

IDENTIFICATION

Product Code: MAINDEC-9A-D7AD-D
Product Name: PDP-9 Basic Exerciser
Date Created: May 3, 1968
Maintainer: Diagnostics Group
Author: J. W. Richardson

Jan-369-9239



1. ABSTRACT

The PDP-9 Basic Exerciser is designed to exercise the CP, core memory and I/O devices associated with a basic PDP-9 configuration. Once initiated, the program will perform tests on all operate and memory reference instructions, tests on the adder, memory checkerboard patterns, tests on the real-time clock, punch, reader, Teletype and program interrupt.

The Basic Exerciser contains a condensed version of the PDP-9 Instruction Test, Parts 1 and 2, and a memory checkerboard test similar to the PDP-9 Basic Memory Checkerboard Test. These tests run continuously, and are interrupted by the punch, reader or Teletype at a device rate. The real-time clock will interrupt and suspend all operations at random time intervals. The instruction test or I/O device resumes operation after the clock interrupt has been serviced.

Nine ACS functions are provided to enable the operator to (1) inhibit the instruction and memory tests and run the real-time clock, program interrupt, and the punch, read, print sequence alone; (2) inhibit program interrupts and run the instruction and memory tests alone; (3) loop continuously on the adder test; (4) loop continuously on the memory checkerboard test; (5) inhibit program relocation; (6) inhibit the real-time clock, but continue testing with program interrupt and all other devices enabled; (7) run the instruction and memory tests, and the clock and punch with the read and print sequence inhibited; (8) run the reader, real-time clock and instruction and memory tests with the punch and Teletype inhibited; (9) run the read and print sequence, real-time clock and instruction and memory tests with the punch inhibited.

2. REQUIREMENTS

2.1 Equipment

A basic PDP-9 configuration.

2.2 Storage

The program requires all 8K of core memory to perform all tests. The program initially resides in memory locations 00000 to 7730. When the program is relocated to the higher 4K field, it occupies locations 10022 to 17730.

3. LOADING PROCEDURE

The tape supplied is punched in the HRI mode.

- a. Set the ADDRESS switches to 00000.
- b. Place all AC switches down.

- c. Place the HRI tape in the reader.
- d. Press I/O RESET, and then READ-IN.

The program is not self-starting.

4. STARTING PROCEDURE

4.1 Starting Addresses

22 or 10022 if the program is currently in the upper 4K field.

4.1.1 Restarting Addresses - 26 or 10026 if the program is currently in the upper 4K field.

4.2 Operator Action

- a. Set the ADDRESS switches to 22.
- b. Place all ACS down for normal program operation. See paragraph 5.1 for use of the ACS to inhibit certain portions of the program.
- c. Press I/O RESET, and then START.
- d. Approximately 3-1/2 feet of leader will be punched. This leader is blank except for one frame which has all channels punched.
- e. Place the punched frame directly over the reader drive sprocket, and position the tape between reader and punch for minimum binding.
- f. Press CONTINUE.
- g. The program will run until an error halt occurs, or manually stopped by the operator.
- h. Steps c through f above are similar to the procedure performed with the PDP-9 punch test.

4.2.1 Restarting Procedure -

- a. Set the ADDRESS switches to 26.
- b. Place all ACS down for normal program operation. See paragraph 5.1 for use of the ACS to inhibit certain portions of the program.
- c. If the punch, read and print sequence is not inhibited, make sure there is tape in the reader. The tape does not have to be blank leader when restarting.
- d. Press I/O RESET, and then START.
- e. The program will run until an error halt occurs, or manually stopped by the operator.

5. OPERATING PROCEDURE

5.1 Operational Switch Settings

Normal program operation is achieved by placing all ACS down before starting from locations 22 or 26.

The operator is provided nine options with which to modify the operation of the program. These may be selected by placing any one or a combination of ACS 0 through 8 up before starting from locations 22 or 26.

To make changes in the ACS settings, the program must be stopped by the operator before the changes are made. The program must then be restarted from location 26 (or 22 if new leader is desired). The program may not recognize the new ACS settings if the above procedure is not followed.

ACS Functions

- 0 (1) Run only the punch, read and print sequence plus the real-time clock. Program interrupt will be enabled.
- 1 (1) Inhibit the punch, read and print sequence. Program interrupt is disabled. The real-time clock is on. The complete instruction test and memory checkerboard test will be performed.
- 2 (1) Loop continuously on the "add random pairs" test. The real-time clock, punch, read and print sequence plus program interrupt will be enabled unless specified otherwise by an ACS.
- 3 (1) Loop continuously on the memory checkerboard test. Program relocation will not take place. Program action otherwise is the same as that described for ACS 2.
- 4 (1) Inhibit program relocation. Unless otherwise specified, the program will run in a normal way, but will not relocate from its current 4K field location to the opposite field after completing the memory checkerboard test.
- 5 (1) Inhibit clock. Unless otherwise specified, program action is normal except that the clock should always be off.
- 6 (1) Inhibit the reader and TTY. The punch will run continuously. Tape must be in the reader to prevent the no-tape indicator from being set. Program action is normal unless otherwise specified.
- 7 (1) Inhibit the punch and TTY. The reader will run continuously. A loop or fan-fold tape with any data may be used. Program action is normal unless otherwise specified.
- 8 (1) Inhibit the punch. The reader will read 52 characters at full speed and then halt. The TTY will then print the 52 characters read. Any tape loop or fan-fold tape may be used. Program action is normal unless otherwise specified.

Any combination of the nine ACS may be used, as long as the operations do not conflict; i.e., if ACS 2 and 3 are both up, the add random pairs test would be looped. Memory checkerboard would not be run unless the program is restarted with ACS 3 alone.

The I/O devices may be controlled with several combinations of ACS 6, 7 and 8. If ACS 7 and 8 are both up the reader will run continuously, as if ACS 7 only were up. If ACS 6 and 7 or 6 and 8 are up, all devices will be inhibited. Program interrupt and the real-time clock will be enabled unless otherwise specified.

5.2 Subroutine Abstracts

The PDP-9 Basic Exerciser may be thought of as three separate programs; i.e., the instruction and memory tests; punch, read and print sequence, and operation of the real-time clock. The instruction and memory tests will be interrupted, at a device rate, by the punch, reader or Teletype. The clock will randomly interrupt any of the above operations at a rate determined by the program. After each clock interrupt, the clock is reinitialized with a new number obtained by a random number generator. The clock interrupts should occur no less than 2 seconds apart, nor more than 9 seconds. The clock interrupts take first priority, followed by the Teletype, reader, and punch.

5.2.1 Instruction and Memory Tests - The instruction test portion of the Basic Exerciser performs tests on all operate group and memory reference instructions. The individual instructions are looped a random number of times before proceeding to the next test. The maximum number of loops made on any one test is 32,767.

The adder is tested using two different methods. The first performs bit by bit tests on the adder using the ADD instruction. Besides checking for correct results after an addition, the link is tested during overflow and no overflow conditions.

The second method, the "Add Random Pairs" test, tests the adder using one pair of random numbers (A and B) and their 1's complement values (-A and -B), and the ADD instruction. These four values are added in various combinations, the results of which are compared against precalculated results. The precalculated results are obtained by adding the two pairs together using the TAD instruction. Four additions are made, the results of which are used in the test. The link is tested after each addition. If it is a 1, a 1 is added to the result to simulate an end-around-carry.

The numbers added and their sums are indicated in the listing using the following symbols:

-B+(-A)	=	SUMNEG
A+B	=	SUMPOS
B-A	=	BMASUM
A-B	=	AMBSUM

The values of A, -A, B and -B plus their sums are used to test the combinations of ADD's shown below.

<u>ADD</u>	<u>SUM SHOULD EQUAL</u>
A + B	SUMPOS
-B + A	AMBSUM
-B + (-A)	SUMNEG
B - A	BMASUM
(A + B) - A	BPOS (B)
(B - A) - B	ANEG (-A)
(-A - B) + A	BNEG (-B)
(A - B) + B	APOS (A)
777777 + A	APOS
A + B - A	BPOS
A + B -A -A	BMASUM (B - A)
A + B -A -A -B	ANEG
A + B -A -A -B -B	SUMNEG (-A -B)
A + B -A -A -B -B + A	BNEG
A + B -A -A -B -B + A + A	AMBSUM (-B + A)
A + B -A -A -B -B + A + A + B	APOS

After completing one pass of the above tests, a second pass is made on the same tests. The second pass makes all "B" constants "A", and all "A" constants "B" before repeating.

Immediately following the second pass, one random number and its 1s complement is obtained and saved in APOS and ANEG, respectively. Bit 0 of APOS is tested for equaling 0 or 1. If the value is 1, the bit remains unchanged, and the respective bit in the complement number is changed to equal a 1. The two numbers are then added together, the sum of which should equal all 0s except for bit 0. If the ADD is successful, the program continues testing all other bit positions in the same manner.

Example: (Bit 0 altered)

<u>Step</u>	<u>APOS Value</u>	<u>ANEG Value</u>
1	577776	200001
2	577776	600001 (bit 0 altered)
3	Add together. Result should = altered bit.	
	$ \begin{array}{r} 101\ 111\ 111\ 111\ 111\ 110 \\ +\ 110\ 000\ 000\ 000\ 000\ 001 \\ \hline 011\ 111\ 111\ 111\ 111\ 111 \\ +\ 1 \\ \hline 100\ 000\ 000\ 000\ 000\ 000 \end{array} $	(end around carry)

The sum equals the altered bit position.

After completing the adder tests, the remaining memory reference instructions are tested. The last test performed by the exerciser is the memory checkerboard test. This routine writes and reads four different checkerboard patterns similar to the Basic Memory Checkerboard program. The memory test is looped three times before program relocation takes place.

After completing the memory checkerboard test, the program relocates the entire Basic Exerciser to the opposite 4K field. All memory reference instructions and memory locations used for testing are adjusted accordingly. All locations within the program which reference any memory location between 0 and 21 are not adjusted. These locations are used during program interrupts, autoindexing tests, etc., and must not be altered. Program interrupt is disabled while relocation is taking place.

After relocation of the program is completed, the exerciser is automatically restarted at location 70 (or 10067). This location is tagged SEQUEN. The operator is able to determine the location of the program by observing MB bit 5. This bit will glow brightly when the program is in the higher 4K field, as compared to when residing in the lower 4K field.

The Teletype BELL will ring once for each completed pass of the program. One pass is defined as the program performing all tests from each 4K field, and then relocating back to the field in which the program was first initiated.

When operating the Basic Exerciser with program interrupt inhibited (ACS 1 up), the message "COMPLETE" will be printed after five complete passes of the program. This message is printed after ten passes when ACS 4 (inhibit relocation) as well as ACS 1 is up. This feature is included as a means to determine the number of successful passes completed by the program if it is to be run for extended periods of time.

5.2.2 Punch, Read, Print Sequence - The instruction and memory tests will be interrupted, at a device rate, by the punch, reader, or Teletype. The data punched consists of the alphabet characters, followed by numbers 0 through 9, with a space character being punched between each letter or numeral character. The reader will read the tape at punch speed, storing away any punched character. Frames of all 0s are ignored. The punch and read sequence consists of 52 ASCII characters punched, read and stored away in an input buffer (tagged TTBUFA). After reading the 52nd character, the contents of TTBUFA are transferred directly to another 52 - location buffer tagged TTBUFB. This second buffer is provided to enable the operator to stop the program and compare the contents of either buffer A or buffer B with the punched data on tape. Punch and read operation is halted after the 52nd character is read and stored. The contents of TTBUFB are then printed on the Teletype. The punch and read sequence continues immediately after the 52nd character is printed. The data punched and read should appear on the Teletype as the example below.

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9
```


The punch and read sequence generates 72 characters altogether, even though only 52 are punched, read and printed at one time. The alphabet and numbers sets with a space between each character enables a full line to be printed. The spacing also enables the operator to more easily detect a misprinted character. Each group of 72 characters is separated by 8 blank frames. The group which is positioned in the reader is the current line being printed. A carriage return and a line feed is punched at the end of each group. The program will punch 6 extra blank frames between two groups, approximately every fourteenth group, to enable the slack between reader and punch to remain constant.

5.2.3 Use of the Real Time Clock - The instruction and memory tests, and the punch, read and print sequence are both interrupted randomly by the real-time clock. When a clock interrupt occurs, all other operations are halted until the clock interrupt has been serviced. The program allows the clock to continue incrementing for 1/2 second after the interrupt occurs. This is evident during every clock interrupt by observing MB indicators 12 through 17 incrementing. Immediately after the clock has incremented, an additional 1/2 second, it is reset to a new random value. This value is chosen by the program to ensure that the clock interrupts no sooner than 2 seconds, nor later than 9 seconds. The clock is again enabled after being reset to a new value, and the instruction test or read, punch and print sequence is allowed to continue from the point of interrupt.

At times, the Basic Exerciser may appear to be caught in a loop after a clock interrupt occurs. The console indicators will show the clock and PIE as being disabled, and the punch, read and print sequence will be halted for several seconds. The program during this time is attempting to generate a number for the clock which falls within the 2 to 9 second limit. All operations will be resumed as soon as a suitable random value is found.

The operator may disable the clock interrupts by restarting from location 26 with ACS 5 up.

5.2.4 Interrupt Service Routine - Program interrupts by the clock, punch, reader or Teletype are all serviced by a common routine. A common routine for reentering the instruction test is also used.

Locations 0 through 6 are used to save the contents of the AC and PC immediately after an interrupt occurs. The contents of the AC are stored in the location tagged SAVAC. The contents of the link and PC are stored in the location tagged RJMP. The program then enters a routine which determines which of the four devices interrupted the program. This routine is tagged SRVINT. SRVINT will test for device flags in the following order: clock, Teletype, no-tape flags, reader, punch. The first device flag found to be set indicates the device which must be reinitiated by the program.

Immediately after selecting the proper device, a routine is entered which will restore the contents of the link and AC at the time of the program interrupt. The routine is tagged RTNIT. RTNIT first restores the AC (from SAVAC); restores the link (by testing bit 0 of RJMP); enables program interrupt; and then returns to the instruction test by a JMP indirect on the contents of RJMP.

The operator may disable program interrupts by restarting from location 26 with ACS 1 up.

5.3 Program and Operator Action

See Sections 4.2 and 5.2.

6. ERRORS

6.1 Error Halts and Description

Reference the program listing for all error halts.

All error halts are tagged EXXX, and are commented to aid debugging. Each test is self-contained, and may be looped. See Section 6.2.1 for looping instructions.

Unless a solution is obvious from following the listing, the proper MAINDEC diagnostic for the device in error should be run. This should be necessary mainly when errors are caused by one of the I/O devices. The diagnostics for the I/O devices are listed below:

<u>Device</u>	<u>Program</u>
Real-Time Clock	Instruction Test Part 1 2
Program Interrupt	Instruction Test Part 1 2
Punch	Punch Test (9A-D2DB)
Reader	Reader Test (9A-D2CB)
Teletype	TTY Test (9A-D2BB)

Incorrect operation of the real-time clock will appear as clock interrupts occurring sooner than 2 seconds apart, or greater than 9 seconds, or possibly no clock interrupts will occur. Also, after a clock interrupt, the clock should not increment further for any longer than approximately 1/2 second.

Printing of incorrect data may be caused by the data being incorrectly punched, read, or printed. Storage registers, and their locations in the program, which the punch, read, print sequence use are listed below.

<u>Tag</u>	<u>Function</u>
SAVAC (7642)	Saves contents of AC after a program interrupt.
RJMP (7643)	Saves contents of PC and link after a program interrupt.
WORK (7630)	Bit 1 if set indicates TTY is in use.
GOPNCH (6763)	Contains contents of PC at exit from punch routine.
SETCLK (6604)	Routine which sets a random value in clock register 7 when program interrupt is disabled.
CLKSET (6621)	Same function as SETCLK, but is used only after a clock interrupt.
TTOUT (7325)	Location pointer for TTBUFB when printing.
TTIN (7326)	Location pointer for TTBUFA when reading.

<u>Tag</u>	<u>Function</u>
TTBUFA (7360 to 7443)	Storage buffer for characters read.
TTBUFB (7444 to 7530)	Storage for characters to be printed. Contents should equal TTBUFA.
STORE (7335)	Contains character punched.
SETTY (7037)	Routine which is entered after 52 characters have been punched and read. Sets up TTBUFB before printing.
GENRAN (6102)	Random number generator used when PI is disabled.
RANGEN (6133)	Same as GENRAN, but used only after an interrupt.

When data is incorrectly printed, stop the program during print-out. This will enable both TTBUFA and TTBUFB to remain unchanged. TTBUFA will be changed as soon as reading begins.

The data punched is in ASCII mode, and one printed line is indicated on the paper tape by 8 blank frames separating each line. The punched data starts with character A (301) and ends with a line feed (212). A space (240) is punched between all alphabet and number characters.

The line of punched characters in the reader is the line currently being printed. The operator may inspect the tape for an incorrect character punched. If it appears correctly on the tape, it may have been read or printed incorrectly. The characters read are stored in a 52-word buffer beginning at location 7360 (tagged TTBUFA). The characters being printed are stored in a 52-word buffer beginning at location 7444 (tagged TTBUFB). If the program was stopped during printing, these two buffers should contain exactly the same information. The first character read or printed is stored in the first location of either buffer. One character is stored per location. If the data was read incorrectly, the contents of TTBUFA will not equal the last 52 characters on the tape. If the data on tape, and in TTBUFA and TTBUFB are equal, the teleprinter may be at fault.

6.2 Error Recovery

Press CONTINUE to receive further error halts or to continue testing, as indicated by the listing.

Recovery from error halts in the Add Random Pairs test is accomplished by pressing CONTINUE one or more times, depending on the type of error encountered. Pressing CONTINUE after a halt due to an incorrect sum will result in a second halt. The AC will equal the incorrect sum at the first halt, and the sum used for comparison at the second halt. If the error halt is the result of a LINK error, the next test in sequence will be executed.

Recovery from memory checkerboard errors is accomplished by pressing CONTINUE four times. The memory test is restarted after each error halt.

The contents of the AC after each halt will equal the information below.

<u>C (PC)</u>	<u>C (AC)</u>
6351	The address at which the memory error occurred.
6353	What the location should equal (000000 or 777777).
6355	The data as read.
6362	The pattern control word used.

Bit suppression may be accomplished by placing the corresponding ACS up after the halt at location 6354. Place all other ACS down, press CONTINUE, place all ACS down again, and then restore ACS 0 through 5 if needed. The bits selected will not be tested again during the memory test. The selected bits must be reselected after each error halt.

The memory checkerboard test may be continuously looped by restarting from location 26 with ACS 3 up. Program relocation will not take place.

6.2.1 Looping on Individual Tests - Looping on individual tests, except for the interrupt routines, Add Random Pairs test and Memory test, is accomplished by placing a JMP instruction in the first location of the next test in sequence. The address in the JMP instruction should equal the first location of the test to be looped. Restart the program at location 26, if program interrupt is to be enabled. Restart at the first location of the test to be looped, if interrupts are not wanted.

The complete series of tests for any one instruction may be looped by placing a NOP in the location which contains ISZ WORK3. This instruction appears at the end of each series of tests for each instruction. Restart at location 26, or at the beginning of the test to be looped.

6.2.2 Looping on Add Random Pairs - The complete series of tests may be looped by restarting from location 26 with ACS 2 up.

The individual tests may be looped by changing the LAW instruction, appearing after each test, to a JMP. For example, to loop on $(A-B) + B = A$ (tagged AMBPBT), change the LAW AMBPBT instruction to JMP AMBPBT. Restart from location 26, or AMBPBT.

6.2.3 Looping on Memory Checkerboard - Place ACS 3 up, and restart from location 26.

6.2.4 Error Print-outs - The program continually tests for reader or punch no tape indicators being set. When either indicator is set the message "R NO TAPE", or "P NO TAPE" will be printed. The program continues on in sequence after either print-out.

7. OPERATING RESTRICTIONS

All MAINDEC diagnostics which apply to a basic PDP-9 configuration should be run before attempting to run this program.

8. MISCELLANEOUS

8.1 Execution Time

Approximately 1-1/2 minutes are required to execute all tests for one 4K field.

9. PROGRAM DESCRIPTION

The Basic Exerciser performs tests on all operate and memory reference instructions plus core memory.

During normal operation (all ACS down) the program exercises the real-time clock, punch, reader, Teletype and program interrupt while the instruction test portion is running. The MB indicators will change according to the portion of the instruction test currently being executed. When a clock interrupt occurs the MB bits 12 to 17 will increment for 1/2 second (due to clock counts), after which the MB indicators will return to their original state upon re-entering the instruction test.

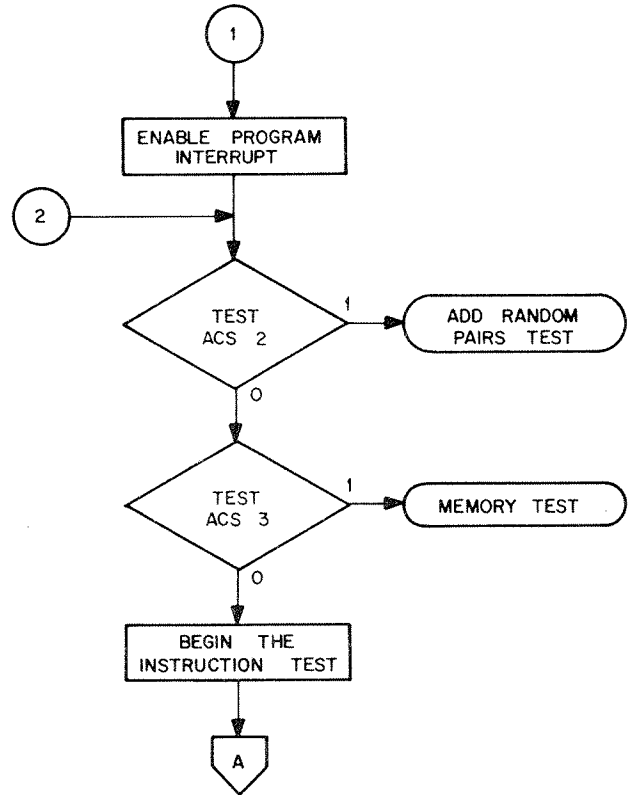
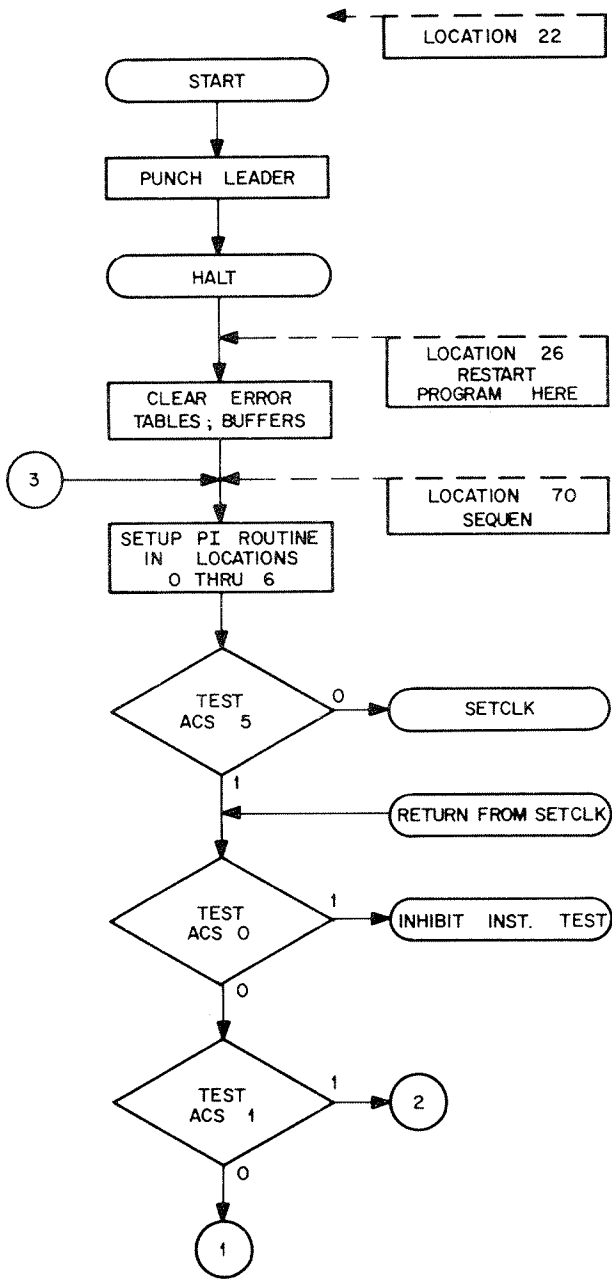
When the punch, read and print sequence is operated with the instruction test inhibited (ACS 0 up) the MB will indicate a constant 600132 (JMP 132). On the listing this is written as JMP at location 132. At location 131 the program interrupt is enabled, and all device interrupts will cause an interrupt immediately after the execution of the JMP instruction. The interrupts are handled in the same manner as if the instruction test portion were operating; the difference being that the interrupt service routine (RTNIT) always returns to location 132 after reinitiating the device which caused the interrupt, instead of returning to the instruction test portion.

The instruction test portion of the program consists of MAINDEC 9A-D01A and D02A (Instruction Test, Parts 1 and 2) condensed onto one tape. Refer to the program listing or flow chart for the testing sequence.

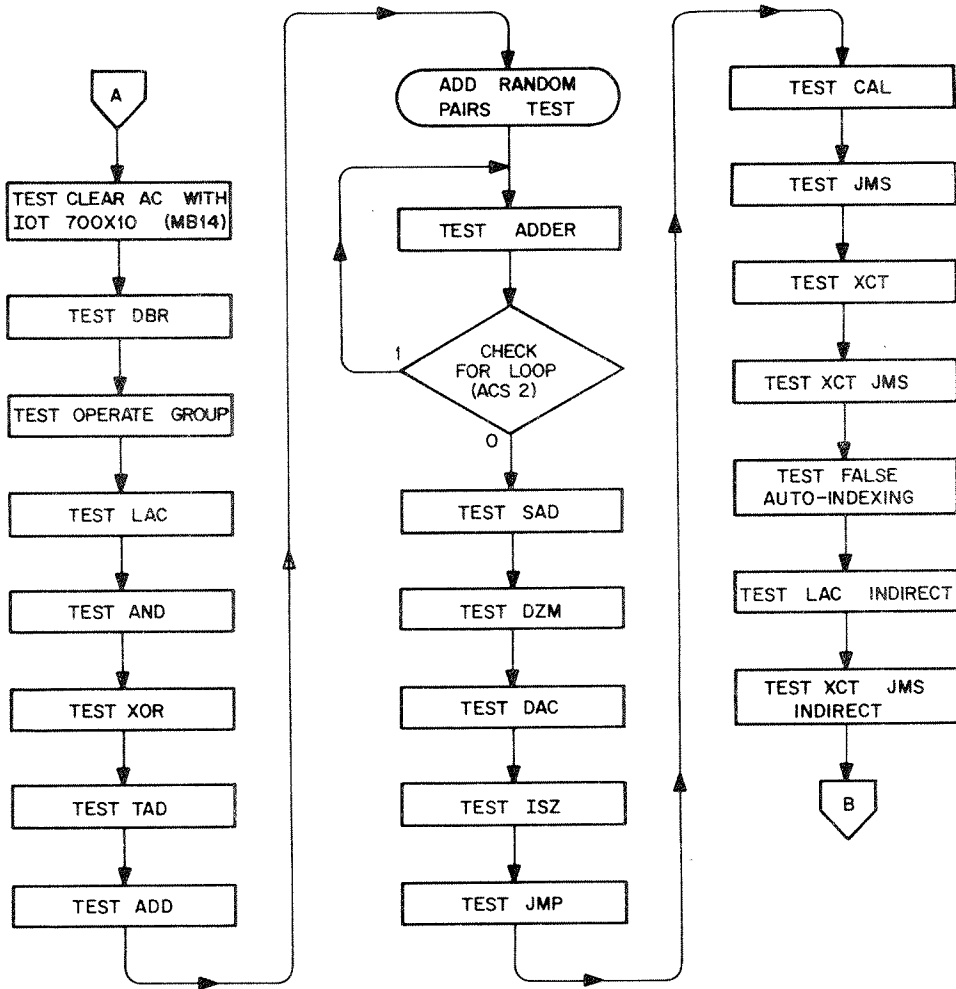
The memory test is executed after the instruction test portion is completed. This test is similar to MAINDEC 9A-D1AB Basic Memory Checkerboard Test. This test is performed three times, after which a routine is entered which relocates the Basic Exerciser to the opposite 4K field in core memory (unless ACS 4 is up). The Exerciser is automatically restarted after relocation is completed.

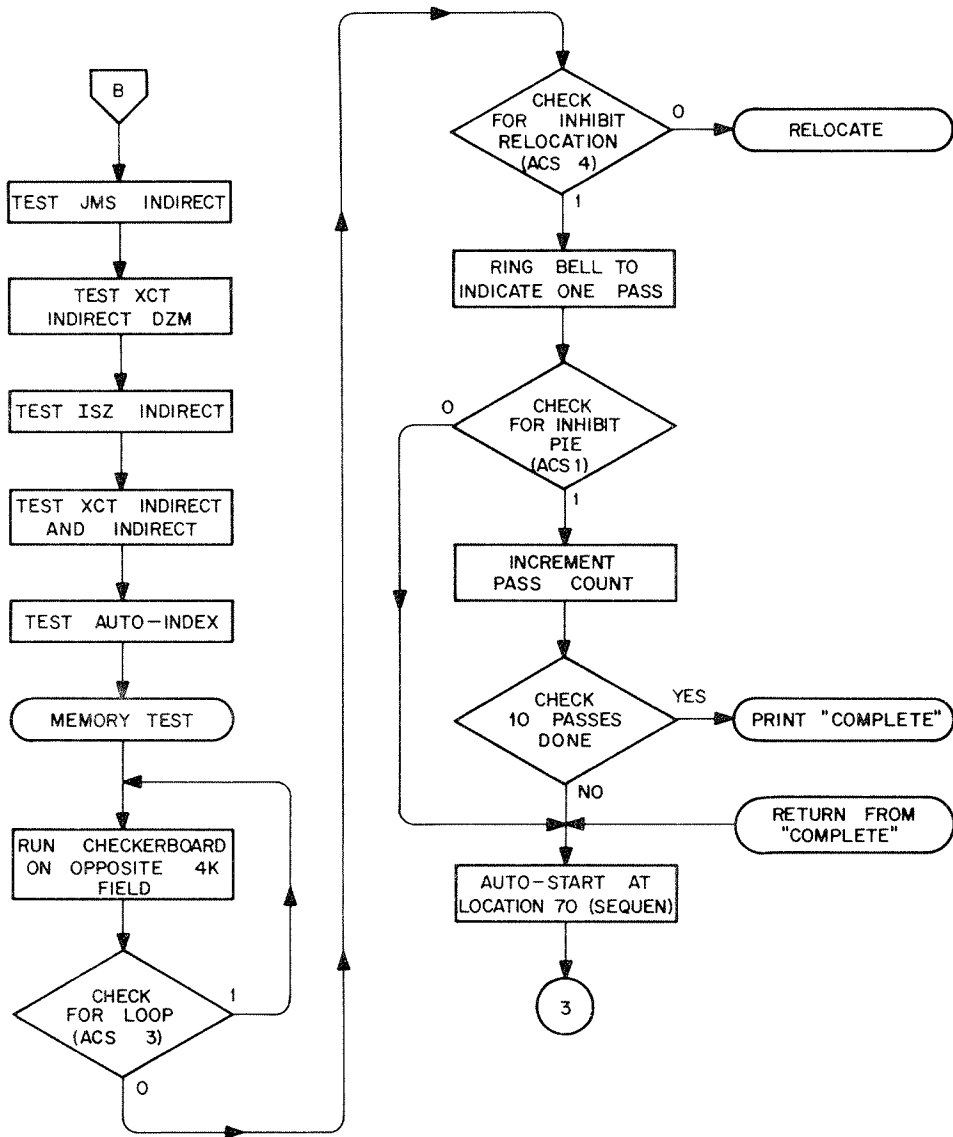
If no errors occur, the Basic Exerciser will run until stopped by the operator.

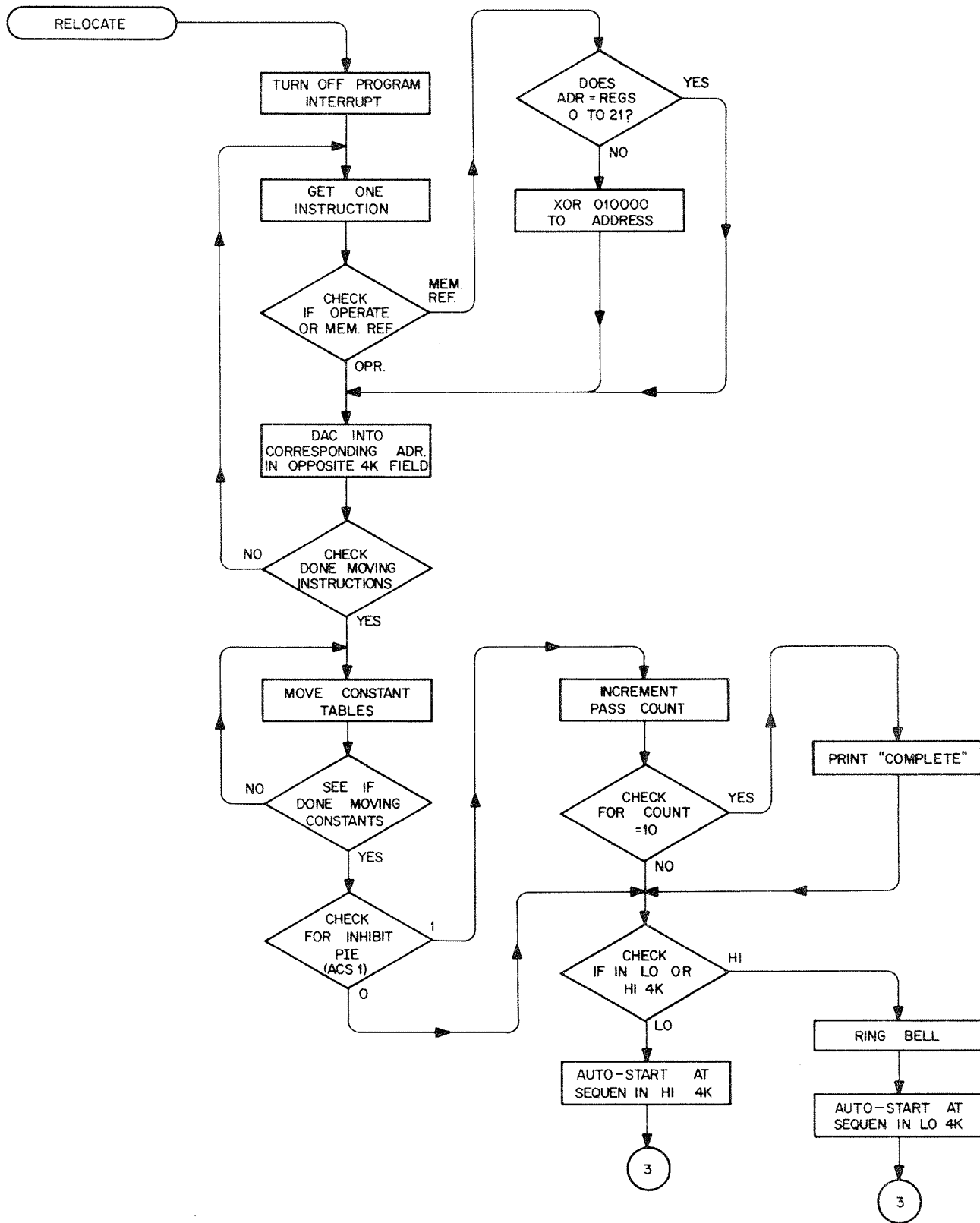
10. FLOW CHARTS



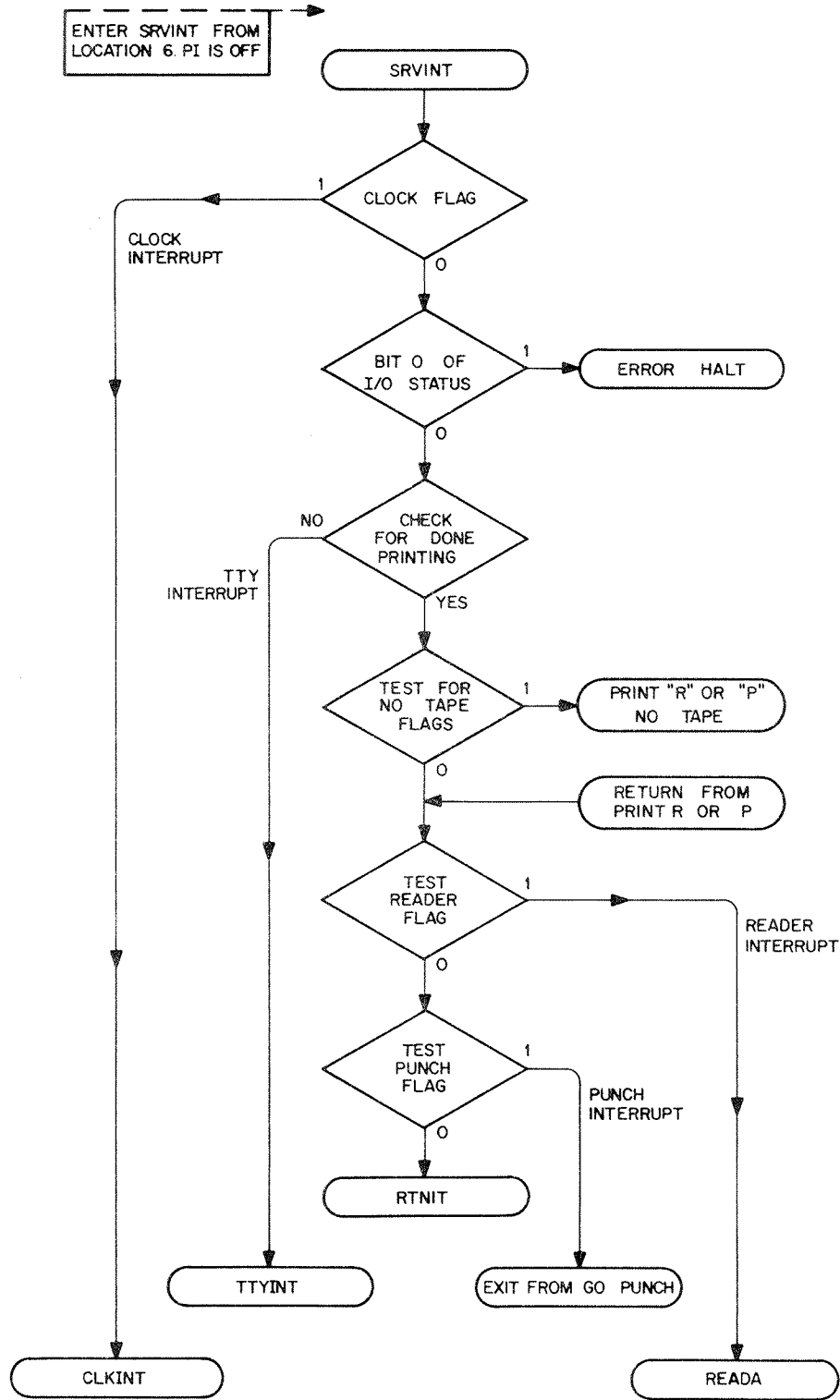
Generalized Flow of PDP-9 Basic Exerciser



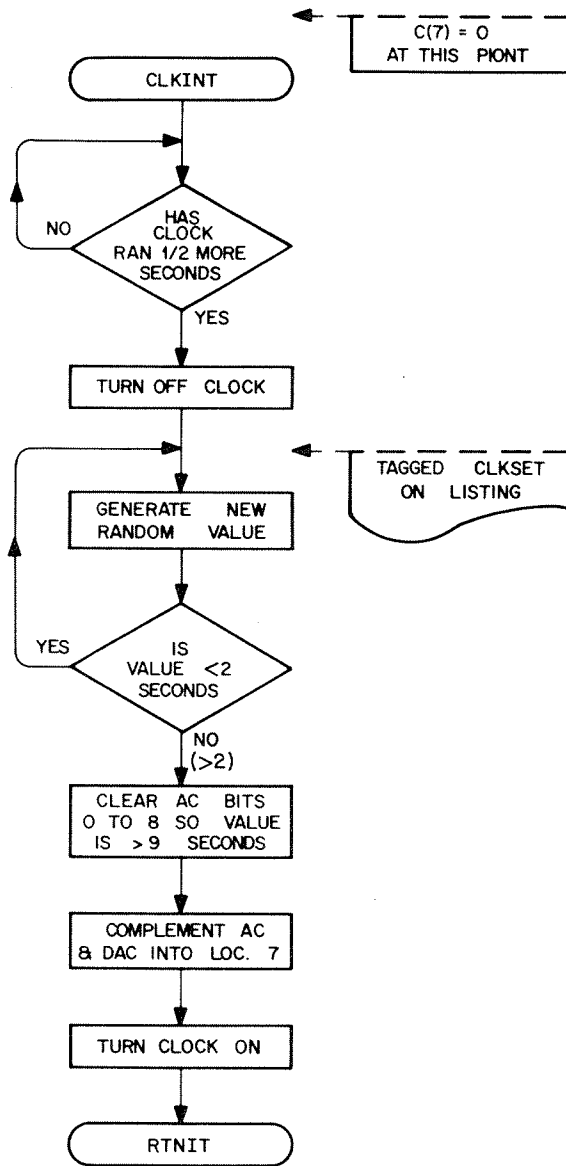




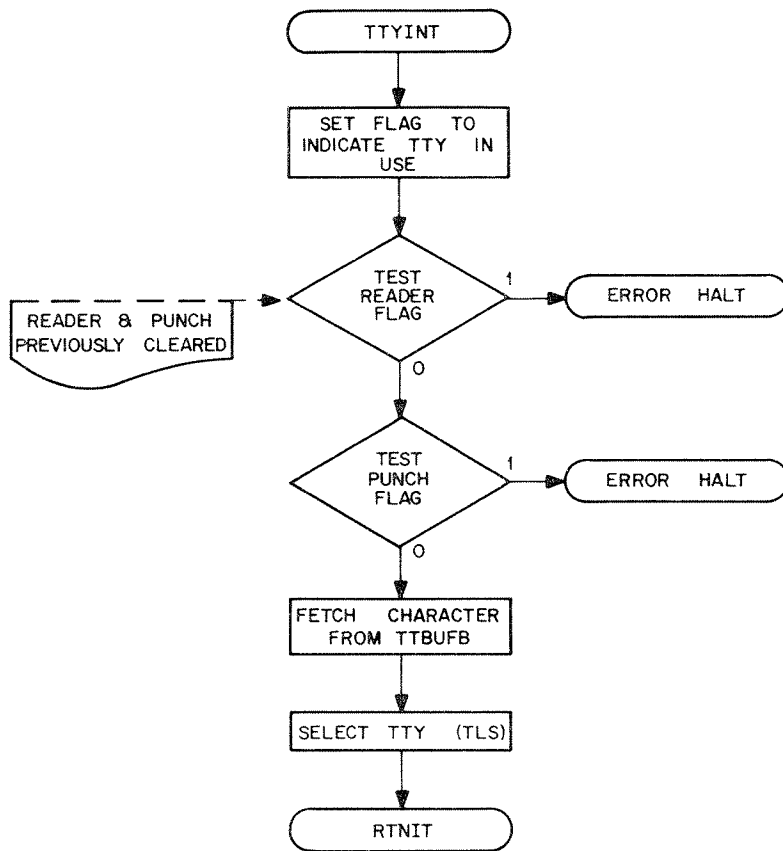
Program Relocation Routine



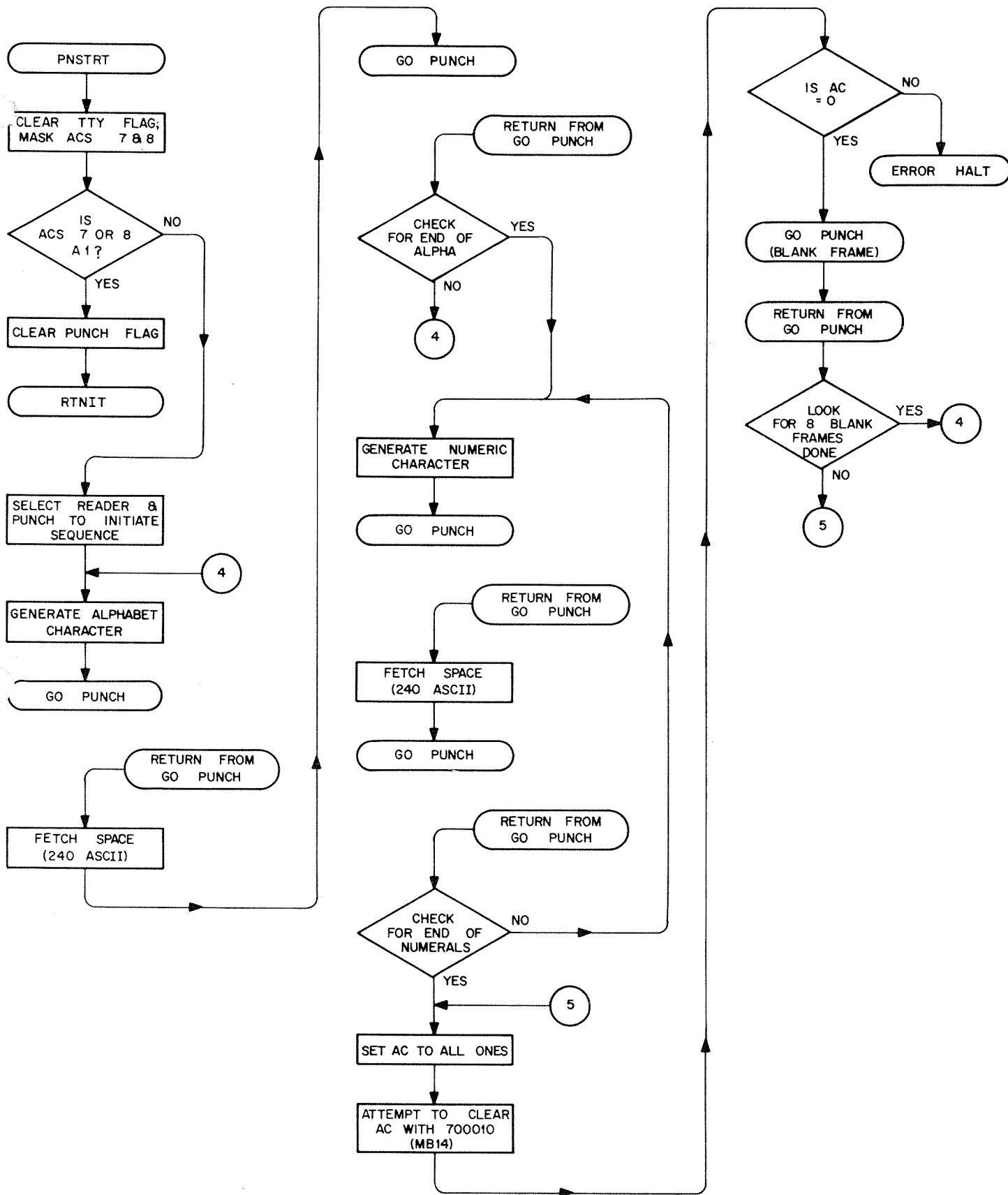
Interrupt Service Routine



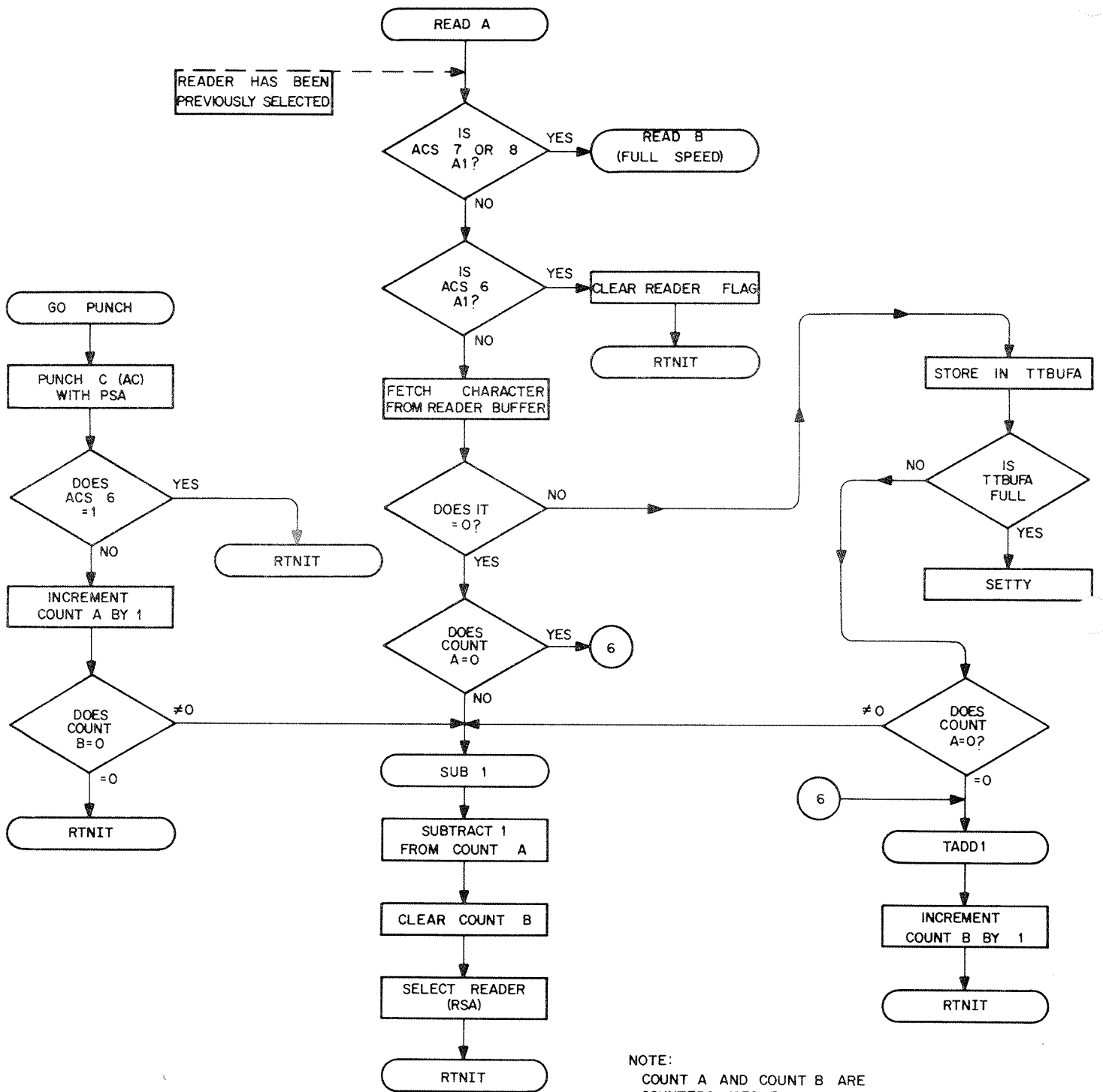
Service Clock Interrupt



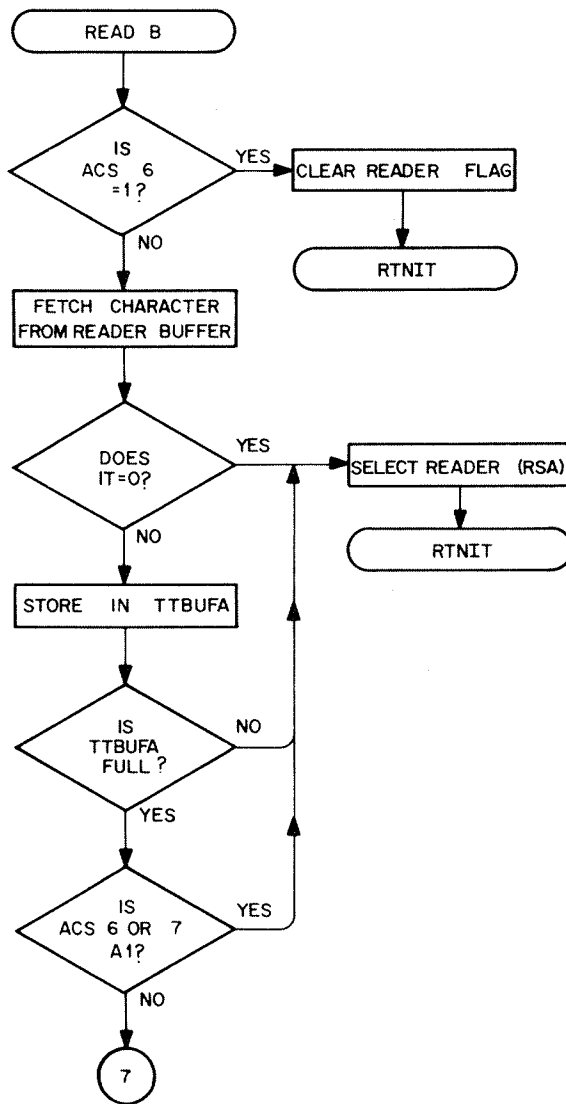
Service TTY Interrupt



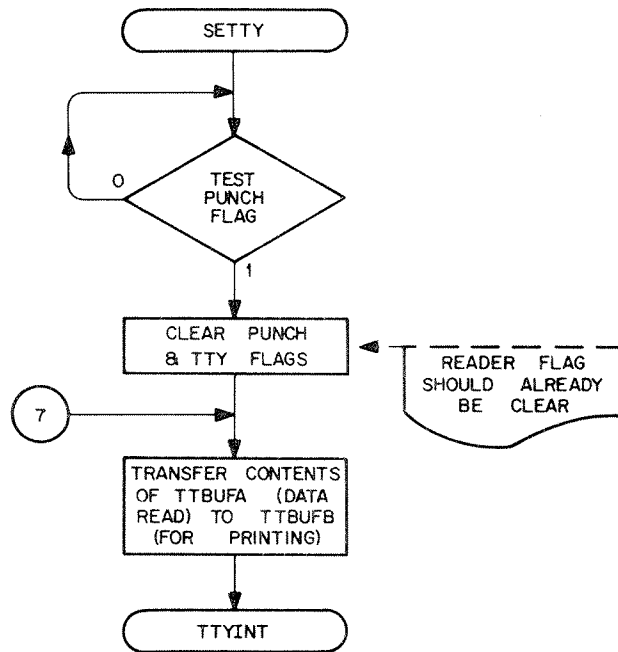
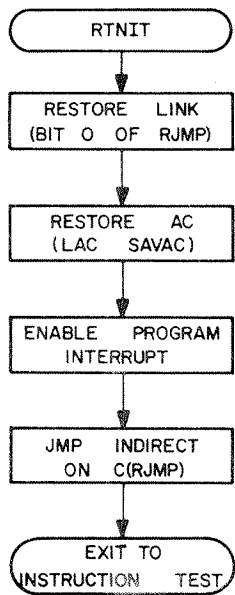
Punch Routine



NOTE:
COUNT A AND COUNT B ARE
COUNTERS USED BY THE PROGRAM
TO ENABLE THE READER TO
OPERATE AT PUNCH SPEED.



Read full speed ACS 7 A 1



11. LISTING

MAINDEC-9A-D7AD-D

