LOOPS TO = 60 US
00677 600676 JMP , -1
00700 600632 JMP BIN /100 US TOTAL
/
00701 000260 00 260 /X
00702 000260 01 260 /X
00703 000256 256 /
/
00704 000260 02 260 /X
00705 000240 240 /SPACE
00706 000315 315 /M
00707 000323 323 /S
00710 000000 0
/
00711 000272 N272 272
00712 000700 PROAD 00-1
00713 777754 TUSE 777754 /-20
/
00714 200712 TOUT LAC PROAD
00715 040011 DAC 11
00716 103163 JMS MSG1 /PRINT
00717 600500 JMP PNSPED
.EJECT

```

↑↑↑↑

		/BASIC DATA CHECKS	
01000		.LOC 1000	
		/	
01000	102643	PSACHK	JMS PNLEDR /PUNCH 1.5 FT. LEADER
01001	102655		JMS PNMARK /PUNCH MARK CHARACTER
01002	102643		JMS PNLEDR /PUNCH MORE LEADER.
01003	740040	PRHLT7	HLT /DONE WITH SETUP. PLACE
			/THE LEADER IN THE READER WITH
			/THE MARK CHARACTER UNDER
			/THE TAPE GUIDE. POSITION
			/THE TAPE BETWEEN PUNCH AND
			/THE READER FOR MINIMUM BINDING.
			/PRESS CONTINUE TO BEGIN TESTING
01004	760261		
01005	043013		
01006	102655		
01007	777773		
01010	042512		
01011	777700		
01012	042514		
01013	042513		
01014	750000		
01015	103037		
01016	442514		
01017	601014		
01020	102767		
01021	741000		
01022	601013		
01023	102440		
01024	442512		
01025	601011		
01026	103106		
01027	601007		
01030	102767		
01031	741000		
01032	601007		
01033	102602		
01034	601100		
		LAW 261	
		DAC PRINT6+1	
		JMS PNMARK	
		LAW -5	
		DAC BLSTOR	
		LAW -100	/-64 DECIMAL
		DAC BLKLMT	
		DAC BLSTOR+1	
		CLA	
		JMS GOPNCH	/PUNCH ZEROS
		ISZ BLKLMT	/CHECK FOR BLOCK OF 84
		JMP .-3	
		JMS SCOPE	/CHECK FOR SCOPE MODE
		SKP	
		JMP PNZERO+2	
		JMS READ	/READ ONE BLOCK
		ISZ BLSTOR	/CHECK FOR 5 BLOCKS
		JMP PNZERO	
		JMS REPEAT	
		JMP PNZERO-2	
		JMS SCOPE	/CHECK FOR SCOPE MODE
		SKP	
		JMP PNZERO-2	/SCOPE
		JMS FINPCH	/READ REST OF DATA
		JMP PNSKEW	
		.EJECT	

01100		/SKEW CHFCX	
			.LOC 1100
01100	760262	/	
01101	043013	PNSKEW	LAW 262
01102	102655		DAC PRINT6+1
01103	777761		JMS PNMARX
01104	042512		LAW -17
01105	777700		DAC BLSTOR
01106	042513		LAW -100
01107	777740		DAC BLSTOR+1
01110	042514	CHNL1	LAW -40
01111	202760		DAC BLKLMX
01112	103037		LAC ONE
01113	442514		JMS GOPNCH
01114	601111		ISZ RLKLMX
01115	777740		JMP .-3
01116	042514		LAW -40
01117	201574		DAC RLKLMX
01120	103037		LAC K200
01121	442514		JMS GOPNCH
01122	601117		ISZ RLKLMX
01123	102767		JMP .-3
01124	741000		JMS SCOPE
01125	601105		SKP
01126	102440		JMP CHNL1-2
01127	442512		JMS READ
01130	601105		ISZ BLSTOR
01131	103106		JMP CHNL1-2
01132	601103		JMS REPEAT
01133	102767		JMP PNSKEW+3
01134	741000		JMS SCOPE
01135	601105		SKP
01136	102602		JMP CHNL1-2
01137	601200		JMS FINPCH
			JMP ALTN10
			.EJECT

/-64 DECIMAL

/-32

/CHANNEL 8

/CHECK FOR SCOPE MODE

/READ ONE BLOCK

/CHECK FOR 15 BLOCKS

↑↑↑↑

01200		/ALL CHANNELS	
		.LOC 1200	
		/	
01200	760263	ALTN10	LAW 263
01201	043013		DAC PRINT6+1
01202	102655		JMS PNMARK
01203	777761		LAW -17
01204	042512		DAC BLSTOR
01205	777700	ALTPCH	LAW -100
01206	042513		DAC BLSTOR+1
01207	042514		DAC BLKLMT
01210	202516		LAC TH77
01211	103037		JMS GOPNCH
01212	442514		ISZ BLKLMT
01213	601210		JMP .-3
01214	102767		JMS SCOPE
01215	741000		SKP
01216	601203		JMP ALTPCH-2
01217	102440		JMS READ
01220	442512		ISZ BLSTOR
01221	601205		JMP ALTPCH
01222	103106		JMS REPEAT
01223	601203		JMP ALTPCH-2
01224	102767		JMS SCOPE
01225	741000		SKP
01226	601203		JMP ALTPCH-2
01227	102602		JMS FINPCH
01230	601300		JMP SLDONE
			.EJECT

			/-64 DECIMAL
			/377
			/CHECK FOR SCOPE MODE
			/READ ONE BLOCK
			/CHECK FOR 15 BLOCKS
			/CHECK FOR SCOPE MODE
			/READ REST OF DATA

++++

