





1. abs( )

THIS PROGRAM TESTS THE ADDER CIRCUITS OF THE PDP-8E. THE PROGRAM IS COMPOSED OF FOUR PARTS.

A SIMULATOR FOR THE TAD INSTRUCTION WHICH TESTS ALL COMBINATIONS OF TWO ARGUMENT ADDITIONS.

A SIMULATOR FOR ROTATE INSTRUCTIONS THAT TESTS ROTATION OF ALL POSSIBLE ARGUMENTS WITH RAL, RAR, RTL, RTR AND BSH.

A CARRY GENERATION TEST

A SERIES OF RANDOM NUMBER TESTS

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-8E EQUIPPED WITH AT LEAST 4K OF MEMORY AND A TELETYPE

2.2 STORAGE

THE PROGRAM IS STORED IN LOCATIONS 0000-5554 AND UTILIZES LOCATIONS 7775-7777 AS A TEST AREA.

2.3 PRELIMINARY PROGRAMS

MAINDEG-8E-D0AA, D0BA

3. LOADING PROCEDURE

THE STANDARD PROCEDURE FOR LOADING BINARY TAPES IS TO BE USED.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

- SR00=1 SUPPRESS HALT ON ERROR
- SR01=1 SUPPRESS ERROR TYPEOUT
- SR02=1 LOOP ON ERROR
- SR03=1 FAST TEST
- SR09=1 HALT AT END OF TEST
- SR10=1 SUPPRESS END OF TEST TYPEOUT
- SR11=1 LOOP ON PRESENT TEST

4.2 STARTING ADDRESSES

NORMAL STARTING ADDRESS-0200  
RESTORE LOADERS-7600

4.3 OPERATOR ACTION

4.3.1 SET SR=0200

4.3.2 PRESS ADDR LOAD SWITCH

4.3.3 SET S.=0000

4.3.4 PRESS CLEAR AND CONT SWITCHES

5. OPERATING PROCEDURE

5.1 FAST TEST

THE ADDITION SIMULATOR NORMALLY STARTS WITH ARG1 AND ARG2 0000. TO SPEED UP THE TEST, THE VALUE OF ARG2 MAY BE SET AT SOME OTHER VALUE INITIALLY. TO DO THIS, DEPOSIT THE DESIRED VALUE IN LOCATION 170, AND PROCEED AS IN 4., BUT WITH SR=0400 INSTEAD OF 0000 IN 4.3.3

5.2 TO RESTORE AND START BINARY LOADER, STOP PROGRAM, LOAD ADDRESS 7000 AND START COMPUTER.

5.3 OPTIONS

SEE 4.1

6. ERRORS

6.1 ERROR MESSAGES

6.1.1 SIMULATED ADDITION TEST

IF A FAILURE OCCURS DURING THE SIMULATED ADDITION TEST, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND THEN HALT:

```
ARG1          SIMULATED ADD TEST FAILED
XXXXXXXXXXXXX ARG2          SIMULATED          ARG1+ARG2          ARG2*ARG1
XXXXXXXXXXXXX XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX
```

ARG1 AND ARG2 ARE THE TWO NUMBERS THAT WERE ADDED, SIMULATED IS THE ANSWER PRODUCED BY THE ADDITION SIMULATOR (LINK AND AC)

(ARG1+ARG2 IS THE RESULT OF ADDING ARG2 TO ARG1  
(ARG1 IS IN AC INITIALLY)  
(ARG2\*ARG1 IS THE RESULT OF ADDING ARG1 TO ARG2  
(ARG2 IS IN AC INITIALLY).

NOTE: EITHER THE SIMULATION OR THE ACTUAL ADDITIONS MAY HAVE FAILED.

### 6.1.2 SIMULATED ROTATE TEST

IF A FAILURE OCCURS DURING THE SIMULATED ROTATE TEST, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND THEN HALT:

```
SIMULATED AAA TEST FAILED
ORIGINAL      SIMULATED      ACTUAL
XXXXXXXXXXXXX X XXXXXXXXXXXXX X XXXXXXXXXXXXXXX
```

ORIGINAL IS THE LINK AND ACCUMULATOR TO BE ROTATED  
SIMULATED IS THE SIMULATED RESULT OF ROTATION  
ACTUAL IS THE REAL RESULT OF ROTATION  
AAA IS THE INSTRUCTION BEING TESTED, I.E. RAL,RAR,RTL,RTR,BSW

### 6.1.3 FALSE CARRY TEST

IF A FAILURE OCCURS DURING THE FALSE CARRY TEST, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND THEN HALT:

```
DATA ERROR
AAAA X XXXXXXXXXXXXX
```

AAAA IS THE STARTING ADDRESS OF THE TEST THAT FAILED  
X XXXXXXXXXXXXX ARE THE CONTENTS OF THE LINK AND AC

NOTE: EACH FALSE CARRY TEST EXPECTS LINK#1 AND AC#0 AS A RESULT.

