

MAINDEC-9A-D0BA-D

IDENTIFICATION

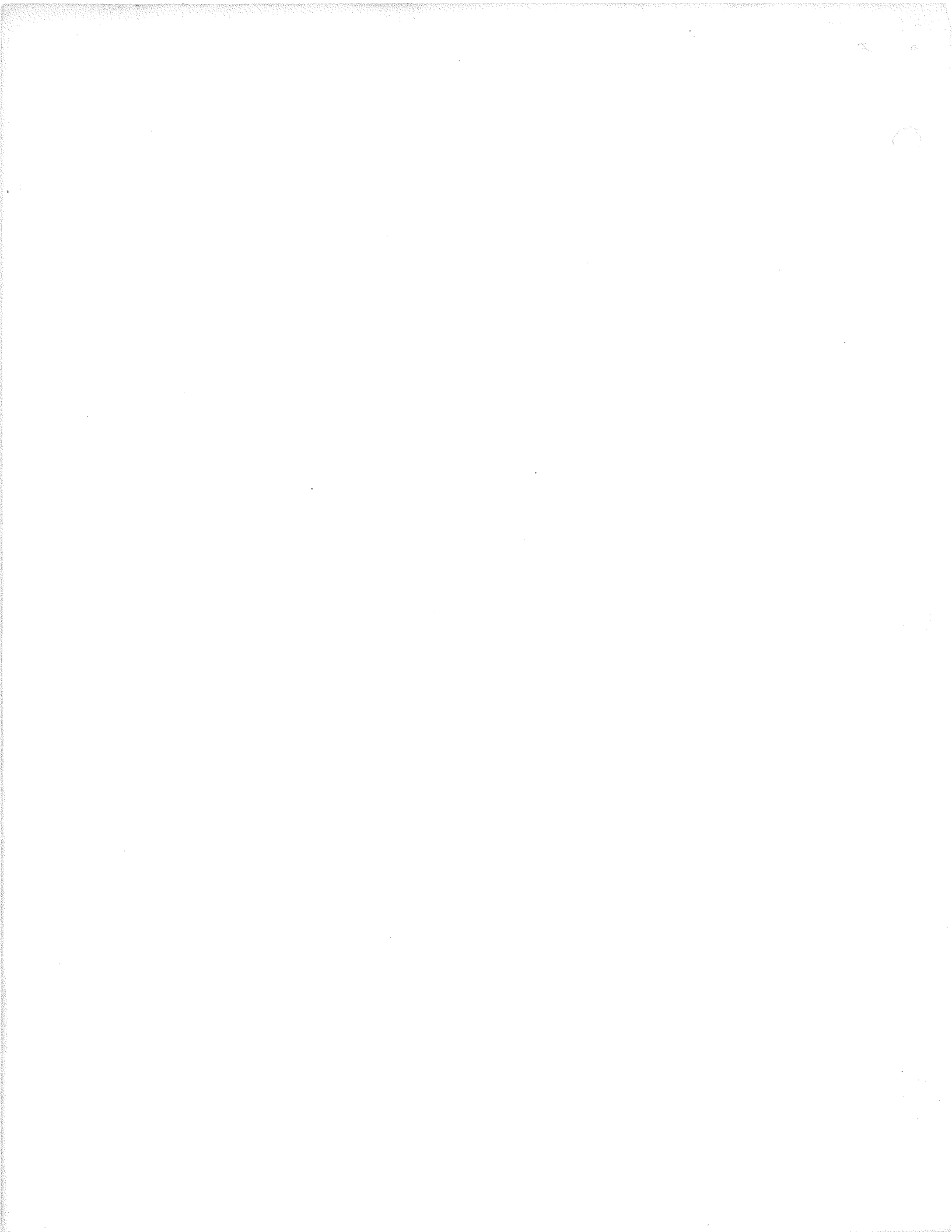
Product Code: MAINDEC-9A-D0BA-D

Product Name: ISZ Test

Date: May 15, 1967

Maintainer: Diagnostic Group

Author: Edward P. Steinberger



1. ABSTRACT

The ISZ Test checks the operation of the ISZ instruction of the PDP-9. Various checks of the ISZ instruction are made, including ISZ of  $777777_8$  to  $0_8$  on all memory locations, and ISZ of random numbers stored in random memory locations from random memory locations. Errors are indicated to the operator via the teleprinter.

2. REQUIREMENTS

2.1 Equipment

Standard PDP-9 computer

2.2 Storage

The program uses all of 8K memory for the program or as a test area. When the program resides in upper memory, it occupies from locations 16440 to 17710. The random ISZ portion of the test tests all locations below 16440.

2.3 Preliminary Programs

Instruction Test - Parts 1 and 2 (MAINDEC-9A-D01A-D and MAINDEC-9A-D02A-D)

3. LOADING PROCEDURE

3.1 Method

- a. Put HRI tape of program in reader.
- b. Set ADDRESS SWITCHES to 00100.
- c. Depress and release READ-IN key.

4. STARTING PROCEDURE

4.1 Control Switch Settings

The following is a table of ACCUMULATOR SWITCH settings and their action on the program:

<u>AC Switch</u>	<u>Set As</u>	<u>Action</u>
0	1	Halt on error
	0	Don't halt on error
1	1	Don't print errors
	0	Print errors

<u>AC Switch</u>	<u>Set As</u>	<u>Action</u>
2	1	Ring bell on error
	0	Ring bell after N passes
3	1	Loop on current conditions
	0	Don't loop on current conditions
4	1	Loop on current test
	0	Don't loop on current test
5	1	Save initial error conditions of random ISZ
	0	Don't save initial error conditions of random ISZ
6	1	Vary location of ISZ instruction
	0	Don't vary location of ISZ instruction
7	1	Vary location of number incremented
	0	Don't vary location of number incremented
8	1	Vary number incremented
	0	Don't vary number incremented

(Switches 6, 7, 8 operate in conjunction with 5; 3 supercedes 4)

N is an arbitrary number (initially 20000<sub>g</sub> for random ISZ's) which is controlled by the LAW-N instruction in location 17052 and may be changed at the operator's discretion.

#### 4.2 Starting Addresses

The starting address of the program is 00100. The restart addresses are 00100, 00144, 17000, 17052, and 17652 (see section 5.3).

#### 4.3 Program and/or Operator Action

- a. Set ADDRESS SWITCHES to 00100.
- b. Set ACCUMULATOR SWITCHES to desired positions (see section 4.1). Normal setting is 510000.
- c. Depress I/O RESET.
- d. Depress START.