```

/NOT DEFINITIONS TO KEEP MACRO-9 HAPPY
/
/
700001 CLSF=700001
700004 CLCF=700004
700044 CLON=700044
/
700101 RSF=700101
700102 RCF=700102
700112 RRR=700112
700104 RSA=700104
700144 RSH=700144
/
700201 FSF=700201
700202 FCF=700202
700204 FSA=700204
700244 FSH=700244
/
700301 KSF=700301
700312 KRR=700312
/
700401 TSF=700401
700402 TCF=700402
700406 TLS=700406
/
.TITLE BASEX9
/POP-9 BASIC EXERCISER
/
.ARS
.LOC 0
/
/POP-9 INSTRUCTION TEST (CONDENSED)
/
00000 000000
00001 741000
00002 740040
F1
00003 047642
00004 200000
00005 047643
00006 606536
/
740000 NOP1=NOP
740000 NOP2=NOP
740000 NOP3=NOP
740040 HALT=HLT
/
00022 .LOC 22
/
00022 107170 REGIN JMS PNI EDR
00023 107202 JMS PNMARK
00024 107170 JMS PNI EDR
00025 740040 HALT
00026 147634 DZM WORK4
00027 147311 DZM BRFAK
/ERROR, PC INCREMENTED BY
/2 AT TIME OF P. I.
/SAVE AC
/SAVE PC AND LINK
/SERVICE INTERRUPT

```

00030 703302
 00031 147630
 00032 146763
 00033 206676
 00034 047326
 00035 047325
 00036 447325
 00037 167325
 00040 207325
 00041 546677
 00042 741000
 00043 600036
 00044 206676
 00045 047325
 00046 206124
 00047 047312
 00050 777737
 00051 047241
 00052 447312
 00053 167312
 00054 447241
 00055 600052

CAF
 DZM WORK
 DZM GORPCH
 LAC DATARL
 DAC TTIN
 DAC TTOUT
 ISZ TTOUT
 DZM* TTOUT
 LAC TTOUT
 SAG ENDRIN
 SKP
 JMP .-5
 LAC DATABL
 DAC TTOUT
 LAC ENDRBL
 DAC WDCNT
 LAW -41
 DAC CRLF
 ISZ WDCNT
 DZM* WDCNT
 ISZ CRLF
 JMP .-3
 .EJECT

/PRESS CONTINUE TO
 /CLEAR GORPCH

/SETUP POINTERS

/CLEAR TTY RIN

/RESTORE POINTER

/CLEAR ERROR TABLE

00056	700002	IOF	/PI OFF
00057	750000	CL	
00060	700104	RS	/INITIALIZE READER, PUNCH
00061	750004	LAS	
00062	507546	AND K3Z	
00063	741200	SNA	/IF ACS 7 OR 8 A 1, DON'T PUNCH
00064	700204	PSA	
00065	200130	LAC INITPI	/INITPI = PNSTR
00066	046763	DAC GOPNCH	
00067	047643	DAC RJMP	/(RJMP) = PNSTR
00070	207616	LAC KSKP	/RESTORE ADDRESS 1 (SKP)
00071	040001	DAC 1	
00072	206533	LAC SAV3	/RESTORE ADDRESS 3
00073	040003	DAC 3	
00074	207651	LAC SAV4	/RESTORE ADDRESS 4
00075	040004	DAC 4	
00076	206534	LAC SAV5	/RESTORE ADDRESS 5
00077	040005	DAC 5	
00100	206535	LAC SAV6	/RESTORE ADDRESS 6
00101	040006	DAC 6	
00102	207647	LAC KHALT	/RESTORE ADDRESS 2 (HALT)
00103	040002	DAC 2	
00104	447634	ISZ WORK4	/PASS COUNTER
00105	750004	LAS	
00106	507555	AND K11K	
00107	741200	SNA	/CHECK ACS 5 FOR INHIBIT CLOCK
00110	106604	JMS SETCLK	
00111	750004	LAS	
00112	741100	SPA	
00113	600131	JMP INHIT	/INHIBIT INST. TEST
00114	740010	RAI	
00115	740100	SMA	/CHECK FOR INHIBIT PI
00116	700042	ION	/PI ON
00117	750004	LAS	
00120	507320	AND K100K	
00121	740200	SZA	/CHECK LOOP ON RANDOM ADD
00122	602360	JMP RANADD-2	/LOOP
00123	750004	LAS	
00124	507552	AND K41K	
00125	740200	SZA	/CHECK LOOP ON CHECKERBOARD
00126	606055	JMP E641+3	/LOOP
00127	600133	JMP IOTST-2	
00130	006702	/	
		INITPI PNSTR	
		/	
		/INHIBIT INSTRUCTION TEST	
00131	700042	INHIT ION	
00132	600132	JMP .	/WAIT FOR PI
		.EJECT	

```

/TEST CLEAR AC AT EVENT TIME 1 WITH MB 14.
/
00133 106102 JMS GENRAN /GET NO. FOR LOOP
00134 106126 JMS CKNO
00135 750001 IOTST CLA!CMA /AC = 777777
00136 700110 SZB
00137 740200 F24 HALT /AC = 0
00140 740040 /ERROR. AC NOT 0
/
00141 750001 CLA!CMA /AC = 777777
00142 700210 F25 HALT /AC = 0
00143 740200 /ERROR. AC NOT 0
00144 740040 /
00145 750001 CLA!CMA /AC = 777777
00146 700310 F26 HALT /AC = 0
00147 740200 /ERROR. AC NOT 0
00150 740040 /
00151 750001 CLA!CMA /AC = 777777
00152 700010 F27 HALT /AC = 0
00153 740200 /ERROR. AC NOT 0
00154 740040 /
00155 447633 ISZ WORK3 /CHECK DONE LOOPING
00156 600135 JMP IOTST /LOOP
00157 106102 JMS GENRAN /GET NO. FOR NEXT TEST
00160 106126 JMS CKNO
.EJECT

```

		/TEST IOT 3344 (DBR), L = 0	
		/	
00161	744000	TSBR: CLI	/LINK = 0
00162	100204	JMS DBRX	
00163	741400	SZI	
00164	740040	E28: HALT	/ERROR. DBR FAILED; LINK NOT 0
		/	
		/TEST IOT 3344 (DBR), < = 1	
		/	
00165	744002	CLI!CMI	/L = 1
00166	100204	JMS DBRX	
00167	740400	SNI	
00170	740040	E29: HALT	/ERROR. DBR FAILED. LINK NOT 1
		/	
		/TEST IOT 3344 (DBR), L = 0	
		/	
00171	754000	CLI!CLA	/AC, L = 0
00172	100207	JMS DBRXX	
00173	740400	SNL	
00174	740040	E30: HALT	/ERROR. DBR FAILED, LINK NOT 1
		/	
		/TEST IOT 3344 (DBR), L = 1	
		/	
00175	754002	CLI!CMI!CLA	/L = 1, AC = 0
00176	100215	JMS DBRXXX	
00177	751400	CLA!SZI	
00200	740040	E31: HALT	/ERROR. DBR FAILED, LINK NOT 0
00201	447633	IS7 WORK3	/CHECK DONE LOOPING
00202	600161	JMP TSBR	/LOOP
00203	600223	JMP OPRAT	/START INSTRUCTION TEST
		/	
00204	000000	DBRX 0	/LEAVE LINK ALONE
00205	703344	703344	/DBR
00206	620204	JMP* DBRX	
		/	
00207	000000	DBRXX 0	
00210	200207	LAC DBRXX	
00211	347553	TAD K400K	/SET LINK TO A ONE
00212	040207	DAC DBRXX	
00213	703344	703344	/DBR
00214	620207	JMP* DBRXX	
		/	
00215	000000	DBRXXX 0	
00216	200215	LAC DBRXXX	
00217	507626	AND M400K	/CLEAR LINK
00220	040215	DAC DBRXXX	
00221	703344	703344	/DBR
00222	620215	JMP* DBRXXX	
		.EJECT	

		/TEST OPERATE GROUP	
		/	
00223	106100	OPRAT JMS GERAN	/GET NO. FOR LOOP ON TEST
00224	106126	JMS CK00	
00225	777777	OPERAT LAW 17777	/AC = 777777
		/	
00226	741000	SKP	/TEST SKP
00227	740040	E32 HALT	/ERROR; SKP FAILED TO SKIP
		/	
		/TEST CLA - SZA	
		CLA	/AC = 0
00230	750000	SZA	
00231	740200	HALT	/ERROR; CLA OR SZA FAILED TO SKIP
00232	740040	E33	
		/	
		/TEST SMA	
		CLA	/AC = 0
00233	750000	SMA	
00234	740100	SKP	
00235	741000	HALT	/ERROR; SAME SKIPPED
00236	740040	E34	
		/	
		/TEST SPA	
		CLA	
00237	750000	SPA	
00240	741100	HALT	/ERROR; SPA FAILED TO SKIP
00241	740040	E35	
		/	
		/TEST SNA	
		CLA	
00242	750000	SNA	
00243	741200	SKP	
00244	741000	HALT	/ERROR; SNA SKIPPED
00245	740040	E36	
		/	
		/TEST SZL - CLL	
		CLL	/LINK = 0
00246	744000	SZL	
00247	741400	HALT	/ERROR; SZL FAILED TO SKIP OR
00250	740040	E37	/CLL FAILED TO CLEAR LINK
		/	
		/TEST SNL	
		CLL	/LINK = 0
00251	744000	SNL	
00252	740400	SKP	
00253	741000	HALT	/ERROR; SNL SKIPPED
00254	740040	E38	
		.EJECT	

00255	754000	/TEST CLA CLI	
00256	740200	CLA:CLI	/AC, LINK = 0
00257	740040	SZL	
		F39 HALT	/ERROR; AC NOT 0
		/	
00260	754000	/TEST CLA CLL	
00261	741400	CLA:CLI	/AC AND LINK = 0
00262	740040	SZI	
		F40 HALT	/ERROR; LINK NOT 0
		/	
00263	750000	/TEST SKP SPA	
00264	741100	CLA	
00265	740040	SKP:SPA	
		F41 HALT	/ERROR; SKP:SPA FAILED TO SKIP
		/	
00266	750000	/TEST SKP SNA	
00267	741200	CLA	
00270	741000	SKP:SNA	
00271	740040	SKP	
		E42 HALT	/ERROR; SKP:SNA SKIPPED
		/	
00272	744000	/TEST SKP SZL	
00273	741400	CLL	/LINK = 1
00274	740040	SKP:SZI	
		F43 HALT	/ERROR; SKP:SZL FAILED TO SKIP
		/	
00275	750000	/TEST SPA SNA	
00276	741300	CLA	
00277	741000	SPA:SNA	
00300	740040	SKP	
		F44 HALT	/ERROR; SPA:SNA SKIPPED
		/	
00301	754000	/TEST SPA SZL	
00302	741500	CLA:CLI	/LINK AND AC = 0
00303	740040	SPA:SZL	
		E45 HALT	/ERROR; SPA:SZL FAILED TO SKIP
		/	
00304	754000	/TEST SNA SZL	
00305	741600	CLA:CLI	/LINK AND AC = 0
00306	741000	SNA:SZI	
00307	740040	SKP	
		E46 HALT	/ERROR; SNA:SZL SKIPPED
		/	
00310	754000	/TEST SNA, SPA, SKP, SZL	
00311	741700	CLA:CLI	/AC AND LINK = 0
00312	741000	SKP:SPA:SZL:SNA	
00313	740040	SKP	
		F47 HALT	/ERROR; SNA:SPA:SKP:SZL SKIPPED
		.EJECT	

00314	750000	/TEST SMA SZA	
00315	740300	CLA	
00316	740040	SMA:SZA	
		F48 HALT	/ERROR; SMA SZA FAILED TO SKIP
		/	
		/TEST SMA SNL	
00317	754000	CLA:CLI	/LINK AND AC = 0
00320	740500	SMA:SNL	
00321	741000	SKP	
00322	740040	F49 HALT	/ERROR; SMA:SNL SKIPPED
		/	
		/TEST SZA SNL	
00323	754000	CLA:CLI	
00324	740600	SZA:SNL	
00325	740040	F50 HALT	/ERROR; SZA:SNL SKIPPED
		/	
		/TEST SMA SZA SNL	
00326	754000	CLA:CLI	
00327	740700	SMA:SZA:SNL	
00330	740040	F51 HALT	/ERROR; SMA:SZA:SNL FAILED TO SKIP
		/	
		/TEST CML - SZL	
00331	744000	CLL	/LINK = 0
00332	740000	CML	/LINK = 1
00333	741400	SZL	
00334	741000	SKP	
00335	740040	F52 HALT	/ERROR; SZL SKIPPED OR /CML FAILED TO SET LINK
		/	
		/TEST CLL	
00336	744000	CLL	/LINK = 0
00337	740000	CML	/LINK = 1
00340	744000	CLL	/LINK = 0
00341	741400	SZL	
00342	740040	F53 HALT	/ERROR; CLL FAILED TO CLEAR LINK.
		/	
		/TEST CML	
00343	744000	CLL	/LINK = 0
00344	740000	CML	/LINK = 1
00345	740000	CML	/LINK = 0
00346	741400	SZL	
00347	740040	F54 HALT	/ERROR; CML FAILED TO SET LINK
		/	
		/TEST CLL CML	
00350	744000	CLL	/LINK = 1
00351	740000	CML	/LINK = 1
00352	744000	CLI:CML	/LINK = 1
00353	741400	SZL	
00354	741000	SKP	
00355	740040	F55 HALT	/ERROR; CLL:CML FAILED TO SET LINK
		.EJECT	


```

/TEST CLL CMI
00356 744000 /LINK = 0
00357 740000 /LINK = 1
00360 744000 /LINK = 0
00361 744000 /LINK = 1
00362 741400
00363 741000
00364 740040 F56 HALT /ERROR; CLL!CML FAILED TO SET LINK
/
/TEST SKP SZL
00365 744000 /LINK = 0
00366 741400
00367 740040 F57 HALT /ERROR; SKP!SZL FAILED TO SKIP
/
/TEST SZL SNA
00370 750000 /AC = 0
00371 744000 /LINK = 1
00372 741600
00373 741000
00374 740040 F58 HALT /ERROR; SZL!SNA SKIPPED
/
/TEST SZL SPA
00375 750000 /AC = 0
00376 744000 /LINK = 1
00377 741500
00400 741000
00401 740040 E59 HALT /ERROR; SZL!SPA SKIPPED
/
/TEST CLA CLL CML
00402 754000 /AC = 0, LINK = 1
00403 741400
00404 741000
00405 740040 F60 HALT /ERROR; LINK NOT 1
/
/TEST CLA CLL CMI
00406 754000 /AC = 0, LINK = 1
00407 740200
00410 740040 F61 HALT /ERROR; AC NOT 0
/
/TEST SNL SZA
00411 754000 /AC = 0, LINK = 1
00412 740600
00413 740040 F62 HALT /ERROR; SNL!SZA FAILED TO SKIP
/
/TEST SNL SMA
00414 754000 /AC = 0, LINK = 1
00415 740500
00416 740040 F63 HALT
.EJECT

```

00417	754000	/TFST SNL SZA SMA	
00420	740700	CLA!CLI!CML	/AC = 0, LINK = 1
00421	740040	SNI!SZA!SMA	
		F64 HALT	/ERROR: SNL!SZA!SMA FAILED TO SKIP
		/	
00422	750000	/TEST CMA CLA	
00423	740001	CLA	/AC = 0
00424	750000	CMA	/AC = ONES
00425	741200	CLA	
00426	741000	SNA	
00427	740040	SKP	
		F65 HALT	/ERROR: CLA FAILED TO CLEAR AC
		/	
00430	750000	/TEST CMA SPA	
00431	740001	CLA	/AC = 0
00432	741100	CMA	/AC = ONES
00433	741000	SPA	
00434	740040	SKP	
		F66 HALT	/ERROR: SPA SKIPPED OR /CMA FAILED TO SET AC BIT
		/	
00435	750000	/TFST CMA SNA	
00436	740001	CLA	/AC = 0
00437	741200	CMA	/AC = ONES
00440	740040	SNA	
		F67 HALT	/ERROR: SNA FAILED TO SKIP /OR CMA FAILED TO SET ANY AC BIT
		/	
00441	750000	/TFST CMA CLA	
00442	740001	CMA	/AC = 0
00443	740001	CMA	/AC = ONES
00444	741200	SNA	/AC = 0
00445	741000	SKP	
00446	740040	F68 HALT	/ERROR: CMA FAILED TO /COMPLEMENT AC TO 0
		/	
00447	750001	/TEST CLA CMA	
00450	741200	CLA!CMA	/AC = ONES
00451	740040	SNA	
		E69 HALT	/ERROR: CLA!CMA FAILED TO /SET ANY AC BIT
		.EJECT	

```

00452 750001 /TEST SZA CLA:CMA /AC = ONFS
00453 740000 SZA
00454 741000 SKP
00455 740040 E70 HALT /ERROR; SZA SKIPPED
/
00456 750001 /TEST SMA CLA:CMA /AC = ONFS
00457 740100 SMA
00460 740040 F71 HALT /ERROR; SMA FAILED TO SKIP
/
00461 750001 /TEST SKP SPA /AC = ONFS
00462 741100 CLA:CMA
00463 741000 SKP!SPA
00464 740040 SKP
E72 HALT /ERROR; SKP!SPA SKIPPED
/
00465 750001 /TEST SKP SNA /AC = ONFS
00466 741200 CLA:CMA
00467 740040 SKP!SNA
E73 HALT /ERROR; SKP!SNA FAILED TO SKIP
/
00470 750001 /TEST SPA SNA /AC = ONFS
00471 741300 CLA:CMA
00472 741000 SPA!SNA
00473 740040 SKP
F74 HALT /ERROR; SPA!SNA SKIPPED
/
00474 754003 /TEST SKP SNA SPA /AC = ONES, LINK = 1
00475 741700 CLA:CMA!CLL!CML
00476 741000 SNA!SPA!SKP!SZL
00477 740040 SKP
E75 HALT /ERROR; SKP!SNA!SPA!SZL SKIPPED
/
00500 750001 /TEST SMA!SZA /AC = ONES
00501 740300 CLA:CMA
00502 740040 SMA!SZA
E76 HALT /ERROR; SMA!SZA FAILED TO SKIP
/
00503 754003 /TEST SMA SZA SNI /AC = ONE, LINK = 1
00504 740700 CLA:CMA!CLL!CML
00505 740040 SMA!SZA!SNL
E77 HALT /ERROR; SMA!SZA!SNL
/
00506 750001 /TEST NOP CLA:CMA /AC = ONFS
00507 740000 NOP
00510 740001 CMA /AC = 0
00511 740200 SZA
00512 740040 F78 HALT /ERROR; NOP ALTERED THE AC
/
.EJECT

```

00513	750000	/TEST NOP	CLA	/AC = 0
00514	740000		NOP	
00515	740200		SZA	
00516	740040	F79	HALT	/ERROR; NOP SET AN AC BIT
		/		
		/TEST NOT	CLL!CML	/LINK = 1
00517	744002		NOP	
00520	740000		SNL	
00521	740400	F80	HALT	/ERROR; NO CLEARED THE LINK
00522	740040	/		
		/TEST NOP	CLL	/LINK = 0
00523	744000		NOP	
00524	740000		SZI	
00525	741400	E81	HALT	/ERROR; NOP SET THE LINK
00526	740040	/		
		/TEST SZA	CMA	/AC = 0
00527	750000		CLA	/AC = ONFS
00530	740201		SZA!CMA	/ERROR; SZA FAILED TO SKIP
00531	740040	F82	HALT	
		/		
		/TEST SZA	CLA	/AC = ONFS
00532	750001		CLA!CMA	/AC = 0
00533	750200		SZA!CLA	
00534	741000		SKP	
00535	740040	E83	HALT	/ERROR; SZA SKIPPED
		/		
		/TEST SZL	CML	/LINK = 0
00536	744000		CLL	
00537	741402		SZI!CML	
00540	740040	F84	HALT	/ERROR; SZL FAILED TO SKIP
		/		
		/TEST SZL	CLL	/LINK = 1
00541	744002		CLL!CML	
00542	745400		SZI!CLL	
00543	741000		SKP	
00544	740040	E85	HALT	/ERROR; SZL SKIPPED
		/		
		/TEST SKP	SZL SPA LA CLL	/AC = ONFS, LINK = 1
00545	754003		CLA!CMA!CLL!CML	/AC = 0, LINK = 0
00546	755000		SKP!SZI!SPA!CLA!CLL	
00547	741000		SKP	
00550	740040	E86	HALT	/ERROR; SKP!SZL!SPA SKIPPED
			.EJECT	