```

01300      /SLIDING ONE
           .LOC 1300
01300      /
           SLDONE   LAW 264
01301      DAC PRINT6+1
01302      JMS PNMARK
01303      LAW -17          /-15
01304      DAC BLSTOR
01305      PCHONE   LAW -130
01306      DAC BLSTOR+1
01307      DAC BLKLMY
01310      CLA:CLL:CML
01311      DAC STORE
01312      SHIFT1  LAW -7
01313      DAC CHACNT
01314      LAC STORE
01315      RAL
01316      DAC STORE
01317      JMS GOPNCH
01320      ISZ BLKLMY          /CHECK FOR BLOCK OF 64
01321      JMP CKRX
01322      JMS SCOPE          /CHECK FOR SCOPE MODE
01323      SKP
01324      JMP SCPMOD
01325      JMS READ
01326      ISZ BLSTOR          /CHECK FOR 15 BLOCKS
01327      JMP PCHONE
01330      JMS REPEAT
01331      JMP PCHONE-2
01332      JMS SCOPE          /CHECK FOR SCOPE MODE
01333      SKP
01334      JMP PCHONE-2
01335      JMS FINPCH          /READ REST OF DATA
01336      JMP SLDZRO          /EXIT
01337      SCPMOD   LAW -100
01340      DAC BLKLMY
01341      CK8X    ISZ CHACNT
01342      JMP SHIFT1+2
01343      LAC K400
01344      DAC STORE
01345      LAW -7
01346      DAC CHACNT
01347      SHFTA1  LAC STORE
01350      RAR
01351      DAC STORE
01352      JMS GOPNCH
01353      ISZ BLKLMY
01354      SKP
01355      JMP CHKEND
01356      ISZ CHACNT
01357      JMP SHFTA1
01360      JMP SHIFT1-2
01361      /
           CHACNT  0

```

PTPD-9 PAGE 16

01362 000400

K400

400
.EJECT