### 5. OPERATING PROCEDURE

#### 5.1 Operational Switch Settings

See section 4.1

5.2 Subroutine Abstracts

None

5.3 Program and/or Operator Action

a. To put the program in the 'scope mode, the ACCUMULATOR SWITCHES should be set to 270000, (don't halt, don't print, bell after N passes, loop on current number (location), loop on current test, save error conditions).

b. To start program initially so that upper memory may be checked, start at location 00100.

c. To start program initially so that lower memory may be checked without checking upper memory, start at location 00144.

d. To restart program to check upper memory after program has moved, restart at 17652.

e. To restart program to check lower memory after program has moved, restart at 17000.

f. To restart program to check random ISZ's after program has moved, restart at 17052.

6. ERRORS

Unless AC switch 1 is a 1, all errors will be printed on the Teletype.

6.1 Error Halts and Description

<u>Location</u>	<u>Description</u>
00342	ISZ on upper memory did not skip
00404	Location in upper memory did not ISZ to 0
17466	ISZ on lower memory did not skip
17530	Location in lower memory did not ISZ to 0
17601	Random ISZ add failure

6.2 Error Recovery

6.2.1 To Repeat Failure - If AC switch 0 is a 1, the computer will halt on an error. To recover and repeat the failure, reset AC switches 0 to 5 as necessary (see section 4.1) and then depress CONTINUE key.

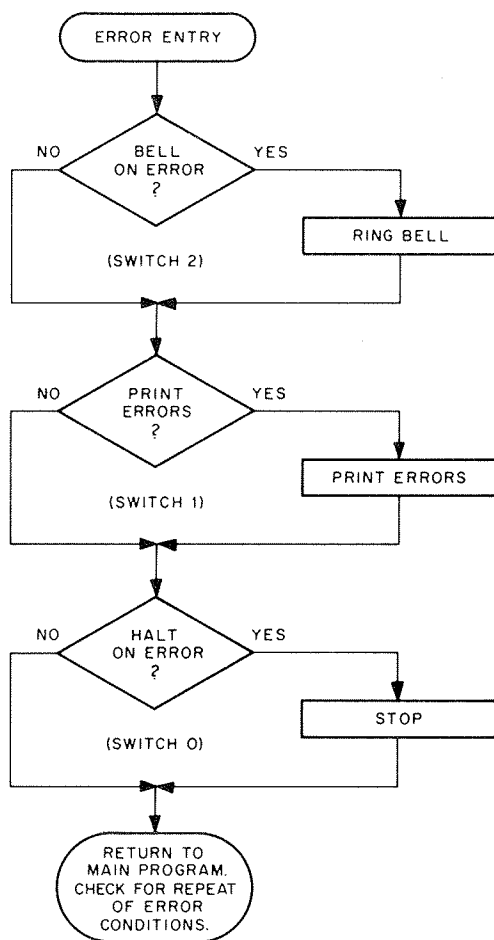
6.2.2 Recovery with Random ISZ - The random ISZ portion of this test has special recovery features. AC switch 3, as with the other tests, may be used to put the program in the 'scope mode (loop on current conditions). If, however, it is desired to save the conditions of an error and vary the parameters

which make up the current conditions, AC switches 5 to 8 may be used. If switch 5 is a 1, and an error occurs, the exact conditions which caused the error will be saved (location of ISZ instruction, location of number ISZ'd, number ISZ'd). By setting switches 6, 7, and/or 8 to a 1, any one or all of these conditions may be changed. Returning 6, 7, and/or 8 to 0 causes the original error condition for that switch to be used again. Thus it is possible to determine which condition(s) is causing the error. Switches 5-8 have no effect, if an error does not occur.

6.2.3 Test Special Conditions with Random ISZ - To test special conditions in the random ISZ test, store the location of the ISZ instruction in 17112 (SAVE1) and 17115 (RAND1), store the location of the number ISZ'd in 17113 (SAVE2) and 17116 (RAND2) and store the number to be ISZ'd in 17114 (SAVE3) and 17117 (RAND3). Restart the program at location 17067 (PROCEED) with AC switches 3 and 5 a 1.

6.2.4 Test Particular Memory Location - To test a particular memory location in the ISZ to 0 tests, store the number of the location to be tested in 00152 (PNTR2) if program is in low memory and an upper memory location is to be tested (if program is not in low memory, move it there by restarting computer at 17652 (MOVEDN)), or in 17046 (PNTR1) if program is in high memory and a lower memory location is to be tested (if program is not in high memory, move it there by restarting computer at 00144 (MOVEUP)). Whether the program is in lower or upper memory can be determined by the program counter while the program is running (it may be easier to stop the computer before looking at the PC). If the PC contains a number above 10000, the program is in upper memory and conversely. Restart computer at 00106 (ISZHI+ 6) or 17005 (ISZLOW+ 5) (as appropriate) with AC switches 3 and 4 a 1.

6.3 Error Switch Hierarchy



6.4 Error Typeout Examples

6.4.1 Increment Memory from -1 to 0

6.4.1.1 No Skip

ISZ DID NOT SKIP, LOC 001234

The above example shows that the number in location 1234 (-1) when incremented, did not skip.

6.4.1.2 Bad Add

ISZ ADD

NUMBER	AT	ORIGINAL	BAD
0 0 1 2 3 4		7 7 7 7 7 7	777777

The above example shows that the number in location 1234 (-1) was not incremented.

6.4.2 Random ISZ Test

ISZ ADD

NUMBER	AT	ORIGINAL	BAD	ISZ	AT
001234		765435	765434	005763	

The above example shows that a number in location 1234 (the number was 765435) was ISZ'd improperly producing a result of 765434 (carry from bit 17 to bit 16 was lost). The ISZ instruction was in location 5763.

7. MISCELLANEOUS

7.1 Execution Time

Not applicable

8. PROGRAM DESCRIPTION

There are three basic portions to the program, a portion which tests upper memory to be sure that all memory locations can be incremented to 0 (ISZHI), a portion which tests lower memory to be sure that all memory locations can be incremented to 0 (ISZLOW), and a portion which assures that random numbers stored in random memory locations can be ISZ'd from random memory location properly (RANISZ).