### 6.1.4 RANDOM ADD TEST 1

IF A FAILURE OCCURS DURING RANDOM ADD TEST 1, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND THEN HALT:

```
RANDOM ADD TEST 1 FAILED
RANDA  RANDC  RESULT
XXXXXXXXXXXXX XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX
```

RANDA IS A RANDOM NUMBER  
RANDC IS THE COMPLEMENT OF RANDA  
RESULT IS THE RESULT OF CONSECUTIVE ADDITIONS OF RANDA AND RANDC

NOTE: THE EXPECTED RESULT IS LINK#1, AC#0

### 6.1.5 RANDOM ADD TEST 2

IF A FAILURE OCCURS DURING RANDOM ADD TEST 2, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND HALT:

```
RANDOM ADD TEST 2 FAILED
ARG1    ARG2    EXPECTED      ARG1+ARG2
XXXXXXXXXXXXX XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX
```

6.1.6 RANDOM ROTATE TESTS

IF A FAILURE OCCURS DURING ONE OF THE RANDOM ROTATE TESTS, THE PROGRAM WILL TYPE THE FOLLOWING MESSAGE AND THEN HALT

RANDOM AAA TEST FAILED  
ORIGINAL ACTUAL  
X XXXXXXXXXXXX X XXXXXXXXXXXX  
AAAAAR, RAL, RTR OR RTL

6.2 ERROR HALTS

THE FOLLOWING TABLE LISTS ERROR HALT LOCATIONS AND THE TEST THAT THEY APPLY TO

LOCATION	TEST
502	SIMAD
1066	SIMROT (WITH LOCATION OF SPECIFIC TEST IN AC)
3035	FCT (WITH LOCATION OF SPECIFIC TEST IN AC)
3510	RNAD1
4041	RNAD2
4661	RANDOM ROTATE (WITH LOCATION OF SPECIFIC TEST IN AC)

6.2 ERROR RECOVERY

DEPRESS CONT TO RESUME TEST

6.3 LOOPING ON ERROR

6.3.1 SWITCH REGISTER CONTROL

SET SR00#1 TO SUPPRESS ERROR HALT  
SET SR01#1 TO SUPPRESS ERROR TYPEOUT  
SET SR02#2 TO LOOP  
DEPRESS CONT

6.3.2 PROGRAM MODIFICATION

THERE ARE NOPS IN EACH TEST PROVIDED TO ALLOW THE OPERATOR TO SET UP LOOPS TIGHTER THAN THOSE AVAILABLE IN 6.3.1.

7. RESTRICTIONS

NONE

8. EXECUTION TIME

APPROXIMATELY 38 MINUTES. IF SR0381, AND (MAXXX#7777(SEE ...))  
ONE PASS TAKES APPROXIMATELY 40 SECONDS.

AS EACH TEST OR GROUP OF TESTS IS COMPLETED, THE NAME OF THAT  
TEST WILL BE TYPED. THE SEQUENCE IS:

SIMAD  
SIMROT  
FCT  
RANDOM

9. PROGRAM DESCRIPTION

9.1 SIMULATED ADDITION TEST

THE SIMULATED ADDITION TESTS SIMULATES THE ADDITION OF TWO  
ARGUMENTS; ARG1 AND ARG2. ACTUAL ADDITIONS ARE PERFORMED, AND  
THEN THE ACTUAL RESULTS ARE COMPARED TO THE SIMULATED  
ANSWER.

THE SIMULATOR OPERATES IN THE FOLLOWING MANNER:  
THE ARGUMENTS ARE "AND'ED" TOGETHER, AND ANY BITS IN THE  
RESULT THAT ARE 1'S WILL BE CARRY BITS. THE ARGUMENTS ARE  
"OR'ED" TOGETHER AND THE RESULT IS STORED. THE PREVIOUSLY  
GENERATED CARRIES ARE ROTATED ONCE TO THE LEFT AND THEN  
"AND'ED" WITH THE "OR" OF THE TWO ARGUMENTS. ANY BITS THAT ARE  
1'S ARE ALSO CARRIES AND THESE ARE COMBINED WITH THE PREVIOUS  
CARRIES. THE PROCEDURE CONTINUES UNTIL NO NEW CARRIES ARE  
GENERATED. THE FINAL CARRY RESULT IS EXCLUSIVE "OR" WITH THE  
"OR" OF THE ARGUMENTS TO GET THE SIMULATED SUM.

9.2 SIMULATED ROTATE TESTS

EACH OF THE ROTATE INSTRUCTIONS, RAR, RAL, RTR, RTL AND BSW  
IS SIMULATED FOR ALL POSSIBLE COMBINATIONS OF AC AND LINK,  
AND THE RESULTS ARE COMPARED TO THE RESULTS OF THE ACTUAL  
ROTATE.

9.3 FALSE CARRY TEST

VARIOUS COMBINATIONS OF INSTRUCTIONS AND DATA ARE USED TO  
DETECT EITHER FALSE CARRIES, OR MISSING CARRIES.

1.4 RANDOM ADD TEST 1

A RANDOM NUMBER AND ITS COMPLEMENT ARE ADDED SUCCESSIVELY AND THE EXPECTED RESULT IS ALWAYS LINK=1, AC=0.

1.5 RANDOM ADD TEST 2

A RANDOM NUMBER, AND ITS MODIFIED COMPLIMENT ARE ADDED TO PRODUCE 1 KNOW BIT IN THE AC, WITH THE LINK=1.

1.6 RANDOM ROTATE TEST

A RANDOM NUMBER IS SUCCESSIVELY ROTATED AND THE EXPECTED RESULT IS THE ORIGINAL NUMBER.