```

01500                                /SLIDING ZERO. READ AT PUNCH SPEED
                                .LOC 1500
/
01500      760265                    /SLDZRO   LAW 265
01501      043013                    DAC PRINT6+1
01502      102655                    JMS PNMARX
01503      201777                    LAC INCR0
01504      040001                    DAC 1
01505      750000                    CLA
01506      040534                    DAC WORK
01507      040535                    DAC WORK1
01510      700104                    RSA
01511      102000                    JMS PNONE
01512      776177                    LAW -1601          /-960
01513      042512                    DAC RLSTOR
01514      201362                    PCHZRO   LAC K400
01515      041573                    DAC STORE
01516      777770                    SHIFTO   LAW -10
01517      041361                    DAC CHACNT
01520      201573                    LAC STORE
01521      744020                    RCR
01522      041573                    DAC STORE
01523      242516                    XOR TH77
01524      102000                    JMS PNONE
/
01525      102767                    CHKBLK  JMS SCOPE
01526      741000                    SKP
01527      601541                    JMP CHECK8
01530      442512                    ISZ RLSTOR          /CHECK FOR 10 BLOCKS
01531      601541                    JMP CHECK8
01532      103106                    JMS REPEAT          /CHECK FOR LOOP ON TEST
01533      601503                    JMP SLDZRO+3
01534      102767                    JMS SCOPE
01535      741000                    SKP
01536      601512                    JMP PCHZRO-2        /SCOPE
01537      102530                    JMS DONPCH
01540      601700                    JMP ADJCH           /EXIT
                                .EJECT

```

01541	441361	CHECK8	ISZ CHACNT	
01542	601520		JMP SHFT0+2	
01543	202760		LAC ONE	
01544	041573		DAC STORE	
01545	777772		LAW -6	
01546	041361		DAC CHACNT	
01547	201573	SHFTA0	LAC STORE	
01550	744010		RCL	
01551	041573		DAC STORE	
01552	242516		XOR TH77	
01553	102000		JMS PNONE	
01554	102767		JMS SCOPE	
01555	741000		SKP	
01556	601570		JMP CHCK8	
01557	442512		ISZ BLSTOR	/CHECK FOR 15 BLOCKS
01560	601570		JMP CHCK8	
01561	103106		JMS REPEAT	
01562	601503		JMP SLDZRO+3	
01563	102767		JMS SCOPE	
		/		
01564	741000		SKP	
01565	601512		JMP PCHZRO-2	/SCOPE
		/		
01566	102530		JMS DONPCH	/READ REST OF DATA
01567	601300		JMP SLDONE	/EXIT
01570	441361	CHCK8	ISZ CHACNT	
01571	601547		JMP SHFTA0	
01572	601514		JMP PCHZRO	
		/		
01573	000000	STORE	0	
01574	000200	K200	200	
01575	000277	K277	277	
			.EJECT	

```

01700
01700 760266
01701 043013
01702 102655
01703 201777
01704 040001
01705 750000
01706 040534
01707 040535
01710 700104
01711 102000
01712 776177
01713 042512
01714 201362
01715 041573
01716 777770
01717 041361
01720 777771
01721 042514
01722 202516
01723 102000
01724 442512
01725 601727
01726 601773
01727 442514
01730 601722
01731 201573
01732 744020
01733 041573
01734 242516
01735 102000
01736 442512
01737 601741
01740 601773
01741 441361
01742 601720

/TEST ADJACENT CHANNELS
/RFAD AT PUNCH SPEED
.LOC 1700
/
ADJCH LAW 266
DAC PRINT6+1
JMS PNMARX
LAC INCR0
DAC 1
CLA
DAC WORK
DAC WORK1
RSA
JMS PNONE
ADJCNT LAW -1601 /-960
DAC BLSTOR
LAC K400
DAC STORE
LAW -10
DAC CHACNT
LAW -7
FRONES DAC BLKLMT
LAC TH77 /377
JMS PNONE /PUNCH ONE FRAME
ISZ BLSTOR /CHECK FOR 15 BLOCKS PUNCHED
JMP CHECK7
JMP DNADJ /READ REST OF DATA
CHECK7 ISZ BLKLMT /CHECK FOR 7 FRAMES OF 377
JMP FRONES /NOT 7 YET
LAC STORE /DROP ONE CHANNEL
RCR
DAC STORF
XOR TH77
JMS PNONE /PUNCH ONE FRAME
ISZ BLSTOR /CHECK FOR DONE PUNCHING
JMP CKDR0P
JMP DNADJ
CKDR0P ISZ CHACNT
JMP FRONES-2 /DO 7 MORE FRAMES OF 377
.EJECT

```

01743	202760		LAC ONF	
01744	041573		DAC STORE	
01745	777772		LAW -6	
01746	041361		DAC CHACNT	
01747	777771		LAW -7	
01750	042514		DAC RLKLMT	
01751	202516	FRONE	LAC TH77	
01752	102000		JMS PNONE	/PUNCH ONE FRAME
01753	442512		ISZ BLSTOR	/CHECK FOR 15 BLOCKS
01754	601756		JMP CHCK7	
01755	601773		JMP DNADJ	
01756	442514	CHCK7	ISZ RLKLMT	/CHECK FOR 7 FRAMES OF 377
01757	601751		JMP FRONE	
01760	201573		LAC STORE	
01761	744010		RCL	/DRP ONE CHANNEL
01762	041573		DAC STORE	
01763	242516		XOR TH77	
01764	102000		JMS PNONE	/PUNCH ONE FRAME
01765	442512		ISZ BLSTOR	/CHECK FOR 15 BLOCKS
01766	601770		JMP CKDRP	
01767	601773		JMP DNADJ	
01770	441361	CKDRP	ISZ CHACNT	
01771	601747		JMP FRONE-2	/DO 7 MOR FRAMES OF 377
01772	601714		JMP ADJCNT+2	
01773	103106	DNADJ	JMS REPEAT	/CHECK FOR LOOP ON TEST
01774	601703		JMP ADJCH+3	/LOOP
01775	102530		JMS DONPCH	
01776	602200		JMP PNBRY	
01777	440000	INCR0	ISZ 0	
02000	000000	PNONE	0	/PUNCH ROUTINE
02001	060017		DAC* 17	
02002	700204		PSA	
02003	200017		LAC 17	
02004	542511		SAD ENDBIN	/CHECK FOR END OF DATA BIN
02005	602013		JMP TOPBIN	
02006	102767	CKSCP	JMS SCOPE	
02007	602016		JMP TOPBIN+3	
02010	700201		PSF	
02011	602010		JMP *-1	/SCOPE
02012	622000		JMP* PNONE	
		/		
02013	202510	TOPBIN	LAC DATAPL	/START AT TOP AGAIN
02014	040017		DAC 17	
02015	602006		JMP CKSCP	
		/		
02016	440534		ISZ WORK	
02017	200535		LAC WORK1	
02020	740200		SZA	
02021	602061		JMP SUR1	
02022	602554		JMP SRVINT	
			.EOT	