8.1 ISZHI

- a. The first function performed is that of initializing the bell counter and setting up for header printing should an error occur.
- b. Then a pointer is set to 10000 to allow the program to access the first location to be tested.
- c. The number  $777777_8$  is stored in the memory location indicated in the pointer and it is then incremented to 0. Checks are made to assure that the computer skipped and the number went to 0.



- d. A check is then made to see if the memory location should be tested again (switch 3). If so, c is repeated immediately. If not, then the address is incremented and then c is repeated.
- e. Step d is repeated until location  $17777_8$  has been tested, at which time a check is made to determine if the sequence should be repeated (switch 4). If so, the program goes back to b. If not, the program is moved to upper memory and then control is transferred to the ISZLOW portion of the program.

## 8.2 ISZLOW

This portion is essentially the same as ISZHI except that locations  $0_8$  to  $7777_8$  are tested and the program is not moved when ISZLOW is completed.

## 9.3 RANISZ

- a. First the initialization of the loop counter and the header is set up.
- b. Then three random numbers are generated for the location of the ISZ, location of the number, and the number respectively.
- c. The number is stored in the appropriate memory location and an ISZ instruction is formed for that location and stored in its proper place.
- d. The ISZ instruction is then executed and the result is checked with a synthesized ISZ for proper addition. If no errors occurred, a check is made to see if the test should be repeated (switch 3). If not then the program returns to step b. If it should be repeated, the program returns to step c.
- e. If an error occurred, the conditions which caused it are indicated to the operator (unless they have been suppressed by AC switch 1) and then a check is made to see if the conditions should be saved (switch 5). If not, the program proceeds on as if no error occurred.
- f. If the error conditions should be saved, the program then determines which of the three variables should be changed, and repeats the test for the new variables and/or the old ones. The operator can always return to original error conditions by setting AC switches 6, 7, and 8 to 0. By setting AC switch 5 to 0, the operator can continue on with testing independent of the error, if he so desires.

```

.TITLE ISZ-9
/ISZ TEST
.FULL
.LOC 100
/ROUTINE TO ISZ UPPER MEMORY FROM -1 TO 0 (OCCUPY LOW MEMORY)
/
ISZHI   LAW
        DAC CNTR2           /SET LOOP COUNTER
        LAC CONST3        /SET UP TO PRINT HEADER
        DAC ERROR4+11
        LAC STRT
        DAC PNTR2         /SET POINTER TO 10000
        CLA:CLL:CMA       /SET AC TO -1
        DAC* PNTR2        /STORE IN MEMORY
        ISZ* PNTR2        /INDEX LOCATION TO 0
        JMS ERROR2        /ERROR, COMPUTER DID NOT SKIP
        LAC* PNTR2        /GET C(MEMORY)
        SZA               /IS IT 0?
        JMS ERROR4        /NO, ERROR
        LAS
        AND MASK12
        SZA               /LOOP ON CURRENT NUMBER?
        JMP ISZHI+6       /YES
        LAS
        RTL
        SPA               /RING BELL?
        JMP .+6           /NO
        LAC CNTR2
        TAD ONE2
        DAC CNTR2
        SNA
        JMS RELL2
        LAC PNTR2         /NO, GET C (PNTR2)
        SAD UPLIM2        /IS IT 17777
        JMP .+4           /YES
        TAD ONE2         /NO, INCREMENT FOR NEXT
        DAC PNTR2        /LOCATION
        JMP ISZHI+6
        LAS
        AND MASK22
        SZA:CLA           /LOOP ON THIS TEST?
        JMP ISZHI+4       /YES
MOVEUP  LAW -A           /NO, GET READY TO
        JMS MOVE2        /MOVE PROGRAM
        ISZHI           /ORIGIN ADDRESS -LOWEST ADDRESS -LOW
        B               /DESTINATION ADDRESS -LOWEST ADDRESS -HIGH
        JMP ISZLOW       /GO TO PROGRAM IN UPPER MEMORY
CNTR2   0
PNTR2   0
ONE2    1
MASK12  40000
MASK22  20000
STRT    10000
00100   760000
00101   040151
00102   200407
00103   040355
00104   200156
00105   040152
00106   754001
00107   060152
00110   460152
00111   100321
00112   220152
00113   740200
00114   100344
00115   750004
00116   500154
00117   740200
00120   600106
00121   750004
00122   742010
00123   741100
00124   600132
00125   200151
00126   340153
00127   040151
00130   741200
00131   100216
00132   200152
00133   540157
00134   600140
00135   340153
00136   040152
00137   600106
00140   750004
00141   500155
00142   750200
00143   600104
00144   776527
00145   100160
00146   000100
00147   016440
00150   617000
00151   000000
00152   000000
00153   000001
00154   040000
00155   020000
00156   010000

```

ISZ-9 PAGE 2

00157 017777

UPLIM2 17777

↑↑↑↑

```

                                /SUBROUTINE TO MOVE THE PROGRAM (OCCUPY LOW MEMORY)
00160      000000      MOVE2      0
00161      040213      DAC TALLY2
00162      220160      LAC* MOVE2
00163      040214      DAC POINT3
00164      200160      LAC MOVE2
00165      340153      TAD ONE2
00166      040160      DAC MOVE2
00167      220160      LAC* MOVE2
00170      040215      DAC POINT4
00171      220214      LOOP2      LAC* POINT3
00172      060215      DAC* POINT4
00173      200214      LAC POINT3
00174      340153      TAD ONE2
00175      040214      DAC POINT3
00176      200215      LAC POINT4
00177      340153      TAD ONE2
00200      040215      DAC POINT4
00201      200213      LAC TALLY2
00202      340153      TAD ONE2
00203      040213      DAC TALLY2
00204      740200      SZA
00205      600171      JMP LOOP2
00206      200160      LAC MOVE2
00207      340153      TAD ONE2
00210      040160      DAC MOVE2
00211      754000      CLA!CLL
00212      620160      JMP* MOVE2
00213      000000      TALLY2      0
00214      000000      POINT3      0
00215      000000      POINT4      0
/
/ROUTINE TO RING BELL -LOW MEMORY
00216      000000      BELL2      0
00217      760207      LAW 207
00220      100431      JMS TYPE2
00221      400100      XCT ISZHI
00222      040151      DAC CNTR2
00223      620216      JMP* BELL2
/
/ROUTINE TO ISSUE CR-LF LOW MEMORY
00224      000000      CRLF2      0
00225      760212      LAW 212
00226      100431      JMS TYPE2
00227      760215      LAW 215
00230      100431      JMS TYPE2
00231      620224      JMP* CRLF2