0010 \*10 /INDEX REGISTERS

0010 TSTIND, 0  
0011 POINT1, 0  
0012 POINT2, 0

0020 \*20 CNTR1, 0  
0021 ADA1, ADA2  
0022 7777

/SIMULATION VARIABLES

0023 ARG1, 0  
0024 ARG2, 0  
0025 SIMAG, 0  
0026 SIMLNK, 0  
0027 A1ORA2, 0  
0030 CARRY, 0  
0031 SUM1, 0  
0032 LINK1, 0  
0033 SUM2, 0  
0034 LINK2, 0

/MESSAGE OUTPUT VARIABLES

0035 AHFLG, 0  
0036 CHAR, 0  
0037 WD1, 0  
0040 WD2, 0

/RANDOM VARIABLES

0041 RANDB, 37  
0042 RANDB, 0  
0043 RANDC, 0  
0044 LINKR, 0  
0045 LINKRC, 0

/INDIRECT POINTERS

0046 XPRINT, PRINT /CHARACTER STRING TYPE  
0047 XTYPE, TYPE /CHARACTER TYPE

0050	1133	XRHD,	RHD	/TYPE ROTATE ERROR HEADER
0051	1200	XSROT,	SROTAL	/COMMON ROTATE SIMULATOR
0052	0756	XRALTA,	RALTAB-1	/RAL MASK TABLE
0053	1157	XRTLTA,	RRLTAB-1	/RTL MASK TABLE
0054	1140	XRTRTA,	RRTTAB-1	/RTR MASK TABLE
0055	1657	XBSWTA,	BSWTAB-1	/BYTE SWAP MASK TABLE
0056	1000	XCOMRO,	COMROT	/ROTATE COMPARISON FOR SIMULATION
0057	1031	XNXTRO,	NXTROT	/ROTATE SETUP FOR SIMULATION
0060	0504	XLNKOU,	LNKOUT	/TYPE LINK
0061	0523	XWDOUT,	WDOUT	/TYPE DATA WORD
0062	3000	XAMEAS,	SAMEAS	/COMPARE DATA
0063	3730	XAMEA,	SAMEA	
0064	3017	XAVREG,	SAVREG	/SAVE AC AND LINK
0065	3037	XDATER,	DATER	/DATA ERROR HANDLER FOR FCT
0066	3027	XHALT2,	HALT2	/DATA ERROR HALT FOR FCT
0067	3046	XLOOP,	LOOP	/LOOP ON TEST
0070	7775	XSTAB,	YSTAB	
0071	7776	XSTAI,	TSTAI	
0072	7777	XSTA2,	TSTA2	
0073	3512	XRAND,	RANDOM	/RANDOM NUMBER GENERATOR
0074	0410	XLOOP2,	HLTA+4	
0075	0552	XLOOP1,	LOOP1	

/WIDELY USED CONSTANTS

0076	0240	K240,	240	
0077	0260	K260,	260	
0100	0261	K261,	261	
0101	6000	K6000,	6000	

0102	0102	XRARTA,		
0103	4000	K4000,	4000	
0104	2000	K2000,	2000	
0105	1000	K1000,	1000	
0106	0400	K0400,	400	
0107	0200	K0200,	200	
0110	0100	K0100,	100	
0111	0040	K0040,	40	
0112	0020	K0020,	20	
0113	0010	K0010,	10	
0114	0004	K0004,	4	
0115	0002	K0002,	2	
0116	0001	K0001,	1	
0117	0000		0	
0120	4000		4000	
0121	0001		1	

/TEST POINTERS FOR FCT

0122	2004	SE01,	FCT1	
0123	2043	SE02,	FCT2	
0124	2076	SE03,	FCT3	
0125	2200	SE04,	FCT4	



/SIMULATE ADDITION BY SIMULATED GENERATEION OF SUM  
/AND CARRY BITS

/FORM OR OF ARG1 WITH ARG2

0203	7340	CLA CLL CMA	
0204	0023	AND ARG1	/LOAD AC WITH ARG1
0205	7421	MQL	/PLACE IN MQ
0206	7040	CMA	
0207	0024	AND ARG2	/LOAD AC WITH ARG2
0210	7501	MQA	/FORM ARG1 OR ARG2
0211	3027	DCA A10RA2	/SAVE ARG1 OR ARG2

/FORM XOR(EXCLUSIVE OR) OF ARG1 WITH ARG2  
/BY A XOR B=(A AND NOTB)OR(NOTA AND B)

7501	MQA	/GET ARG1 FROM MQ
7040	CMA	/FORM NOTARG1
0024	AND ARG2	/AND WITH ARG2 TO GET ARG2 AND NOTARG1
7421	MQL	/SAVE IN MQ
7040	CMA	
0024	AND ARG2	/LOAD AC WITH ARG2
7040	CMA	/FORM NOTARG2
0023	AND ARG1	/AND WITH ARG1 TO GET ARG1 AND NOTARG2
7501	MQA	/OR WITH ARG2 AND NOTARG1
3025	DCA	SIMAC
3026	DCA	SIMLNK

/AND ARG1 WITH ARG2  
/TEST FOR CARRIES  
/IF THERE ARE NO BITS IN COMMON BETWEEN ARG1 AND ARG2  
/THERE WILL BE NO CARRIES GENERATED

7040	CMA	/LOAD AC WITH ARG1
0023	AND ARG1	/AND WITH ARG2
0024	AND ARG2	/ARE THERE ANY CARRIES
7450	SNA	/NO, TERMINATE SIMULATION
5273	JMP ADD	