```

00551 754002 /TEST SZA SNI CMA CLL
00552 744601 CLA:CLI:CML /AC = 2, LINK = 1
00553 740040 SZA:SNI:CMA:CLI /AC=ONES, LINK =0
F87 HALT /ERROR, SZA:SNI FAILED TO SKP
/
/TEST CLA SKP
00554 750001 CLA:CMA /AC = ONES
00555 751000 SKP:CLA /AC = 0
00556 740000 NOP
00557 740200 SZA
00560 740040 F88 HALT /ERROR, CLA FAILED TO CLEAR AC
/
/TEST SKP CLA CMA
00561 750000 CLA /AC = 0
00562 751001 SKP:CLA:CMA /AC = ONES
00563 740000 NOP
00564 740001 CMA
00565 740200 SZA
00566 740040 F89 HALT /ERROR, CLA:CMA FAILED TO
/COMPLEMENT THE AC
/
/TEST SKP CLL CML
00567 744000 CLL /LINK = 0
00570 745002 SKP:CLL:CML /LINK = 1
00571 740000 NOP
00572 740400 SNI
00573 740040 F90 HALT /ERROR, CLL:CML FAILED TO SET THE LINK
/
/TEST CMA SERIES
00574 750001 CLA:CMA /AC = ONES
00575 740001 CMA /AC = 0
00576 740001 CMA /AC = ONES
00577 740001 CMA /AC = 0
00600 740001 CMA /AC = ONES
00601 740001 CMA /AC = 0
00602 740200 SZA
00603 740040 F91 HALT /ERROR, AC NOT 0 CMA FAILED
/
/TEST CML SERIES
00604 744002 CLL:CML /LINK = 1
00605 740002 CML /LINK = 0
00606 740002 CML /LINK = 1
00607 740002 CML /LINK = 0
00610 740002 CML /LINK = 1
00611 740002 CML /LINK = 0
00612 741400 SZL
00613 740040 F92 HALT /ERROR, LINK NOT 0 CML FAILED
/
00614 447633 ISZ WORK3 /CHECK DONE LOOPING
00615 600225 JMP OPERAT /LOOP
00616 106102 JMS GERAN /GET NO. FOR NEXT LOOP
00617 106126 JMS CKNO
.EJECT

```

```

/
/TEST RA# SERIES AND LINK
RTAT CLACLE:CMU /AC = 0, LINK = 1
00620 754000
00621 740020 RA#
00622 740020 RA#
00623 740020 RA#
00624 740020 RA#
00625 740020 RA#
00626 740020 RA#
00627 740020 RA#
00630 740020 RA#
00631 740020 RA#
00632 740020 RA#
00633 740020 RA#
00634 740020 RA#
00635 740020 RA#
00636 740020 RA#
00637 740020 RA#
00640 740020 RA#
00641 740020 RA#
00642 740020 RA#
00643 741600 SNA!SER
00644 740040 HALT

E113

/EJECT

/ERROR: AC BIT 17 NOT 1, OR LINK = 1
/AFTR ROTATE SERIES
```

```

/TEST RLI SERIES AND LINK
00645 754000 CLA:CLI:OML /AC = 0, LINK = 1
00646 740010 RAI
00647 740010 RAI
00650 740010 RAI
00651 740010 RAI
00652 740010 RAI
00653 740010 RAI
00654 740010 RAI
00655 740010 RAI
00656 740010 RAI
00657 740010 RAI
00660 740010 RAI
00661 740010 RAI
00662 740010 RAI
00663 740010 RAI
00664 740010 RAI
00665 740010 RAI
00666 740010 RAI
00667 740010 RAI
00670 741600 SNA:SZI
00671 740040 HALT /ERROR; AC BIT 0 NOT 1, OR LINK = 1
/ AFTER ROTATE SERIES
/
/TEST RTI SERIES AND LINK
00672 754000 CLA:CLI:OML /AC = 0, LINK = 1
00673 742010 RTI
00674 742010 RTI
00675 742010 RTI
00676 742010 RTI
00677 742010 RTI
00700 742010 RTI
00701 742010 RTI
00702 742010 RTI
00703 742010 RTI
00704 741600 SNA:SZI
00705 740040 HALT /ERROR; AC BIT 0 NOT 1, OR LINK = 1
/ AFTER ROTATE SERIES
/
/TEST RTR SERIES AND LINK
00706 754000 CLA:CLI:OML /AC = 0, LINK = 1
00707 742020 RTR
00710 742020 RTR
00711 742020 RTR
00712 742020 RTR
00713 742020 RTR
00714 742020 RTR
00715 742020 RTR
00716 742020 RTR
00717 742020 RTR
00720 741600 SNA:SZI
00721 740040 HALT /ERROR; AC BIT 17 NOT 1, OR LINK = 1
/ AFTER ROTATE SERIES
/PDP-9 BASIC EXERCISER - TAPE 2

```

```

/
/RAR SERIES
RTSS CLA:CM:ICLL /AC = ONES, LINK = 0
RAR; RAR; RAR; RAR
00722 754001
00723 740020
00724 740020
00725 740020
00726 740020
00727 740020 RAR; RAR; RAR; RAR
00730 740020
00731 740020
00732 740020
00733 740020 RAR; RAR; RAR; RAR
00734 740020
00735 740020
00736 740020
00737 740020 RAR; RAR; RAR; RAR
00740 740020
00741 740020
00742 740020
00743 740020 RAR; RAR
00744 740020
00745 740003 CMA:CM: /AC = 000001, LINK = 0
00746 741600 SNA:SZL
00747 740040 HALT /ERROR: AC BIT 17 NOT 1, OR LINK = 0
/ /AFTER ROTATE SERIES

```

E140

```

/
/TEST RAL SERIES TEST
CLA:CM:ICLL /AC = ONES, LINK = 0
RAL; RAL; RAL; RAL
00750 754001
00751 740010
00752 740010
00753 740010
00754 740010
00755 740010 RAL; RAL; RAL; RAL
00756 740010
00757 740010
00760 740010
00761 740010 RAL; RAL; RAL; RAL
00762 740010
00763 740010
00764 740010
00765 740010 RAL; RAL; RAL; RAL
00766 740010
00767 740010
00770 740010 RAL; RAL; CML /AC = 377777, LINK = 0
00771 740010
00772 740010
00773 740002
00774 741500 SPA:SZL
00775 740040 HALT /ERROR: AC BIT 0 NOT 0, OR LINK = 0
/ /AFTER ROTATE SERIES
.EJECT

```

E141


```

00776      754001
00777      742010
01000      742010
01001      742010
01002      742010
01003      742010
01004      742010
01005      742010
01006      742010
01007      742010
01010      740002
01011      741500
01012      740040
E142
/TEST RTL SERIES
CLA!CMA!CLL      /AC = ONES, LINK = 0
RTL;      RTL;      RTL;      RTL
RTL;      RTL;      RTL;      RTL
RTL;      CML      /LINK = 0
SPA!SZL
HALT      /ERROR; AC BIT 0 NOT 0, OR LINK = 0
/AFTER ROTATE SERIES

/
01013      754001
01014      742020
01015      742020
01016      742020
01017      742020
01020      742020
01021      742020
01022      742020
01023      742020
01024      742020
01025      740003
01026      741600
01027      740040
E143
/TEST RTR SERIES
CLA!CMA!CLL      /AC = ONES, LINK = 0
RTR;      RTR;      RTR;      RTR
RTR;      RTR;      RTR;      RTR
RTP
CMA!CMI      /AC = 000001, LINK = 0
SNA!SZL
HALT      /ERROR; AC BIT 17 NOT 1, OR LINK = 0
/AFTER ROTATE SERIES

.EJECT

```

01030	754000	/		
01031	741210	/TEST RAL:SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01032	741000		RAL:SNA	
01033	740040	F162	SKP	/ERROR; SNA SKIPPED
			HALT	
		/		
01034	754000	/TEST RAR:SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01035	741220		RAR:SNA	
01036	741000	F163	SKP	/ERROR; SNA SKIPPED
01037	740040		HALT	
		/		
01040	754000	/TEST RTL"SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01041	743210		RTL:SNA	
01042	741000	F164	SKP	/ERROR; SNA SKIPPED
01043	740040		HALT	
		/		
01044	754000	/TEST RTR:SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01045	743220		RTR:SNA	
01046	741000	F165	SKP	/ERROR; SNA SKIPPED
01047	740040		HALT	
		/		
01050	754000	/TEST RAL:SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01051	740020		RAR	/AC = 400000
01052	741210		SNA:RAL	
01053	740040	E166	HALT	/ERROR. SNA FAILED TO SKIP
		/		
01054	754000	/TEST RAR:SNA	CLA:CLI:CML	/AC = 0, LINK = 1
01055	740010		RAL	/AC = 000001
01056	741220		SNA:RAR	
01057	740040	F167	HALT	/ERROR; SNA FAILED TO SKIP
			.EJECT	

01060	754002	/TFST RTL:SNA	
01061	742020	CLA:CLI:CML	/AC = 0, LINK = 1
01062	743210	RTR	/AC = 200000
01063	740040	SNA:RTI	
		F168 HALT	/ERROR; SNA FAILED TO SKIP
		/	
		/TFST RTR:SNA	
01064	754002	CLA:CLI:CML	/AC = 0, LINK = 1
01065	742010	RTI	
01066	743220	SNA:RTI	
01067	740040	F169 HALT	/ERROR; SNA FAILED TO SKIP
		/	
		/TEST CLL:SNA:RAR	
01070	754001	CLA:CMA:CLL	/AC = ONES, LINK = 0
01071	751220	CLA:SNA:RAR	
01072	740040	E170 HALT	/ERROR; SNA FAILED TO SKIP
01073	447633	IS7 WORK3	/CHECK DONE LOOPING
01074	600620	JMP RTAT	/LOOP
01075	106102	JMS GENRAN	/GET NO FOR NEXT LOOP
01076	106126	JMS CKNO	
		/	
		.EJECT	

```

/
/TEST LAW 760000
TLAW          CLA:CLI          /AC = 0
              LAW 000000     /AC = 760000
              RAL            /AC = 740000
              SNL:CLI        /LINK = 1
              HALT           /ERROR; AC NOT 0 NOT A 1,
                              /LAW 760000 FAILED

011077        754000
011100        760000
011101        740010
011102        744400
011103        740040      F206    RAL
                              /AC = 700000
                              /LINK = 1
                              /ERROR; AC BIT 1 NOT A 1
                              /LAW 760000 FAILED

011104        740010
011105        744400
011106        740040      F207    RAL
                              /AC = 700000
                              /LINK = 1
                              /ERROR; AC BIT 1 NOT A 1
                              /LAW 760000 FAILED

011107        740010
011110        744400
011111        740040      F208    RAL
                              /AC = 600000
                              /LINK = 1
                              /ERROR; AC BIT 2 NOT A 1
                              /LAW 760000 FAILED

011112        740010
011113        744400
011114        740040      F209    RAL
                              /AC = 400000
                              /LINK = 1
                              /ERROR; AC BIT 3 NOT A 1
                              /LAW 760000 FAILED

011115        740010
011116        744400
011117        740040      F210    RAL
                              /AC = 000000
                              /LINK = 1
                              /ERROR; AC BIT 4 NOT A 1
                              /LAW 760000 FAILED

011120        740200
011121        740040      E211    SZA
                              /AC = 000000
                              /ERROR; AC BITS 5-17 NOT 0
                              /LAW 760000 FAILED

/
/TEST LAW 760000, AC = ONES
              CLA:CMA:CLL     /AC = ONES, LINK = 0
              LAW 000000     /AC = 760000
              SZL
              HALT           /ERROR; LINK NOT A 0, LAW SET LINK

011122        754001
011123        760000
011124        741400
011125        740040      F212    RAL
                              /AC = 740000
                              /LINK = 1
                              /ERROR; AC BIT 0 NOT A 1,
                              /LAW 760000 FAILED

011126        740010
011127        744400
011130        740040      F213    SNL:CLL
                              /AC = 700000
                              /LINK = 1
                              /ERROR; AC BIT 1 NOT A 1
                              /LAW 760000 FAILED

011131        740010
011132        744400
011133        740040      F214    RAL
                              /AC = 600000
                              /LINK = 1
                              /ERROR; AC BIT 2 NOT A 1
                              /LAW 760000 FAILED

/
011134        740010
011135        744400
011136        740040      F215    RAL
                              /AC = 400000
                              /LINK = 1
                              /ERROR; AC BIT 3 NOT A 1, LAW 760000 FAILED

011137        740010
011140        744400
011141        740040      F216    RAL
                              /AC = 000000
                              /LINK = 1
                              /ERROR; AC BIT 4 NOT A 1, LAW 760000 FAILED

011142        740010
011143        744400
011144        740040      F217    SNL:CLL
                              /AC = 000000
                              /ERROR; AC BITS 5-17 NOT 0
                              /LAW 760000 FAILED

011145        740200
011146        740040      F218    SZA
                              /AC = 000000
                              /ERROR; AC BITS 5-17 NOT 0
                              /LAW 760000 FAILED

```

.EJECT

01147	754000	/TEST LAW 777777. AC=0, L=0	
01150	777777	CLA:CLI	/AC = 0
01151	740001	LAW 17777	/AC = 760200
01152	740200	CMA	
01153	740040	SZA	
		F219 HALT	/ERROR, AC NOT 0
			/LAW 17777 FAILED
01154	741400	SZI	/AC = 400760
01155	740040	F220 HALT	/LINK NOT 0
		/	
		/TEST LAW 777777. AC=0, L=1	
01156	754002	CLA:CLI:CML	/AC = 0
01157	777777	LAW 17777	
01160	740001	CMA	
01161	740200	SZA	
01162	740040	F221 HALT	/ERROR, LINK NOT 0
01163	740400	SNI	
01164	740040	F222 HALT	/ERROR, LINK NOT 0
		/	
		/TEST LAW 777777. AC=1, L=0	
01165	754001	CLA:CMA:CLL	/AC = 0
01166	777777	LAW 17777	
01167	740001	CMA	
01170	740200	SZA	
01171	740040	E223 HALT	/ERROR, AC NOT 0
01172	741400	SZI	/AC = 100760
01173	740040	E224 HALT	/ERROR, LINK NOT 0
		.EJECT	

01174 754003
01175 777777
01176 740001
01177 740200
01200 740040
01201 740400
01202 740040
01203 447633
01204 601077
01205 106102
01206 106126

```
/TFST LAW 777777, AC=1, I=1  
CLA!CMA!CLL!CMI /AC = 0  
LAW 17777  
CMA  
SZA  
F225 HALT /ERROR, AC NOT 0  
SNI  
E226 HALT /ERROR, LINK NOT 1  
IS? WORK3 /CHECK DONE LOOPING  
JMP TLAW /LOOP  
JMS GENRAN /GET NO. FOR NEXT LOOP  
JMS CKND  
.EJECT
```

```

/TEST LAC 0'S
/
01207 754000 LACK CLA:CLI LAC = 0, LINK = 0
01210 207530 LAC K0 /000000
01211 740200 SZA
01212 740040 F258 HALT /ERROR. AC NOT 0 AFTER LAC K0
01213 741400 SZL
01214 740040 F259 HALT /ERROR. LINK NOT 0 AFTER LAC K0
/
01215 754002 CLA:CLI:CML LAC = 0, LINK = 1
01216 207530 LAC K0
01217 740200 SZA
01220 740040 F260 HALT /ERROR. AC NOT 0
01221 740400 SNL
01222 740040 E261 HALT /ERROR. LINK NOT 1 AFTER LAC K0
/
01223 754001 CLA:CMA:CLL LAC = 1'S, LIN = 0
01224 207530 LAC K0
01225 740200 SZA
01226 740040 F262 HALT /ERROR. AC NOT 0 AFTER LAC K0
01227 741400 SZL
01230 740040 F263 HALT /ERROR. LINK NOT 0 AFTER LAC K0
/
01231 754003 CLA:CMA:CLL:CML LAC = 1'S, LINK = 1
01232 207530 LAC K0
01233 740200 SZA
01234 740040 F264 HALT /ERROR. AC NOT 0
01235 740400 SNL
01236 740040 F265 HALT /ERROR. LINK NOT 1
/TEST LAC 1'S
/
01237 754000 CLA:CLL LAC = 0, LINK = 0
01240 207572 LAC K7S /777777
01241 740001 CMA
01242 740200 SZA
01243 740040 F266 HALT /ERROR. AC NOT 0 LAC K7S FAILED
01244 741400 SZL
01245 740040 F267 HALT /ERROR. LINK NOT 0 AFTER LAC K7S
/
01246 754002 CLA:CLI:CML LAC = 0, LINK = 1
01247 207572 LAC K7S
01250 740001 CMA
01251 740200 SZA
01252 740040 F268 HALT /ERROR. AC NOT 0
01253 740400 SNL
01254 740040 F269 HALT /ERROR. LINK NOT 1 AFTER LAC K7S
.EJECT

```


01255	754001		CLA!CMA!CLL	/AC = 1'S, LINK = 0
01256	207572		LAC K7S	
01257	740001		CMA	
01260	740200		SZA	
01261	740040	F270	HALT	/ERROR. AC NOT 0. LAC K7S FAILED
01262	741400		SZI	
01263	740040	E271	HALT	/ERROR. LINK NOT 1 AFTER LAC K7S
		/		
01264	754003		CLA!CMA!CLL!CML	/AC = 1'S, LINK = 1
01265	207572		LAC K7S	
01266	740001		CMA	
01267	740200		SZA	
01270	740040	F272	HALT	/ERROR. AC NOT 0
01271	740400		SNL	
01272	740040		HALT	/ERROR. LINK NOT 1 AFTER LAC K7S
		/		
01273	750000		CLA	
01274	207605		LAC K101	/AC = 525252
01275	207604		LAC K010	/AC = 252525
01276	207572		LAC K7S	/AC = 777777
01277	740001		CMA	
01300	740200		SZA	
01301	740040	F273	HALT	/ERROR. AC NOT 0
01302	447633		ISZ WORK3	/CHECK FOR DONE LOOPING
01303	601207		JMP LACK	/LOOP
01304	106102		JMS GENRAN	/GET NO. FOR LOOP.
01305	106126		JMS CKWD	
		/		
		/TEST AND		
		/		
01306	750000		ANDAC	/AC = 0
01307	507530		AND K0	
01310	740200		SZA	
01311	740040	F274	HALT	/ERROR. AC NOT 0 AFTER AND K0
		/		
01312	750001		CLA!CMA	/AC = 1'S
01313	507530		AND K0	
01314	740200		SZA	
01315	740040	E275	HALT	/ERROR. AC NOT 0 AFTER AND K0
		/		
01316	750000		CLA	/AC = 0
01317	507572		AND K7S	
01320	740200		SZA	
01321	740040	F276	HALT	/ERROR. AC NOT 0 AFTER AND K7S
		/		
01322	750001		CLA!CMA	/AC = 1'S
01323	507572		AND K7S	
01324	740001		CMA	
01325	740200		SZA	
01326	740040	E277	HALT	/ERROR. AC NOT 0 AFTER AND K7S
			.EJECT	

```

/SEQUENTIAL AND
/
01327 754002 CLA!CLL!CML /AC = 0, LINK = 1
01330 507530 AND K0
01331 507572 AND K7S
01332 507605 AND K101
01333 507604 AND K010
01334 740001 CMA
01335 507530 AND K0
01336 507572 AND K7S
01337 507605 AND K101
01340 507604 AND K010
01341 740200 SZA
01342 740040 F278 HALT /ERROR. AC NOT 0
01343 740400 SNL
01344 740040 F279 HALT /ERROR. LINK NOT 1
01345 447633 ISZ WORK3 /CHECK FOR DONE LOOPING
01346 601306 JMP ANBAC /LOOP
01347 106102 JMS GENRAN /GET NO. FOR NEXT LOOP
01350 106126 JMS CKNO
/
/TEST XOR
/
01351 750000 XORAC CLA /AC = 0
01352 247530 XOR K0
01353 740200 SZA
01354 740040 F280 HALT /ERROR. AC NOT 0 AFTER XOR K0
/
01355 750001 CLA!CMA /AC = 1'S
01356 247530 XOR K0
01357 740001 CMA
01360 740200 SZA
01361 740040 F281 HALT /ERROR. AC NOT 0
/
01362 750000 CLA /AC = 0
01363 247572 XOR K7S /777777
01364 740001 CMA
01365 740200 SZA
01366 740040 F282 HALT /ERROR. AC NOT 0 AFTER XOR K7S
/
01367 750001 CLA!CMA /AC = 1'S
01370 247572 XOR K7S
01371 740200 SZA
01372 740040 F283 HALT /ERROR. AC NOT 0 AFTER XOR K7S
.EJECT

```

```

/SFQUENTIAL XOR
/
01373 750000 CLA /AC = 0
01374 247605 XOR K101 /525252
01375 247604 XOR K010 /252525
01376 247530 XOR K0 /000000
01377 247572 XOR K7S /777777
01400 247604 XOR K010
01401 247605 XOR K101
01402 247605 XOR K101
01403 247604 XOR K010
01404 740200 SZA
01405 740040 F284 HALT /ERROR. AC NOT 0
/
01406 447633 ISZ WORK3 /CHECK FOR DONE LOOPING
01407 601351 JMP XORAC /LOOP
01410 106102 JMS GENRAN /GET NO. FOR NEXT LOOP
01411 106126 JMS CKNO
/
/TFST TAD
/TADAC
01412 754000 TADAC CLA!CLI /AC = 0, LINK = 0
01413 347530 TAD K0
01414 740200 SZA
01415 740040 F285 HALT /ERROR. AC NOT 0 AFTER TAD K0
01416 741400 SZL
01417 740040 F286 HALT /ERROR. LINK NOT 0 AFTER TAD K0
/
01420 754001 CLA!CMA!CLL /AC = 1'S, LINK = 0
01421 347530 TAD K0
01422 740001 CMA
01423 740200 SZA
01424 740040 F287 HALT /ERROR.AC NOT 0
01425 741400 SZL
01426 740040 F288 HALT /ERROR. LINK NOT 0
/
01427 754002 CLA!CLL!CML /AC = 0, LINK = 1
01430 347572 TAD K7S /777777
01431 740001 CMA
01432 740200 SZA
01433 740040 F289 HALT /ERROR. TAD K7S FAILED
01434 740400 SNL
01435 740040 F290 HALT /ERROR. CARRY OUT OR OVERFLOW
/ERROR. LINK NOT 0
/
01436 754001 CLA!CMA!CLL /AC = 1'S, LINK = 0
01437 347572 TAD K7S
01440 740020 RAR
01441 740001 CMA
01442 740200 SZA
01443 740040 F291 HALT /ERROR. TAD K7S TO 1'S FAILED
01444 741400 SZL
01445 740040 F292 HALT /ERROR. LINK NOT 0
.EJECT

```

```

/TEST OVERFLOW
/
01446 754001          CLA!CMA!CLL          /AC = 1'S, LINK = 0
01447 347531          TAD K1              /000001
01450 740200          SZA
F293  HALT              /ERROR. AC NOT 0 AFTER TAD K1
01451 740040          SNL
F294  HALT              /ERROR. LINK NOT 1 OVERFLOW FAILED
01452 740400          /
01453 740040          CLA!CMA!CLL!CML    /AC = 1'S, LINK = 1
01454 754003          TAD K1
01455 347531          SZA
F295  HALT              /ERROR. AC NOT 0
01456 740200          SZL
01457 740040          F296  HALT          /ERROR. LINK NOT 0 OVERFLOW FAILED
01460 741400          /
01461 740040          /TAD 525252, AC = 252525, LINK = 0
/
01462 754000          CLA!CLL
01463 347604          TAD K010
01464 347605          TAD K111          /AC = 1'S
01465 740001          CMA
01466 740200          SZA
01467 740040          E297  HALT          /ERROR. AC NOT 0
01470 741400          SZL
01471 740040          E298  HALT          /ERROR. LINK NOT 0
/
/TAD 252525, AC = 525252, LINK = 1
/
01472 754002          CLA!CLI!CML
01473 347605          TAD K111
01474 347604          TAD K010          /AC = 1'S
01475 740001          CMA
01476 740200          SZA
01477 740040          E299  HALT          /ERROR. AC NOT 0
01500 740400          SNL
01501 740040          F300  HALT          /ERROR. LINK NOT 0
/
/SEQUENTIAL LAC, TAD, XOR
/
01502 754000          CLA!CLI
01503 207604          LAC K010          /AC = 0, LINK = 0
01504 347605          TAD K111          /AC = 1'S
01505 247604          XOR K010
01506 347604          TAD K010          /AC = 1'S
01507 347531          TAD K1
01510 740400          SNL
.EJECT

```

```

01511 740040 F301 HALT /ERROR, LINK NOT 1
01512 207605 LAC K1'1
01513 247572 XOR K7S
01514 347605 TAD K1'1
01515 247572 XOR K7S
01516 740200 SZA
01517 740040 F302 HALT /ERROR, AC NOT 0
01520 447633 ISZ WORK3 /CHECK DONE LOOPING
01521 601412 JMP TADAC /LOOP
01522 106102 JMS GENRAN /GET NO. FOR NEXT LOOP
01523 106126 JMS CKNO
/
/
/TEST ADD
/TEST ADD K1S TO K6S, LINK = 0
01524 754000 ADDAC CLA:CLL /AC = 0, LINK = 0
01525 307564 ADD K1S /111111
01526 307571 ADD K6S /666666
01527 740001 CMA /AC = 0
01530 740200 SZA
01531 740040 E303 HALT /ERROR; ADD K1S TO K6S FAILED
01532 741400 SZL
01533 740040 F304 HALT /ERROR; LINK NOT A 0
/
/TEST ADD K2S TO K5S, LINK = 0
01534 754000 CLA:CLL /AC, LINK = 0
01535 307565 ADD K2S /22222
01536 307570 ADD K5S /555555
01537 740001 CMA /AC = 0
01540 740200 SZA
01541 740040 E305 HALT /ERROR; ADD K2S TO K5S FAILED
01542 741400 SZL
01543 740040 F306 HALT /ERROR; LINK NOT A 0
/
/TEST ADD K3S TO K4S, LINK = 0
01544 754000 CLA:CLL /AC, LINK = 0
01545 307566 ADD K3S /33333
01546 307567 ADD K4S /444444
01547 740001 CMA /AC = 0
01550 740200 SZA
01551 740040 E307 HALT /ERROR; ADD K3S TO K4S FAILED
01552 741400 SZL
01553 740040 F308 HALT /ERROR; LINK NOT A 0
/
/TEST ADD K4S TO K3S, LINK = 0
01554 754000 CLA:CLL /AC, LINK = 0
01555 307567 ADD K4S /444444
01556 307566 ADD K3S /333333
01557 740001 CMA /AC = 0
01560 740200 SZA
01561 740040 E309 HALT /ERROR; AND K4S TO K3S FAILED
01562 741400 SZL
01563 740040 F310 HALT /ERROR; LINK NOT A 0
.EJECT

```

```

/TEST ADD K5S TO K2S, LINK = 0
01564      754000      CLA:CLI      /AC, LINK = 0
01565      307570      ADD K5S      /555555
01566      307565      ADD K2S      /222222
01567      740001      CMA          /AC = 0
01570      740200      SZA
01571      740040      E311        HALT          /ERROR: ADD K5S TO K2S FAILED
01572      741400      SZL
01573      740040      E312        HALT          /ERROR: LINK NOT A 0
/
/TEST AND K6S TO K1S, LINK = 0
01574      754000      CLA:CLI      /AC, LINK = 0
01575      307571      ADD K6S      /666666
01576      307564      ADD K1S      /111111
01577      740001      CMA          /AC = 0
01600      740200      SZA
01601      740040      E313        HALT          /ERROR: ADD K6S TO K1S FAILED
01602      741400      SZL
01603      740040      E314        HALT          /ERROR: LINK NOT A 0
/
/TEST ADD K7S TO K0S, LINK = 0
01604      754000      CLA:CLI      /AC, LINK = 0
01605      307572      ADD K7S      /777777
01606      307530      ADD K0      /000000
01607      740001      CMA          /AC = 0
01610      740200      SZA
01611      740040      E315        HALT          /ERROR: ADD K7S TO K0S FAILED
01612      741400      SZL
01613      740040      E316        HALT          /ERROR: LINK NOT A 0
/
/TEST ADD 252525, AC = 525252, LINK = 0
01614      754001      CLL:CLA:CMA  /AC = ONFS, LINK = 0
01615      207605      LAC K101     /AC = 525252
01616      307604      ADD K010     /AC = 252525
01617      740001      CMA          /AC = 0
01620      740200      SZA
01621      740040      E317        HALT          /ERROR: ADD K101 TO K010 FAILED
01622      741400      SZL
01623      740040      E318        HALT          /ERROR: LINK NOT A 0
/
/TEST ADD 525252, AC = 252525, LINK = 0
01624      744000      CLL          /LINK = 0
01625      207604      LAC K010     /AC = 252525
01626      307605      ADD K101     /525252
01627      740001      CMA          /AC = 0
01630      740200      SZA
01631      740040      E319        HALT          /ERROR: ADD K010 TO K101 FAILED
01632      741400      SZL
01633      740040      E320        HALT          /ERROR: LINK NOT A 0
.EJECT

```

```

/TEST ADD K7S, AC = K400K, LINK = 0
01634 754001 CLA:CM:CLL /AC = ONES, LINK = 0
01635 207553 LAC K400K /AC = 400K
01636 307572 ADD K7S /ONES
01637 507572 AND K7S /AC = 400K
01640 247553 XOR K400K
01641 740200 SZA /AC = 0
01642 740040 F321 HALT /ERROR: ADD-0 TO K400K FAILED
01643 741400 SZL
01644 740040 F322 HALT /ERROR: LINK NOT A 0, CARRY FAILED
/
/TEST ADD K200K, AC = K200K, LINK = 0
01645 754001 CLA:CM:CLL /AC = ONES, LINK = 0
01646 207553 LAC K200K /AC = 200K
01647 307557 ADD K200K /ONES
01650 507572 AND K7S /AC = 400K
01651 247553 XOR K400K
01652 740200 SZA /AC = 0
01653 740040 E323 HALT /ERROR: ADD K200K TO K200K FAILED
01654 740400 SNL
01655 740040 F324 HALT /ERROR: LINK NOT A ONES, CARRY FAILED
/
/TEST ADD K7S, AC = K100K, LINK = 1
01656 754003 CLA:CM:CLL:CML /AC = ONES, LINK = 1
01657 207320 LAC K100K /AC = 100K
01660 307572 ADD K7S /ONES
01661 507572 AND K7S /AC = 100K
01662 247320 XOR K100K
01663 740200 SZA /AC = 0
01664 740040 F325 HALT /ERROR: ADD-0 TO K100K FAILED
01665 740400 SNL
01666 740040 F326 HALT /ERROR: LINK NOT A ONE, LINK RESET
/
/TEST ADD K7S, AC = K40K, LINK = 0
01667 754001 CLA:CM:CLL /AC = ONES, LINK = 0
01670 207552 LAC K40K /AC = 40K
01671 307572 ADD K7S /ONES
01672 507572 AND K7S /AC = 40K
01673 247552 XOR K40K
01674 740200 SZA /AC = 0
01675 740040 E327 HALT /ERROR: ADD-0 TO K40K FAILED
01676 741400 SZL
01677 740040 F328 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
.EJECT

```

```

01700      754001      /TEST ADD K7S, AC = K20K, LINK = 0
01701      207556      CLAI:DMA:CLL      /AC = ONFS, LINK = 0
01702      307572      LAC K20K      /AC = 20K
01703      507572      ADD K7S      /ONFS
01704      247556      AND K7S      /AC = 20K
01705      740200      XOR K20K
01706      740040      SZA      /AC = 0
01707      741400      E329 HALT      /ERROR: ADD-0 TO K20K FAILED
01710      740040      SZA
E330 HALT      /ERROR: LINK NOT A ZERO, CARRY FAILED
/
01711      754001      /TEST AND K7S, AC = K10K, LINK = 0
01712      207555      CLAI:DMA:CLL      /AC = ONFS, LINK = 0
01713      307572      LAC K10K      /AC = 10K
01714      507572      ADD K7S      /ONFS
01715      247555      AND K7S      /AC = 10K
01716      740200      XOR K10K
01717      740040      SZA      /AC = 0
01720      741400      E331 HALT      /ERROR: ADD-0 TO K10K FAILED
01721      740040      SZA
E332 HALT      /ERROR: LINK NOT A ZERO, CARRY FAILED
/
01722      754001      /TEST ADD K7S, AC = K4K, LINK = 0
01723      207547      CLAI:DMA:CLL      /AC = ONFS, LINK = 0
01724      307572      LAC K4K      /AC = 4K
01725      507572      ADD K7S      /ONFS
01726      247547      AND K7S      /AC = 4K
01727      740200      XOR K4K
01730      740040      SZA      /AC = 0
01731      741400      F333 HALT      /ERROR: ADD-0 TO K4K FAILED
01732      740040      SZA
E334 HALT      /ERROR: LINK NOT A ZERO, CARRY FAILED
/
01733      754001      /TEST ADD K7S, AC = K2K, LINK = 0
01734      207545      CLAI:DMA:CLL      /AC = ONFS, LINK = 0
01735      307572      LAC K2K      /AC = 2K
01736      507572      ADD K7S      /ONFS
01737      247545      AND K7S      /AC = 2K
01740      740200      XOR K2K
01741      740040      SZA      /AC = 0
01742      741400      E335 HALT      /ERROR: ADD-0 TO K2K2 FAILED
01743      740040      SZA
E336 HALT      /ERROR: LINK NOT A ZERO, CARRY FAILED
.EJECT

```


01744	754001	/TEST ADD K7S, AC = K1K, LINK = 0	
01745	207543	CLA!CMA!CLL	/AC = ONES, LINK = 0
01746	307572	LAC K1K	/AC = 1K
01747	507572	ADD K7S	/ONES
01750	247543	AND K7S	/AC = 1K
01751	740200	XOR K1K	
01752	740040	SZA	/AC = 0
01753	741400	E337 HALT	/ERROR; ADD-0 TO K1K FAILED
01754	740040	SZL	
		F338 HALT	/ERROR; LINK NOT A ZERO, CARRY FAILED
		/	
01755	754001	/TEST ADD K7S, AC = K400, LINK = 0	
01756	207544	CLA!CMA!CLL	/AC = ONES, LINK = 0
01757	307572	LAC K400	/AC = 400
01760	507572	ADD K7S	/ONES
01761	247544	AND K7S	/AC = 400
01762	740200	XOR K400	
01763	740040	SZA	/AC = 0
01764	741400	F339 HALT	/ERROR; ADD-0 TO K400 FAILED
01765	740040	SZL	
01766	447633	E340 HALT	/ERROR; LINK NOT A ZERO, CARRY FAILED
01767	601524	ISZ WORK3	/CHECK DONE LOOPING
01770	106102	JMP ADDAC	/LOOP
01771	106126	JMS GEMRAN	/GET NO. FOR NEXT LOOP
		JMS CKNO	
		.EJECT	