```

↑↑↑↑

00232 000000  
 00233 040255  
 00234 777772  
 00235 040213  
 00236 200255  
 00237 740010  
 00240 740010  
 00241 742010  
 00242 040255  
 00243 500256  
 00244 240257  
 00245 100431  
 00246 200213  
 00247 340153  
 00250 040213  
 00251 741200  
 00252 620232  
 00253 200255  
 00254 600240  
 00255 000000  
 00256 000007  
 00257 000260

/TYPE OUT THE CONTENTS OF THE AC IN OCTAL (LOW)  
 TYPOL0 0

DAC TEMP2  
 LAW -6  
 DAC TALLY2  
 LAC TEMP2  
 RAL  
 RAL  
 RTL  
 DAC TEMP2  
 AND SEVEN2  
 XOR ASKII2  
 JMS TYPE2  
 LAC TALLY2  
 TAD ONE2  
 DAC TALLY2  
 SNA  
 JMP\* TYPOL0  
 LAC TEMP2  
 JMP .-14

TEMP2 0  
 SEVEN2 7  
 ASKII2 260

/

/ERROR MESSAGES (LOW MEMORY)

/

00260 000261  
 00261 311323  
 00262 332240  
 00263 304311  
 00264 304240  
 00265 316317  
 00266 324240  
 00267 323313  
 00270 311320  
 00271 254240  
 00272 314317  
 00273 303240  
 00274 377000

MESS12 .+1  
 311323 /I,S  
 332240 /Z,SP  
 304311 /D,I  
 304240 /D,SP  
 316317 /N,O  
 324240 /T,SP  
 323313 /S,K  
 311320 /I,P  
 254240 /,,SP  
 314317 /L,O  
 303240 /C,SP  
 377000 /RO

ISZ-9 PAGE 5

↑↑↑↑

00275	000276	MESS22	+.1	
00276	215212		215212	/CR,LF
00277	311323		311323	/I,S
00300	332240		332240	/Z,SP
00301	301304		301304	/A,D
00302	304215		304215	/D,CR
00303	212212		212212	/LF,LF
00304	316325		316325	/N,U
00305	315302		315302	/M,B
00306	305322		305322	/E,R
00307	240301		240301	/SP,A
00310	324240		324240	/T,SP
00311	317322		317322	/O,R
00312	311307		311307	/I,G
00313	311316		311316	/I,N
00314	301314		301314	/A,L
00315	240240		240240	/SP,SP
00316	302301		302301	/B,A
00317	304240		304240	/D,SP
00320	377000		377000	/RO,

/  
 /ERROR REPORTING SUBROUTINE 2 (LOW MEMORY)  
 ERROR2 0

00321	000000	LAS	
00322	750004	RTL	
00323	742010	SPA	/BELL ON ERROR?
00324	741100	JMS BELL2	
00325	100216	LAS	
00326	750004	RAL	
00327	740010	SPA	/PRINT ERRORS?
00330	741100	JMP .+7	/NO
00331	600340	JMS CRLF2	/YES, CRLF
00332	100224	LAC MESS12	
00333	200260	JMS TMESS2	/TYPE OUT "ISZ DID NOT SKIP, LOC"
00334	100410	LAC PNTR2	
00335	200152	JMS TYPOLO	/TYPE OUT NUMBER OF LOCATION
00336	100232	JMS CRLF2	/CR-LF
00337	100224	LAS	
00340	750004	SPA	/HALT ON ERROR?
00341	741100	XX	/YES
00342	740040	JMP* ERROR2	/EXIT
00343	620321		

↑↑↑↑

```

00344      000000      /ERROR MESSAGE REPORTING SUBROUTINE 4 (LOW MEMORY)
00345      750004      ERROR4      0
00346      742010      LAS
00347      741100      RTL
00350      100216      SPA      /BELL ON ERROR?
00351      750004      JMS BELL2      /YES
00352      740010      LAS
00353      741100      RAL
00354      600402      SPA      /PRINT ERRORS?
00355      200275      JMP .+26      /NO
00356      100410      LAC MESS22
00357      100224      JMS TMESS2      /TYPE OUT HEADER
00360      200406      LAC CONST2
00361      040355      DAC .-4
00362      200152      LAC PNTR2
00363      100232      JMS TYPOLO      /TYPE OUT LOCATION OF NUMBER
00364      760240      LAW 240
00365      100431      JMS TYPE2      /TYPE 5 SPACES
00366      100431      JMS TYPE2
00367      100431      JMS TYPE2
00370      100431      JMS TYPE2
00371      100431      JMS TYPE2
00372      750001      CLA!CMA
00373      100232      JMS TYPOLO      /TYPE ORIGINAL NUMBER
00374      760240      LAW 240
00375      100431      JMS TYPE2
00376      100431      JMS TYPE2
00377      220152      LAC* PNTR2
00400      100232      JMS TYPOLO      /TYPE BAD RESULT
00401      100224      JMS CRLF2      /CR-LF
00402      750004      LAS
00403      741100      SPA      /HALT ON ERROR?
00404      740040      XX      /YES
00405      620344      JMP* ERROR4      /EXIT
00406      600362      CONST2      JMP ERROR4+16
00407      200275      CONST3      LAC MESS22
    
```

ISZ-9 PAGE 7

↑↑↑↑

00410	000000	/MESRAGE TYPEOUT SUBROUTINE (LOW)
00411	040214	TMESS2 0
00412	220214	DAC POINT3
00413	740020	LAC* POINT3
00414	742020	RAR
00415	742020	RTR
00416	742020	RTR
00417	742020	RTR
00420	100431	JMS TYPE2
00421	540437	SAD RUBOT2
00422	620410	JMP* TMESS2
00423	220214	LAC* POINT3
00424	100431	JMS TYPE2
00425	440214	ISZ POINT3
00426	540437	SAD RUBOT2
00427	620410	JMP* TMESS2
00430	600412	JMP TMESS2+2
00431	000000	TYPE2 0
00432	500437	AND RUBOT2
00433	700406	TLS
00434	700401	TSF
00435	600434	JMP .-1
00436	620431	JMP* TYPE2
00437	000377	RUBOT2 377