/GENERATE CARRIES

7421	MQL	/SAVE FIRST CARRIES
7521	MQA MQL	/GET CARRIES FROM MQ
0027	AND A10RA2	/AND WITH A10RA2 TO SEE IF MORE CARRIES ARE GENERATED
7450	SNA	/ARE THERE ANY MORE CARRIES
5243	JMP ENCAR	/NO, END SIMULATION OF CARRIES
7104	CLL RAL	/PROPAGATE CARRIES
7521	MQA MQL	/GET PREVIOUS CARRIES FROM MQ, SAVE NEW CARRIES
7501	MQA	/OR NEW CARRIES WITH PREVIOUS CARRIES

```

0242 0243 0244 0245 0246 0247 0250 0251 0252 0253 0254 0255 0256
      JMP     NXTCAR      /CONTINUE
      /TEST FOR CARRY INTO LINK
      /
      ENCAR,  MQA      /GET CARRIES
              AND     A10RA2 /AND WITH A10RA2
              AND     K4000 /TEST BIT 00
              SNA     ENCAR1 /IS BIT 00 1
              JMP     SIMLNK /NO, CARRIES DID NOT PROPAGATE INTO LINK
              DCA     XORALL /YES, SAVE CARRY INTO LINK
              JMP     RAR    /COMPLETE SIMULATION
              CLL    CML RAR /SET AC=4000
              AND     ARG1  /AND WITH ARG1
              AND     ARG2  /AND WITH ARG2 TO SEE IF ORIGINAL
              SEA     SIMLNK /NUMBERS GENERATED CARRY INTO LINK
              DCA     SIMLNK /SAVE SIMULATED LINK

```

```

      /FORM XOR OF ARG1, ARG2 AND CARRIES
      /TO GET FINAL SIMULATED SUM
      /
      XORALL, MQA      /SAVE SIMULATED CARRIES
              DCA     CARRY
              MQA     CARRY
              AND     SIMAC /FORM A10RA2 AND NOTCARRY
              AND     SIMAC /SAVE IN MQ
              AND     CARRY /FORM CARRY AND NOTA10RA2
              MQA     SIMAC /OR WITH CONTENTS OF MQ
              DCA     SIMAC /TO GET FINAL SIMULATED SUM

```

```

      /PERFORM ADDITIONS ARG1+ARG2 AND ARG2+ARG1
      /
      ADD,    CLA    CMA
              AND     ARG1  /LOAD AC WITH ARG1
              TAD     ARG2  /ADD ARG2
              NOP
              DCA     SUM1  /SAVE RESULT
              RAR     LINK1 /SAVE LINK
              DCA     ARG2  /LOAD AC WITH ARG2
              AND     ARG1  /ADD ARG1
              NOP     SUM2  /SAVE RESULT
              DCA     RAR   /SAVE LINK
              DCA     DCA   /LINK2
              NOP

```



```

PAL10      V141      11-NOV-70      21120      PAGE 1-7
/          AND      SIMLNK      /GET LINK FROM SIMULATION
0357      0026      AND      CLA      /ARE THEY THE SAME
0360      7640      CML      /NO, ERROR
0361      7020      SEL      ERROR1
0362      7430      JMP
0363      5377      /
/          /SET UP FOR NEXT ADDITION
/
/
/          JMP I      XLOOP2      /TEST FOR SIMULATION WITH SAME DATA
0364      5474      ISE      ARG1      /INCREMENT ARG1
0365      2023      JMP      SIMAD      /GO TO SIMULATION
0366      3203      ISE      ARG2      /INCREMENT ARG2
0367      2024      SKP      I      XLOOP1      /GO TO SIMULATION
0370      7410      JMP I      XLOOP1      /TEST FOR TRANSFER TO NEXT TEST
0371      5475      CLA      CHA      /TRANSFER ARG2 TO ARG1
0372      7240      AND      ARG2      /CONTINUE SIMULATION
0373      0024      DCA      ARG1      /CONTINUE SIMULATION
0374      3023      JMP      SIMAD
0375      5203
*377      0377
ERROR1,    NOP
/          /ERROR HANDLER FOR ADDITION TEST
/
/          *400
ADDERR,    LAS      /GET SWITCHES
0400      7604      AND      SR01      /TEST SR01
0401      0104      SNA      CLA      /SUPPRESS TYPEOUT IF SR01=1
0402      7650      JMS      CLA      /TYPE ERROR MESSAGE
0403      4217      LAS      ADPRT
0404      7604      AND      SR00      /HALT IF SR00=0
0405      0103      SNA      CLA      /HALT WITH ADDRESS OF TEST IN AC
0406      7650      JMS      HALTA
0407      4277      LAS      SR02      /TEST SR02
0410      7604      AND      SR02      /LOOP WITH SAME DATA IF SR02=1
0411      0105      SNA      CLA      /LOOP WITH SAME DATA
0412      7640      JMP I      XADD      /TYPE ERROR MESSAGE FOR ADDITION TEST
0413      5615      JMP I      XNXTAD
0414      5616      XADD,    ADD
0415      0273      XNXTAD, NXTADD+1
0416      0365
/          /TYPE ERROR MESSAGE FOR ADDITION TEST
/
/          0
ADPRT,    0
0417      0000      CLA      CLL      CMA
0420      7340      AND      AHFLG
0421      0035      SNA      CLA      /GET FLAG FOR ERROR MESSAGE HEADER TYPEOUT
0422      7650      JMS      AHOUT      /HAS HEADER FOR TEST BEEN TYPED
0423      4267      CMA      /NO TYPE HEADER
0424      7040      AND      ARG1
0425      0023      DCA      WD1
0426      3037      JMS      WDOUT
0427      4323      CMA      /OUTPUT ARG1
0430      7040