```

/TEST ADD K7S, AC = K20, LINK = 0
01772 754001 ADDAC1 CLA:CMA:CLL /AC = ONES, LINK = 0
01773 207540 LAC K20 /AC = 20
01774 307572 ADD K7S /ONES
01775 507540 AND K20 /AC = 20
01776 247540 XOR K20
01777 740200 SZA /AC = 0
02000 740040 F347 HALT /ERROR: ADD -0 TO K20 FAILED
02001 741400 SZL
02002 740040 F348 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
/
/TEST ADD K7S, AC = K10, LINK = 0
02003 754001 CLA:CMA:CLL /AC = ONES, LINK = 0
02004 207534 LAC K10 /AC = 10
02005 307572 ADD K7S /ONES
02006 507572 AND K7S /AC = 10
02007 247534 XOR K10
02010 740200 SZA /AC = 0
02011 740040 F349 HALT /ERROR: ADD = 0 TO K10 FAILED
02012 741400 SZL
02013 740040 F350 HALT /ERROR: LINK NOT A ZERO CARRY FAILED
/
/TEST ADD K7S, AC = 4, LINK = 0
02014 754001 CLA:CMA:CLL /AC = ONES, LINK = 0
02015 207533 LAC K4 /AC = 4
02016 307572 ADD K7S /ONES
02017 507572 AND K7S /AC = 4
02020 247533 XOR K4
02021 740200 SZA /AC = 0
02022 740040 F351 HALT /ERROR: ADD -0 TO K4 FAILED
02023 741400 SZL
02024 740040 F352 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
/
/TEST ADD K7S, AC = K2, LINK = 0
02025 754001 CLA:CMA:CLL /AC = ONES, LINK = 0
02026 207532 LAC K2 /AC = 2
02027 307572 ADD K7S /ONES
02030 507572 AND K7S /AC = 2
02031 247532 XOR K2
02032 740200 SZA /AC = 0
02033 740040 F353 HALT /ERROR: ADD -0 TO K2 FAILED
02034 741400 SZL
02035 740040 F354 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
/
/TEST ADD K7S, AC = K1, LINK = 0
02036 754001 CLA:CMA:CLL /AC = ONES, LINK = 0
02037 207531 LAC K1 /AC = 1
02040 307572 ADD K7S /ONES
02041 507572 AND K7S /AC = 1
02042 247531 XOR K1
02043 740200 SZA /AC = 0
02044 740040 F355 HALT /ERROR: ADD -0 TO K1 FAILED
02045 741400 SZL
02046 740040 F356 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
.EJECT

```

```

02047 744000 /TEST ADD K400K, AC = ONES, LINK = 0
02050 777777 CLL /LINK = 0
02051 507572 LAW 17777 /AC = ONES
02052 307553 AND K7S /AC = ONES
02053 247553 ADD K400K /400K
02054 740200 XOR K400K
02055 740040 SZA /AC = 0
E357 HALT /ERROR: ADD K400K TO -0 FAILED
02056 741400 SZL
E358 HALT /ERROR: LINK NOT A ZERO, CARRY FAILED
/
02060 744002 /TEST ADD K200K, AC = ONES, LINK = 1
02061 777777 CLL:CML /LINK = 1
02062 507572 LAW 17777 /AC = ONES
02063 307557 AND K7S /AC = ONES
02064 247557 ADD K200K /200K
02065 740200 XOR K200K
02066 740040 SZA /AC = 0
E359 HALT /ERROR: ADD K200K TO -0 FAILED
02067 740400 SNL
E360 HALT /ERROR: LINK NOT A ONE, LINK RESET
/
02071 744002 /TEST ADD K100K, AC = ONES, LINK = 1
02072 777777 CLL:CML /LINK = 1
02073 507572 LAW 17777 /AC = ONES
02074 307320 AND K7S /AC = ONES
02075 247320 ADD K100K /100K
02076 740200 XOR K100K
02077 740040 SZA /AC = 0
E361 HALT /ERROR: ADD K100K TO -0 FAILED
02100 740400 SNL
E362 HALT /ERROR: LINK NOT A ONE, LINK RESET
/
02102 744002 /TEST ADD K40K, AC = ONES, LINK = 1
02103 777777 CLL:CML /LINK = 1
02104 507572 LAW 17777 /AC = ONES
02105 307552 AND K7S /AC = ONES
02106 247552 ADD K40K /40K
02107 740200 XOR K40K
02110 740040 SZA /AC = 0
E363 HALT /ERROR: ADD K40K TO -0 FAILED
02111 740400 SNL
E364 HALT /ERROR: LINK NOT A ONE, LINK RESET
/
02113 744002 /TEST ADD K20K, AC = ONES, LINK = 1
02114 777777 CLL:CML /LINK = 1
02115 507572 LAW 17777 /AC = ONES
02116 307556 AND K7S /AC = ONES
02117 247556 ADD K20K /20K
02120 740200 XOR K20K
02121 740040 SZA /AC = 0
E365 HALT /ERROR: ADD K20K TO -0 FAILED
02122 740400 SNL
E366 HALT /ERROR: LINK NOT A ONE, LINK RESET
02123 740040

```

```

02124 744002 /TFST ADD K10K, AC = ONES, LINK = 1
02125 777777 CLL!CML /LINK = 1
02126 507572 LAW 17777 /AC = ONES
02127 307555 AND K7S /AC = ONES
02130 247555 ADD K10K /10K
02131 740200 XOR K10K
02132 740040 SZA /AC = 0
02133 740400 E367 HALT /ERROR; ADD K10, TO -0 FAILED
02134 740040 SNL
E368 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02135 744002 /TFST ADD K4K, AC = ONES, LINK = 1
02136 777777 CLL!CML /LINK = 1
02137 507572 LAW 17777 /AC = ONES
02140 307547 AND K4K /AC = ONES
02141 247547 ADD K4K /4K
02142 740200 XOR K4K
02143 740040 SZA /AC = 0
02144 740400 E369 HALT /ERROR; ADD K4K TO -0 FAILED
02145 740040 SNL
E370 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02146 744002 /TFST ADD K2K, AC = ONES, LINK = 1
02147 777777 CLL!CML /LINK = 1
02150 507572 LAW 17777 /AC = ONES
02151 307545 AND K2K /AC = ONES
02152 247545 ADD K2K /2K
02153 740200 XOR K2K
02154 740040 SZA /AC = 0
02155 740400 E371 HALT /ERROR; AC K2K TO -0 FAILED
02156 740040 SNL
E372 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02157 744002 /TFST ADD K1K, AC = ONES, LINK = 1
02160 777777 CLL!CML /LINK = 1
02161 507572 LAW 17777 /AC = ONES
02162 307543 AND K1K /AC = ONES
02163 247543 ADD K1K /1K
02164 740200 XOR K1K
02165 740040 SZA /AC = 0
02166 740400 E373 HALT /ERROR; ADD K1K TO -0 FAILED
02167 740040 SNL
E374 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02170 744002 /TFST ADD K400, AC = ONES, LINK = 1
02171 777777 CLL!CML /LINK = 1
02172 507572 LAW 17777 /AC = ONES
02173 307544 AND K400 /AC = ONES
02174 247544 ADD K400 /400
02175 740200 XOR K400
02176 740040 SZA /AC = 0
02177 740400 E375 HALT /ERROR; ADD K400 TO -0 FAILED
02200 740040 SNL
E376 HALT /ERROR; LINK NOT A ONE, LINK RESET
.EJECT

```

```

02201 744002 /TEST ADD K200, AC = ONES, LINK = 1
02202 777777 CLL!CML /LINK = 1
02203 507572 LAW 17777 /AC = ONES
02204 307551 AND K7S /AC = ONES
02205 247551 ADD K200 /200
02206 740200 SZA /AC = 0
02207 740040 E377 HALT /ERROR; ADD K200 TO -0 FAILED
02210 740400 SNL
02211 740040 E378 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02212 744002 /TEST ADD K100, AC = ONES, LINK = 1
02213 777777 CLL!CML /LINK = 1
02214 507572 LAW 17777 /AC = ONES
02215 307537 AND K7S /AC = ONES
02216 247537 ADD K100 /100
02217 740200 SZA /AC = 0
02220 740040 E379 HALT /ERROR; ADD K100 TO -0 FAILED
02221 740400 SNL
02222 740040 E380 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02223 744002 /TEST ADD K40, AC = ONES, LINK = 1
02224 777777 CLL!CML /LINK = 1
02225 507572 LAW 17777 /AC = ONES
02226 307542 AND K7S /AC = ONES
02227 247542 ADD K40 /40
02230 740200 SZA /AC = 0
02231 740040 E381 HALT /ERROR; ADD K40 TO -0 FAILED
02232 740400 SNL
02233 740040 E382 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02234 744002 /TEST ADD K20, AC = ONES, LINK = 1
02235 777777 CLL!CML /LINK = 1
02236 507572 LAW 17777 /AC = ONES
02237 307540 AND K7S /AC = ONES
02240 247540 ADD K20 /20
02241 740200 SZA /AC = 0
02242 740040 E383 HALT /ERROR; ADD K20 TO -0 FAILED
02243 740400 SNL
02244 740040 E384 HALT /ERROR; LINK NOT A ONE, LINK RESET
/
02245 744002 /TEST ADD K10, AC = ONES, LINK = 1
02246 777777 CLL!CML /LINK = 1
02247 507572 LAW 17777 /AC = ONES
02250 307534 AND K7S /AC = ONES
02251 247534 ADD K10
02252 740200 SZA /AC = 0
02253 740040 E385 HALT /ERROR; ADD K10 TO -0 FAILED
02254 740400 SNL
02255 740040 E386 HALT /ERROR; LINK NOT A ONE, LINK RESET
.EJECT

```

```

02256      744002      /TEST ADD K4, AC = ONES, LINK = 1
02257      777777      CLL:CM1      /LINK = 1
02260      507572      LAW 17777      /AC = ONFS
02261      307533      AND K7S      /AC = ONFS
02262      247533      ADD K4      /4
02263      740200      XOR K4
02264      740040      SZA      /AC = 0
02265      740400      F387 HALT      /ERROR: ADD K4 TO -0 FAILED
02266      740040      SNL
F388 HALT      /ERROR: LINK NOT A ONE, LINK RESET
/
02267      744002      /TEST ADD K2, AC = ONES, LINK = 1
02270      777777      CLL:CM1      /LINK = 1
02271      507572      LAW 17777      /AC = ONFS
02272      307532      AND K7S      /AC = ONFS
02273      307532      ADD K2      /2
02274      247532      XOR K2
02274      740200      SZA      /AC = 0
02275      740040      F389 HALT      /ERROR: ADD K2 TO -0 FAILED
02276      740400      SNL
02277      740040      F390 HALT      /ERROR: LINK NOT A ONE, LINK RESET
/
02300      744002      /TEST ADD K1, AC = ONES, LINK = 1
02301      777777      CLL:CM1      /LINK = 1
02302      507572      LAW 17777      /AC = ONFS
02303      307531      AND K7S      /AC = ONFS
02304      247531      ADD K1      /1
02305      740200      XOR K1
02306      740040      SZA      /AC = 0
02307      740400      F391 HALT      /ERROR: ADD K1 TO -0 FAILED
02310      740040      F392 HALT      /ERROR: LINK NOT A ONE, LINK RESET
/
02311      744000      /TEST ADD K7S, AC = ONFS, LINK = 0
02312      207572      CLL      /LINK = 0
02313      307572      LAC K7S      /AC = ONFS
02314      740001      ADD K7S      /ONES
02315      740200      CMA      /AC = ONES
02316      740040      SZA      /AC = 0
02317      741400      F393 HALT      /ERROR: ADD K7S TO ALL ONES FAILED
02320      740040      F394 HALT      /ERROR: LINK NOT A ONE, LINK RESET
/
02321      744002      /TEST ADD 525253, AC = 252525, LINK = 1
02322      207604      CLL:CM1      /LINK = 1
02323      307606      LAC K010      /AC = 252525
02324      247531      ADD K53      /525253
02325      740200      XOR K1      /000001
02326      740040      SZA      /AC = 0
02327      740400      F395 HALT      /ERROR: ADD K5253 TO K5252 FAILED
02330      740040      F396 HALT      /ERROR: LINK NOT A ONE, LINK RESET
      .EJECT

```

02331	744000	/TEST ADD 252525, AC = 525253, LINK RESET	
02332	207606	CLI	/LINK = 0
02333	307604	LAC K53	/AC = 525253
02334	247531	ADD K010	/252525
02335	740200	XOR K1	/000001
02336	740040	SZA	/AC = 0
02337	741400	F397 HALT	/ERROR; ADD K2525 TO K5253 FAILED
02340	740040	F398 SZI	/ERROR; LINK NOT A ZERO, CARRY FAILED
		/	
		/TEST ADD SERIES	
02341	754000	CLA:CLI	/LINK = 0, AC = 0
02342	307564	ADD K1S	/AC = 111111
02343	307565	ADD K2S	/AC = 333333
02344	307566	ADD K3S	/AC = 666666
02345	307567	ADD K4S	/AC = 333333, LINK = 1
02346	307570	ADD K5S	/AC = 111111
02347	307571	ADD K6S	/AC = 777777
02350	307572	ADD K7S	/AC = 777777
02351	740001	CMA	/AC = 0
02352	740200	SZA	
02353	740040	E399 HALT	/ERROR; ADD SERIES FAILED
02354	740400	SNL	
02355	740040	F400 HALT	/ERROR; LINK NOT A 1
02356	447633	ISZ WORK3	/CHECK DONE LOOPING
02357	601772	JMP ADDAC1	/LOOP
02360	106102	JMS GENRAN	/GET NO. FOR NEXT LOOP
02361	106126	JMS CKNO	
		.EJECT	

```

/ADD RANDOM PAIRS TEST
/
00362 106102 RANADD JMP GETRAN /GET RANDOM NUMBER
00363 741100 SPA /* NO
00364 740001 CMA /- MAKE IT +
00365 741200 SNA /0 NOT ALLOWED
00366 602362 JMP RA ADD
00367 343016 DAC APOS /IT IS + A
00370 740001 CMA /1 COMPLEMENT
00371 043017 MINUSA DAC ANDG /IT IS -A
00372 106102 JMP GETRAN /GET NEXT RANDOM
00373 741100 SPA /* NO
00374 740001 CMA /- MAKE +
00375 741200 SNA /0 NOT ALLOWED
00376 602372 JMP MINUSA+1
00377 043020 DAC BPOS /IT IS + R
00400 740001 CMA /MAKE 1'S COMP
00401 043021 MINUSB DAC BNEG /IT IS - R
00402 777777 LAR -1
00403 043027 DAC PASS2
00404 744000 CLL
00405 203021 LAC BNEG /RESTART HERE TO REGENERATE NEW COMPARES
00406 343017 TAD BNEG /-R -A
00407 741400 SZL /EOC IF ADD
00410 347531 TAD K1 /YES MAKE CARRY
00411 043022 MINSAR DAC SUMNEG /SAVE -A -R
/
/NOW GENERATE A + B
00412 203016 LAC APOS /GET +A
00413 744000 CLL
00414 343020 TAD BPOS /*B
00415 741400 SZL /EOC IF ADD
00416 347531 TAD K1 /YES ADD CARRY
00417 043023 APLUSB DAC SUMPOS
.EJECT

```


02420	203020	/NOW GENERATE B-A	
02421	744000	LAC BP0S	/GET B
02422	343017	CLL	
02423	741400	TAD ANFG	/B-A
02424	347531	SZL	/EOC
02425	043024	TAD K1	/YES ADD CARRY
		BMINSA DAC BMSUM	/SAVE B-A
		/	
		/NOW GENERATE A-B	
02426	203016	LAC APOS	/GET A
02427	744000	CLL	
02430	343021	TAD BNFG	/-B
02431	741400	SZL	/EOC
02432	347531	TAD K1	/YES ADD CARRY
02433	043025	AMINSB DAC AMRSUM	/A-B
		/	
		/IF A+B IS AN OVERFLOW SITUATION	
		/MAKE OFLOW TESTS THAT APPLY = SNL	
		/IF A+B IS NOT OVERFLOW MAKE	
		/OVERFLOW TEST THAT APPLY = SZL	
		/	
02434	203023	LAC SUMPOS	/GET A+B
02435	751100	SPA:CLA	/STILL POS RESULT
02436	207653	LAC KSNL	/NEG RESULT IS OVERFLOW
02437	741200	SNA	/AC = SNL IS OVERFLOW
02440	207652	LAC KSZL	/+ RFSULT IS NO OVERFLOW
02441	042464	DAC OFLCK1	/SET UP ALL OFLOW
02442	042512	DAC OFLCK3	/TESTS WHERE OFLOW
02443	042627	DAC OFLCH1	/MAY OR MAY NOT OCCUR
02444	042644	DAC OFLCH2	/AC = SNL IS A+B OFLOW
02445	042662	DAC OFLCH3	/AC = SZL IS A+B NOT OFLOW
02446	042701	DAC OFLCH4	/IF A+B OFLOW -A-B DOES ALSO
02447	042721	DAC OFLCH5	/IF A+B NOT OFLOW
02450	042742	DAC OFLCH6	/THEN NONE OF THESE
02451	042764	DAC OFLCH7	/ADDS CAN OVERFLOW
02452	042540	DAC OFLCK5	
02453	042566	DAC OFLCK7	
		.EJECT	

/NOW DO A COMPLETE SERIES OF
 /ONES COMP ADDITIONS
 /SHOULD GET THE SAME RESULTS AS
 /THE TAB'S WITH FDC TAB (1
 /
 /FIRST TEST A+B
 /

00454 744000
 00455 203016
 00456 303020
 00457 543023
 00460 602464
 00461 740040
 00462 203023
 00463 740040
 00464 741400
 00465 740040
 00466 762454

```

APLSRT  CLI           /FOR OVERFLOW CHECK
          LAC AP+S     /GET A
          ADD RP+S     /A+B
          SAB SUMPOS   /SHOULD = PREVIOUS A+B
          JMP .+1      /OK
E4W1    HLT           /DISPLAY 1'S A+B
          LAC SUMPOS   /GET 2'S COMP GEN
          HLT           /DISPLAY 2'S A+B
OFLOCK1 SZL           /OR SNI IF OVERFLOW
E4W2    HLT           /LINK OR OVERFLOW FAILED
          LAK APLSRT   /MAKE JUMP FOR SCOPE LOOP
          .EJECT
  
```

```

00467 744000 /2ND TEST -B+A
00470 203021 RMNSRT CLI
00471 303016 LAC BNFG /GET A
00472 543025 ADD APIS /A-B
00473 602477 SAA AMRSUM /SHOULD = PREVIOUS A-B
00474 740040 JMP .+ /OK
00475 203025 F403 HLT /DISPLAY 1'S A-B
00476 740040 LAC AMRSUM
00477 741400 HLT /DISPLAY 2'S A-B
00500 740040 OFLCK2 SZL /SHOULD NOT OVERFLOW
00501 762467 F404 HLT
LAW AMRSRT /MAKE JMP FOR SCOPE
/
/NOW 3RD TEST IS -A -B
00502 744000 MAPLMB CLI
00503 203017 LAC ANFG /GET -A
00504 303021 ADD BNFG /PLS -B
00505 543022 SAA SUMNEG /SHOULD = PREVIOUS -A-B
00506 602512 JMP .+4 /OK
00507 740040 F405 HLT /DISPLAY 1'S -A-B
00510 203022 LAC SUMNEG
00511 740040 HLT /DISPLAY 2'S -A-B
00512 741400 OFLCK3 SZL /OR SNL
00513 740040 F406 HLT /LINK FAILED
00514 762502 LAW MAPLMB /MAKE JMP FOR SCOPE
/
/FOURTH TEST IN THIS SERIES
/IS TEST B-A
00515 744000 RMNSAT CLI
00516 203020 LAC BPOS /GET B
00517 303017 ADD ANFG /ADD -A
00520 543024 SAA BMASUM /SHOULD = PREVIOUS B-A
00521 602525 JMP .+4 /OK
00522 740040 HLT /DISPLAY 1'S B-A
00523 203024 LAC BMASUM
00524 740040 HLT /DISPLAY 2'S B-A
00525 741400 SZL /CAN NOT OVERFLOW
00526 740040 HLT /OVERFLOW FAILED
00527 762515 LAW RMNSAT /MAKE JMP FOR SCOPE
.EJECT

```

/FIFTH TEST IN THIS SERIES
/IS TEST (A+B)-A = B

02530 744000
02531 203023
02532 303017
02533 543020
02534 602540
02535 740040
02536 203020
02537 740040
02540 741400
02541 740040
02542 762530

ARMATS CLL
LAC SUMPOS
ADD ANFG
SAD RPOS
JMP .+4
E407 HLT
LAC RPOS
HLT
OFLCK5 SZL
E408 HLT
LAW ARMATS

/CAN OVERFLOW SNL IF A+B OVERFLOW
/ILLFGAL LINK
/MAKE JMP FOR SCOPE

/SIXTH TEST IN THIS SERIES
/IS TEST (B-A)-B = -A

02543 744000
02544 203024
02545 303021
02546 543017
02547 602553
02550 740040
02551 203017
02552 740040
02553 741400
02554 740040
02555 762543

RMAMBT CLL
LAC RMASUM
ADD BNFG
SAD ANFG
JMP .+4
E409 HLT
LAC ANFG
HLT
OFLCK6 SZL
E410 HLT
LAW RMAMBT

/CAN NOT OVERFLOW
/MAKE JMP FOR SCOPE

/SEVENTH TEST IN THIS SERIES
/IS (-A-B)+A = -B

02556 744000
02557 203022
02560 303016
02561 543021
02562 602566
02563 740040
02564 203021
02565 740040
02566 741400
02567 740040
02570 762556

MARPAT CLL
LAC SUMNEG
ADD APOS
SAD BNEG
JMP .+4
E411 HLT
LAC BNEG
HLT
OFLCK7 SZL
E412 HLT
LAW MARPAT
.EJECT

/CAN BE OVERFLOW IF A+B OVERFLOW THEN IS SNL
/ILLFGAL LINK
/MAKE JMP FOR SCOPE

```

/ EIGHTH TEST OF THE SERIES
/ IS (A-1) + R = A
AMPRT  CLI
      LAC AMLSUM
      ADD RPS
      SAE APDS
      JMP ,+1
F413  HLT
      LAC APDS
      HLT
BFCK8  SZL /CAN NOT OVERFLOW
F414  HLT
      LAC AMPRT /MAKE JMP FOR SCOPE
/
/ 9TH TEST OF SERIES
/ NOW TEST AC = 777777 + A = A
/
M0ACPA  CLI!CLA!CMA /SET AC = 777777
      ADD APDS /+ A
      SAE APDS /SHOULD = A
      JMP ,+4 /TEST LINK
F415  HLT /FAILED RESULTS
      LAC APDS
      HLT /DISPLAY A
BFCK9  SZL /CANNOT OVERFLOW
F416  HLT /OVERFLOW FAILED L - 1
      LAC M0ACPA /MAKE JMP FOR SCOPE
      .EJECT

```

```

00571  744000
00572  203025
00573  303020
00574  543016
00575  602601
00576  740040
00577  203016
00600  740040
00601  741400
00602  740040
00603  762671

```

```

00604  754001
00605  303016
00606  543016
00607  602613
00610  740040
00611  203016
00612  740040
00613  741400
00614  740040
00615  762604

```

/THE NEXT SERIES OF TESTS
 /ARE ADD SEQUENCES THE RESULTS
 /OF WHICH HAVE ALREADY BEEN
 /COMPUTED AND VERIFIED
 /
 /FIRST SERIES TESTS A+B OK, THEN (A+B)-A = B
 /SEE ARMATS FOR SHORTER TEST OR APLSRT OR M0ACPA
 /NOW TRY A+B-A = B

02616 754001
 02617 303016
 02620 303020
 02621 303017
 02622 543020
 02623 602627
 02624 740040
 02625 203020
 02626 740040
 02627 741400
 02630 740040
 02631 762616

SERS01 CLL!CLA!CMA
 ADD APDS
 ADD BPDS
 ADD ANFG
 SAA BPDS
 JMP .+4
 E417 HLT
 LAC BPDS
 HLT
 OFLCH1 SZL
 E418 HLT
 LAW SERS01

/OR SNL IF A+B OVERFLOW
 /LINK FAILURE
 /MAKE JMP FOR SCOPF

/
 /HAVE TESTED B-A PREVIOUS
 /SEE BMNSAT FOR SHORTER TEST
 /NOW TRY A+B-A-A = B-A

02632 754001
 02633 303016
 02634 303020
 02635 303017
 02636 303017
 02637 543024
 02640 602644
 02641 740040
 02642 203024
 02643 740040
 02644 740400
 02645 740040
 02646 762632

SERS02 CLL!CLA!CMA
 ADD APDS
 ADD BPDS
 ADD ANFG
 ADD ANFG
 SAA BMSUM
 JMP .+4
 E419 HLT
 LAC BMSUM
 HLT
 OFLCH2 SNI
 E420 HLT
 LAW SERS02
 .EJECT

/OR SZL IF NO OVERFLOW
 /MAKE JMP FOR SCOPF

```

/HAVE TESTED (R-A)-B = -A PREVIOUS
/SFE RMAMBT FOR SHORTER TEST
/NOW TRY A+B-A-A-B = -A
SER503  CLI!CLA!CMA
        ADD APOS
        ADD BPOS
        ADD ANFG
        ADD ANFG
        ADD RNFG
        SAA ANFG
        JMP .+4
F421    HLT
        LAC ANFG
        HLT
OFLCH3  SZL /SNL IF A+B OVERFLOW
F422    HLT /OVERFLOW FAILED
        LAW SER503 /MAKE JMP FOR SCOPE
/
/HAVE TEST -A-B NOW TRY A+B-A-A-B-R = -A-B
/SFE MAPLMR FOR SHORTER TEST
SER504  CLI!CLA!CMA
        ADD APOS
        ADD BPOS
        ADD ANFG
        ADD ANFG
        ADD RNFG
        ADD RNFG
        SAA SUMNEG
        JMP .+4
F423    HLT
        LAC SUMNEG
        HLT
OFLCH4  SZL /SNL IF A+B OVERFLOW
F424    HLT /OVERFLOW FAILED OR LINK FAILED
        LAW SER504 /MAKE JMP FOR SCOPE LOOP
        .EJECT

```

02647	754001
02650	303016
02651	303020
02652	303017
02653	303017
02654	303021
02655	543017
02656	602662
02657	740040
02660	203017
02661	740040
02662	741400
02663	740040
02664	762647

02665	754001
02666	303016
02667	303020
02670	303017
02671	303017
02672	303021
02673	303021
02674	543022
02675	602701
02676	740040
02677	203022
02700	740040
02701	741400
02702	740040
02703	762665

/HAVE TESTED (-A-R)+A = -R NOW A+R-A-A-B-R+A = -R
 /USE MARPAT FOR SHORTER TEST

02704 754001
 02705 303016
 02706 303020
 02707 303017
 02710 303017
 02711 303021
 02712 303021
 02713 303016
 02714 543021
 02715 602721
 02716 740040
 02717 203021
 02720 740040
 02721 740040
 02722 740040
 02723 762704

SERS05 CLL!CLA!CMA
 ADD APOS
 ADD BPOS
 ADD ANFG
 ADD ANFG
 ADD BNFG
 ADD BNFG
 ADD APOS
 SAA BNFG
 JMP .+4
 E425 HLT
 LAC BNFG
 HLT
 OFLCH5 SNL
 E426 HLT
 LAW SERS05

/OR SZL IF A+B DO NOT OVERFLOW
 /LINK OR OVERFLOW FAILED
 /MAKE JMP FOR SCOPE

/
 /HAVE DONE -R+A PREVIOUSLY
 /NOW DO A+R-A-A-R-B+A+A = -R+A
 /USE AMNSRT FOR SHORTER TEST

02724 754001
 02725 303016
 02726 303020
 02727 303017
 02730 303017
 02731 303021
 02732 303021
 02733 303016
 02734 303016
 02735 543025
 02736 602742
 02737 740040
 02740 203025
 02741 740040
 02742 741400
 02743 740040
 02744 762724

SERS06 CLL!CLA!CMA
 ADD APOS
 ADD BPOS
 ADD ANFG
 ADD ANFG
 ADD BNFG
 ADD BNFG
 ADD APOS
 ADD APOS
 SAA AMRSUM
 JMP .+4
 E427 HLT
 LAC AMRSUM
 HLT
 OFLCH6 SZL
 E428 HLT
 LAW SERS06
 .EJECT

/OR SNL IF A+B OVERFLOW
 /OVERFLOW OR LINK FAILED
 /MAKE JMP FOR SCOPE

/HAVE DONE (-B+A)+R PREVIOUSLY
 /NOW DOE A+B-A-A-R-B+A+A+B = A
 /USE AMRPBT FOR SHORTER EST

00745 754001
 00746 303016
 00747 303020
 00750 303017
 00751 303017
 00752 303021
 00753 303021
 00754 303016
 00755 303016
 00756 303020
 00757 543016
 00760 602764
 00761 740040
 00762 203016
 00763 740040
 00764 740040
 00765 740040
 00766 762745

SERS07 CLL:CLL:OMA
 ADD AP:IS
 ADD BP:IS
 ADD ANFG
 ADD BNFG
 ADD BNFG
 ADD AP:IS
 ADD AP:IS
 ADD BP:IS
 SAD AP:IS
 JMP .+4
 F429 HLT
 LAC AP:IS
 HLT
 OFLCH7 SNL
 F430 HLT
 LAW SERS07

/OR SZL IF A+B NOT OVERFLOW
 /LINK OR OVERFLOW FAILED
 /MAKE JMP FOR SCOPE LOOP

/
 /AFTER ONE PASS
 /MAKE ALL R CONSTANTS A
 /AND MAKE ALL A CONSTANTS R
 /

00767 443027
 00770 603011
 00771 203016
 00772 043020
 00773 203021
 00774 043017
 00775 740001
 00776 043016
 00777 203020
 03000 740001
 03001 043021
 03002 203024
 03003 040010
 03004 203025
 03005 043024
 03006 200010
 03007 043025
 03010 602454

CONCHG IS7 PASS2 /2ND PASS
 JMP CKLP /YES DONF 2ND
 LAC AP:IS /A
 DAC BP:IS /IS NOW R
 LAC BNFG /RNEG
 DAC ANFG /IS ANEG
 CMA
 DAC AP:IS /B IS A
 LAC BP:IS
 CMA /ANEG
 DAC BNFG /IS NEG
 LAC BMASUM
 DAC 10
 LAC AMRSUM
 DAC BMASUM
 LAC 10
 DAC AMRSUM
 JMP APLSRT
 .EJECT

/A-B
 /IS NOW A-R
 /R-A
 /IS NOW A-R
 /OVERFLOW SERUP

03011 447633
 03012 602362
 03013 106102
 03014 106126
 03015 603030

CKLP ISZ WORK3
 JMP RARADD
 JMS GENRAN
 JMS CKVO
 JMP ADFDON

/CHECK DONE LOOPING
 /LOOP
 /GET NO. FOR NEXT LOOP

03016 000000
 03017 000000
 03020 000000
 03021 000000
 03022 000000
 03023 000000
 03024 000000
 03025 000000

/
 /
 APOS 0 /A
 ANEG 0 /-A
 FPOS 0 /B
 RNEG 0 /-B
 SUMNEG 0 /-A+(-B)
 SUMPOS 0 /A+B
 RMASUM 0 /B+(-A)
 AMRSUM 0 /A+(-B)

03026 000000

/
 MSKBIT 0
 .EJECT

03027 000000

PASS- 0
 /GET A RANDOM NUMBER AND ITS 1'S COMPLIMENT
 /EACH BIT WILL HAVE A 0 IN ONE OF THE TWO NUMBERS
 /MAKE THE 0 BIT = 1 AND ADD THE NUMBERS BOTH WAYS
 /FIRST ADD IS THE (AC) IS THE ALTERED 0 = 1
 /SECOND ADD IS THE (MB) IS THE ALTERED 0 = 1
 /THE RESULT OF BOTH ADDS SHOULD = THE ALTERED BIT = 1
 /

03030 106102
 03031 043016
 03032 740001
 03033 043017

ADDEDN JMS GERAN /GET RANDOM NUMBER
 DAC APOS /SAVE IT
 CMA /MAKE ONES COMPLIMENT
 DAC ANEG /AND SAVE IT

03034 207553
 03035 043026

/THE FIRST BIT TO BE ALTERED IS 0 THEN CONTINUE TO 17
 LAC K400K
 DAC MSKBIT

03036 203016
 03037 503026
 03040 740200
 03041 603050
 03042 203016
 03043 243026
 03044 043020
 03045 203017
 03046 043021
 03047 603055

/SET UP NEXT BIT TO TEST - ALTERED NUMBER GOES TO BPOS
 R1SETU LAC APOS
 AND MSKBIT
 SZA /DOES APOS BIT = 0
 JMP MODNEG /NO ALTER ANEG
 LAC APOS
 XOR MSKBIT
 DAC BPOS /MODIFIED NUMBER GOES TO APOS
 LAC ANEG
 DAC BNFG /UNMOD NUMBER GOES TO RNEG
 JMP BITTS1

03050 203017
 03051 243026
 03052 043020
 03053 203016
 03054 043021

/THE ONES COMP NUMBER HAS THE 0 BIT MODIFY IT
 MODNEG LAC ANFG
 XOR MSKBIT
 DAC BPOS /MOD NUMBER TO BPOS
 LAC APOS
 DAC BNFG /UNMOD NUMBER TO RNEG

03055 744000
 03056 203020
 03057 303021
 03060 543026
 03061 603065
 03062 740040
 03063 203026
 03064 740040
 03065 741400
 03066 740040
 03067 763055

/COMPLEMENTED BIT TEST1 (AC) = MODIFIED NUMBER AT ADD
 /
 BITTS1 CLL
 LAC BPOS /GET MODIFIED NUMBER
 ADD RNFG /ADD UNMODIFIED
 SDA MSKBIT /RESULT SHOULD = BIT CHANGED
 JMP .+4
 F431 HLT /DISPLAY INCORRECT RESULTS
 LAC MSKBIT
 HLT
 OFLCH8 SZA /DISPLAY BIT ALTERED AND EXP
 /NO OVERFLOW
 F432 HLT /OVERFLOW NOT ALLOWED
 LAW BITTS1 /MAKE JMP FOR SCOPE LOOP
 .EJECT

```

/COMP BIT TEST 2 (MR) = MODIFIED NUMBER AT ADD
/
03070 744000
03071 203021
03072 303020
03073 543026
03074 603100
03075 740040
03076 203026
03077 740040
03100 741400
03101 740040
03102 763070
E432 CLL
E433 LAC RNEG
ADD BPOS
SAC MSKBIT
JMP .+1
HLT
LAC MSKBIT
SZI
E434 HLT
LAK BITTS2
/GET UNMODIFIED NUMBER
/ADD MODIFIED
/RESULT SHOULD = BIT CHANGED
/OK
/DISPLAY INCORRECT RESULTS
/DISPLAY BIT ALTERED AND EXP
/SHOULD NOT OVERFLOW
/MAKE JMP FOR SCOPE LOOP
/
/POSITION MASK BIT OVER 1 PLACE
/IF 17 HAS BEEN DONE CONTINUE
/
03103 203026
03104 744020
03105 043026
03106 740200
03107 603036
LAC MSKBIT
RCP
DAC MSKBIT
SZA
JMP RISETU
/GET LAST
/POSITION
/SAVE
/DONE ALL BITS
/DO FOR NEXT BIT
/
/END OF TEST SEQUENCE
/
03110 447633
03111 603030
03112 106102
03113 106126
03114 750004
03115 742010
03116 741100
03117 602362
IS7 WORK3
JMP ADFDON
JMS GENRAN
JMS CKNO
LAS
RTL
SPA
JMP RAWADD
/CHECK DONE LOOPING
/LOOP
/GET NO. FOR NEXT LOOP
/CHECK FOR CONTINUOUS LOOP
/CK ACS ?
/LOOP
/PDP-9 BASIC EXERCISER - TAPE 4
/TEST SAD
/
03120 207530
03121 547530
03122 741000
03123 740040
SADAC LAC K0
SAD K0
SKP
E435 HALT
/AC = 0
/ERROR, SAD K0 SKIPPED
/
03124 207530
03125 547572
03126 740040
LAC K0
SAD K7S
E436 HALT
/AC = 0
/ERROR, SAD K7S FAILED TO SKIP
/
03127 207572
03130 547530
03131 740040
LAC K7S
SAD K0
E437 HALT
/AC = 1'S
/ERROR, SAD K0 FAILED TO SKIP
/
03132 207572
03133 547572
03134 741000
03135 740040
LAC K7S
SAD K7S
SKP
E438 HALT
/AC = 1'S
/ERROR, SAD K7S SKIPPED
/
/SAD, TAD

```

03136	750000	/	CLA	/AC = 0
03137	347530		TAD K0	
03140	547530		SAD K0	
03141	741000		SKP	
03142	740040	F439	HALT	/ERROR, SAD K0 SKIPPED
		/		
03143	750000		CLA	/AC = 0
03144	347530		TAD K0	
03145	547572		SAD K7S	
03146	740040	F440	HALT	/ERROR, SAD K7S FAILED TO SKIP
		/		
03147	750000		CLA	/AC = 0
03150	347572		TAD K7S	
03151	547530		SAD K0	
03152	740040	F441	HALT	/ERROR, SAD K0 FAILED TO SKIP
		/		
03153	750000		CLA	/AC = 0
03154	347572		TAD K7S	
03155	547572		SAD K7S	
03156	741000		SKP	
03157	740040	F442	HALT	/ERROR, SAD K7S SKIPPED
		/		
			.EJECT	

/SEQUENTIAL SAD

03160	207530	LAC KW	/AC = 0
03161	547572	SAD K7S	
03162	760001	LAW 1	/760001
03163	547572	SAD K7S	
03164	760002	LAW 2	/760002
03165	547572	SAD K7S	
03166	760004	LAW 4	/760004
03167	547572	SAD K7S	
03170	760010	LAW 10	/760010
03171	547572	SAD K7S	
03172	760020	LAW 20	/760020
03173	547572	SAD K7S	
03174	760040	LAW 40	/760040
03175	740200	SZA	
03176	740040	HALT	/ERROR. AC NOT 0. CONTENTS OF /AC = LAST SAD THAT FAILED

E443

/TEST SAD, SKP SERIES

03177	750000	CLA	/AC = 0
03200	547572	SAD K7S	
03201	760001	LAW 1	/760001
03202	741000	SKP	
03203	760002	LAW 2	/760002
03204	547572	SAD K7S	
03205	760004	LAW 4	/760004
03206	741000	SKP	
03207	760010	LAW 10	/760010
03210	547572	SAD K7S	
03211	760020	LAW 20	/760020
03212	741000	SKP	
03213	760040	LAW 40	/760040
03214	740200	SZA	
03215	740040	HALT	/ERROR. AC NOT 0. CONTENTS /OF AC = LAST SAD OR SKP

E444

03216	447633	ISZ WORK3	/CHECK DONE LOOPING
03217	603120	JMP SADAC	/LOOP
03220	106102	JMS GENRAN	/GET NO. FOR NEXT LOOP
03221	106126	JMS CKNO	

.EJECT