ISZ-9 PAGE 8

↑↑↑↑

17000		.LOC 17000	
		/ROUTINE TO ISZ LOWER MEMORY FROM 777777 TO 0 (OCCUPY HIGH MEMORY)	
17000	760000	ISZLOW LAW	
17001	057045	DAC CNTR1	
17002	217533	LAC CONST5	
17003	057501	DAC ERROR3+11	
17004	157046	DZM PNTR1	/ZERO POINTER
17005	754001	CLA!CLL!CMA	/SET AC TO -1
17006	077046	DAC* PNTR1	/STORE -1 IN MEMORY
17007	477046	ISZ* PNTR1	/INDEX LOCATION TO 0
17010	117445	JMS ERROR1	/GO TO ERROR SUBROUTINE
17011	237046	LAC* PNTR1	/GET CONTENTS OF MEMORY
17012	740200	SZA	/IS IT 0?
17013	117470	JMS ERROR3	/NO, ERROR
17014	750004	LAS	
17015	517050	AND MASK11	
17016	740200	SZA	/LOOP ON CURRENT NUMBER?
17017	617005	JMP ISZLOW+5	/YES
17020	750004	LAS	/GET C (ACS)
17021	742010	RTL	/MOVE 2 LEFT
17022	741100	SPA	/RING BELL?
17023	617031	JMP .+6	/NO
17024	217045	LAC CNTR1	
17025	357047	TAD ONE1	
17026	057045	DAC CNTR1	
17027	741200	SNA	/IS CNTR 0?
17030	117276	JMS BELL1	/YES, RING BELL
17031	217046	LAC PNTR1	/NO, GET C (PNTR)
17032	557044	SAD UPLIM1	/IS IT 7777
17033	617037	JMP .+4	/YES
17034	357047	TAD ONE1	
17035	057046	DAC PNTR1	/NO, INCREMENT FOR NEXT LOCATION
17036	617005	JMP ISZLOW+5	
17037	750004	LAS	
17040	517051	AND MASK21	
17041	750200	SZA!CLA	/LOOP ON THIS TEST
17042	617004	JMP ISZLOW+4	/YES
17043	617052	JMP RANISZ	/NO
17044	007777	UPLIM1 7777	
17045	000000	CNTR1 0	
17046	000000	PNTR1 0	
17047	000001	ONE1 1	
17050	040000	MASK11 40000	
17051	020000	MASK21 20000	

++++

/RANDOM ISZ TEST (OCCUPIES HIGH MEMORY)

17052	760000	RANISZ	LAW	
17053	057045		DAC CNTR1	/SET UP TO COUNT LOOPS
17054	217533		LAC CONST5	
17055	057545		DAC ERROR5+11	
17056	117312		JMS GET1	/GET LOCATION OF ISZ
17057	057112		DAC SAVE1	
17060	117323		JMS GET2	/GET LOCATION TO BE ISZ'D
17061	057113		DAC SAVE2	
17062	117334		JMS COMPAR	/COMPARE RAND1 AND RAND2
17063	617060		JMP .-3	/TO BE SURE THEY ARE DIFFERENT
17064	117224		JMS GEN3	/GET NUMBER TO BE ISZ'D
17065	057117		DAC RAND3	
17066	057114		DAC SAVE3	
17067	117123	PROCED	JMS ISZTST	/PERFORM AND CHECK THE ISZ
17070	117534		JMS ERROR5	/ERROR, RETURN TO THIS INSTRUCTION
17071	750004		LAS	/NO ERROR, RETURN HERE
17072	517050		AND MASK11	
17073	740200		SZA	/LOOP ON CURRENT NUMBERS
17074	617067		JMP PROCED	/YES
17075	750004		LAS	
17076	742010		RTL	
17077	741100		SPA	/RING BELL?
17100	617111		JMP .+11	/NO
17101	217045		LAC CNTR1	/YES
17102	357047		TAD ONE1	
17103	057045		DAC CNTR1	
17104	740200		SZA	
17105	617111		JMP .+4	
17106	117276		JMS BELL1	
17107	417052		XCT RANISZ	
17110	057045		DAC CNTR1	
17111	617056		JMP RANISZ+4	
17112	000000	SAVE1	0	
17113	000000	SAVE2	0	
17114	000000	SAVE3	0	
17115	000000	RAND1	0	
17116	000000	RAND2	0	
17117	000000	RAND3	0	
17120	761340	UPLIM3	-B	/MINUS LOWER LIMIT OF UPPER PROGRAM
17121	440000	ISZCON	ISZ	
17122	017777	CONST1	17777	

ISZ-9 PAGE 10

++++

```

17123      000000      /ISZ TEST SETUP AND EXECUTION SUBROUTINE
17124      217117      ISZTST      0
17125      077116      LAC RAND3
17126      217121      DAC* RAND2      /STORE NUMBER TO BE ISZ'D
17127      357116      LAC ISZCON      /FORM ISZ
17130      077115      TAD RAND2      /INSTRUCTION, AND
17131      437115      DAC* RAND1      /STORE IT, THEN
17132      740000      XCT* RAND1      /EXECUTE IT
17133      217117      NOP      /FILLER
17134      357047      LAC RAND3      /SYNTHESIZE THE
17135      577116      TAD ONE1      /ISZ AND CHECK
17136      741000      SAD* RAND2      /TO SEE THAT THE
17137      637123      SKP      /ANSWERS AGREE
17140      217123      JMP* ISZTST      /THEY DON'T, EXIT
17141      357047      LAC ISZTST      /INCREMENT ISZTST
17142      057123      TAD ONE1      /THE HARD WAY
17143      637123      DAC ISZTST
17143      637123      JMP* ISZTST

/
/SUBROUTINE TO MOVE THE PROGRAM (OCCUPY HIGH MEMORY)
17144      000000      MOVE1      0
17145      057177      DAC TALLY1
17146      237144      LAC* MOVE1
17147      057200      DAC POINT1
17150      217144      LAC MOVE1
17151      357047      TAD ONE1
17152      057144      DAC MOVE1
17153      237144      LAC* MOVE1
17154      057201      DAC POINT2
17155      237200      LOOP1    LAC* POINT1
17156      077201      DAC* POINT2
17157      217200      LAC POINT1
17160      357047      TAD ONE1
17161      057200      DAC POINT1
17162      217201      LAC POINT2
17163      357047      TAD ONE1
17164      057201      DAC POINT2
17165      217177      LAC TALLY1
17166      357047      TAD ONE1
17167      057177      DAC TALLY1
17170      740200      SZA
17171      617155      JMP LOOP1
17172      217144      LAC MOVE1
17173      357047      TAD ONE1
17174      057144      DAC MOVE1
17175      754000      CLA:CLL
17176      637144      JMP* MOVE1

/
17177      000000      TALLY1      0
17200      000000      POINT1      0
17201      000000      POINT2      0
    
```