```





V141

PAL10

```

0505 7340 CLA CLL CMA
0506 0040 AND WD2
0507 7640 SZA CLA
0510 9320 JMP OUT1
0511 7040 CMA
0512 0077 AND K260
0513 4447 JMS I XTYPE
0514 7040 CMA
0515 0076 AND K240
0516 4447 JMS I XTYPE
0517 5704 JMP I LNKOUT
0520 7040 CMA
0521 0100 AND K261
0522 5313 JMP TYLNK
    
```

TYLNK,

OUT1,

/TYPE DATA WORD

```

0523 0000 /
0524 7340 /
0525 0102 CLA CLL CMA
0526 3011 AND XRARTA
0527 7040 DCA POINT1
0530 0411 CMA I POINT1
0531 7450 SNA
0532 5345 JMP SP1
0533 0037 AND WD1
0534 7640 SZA CLA
0535 5342 JMP OUT1A
0536 7040 CMA
0537 0077 AND K260
    
```

WDOUT,

NXBIT,

```

4447 JMS I XTYPE
5327 JMP NXBIT
7040 CMA
0100 AND K261
5340 JMP TYBIT
7040 CMA
0076 AND K240
4447 JMS I XTYPE
5723 JMP I WDOUT
0203 SIMAD
    
```

TYBIT,

OUT1A,

SP1,

ADT,

/END OF SIMULATED ADD TEST

```

7604 LAS
0115 AND SR10
7650 SNA CLA
5370 JMP SADOK
7604 LAS SR09
0114 AND SR09
7640 SZA CLA
1 7402 HLT
    
```

LOOP1,

ADHLT,

/TEST SR10  
 /IS SR10=1  
 /NO, TYPE END OF TEST MESSAGE  
 /TEST SR09  
 /IS SR  
 /YES, ( AT END OF TEST

0562 7604  
 0563 0116  
 0564 7650  
 0565 9377  
 0566 5767  
 0567 0203  
 0570 4446  
 0571 5521  
 0572 5356  
 0577 7000

LAS  
 AND SR11  
 SNA CLA  
 JMP SIMR  
 JMP I .\*1  
 SIMAD  
 JMS I XPRINT  
 OK1-1  
 JMP ADHLT  
 NOP  
 \*577  
 SIMR,

/TEST SR11  
 /IS SR11\*1  
 /NO, GO TO NEXT TEST  
 /REPEAT SIMAD

SADOK,

\*577  
SIMR,

/TEST ROTATION BY COMPARISON OF REAL AND SIMULATED  
 /ROTATES

0600 0600  
 0601 4752

\*600  
 SIMR01,  
 JMS I XR1  
 /TEST RAL

/SET UP FOR RAL TEST

0601 7340  
 0602 0052  
 0603 3012  
 0604 4451  
 0605 7340  
 0606 0024  
 0607 7640  
 0610 7020  
 0611 7040  
 0612 0023  
 0613 7004  
 0614 7000  
 0615 3031  
 0616 7430  
 0617 7040  
 0620 3033  
 0621 4456  
 0622 5205  
 0623 4457  
 0624 5201

SIMRAL,  
 CLA CLL CMA  
 AND XRALTA  
 DCA POINT2  
 JMS I XSROT  
 CLA CLL CMA  
 AND RLNK  
 SEA CLA  
 CML  
 CMA  
 AND RAC  
 RAL  
 NOP RRAC  
 DCA  
 SEL  
 CMA  
 DCA  
 JMS I XCOMRO  
 JMP RRAL  
 JMS I XNXTRO  
 JMP SIMRAL

/GET MASK TABLE FOR  
 /SIMULATED RAL  
 /SIMULATE RAL

/SET UP TO DO REAL ROTATES

/DO REAL RAL

/SAVE ROTATED ACCUMULATOR

/SAVE ROTATED LINK  
 /COMPARE ROTATES  
 /RETURN HERE FOR LOOP ON ERROR  
 /SET UP FOR NEXT ROTATE  
 /CONTINUE RAL TEST

0625 4753

SIMR02,  
 JMS I XR2  
 /TEST RAR

0626 7340  
 0627 0102  
 0630 3012  
 0631 4451

SIMRAR,  
 CLA CLL CMA  
 AND XRARTA  
 DCA POINT2  
 JMS I XSROT

/GET MASK TABLE FOR  
 /SIMULATED RAR  
 /SIMULATED RAR

```

0632 7340 RRAR, CLA CLL CMA /SET UP TP DO REAL RAR
0633 0024 AND RLNK
0634 7640 SZA CLA /DO REAL RAR
0635 7020 CML /SAVE ROTATED ACCUMULATOR
0636 7040 CMA RAC
0637 0023 AND RAR
0640 7010 RAR RRAC
0641 7000 NOP
0642 3031 DCA
0643 7430 SEL
0644 7040 CMA
0645 3033 DCA
0646 4456 JMS I RRLNK
0647 5232 JMP RRAR XCOMRO
0650 4457 JMS I XNXTRO /RETURN HERE FOR LOOP ON ERROR
0651 5226 JMP SIMRAR /SET UP FOR NEXT ROTATE
/CONTINUE RAR TEST