```

/TEST DZM
/
03222 207647 DZMAC LAC KHALT /AC = 740040
03223 151111 DZM 11111 /ADDR 1111 OR 01111
03224 211111 LAC 11111
03225 740200 SZA
03226 740040 F445 HALT /ERROR. DZM FAILED AT 1111
/OR 01111

/
03227 207647 LAC KHALT /AC = 740040
03230 152222 DZM 12222
03231 212222 LAC 12222
03232 740200 SZA
03233 740040 F446 HALT /ERROR. DZM FAILED AT 1222 OR
/02222

/
03234 207647 LAC KHALT /AC = 740040
03235 153333 DZM 13333
03236 213333 LAC 13333
03237 740200 SZA
03240 740040 F447 HALT /ERROR. DZM FAILED AT 13333 OR
/03333

/
03241 207647 LAC KHALT /AC = 740040
03242 154444 DZM 14444
03243 214444 LAC 14444
03244 740200 SZA
03245 740040 F448 HALT /ERROR. DZM FAILED AT 14444
/OR 04444

/
03246 207647 LAC KHALT /AC = 740040
03247 155555 DZM 15555
03250 215555 LAC 15555
03251 740200 SZA
03252 740040 F449 HALT /ERROR. DZM FAILED AT 15555
/OR 05555

/
03253 207647 LAC KHALT /AC = 740040
03254 156666 DZM 16666
03255 216666 LAC 16666
03256 740200 SZA
03257 740040 F450 HALT /ERROR. DZM FAILED AT 16666
/OR 06666

/
03260 207647 LAC KHALT /AC = 740040
03261 157777 DZM 17777
03262 217777 LAC 17777
03263 740200 SZA
03264 740040 F451 HALT /ERROR. DZM FAILED AT 17777
/OR 07777

```

.EJECT

03265 207647
 03266 152525
 03267 212525
 03270 740200
 03271 740040

/
 LAC KHALT /AC = 740040
 DZM 12525
 LAC 12525
 SZA
 F452 HALT /ERROR. DZM FAILED AT 12525
 /OR 02525

03272 207647
 03273 152525
 03274 212525
 03275 740200
 03276 740040

/
 LAC KHALT /AC = 740040
 DZM 15252
 LAC 15252
 SZA
 E453 HALT /ERROR. DZM FAILED AT 15252 OR 05252

/TEST AC AFTER A DZM

03277 207572
 03300 157777
 03301 740001
 03302 740200
 03303 740040

/
 LAC K7S /AC = 777777
 DZM 17777
 CMA
 SZA
 F454 HALT /ERROR. AC CHANGED AFTER A DZM

//TEST AC, LINK, ADR. 17777 OR 07777 AFTER A DZM

03304 754001
 03305 307572
 03306 157777
 03307 740001
 03310 740200
 03311 740040
 03312 741400
 03313 740040
 03314 217777
 03315 740200
 03316 740040

/
 CLA!CMA!CLL /AC = 1'S, LINK = 0
 ADD K7S
 DZM 17777
 CMA
 SZA
 F455 HALT /ERROR. AC NOT 1'S AFTER A DZM
 SZL
 F456 HALT /ERROR. LINK NOT 0
 LAC 17777
 SZA
 F457 HALT /ERROR. DZM FAILED AT 177777 OR 07777

/SEQUENTIAL DZM

03317 207572
 03320 152525
 03321 152525
 03322 157777
 03323 150000
 03324 740001
 03325 750200
 03326 740040

/
 LAC K7S /AC = 1'S
 DZM 12525
 DZM 15252
 DZM 17777
 DZM 10000
 CMA /AC = 0
 CLA!SZA
 F458 HALT /ERROR. AC NOT 1'S AFTER
 /DZM SERIES

.EJECT

03327	352525		TAD 12525	
03330	355252		TAD 15252	
03331	357777		TAD 17777	
03332	350000		TAD 10000	
03333	740200		SKP	
03334	740040	F459	HALT	/ERROR. DZM FAILED
		/		
03335	447633		ISZ WORK3	/CHECK DONE LOOPING
03336	603222		JMP DZMAC	/LOOP
03337	106102		JMS GENRAN	/GET NO. FOR NEXT LOOP
03340	106126		JMS CKNO	
		/		
		/		
		/TEST DAC		
		/		
03341	207573	DACAC	LAC K51S	/AC = 11111
03342	051111		DAC 11111	
03343	551111		SAD 11111	
03344	741000		SKP	
03345	740040	E460	HALT	/ERROR. DAC ADR CONTENTS NOT EQUAL /TO AC. DAC FAILED
03346	207574		LAC K12S	/AC = 12222
03347	052222		DAC 12222	
03350	552222		SAD 12222	
03351	741000		SKP	
03352	740040	F461	HALT	/ERROR. 122222 OR 022222 CONTENTS /NOT = TO AC. DAC FAILED
03353	207575		LAC K13S	/AC = 13333
03354	053333		DAC 13333	
03355	553333		SAD 13333	
03356	741000		SKP	
03357	740040	F462	HALT	/ERROR. 13333 OR 03333 CONTENTS /NOT = TO AC DAC FAILED
03360	207576		LAC K14S	/AC = 14444
03361	054444		DAC 14444	
03362	554444		SAD 14444	
03363	741000		SKP	
03364	740040	F463	HALT	/ERROR. 14444 OR 03333 CONTENTS /NOT = TO AC. DAC FAILED
03365	207577		LAC K15S	/AC = 15555
03366	055555		DAC 15555	
03367	555555		SAD 15555	
03370	741000		SKP	
03371	740040	F464	HALT	/ERROR. 15555 OR 05555 CONTENTS /NO = AC. DAC FAILED
			.EJECT	

03372	207600		LAC K16S	/AC = 166666
03373	056666		DAC 16666	
03374	556666		SAD 16666	
03375	741000		SKP	
03376	740040	E465	HALT	/ERROR. 16666 OR 06666 CONTENTS /NOT = AC. DAC FAILED /AC = 17777
03377	207601		LAC K17S	
03400	057777		DAC 17777	
03401	557777		SAD 17777	
03402	741000		SKP	
03403	740040	E466	HALT	/ERROR. 17777 OR 07777 CONTENTS /NOT = AC. DAC FAILED /AC = 252525
03404	207604		LAC K010	
03405	052525		DAC 12525	
03406	552525		SAD 12525	
03407	741000		SKP	
03410	740040	E467	HALT	/ERROR. 12525 OR 02525 CONTENTS /AC = AC. DAC FAILED /AC = 525252
03411	207605		LAC K111	
03412	055252		DAC 15252	
03413	555252		SAD 15252	
03414	741000		SKP	
03415	740040	E468	HALT	/ERROR. 15252 OR 05252 CONTENTS /NOT = AC. DAC FAILED
		/		
		/		
		/	SEQUENTIAL DAC	
		/		
03416	744000		CLL	/L = 0
03417	207572		LAC K7S	/AC = 1'S
03420	052525		DAC 12525	
03421	055252		DAC 15252	
03422	057777		DAC 17777	
03423	051000		DAC 11000	
03424	051111		DAC 11111	
03425	740001		CMA	/AC = 0
03426	750200		CLA!SZA	
03427	740040	E469	HALT	/ERROR. AC NOT 1'S AFTER DAC SERIES
03430	312525		ADD 12525	/ONES
03431	315252		ADD 15252	/ONES
03432	317777		ADD 17777	/ONES
03433	311000		ADD 11000	/ONES
03434	311111		ADD 11111	/ONES
03435	740001		CMA	/AC = 0
03436	740200		SZA	
03437	740040	E470	HALT	/ERROR. DAC FAILED. ONE OR MORE /ADDRESSES NOT ONES
		/		
03440	447633		ISZ WORK3	/CHECK DONE LOOPING
03441	603341		JMP DACAC	/LOOP
03442	106102		JMS GENRAN	/GET NO. FOR NEXT LOOP
03443	106126		JMS CKNO	
			.EJECT	

```

/TEST ISZ
/
ISZAC   LAC K3           /AC = 0
        DAC 10100
        ISZ 10100
        SKP!CLA!CMA
F471    HALT             /ERROR. ISZ SKIPPED
        AND K1           /AC = 1
        SAD 10100
        SKP
F472    HALT             /ERROR. 10100 OR 00100 NOT 1
        /ISZ FAILED
        /AC = 377777
        LAC M400K
        DAC 10100
        ISZ 10100
        SKP!CLA!CMA
E473    HALT             /ERROR. ISZ SKIPPED
        AND K400K
        SAD 10100
        SKP
E474    HALT             /ERROR. 10100 OR 00100 NOT 400000
        /ISZ FAILED
        /AC = 777776
        LAC M1
        DAC 10100
        ISZ 10100
        SKP!CLA!CMA
E475    HALT             /ERROR. ISZ SKIPPED
        AND K7S
        SAD 10100
        SKP
E476    HALT             /ERROR. 10100 OR 00100 NOT 777777
        /ISZ FAILED
        /AC = 777776
        LAC M1
        DAC 17777
        ISZ 17777
        SKP!CLA!CMA
F477    HALT             /ERROR. ISZ SKIPPED
        AND K7S
        SAD 17777
        SKP
F478    HALT             /ERROR. 1777 OR 07777 NOT 777777
        /ISZ FAILED
        /AC = 377777
        LAC M400K
        DAC 17777
        ISZ 17777
        SKP!CLA!CMA
E479    HALT             /ERROR. ISZ SKIPPED
        AND K400K
        SAD 17777
        SKP
        .EJECT

```

```

03444  207530
03445  050100
03446  450100
03447  751001
03450  740040
03451  507531
03452  550100
03453  741000
03454  740040
03455  207626
03456  050100
03457  450100
03460  751001
03461  740040
03462  507553
03463  550100
03464  741000
03465  740040
03466  207620
03467  050100
03470  450100
03471  751001
03472  740040
03473  507572
03474  550100
03475  741000
03476  740040
03477  207620
03500  057777
03501  457777
03502  751001
03503  740040
03504  507572
03505  557777
03506  741000
03507  740040
03510  207626
03511  057777
03512  457777
03513  751001
03514  740040
03515  507553
03516  557777
03517  741000

```

03520	740040	F480	HALT	/ERROR. 17777 OR 0777 NOT 400000 /ISZ FAILED
03521	207530		LAC K0	
03522	057777		DAC 17777	
03523	457777		ISZ 17777	
03524	751001		SKP!CL!CMA	
03525	740040	F481	HALT	/ERROR. ISZ SKIPPED
03526	507531		AND K1	
03527	557777		SAB 17777	
03530	741000		SKP	
03531	740040	F482	HALT	/ERROR. 17777 OR 07777 NOT 1 /ISZ FAILED
03532	750000		CLA	/AC = 0
03533	247572		XOR K7S	/AC = 1'S
03534	051111		DAC 11111	
03535	451111		ISZ 11111	
03536	740040	F483	HALT	/ERROR. ISZ FAILED TO SKIP
03537	211111		LAC 11111	
03540	740200		SZA	
03541	740040	F484	HALT	/ERROR. 11111 OR 01111 NOT 0 /ISZ FAILED
03542	750000		CLA	/AC = 0
03543	247572		XOR K7S	/AC = 1'S
03544	052222		DAC 12222	
03545	452222		ISZ 12222	
03546	740040	F485	HALT	/ERROR. ISZ FAILED TO SKIP
03547	212222		LAC 12222	
03550	740200		SZA	
03551	740040	F486	HALT	/ERROR. 12222 OR 02222 NOT 0 /ISZ FAILED
03552	750000		CLA	/AC = 1'S
03553	247572		XOR K7S	
03554	053333		DAC 13333	
03555	453333		ISZ 13333	
03556	740040	F487	HALT	/ERROR. ISZ DID NOT SKIP
03557	213333		LAC 13333	
03560	740200		SZA	
03561	740040	F488	HALT	/ERROR. 13333 OR 03333 NOT 0 /ISZ FAILED
03562	750000		CLA	/AC = 1'S
03563	247572		XOR K7S	
03564	054444		DAC 14444	
03565	454444		ISZ 14444	
03566	740040	F489	HALT	/ERROR. ISZ DID NOT SKIP
03567	214444		LAC 14444	
03570	740200		SZA	
03571	740040	F490	HALT	/ERROR. 14444 OR 04444 NOT 0 /ISZ FAILED

.EJECT

03572	750000		CLA	
03573	247572		XOR K7S	/AC = 1'S
03574	055555		DAC 15555	
03575	455555		ISZ 15555	
03576	740040	F491	HALT	/ERROR. ISZ DID NOT SKIP
03577	215555		LAC 15555	
03600	740200		SZA	
03601	740040	F492	HALT	/ERROR. 15555 OR 05555 NOT 0 /ISZ FAILED
03602	750000		CLA	
03603	247572		XOR K7S	/AC = 1'S
03604	056666		DAC 16666	
03605	456666		ISZ 16666	
03606	740040	F493	HALT	/ERROR. ISZ DID NOT SKIP
03607	216666		LAC 16666	
03610	740200		SZA	
03611	740040	F494	HALT	/ERROR. 16666 OR 06666 NOT 0 /ISZ FAILED
03612	750000		CLA	
03613	247572		XOR K7S	/AC = 1'S
03614	057777		DAC 17777	
03615	457777		ISZ 17777	
03616	740040	E495	HALT	/ERROR. ISZ DID NOT SKIP
03617	217777		LAC 17777	
03620	740200		SZA	
03621	740040	F496	HALT	/ERROR. 17777 OR 07777 NOT 0 /ISZ FAILED
03622	750000		CLA	
03623	247572		XOR K7S	/AC = 1'S
03624	052525		DAC 12525	
03625	452525		ISZ 12525	
03626	740040	E497	HALT	/ERROR. ISZ DID NOT SKIP
03627	212525		LAC 12525	
03630	740200		SZA	
03631	740040	F498	HALT	/ERROR. 12525 OR 02525 NOT 0 /ISZ FAILED
03632	750000		CLA	
03633	247572		XOR K7S	/AC = 1'S
03634	055252		DAC 15252	
03635	455252		ISZ 15252	
03636	740040	F499	HALT	/ERROR. ISZ DID NOT SKIP
03637	215252		LAC 15252	
03640	740200		SZA	
03641	740040	F500	HALT	/ERROR. 15252 OR 05252 NOT 0 /ISZ FAILED

.EJECT

/TEST ISZ, SKP

03642 207572
 03643 052525
 03644 355252
 03645 057777
 03646 051000
 03647 050100
 03650 452525
 03651 741000
 03652 455252
 03653 741000
 03654 457777
 03655 741000
 03656 451000
 03657 741000
 03660 450100
 03661 740040
 03662 312525
 03663 315252
 03664 317777
 03665 311000
 03666 310100
 03667 740000
 03670 740200
 03671 740040

/
 LAC K75
 DAC 12525
 DAC 15252
 DAC 17777
 DAC 11000
 DAC 10100
 ISZ 12525
 SKP
 ISZ 15252
 SKP
 ISZ 17777
 SKP
 ISZ 11000
 SKP
 ISZ 10100
 F501 HALT
 ADD 12525
 ADD 15252
 ADD 17777
 ADD 11000
 ADD 10100
 CMA
 SZA
 F502 HALT

/AC = 1'S
 /12525 OR 02525
 /15252 OR 05252
 /17777 OR 07777
 /11000 OR 01000
 /10100 OR 00100

/ERROR. ISZ DID NOT SKIP

/ERROR. ALL ADRS. NOT 0

/SEQUENTIAL ISZ, NO-SKIP

03672 207626
 03673 052525
 03674 055252
 03675 057777
 03676 051000
 03677 050100
 03700 452525
 03701 455252
 03702 457777
 03703 451000
 03704 450100
 03705 750000
 03706 312525
 03707 315252
 03710 317777
 03711 311000
 03712 310100
 03713 247554
 03714 740200
 03715 740040

/
 LAC M400K
 DAC 12525
 DAC 15252
 DAC 17777
 DAC 11000
 DAC 10100
 ISZ 12525
 ISZ 15252
 ISZ 17777
 ISZ 11000
 ISZ 10100
 CLA
 ADD 12525
 ADD 15252
 ADD 17777
 ADD 11000
 ADD 10100
 XOR K402K
 SZA
 F503 HALT
 .EJECT

/AC = 377777
 /OR 02525
 /OR 05252
 /OR 07777
 /OR 01000
 /OR 00100

/AC = 0

/RESULT = 400002

/ERROR. ALL ADRES. NOT 400000

		/TEST ISZ-SKP, SKIP	
		/	
03716	207572	LAC K78	/AC = 1'S
03717	055252	DAC 15252	
03720	455252	ISZ 15252	
03721	741000	SKP	
03722	741000	SKP	
03723	740040	F504 HALT	/ERROR. ISZ-SKP DID NOT SKIP
		/	
		/TEST SKP-ISZ, SKIP	
		/	
03724	207572	LAC K79	/AC = 17S
03725	055252	DAC 15252	
03726	741000	SKP	
03727	740000	NOB	
03730	455252	ISZ 15252	
03731	740040	F505 HALT	/ERROR. SKP-ISZ DID NOT SKIP
		/	
		/TEST SKP-ISZ, NO-SKIP	
		/	
03732	207530	LAC K0	/AC = 0
03733	055252	DAC 15252	
03734	741000	SKP	
03735	740000	NOB	
03736	455252	ISZ 15252	
03737	741000	SKP	
03740	740040	F506 HALT	/ERROR. SKP-ISZ SKIPPED
		/	
03741	447633	ISZ WORK3	/CHECK DONE LOOPING
03742	603444	JMP ISZAC	/LOOP
03743	106102	JMS GENRAN	/GET NO. FOR NEXT LOOP
03744	106126	JMS CKNO	
		.EJECT	

```

/TFST JMP
/
03745 207666 LAC JMPRET
03746 740200 SZA
03747 740040 F507 HALT /ERROR JMP .-7, .+4 OR .+5 FAILED
03750 204130 INIT4K LAC JMPSEQ /LOAD 4K WITH JMP TO 70
03751 047666 DAC JMPRET /SEE IF IN UPPER 4K NOW
03752 203752 LAC .
03753 507555 AND K1 K /IN UPPER 4K
03754 740200 SZA /LOWER 4K
03755 603761 JMP .+4
03756 207555 LAC K1 K
03757 047627 DAC RJCNT
03760 603765 JMP .+5
03761 207540 LAC K2 K
03762 047627 DAC RJCNT
03763 741000 SKP
03764 147666 DZM JMPRET /CLEAR ERROR TABLE
03765 204130 LAC JMPSEQ
03766 067627 DAC RJCNT /STORE JMP 70
03767 447627 ISZ RJCNT /INCR. ADDRESS
03770 207627 LAC RJCNT
03771 546061 SAA K1777 /WILL = SAA 07777 WHEN IN
03772 741000 SKP / UPPER 4K
03773 603764 JMP .-7
03774 207666 LAC JMPRET
03775 740200 SZA
03776 740040 F508 HALT /ERROR. JMPP.+4, .+5 OR .-7 FAILED
03777 204131 LAC MOD /PRESS CONTINUE TO DETERMINE
04000 043745 DAC F507-2 /JMP FAILURE
04001 740000 MODX NOP
/
04002 207667 LAC J111
04003 740200 SZA
04004 740040 F509 HALT /ERROR. RJMP OR JMP TO 11111
/OR 01111 FAILED
04005 204132 LAC RJ111
04006 047667 DAC J111 /STORE JMP ADDRESS IN TABLE
04007 051111 DAC 11111
04010 611111 JMP 11111 /JMP TO 11111 OR 01111
04011 741000 SKP
04012 147667 RJMP1 DZM J111 /CLEAR ERROR WORD TABLE
04013 207670 LAC J222
04014 740200 SZA
04015 740040 F510 HALT /ERROR. RJMP OR JMP TO 12222
/OR 02222 FAILED
04016 204133 LAC RJ222
04017 047670 DAC J222
04020 052222 DAC 12222
04021 612222 JMP 12222 /JMP 1222 OR 02222
04022 741000 SKP
04023 147670 RJMP2 DZM J222 /CLEAR ERROR TABLE
.EJECT

```


04024	207671		LAC J333	
04025	740200		SZA	
04026	740040	F511	HALT	/ERROR. RJMP OR JMP TO 13333 /OR 03333 FAILED
04027	204134		LAC RJ333	
04030	047671		DAC J333	
04031	053333		DAC 13333	
04032	613333		JMP 13333	/JMP TO 13333 OR 03333
04033	741000		SKP	
04034	147671	RJMP3 /	DZM J333	/CLEAR ERROR TABLE
04035	207672		LAC J444	
04036	740200		SZA	
04037	740040	F512	HALT	/ERROR. RJMP OR JMP TO 14444 /OR 04444 FAILED
04040	204135		LAC RJ444	
04041	047672		DAC J444	
04042	054444		DAC 14444	
04043	614444		JMP 14444	/JMP TO 14444 OR 04444
04044	741000		SKP	
04045	147672	RJMP4 /	DZM J444	/CLEAR ERROR TABLE
04046	207673		LAC J555	
04047	740200		SZA	
04050	740040	E513	HALT	/ERROR. RJMP OR JMP TO 15555 /OR 05555 FAILED
04051	204136		LAC RJ555	
04052	047673		DAC J555	
04053	055555		DAC 15555	
04054	615555		JMP 15555	/JMP TO 15555 OR 05555
04055	741000		SKP	
04056	147673	RJMP5 /	DZM J555	/CLEAR ERROR TABLE
04057	207674		LAC J666	
04060	740200		SZA	
04061	740040	F514	HALT	/ERROR. RJMP OR JMP TO 16666 /OR 06666 FAILED
04062	204137		LAC RJ666	
04063	047674		DAC J666	
04064	056666		DAC 16666	
04065	616666		JMP 16666	/JMP TO 16666 OR 06666
04066	741000		SKP	
04067	147674	RJMP6 /	DZM J666	/CLEAR ERROR TABLE

.EJECT

```

04070 207675 LAC J777
04071 740200 SZA
04072 740040 E515 HALT /ERROR, RJMP OR JMP TO 17777
/ OR 07777 FAILED

04073 204140 LAC RJ777
04074 047675 DAC J777
04075 057777 DAC 17777
04076 617777 JMP 17777 /JMP TO 17777 OR 07777
04077 741000 SKP
04100 147675 RJMP7 DZM J777 /CLEAR ERROR TABLE
/

04101 207677 LAC J252
04102 740200 SZA
04103 740040 F516 HALT /ERROR, RJMP OR MP TO 12525
/ OR 02525 FAILED

04104 204141 LAC RJ252
04105 047677 DAC J252
04106 052525 DAC 12525
04107 612525 JMP 12525 /JMP TO 12525 OR 02525
04110 741000 SKP
04111 147677 RJMP8 DZM J252 /CLEAR ERROR TABLE
/

04112 207676 LAC J525
04113 740200 SZA
04114 740040 F517 HALT /ERROR, RJMP OR JMP TO 15252
/ OR 05252 FAILED

04115 204142 LAC RJ525
04116 047676 DAC J525
04117 055252 DAC 15252
04120 615252 JMP 15252 /JMP TO 15252 OR 05252
04121 741000 SKP
04122 147676 RJMP9 DZM J525 /CLEAR ERROR TABLE
/

04123 447633 ISZ WORK3 /CHECK DONE LOOPING
04124 604002 JMP MOIX+1 /LOOP
04125 106102 JMS GENRAN /GET NO. FOR NEXT LOOP
04126 106126 JMS CKNO
04127 604143 JMP TSCAL /TEST CAL
/

/JMP CONSTANTS, THESE ARE MODIFIED WHEN IN HI 4K
/
04130 600070 JMPS:0 JMP SEQUEN
04131 604002 MO00 JMP MOIX+1
04132 604012 RJ111 JMP RJMP1
04133 604023 RJ222 JMP RJMP2
04134 604034 RJ333 JMP RJMP3
04135 604045 RJ444 JMP RJMP4
04136 604056 RJ555 JMP RJMP5
04137 604067 RJ666 JMP RJMP6
04140 604100 RJ777 JMP RJMP7
04141 604111 PJ252 JMP RJMP8
04142 604122 RJ525 JMP RJMP9
.EJECT

```

```

/TFST CAL
/
04143 207700 TSCAL IAC CAL0
04144 740200 SZA
04145 740040 F518 HALT /ERROR. CAL FROM 17757 OR 07757
04146 707704 IEM /CLEAR EXTEND MODE
04147 754000 CLA:CLI /AC = 0, LINK = 0
04150 770020 LAW 10020 /AC = 770020
04151 150704 DZM 10704 /CAL AT 10704 OR 00704
04152 207562 LAC K2021
04153 047700 DAC CAL0 /STORE ERROR CODE 2021
04154 204215 LAC RCAL0
04155 040021 DAC 21 /RJMP FROM CAL
04156 610704 JMP 10704 /JMP TO 10704 OR 00704
04157 147700 RCALS0 DZM CAL0 /CLEAR ERROR TABLE
04160 200020 LAC 20
04161 544216 SAD KCAL0 /10705 OR 00705
04162 741000 SKP
04163 740040 E519 HALT /ERROR. (20) NOT 10705 OR 00705
04164 210704 LAC 10704
04165 740200 SZA
04166 740040 E519A HALT /ERROR (10704 OR 00704) NOT 0
/
/TFST CAL LINK = 1
/
04167 207701 LAC CAL1
04170 740200 SZA
04171 740040 E520 HALT /ERROR. CAL FROM 10704 OR 00704
04172 744002 CLL:CLI /LINK = 1
04173 207563 LAC K2120
04174 047701 DAC CAL1 /STORE ERROR CODE 2120
04175 204217 LAC RCAL1
04176 040021 DAC 21 /RJMP FROM CAL
04177 610704 JMP 10704 /JMP TO 10704 OR 00704
04200 147701 RCALS1 DZM CAL1 /CLEAR ERROR TABLE
04201 200020 LAC 20
04202 544220 SAD KCALF /410705 OR 400705
04203 741000 SKP
04204 740040 F521 HALT /ERROR. (20) NOT 410705 OR 400705
04205 210704 LAC 10704
04206 740200 SZA
04207 740040 F521A HALT /ERROR (10704 OR 00704) NOT 0
/
04210 447633 ISZ WOPK3 /CHECK DONE LOOPING
04211 604143 JMP TSCAL /LOOP
04212 106102 JMS GENRAN /GET NO. FOR NEXT LOOP
04213 106126 JMS CKNO
04214 604221 JMP TSJMS /TEST JMS
/
.EJECT

```

04215 604157
04216 010705
04217 604200
04220 410705

/CAL CONSTANTS. THESE ARE MODIFIED WHEN IN HI 4K
/
RCAL0 JMP RCALS0
RCAL1 10705
RCAL1 JMP RCALS1
RCAL2 410705
.EJECT

```

/TEST JMS
/
04221 207700 TSJMS LAC JSM71
04222 740200 SZA
04223 740040 E522 HALT /ERROR. JMS FROM 07777 TO 11111
/OR FROM 17777 TO 01111
/LINK = 0
04224 744000 CLL /JMP TO RJMS71
04225 204537 LAC RJSM71
04226 051112 DAC 11112 /RJMP FROM JMS DEST'N
04227 204540 LAC RSM71 /JMS 11111
04230 047777 DAC 07777
04231 047702 DAC JSM71
04232 771112 LAW 11112 /AC = 771112
04233 607777 JMP 07777
04234 147702 RJMS71 DZM JSM71 /CLEAR ERROR TABLE
04235 211111 LAC 11111
04236 544541 SAK K10000
04237 741000 SKP
04240 740040 E523 HALT /ERROR. (11111 OR 01111) NOT
/10000 OR 00000
04241 207703 LAC JSM72
04242 740200 SZA
04243 740040 E524 HALT /ERROR. JMS FROM 07776 TO 12222
/OR FROM 17776 TO 02222
04244 707704 LEM
04245 744000 CLL
04246 204542 LAC RJSM72 /JMP TO RJMS72
04247 052223 DAC 12223
04250 204543 LAC RSM72 /JMS 12222
04251 047776 DAC 07776
04252 047703 DAC JSM72
04253 772223 LAW 12223 /AC = 772223
04254 607776 JMP 07776
04255 147703 RJMS72 DZM JSM72 /CLEAR ERROR TABLE
04256 212222 LAC 12222
04257 544544 SAK K77
04260 741000 SKP
04261 740040 F525 HALT /ERROR. (12222 OR 02222) NOT
/07777 OR 17777
04262 207704 LAC JSM73
04263 740200 SZA
04264 740040 E526 HALT /ERROR. JMS FROM 07775 TO 13333
/OR FROM 17775 TO 03333
04265 707704 LEM
04266 744000 CLL
04267 204545 LAC RJSM73 /JMP TO RJMS73
04270 053334 DAC 13334
04271 204546 LAC RSM73 /JMS 13333
04272 047775 DAC 07775
04273 047704 DAC JSM73
04274 773334 LAW 13334 /AC = 773334
04275 607775 JMP 07775
.EJECT

```

04276	147704	RJMS73	OZM JS473	/CLEAR ERROR TABLE
04277	213333		LAC 13333	
04300	544547		SAD K7-	
04301	741000		SKP	
04302	740040	F527	HALT	/ERROR. (13333 OR 13333) NOT /07776 OR 17776
04303	207705		LAC JS474	
04304	740000		SZA	
04305	740040	F528	HALT	/ERROR. JMS FROM 07774 TO 14444 /OR FROM 17774 TO 04444
04306	707704		LEM	
04307	744000		CLL	
04310	204550		LAC RJMS74	/JMP RJMS74
04311	054445		DAC 14445	
04312	204551		LAC RS474	/JMS 14444
04313	047774		DAC 07774	
04314	047705		DAC JS474	
04315	774445		LAW 14445	/AC = 774445
04316	607774		JMP 07774	
04317	147705	RJMS74	OZM JS474	/CLEAR ERROR TABLE
04320	214444		LAC 14444	
04321	544552		SAD K7-	
04322	741000		SKP	
04323	740040	F529	HALT	/ERROR. (14444 OR 04444) NOT /07775 OR 17775
04324	207706		LAC JS475	
04325	740000		SZA	
04326	740040	F530	HALT	/ERROR. JMS FROM 07773 TO 15555 /OR FROM 17773 TO 05555
04327	707704		LEM	
04330	744000		CLL	
04331	204553		LAC RJMS75	/JMP TO RJMS75
04332	055556		DAC 15556	
04333	204554		LAC RS475	/JMS 15555
04334	047773		DAC 07773	
04335	047706		DAC JS475	
04336	775556		LAW 15556	/AC = 775556
04337	607773		JMP 07773	
04340	147706	RJMS75	OZM JS475	/CLEAR ERROR TABLE
04341	215555		LAC 15555	
04342	544555		SAD K7-	
04343	741000		SKP	
04344	740040	E531	HALT	/ERROR. (15555 OR 05555) NOT /07774 OR 17774
04345	207707		LAC JS476	
04346	740000		SZA	
04347	740040	F532	HALT	/ERROR. JMS FROM 07772 TO 16666 /OR 17772 TO 06666

.EJECT

04350	707704		LFM	
04351	744000		CLI	
04352	204556		LAC RJSM76	/JMP TO RJMS76
04353	056667		DAC 16667	
04354	204557		LAC RSM76	/JMS 16666
04355	047772		DAC 07772	
04356	047707		DAC JSM76	
04357	776667		LAW 16667	/AC = 776667
04360	607772		JMP 07772	
04361	147707	RJMS76	DZM JS76	/CLEAR ERROR TABLE
04362	216666		LAC 16666	
04363	544560		SAD K72	
04364	741000		SKP	
04365	740040	F533	HALT	/ERROR. (16666 OR 06666) NOT /07773 OR 17773
04366	207710		LAC JSM77	
04367	740200		SZA	
04370	740040	F534	HALT	/ERROR.JMS FROM 07771 TO 17777 /OR 17771 TO 07777
04371	707704		LFM	
04372	744000		CLI	
04373	700004		CLOF	
04374	700002		IOF	/PI OFF FOR THIS TEST
04375	204561		LAC RJSM77	/JMP TO RJMS77
04376	040000		DAC 0000	
04377	204562		LAC RSM77	/JMS 17777
04400	047771		DAC 07771	
04401	047710		DAC JSM77	
04402	760000		LAW 0.	/AC = 760000
04403	607771		JMP 07771	
04404	147710	RJMS77	DZM JSM77	/CLEAR ERROR TABLE
04405	217777		LAC 17777	
04406	544563		SAD K72	
04407	741000		SKP	
04410	740040	E535	HALT	/ERROR (17777 OR 07777) NOT /07772 OR 17772
04411	750004		LAS	
04412	740010		RAI	
04413	740100		SMA	
04414	700042		ION	/PI BACK ON
04415	507556		AND K24K	
04416	741200		SNA	/CHECK ACS 5
04417	106604		JMS SETCLK	/CLOCK BACK ON
04420	207711		LAC JS252	
04421	740200		SZA	
04422	740040	E536	HALT	/ERROR. JMS FROM 12525 TO 15252 /OR FROM 02525 TO 05252

.EJECT

04423	744002		CLL!CMI	/LINK = 1
04424	204564		LAC RJSM25	/JMP TO RJMS14
04425	055253		DAC 15253	
04426	204565		LAC RSM25	/JMS 15252
04427	052525		DAC 12525	
04430	047711		DAC JS252	
04431	775253		LAW 15253	/AC = 775253
04432	612525		JMP 12525	
04433	147711	RJMS14	DZM JS252	/CLEAR ERROR TABLE
04434	215252		LAC 15252	
04435	544566		SAD K426	/412526 OR 402526
04436	741000		SKP	
04437	740040	E537	HALT	/ERROR. (15252 OR 05252) NOT /412526 OR 402526
04440	207712		LAC JS252	
04441	740200		SZA	
04442	740040	F538	HALT	/ERROR. JMS FROM 15252 TO 12525 /OR 05252 TO 02525
04443	744002		CLL!CMI	/LINK = 1
04444	204567		LAC RJSM52	/RJMP TO RJMS15
04445	052526		DAC 12526	
04446	204570		LAC RSM52	/JMS 12525
04447	055252		DAC 15252	
04450	047712		DAC JS252	
04451	772526		LAW 12526	/AC = 772526
04452	615252		JMP 15252	
04453	147712	RJMS15	DZM JS252	/CLEAR ERROR TABLE
04454	212525		LAC 12525	
04455	544572		SAD K415	/415253 OR 405253
04456	741000		SKP	
04457	740040	E539	HALT	/ERROR. (12525 OR 02525) NOT /415253 OR 405253
/PDP-9 BASIC EXERCISER - TAPF 5				
/TFST JMS SERIES				
/				
04460	207713		LAC JSSS	
04461	740200		SZA	
04462	740040	E540	HALT	/ERROR. JMS SERIES FAILED
/				
04463	744002		CLL!CMI	/LINK = 1
04464	104465	JS1	JMS .+1	
04465	740040	F541	HALT	/ERROR. JMS SERIES
04466	104467	JS2	JMS .+1	
04467	740040	F542	HALT	/ERROR. JMS SERIES
04470	104471	JS3	JMS .+1	
04471	740040	F543	HALT	/ERROR. JMS SERIES
04472	707704		LEM	
04473	744000		CLL	
04474	104475	JS4	JMS .+1	
04475	740040	F544	HALT	/ERROR. JMS SERIES
04476	147713	RJMSS	DZM JSSS	/CLEAR ERROR TABLE
04477	204573		LAC KJS1	/TEST JS1, LINK = 1
04500	347553		TAD K400K	
04501	740001		CMA	
04502	344465		TAD JS1+1	

04503	740001		CMA	
04504	740200		SZA	
04505	740040	E545	HALT	/ERROR. JS1+1
04506	204574		LAC KJS2	/TEST JS2, LINN = 1
04507	347553		TAD K400K	
04510	740001		CMA	
04511	344467		TAD JS2+1	
04512	740001		CMA	
04513	740200		SZA	
04514	740040	F546	HALT	/ERROR. JS2+1
04515	204575		LAC KJS3	/TEST JS3, LINK = 1
04516	347553		TAD K400K	
04517	740001		CMA	
04520	344471		TAD JS3+1	
04521	740001		CMA	
04522	740200		SZA	
04523	740040	F547	HALT	/ERROR. JS3+1
04524	204576		LAC KJS4	/TEST JS4, EXT = 0, LINK = 0
04525	740001		CMA	
04526	344475		TAD JS4+1	
04527	740001		CMA	
04530	740200		SZA	
04531	740040	F548	HALT	/ERROR. JS4+1
		/		
04532	447633		IS7 WORK3	/CHECK DONE LOOPING
04533	604221		JMP TSJMS	/LOOP
04534	106102		JMS GENRAN	/GET NO. FOR NEXT LOOP
04535	106126		JMS CKNO	
04536	604577		JMP TSXCT	/TEST XCT
			.EJECT	

/CONSTANTS FOR JMS. MODIFIED WHEN IN HI 4K

04537	604234	RJSM71	JMP RJMS71
04540	111111	RSM71	JMS 11111
04541	010000	K10000	10000
04542	604255	RJSM72	JMP RJMS72
04543	112222	RSM72	JMS 12222
04544	007777	K77	07777
04545	604276	RJSM73	JMP RJMS73
04546	113333	RSM73	JMS 13333
04547	007776	K76	07776
04550	604317	RJSM74	JMP RJMS74
04551	114444	RSM74	JMS 14444
04552	007775	K75	07775
04553	604340	RJSM75	JMP RJMS75
04554	115555	RSM75	JMS 15555
04555	007774	K74	07774
04556	604361	RJSM76	JMP RJMS76
04557	116666	RSM76	JMS 16666
04560	007773	K73	07773
04561	604404	RJSM77	JMP RJMS77
04562	117777	RSM77	JMS 17777
04563	007772	K72	07772
04564	604433	RJSM75	JMP RJMS14
04565	115252	RSM25	JMS 15252
04566	412526	K426	412526
04567	604453	RJSM52	JMP RJMS15
04570	112525	RSM52	JMS 12525
04571	007771	K71	07771
04572	415253	K415	415253
04573	004465	KJS1	JS1+1
04574	004467	KJS2	JS2+1
04575	004471	KJS3	JS3+1
04576	004475	KJS4	JS4+1

/.
.EJECT

		/TEST XCT	
		/	
04577	754003	TSXCT	CLA!CMA!CLL!CMI /AC = ONES, LINK = 0
04600	404601		XCT .+1 /NOP
04601	740000		NOP
04602	740400		SNI
04603	740040	F549	HALT /ERROR; XCT NOP; LINK WAS RESET
04604	740001		CMA
04605	740200		SZA
04606	740040	F550	HALT /ERROR; XCT NOP; AC NOT ONES
		/	
		/TEST EXECUTE NOP, AC = 0, LINK = 0	
			CLA!CLL /AC = 0, LINK = 0
04607	754000		XCT .+1
04610	404611		NOP
04611	740000		SZI
04612	741400		SZA
04613	740040	F551	HALT /ERROR; XCT NOP; LINK WAS SET
04614	740200		SZA
04615	740040	E552	HALT /ERROR; XCT NOP; AC NOT 0
		/	
		/TEST XCT SKP	
			XCT KSKP /SKIP
04616	407616	F553	HALT /ERROR; XCT SKP FAILED
04617	740040		CLA!CMA /AC = ONES
04620	750001		XCT KCLA /CLA
04621	407617		SZA
04622	740200	E554	HALT /ERROR; XCT CLA FAILED
04623	740040		
		/	
		/TEST XCT LAW	
			CLA /AC = 0
04624	750000		XCT K7S /LAW = 17777
04625	407572		CMA /AC = 0
04626	740001		SZA
04627	740200	F555	HALT /ERROR; XCT LAW FAILED
04630	740040		
		/TEST XCT ISZ	
			CLA!CMA /AC = ONES
04631	750001		DAC 17777
04632	057777		XCT XCTISZ /ISZ 17777
04633	405052	F556	HALT /ERROR; XCT ISZ FAILED TO SKP
04634	740040		
		/	
		/TEST XCT TAD	
			CLL!CMI /LINK = 1
04635	744002		LAW 17777 /AC = ONES
04636	777777		DAC 17777 /17777=777777
04637	057777		XCT XCTTAD /TAD K1
04640	405054		SZA
04641	740200		HALT
04642	740040	E557	HALT /ERROR; XCT TAD FAILED, AC NOT 0
04643	741400		SZL
04644	740040	F558	HALT /ERROR; XCT TAD FAILED LINK
			.EJECT

04645	754003	/TEST XCT RAL, AC = ONFS, LINK = 1	
04646	407640	CLA!CMA!CLL!CML	/AC = ONFS, LINK = 1
04647	740001	XCT XCTRAL	/RAL
04650	740200	CMA	/AC = 0
04651	740040	SZA	
04652	744400	E559 HALT	/ERROR: XCT RAL FAILED AC DROPPED A BIT
04653	740040	SNL!CLL	
		E560 HALT	/ERROR: XCTRAL FAILED LINK DROPPED
		/	
		/TEST XCT DAC	
04654	207566	LAC K3S	/AC = 333333
04655	405053	XCT XCTDAC	/DAC 17777
04656	347567	TAD K4S	/AC = 777777
04657	740001	CMA	/AC = 0
04660	740200	SZA	
04661	740040	E561 HALT	/ERROR: XCT DAC FAILED, K3S
			/NOT STORED AT 17777
		.EJECT	