↑↑↑↑

		/RANDOM NUMBER GENERATORS HIGH MEMORY	
17202	000000	GEN1	0
17203	217211		LAC R1
17204	744010		RAL:CLL
17205	741400		SZL
17206	357212		TAD R1+1
17207	057211		DAC R1
17210	637202		JMP* GEN1
17211	000037	R1	000037
17212	000003		3
17213	000000	GEN2	0
17214	217222		LAC R2
17215	744010		RAL:CLL
17216	741400		SZL
17217	357223		TAD R2+1
17220	057222		DAC R2
17221	637213		JMP* GEN2
17222	000001	R2	000001
17223	000003		3
17224	000000	GEN3	0
17225	217252		LAC R3
17226	744010		RAL:CLL
17227	741400		SZL
17230	357253		TAD R3+1
17231	057252		DAC R3
17232	057254		DAC R3+2
17233	577255		SAD* R3+3
17234	741000		SKP
17235	637224		JMP* GEN3
17236	217255		LAC R3+3
17237	357047		TAD ONE1
17240	057255		DAC R3+3
17241	237255		LAC* R3+3
17242	057252		DAC R3
17243	741200		SNA
17244	617247		JMP .+3
17245	217254		LAC R3+2
17246	637224		JMP* GEN3
17247	777256		LAW R3+4
17250	057255		DAC R3+3
17251	617245		JMP .-4

IS#-9 PAGE 12

↑↑↑↑

17252	000000	R3	000000
17253	000003		3
17254	000000		0
17255	017256		R3+4
17256	000000		000000
17257	777775		777775
17260	056427		056427
17261	000175		000175
17262	000171		000171
17263	000137		000137
17264	000065		000065
17265	000037		000037
17266	000031		000031
17267	000023		000023
17270	000021		000021
17271	000015		000015
17272	000013		000013
17273	000005		000005
17274	000001		000001
17275	000000		000000

/

/ROUTINE TO RING BELL HIGH MEMORY

/

17276	000000	BELL1	0
17277	760207		LAW 207
17300	117702		JMS TYPE1
17301	417000		XCT ISZLOW
17302	057045		DAC CNTR1
17303	637276		JMP* BELL1

/

/

/ROUTINE TO ISSUE CR-LF HIGH MEMORY

/

17304	000000	CRLF1	0
17305	760215		LAW 215
17306	117702		JMS TYPE1
17307	760212		LAW 212
17310	117702		JMS TYPE1
17311	637304		JMP* CRLF1

↑↑↑↑

```

17312      000000      /GET RANDOM NUMBER SUBROUTINES
17313      117202      GET1      0
17314      517122      JMS GEN1      /GET RANDOM
17315      057115      AND CONST1     /MASK
17316      357120      DAC RAND1      /STORE
17317      740100      TAD UPLIM3     /COMPARE TO SEE
17320      617313      SMA              /IF IT IS IN THE PROGRAM
17321      217115      JMP .-5        /IT IS, GENERATE ANOTHER
17322      637312      LAC RAND1      /NO, IT ISN'T, EXIT
17323      000000      JMP* GET1     /WITH NUMBER IN AC
17324      117213      GET2      0
17325      517122      JMS GEN2
17326      057116      AND CONST1
17327      357120      DAC RAND2
17330      740100      TAD UPLIM3
17331      617324      SMA
17332      217116      JMP .-5
17333      637323      LAC RAND2
17334      000000      JMP* GET2
17335      217115      /
17336      740001      /COMPARE RAND1 AND RAND2 SUBROUTINE
17337      357047      COMPAR      0
17340      357116      LAC RAND1
17341      741200      CMA
17342      637334      TAD ONE1
17343      217334      TAD RAND2
17344      357047      SNA
17345      057334      JMP* COMPAR
17346      637334      LAC COMPAR
                        TAD ONE1
                        DAC COMPAR
                        JMP* COMPAR

```

ISZ-9 PAGE 14

↑↑↑↑

17347	000000	/TYPE OUT THE CONTENTS OF THE AC IN OCTAL (HIGH)
17350	057372	TYP0HI 0
17351	777772	DAC TEMP1
17352	057177	LAW -6
17353	217372	DAC TALLY1
17354	744010	LAC TEMP1
17355	740010	RAL!CLL
17356	742010	RAL
17357	057372	RTL
17360	517373	DAC TEMP1
17361	257374	AND SEVEN1
17362	117702	XOR ASKII1
17363	217177	JMS TYPE1
17364	357047	LAC TALLY1
17365	057177	TAD ONE1
17366	741200	DAC TALLY1
17367	637347	SNA
17370	217372	JMP* TYP0HI
17371	617355	LAC TEMP1
17372	000000	JMP .-14
17373	000007	TEMP1 0
17374	000260	SEVEN1 7
		ASKII1 260

ISZ-9 PAGE 15

↑↑↑↑

/ERROR MESSAGES (HIGH MEMORY)

17375	017376	MESS11	.+1	
17376	311323		311323	/I,S
17377	332240		332240	/Z,SP
17400	304311		304311	/D,I
17401	304240		304240	/D,SP
17402	316317		316317	/N,O
17403	324240		324240	/T,SP
17404	323313		323313	/S,K
17405	311320		311320	/I,P
17406	254240		254240	/,,SP
17407	314317		314317	/L,O
17410	303240		303240	/C,SP
17411	377000		377000	/RO
17412	017413	MESS21	.+1	
17413	215212		215212	/CR,LF
17414	311323		311323	/I,S
17415	332240		332240	/Z,SP
17416	301304		301304	/A,D
17417	304215		304215	/D,CR
17420	212212		212212	/LF,LF
17421	316325		316325	/N,U
17422	315302		315302	/M,B
17423	305322		305322	/E,R
17424	240301		240301	/SP,A
17425	324240		324240	/T,SP
17426	317322		317322	/O,R
17427	311307		311307	/I,G
17430	311316		311316	/I,N
17431	301314		301314	/A,L
17432	240240		240240	/SP SP
17433	302301		302301	/B,A
17434	304240		304240	/D,SP
17435	377000		377000	/RO
17436	017437	MESS31	.+1	
17437	240240		240240	/SP SP
17440	311323		311323	/I,S
17441	332240		332240	/Z,SP
17442	301324		301324	/A,T
17443	215212		215212	/CR,LF
17444	377000		377000	/RO