```

```

0652 4754 SIMR03, JMS I XR3 /TEST RTL
/
SIMRTL, CLA CLL CMA /GET MASK TABLE FOR
AND XRTLTA /SIMULATED RTL
DCA POINT2 /SIMULATE RTL
JMS I XSROT
RRTL, CLA CLL CMA /SET UP TO DO REAL ROTATE
AND RLNK
SZA CLA /DO REAL ROTATE
CML /SAVE ROTATED ACCUMULATOR
CMA RAC
AND RAR
RTL XCOMRO /SAVE ROTATED LINK
NOP RRTL XCOMRO /COMPARE ROTATES
DCA RRAR /RETURN HERE FOR LOOP ON ERROR
SEL DCA XNXTRO /SET UP TO DO NEXT ROTATE
CMA RRAC /CONTINUE RTL TEST

```

```

0677 4755 SIMR04, JMS I XR4 /TEST RTR
/
SIMRTR, CLA CLL CMA /GET MASK TABLE FOR
AND XRTLTA /SIMULATED RTR
DCA POINT2 /SIMULATE RTR
JMS I XSROT
RTR, CLA CLL CMA
0700 7340
0701 0054
0702 3012
0703 4451
0704 7340

```

```

0705 0024 AND
0706 7640 SEA CLA
0707 7020 CML
0708 7040 CMA
0709 0023 AND
0710 7012 RTR
0711 7000 NOP
0712 3031 DCA
0713 7430 SEL
0714 7040 CMA
0715 3033 DCA
0716 4456 JMS I
0717 5304 JMP
0718 4457 JMS I
0719 5300 JMP
0720 0024 AND
0721 7640 SEA CLA
0722 7020 CML
0723 7040 CMA
0724 0023 AND
0725 7012 RTR
0726 7000 NOP
0727 3031 DCA
0728 7430 SEL
0729 7040 CMA
0730 3033 DCA
0731 4456 JMS I
0732 5304 JMP
0733 4457 JMS I
0734 5300 JMP
0735 0024 AND
0736 7640 SEA CLA
0737 7020 CML
0738 7040 CMA
0739 0023 AND
0740 7002 BSW
0741 7000 NOP
0742 3031 DCA
0743 7430 SEL
0744 7040 CMA
0745 3033 DCA
0746 4456 JMS I
0747 5304 JMP
0748 4457 JMS I
0749 5325 JMP
0750 5777 JMP I
0751 1400 R1
0752 1410 R2
0753 1420 R3
0754 1430 R4
0755 1440 R5
0756 0001 RALTAB, 1
0757 0002 RALTAB, 2
0758 0004 RALTAB, 4
0759 0010 RALTAB, 10
0760 0020 RALTAB, 20
0761 0040 RALTAB, 40

```

```

/SET UP TO DO REAL ROTATE
/DO REAL ROTATE
/SAVE ROTATED ACCUMULATOR
/SAVE ROTATED LINK
/COMPARE ROTATES
/RETURN HERE FOR LOOP ON ERROR
/SET UP TO DO NEXT ROTATE
/CONTINUE RTR TEST

```

```

0724 4756 SIMR05, JMS I XR5
/TEST BYTE SWAP
/
8IMBSW, CLA CLL CMA
0725 7340 AND XBSWTA
0726 0955 DCA POINT2
0727 3012 JMS I XSBSW
0728 4776 CLA CLL CMA
0729 7340 AND RLNK
0730 0024 SEA CLA
0731 7640 CML
0732 7020 CMA
0733 7040 AND
0734 0023 BSW
0735 7002 NOP
0736 0023 DCA
0737 7002 SEL
0738 3031 CMA
0739 7040 DCA
0740 3033 JMS I
0741 4456 JMP
0742 4457 JMS I
0743 5325 JMP
0744 5777 JMP I
0745 1400 R1
0746 1410 R2
0747 1420 R3
0748 1430 R4
0749 1440 R5
0750 0001 RALTAB, 1
0751 0002 RALTAB, 2
0752 0004 RALTAB, 4
0753 0010 RALTAB, 10
0754 0020 RALTAB, 20
0755 0040 RALTAB, 40

```

```

/GET MASK TABLE FOR
/SIMULATED BSW
/SIMULATE BSW
/SET UP FOR REAL BSW
/DO REAL BSW
/SAVE ROTATED ACCUMULATOR
/SAVE ROTATED LINK
/COMPARE ROTATES
/RETURN HERE FOR LOOP ON ERROR
/SET UP FOR NEXT ROTATE
/CONTINUE BSW TEST
/END OF ROTATE SIMULATION TESTS

```

```

0724 4756 SIMR05, JMS I XR5
/TEST BYTE SWAP
/
8IMBSW, CLA CLL CMA
0725 7340 AND XBSWTA
0726 0955 DCA POINT2
0727 3012 JMS I XSBSW
0728 4776 CLA CLL CMA
0729 7340 AND RLNK
0730 0024 SEA CLA
0731 7640 CML
0732 7020 CMA
0733 7040 AND
0734 0023 BSW
0735 7002 NOP
0736 0023 DCA
0737 7002 SEL
0738 3031 CMA
0739 7040 DCA
0740 3033 JMS I
0741 4456 JMP
0742 4457 JMS I
0743 5325 JMP
0744 5777 JMP I
0745 1400 R1
0746 1410 R2
0747 1420 R3
0748 1430 R4
0749 1440 R5
0750 0001 RALTAB, 1
0751 0002 RALTAB, 2
0752 0004 RALTAB, 4
0753 0010 RALTAB, 10
0754 0020 RALTAB, 20
0755 0040 RALTAB, 40

```

```

/SAVE ROTATED LINK
/COMPARE ROTATES
/RETURN HERE FOR LOOP ON ERROR
/SET UP FOR NEXT ROTATE
/CONTINUE BSW TEST
/END OF ROTATE SIMULATION TESTS