```

/TEST XCT JMS
/
04662 207714 LAC XCT11
04663 740200 SZ/
04664 740040 F562 HALT /ERROR, XCT (16666 OR 06666)
/ FROM 11111 OR 01111
/XCT (16666 OR 06666)
04665 205030 LAC XT11S
04666 047714 DAC XC11
04667 051111 DAC 11111 /OR 01111
04670 205031 LAC XT11 /JMS 11111 OR 01111
04671 056666 DAC 16666 /OR 06666
04672 205032 LAC XT1R /RJMP TO RXCT1
04673 051112 DAC 11112 /OR 01112
04674 611111 JMP 11111 /OR 01111 AND XCT (16666 OR 06666)
04675 147714 RXCT1 DZM XCT11 /CLEAR ERROR TABLE
04676 211111 LAC 11111 /OR 01111
04677 545033 SADR K12
04700 741000 SKP
04701 740040 F563 HALT /ERROR, RJMP ADR, NOT 1112
/OR 01112
04702 207715 LAC XCT12
04703 740200 SZ/
04704 740040 F564 HALT /ERROR, XCT (15555 OR 05555)
/ FROM 12222 OR 02222
/XCT (15555 OR 05555)
04705 205034 LAC XT12S
04706 047715 DAC XCT12
04707 052222 DAC 12222 /JMS 12222 OR 02222
04710 205035 LAC XT12 /OR 02222
04711 055555 DAC 15555 /RJMP TO RXCT2
04712 205036 LAC XT2R
04713 052223 DAC 12223
04714 612222 JMP 12222
04715 147715 RXCT2 DZM XCT12 /CLEAR ERROR TABLE
04716 212222 LAC 12222
04717 545037 SADR K23
04720 741000 SKP
04721 740040 F565 HALT /ERROR, RJMP NOT 12223
/OR 02223
04722 207716 LAC XCT13
04723 740200 SZ/
04724 740040 F566 HALT /ERROR, XCT (14444 OR 04444)
/ FROM 13333 OR 03333
.EJECT

```

04725	205040	LAC XT135	/XCT (14444 OR 04444)
04726	047716	DAC XCT13	
04727	053333	DAC 13333	
04730	205041	LAC XT-13	/JMS 13333 OR 03333
04731	054444	DAC 14444	
04732	205042	LAC XT-R	/RJMP TO RXCT3
04733	053334	DAC 13334	
04734	613333	JMP 13333	
04735	147716	DZM XCT13	/CLEAR ERROR TABLE
04736	213333	LAC 13333	
04737	545043	SAD K34	
04740	741000	SKP	
04741	740040	HALT	/ERROR. RJMP NOT 13334 OR 03334
04742	207717	LAC XCT17	
04743	740200	SZA	
04744	740040	HALT	/ERROR. XCT (17776 OR 07776) /FROM 07776 OR 17776 /XCT (17776 OR 07776)
04745	205044	LAC XT17S	
04746	047717	DAC XCT17	
04747	047776	DAC 07776	/OR 17776
04750	205045	LAC XTR17	/JMS 07776 OR 17776
04751	057776	DAC 17776	/OR 07776
04752	205046	LAC XT-R	/RJMP TO RXCT4
04753	047777	DAC 07777	/OR 17777
04754	607776	JMP 07776	/OR 17776
04755	147717	DZM XCT17	/CLEAR ERROR TABLE
04756	207776	LAC 07776	/FOR 17776
04757	544544	SAD K77	
04760	741000	SKP	
04761	740040	HALT	/ERROR. RJMP NOT 07777 OR 17777
04762	207720	LAC XCT125	
04763	740200	SZA	
04764	740040	HALT	/ERROR. XCT (12525 OR 02525) /FROM 15252 OR 05252
04765	205047	LAC XCT12S	
04766	047720	DAC XCT125	/XCT (12525 OR 02525)
04767	055252	DAC 15252	/OR 05252
04770	205050	LAC XCTR12	
04771	052525	DAC 12525	/JMS 15252 OR 05252
04772	205051	LAC XT-R	/RJMP TO RXCT5
04773	055253	DAC 15253	
04774	615252	JMP 15252	
04775	147720	DZM XCT125	/CLEAR ERROR TABLE
04776	215252	LAC 15252	
04777	545435	SAD K15253	
05000	741000	SKP	
05001	740040	HALT	/ERROR. RJMP NOT 15253 OR 05253
		.EJECT	

/TEST XCT SERIES

```

/
05002 754003 CLA!CMA!CLL!CML /AC = ONES, LINK = 1
05003 405004 XCT .+1; XCT .+1; XCT .+1
05004 405005
05005 405006
05006 405007 XCT .+1; XCT .+1; XCT .+1
05007 405010
05010 405011
05011 405012 XCT .+1; XCT .+1; XCT .+1
05012 405013
05013 405014
05014 405015 XCT .+1
05015 740000 NOP
05016 740001 CMA
05017 740000 SZA
05020 740040 E572 HALT /ERROR, XCT SERIES FAILED AC NOT ONES
05021 740000 SNL
05022 740040 E573 HALT /ERROR, LINK CHANGED
05023 447633 ISZ WORK3 /CHECK DONE LOOPING
05024 604577 JMP TSXCT /LOOP
05025 106102 JMS GENRAN /GET NO FOR NEXT LOOP
05026 106126 JMS CKNO
05027 605055 JMP TSAUTO /TEST AUTO-INDEXING

```

/XCT CONSTANTS, MODIFIED WHEN IN UPPER 4K

```

/
05030 416666 XT11S XCT 16666
05031 111111 XTR11 JMS 11111
05032 604675 XT1R JMP RXCT1
05033 011112 K12 11112
05034 415555 XT12S XCT 15555
05035 112222 XTR12 JMS 12222
05036 604715 XT2R JMP RXCT2
05037 012223 K23 12223
05040 414444 XT13S XCT 14444
05041 113333 XTR13 JMS 13333
05042 604735 XT3R JMP RXCT3
05043 013334 K34 13334
05044 417776 XT17S XCT 17776
05045 107776 XTR17 JMS 07776
05046 604755 XT4R JMP RXCT4
05047 412525 XCT12S XCT 12525
05050 115252 XCTR12 JMS 15252
05051 604775 XT5R JMP RXCT5
05052 457777 XCTISZ ISZ 17777
05053 057777 XCTDAC DAC 17777
05054 347531 XCTTAD TAD K1
.EJECT

```

		/AUTO-INDEX		
		/		
05055	200020	TSAUTO	LAC 20	/TEST AUTO-INDEX 20
05056	047636		DAC AUTNOT	/STORE (20) AT AUTNOT
05057	220020		LAC* 20	/FALSE AUTO-INDEX 20
05060	207636		LAC AUTNOT	
05061	540020		SAD 20	/COMPARE (AUTNOT) WITH (20)
05062	741000		SKP	
05063	740040	F574	HALT	/ERROR; (20) AUTO-INDEXED
05064	040020		DAC 20	/RESTORE (20)
				/TEST AUTO-INDEX (30)
				/STORE (30) AT AUTNOT
05065	200030		LAC 30	/FALSE AUTO INDEX 30
05066	047636		DAC AUTNOT	
05067	220030		LAC* 30	/COMPARE (AUTNOT) WITH (30)
05070	207636		LAC AUTNOT	
05071	540030		SAD 30	/ERROR; (30) AUTO-INDEXED
05072	741000		SKP	/RESTORE 30
05073	740040	F575	HALT	
05074	040030		DAC 30	
		/		
		/TEST AUTO-INDEX 50		
			LAC 50	/STORE (50) AT AUTNOT
05075	200050		DAC AUTNOT	/FALSE AUTO-INDEX 50
05076	047636		LAC* 50	
05077	220050		LAC AUTNOT	/COMPARE (AUTNOT) WITH (50)
05100	207636		SAD 50	
05101	540050		SKP	/ERROR; (50) AUTO-INDEXED
05102	741000	F576	HALT	/RESTORE 50
05103	740040		DAC 50	
05104	040050			
		/		
		/TEST AUTO-INDEX 110		
			LAC 110	/STORE (110) AT AUTNOT
05105	200110		DAC AUTNOT	/FALSE AUTO INDEX 110
05106	047636		LAC* 110	
05107	220110		LAC AUTNOT	/COMPARE (AUTNOT) WITH (110)
05110	207636		SAD 110	
05111	540110		SKP	/ERROR; (110) AUTO-INDEXED
05112	741000	F577	HALT	/RESTORE 110
05113	740040		DAC 110	
05114	040110			
		/		
		/TEST AUTO-INDEX 210		
			LAC 210	/STORE (210) AT AUTNOT
05115	200210		DAC AUTNOT	/FALSE AUTO-INDEX 210
05116	047636		LAC* 210	
05117	220210		LAC AUTNOT	/COMPARE (AUTNOT) WITH (210)
05120	207636		SAD 210	
05121	540210		SKP	/ERROR; (210) AUTO-INDEXED
05122	741000	F578	HALT	/RESTORE 210
05123	740040		DAC 210	
05124	040210		.EJECT	

05125	200410	/TEST AUTO-INDEX 410	
05126	047636	LAC 410	/STORE (410) AT AUTNOT
05127	224410	DAC AUTNOT	/FALSE AUTOINDEX 410
05130	207636	LAC* 410	
05131	540410	LAC AUTNOT	/COMPARE (AUTNOT) WITH (410)
05132	741000	SAD 410	
05133	740040	SKP	/ERROR; (410) AUT-INDEXED
05134	040410	F579 HALT	/RESTORE 410
		DAC 410	
		/	
05135	201010	/TEST AUTO-INDEX 1010	
05136	047636	LAC 1010	/STORE (1010) AT AUTNOT
05137	221010	DAC AUTNOT	/FALSE AUTO-INDEX 1010
05140	207636	LAC* 1010	
05141	541010	LAC AUTNOT	/COMPARE (AUTNOT) WITH (1010)
05142	741000	SAD 1010	
05143	740040	SKP	/ERROR; (1010) AUT-INDEXED
05144	041010	F580 HALT	/RESTORE 1010
		DAC 1010	
		/	
05145	202010	/TEST AUTO-INDEX 2010	
05146	047636	LAC 2010	/STORE (2010) AT AUTNOT
05147	222010	DAC AUTNOT	/FALSE AUTO INDEX 2010
05150	207636	LAC* 2010	
05151	542010	LAC AUTNOT	/COMPARE (AUTNOT) WITH (2010)
05152	741000	SAD 2010	
05153	740040	SKP	/ERROR; (2010) AUTO-INDEXED
05154	042010	F581 HALT	/RESTORE 2010
		DAC 2010	
		/	
05155	204010	/TEST AUTO-INDEX 4010	
05156	047636	LAC 4010	/STORE (4010) AT AUTNOT
05157	224010	DAC AUTNOT	/FALSE AUTO INDEX 4010
05160	207636	LAC* 4010	
05161	544010	LAC AUTNOT	/COMPARE (AUTNOT) WITH (4010)
05162	741000	SAD 4010	
05163	740040	SKP	/ERROR; (4010) AUTO-INDEXED
05164	044010	F582 HALT	/RESTORE 4010
		DAC 4010	
		/	
05165	447633	ISZ WORK3	/CHECK DONE LOOPING
05166	605055	JMP TSAUTO	/LOOP
05167	106102	JMC GENRAN	/GET NO. FOR NEXT LOOP
05170	106126	JMC CKND	
		.EJECT	

```

/TFST LAC INDIRECT
/
05171 204544 LAC K77
05172 057777 DAC 17777 /((17777 OR 07777) = 07777 OR 17777
05173 207626 LAC M400K /377777
05174 047777 DAC 07777 /OR 17777
05175 237777 LAC* 17777 /OR 07777
05176 347553 TAD K400K /AC = 777777
05177 740001 CMA
05200 740200 SZA
05201 740040 F584 HALT /ERROR. LAC* 17777 OR 07777 FAILED
/
05202 204547 LAC K75
05203 056666 DAC 16666 /((16666 OR 06666) = 07776 OR 17776
05204 207625 LAC M40K /737777
05205 047776 DAC 07776
05206 236666 LAC* 16666 /AC = 737777
05207 347552 TAD K40K /AC = 777777
05210 740001 CMA
05211 740200 SZA
05212 740040 F585 HALT /ERROR. LAC* 16666 OR 06666 FAILED
/
05213 204552 LAC K75
05214 055555 DAC 15555 /((15555 OR 05555) = 07775 OR 17775
05215 207624 LAC M4K /AC = 773777
05216 047775 DAC 07775
05217 235555 LAC* 15555 /AC = 773777
05220 347547 TAD K4K /AC = 777777
05221 740001 CMA
05222 740200 SZA
05223 740040 F586 HALT /ERROR. LAC* 15555 OR 05555 FAILED
/
05224 204555 LAC K74
05225 054444 DAC 14444 /((14444 OR 04444) = 07774 OR 17774
05226 207623 LAC M400 /AC = 777377
05227 047774 DAC 07774
05230 234444 LAC* 14444 /AC = 777377
05231 347544 TAD K400 /AC = 777777
05232 740001 CMA
05233 740200 SZA
05234 740040 F587 HALT /ERROR. LAC* 14444 OR 04444 FAILED
/
05235 204560 LAC K74
05236 053333 DAC 13333 /((13333 OR 03333) = 07773 OR 17773
05237 207622 LAC M40 /AC = 777737
05240 047773 DAC 07773
05241 233333 LAC* 13333 /AC = 777737
05242 347542 TAD K40 /AC = 77777
05243 740001 CMA
05244 740200 SZA
05245 740040 F588 HALT /ERROR. LAC* 13333 OR 03333 FAILED
.EJECT

```

05246	204563	LAC K7	
05247	052222	DAC 12222	/(12222 OR 02222) = 07772 OR 17772
05250	207621	LAC M4	/AC = 77773
05251	047772	DAC 07772	
05252	232222	LAC* 12222	/AC = 77773
05253	347533	TAD K4	/AC = 77777
05254	740001	GMA	
05255	740200	SZA	
05256	740040	HALT	/ERROR, LAC* 12222 OR 02222 FAILED
			/
05257	204571	LAC K71	
05260	051111	DAC 11111	/(11111 OR 01111) = 07771 OR 17771
05261	207620	LAC M1	/AC = 77776
05262	047771	DAC 07771	
05263	231111	LAC* 11111	/AC = 77776
05264	347531	TAD K1	/AC = 77777
05265	740001	GMA	
05266	740200	SZA	
05267	740040	HALT	/ERROR, LAC* 11111 OR 01111 FAILED
			/
05270	205427	LAC INK52	
05271	055252	DAC 15252	/(15252 OR 05252) = 02525 OR 12525
05272	207605	LAC K121	/AC = 525252
05273	052525	DAC 12525	
05274	235252	LAC* 15252	/AC = 525252
05275	347604	TAD K010	/AC = 77777
05276	740001	GMA	
05277	740200	SZA	
05300	740040	HALT	/ERROR, LAC* 15252 OR 05252 FAILED
			/
05301	447633	IS7 WORK3	/CHECK DONE LOOPING
05302	605171	JMP LACIN	/LOOP
05303	106102	JMS GENRAN	/GET NO. FOR NEXT LOOP
05304	106126	JMS CKNO	
		.EJECT	

```

/TEST XCT JMS INDIRECT
/
05305      744000      XTJMSI      CLL
05306      750000      CLA
05307      547721      SAD JST77
05310      741000      SKP
05311      740040      E592      HALT      /ERROR. JMS DEST'B ERROR
/
05312      206060      LAC K17776      /DIRECT ADDRESS
05313      047777      DAC 07777      /JMS* 07777 OR 17777
05314      205430      LAC JMSI1      /ERROR TABLE
05315      047721      DAC JST77
05316      055252      DAC 15252
05317      205431      LAC RJMI1      /JMP RJSI1
05320      057777      DAC 17777
05321      415252      XCT 15252      /XCT TEST
05322      741000      SKP
05323      147721      RJSI1      DZM JST77
05324      217776      LAC 17776
05325      545432      SAD RJSI1X      /RJSI1X = RJSI1-1
05326      751000      CLA!SKP
05327      740040      E593      HALT      /ERROR (17776 OR 07776) NOT =
/TO RJSI1-1
/
/TEST JMS INDIRECT
/
05330      744000      CLL
05331      750000      CLA
05332      547722      SAD JST66
05333      741000      SKP
05334      740040      E594      HALT      /ERROR. JMS DEST'N ERROR
/
05335      206075      LAC K11111      /DIRECT ADR. 11111 OR 01111
05336      056666      DAC 16666      /OR 06666
05337      205433      LAC JSI66      /JMS* 16666 OR 06666
05340      047722      DAC JST66
05341      055252      DAC 15252      /JMS* 16666 OR 06666 AT 15252
/OR 05252
/
05342      205434      LAC RJMI2      /JMP RJSI2
05343      051112      DAC 11112 /OR 01112      /OR 05252
05344      615252      JMP 15252
05345      741000      SKP
05346      147722      RJSI2      DZM JST66      /CLEAR ERROR TABLE
05347      211111      LAC 11111
05350      545435      SAD K15253
05351      751000      CLA!SKP
05352      740040      E595      HALT      /ERROR. RJMP ADR. (11111 OR 01111)
/NOT 15253 OR 05253
.EJECT

```

05353	744000		CLL	
05354	750000		CLA	
05355	547723		SAD JST55	
05356	741000		SKP	
05357	740040	E596	HALT	/ERROR, JMS DEST'N ERROR
05360	206073		LAC K12222	/DIRFCT ADR, 12222 OR 02222
05361	055555		DAC 15255 /OR 055555	
05362	205436		LAC JS155	/JMS* 15555 OR 05555
05363	047723		DAC JS155	
05364	055252		DAC 15252	/JMS* 15555 OR 05555 AT /15252 OR 05252
05365	205437		LAC RJM13	/JMP RJSI3
05366	052223		DAC 12223	/OR 02223
05367	615252		JMP 15252	/OR 05252
05370	741000		SKP	
05371	147723	RJSI3	DZM JST55	/CLEAR ERROR TABLE
05372	212222		LAC 12222	/OR 02222
05373	545435		SAD K15253	
05374	751000		CLA!SKP	
05375	740040	E597	HALT	/ERROR, RJMP ADR, (12222 OR 02222) /NOT 15253 OR 05253
05376	744000		CLL	
05377	750000		CLA	
05400	547724		SAD JST44	
05401	741000		SKP	
05402	740040	E598	HALT	/ERROR, JMS DEST'N ERROR
05403	206071	/	LAC K13333	/DIRFCT ADR, 13333 OR 03333
05404	054444		DAC 14444	/OR 04444
05405	205440		LAC JST44	/JMS* 14444 OR 04444
05406	047724		DAC JST44	
05407	055252		DAC 15252	/OR 05252
05410	205441		LAC RJM14	
05411	053334		DAC 13334	/OR 03334
05412	615252		JMP 15252	/OR 05252
05413	741000		SKP	
05414	147724	RJSI4	DZM JST44	/CLEAR ERROR TABLE
05415	213333		LAC 13333	/OR 03333
05416	545435		SAD K15253	
05417	751000		CLA!SKP	
05420	740040	E599	HALT	/ERROR, RJMP ADR (13333 OR 03333) /NOT 15253 OR 05253
05421	447633		ISZ WORK3	/CHECK DONE LOOPING
05422	605305		JMP XTJMSI	/LOOP
05423	106102		JMS GENRAN	/GET NO. FOR NEXT LOOP
05424	106126		JMS CKNO	
05425	700004		CLOF	
05426	605442		JMP XTJCT	/TEST XCT INDIRECTS
			.EJECT	

/CONSTANTS FOR LAC*, XCT JMS* MODIFIED
/WHEN IN UPPER 4*

05427 012525
05430 127777
05431 605323
05432 005322
05433 136666
05434 605346
05435 015253
05436 135555
05437 605371
05440 134444
05441 605414

JMS12 12525
JMS11 JMS* 07777
RJM11 JMP RJF11
RJS11X RJS11-1
JSI66 JMS* 16666
RJM12 JMP RJS12
K15253 15253
JSI55 JMS* 15555
RJM13 JMP RJS13
JSI44 JMS* 14444
RJM14 JMP RJS14
/PDP-9 BASIC EXERCISER - TAPE 6

/TEST XCT INDIRECT

05442 700002
05443 207530
05444 057777
05445 205516
05446 040000
05447 777777
05450 052525
05451 437777
05452 212525
05453 740200
05454 740040

XTXCT 10F /PI OFF FOR XCT* TO USE
/LOCATION 0
LAC K0 /OR 07777
DAC 17777
LAC XCTDZM /DZM 12525 OR 02525
DAC 0
LAW 17777
DAC 12525 /((12525 OR 02525) = 777777
XCT* 17777 /OR 07777
LAC 12525 /OR 02525
SZ/ F600 HALT /ERROR, XCT* A DZM FAILED
/(12525 OR 02525) NOT 777777

/TEST ISZ INDIRECT. PI IS OFF

05455 207530
05456 057777
05457 777777
05460 040000
05461 477777
05462 740040

LAC K0
DAC 17777 /OR 07777
LAW 17777
DAC 0 /((0) = 777777
ISZ* 17777
E601 HALT /ERROR ISZ* FAILED TO SKIP/
.EJECT

/TEST XCT* AND* PI IS OFF

0463 207530
 0464 057777
 0465 207531
 0466 057776
 0467 205517
 0470 040000
 0471 207602
 0472 040001
 0473 750001
 0474 437777
 0475 547602
 0476 741000
 0477 740040

E602

LAC K0
 DAC 17777
 LAC K1
 DAC 17776
 LAC ANDI
 DAC 0
 LAC K2525
 DAC 1
 CLA:OMA
 XCT* 17777
 SAD K2525
 SKP
 HALT

/OR 07777, INDIRECT ADR FOR XCT*
 /DIRECT ADR. FOR AND*
 /OR 07776
 /AND* 17776 OR 07776
 /(1) = 2525
 /AC = ONFS
 /OR 07777

/ERROR, XCT* 17777 OR 07777
 /FOLLOWED BY AND* 17776
 /OR 07776 FAILED
 /CHECK DONE LOOPING
 /LOOP
 /GET NO. FOR NEXT LOOP

05500 447633
 05501 605442
 05502 106102
 05503 106126
 05504 207616
 05505 040001
 05506 750004
 05507 740010
 05510 740100
 05511 700042
 05512 507556
 05513 741200
 05514 106604
 05515 605520

ISZ WORK3
 JMP XTXCT
 JMS GENRAN
 JMS CKNO
 LAC KSKP
 DAC 1
 LAS
 RAL
 SMA
 TON
 AND K20K
 SNA
 JMS SETCLK
 JMP AUTOIN
 .EJECT

/RESTORE LOC 1

/PI RACK ON

/TEST ACS5
 /CLOCK BACK ON
 /TEST INDEX REGISTERS

```

/CONSTANTS FOR PRECEDING LOOPS, MODIFIED WHEN IN UPPER 4K
/
05516 152525 XGTDM DZM 12525
05517 537776 ANDI AND* 17776
/
/TEST AUTO-INDEX (XOR* 10)
/
05520 206060 AUTOIN LAC K17776
05521 040010 DAC 10 /((10) = 17776 OR 07776
05522 207564 LAC K15 /AC = 11111
05523 057777 DAC 17777 /OR 07777
05524 260010 XOR* 10
05525 740200 SZA
05526 740040 F603 HALT /ERROR. XOR* 10 FAILED
/AC NOT 11111

05527 200010 LAC 10
05530 546061 SAD K17777
05531 741000 SKP
05532 740040 E604 HALT /ERROR. (10) NOT INCREMENTED+1
/

05533 206062 LAC K16665
05534 040010 DAC 10 /((10) = 16665 OR 06665
05535 207565 LAC K25 /AC = 22222
05536 056666 DAC 16666 /OR 06666
05537 260010 XOR* 10
05540 740200 SZA
05541 740040 F605 HALT /ERROR. XOR* 10 FAILED
/AC NOT 22222

05542 200010 LAC 10
05543 546063 SAD K16666
05544 741000 SKP
05545 740040 F606 HALT /ERROR. (10) NOT INCREMENTED+1
/

05546 206064 LAC K15554
05547 040010 DAC 10 /((10) = 15554 OR 05554
05550 207566 LAC K35 /AC = 333333
05551 055555 DAC 15555
05552 260010 XOR* 10
05553 740200 SZA
05554 740040 F607 HALT /ERROR. XOR* 10 FAILED. AC NOT 333333
/AC NOT 333333

05555 200010 LAC 10
05556 546065 SAD K15555
05557 741000 SKP
05560 740040 E608 HALT /ERROR. (10) NOT INCREMENTED +1
.EJECT

```


05561	206066	LAC K14443	
05562	040010	DAC 10	/(10) = 14443 OR 04443
05563	207567	LAC K43	/AC = 444444
05564	054444	DAC 14444	
05565	260010	XOR* 10	
05566	740200	SZA	
05567	740040	HALT	/ERROR. XOR* 10 FAILED AC NOT 444444
05570	200010	LAC 10	
05571	546067	SAD K14444	
05572	741000	SKP	
05573	740040	HALT	/ERROR. (10) NOT INCREMENTED +1
05574	206070	LAC K13332	
05575	040010	DAC 10	/(10) = 13332 OR 03332
05576	207570	LAC K55	/AC = 55555
05577	053333	DAC 13333	
05600	260010	XOR* 10	
05601	740200	SZA	
05602	740040	HALT	/ERROR. XOR* 10 FAILED AC NOT 555555
05603	200010	LAC 10	
05604	546071	SAD K13333	
05605	741000	SKP	
05606	740040	HALT	/ERROR. (10) NOT INCREMENTED+1
05607	206072	LAC K12221	
05610	040010	DAC 10	/(10) = 12221 OR 02221
05611	207571	LAC K66	/AC = 666666
05612	052222	DAC 12222	
05613	260010	XOR* 10	
05614	740200	SZA	
05615	740040	HALT	/ERROR. XOR* 10 FAILED. AC NOT 666666
05616	200010	LAC 10	
05617	546073	SAD K12222	
05620	741000	SKP	
05621	740040	HALT	/ERROR. (10 NOT INCREMENTED+1
05622	206074	LAC K11110	
05623	040010	DAC 10	/(10) = 11110 OR 01110
05624	207572	LAC K75	/AC = 777777
05625	051111	DAC 11111	
05626	260010	XOR* 10	
05627	740200	SZA	
05630	740040	HALT	/ERROR. XOR* 10 FAILED. AC NOT 777777
05631	200010	LAC 10	
05632	546075	SAD K11111	
05633	741000	SKP	
05634	740040	HALT	/ERROR. (10) NOT INCREMENTED +1
		.EJECT	

```

/TEST ISZ* 11
/
05635 206076 LAC K15252
05636 040011 DAC 11 /((11) = 15252 OR 05252
05637 207572 LAC K75
05640 055253 DAC 15253 /OR 05253
05641 460011 ISZ* 11
05642 740040 F617 HALT /ERROR. ISZ FAILED TO SKIP
/AUTO-INDEX 11 FAILED

05643 215253 LAC 15253
05644 740200 SZA
05645 740040 F618 HALT /ERROR. (15253 OR 05253) NOT 0
/ISZ FAILED

05646 200011 LAC 11
05647 545435 SAA K15253
05650 741000 SKP
05651 740040 F619 HALT /ERROR. (11) NOT INCREMENTED+1
/
/AUTO-INDEX JMP* 12.
/
05652 207725 LAC AUTJMP
05653 740200 SZA
05654 740040 F620 HALT /ERROR. JMP* 12 FAILED TO REACH 15253
/JMP* 12
05655 207650 LAC JMPAUT
05656 047725 DAC AUTJMP
05657 206077 LAC AUTRET /RJMP TO AUTR
05660 055253 DAC 15253
05661 206076 LAC K15252
05662 040012 DAC 12 /((12) = 15252
05663 620012 JMP* 12
05664 741000 SKP
05665 147725 AUTR DZM AUTJMP /CLEAR ERROR TABLE
05666 200012 LAC 12
05667 545435 SAA K15253
05670 741000 SKP
05671 740040 F621 HALT /ERROR. (12) NOT INCREMENTED+1
/
/AUTO-INDEX (DAC* 13).
/
05672 204547 LAC K75
05673 040013 DAC 13 /((13) = 07776 OR 17776
05674 204547 LAC K75
05675 047777 DAC 07777 /((07777) = 07776
05676 207530 LAC K0
05677 060013 DAC* 13
05700 207777 LAC 07777
05701 740200 SZA
05702 740040 F622 HALT /ERROR. (07777) NOT 0. DAC* 13 FAILED
05703 200013 LAC 13
05704 544544 SAA K77
05705 741000 SKP
05706 740040 F623 HALT /ERROR (13) NOT INCREMENTED+1
.EJECT

```

```

/AUTO-INDEX (XCT* 14).
/
05707 204571 LAC K71
05710 040014 DAC 14 /(14) = 07771
05711 207641 LAC AU:CMA
05712 047772 DAC 07772 /(07772) = CMA
05713 750001 CLA:CMA
05714 420014 XCT* 14
05715 740200 SZA
05716 740040 E624 HALT /ERROR. AC NOT 0 (XCT* 14) A CMA
05717 200014 LAC 14
05720 544563 SAD K72
05721 741000 SKP
05722 740040 E625 HALT /ERROR. (14) NOT INCREMENTED+1
/

```

```

/AUTO-INDEX (TAD* 15).
/
05723 206060 LAC K17776
05724 040015 DAC 15 /(15) = 17776 OR 07776
05725 207531 LAC K1
05726 057777 DAC 17777 /OR 07777
05727 754001 CLL:CLA:CMA
05730 360015 TAD* 15
05731 740200 SZA
05732 740040 E626 HALT /ERROR. AC NOT 0 (TAD* 15)
05733 740400 SNL
05734 740040 E627 HALT /ERROR. LINK NOT 1 (TAD* 15)
05735 200015 LAC 15
05736 546061 SAD K17777
05737 741000 SKP
05740 740040 E628 HALT /ERROR. (15) NOT INCREMENTED+1
05741 217777 LAC 17777
05742 547531 SAD K1
05743 741000 SKP
05744 740040 E629 HALT /ERROR. (17777 OR 07777) NOT 1
/

```

```

/AUTO-INDEX (SAD* 16).
/
05745 204555 LAC K74
05746 040016 DAC 16 /(16) = 07774
05747 207603 LAC K5252
05750 047775 DAC 07775 /(07775) = 5252
05751 560016 SAD* 16
05752 741000 SKP
05753 740040 E630 HALT /ERROR. SAD SKIPPED (SAD* 16)
05754 207775 LAC 07775
05755 547603 SAD K5252
05756 741000 SKP
05757 740040 E631 HALT /ERROR. (07775) NOT 5252
05760 200016 LAC 16
05761 544552 SAD K75
05762 741000 SKP
05763 740040 E632 HALT /ERROR. (16) NOT INCREMENTED+1
.EJECT

```

```

/AUTO-INDEX (JMS* 17).
/
0F764 207726 LAC AUTJMS
0F765 744200 SZA!CLI
F633 HALT /ERROR. ERROR TABLE NOT 0
0F766 740040 LAC AUTRJM
0F767 206100 DAC 07773 /JMP AUTRE1
0F770 047773 LAC K71
0F771 204571 DAC 17 /((17) = 07771
0F772 040017 LAC JMC:AUT
0F773 207646 DAC AUTJMS /JMS* 17 IN ERROR TABLE
0F774 047726 JMS* 17
0F775 120017 SKP
0F776 741000 AUTRE1 DZM AUTJMS /CLEAR ERROR TABLE
0F777 147726 LAC 07772
0F000 207772 SAD AURJMP
0F001 546101 SKP
0F002 741000 E634 HALT /ERROR. (07772) STORED WRONG
0F003 740040 /((07772) SHOULD = AUTRE1-1

0F004 200017 LAC 17
0F005 544563 SAD K72
0F006 741000 SKP
0F007 740040 E635 HALT /ERROR. (17) NOT INCREMENTED+1

/AUTO-INDEX (ISZ* 10) (10) = 10.
/
0F010 207534 LAC K10
0F011 040010 DAC 10
0F012 207572 LAC K7S
0F013 040011 DAC 11 /((11) = 777777
0F014 460010 ISZ* 10
0F015 740040 E636 HALT /ERROR. ISZ FAILED TO SKIP
0F016 200010 LAC 10
0F017 547535 SAD K11
0F020 741000 SKP
0F021 740040 F637 HALT /ERROR. (10) NOT 11
0F022 200011 LAC 11
0F023 740200 SZA
0F024 740040 F638 HALT /ERROR. (11) NOT 0

/AUTO-INDEX (ISZ* 11) (11) = 10.
/
0F025 207534 LAC K11
0F026 040011 DAC 11 /((11) = 10
0F027 460011 ISZ* 11
0F030 200011 LAC 11
0F031 547536 SAD K12A
0F032 741000 SKP
0F033 740040 F639 HALT /ERROR. ISZ* 11 FAILED TO
/INCREMENT +2 (11)