ISZ-9 PAGE 16

↑↑↑↑

```

17445      000000      /ERROR REPORTING SUBROUTINE 1 (HIGH MEMORY)
17446      750004      ERROR1      0
17447      742010      LAS
17450      741100      RTL
17451      117276      SPA                      /BELL ON ERROR?
17452      750004      JMS BELL1                      /YES
17453      740010      LAS
17454      741100      RAL
17455      617464      SPA                      /PRINT ERRORS
17456      117304      JMP .+7                      /NO
17457      217375      JMS CRLF1                      /YES, CR-LF
17460      117657      LAC MESS11
17461      217046      JMS TMESS1                      /TYPE OUT "ISZ DID NOT SKIP, LOC"
17462      117347      LAC PNTR1
17463      617304      JMS TYPOHI                      /TYPE OUT NUMBER OF LOCATION
17464      750004      JMP CRLF1                      /CR-LF
17465      741100      LAS
17466      740040      SPA                      /HALT ON ERROR
17467      637445      JMS* ERROR1                    /YES
                                           /EXIT

```

```

/
17470      000000      /ERROR MESSAGE REPORTING SUBROUTINE 3 (HIGH MEMORY)
17471      750004      ERROR3      0
17472      742010      LAS
17473      741100      RTL
17474      117276      SPA                      /BELL ON ERROR
17475      750004      JMS BELL1                      /YES
17476      740010      LAS
17477      741100      RAL
17500      617526      SPA                      /PRINT ERRORS?
17501      217412      JMP .+26                      /NO
17502      117657      LAC MESS21
17503      117304      JMS TMESS1                      /TYPE OUT HEADER
17504      217532      JMS CRLF1
17505      057501      LAC CONST4
17506      217046      DAC .-4
17507      117347      LAC PNTR1
17510      760240      JMS TYPOHI                      /TYPE OUT LOCATION OF NUMBER
17511      117702      LAW 240
17512      117702      JMS TYPE1                      /5 SPACES
17513      117702      JMS TYPE1
17514      117702      JMS TYPE1
17515      117702      JMS TYPE1

```

ISZ-9 PAGE 17

↑↑↑↑

17516	750001	CLA!CMA	
17517	117347	JMS TYPOHI	/TYPE OUT ORIGINAL NUMBER
17520	760240	LAW 240	
17521	117702	JMS TYPE1	
17522	117702	JMS TYPE1	
17523	237046	LAC* PNTR1	
17524	117347	JMS TYPOHI	/TYPE BAD RESULT
17525	117304	JMS CRLF1	/CR-LF
17526	750004	LAS	
17527	741100	SPA	/HALT ON ERROR?
17530	740040	XX	/YES
17531	637470	JMP* ERROR3	/EXIT
17532	617506	JMP ERROR3+16	
17533	217412	LAC MESS21	
		CONST4	
		CONST5	

ISZ-9 PAGE 18

↑↑↑↑

```

17534      000000      /ERROR REPORTING SUBROUTINE 5 (HIGH MEMORY)
17535      750004      ERROR5      0
17536      742010      LAS
17537      741100      RTL
17540      117276      SPA      /BELL ON ERROR
17541      750004      JMS BELL1      /YES
17542      740010      LAS
17543      741100      RAL
17544      617577      SPA      /PRINT ERRORS?
17545      217412      JMP .+33      /NO
17546      117657      LAC MESS21
17547      217436      JMS TMESS1      /TYPE HEADER
17550      117657      LAC MESS31
17551      217651      JMS TMESS1      /TYPE "ISZ AT"
17552      057545      LAC CONST6
17553      217116      DAC .-5
17554      117347      LAC RAND2
17555      760240      JMS TYPOHI      /TYPE LOCATION OF NUMBER
17556      117702      LAW 240
17557      117702      JMS TYPE1      /5 SPACES
17560      117702      JMS TYPE1
17561      117702      JMS TYPE1
17562      117702      JMS TYPE1
17563      217117      LAC RAND3
17564      117347      JMS TYPOHI      /TYPE ORIGINAL NUMBER
17565      760240      LAW 240
17566      117702      JMS TYPE1
17567      117702      JMS TYPE1
17570      237116      LAC* RAND2
17571      117347      JMS TYPOHI      /TYPE BAD NUMBER
17572      760240      LAW 240
17573      117702      JMS TYPE1
17574      217115      LAC RAND1
17575      117347      JMS TYPOHI      /TYPE LOCATION OF ISZ
17576      117304      JMS CRLF1
17577      750004      LAS
17600      741100      SPA      /HALT ON ERROR?
17601      740040      XX      /YES
17602      750004      BACK      LAS
17603      517645      AND MASK31
17604      741200      SNA
17605      637534      JMP* ERROR5      /SAVE ERROR CONDITIONS?
                                           /NO
    
```

ISZ-9 PAGE 19

```

++++
17606      750004      ISZLOC  LAS
17607      517646      AND MASK41
17610      741200      SNA                /VARY LOCATION OF ISZ?
17611      617616      JMP .+5            /NO
17612      117312      JMS GET1          /YES, GET ANOTHER ADDRESS
17613      117334      JMS COMPAR       /IS RAND1=RAND2
17614      617612      JMP .-2          /YES, TRY AGAIN
17615      617620      JMP OPLOC        /ALL OK, GO ON
17616      217112      LAC SAVE1       /TRANSFER C(SAVE1)
17617      057115      DAC RAND1       /TO RAND1
17620      750004      OPLOC  LAS
17621      517647      AND MASK51
17622      741200      SNA                /VARY LOCATION OF OPERAND?
17623      617630      JMP .+5            /NO
17624      117323      JMS GET2          /YES, GET OP ADDRESS
17625      117334      JMS COMPAR       /IS RAND1=RAND2?
17626      617624      JMP .-2          /YES, TRY AGAIN
17627      617632      JMP OPNUM        /ALL OK, GO ON
17630      217113      LAC SAVE2       /TRANSFER C(SAVE2)
17631      057116      DAC RAND2       /TO RAND2
17632      750004      OPNUM  LAS
17633      517650      AND MASK61
17634      741200      SNA                /VARY OPERAND?
17635      617640      JMP .+3            /NO
17636      117224      JMS GEN3         /YES
17637      617641      JMP .+2
17640      217114      LAC SAVE3
17641      057117      DAC RAND3
17642      117123      TRYDIF JMS ISZTST     /PERFORM AND CHECK THE ISZ
17643      617535      JMP ERROR5+1    /ERROR RETURNS HERE
17644      617602      JMP BACK        /NO ERROR RETURNS HERE
17645      010000      MASK31 10000
17646      004000      MASK41 4000
17647      002000      MASK51 2000
17650      001000      MASK61 1000
17651      617553      CONST6 JMP ERROR5+17