```

/PDP-9 PAPER TAPE PUNCH DIAGNOSTIC - TAPE 2
/READ ROUTINE FOR TEST 5 AND 6
/
02023 202507 READA LAC FLAG
02024 740200 SZA /SEE IF IN SYNC
02025 602042 JMP SYNCY /IN SYNC
02026 700112 RRR
02027 740200 SZA
02030 602035 JMP ZRONOT /GOT SYNC CHAR
02031 200534 LAC WORK /SEE IF COUNT = 0
02032 740200 SZA
02033 602061 JMP SUR1 /NO. SUBTRACT 1
02034 602057 JMP TADD1 /YES. ADD 1
02035 542516 ZRONOT SAD TH77 /SYNC MUST = 377
02036 741000 SKP
02037 740040 SYHLT HLT /ILLEGAL SYNC. PRESS CONTINUE TO IGNORE
/OR RESTART TEST
/SET IN SYNC FLAG
02040 442507 ISZ FLAG
02041 602031 JMP ZRONOT-4
02042 700112 SYNCY RRR /IN SYNC, READ BUFFER
02043 560016 SAD* 16 /COMPARE
02044 741000 SKP /OK
02045 602067 JMP SETUP /ERROR
02046 200016 LAC 16
02047 542511 SAD ENDBIN
02050 741000 SKP
02051 602054 JMP ,+3
02052 202510 LAC DATARL
02053 040016 DAC 16
02054 200534 LAC WORK
02055 740200 SZA
02056 602061 JMP ,+3
02057 440535 TADD1 ISZ WORK1
02060 602554 JMP SRVINT
02061 777777 SUR1 LAW -1
02062 340534 TAD WORK /SUB. 1 FROM COUNT
02063 040534 DAC WORK
02064 140535 DZM WORK1
02065 700104 RSA
02066 602554 JMP SRVINT /GO TO INTERRUPT ROUTINE
02067 102702 SETUP JMS SETUP4 /PRINT ERROR INFORMATION
02070 200016 LAC 16
02071 542511 SAD ENDBIN
02072 741000 SKP
02073 602076 JMP ,+3
02074 202510 LAC DATARL
02075 040016 DAC 16
02076 140534 DZM WORK
02077 140535 DZM WORK1
02100 700104 RSA /SELECT READER
02101 622000 JMP* PNONE /START INTERRUPT SEQUENCE AGAIN
.EJECT

```