```

```

0765 0100
0766 0200
0767 0400
0770 1000
0771 2000
0772 4000
0773 0000
0774 0001
0775 4000
0776 1236
0777 1323

XSBSW,
XROTDN, ROTDNE
    
```

```

/ (TAPE 2)
/ COMPARE RESULTS OF REAL AND SIMULATED ROTATES
/
1000 *1000
COMROT, 0
/ COMPARE ROTATED ACCUMULATORS
/
CLA CLL CMA /GET SIMULATED ROTATED ACCUMULATOR
AND SIMAC /COMPLEMENT
CMA /AND WITH REAL ROTATED ACCUMULATOR
AND RRAC /IS NOTSIMAC AND RRAC=0
SEA 7440 /NO, ERROR
JMP ERROT /GET REAL ROTATED ACCUMULATOR
CMA RRAC /COMPLEMENT
AND CMA /AND WITH SIMULATED ROTATED ACCUMULATOR
CMA SIMAC /IS SIMAC AND NOTRRAC=0
AND ERROT /NO, ERROR
JMP
    
```

```

/ COMPARE ROTATED LINKS
/
CLA CLL CMA /GET SIMULATED LINK
AND SIMLNK /GET REAL ROTATED LINK
SEA CLA /ARE THEY THE SAME
CML /NO, ERROR
CMA /RETURN HERE IF NO LOOP ON ERROR
AND RRLNK
SEA
CML
SZL
JMP ERROR2
ISZ COMROT
JMP I COMROT
    
```

/SET UP TO DO NEXT ROTATE

```

1 31 0000 /
1 32 7340 /
1 33 0024 /
1 34 7640 /
1 35 5244 /
1 36 7040 /
1 37 3024 /
1 40 2023 /
1 41 9631 /
1 42 2231 /
1 43 5631 /
1 44 3024 /
1 45 5631 /

NXTROT, 0
AND CLL CMA
AND RLNK
SZA CLA
JMP NEWLNK
CMA
DCA
ISE
JMP I
ISE
JMP I
DCA
JMP I

NEWLNK,
RLNK
RAC
NXTROT
NXTROT
NXTROT
RLNK
NXTROT

/GET LINK OF WORD TO BE ROTATED
/IS IT 0
/NO, CLEAR IT
/YES, SET IT

/INCREMENT NUMBER TO BE ROTATED
/CONTINUE SIMULATION OF PRESENT ROTATE INSTRUCTION
/PRESENT SIMULATION DONE
/GO TO NEXT TEST
    
```

```

/
/ERROR HANDLER FOR ROTATE TEST
/
LAS SR01 /TEST SR01
SNA CLA ROTPRT /IS SR01=1
JMS /NO, TYPE ERROR MESSAGE
LAS SR00 /TEST SR00
SNA CLA HALTB /IS SR00=1
JMP /NO, HALT WITH ADDRESS OF TEST IN AC
LAS SR02 /TEST SR02
AND SNA CLA ERROT+1 /IS SR02=1
JMP ERROT+2 /NO, GO TO NEW DATA
CLA CLL CMA /YES, LOOP WITH SAME DATA
AND I XSROT
TAD M4
HLT
JMP HALTB+4
-4
M4.
    
```

```

ERROR2, 7604
1046 7604
1047 0104
1050 7650
1051 4271
1052 7604
1053 0103
1054 7650
1055 5263
1056 7604
1057 0105
1060 7650
1061 5227
1062 5230
1063 7340
1064 0451
1065 1270
1066 7402
1067 5256
1070 7774

HALTB,
CLA CLL CMA
AND I XSROT
TAD M4
HLT
JMP HALTB+4
-4
M4.
    
```

```

/ERROR TYPEOUT FOR SIMULATED ROTATE TEST ERRORS
/
ROTPRT, 0
1071 0000
1072 7340
1073 0035
1074 7650
1075 4331
1076 7040
1077 0023
1100 3037
1101 7040

CLA CLL CMA
AND RHFLG
SNA CLA RHOUT
JMS CMA
AND DCA
DCA WDI
CMA

/GET ROTATE TEST HEADER FLAG
/HAS HEADER BEEN TYPED
/NO, TYPE HEADER
    
```

```

1102 0024 AND RLNK
1103 3040 DCA WD2
1104 4460 JMS I XLNKOU /OUTPUT ORIGINAL LINK
1105 4461 JMS I XWDOUT /OUTPUT ORIGINAL WORD
1106 7040 CMA
1107 0025 AND SIMAG
1110 3037 DCA WD1
1111 7040 CMA
1112 0026 AND SIMLNK
1113 3040 DCA WD2
1114 4460 JMS I XLNKOU /OUTPUT SIMULATED ROTATED LINK
1115 4461 JMS I XWDOUT /OUTPUT SIMULATED ROTATED WORD
1116 7040 CMA
1117 0031 AND RRAC
1120 3037 DCA WD1
1121 7040 CMA
1122 0033 AND RRLNK
1123 3040 DCA WD2
1124 4460 JMS I XLNKOU /OUTPUT ACTUAL ROTATED LINK
1125 4461 JMS I XWDOUT /OUTPUT ACTUAL ROTATED WORD
1126 4446 JMS I XPRINT
1127 5542 CRLF-1
1130 5671 JMP I ROTPRT