```

ISZ-9 PAGE 20

↑↑↑↑

17652 776527  
 17653 117144  
 17654 016440  
 17655 000100  
 17656 600100  
  
 17657 000000  
 17660 057200  
 17661 237200  
 17662 740020  
 17663 742020  
 17664 742020  
 17665 742020  
 17666 742020  
 17667 117702  
 17670 557710  
 17671 637657  
 17672 237200  
 17673 117702  
 17674 557710  
 17675 637657  
 17676 217200  
 17677 357047  
 17700 057200  
 17701 617661  
 17702 000000  
 17703 517710  
 17704 700406  
 17705 700401  
 17706 617705  
 17707 637702  
 17710 000377  
 001251  
 016440  
 000000

```

/Routine TO MOVE PROGRAM TO LOWER MEMORY AND START ISZHI
MOVEDN  LAW -A          /GET READY TO
        JMS MOVE1      /MOVE A WORDS
        B              /FROM HIGH TO
        ISZHI          /LOW MEMORY
        JMP ISZHI      /THEN TRANSFER CONTROL TO ISZHI
/
/MESSAGE TYPEOUT SUBROUTINE (HIGH)
TMESS1  0
        DAC POINT1
        LAC* POINT1
        RAR
        RTR
        RTR
        RTR
        RTR
        JMS TYPE1
        SAD RUBOT1
        JMP* TMESS1
        LAC* POINT1
        JMS TYPE1
        SAD RUBOT1
        JMP* TMESS1
        LAC POINT1
        TAD ONE1
        DAC POINT1
        JMP TMESS1+2
TYPE1   0
        AND RUBOT1
        TLS
        TSF
        JMP .-1
        JMP* TYPE1
RUBOT1  377
A=RUBOT2-ISZHI+1+RUBOT1-ISZLOW+1
B=ISZLOW-RUBOT2+ISZHI-1
.END
    
```

ISZ-9 PAGE 21

A	001251
ASKI11	17374
ASKI12	00257
B	016440
BACK	17602
BELL1	17276
BELL2	00216
CNTR1	17045
CNTR2	00151
COMPAR	17334
CONST1	17122
CONST2	00406
CONST3	00407
CONST4	17532
CONST5	17533
CONST6	17651
CRLF1	17304
CRLF2	00224
ERROR1	17445
ERROR2	00321
ERROR3	17470
ERROR4	00344
ERROR5	17534
GEN1	17202
GEN2	17213
GEN3	17224
GET1	17312
GET2	17323
ISZCON	17121
ISZHI	00100
ISZLOC	17606
ISZLOW	17000
ISZTST	17123
LOOP1	17155
LOOP2	00171
MASK11	17050
MASK12	00154
MASK21	17051
MASK22	00155
MASK31	17645
MASK41	17646
MASK51	17647
MASK61	17650
MESS11	17375
MESS12	00260
MESS21	17412
MESS22	00275
MESS31	17436
MOVEDN	17652
MOVEUP	00144
MOVE1	17144
MOVE2	00160
UNE1	17047

ISZ-9 PAGE 22

ONE2	00153
UPLOC	17620
UPNUM	17632
PNTR1	17046
PNTR2	00152
POINT1	17200
POINT2	17201
POINT3	00214
POINT4	00215
PROCFD	17067
RAND1	17115
RAND2	17116
RAND3	17117
RANISZ	17052
RUBOT1	17710
RUBOT2	00437
R1	17211
R2	17222
R3	17252
SAVE1	17112
SAVE2	17113
SAVE3	17114
SEVEN1	17373
SEVEN2	00256
STRT	00156
TALLY1	17177
TALLY2	00213
TEMP1	17372
TEMP2	00255
TMESS1	17657
TMESS2	00410
TRYDIF	17642
TYPE1	17702
TYPE2	00431
TYP0H1	17347
TYP0L0	00232
UPLIM1	17044
UPLIM2	00157
UPLIM3	17120

ISZ-9 PAGE 23

ISZHI	00100
MOVEUP	00144
CNTR2	00151
PNTR2	00152
ONE2	00153
MASK12	00154
MASK22	00155
STRT	00156
UPLIM2	00157
MOVE2	00160
LOOP2	00171
TALLY2	00213
POINT3	00214
POINT4	00215
BELL2	00216
CRLF2	00224
TYPOL0	00232
TEMP2	00255
SEVEN2	00256
ASKII2	00257
MESS12	00260
MESS22	00275
ERROR2	00321
ERROR4	00344
CONST2	00406
CONST3	00407
TMESS2	00410
TYPE2	00431
RUBOT2	00437
A	001251
B	016440
ISZLOW	17000
UPLIM1	17044
CNTR1	17045
PNTR1	17046
ONE1	17047
MASK11	17050
MASK21	17051
RANISZ	17052
PROCFD	17067
SAVE1	17112
SAVE2	17113
SAVE3	17114
RAND1	17115
RAND2	17116
RAND3	17117
UPLIM3	17120
ISZCON	17121
CONST1	17122
ISZTST	17123
MOVE1	17144
LOOP1	17155
TALLY1	17177



ISZ-9	PAGE	24
POINT1	17200	
POINT2	17201	
GEN1	17202	
R1	17211	
GEN2	17213	
R2	17222	
GEN3	17224	
R3	17252	
BELL1	17276	
CRLF1	17304	
GET1	17312	
GET2	17323	
COMPAR	17334	
TYPOHI	17347	
TEMP1	17372	
SEVEN1	17373	
ASKI11	17374	
MESS11	17375	
MESS21	17412	
MESS31	17436	
ERROR1	17445	
ERROR3	17470	
CONST4	17532	
CONST5	17533	
ERROR5	17534	
BACK	17602	
ISZLOC	17606	
OPLOC	17620	
OPNUM	17632	
TRYDIF	17642	
MASK31	17645	
MASK41	17646	
MASK51	17647	
MASK61	17650	
CONST6	17651	
MOVEDN	17652	
TMESS1	17657	
TYPE1	17702	
RUBOT1	17710	