```

/PUNCH BINARY COUNT WITH PSA
/
02200          760267          PNRNRY          .LOC 2200
02200          043013          LAW 267
02201          102655          DAC PRINT6+1
02202          777761          JMS PNMAR
02203          042512          LAW -17
02204          141573          DAC BLSTOR
02205          777700          DZM STORE
02206          042514          LAW -100
02207          042513          DAC BLKLM
02210          441573          PLUS1          DAC BLSTOR+1
02211          201573          ISZ STORE          /INCREMENT COUNT
02212          502516          LAC STORE
02213          103037          AND TH77
02214          442514          JMS GOPNCH
02215          602211          ISZ BLKLM          /CHECK FOR BLOCK OF 84
02216          102767          JMP PLUS1
02217          741000          JMS SCOPE          /CHECK FOR SCOPE MODE
02220          602206          SKP
02221          102440          JMP PLUS1-3
02222          442512          JMS READ          /READ A BLOCK
02223          602206          ISZ BLSTOR          /CHECK FOR 15 BLOCKS
02224          103106          JMP PLUS1-3
02225          602203          JMS REPEAT
02226          102767          JMP PNRNRY+3
02227          741000          JMS SCOPE          /CHECK FOR SCOPE MODE
02230          602203          SKP
02231          102602          JMP PNRNRY+3
02232          602300          JMS FINPCH          /READ REST OF DATA
02233          JMS PSRNRY
          .EJECT
    
```

```

/PUNCH BINARY COUNT WITH PSB
/
02300          760270      PSRNRV      .LOC 2300
02300          043013      LAW 270
02301          102655      DAC PRINT6+1
02302          777761      JMS PNMARK
02303          042512      LAW -17
02304          141573      DAC RLSTOR
02305          777700      DZM STORE
02306          042513      LAW -100
02307          042514      DAC RLSTOR+1
02310          441573      PLUS18     DAC BLKLMT
02311          201573      ISZ STORE
02312          502516      LAC STORE
02313          700244      AND TH77
02314          700201      PSB        /PUNCH BINARY
02315          602315      JMP .-1
02316          501574      AND K200
02317          541574      SAD K200
02320          602327      JMP .+6
02322          201573      LAC STORE
02323          241574      XOR K200
02324          501575      AND K277
02325          060017      DAC* 17
02326          602332      JMP .+4
02327          201573      LAC STORE
02330          501575      AND K277
02331          060017      DAC* 17    /STORE GOOD DATA
02332          200017      LAC 17
02333          542511      SAD ENDBIN
02334          602354      JMP RINTOP
02335          442514      CK64       ISZ BLKLMT    /CHECK FOR BLOCK OF 64
02336          602311      JMP PLUS18
02337          102767      JMS SCOPE  /CHECK FOR SCOPE MODE
/
02340          741000      SKP
02341          602306      JMP PLUS18-3
02342          102440      JMS READ   /READ A BLOCK
02343          442512      ISZ RLSTOR /CHECK FOR 15 BLOCKS
02344          602306      JMP PLUS18-3
02345          103106      JMS REPEAT
02346          602303      JMP PSRNRV+3
02347          102767      JMS SCOPE  /CHECK FOR SCOPE MODE
02350          741000      SKP
02351          602303      JMP PSRNRV+3
02352          102602      JMS FINPCH /READ REST OF DATA
02353          602400      JMP PNHSKP-2 /EXIT
02354          202510      RINTOP     LAC DATARL
02355          040017      DAC 17
02356          602335      JMP CK64
                .EJECT

```

```

/PUNCH RANDOM STALL AND BLOCK LENGTHS WITH FIXED DATA PATTERN
/
02400 .LOC 2400
/
02400 760271 LAW 271
02401 043013 DAC PRINT6+1
02402 102655 PNHSKIP JMS PNMARK
02403 777740 LAW -40
02404 042514 DAC BLKLMT
02405 103310 NWBLK JMS GENRAN /RANDOM
02406 042512 DAC BLSTOR /BLOCK LENGTH
02407 202512 LAC BLSTOR
02410 502433 AND K7
02411 740001 CMA
02412 042512 DAC BLSTOR
02413 750000 NEWDAT CLA
02414 103037 JMS GOPNCH
02415 202516 LAC TH77
02416 103037 JMS GOPNCH
02417 102767 JMS SCOPE /CHECK FOR SCOPE MODF
02420 741000 SKP
02421 602403 JMP PNHSKIP+1
02422 442512 ISZ BLSTOR
02423 741000 SKP
02424 602434 JMP STLL
02425 442514 ISZ BLKLMT
02426 602413 JMP NEWDAT /PUNCH MORE DATA
02427 777700 DNBLK LAW -100
02430 042513 DAC BLSTOR+1
02431 102440 JMS READ
02432 602403 JMP PNHSKIP+1
/
02433 000007 K7 7
/
02434 102540 STLL JMS STALL
02435 442514 ISZ BLKLMT
02436 602405 JMP NWRLK
02437 602427 JMP DNRLK
/
.EJECT

```