.EJECT

```

/AUTO-INDEX (XCT* 15).

06034 204560
 06035 040015
 06036 207644
 06037 047774
 06040 207645
 06041 047775
 06042 750000
 06043 420015
 06044 740001
 06045 740200
 06046 740040

 06047 200015
 06050 544552
 06051 741000
 06052 740040
 06053 447633
 06054 605520
 06055 777775
 06056 047633
 06057 606154

E640

E641

LAC K74
 DAC 15
 LAC LAWAUT
 DAC 07774
 LAC LAWFUL
 DAC 07775
 CLA
 XCT* 15
 CMA
 SZA
 HALT

LAC 15
 SMO K74
 SKP
 HALT
 ISZ WORK3
 JMP AUTOIN
 LAW -3
 DAC WORK3
 JMP CHKBRD
 .EJECT

/(15) = 07773

/(07774) = XCT* 15

/(07775) = LAW 17777

/AC = 0

/ERROR. AC NOT ONES

/LAW 1777 DID NOT OCCUR

/ERROR. (15) NOT 7775 (XCT* 15)

/CHECK DONE LOOPING

/LOOP

/BASIC MEMORY CHECKERBOARD

/CONSTANTS FOR AUTO-INDEXING. MODIFIED WHEN IN HI 4K

01060	017776	K17776	17776
01061	017777	K17777	17777
01062	016665	K16665	16665
01063	016666	K16666	16666
01064	015554	K15554	15554
01065	015555	K15555	15555
01066	014443	K14443	14443
01067	014444	K14444	14444
01070	013332	K13332	13332
01071	013333	K13333	13333
01072	012221	K12221	12221
01073	012222	K12222	12222
01074	011110	K11110	11110
01075	011111	K11111	11111
01076	015252	K15252	15252
01077	005665	AUTRFT	JMP AUTR
01100	005777	AUTRJM	JMP AUTRE1
01101	005776	AURJMP	AUTRE1-1 .EJECT

```

/RANDOM NUMBER GENERATORS
/
GENRAN 0
      LAC RANDEX
      SAD ENDTBL
      SKP
      JMP RANTAD-1
      LAC TBLTOP
      DAC RANDEX
      LAC RANCON
      CLL:RAL
      SZL
      TAD K1
      DAC RANCON
      LAC* RANDEX
      TAD RANCON
      DAC* RANDEX
      ISZ RANDEX
      JMP* GENRAN
      /CHECK FOR END OF TABLE
/END
/GENERATE RANDOM
/RESET INDEX TO FIRST
/POSITION MODIFIER
/1 LFFT
/WAS BIT 0 A 1
/YES MAKE 17 A 1
/RESTORE MODIFIER
/GET FIRST CONTROL
/ADD MODIFIER
/NEW CONTROL = RANDOM
/STEP POINTER
/EXIT

06102 000000
06103 206123
06104 546124
06105 741000
06106 606116
06107 206125
06110 046123
06111 207654
06112 744010
06113 741400
06114 347531
06115 047654
06116 226123
06117 347654
06120 066123
06121 446123
06122 626102

RANTAD
/
RANDEX RANTBL+10
ENDTRL RANTBL+10
TBLTOP RANTRL
/
CKNO 0
      AND K37S
      CMA
      DAC WORK3
      JMP* CKNO
      /MAKE 65K OR LESS
/LOOP COUNTER
/EXIT

/
RANGEN 0
      LAC RANDEX
      SAD ENDTBL
      SKP
      JMP TADRAN-1
      LAC TBLTOP
      DAC RANDEX
      LAC RANCON
      CLL:RAL
      SZL
      TAD K1
      DAC RANCON
      LAC* RANDEX
      TAD RANCON
      DAC* RANDEX
      NOP
      JMP* RANGEN
      /CHECK FOR TABLE END
/END
/GENERATE RANDOM
/RESET INDEX TO FIRST
/POSITION MODIFIER
/1 LFFT
/WAS BIT 0 A 1
/MAKE 17 A 1
/RESTORE MODIFIER
/GET FIRST CONTROL
/ADD MODIFIER
/NEW CONTROL = RANDOM
/EXIT

06123 007655
06124 007655
06125 007655

06126 000000
06127 507607
06130 740001
06131 047633
06132 626126

06133 000000
06134 206123
06135 546124
06136 741000
06137 606147
06140 206125
06141 046123
06142 207654
06143 744010
06144 741400
06145 347531
06146 047654
06147 226123
06150 347654
06151 066123
06152 740000
06153 626133

TADRAN
      EJECT

```

0A154 700002
 0A155 750004
 0A156 740010
 0A157 740100
 0A160 700042
 0A161 777777
 0A162 047316
 0A163 206165
 0A164 046332
 0A165 207300
 0A166 047304
 0A167 147631
 0A170 707704

CHKRRD IOF
 LAS
 RAI
 SMA
 IOM
 LAW -1
 DAC RITSUP
 LAC ,+3
 DAC NEYPAT
 LAC KPAT
 DAC MPAT
 DZM WORK1
 LEM

/CHECK PI INHIBITED
 /PI ON

/LOAD CHECKERBOARD

0A171 106312
 0A172 207304
 0A173 047306
 0A174 777776
 0A175 047313
 0A176 777770
 0A177 047276
 0A200 777760
 0A201 047275
 0A202 207306
 0A203 047305
 0A204 207305
 0A205 744010
 0A206 047305
 0A207 751400
 0A210 740001
 0A211 067277
 0A212 207277
 0A213 546061
 0A214 606234
 0A215 447277
 0A216 447275
 0A217 606204
 0A220 447276
 0A221 606200
 0A222 207304
 0A223 744020
 0A224 740400
 0A225 606230
 0A226 447313
 0A227 606176
 0A230 207306
 0A231 740001
 0A232 047306
 0A233 606174

LOAD JMS ADJUST
 LAC MPAT
 DAC PATWD
 LAW -2
 DAC WCP56
 LCNTA LAW -10
 DAC WC128
 LCNTR LAW -20
 DAC WC16
 LAC PATWD
 DAC PATR
 WLOOP LAC PATR
 RCL
 DAC PATR
 SZL !CLA
 CMA
 DAC+ LLREG
 LAC LLREG
 SDB K17777
 JMP READ
 ISZ LLREG
 ISZ WC16
 JMP WLOOP
 ISZ WC128
 JMP LCNTR
 LAC MPAT
 RCL
 SNI
 JMP ,+3
 ISZ WCP56
 JMP LCNTA
 LAC PATWD
 CMA
 DAC PATWD
 JMP LOAD+2
 .EJECT

/-16 DECIMAL

/TEST FOR A 1 OR 0

/STORE WORD

/DONE LOADING?

/YES

/INCR. ADR

/16 WORDS?

/NO

06234	207304	/READ CHECKERBOARD	
06235	047632	READ	LAC MPAT
06236	106312		DAC WORK2
06237	777776		JMS ADJUST
06240	047313	RCNTA	LAW -2
06241	777770		DAC WC256
06242	047276		LAW -14
06243	207632	RCNTB	DAC WC128
06244	047305		LAC WORK2
06245	777760		DAC PATR
06246	047275		LAW -24
06247	777772	RCLOOP	DAC WC16
06250	047241		LAW -6
06251	207305		DAC CRIF
06252	744010		LAC PATR
06253	047305		RCL
06254	751400		DAC PATR
06255	740001		SZL:CLA
06256	047306		CMA
06257	227277		DAC PATWD
06260	740001		LAC* LLREG
06261	067277		CMA
06262	447241		DAC* LLREG
06263	606257		ISZ CRIF
06264	227277		JMP .-4
06265	547306		LAC* LLREG
06266	741000		SAD PATWD
06267	606340		SKP
06270	207277		JMP ERROR
06271	546061	RDRTN	LAC LLREG
06272	606325		SAD K1777
06273	447277		JMP NXTST
06274	447275		ISZ LLREG
06275	606247		ISZ WC16
06276	447276		JMP RCLOOP
06277	606243		ISZ WC128
06300	207304		JMP RCNTB
06301	744020		LAC MPAT
06302	740400		RDR
06303	606306		SNL
06304	447313		JMP .+3
06305	606241		ISZ WC256
06306	207632		JMP RCNTA+2
06307	740001		LAC WORK2
06310	047632		CMA
06311	606237		DAC WORK2
			JMP RCNTA
			.EJECT

/RSTORE PATTERN GEN

/-26 DECIMAL

/SAVE FOR COMPARE

/COMPARE

/OK

/DONE READING?

/YES

/INCR. ADR

/16 WORDS?

/NO.

0A312 000000
 0A313 206313
 0A314 507555
 0A315 740200
 0A316 606322
 0A317 207555
 0A320 047277
 0A321 626312

 0A322 207544
 0A323 047277
 0A324 626312

 0A325 207631
 0A326 247560
 0A327 741200
 0A330 606363
 0A331 446332
 0A332 207300
 0A333 047304
 0A334 207631
 0A335 347557
 0A336 047631
 0A337 606171

ADJUST 0
 LAC .
 AND K10K
 SZA
 JMP ULADJ
 LAC K10K
 DAC LLREG
 JMP* ADJUST

 /
 ULADJ LAC K400
 DAC LLREG
 JMP* ADJUST

 /
 NXTST LAC WORK1
 XOR K600K
 SNA
 JMP CKLOOP
 ISZ .+1

 NEXPAT LAC KPAT
 DAC MPAT
 LAC WORK1
 TAD K200K
 DAC WORK1
 JMP LOAD-1
 .EJECT

/SA = 010000 FOR PATTERN

/SA = 400 FOR PATTERN

```

/BASIC MEMORY CHECKERBOARD ERROR ROUTINE.
06340 047333 FRPDR DAC WORK3
06341 507316 AND RITSUP
06342 740200 SZA
06343 740001 CMA
06344 507316 AND RITSUP
06345 741200 SNA
06346 606270 JMP RPTN
06347 207277 LAC LLREG /DISPLAY ADDRESS
06350 740040 F642 HLT
06351 207306 LAC PATWD /GOOD DATA
06352 740040 GDATA HLT
06353 207633 RDATA LAC WORK3 /BAD DATA
06354 740040 HLT
06355 750004 LAS /SUPPRESS HERE
06356 740001 CMA
06357 047316 DAC RITSUP
06360 207632 LAC WORK2 /PATTERN CONTROL WORD
06361 740040 PATT HLT
06362 606055 JMP E641+3 /START OVER
/
CKLOOP LAS
RTI
RAL
SPA
JMP CHKBRD /LOOP ON MEMORY TEST
ISZ WORK3 /CHECK DONE LOOPING
JMP CHKBRD /LOOP
IOF
LAS
AND K2K /CHECK ACS 4 FOR
/INHIBIT RELOCATION
SNA
JMP ENTST /RELOCATE
LAW 207
JMS TLSSF /RELL FOR ONE PASS
LAS
RAL
SPA /CHECK FOR PI INHIBITED
JMS PINOT /INHIBITED
JMP SEQUEN /START OVER IN THIS 4K
/PDP-9 BASIC EXERCISER - TAPE 7
/
/ROUTINE FOR PROGRAM RELOCATION
/
ENTST IOF /PI OFF DURING RELOCATION
LAW -7755
DAC WDCNT
LAC .
AND K10K
SZA /SEE IF IN LO OR HI 4K
JMP MVRK /HI 4K
CMA
DAC WORK2 /SOURCE ADDRESS
LAC K777 /DEST'N ADR. TO HI 4K
06406 700002
06407 770023
06410 047312
06411 206411
06412 507555
06413 740200
06414 606510
06415 740001
06416 047632
06417 207317

```

06420	047633		DAC WORK3	
06421	207632	MOVE	LAC WORK2	
06422	047314		DAC MOVES	
06423	207633		LAC WORK3	
06424	047315		DAC MOVED	
06425	447315		ISZ MOVED	
06426	167315		DZM* MOVED	/CLEAR DEST'N TO 0'S
06427	447312		ISZ WDCNT	
06430	606425		JMP .-5	
06431	207633		LAC WORK3	
06432	047315		DAC MOVED	/RESTORE DEST'N S.A.
06433	447314	RFROM	ISZ MOVES	
06434	227314		LAC* MOVES	/SOURCE ADR.
06435	047633		DAC WORK3	/SAVE INSTRUCTION
06436	507561		AND K700K	
06437	247561		XOR K700K	
06440	740200		SZA	/OPERATE INST. IF 0
06441	606515		JMP MRINS	/MEMORY REF
06442	207633		LAC WORK3	
06443	447315	MVRTN	ISZ MOVED	
06444	067315		DAC* MOVED	/STORE IN OPPOSITE 4K
06445	547274		SAD LIMITA	/DONE WITH INST. IF EQUAL
06446	741000		SKP	/MOVE CONSTANT TABLES
06447	606433		JMP RFROM	/MOVE ANOTHER INST.
06450	147633		DZM WORK3	
06451	447314	MVCST	ISZ MOVES	
06452	227314		LAC* MOVES	/SOURCE
06453	447315		ISZ MOVED	
06454	067315		DAC* MOVED	/DEST'N
06455	547274		SAD LIMITA	/DONE MOVING IF EQUAL
06456	741000		SKP	/DONE
06457	606451		JMP MVCST	/MOVE ANOTHER CONSTANT
			.EJECT	

06460 204541
06461 247555
06462 054541
06463 204376
06464 247555
06465 054376
06466 207326
06467 247555
06470 057326
06471 207325
06472 247555
06473 057325
06474 750004
06475 740010
06476 741100
06477 107250

LAC K11000
XOR K11K
DAC K10000+10000
LAC F534+6
XOR K11K
DAC F534+6+10000
LAC TTIN
XOR K11K
DAC TTIN+10000
LAC TTOUT
XOR K11K
DAC TTOUT+10000
LAS
RAI
SPA
JMS PINOT
.EJECT

/LOWER

/CHECK FOR INHIBIT PI
/INHIBITED

06500	206500	BGNAGN	LAC .	
06501	507555		AND K10K	
06502	740200		SZA	/SEF WHICH 4K
06503	741000		SKE	
06504	627310		JMP* RGNHI	/START OVER IN HI 4K
06505	760207		LAR 207	/RELI
06506	107210		JMS TLSSF	
06507	627307		JMP* R-NLO	/START IN LOW 4K
/				
/SFTUP TO MOVE TO LOW 4K				
/				
06510	777777	MVRK	LAR 17777	/DESTIN
06511	047633		DAC WORK3	
06512	207317		LAC K7777	/SOURCE
06513	047314		DAC MOVES	
06514	606423		JMP MOVE+2	/MOVE PROGRAM
/				
/ADJUST MEMORY REF. INSTRUCTIONS, DO NOT ADJUST IF				
/ADR. PORTION=ANY ADR. FROM 0 TO 21.				
/				
06515	147312	MRINS	DZM WDCNT	/ADR. COMPARE WORD
06516	207633		LAC WORK3	/INST. TO BE MODIFIED
06517	507317		AND K7777	/CLEAR BITS 0-5
06520	047632		DAC WORK2	/SAVE
06521	207312		LAC WDCNT	
06522	547541		SAD K22	/DONE IF EQUAL TO 22
06523	606530		JMP .+5	
06524	547632		SAD WORK2	/COMPARE
06525	606442		JMP MVRTN-1	/ADR. IS SOME REG. FROM 0
/				
06526	447312		ISZ WDCNT	/TO 21, MOVE WITHOUT ADJUSTING
06527	606521		JMP .-6	/ADR. COUNT+1
06530	207555		LAC K10K	/10000
06531	247633		XOR WORK3	/ADJUST INST. BY 10000
06532	606443		JMP MVRTN	/MOVE
/				
06533	047642	SAV3	DAC SAVAC	/THESE ARE MODIFIED FOR
06534	047643	SAV5	DAC RJMP	/RELOCATION
06535	606536	SAV6	JMP SRVINT	
.FJECT				

```

/
/SERVICE ALL INTERRUPTS
/
06536 700001  SRVINT  CLSF  /CHECK FOR PI FROM CLOCK
06537 741000  SKP  /SOME OTHER DEVICE
06540 606575  JMP CLKINT
06541 700314  IORS
06542 741100  SPA  /STATUS WORD BIT 0 MUST = 0
06543 740040  F643  HALT  /ERROR. BIT 0 SET. PI OFF
06544 207630  LAC WORK
06545 740010  RAI
06546 741100  SPA  /SEE IF TTY IN USE AT TIME
                                /OF PI
06547 606636  JMP TTYINT  /CONTINUE PRINTING
06550 700314  IORS  /I/O STATUS WORD
06551 507321  AND K1400
06552 740200  SZA  /CHECK FOR NO TAPE FLAGS
06553 107141  JMS RNFLG  /EITHER READER OR PUNCH NO TAPE
06554 700101  RSE  /CHECK FOR PI FROM READER
06555 741000  SKP
06556 606776  JMP READA  /READ MORE
06557 700201  PSF  /CHECK PUNCH PI
06560 606562  JMP RTNIT  /SOME OTHER DEVICE
06561 626763  JMP* GOPNCH /PUNCH MORE

/
/SFTUP TO RETURN TO INSTRUCTION TEST
/
06562 744000  RTNIT  CLL
06563 207643  LAC RJMP  /C(0) AT PI
06564 741100  SPA  /CHECK LINK
06565 744002  STL  /RESTORE LINK
06566 507601  AND K175
06567 546574  SAD ILINT
06570 740040  F644  HALT  /ERROR. PI OCCURRED AFTER LAC SAVAC
                                /INSTEAD OF JMP * RJMP.
06571 207642  LAC SAVAC  /AC AT TIME OF PI
06572 700042  PION  ION  /PI ON
06573 627643  JMP* RJMP  /CONTENTS OF (0) AFTER PI

/
ILINT  PION
/
06575 200007  CLKINT  LAC 7
06576 547537  SADR K100  /LEFT CLOCK CONTINUE FOR 1/2 SEC
06577 741000  SKP
06600 606575  JMP .-3
06601 700004  CLOF
06602 106621  JMS CLKSET  /RESFT CLOCK TO RANDOM VALUE
06603 606562  JMP RTNIT  /RETURN TO INST. TEST
                                .EJECT

```

		/		
		/SETUP CLOCK VALUES		
		/		
06604	000000	SETCLK	0	/GET A NO. FOR CLOCK
06605	106102		JMS GELRAN	/SAVE
06606	047312		DAC WDCNT	/MAX. TIME = 9 SECS.
06607	507323		AND K777	/SAVE
06610	047312		DAC WDCNT	/MIN. TIME = 2 SEC.
06611	347324		TAD M167	/POS. = 2 SECS. OR MORE
06612	741100		SPA	/NEG = LESS THAN 2 SEC.
06613	606605		JMP SETCLK+1	
06614	207312		LAC WDCNT	
06615	740001		CMA	/PUT VALUE IN (7)
06616	040007		DAC 7	/CLOCK ON
06617	700044		CLON	/EXIT
06620	626604		JMP* SFTCLK	
		/		
		CLKSET	0	/GET A NO. FOR CLOCK
06621	000000		JMS RANGEN	/SAVE
06622	106133		DAC WDCNT	/MAX. TIME = 9 SECS.
06623	047312		AND K777	/SAVE
06624	507323		DAC WDCNT	/MIN. TIME = 2 SECS.
06625	047312		TAD M167	/POS. = 2 SECS. OR MORE
06626	347324		SPA	/NEG. = LESS THAN 2 SECS.
06627	741100		JMP CLKSET+1	
06630	606622		LAC WDCNT	
06631	207312		CMA	/PUT VALUE IN (7)
06632	740001		DAC 7	/CLOCK ON
06633	040007		CLON	/EXIT
06634	700044		JMP* CLKSET	
06635	626621		.EJECT	


```

/SETUP FOR READ, PUNCH, OR PRINT
/
00636 207325 TTYINT LAC TTOUT /IF EQUAL GO PUNCH AND READ
00637 546701 SMO ENDOUT
00640 606657 JMP PREADY
00641 207630 LAC WORK
00642 507530 AND KA
00643 247557 XOR K200K
00644 047630 DAC WORK
00645 700101 RSE
00646 741000 SKP
00647 740040 E645 HALT /ERROR. READ FLAG UP
00650 700201 PSI
00651 741000 SKP
00652 740040 E646 HALT /ERROR. PUNCH FLAG UP
00653 447325 IS? TTOUT /TTOUT = CHAR. BIN POINTER
00654 227325 LAC* TTOUT /GET CHAR. FROM TTY BIN
00655 700406 TLS /PRINT ONE CHARACTER
00656 606562 JMP RTNIT /RETURN TO INST. TEST

/
00657 147630 PREADY DZM WORK
00660 147327 DZM CNTA
00661 147330 DZM CNTB
00662 700402 TCF /CLEAR TTY FLAG
00663 750004 LAS
00664 507543 AND K1K
00665 741200 SNA /TEST ACS 8 A 1
00666 606671 JMP .+3
00667 700104 RSA /SELECT READER
00670 606562 JMP RTNIT /RETURN TO INST. TEST
00671 206763 LAC GOPNCH
00672 741200 SNA /0=1ST TIME THRU
00673 606702 JMP PNSTRT /START SEQUENCE
00674 700104 RSA /SELECT READER
00675 626763 JMP* GOPNCH /CONTINUE SEQUENCE

/
00676 007357 DATARL TTRUFA-1
00677 007443 FNDBIN TTRUFA+63
00700 007443 OUTTOP TTRUFB-1
00701 007527 ENDOUT TTRUFB+63
.EJECT

```

```

/
/PUNCH DATA
/
06702 107241 PNST=I JMS CRIF /CR,LF
06703 700402 TCF /CLEAR TTY FLAG
06704 750004 LAS
06705 507546 AND K3K /MASK ACS 7 AND 8
06706 741200 SNA /IF EITHER IS A 1, DON'T PUNCH
06707 606712 JMP .+3 /PUNCH DATA
06710 700202 PCF /CLEAR PUNCH FLAG. NO MORE
/PI'S FROM PUNCH SHOULD OCCUR.
/RETURN TO INST. TEST

06711 606562 JMP RTNIT
06712 750000 CLA
06713 700104 RSA /SELECT READER AND PUNCH
06714 106763 JMS GOPNCH /TO INITIATE SEQUENCE
06715 207333 LAC K300
06716 047335 DAC STORE
06717 447335 PNXT ISZ STORE
06720 207335 LAC STORE
06721 106763 JMS GOPNCH /PUNCH CHAR. IN AC 10-17
06722 207336 LAC SPCE /(SPCE)=240
06723 106763 JMS GOPNCH /PUNCH SPACE
06724 207335 LAC STORE
06725 547337 SAD K332 /DONE WITH ALPHABET IF EQUAL
06726 741000 SKP
06727 606717 JMP PNXT /PUNCH MORE CHARS.
06730 207331 LAC K257
06731 047335 PNXTA DAC STORE
06732 447335 ISZ STORE
06733 207335 LAC STORE
06734 106763 JMS GOPNCH
06735 207336 LAC SPCE
06736 106763 JMS GOPNCH
06737 207335 LAC STORE
06740 547332 SAD K271
06741 741000 SKP
06742 606732 JMP PNXTA
06743 207340 LAC KCRLF /(KCRLF)=CR,LF
06744 047335 DAC STORE
06745 106763 JMS GOPNCH /PUNCH CR
06746 207335 LAC STORE
06747 107232 JMS ROTAT9 /ROTATE 9 RIGHT
06750 106763 JMS GOPNCH /PUNCH LF
06751 777770 IAK -14
06752 047335 DAC STORE
06753 750001 CLA:OMA /AC = 777777
06754 700010 /CLEAR AC WITH MR 14
06755 740200 SZ#
06756 740040 E647 HALT /ERROR. EVENT TIME 1 DIDN'T
/CLEAR AC.
/PUNCH 8 FRAMES OF 0'S

06757 106763 JMS GOPNCH
06760 447335 ISZ STORE
06761 606753 JMP .-4
06762 606715 JMP PNXT-2 /START NEW LINE
.EJECT

```

06763	000000	GOPNTH	PSA	
06764	700004		LAC	
06765	750004		AND K4K	/MASK ACS 6
06766	507547		SZA	/IF A 1, DON'T USE CNTA OR CNTB
06767	740200		JMP RTNIT	/RETURN TO INST. TEST
06770	606562		ISZ CNTA	/CNTA=PUNCH SELECTED
06771	447327		LAC CNTB	
06772	207330		SZA	/0=WAIT FOR PI
06773	740200		JMP SUB1	/1=SFLECT READER AGAIN
06774	607031		JMP RTNIT	/RETURN TO INST. TEST
06775	606562			
		/READ PUNCHED INFO		
		/		
07776	750004	READA	LAS	
07777	507546		AND K3K	/MASK ACS 7 AND 8
07000	740200		SZA	/IF EITHER IS A 1, READ FULL SPEED
07001	607112		JMP READB	
07002	750004		LAS	
07003	507547		AND K4K	/MASK ACS 6
07004	741200		SNA	/IF A 1, CLEAR READER FLAG. NO
				/MORE PI'S FROM READER
07005	607010		JMP .+<	
07006	700112		RRR	/CLEAR READER FLAG
07007	606562		JMP RTNIT	/RETURN TO INST. TEST AND
				/WAIT FOR PUNCH PI
07010	700112		RRR	/READ ONE
07011	740200		SZA	/0=NO DATA IN READER YET
07012	607017		JMP ZRONOT	
07013	207327		LAC CNTA	/SEF IF PUNCH IS SELECTED
07014	740200		SZA	
07015	607031		JMP SUB1	/YES, SUBTRACT FROM CNTA
07016	607027		JMP TADD1	/READER SELECTED
07017	447326	ZRONOT	ISZ TTIN	/STORE CHAR. IN TTY BIN
07020	067326		DAC* TTIN	
07021	207326		LAC TTIN	
07022	546677		SAD ENDBIN	/CHECK FOR 52 CHARACTERS STORED
07023	607037		JMP SETTY	/DNF, SFTUP TO PRINT
07024	207327		LAC CNTA	
07025	740200		SZA	
07026	607031		JMP .+<	
		/		
07027	447330	TADD1	ISZ CNTB	
07030	606562		JMP RTNIT	/RETURN TO INST. TEST
07031	777777	SUB1	LAW -1	
07032	347327		TAD CNTA	/(CNTA)-1
07033	047327		DAC CNTA	
07034	147330		D7M CNTB	
07035	700104		RSA	
07036	606562		JMP RTNIT	/RETURN TO INST. TEST
		/		
		.EJECT		

```

07037 700201 SETTY PSF /WAIT FOR PUNCH
07040 607037 JMP .-1
07041 700202 PCF /CLEAR PUNCH FLAG
07042 700402 TCF /CLEAR TTY FLAG
07043 207311 LAC BRFAK
07044 547536 SAC K12A
07045 607070 JMP BU36
07046 447311 ISZ BRFAK
07047 206676 LAC DATARL /RESTORE INPUT/OUTPUT POINTERS
07050 047326 DAC TTIN /READER TO BIN
07051 206700 LAC OUTTOP
07052 047325 DAC TTOUT /RIN TO TTY
07053 447326 XFR1 ISZ TTIN /TRANSFER BUFFER A TO BUFFER B
07054 227326 LAC* TTIN
07055 447325 ISZ TTOUT
07056 067325 DAC* TTOUT
07057 207325 LAC TTOUT
07060 546701 SAD ENDOUT /DONE IF EQUAL TO TTBUFR+63
07061 741000 SKP
07062 607053 JMP XFR1 /TRANSFER ANOTHER
07063 206676 LAC DATARL /RESTORE BUFFER POINTERS
07064 047326 DAC TTIN
07065 206700 LAC OUTTOP /TTBUFR-1
07066 047325 DAC TTOUT
07067 606641 JMP TTYINT+3 /REGIN PRINTING

/
PUN6 LAC GOPNCH
07070 206763 SAD K647 /PUNCH 0'S ONLY AT END OF BLOCK
07071 547111 SKP
07072 741000 JMP XFR1-4 /((GOPNCH) NOT=E647+2
07073 607047 LAW -6
07074 777772 DAC BRFAK /FRAME COUNTER
07075 047311 LAW -1 /AC=777777
07076 777777 /CLEAR AC WITH BIT 14
07077 700010 SZA
07100 740200 E648 HALT /ERROR. MB14 DIDN'T CLEAR AC
07101 740040 PSA /PUNCH BLANK FRAME
07102 700204 PSF
07103 700201 JMP .-1
07104 607103 ISZ BRFAK
07105 447311 JMP E648-3
07106 607076 PCF /CLEAR PUNCH FLAG
07107 700202 JMP XFR1-4 /SETUP TO PRINT
07110 607047

/
K647 F647+2
/
.EJECT

```

/READR ROUTINE IS USED ONLY WHEN PUNCH IS INHIBITED
 /BY ACS 7 OR 8. OR BOTH, READER RUNS AT FULL SPEED.

```

/
07112 750004 READR LAS
07113 507547 AND K4K /MASK ACS 6
07114 741200 SNA /IF A 1, DON'T READ
07115 607120 JMP .+4
07116 700112 RRR /CLFAR READER FLAG
07117 606562 JMP RTNIT /RETURN TO INST. TEST
07120 700112 RRR /GET CHAR. FROM BUFFER.
07121 741200 SNA /0 = NO DATA IN READER YET.
07122 607130 JMP SELECT /SELECT READER AGAIN
07123 447326 ISZ TTIN /BUFFER POINTER +1
07124 067326 DAC* TTIN /STORE CHAR. IN TTBUFA
07125 207326 LAC TTIN
07126 546677 SNO ENDBIN /CHECK FOR 52 CHARS. STORED
07127 607132 JMP .+4 /TTBUFA IS FULL
07130 700104 SELECT RSA /SELECT READER
07131 606562 JMP RTNIT /RETURN TO INST. TEST

/
07132 206676 LAC DATARL
07133 047326 DAC TTIN /RESTORE TTBUFA POINTER
07134 750004 LAS
07135 507550 AND K6K /MASK ACS 6 AND 7
07136 740200 SZA /IF EITHER A 1, DON'T PRINT
07137 607130 JMP SELECT /SELECT READER AGAIN
07140 607042 JMP SETTY+3 /SETUP TO PRINT

/
/SERVICE NO TAPE CONDITIONS
/
07141 000000 RNFLG 0
07142 507543 AND K1K /CHECK FOR READER NO TAPE
07143 740200 SZA
07144 607151 JMP .+5 /READER
07145 760320 LAW 320 /PUNCH NO TAPE
07146 247341 XOR K520K
07147 047351 DAC NTFLG+1
07150 607153 JMP OUTFLG /PRINT R OR P NO TAPE
07151 760322 LAW 322 /READER NO TAPE
07152 607146 JMP .-4

/
07153 207350 OUTFLG LAC NTFLG
07154 047312 DAC WDCNT
07155 107241 JMS CRIF /CR,IF
07156 447312 ISZ WDCNT
07157 227312 LAC* WDCNT
07160 741200 SNA
07161 607166 JMP CLRFLG
07162 107210 JMS TLSSF
07163 107232 JMS ROTAT9
07164 107210 JMS TLSSF
07165 607156 JMP OUTFLG+3
07166 107241 CLRFLG JMS CRLF /CR,IF
07167 627141 JMP* RNFLG /RETURN TO SEQUENCE
  
```

BASEFX9 PAGE 110

.EJECT

07170	000000
07171	777440
07172	047312
07173	750000
07174	700204
07175	700201
07176	607175
07177	447312
07200	607174
07201	627170
07202	000000
07203	777777
07204	700204
07205	700201
07206	607205
07207	627202

/PUNCH LEADER

/
PNI E R 0
LAW -340
DAC WDCNT
CLA
PSA
PSE
JMP .-1
ISZ WDCNT
JMP .-4
JMP* PNLEDR

/
PNMARK

0
LAW -1
PSA
PSE
JMP .-1
JMP* PNMARK
.EJECT

/EXIT

```

/PRINT A CHARACTER
/
TLSSF      0
           047315  DAC MOVED
           207630  LAC WORK
           740010  RAL
           740100  SMA /CHECK TTY FLAG
           607220  JMP .+3
           700401  TSF
           607216  JMP .-1 /WAIT FOR FLAG
           207315  LAC MOVED
           700406  TLS
           700401  TSF
           607222  JMP .-1
           207630  LAC WORK /CLEAR TTY IF BIT 1 = 0
           740010  RAL
           740100  SMA
           700402  TCF
           207315  LAC MOVED
           627210  JMP* TLSSF

/ROTATE 9 RIGHT
/
ROTAT9     0
           RTR;      RTR;      RTR
           RTR;      RAR
           JMP*      ROTAT9

/CARRIAGE RETURN, LINEFEED
/
CRLF       0
           LAW 215 /CR
           JMS TLSSF
           SADR .+2
           JMP* CRLF /EXIT
           LAW 212 /LF
           JMP CRLF+2
           .EJECT

```

07210 000000
07211 047315
07212 207630
07213 740010
07214 740100
07215 607220
07216 700401
07217 607216
07220 207315
07221 700406
07222 700401
07223 607222
07224 207630
07225 740010
07226 740100
07227 700402
07230 207315
07231 627210

07232 000000
07233 742020
07234 742020
07235 742020
07236 742020
07237 740020
07240 627232

07241 000000
07242 760215
07243 107210
07244 547246
07245 627241
07246 760212
07247 607243

		/PRINT "COMPLETE"	
		/	
07250	000000	PINOT	0
07251	207634		LAC WORK4
07252	547536		SAD K12A
07253	741000		SKP
07254	627250		JMP* PINOT
07255	147634		DZM WORK4
07256	157634		DZM WORK4+10000
07257	207273		LAC COMPA
07260	040014		DAC 14
07261	107241		JMS CRIF
07262	220014		LAC* 14
07263	741200		SNA
07264	607271		JMP .+6
07265	107210		JMS TLSSF
07266	107232		JMS ROTAT9
07267	107210		JMS TLSSF
07270	607262		JMP .-6
07271	107241		JMS CRLF
07272	627250		JMP* PINOT
		/	
07273	007342	COMPA	COMP
			.EJECT

		/PASS COUNTER
		/PRINT IF EQUAL TO 10
		/START PROGRAM
		/PRINT COMPLETE
		/DONE PRINTING IF 0
		/PRINT 1 CHAR
		/PRINT 2ND
		/GET NEXT PAIR
		/CR, LF

```

/CONSTANT TABLE FOR CHECKERBOARD AND PI
/SERVICE ROUTINES
/
LIMITA      752525      /DELIMITER
WC16        0
WC12        0
LLREF        0
KPAT        037700
           740076
           037701
           037707
           037700      KPAT      037700
           000000      PATR        0
           000000      PATW        0
           000070      RGNLO      SEQUFN
           010070      RGNHI      SEQUFN+10000
           000000      RREF        0
           000000      WDCNT        0
           777776      WC256      777776
           000000      MOVES        0
           000000      MOVE        0
           777777      RITSUP      777777
           007777      K7777      7777
           100000      K100K      100000
           001400      K140K      1400
           500000      K500K      500000
           000777      K777       777
           777611      K167       777611
           000000      TTOUT        0
           000000      TTIN         0
           000000      CNTA         0
           000000      CNTR         0
           000257      K257        257
           000271      K271        271
           000300      K300        300
           000301      K301        301
           000000      STORE        0
           000240      SPCE        240
           000332      K332        332
           212215      KCRLE      212215
           520000      K520K      520000
           .EJECT

```

```

07274      752525
07275      000000
07276      000000
07277      000000
07300      037700
07301      740076
07302      037701
07303      740077
07304      037700
07305      000000
07306      000000
07307      000070
07310      010070
07311      000000
07312      000000
07313      777776
07314      000000
07315      000000
07316      777777
07317      007777
07320      100000
07321      001400
07322      500000
07323      000777
07324      777611
07325      000000
07326      000000
07327      000000
07330      000000
07331      000257
07332      000271
07333      000300
07334      000301
07335      000000
07336      000240
07337      000332
07340      212215
07341      520000

```

/
/PRINT ROUTINE CONSTANTS
/"COMPLETE"

07342 007342
07343 317303
07344 320315
07345 305314
07346 305324
07347 000000

/
COMP .
317303: 320315: 305314: 305324: 0

/
/R OR P NO TAPE
/
NTFLG .

07350 007350
07351 000000
07352 317316
07353 324240
07354 320301
07355 240305
07356 207207
07357 000000

0
317316: 324240: 320301
240305: 207207
0

/
/TTY BIN

07360

/TTBUFA .BLOCK 64 /READER BUF = 52 LOCS. (DECIMAL)

07444

/TTBUFB .BLOCK 64 /TTY BUF = 52 LOCS. (DECIMAL)
.EJECT

/CONSTANT AND ERROR TABLES. NOT MODIFIED WHEN IN HI 4K

07530	000000	K0	0
07531	000001	K1	1
07532	000002	K2	2
07533	000004	K4	4
07534	000010	K10	10
07535	000011	K11	11
07536	000012	K12A	12
07537	000100	K100	100
07540	000020	K20	20
07541	000022	K22	22
07542	000040	K40	40
07543	001000	K1K	1000
07544	000400	K400	400
07545	002000	K2K	2000
07546	003000	K3K	3000
07547	004000	K4K	4000
07550	006000	K6K	6000
07551	000200	K200	200
07552	040000	K40K	40000
07553	400000	K400K	400000
07554	400002	K400K	400002
07555	010000	K10K	10000
07556	020000	K20K	20000
07557	200000	K200K	200000
07560	600000	K600K	600000
07561	700000	K700K	700000
07562	002021	K2021	2021
07563	002120	K2120	2120
07564	111111	K1S	111111
07565	222222	K2S	222222
07566	333333	K3S	333333
07567	444444	K4S	444444
07570	555555	K5S	555555
07571	666666	K6S	666666
07572	777777	K7S	777777
07573	011111	K51S	11111
07574	012222	K12S	12222
07575	013333	K13S	13333
07576	014444	K14S	14444
07577	015555	K15S	15555
07600	016666	K16S	16666
07601	017777	K17S	17777
07602	002525	K2525	2525
07603	005252	K5252	5252
07604	252525	K010	252525
07605	525252	K101	525252
07606	525253	K53	525253
07607	077777	K37S	077777

.EJECT

07610	700042	K7X42	700042
07611	700002	K7XX2	700002
07612	760002	K76X2	760002
07613	100002	K1XX2	100002
07614	604002	K6X42	604002
07615	344002	K344X2	344002
07616	741000	KSKP	SKP
07617	750000	KCLA	CLA
07620	777776	M1	777776
07621	777773	M4	777773
07622	777737	M40	777737
07623	777377	M400	777377
07624	773777	M4K	773777
07625	737777	M40K	737777
07626	377777	M400K	377777
		/	
07627	000000	RJCNT	0
07630	000000	WORK	0
07631	000000	WORK1	0
07632	000000	WORK2	0
07633	000000	WORK3	0
07634	000000	WORK4	0
07635	000000	IJADR	0
07636	000000	AUTN0T	0
07637	000000	TCLK	0
07640	740010	XCTRAL	RAL
07641	740001	AUTCMA	CMA
07642	000000	SAVAC	0
07643	000000	RJMP	0
07644	420015	LAWAUT	XCT* 15
07645	777777	LAWFUL	LAW 17777
07646	120015	JMSAUT	JMS* 15
07647	740040	KHALT	740040
07650	620012	JMPAUT	JMP* 12
07651	200000	SAV4	LAC 0
07652	741400	KSZL	741400
07653	740400	KSNL	740400
07654	123456	RANCON	123456
07655	654321	RANTRL	654321
07656	361416		361416
07657	055363		055363
07660	546060		546060
07661	243035		243035
07662	762572		762572
07663	453237		453237
07664	150214		150214
07665	000000		0

.EJECT

/ERROR TABLES

```

/
07666 000000 JMPRET 0 /JMP 22
07667 000000 J111 0 /JMP 11111 (E509)
07670 000000 J222 0 /JMP 12222 (E510)
07671 000000 J333 0 /JMP 13333 (E5111)
07672 000000 J444 0 /JMP 14444 (E512)
07673 000000 J555 0 /JMP 15555 (E513)
07674 000000 J666 0 /JMP 16666 (E514)
07675 000000 J777 0 /JMP 17777 (E515)
07676 000000 J525 0 /JMP 15252 (E516)
07677 000000 J252 0 /JMP 12525 (E517)
07700 000000 CAL0 0 /CAL FROM 17757 EXT, LINK = 0 (E518)
07701 000000 CAL1 0 /CAL FROM 17757, LINK = 1 (E520)
07702 000000 JSM71 0 /JMS FROM 07777 TO 11111 (E522)
07703 000000 JSM72 0 /JMS FROM 07776 TO 12222 (E524)
07704 000000 JSM73 0 /JMS FROM 07775 TO 13333 (E526)
07705 000000 JSM74 0 /JMS FROM 07774 TO 14444 (E528)
07706 000000 JSM75 0 /JMS FROM 07773 TO 15555 (E530)
07707 000000 JSM76 0 /JMS FROM 07772 TO 16666 (E532)
07710 000000 JSM77 0 /JMS FROM 07771 TO 17777 (E534)
07711 000000 JS252 0 /JMS FROM 12525 TO 15252 (E536)
07712 000000 JS525 0 /JMS FROM 15252 TO 12525 (E538)
07713 000000 JSSS 0 /JMS SERIES TEST (E540)
/
07714 000000 XCT11 0 /XCT JMS. FROM 11111 XCT (16666) (E562)
07715 000000 XCT12 0 /XCT JMS. FROM 12222 XCT (15555) (E564)
07716 000000 XCT13 0 /XCT JMS. FROM 13333 XCT (14444) (E566)
07717 000000 XCT17 0 /XCT J.S FROM 07776 XCT (17776) (E568)
07720 000000 XCT125 0 /XCT JMS. FROM 12525 XCT (15252)
/
07721 000000 JST77 0 /JMS* 07777 (E592)
07722 000000 JST66 0 /JMS* 16666 (E594)
07723 000000 JST55 0 /JMS* 15555 (E596)
07724 000000 JST44 0 /JMS* 14444 (E598)
07725 000000 AUTJMP 0 /JMP* 12 (AUTO-INDEX) (E620)
07726 000000 AUTJMS 0 /JMS* 17 (AUTO-INDEX) (E633)
07727 752525 752525
000000 .END
NO ERROR LINES

```

ABRATS	02530
ADIFAC	01524
ADIFAC1	01772
ADIFDON	03030
ADJUST	06312
AMPRPT	02571
AMSUM	03025
AMNSB	02433
AMNSRT	02467
ANFAC	01306
ANFI	05517
ANFI	03017
APISRT	02454
APIUSB	02417
APPS	03016
AURJMP	06101
AUTCMA	07641
AUTJMP	07725
AUTJMS	07726
AUTNOT	07636
AUTOIN	05520
AUTR	05665
AUTRFT	06077
AUTRFT	05777
AUTRJM	06100
BDATA	06354
BFIN	00022
BGNACN	06500
BGHT	07310
BGALO	07307
BISITU	03036
BITSHP	07316
BITS1	03055
BITS2	03070
BMMRT	02543
BMSUM	03024
BMINSA	02425
BMSAT	02515
BNEG	03021
BPS	03020
BREAK	07311
CAI 0	07700
CAI 1	07701
CHKBRD	06154
CKI ODP	06363
CKIP	03011
CKNO	06126
CLKINT	06575
CLKSFT	06621
CLCF	700004
CLON	700044
CLRFIG	07166
CLSF	700001
CONA	07327
CONR	07330

COMP	07342
COMPB	07273
COMPCHG	07767
CRIF	07241
MACAC	03341
MACARL	06676
IRXY	00204
IRXX	00207
IRXXX	00215
ZMAC	03222
ENBRIN	06677
ENBOUT	06701
ENCTRL	06124
ENTST	06406
ERROR	06340
E1	00002
E113	00644
E114	00671
E115	00705
E116	00721
E140	00747
E141	00775
E142	01012
E143	01027
E162	01033
E163	01037
E164	01043
E165	01047
E166	01053
E167	01057
E168	01063
E169	01067
E170	01072
E206	01103
E207	01106
E208	01111
E209	01114
E210	01117
E211	01121
E212	01125
E213	01130
E214	01133
E215	01136
E216	01141
E217	01144
E218	01146
E219	01153
E220	01155
E221	01162
E222	01164
E223	01171
E224	01173
E225	01200
E226	01202
E24	00140

E25	00144
E258	01212
E259	01214
E26	00150
E260	01220
E261	01222
E262	01226
E263	01230
E264	01234
E265	01236
E266	01243
E267	01245
E268	01252
E269	01254
E27	00154
E270	01261
E271	01263
E272	01270
E273	01301
E274	01311
E275	01315
E276	01321
E277	01326
E278	01342
E279	01344
E28	00164
E280	01354
E281	01361
E282	01366
E283	01372
E284	01405
E285	01415
E286	01417
E287	01424
E288	01426
E289	01433
E29	00170
E290	01435
E291	01443
E292	01445
E293	01451
E294	01453
E295	01457
E296	01461
E297	01467
E298	01471
E299	01477
E30	00174
E300	01501
E301	01511
E302	01517
E303	01531
E304	01533
E305	01541
E306	01543

E307	01551
E308	01553
E309	01561
E31	00200
E310	01563
E311	01571
E312	01573
E313	01601
E314	01603
E315	01611
E316	01613
E317	01621
E318	01623
E319	01631
E32	00227
E320	01633
E321	01642
E322	01644
E323	01653
E324	01655
E325	01664
E326	01666
E327	01675
E328	01677
E329	01706
E33	00232
E330	01710
E331	01717
E332	01721
E333	01730
E334	01732
E335	01741
E336	01743
E337	01752
E338	01754
E339	01763
E34	00236
E340	01765
E347	02000
E348	02002
E349	02011
E35	00241
E350	02013
E351	02022
E352	02024
E353	02033
E354	02035
E355	02044
E356	02046
E357	02055
E358	02057
E359	02066
E36	00245
E360	02070
E361	02077

E32	02101
E33	02110
E34	02112
E35	02121
E36	02123
E37	02132
E38	02134
E39	02143
E37	00250
E370	02145
E371	02154
E372	02156
E373	02165
E374	02167
E375	02176
E376	02200
E377	02207
E378	02211
E379	02220
E38	00254
E380	02222
E381	02231
E382	02233
E383	02242
E384	02244
E385	02253
E386	02255
E387	02264
E388	02266
E389	02275
E39	00257
E390	02277
E391	02306
E392	02310
E393	02316
E394	02320
E395	02326
E396	02330
E397	02336
E398	02340
E399	02353
E40	00262
E400	02355
E401	02461
E402	02465
E403	02474
E404	02500
E405	02507
E406	02513
E407	02535
E408	02541
E409	02550
E41	00265
E410	02554
E411	02563

E412	02567
E413	02576
E414	02602
E415	02610
E416	02614
E417	02624
E418	02630
E419	02641
E420	02645
E421	02657
E422	02663
E423	02676
E424	02702
E425	02716
E426	02722
E427	02737
E428	02743
E429	02761
E430	02774
E431	02765
E432	03062
E433	03066
E434	03075
E435	03101
E436	03123
E437	03126
E438	03131
E439	03135
E440	03142
E441	03300
E442	03146
E443	03152
E444	03157
E445	03176
E446	03215
E447	03226
E448	03233
E449	03240
E450	03245
E451	03252
E452	03303
E453	03257
E454	03264
E455	03271
E456	03276
E457	03303
E458	03311
E459	03313
E460	03316
E461	03326
E462	03334
E463	03307
E464	03345
E465	03352

462	03357
463	03364
464	03371
465	03376
466	03403
467	03410
468	03415
469	03427
47	00313
470	03437
471	03450
472	03454
473	03461
474	03465
475	03472
476	03476
477	03503
478	03507
479	03514
48	00316
480	03520
481	03525
482	03531
483	03536
484	03541
485	03546
486	03551
487	03556
488	03561
489	03566
49	00322
490	03571
491	03576
492	03601
493	03606
494	03611
495	03616
496	03621
497	03626
498	03631
499	03636
50	00325
500	03641
501	03661
502	03671
503	03715
504	03723
505	03731
506	03740
507	03747
508	03776
509	04004
51	00330
510	04015
511	04026

E512	04037
E513	04050
E514	04061
E515	04072
E516	04103
E517	04114
E518	04145
E519	04163
E519A	04166
E520	00335
E520	04171
E521	04204
E521A	04207
E522	04223
E523	04240
E524	04243
E525	04261
E526	04264
E527	04302
E528	04305
E529	04323
E530	00342
E530	04326
E531	04344
E532	04347
E533	04365
E534	04370
E535	04410
E536	04422
E537	04437
E538	04442
E539	04457
E540	00347
E540	04462
E541	04465
E542	04467
E543	04471
E544	04475
E545	04505
E546	04514
E547	04523
E548	04531
E549	04603
E550	00355
E550	04606
E551	04613
E552	04615
E553	04617
E554	04623
E555	04630
E556	04634
E557	04642
E558	04644
E559	04651
E560	00364

560	04653
561	04661
562	04664
563	04701
564	04704
565	04721
566	04724
567	04741
568	04744
569	04761
57	00367
570	04764
571	05001
572	05020
573	05022
574	05063
575	05073
576	05103
577	05113
578	05123
579	05133
58	00374
580	05143
581	05153
582	05163
584	05201
585	05212
586	05223
587	05234
588	05245
589	05256
59	00401
590	05267
591	05300
592	05311
593	05327
594	05334
595	05352
596	05357
597	05375
598	05402
599	05420
60	00405
600	05454
601	05462
602	05477
603	05526
604	05532
605	05541
606	05545
607	05554
608	05560
609	05567
61	00410
610	05573

E611	05602
E612	05606
E613	05615
E614	05621
E615	05630
E616	05634
E617	05642
E618	05645
E619	05651
E620	00413
E620	05654
E621	05671
E622	05702
E623	05706
E624	05716
E625	05722
E626	05732
E627	05734
E628	05740
E629	05744
E630	00416
E630	05753
E631	05757
E632	05763
E633	05766
E634	06003
E635	06007
E636	06015
E637	06021
E638	06024
E639	06033
E640	00421
E640	06046
E641	06052
E642	06350
E643	06543
E644	06570
E645	06647
E646	06652
E647	06756
E648	07101
E650	00427
E660	00434
E670	00440
E680	00446
E690	00451
E700	00455
E710	00460
E720	00464
E730	00467
E740	00473
E750	00477
E760	00502
E770	00505
E780	00512

E70	00516
E80	00522
E81	00526
E82	00531
E83	00535
E84	00540
E85	00544
E86	00550
E87	00553
E88	00560
E89	00566
E90	00573
E91	00603
E92	00613
SDATA	06352
SEARAN	06102
SOPNCH	06763
MAIT	740040
LIADR	07635
LLINT	06574
IN-IT	00131
INITPI	00130
INIT4K	03750
INK52	05427
LOTST	00135
LS-AC	03444
MPAHT	07650
JMPRET	07666
JMPSEQ	04130
JMSAUT	07646
JMS11	05430
JST44	05440
JST55	05436
JST66	05433
JSM71	07702
JSM72	07703
JSM73	07704
JSM74	07705
JSM75	07706
JSM76	07707
JSM77	07710
JS-S	07713
JST44	07724
JST55	07723
JST66	07722
JST77	07721
JS1	04464
JS2	04466
JS252	07711
JS3	04470
JS4	04474
JS525	07712
J111	07667
J222	07670
J252	07677

J333	07671
J444	07672
J555	07676
J555	07673
J666	07674
J777	07675
KCALF	04220
KCAL0	04216
KCLA	07617
KCHLF	07340
KHALT	07647
KJS1	04573
KJS2	04574
KJS3	04575
KJS4	04576
KPAT	07300
KRH	700312
KSF	700301
KSKP	07616
KSNL	07653
KSYL	07652
K0	07530
K010	07604
K1	07531
K1K	07543
K1S	07564
K1XX2	07613
K1V	07534
K1VK	07555
K1V0	07537
K1V0K	07320
K1V000	04541
K1V1	07605
K11	07535
K11110	06074
K11111	06075
K12	05033
K12A	07536
K12S	07574
K12221	06072
K12222	06073
K13S	07575
K13332	06070
K13333	06071
K14S	07576
K1400	07321
K14443	06066
K14444	06067
K15S	07577
K15252	06076
K15253	05435
K15554	06064
K15555	06065
K16S	07600
K16665	06062

11666	06063
17S	07601
17776	06060
17777	06061
2	07532
2K	07545
2S	07565
2	07540
2K	07556
2A	07551
2VOK	07557
2V21	07562
2120	07563
2P	07541
2A	05037
2525	07602
257	07331
271	07332
3K	07546
3S	07566
300	07333
301	07334
332	07337
34	05043
344x2	07615
37S	07607
4	07533
4K	07547
4S	07567
40	07542
4K	07552
400	07544
4VOK	07553
4V2K	07554
415	04572
426	04566
5S	07570
5VOK	07322
51S	07573
520K	07341
5252	07603
53	07606
6K	07550
6S	07571
6X42	07614
6VOK	07560
647	07111
7S	07572
7XX2	07611
7X42	07610
7VOK	07561
71	04571
72	04563
73	04560
74	04555

K75	04552
K76	04547
K70X2	07612
K77	04544
K777	07323
K7777	07317
LACIN	05171
LACK	01207
LAWAIT	07644
LAWFUL	07645
LCNTA	06176
LCNTR	06200
LIMITA	07274
LLREG	07277
LOAD	06172
MAPAT	02556
MAPLMB	02502
MINSAB	02411
MINUSA	02371
MINUSR	02401
MOH	04131
MONFG	03050
MOBX	04001
MOVE	06421
MOVED	07315
MOVES	07314
MPAT	07304
MRINS	06515
MSKBIT	03026
MVHK	06510
MVCST	06451
MVRTN	06443
MOACPA	02604
M1	07620
M167	07324
M4	07621
M4K	07624
M4A	07622
M4K	07625
M400	07623
M400K	07626
MEVPAT	06332
NOPI	740000
NOPI2	740000
NOPI3	740000
NTFLC	07350
NTST	06325
OFLCH1	02627
OFLCH2	02644
OFLCH3	02662
OFLCH4	02701
OFLCH5	02721
OFLCH6	02742
OFLCH7	02764
OFLCH8	03065

RFICH9	03100
RFICK1	02464
RFICK2	02477
RFICK3	02512
RFICK5	02540
RFICK6	02553
RFICK7	02566
RFICK8	02601
RFICK9	02613
RPRAT	00225
RPRAT	00223
RUTFIG	07153
RUTTOP	06700
RASS2	03027
RATR	07305
RATT	06361
RATWD	07306
RCF	700202
RINOT	07250
RIGN	06572
RNIENR	07170
RNNARK	07202
RNSTRT	06702
RNXT	06717
RNXTA	06732
RREFADY	06657
RSA	700204
RS-	700244
RS-	700201
RUX6	07070
RANADD	02362
RANCON	07654
RANDEFX	06123
RANGFN	06133
RANTAD	06117
RANTRL	07655
RCALS0	04157
RCALS1	04200
RCAL0	04215
RCAL1	04217
RCF	700102
RCI00P	06247
RCNTA	06237
RCNTR	06243
RCRTN	06270
READ	06234
READA	06776
READR	07112
RFROM	06433
RJCNT	07627
RJMI1	05431
RJMI2	05434
RJMI3	05437
RJMI4	05441
RJMP	07643

RJMP1	04012
RJMP2	04023
RJMP3	04034
RJMP4	04045
RJMP5	04056
RJMP6	04067
RJMP7	04100
RJMP8	04111
RJMP9	04122
RJSS	04476
RJMS14	04433
RJMS15	04453
RJMS71	04234
RJMS72	04255
RJMS73	04276
RJMS74	04317
RJMS75	04340
RJMS76	04361
RJMS77	04404
RJSI1	05323
RJSI1X	05432
RJSI2	05346
RJSI3	05371
RJSI4	05414
RJSM25	04564
RJSM52	04567
RJSM71	04537
RJSM72	04542
RJSM73	04545
RJSM74	04550
RJSM75	04553
RJSM76	04556
RJSM77	04561
RJ111	04132
RJ222	04133
RJ252	04141
RJ333	04134
RJ444	04135
RJ525	04142
RJ555	04136
RJ666	04137
RJ777	04140
RNFLC	07141
ROTAT9	07232
RR	700112
RS*	700104
RS*	700144
RS*	700101
RS*25	04565
RS*52	04570
RS*71	04540
RS*72	04543
RS*73	04546
RS*74	04551
RS*75	04554

AS76	04557
AS77	04562
AT	00620
ATIT	06562
ATSS	00722
AXCT1	04675
AXCT2	04715
AXCT3	04735
AXCT4	04755
AXCT5	04775
ADAC	03120
ADAC	07642
AV3	06533
AV4	07651
AV5	06534
AV6	06535
SELECT	07130
SEDFEN	00070
SEFS01	02616
SEFS02	02632
SEFS03	02647
SEFS04	02665
SEFS05	02704
SEFS06	02724
SEFS07	02745
ETCLK	06604
ETTY	07037
SPPE	07336
SRVINT	06536
STORE	07335
SUR1	07031
SURNEG	03022
SURPOS	03023
ADAC	01412
AD1	07027
ADRAN	06150
IBTOP	06125
ICF	700402
ICK	07637
ILAW	01077
ILS	700406
ILSSF	07210
ISAUTO	05055
ISCAI	04143
ISORR	00161
ISF	700401
ISIMS	04221
ISXCT	04577
ITRUF A	07360
ITRUF B	07444
ITIN	07326
ITOUT	07325
ITYINT	06636
MLADJ	06322
SCLOP	06204

XCI2R	07276
XCI6	07275
XCI25A	07313
XCONT	07312
XOK	07630
XOKK1	07631
XOKK2	07632
XOKK3	07633
XOKK4	07634
XCTDAC	05053
XCTD7M	05516
XCTISZ	05052
XCTRAL	07640
XCTR12	05050
XCTTAD	05054
XCT11	07714
XCT12	07715
XCT12S	05047
XCT125	07720
XCT13	07716
XCT17	07717
XFR1	07053
XORAC	01351
XJMSI	05305
XTR11	05031
XTR12	05035
XTR13	05041
XTR17	05045
XTXCT	05442
XT1R	05032
XT11S	05030
XT12S	05034
XT13S	05040
XT17S	05044
XT2R	05036
XT3R	05042
XT4R	05046
XT5R	05051
XRONOT	07017

E-1	00002
E-1M	00022
E-1UFM	00070
E-1TPI	00130
E-1IT	00131
E-1ST	00135
E-24	00140
E-25	00144
E-26	00150
E-27	00154
E-28RR	00161
E-29	00164
E-29	00170
E-30	00174
E-31	00200
E-3X	00204
E-3XX	00207
E-3XXX	00215
E-3AT	00223
E-3RAT	00225
E-3-	00227
E-34	00232
E-34	00236
E-35	00241
E-36	00245
E-37	00250
E-38	00254
E-39	00257
E-40	00262
E-41	00265
E-42	00271
E-43	00274
E-44	00300
E-45	00303
E-46	00307
E-47	00313
E-48	00316
E-49	00322
E-50	00325
E-51	00330
E-52	00335
E-53	00342
E-54	00347
E-55	00355
E-56	00364
E-57	00367
E-58	00374
E-59	00401
E-60	00405
E-61	00410
E-62	00413
E-63	00416
E-64	00421
E-65	00427
E-66	00434

E67	00440
E68	00446
E69	00451
E70	00455
E71	00460
E72	00464
E73	00467
E74	00473
E75	00477
E76	00502
E77	00505
E78	00512
E79	00516
E80	00522
E81	00526
E82	00531
E83	00535
E84	00540
E85	00544
E86	00550
E87	00553
E88	00560
E89	00566
E90	00573
E91	00603
E92	00613
E101	00620
E113	00644
E114	00671
E115	00705
E116	00721
RTSS	00722
E140	00747
E141	00775
E142	01012
E143	01027
E162	01033
E163	01037
E164	01043
E165	01047
E166	01053
E167	01057
E168	01063
E169	01067
E170	01072
FLAW	01077
E206	01103
E207	01106
E208	01111
E209	01114
E210	01117
E211	01121
E212	01125
E213	01130
E214	01133

-215	01136
-216	01141
-217	01144
-218	01146
-219	01153
-220	01155
-221	01162
-222	01164
-223	01171
-224	01173
-225	01200
-226	01202
LAOK	01207
-258	01212
-259	01214
-260	01220
-261	01222
-262	01226
-263	01230
-264	01234
-265	01236
-266	01243
-267	01245
-268	01252
-269	01254
-270	01261
-271	01263
-272	01270
-273	01301
NIAC	01306
-274	01311
-275	01315
-276	01321
-277	01326
-278	01342
-279	01344
NIAC	01351
-280	01354
-281	01361
-282	01366
-283	01372
-284	01405
TAIAC	01412
-285	01415
-286	01417
-287	01424
-288	01426
-289	01433
-290	01435
-291	01443
-292	01445
-293	01451
-294	01453
-295	01457
-296	01461

E297	01467
E298	01471
E299	01477
E300	01501
E301	01511
E302	01517
ADUAC	01524
E303	01531
E304	01533
E305	01541
E306	01543
E307	01551
E308	01553
E309	01561
E310	01563
E311	01571
E312	01573
E313	01601
E314	01603
E315	01611
E316	01613
E317	01621
E318	01623
E319	01631
E320	01633
E321	01642
E322	01644
E323	01653
E324	01655
E325	01664
E326	01666
E327	01675
E328	01677
E329	01706
E330	01710
E331	01717
E332	01721
E333	01730
E334	01732
E335	01741
E336	01743
E337	01752
E338	01754
E339	01763
E340	01765
ADUAC1	01772
E347	02000
E348	02002
E349	02011
E350	02013
E351	02022
E352	02024
E353	02033
E354	02035
E355	02044

3-6	02046
3-7	02055
3-8	02067
3-9	02066
3-0	02070
3-1	02077
3-2	02101
3-3	02110
3-4	02112
3-5	02121
3-6	02123
3-7	02132
3-8	02134
3-9	02143
370	02145
371	02154
372	02156
373	02165
374	02167
375	02176
376	02200
377	02207
378	02211
379	02220
3-0	02222
3-1	02231
3-2	02233
3-3	02242
3-4	02244
3-5	02253
3-6	02255
3-7	02264
3-8	02266
3-9	02275
390	02277
3-1	02306
3-2	02310
3-3	02316
3-4	02320
3-5	02326
3-6	02330
3-7	02336
3-8	02340
3-9	02353
4-0	02355
AVAND	02362
4INUSA	02371
4INUSB	02401
4INSAB	02411
APIUSB	02417
AMINSA	02425
AMINSB	02433
API SRT	02454
E4M1	02461
UFLCK1	02464

E412	02465
AMRSRT	02467
E413	02474
DFI CK2	02477
E414	02500
AMLR	02502
E415	02507
DFI CK3	02512
E416	02513
AMRSAT	02515
AMRATS	02530
E417	02535
DFI CK5	02540
E418	02541
AMMRT	02543
E419	02550
DFI CK6	02553
E410	02554
AMR PAT	02556
E411	02563
DFI CK7	02566
E412	02567
AMMRPT	02571
E413	02576
DFI CK8	02601
E414	02602
AMACPA	02604
E415	02610
DFI CK9	02613
E416	02614
SEMS01	02616
E417	02624
DFI CH1	02627
E418	02630
SEMS02	02632
E419	02641
DFI CH2	02644
E420	02645
SEMS03	02647
E421	02657
DFI CH3	02662
E422	02663
SEMS04	02665
E423	02676
DFI CH4	02701
E424	02702
SEMS05	02704
E425	02716
DFI CH5	02721
E426	02722
SEMS06	02724
E427	02737
DFI CH6	02742
E428	02743
SEMS07	02745

E 429	02761
E 430	02764
E 431	02765
E 432	02767
E 433	03011
E 434	03016
E 435	03017
E 436	03020
E 437	03021
E 438	03022
E 439	03023
E 440	03024
E 441	03025
E 442	03026
E 443	03027
E 444	03030
E 445	03036
E 446	03050
E 447	03055
E 448	03062
E 449	03065
E 450	03066
E 451	03070
E 452	03075
E 453	03100
E 454	03101
E 455	03120
E 456	03123
E 457	03126
E 458	03131
E 459	03135
E 460	03142
E 461	03146
E 462	03152
E 463	03157
E 464	03176
E 465	03215
E 466	03222
E 467	03226
E 468	03233
E 469	03240
E 470	03245
E 471	03252
E 472	03257
E 473	03264
E 474	03271
E 475	03276
E 476	03303
E 477	03311
E 478	03313
E 479	03316
E 480	03326
E 481	03334
E 482	03341
E 483	03345

E441	03352
E442	03357
E443	03364
E444	03371
E445	03376
E446	03403
E447	03410
E448	03415
E449	03427
E450	03437
ES7AC	03444
E471	03450
E472	03454
E473	03461
E474	03465
E475	03472
E476	03476
E477	03503
E478	03507
E479	03514
E480	03520
E481	03525
E482	03531
E483	03536
E484	03541
E485	03546
E486	03551
E487	03556
E488	03561
E489	03566
E490	03571
E491	03576
E492	03601
E493	03606
E494	03611
E495	03616
E496	03621
E497	03626
E498	03631
E499	03636
E500	03641
E501	03661
E502	03671
E503	03715
E504	03723
E505	03731
E506	03740
E507	03747
INIT4K	03750
E508	03776
OPX	04001
E509	04004
RJMP1	04012
E510	04015
RJMP2	04023

511	04026
RJ P3	04044
512	04037
RJ P4	04045
513	04050
RJ P5	04056
514	04061
RJ P6	04067
515	04072
RJ P7	04100
516	04103
RJ PR	04111
517	04114
RJ P9	04122
JMHSF0	04130
00	04131
RJ11	04132
RJ22	04133
RJ33	04134
RJ44	04135
RJ55	04136
RJ66	04137
RJ77	04140
RJ252	04141
RJ525	04142
ISCAI	04143
518	04145
RCALS0	04157
519	04163
519A	04166
520	04171
RCALS1	04200
521	04204
521A	04207
RCAL0	04215
RCAL0	04216
RCAL1	04217
RCALF	04220
ISJMS	04221
522	04223
RJMS71	04234
523	04240
524	04243
RJMS72	04255
525	04261
526	04264
RJMS73	04276
527	04302
528	04305
RJMS74	04317
529	04323
530	04326
RJMS75	04340
531	04344
532	04347

KJS76	04361
543	04365
544	04370
KJS77	04444
545	04410
546	04422
KJS14	04433
547	04437
548	04442
KJS15	04453
549	04457
540	04462
JS1	04464
541	04465
542	04466
542	04467
JS4	04470
543	04471
JS4	04474
544	04475
KJSM5	04476
545	04505
546	04514
547	04523
548	04531
KJSM71	04537
SM71	04540
41000	04541
KJSM72	04542
SM72	04543
77	04544
KJSM73	04545
SM73	04546
76	04547
KJSM74	04550
SM74	04551
75	04552
KJSM75	04553
SM75	04554
74	04555
KJSM76	04556
SM76	04557
74	04560
KJSM77	04561
SM77	04562
74	04563
KJSM25	04564
SM25	04565
446	04566
KJSM52	04567
SM52	04570
71	04571
415	04572
KJS1	04573
KJS2	04574

0J 3	04575
0J 4	04576
ESICT	04577
-509	04603
-500	04606
-501	04613
-502	04615
-503	04617
-504	04623
-505	04630
-506	04634
-507	04642
-508	04644
-509	04651
-500	04653
-501	04661
-502	04664
XCT1	04675
-503	04701
-504	04704
XCT2	04715
-505	04721
-506	04724
XCT3	04735
-507	04741
-508	04744
XCT4	04755
-509	04761
-510	04764
XCT5	04775
-511	05001
-512	05020
-513	05022
XT1S	05030
XT11	05031
XT1R	05032
-12	05033
XT12S	05034
XT12	05035
XT1R	05036
X23	05037
XT13S	05040
XT13	05041
XT1R	05042
X34	05043
XT17S	05044
XT17	05045
XT4R	05046
XCT12S	05047
XCTR12	05050
XT5R	05051
XCTISZ	05052
XCTDAC	05053
XCTAD	05054
ISAUTO	05055

E574	05063
E575	05073
E576	05103
E577	05113
E578	05123
E579	05133
E580	05143
E581	05153
E582	05163
LAJIN	05171
E584	05201
E585	05212
E586	05223
E587	05234
E588	05245
E589	05256
E590	05267
E591	05300
XTJMSI	05305
E592	05311
RJS14	05323
E593	05327
E594	05334
RJS12	05346
E595	05352
E596	05357
RJS13	05371
E597	05375
E598	05402
RJS14	05414
E599	05420
INK52	05427
JMS11	05430
RJ14	05431
RJS11X	05432
JS166	05433
RJ12	05434
X15253	05435
JS155	05436
RJ13	05437
JS144	05440
RJ14	05441
XTXCT	05442
E600	05454
E601	05462
E602	05477
XCFD7M	05516
ANI	05517
AUTOIN	05520
E603	05526
E604	05532
E605	05541
E606	05545
E607	05554
E608	05560

609	05567
610	05573
611	05602
612	05606
613	05615
614	05621
615	05630
616	05634
617	05642
618	05645
619	05651
620	05654
AUTP	05665
621	05671
622	05702
623	05706
624	05716
625	05722
626	05732
627	05734
628	05740
629	05744
630	05753
631	05757
632	05763
633	05766
AURF1	05777
634	06003
635	06007
636	06015
637	06021
638	06024
639	06033
640	06046
641	06052
17776	06060
17777	06061
16665	06062
16666	06063
15554	06064
15555	06065
14443	06066
14444	06067
13332	06070
13333	06071
12221	06072
12222	06073
11110	06074
11111	06075
15252	06076
AURF1	06077
AURJM	06100
AURJMP	06101
GENRAN	06102
RANTAD	06117

INDEX	06123
INSTR	06124
INTOP	06125
INTO	06126
INSTR	06133
INSTR	06150
INSTR	06154
LOAD	06172
LOAD	06176
LOAD	06200
LOAD	06204
LOAD	06234
LOAD	06237
LOAD	06243
LOAD	06247
LOAD	06270
LOAD	06312
LOAD	06322
LOAD	06325
LOAD	06332
LOAD	06340
LOAD	06350
LOAD	06352
LOAD	06354
LOAD	06361
LOAD	06363
LOAD	06406
LOAD	06421
LOAD	06433
LOAD	06443
LOAD	06451
LOAD	06500
LOAD	06510
LOAD	06515
LOAD	06533
LOAD	06534
LOAD	06535
LOAD	06536
LOAD	06543
LOAD	06562
LOAD	06570
LOAD	06572
LOAD	06574
LOAD	06575
LOAD	06604
LOAD	06621
LOAD	06636
LOAD	06647
LOAD	06652
LOAD	06657
LOAD	06676
LOAD	06677
LOAD	06700
LOAD	06701
LOAD	06702

ANCT	06717
ANCTA	06732
ANCT7	06746
ANCTH	06763
ANCTA	06776
ANCTOT	07017
ANCTD1	07027
ANCT1	07031
ANCTTY	07037
ANCT1	07053
ANCT6	07070
ANCT8	07101
ANCT7	07111
ANCTDR	07112
ANCTELECT	07130
ANCTLC	07141
ANCTFLG	07153
ANCTFLG	07166
ANCTEDR	07170
ANCTARK	07202
ANCTSE	07210
ANCTAT9	07232
ANCTF	07241
ANCTOT	07250
ANCTPA	07273
ANCTA	07274
ANCT16	07275
ANCT28	07276
ANCTFG	07277
ANCTAV	07300
ANCTPAT	07304
ANCTATR	07305
ANCTATWD	07306
ANCTALO	07307
ANCTGHT	07310
ANCTBREAK	07311
ANCTDINT	07312
ANCTC256	07313
ANCTOVES	07314
ANCTOVEN	07315
ANCTTISHP	07316
ANCT777	07317
ANCT10K	07320
ANCT1400	07321
ANCT500K	07322
ANCT777	07323
ANCT167	07324
ANCTTOUT	07325
ANCTTIN	07326
ANCTA	07327
ANCTB	07330
ANCT57	07331
ANCT71	07332
ANCT00	07333
ANCT01	07334

ST-RF	07335
SP-F	07336
X342	07337
SC-LF	07340
X520K	07341
20-FP	07342
FT-FC	07350
FT-UF-A	07360
FT-UF-B	07444
X0	07530
1	07531
2	07532
4	07533
11	07534
11	07535
X12A	07536
X170	07537
X20	07540
X22	07541
X40	07542
X1K	07543
X400	07544
X2K	07545
X3K	07546
X4K	07547
X6K	07550
X270	07551
X40K	07552
X470K	07553
X402K	07554
X10K	07555
X20K	07556
X200K	07557
X600K	07560
X700K	07561
X2021	07562
X2120	07563
X1S	07564
X2S	07565
X3S	07566
X4S	07567
X5S	07570
X6S	07571
X7S	07572
X51S	07573
X12S	07574
X13S	07575
X14S	07576
X15S	07577
X16S	07600
X17S	07601
X2025	07602
X5252	07603
X010	07604
X11	07605

5	07606
37S	07607
742	07610
7X2	07611
7X2	07612
1X2	07613
642	07614
344X2	07615
SKP	07616
CLA	07617
1	07620
4	07621
4	07622
400	07623
4K	07624
4K	07625
400K	07626
4JNT	07627
00K	07630
00K1	07631
00K2	07632
00K3	07633
00K4	07634
IIDR	07635
UINOT	07636
OLK	07637
CTRAL	07640
UTCMA	07641
AVAC	07642
RJMP	07643
LAWAIT	07644
LAWFIL	07645
JMSAIT	07646
XHLT	07647
JMPAIT	07650
AV4	07651
ASL	07652
ASNL	07653
AVCON	07654
AVNTRL	07655
JMPRFT	07666
111	07667
222	07670
333	07671
444	07672
555	07673
666	07674
777	07675
888	07676
999	07677
CAI 0	07700
CAI 1	07701
JSM71	07702
JSM72	07703
JSM73	07704

IS-74	07705
IS-75	07706
IS-76	07707
IS-77	07710
IS-252	07711
IS-25	07712
ISSS	07713
SCI11	07714
SCI12	07715
SCI13	07716
SCI17	07717
SCI125	07720
ISI77	07721
ISI66	07722
ISI55	07723
ISI44	07724
AUTJMP	07725
AUTJMS	07726
CLRF	700001
CLRF	700004
CLON	700044
KSF	700101
KCF	700102
KSA	700104
KRR	700112
KSH	700144
PSE	700201
PCE	700202
PSA	700204
PSR	700244
KSE	700301
KRR	700312
ISF	700401
ICE	700402
ILK	700406
POP1	740000
POP2	740000
POP3	740000
HAI T	740040