```

/OUTPUT HEADER FOR ROTATE ERROR MESSAGE

```

RHOUT, 0 JMS I XPRINT /TYPE SIMULATED XXX TEST FAILED
RHD, 0 JMS I XPRINT /WHERE XXX IS THE INSTRUCTION THAT FAILED
DH2-1
CLA CMA
DCA RHFLG
JMP I RHOUT

```

```

RRTYAB, 2000
1141 2000
1142 0400
1143 0100
1144 0020
1145 0004
1146 0001
1147 4000
1150 1000
1151 0200
1152 0040
1153 0010
1154 0002
1155 0000
1156 2000
1157 0002

```

RTLTAB, 2



1161 0010  
 1162 0040  
 1163 0200  
 1164 1000  
 1165 4000  
 1166 0001  
 1167 0004  
 1170 0020  
 1171 0100  
 1172 0400  
 1173 2000  
 1174 0000  
 1175 0002  
 1176 2000

10  
 40  
 200  
 1000  
 4000  
 1  
 4  
 20  
 100  
 400  
 2000  
 0  
 2  
 2000

/ ROTATION SIMULATOR COMMON ROUTINE  
 / ROTATE FUNCTION SIMULATED DEPENDS  
 / UPON MASK TABLE SELECTED

\*1200  
 SROTAL, 0

1200 0000  
 1201 7300  
 1202 3025  
 1203 3026  
 1204 7040  
 1205 0412  
 1206 3037  
 1207 7040  
 1210 0412  
 1211 7450  
 1212 5303  
 1213 3040  
 1214 7040  
 1215 0023  
 1216 0037  
 1217 7440  
 1220 4225  
 1221 7040  
 1222 0040  
 1223 3037  
 1224 5207

CLA CLL  
 DCA SIMAC  
 DCA SIMLNK  
 CMA  
 AND I POINT2  
 DCA WD1  
 CMA  
 AND I POINT2  
 SNA  
 JMP ENDROT  
 DCA WD2  
 CMA  
 AND RAC  
 AND WD1  
 SEA  
 JMS OR1  
 CMA  
 AND WD2  
 DCA WD1  
 JMP NBIT

/ CLEAR SIMULATION ARGUMENTS  
 / GET FIRST MASK BIT FROM TABLE  
 / GET MASK BIT FROM TABLE  
 / IS IT 0  
 / YES, FINISH SIMULATION  
 / LOAD AC WITH WORD TO BE ROTATED  
 / TEST BIT TO BE ROTATED  
 / IS IS 0  
 / NO, PLACE BIT INTO NEW POSITION  
 / BIT TO BE ROTATED  
 / BECOMES NEW MASK  
 / CONTINUE SIMULATION

/ OR BITS TO FORM PARTIALLY ROTATED WORD

1225 0000  
 1226 7240  
 1227 0040  
 1230 7421  
 1231 7040  
 1232 0025  
 1233 7501

OR1,  
 0  
 CLA CMA  
 AND WD2  
 MQL  
 CMA  
 AND SIMAC  
 MQA

/ GET BIT TO BE INSERTED  
 / SAVE IN MQ  
 / GET SIMULATED ROTATED WORD  
 / OR BIT INTO POSITION

DCA SIMAC /SAVE PARTIALLY ROTATED WORD  
JMP I OR1

RAL10 1234 3025  
1235 5625

/SIMULATE BYTE SWAP

```

1236 0000
1237 7340
1240 0236
1241 3451
1242 3025
1243 3026
1244 7040
1245 0412
1246 7450
1247 5277
1250 3037
1251 7040
1252 0412
1253 3040
1254 7040
1255 0023
1256 0037
1257 7440
1260 4225
1261 7040
1262 0037
1263 7421
1264 7040
1265 0040
1266 3037
1267 7501
1270 3040
1271 7040
1272 0023
1273 0037
1274 7440
1275 4225
1276 5244
1277 7340
1300 0024
1301 3026
1302 5636

SBSW,
0
CLA CLL CMA
AND SBSW
DCA I XSROT
DCA SIMAC
DCA SIMLNK
CMA AND I POINT2
SNA ENDBSH
JMP WD1
DCA CMA AND I POINT2
DCA WD2
CMA AND RAC
AND WD1
SZA OR1
JMS CMA AND WD1
MQL CMA AND WD1
AND WD2
AND WD1
DCA MGA WD2
DCA CMA AND RAC
AND WD1
SZA OR1
JMS N1BIT
JMP CLA CLL CMA
AND RLNK
DCA SIMLNK
JMP I SBSW

N1BIT,
0
ENDBSW,
0
ENDROT,
0

```

```

/SET UP FOR ERROR RETURN
/CLEAR SIMULATION ARGUMENTS
/GET MASK FROM TABLE
/IS IT 0
/YES, FINISH SIMULATION

/GET WORD TO BE ROTATED
/TEST BIT TO BE ROTATED
/IS IT 0
/NO, PLACE BIT IN NEW POSITION

/INTERCHANGE MASK AND BIT TO BE ROTATED

/GET WORD TO BE ROTATED
/TEST BIT TO BE ROTATED
/IS IT 0
/NO, PLACE BIT IN NEW POSITION
/CONTINUE SIMULATION

```

```

/END OF ROTATE, SHIFT LINK
/GET BIT TO BE ROTATED FROM LINK

```

```

1306 7040 CMA
1307 0116 AND
1310 0024 AND
1311 7440 SZA
1312 4225 JMS
1313 7040 CMA
1314 0412 AND I
1315 0023 AND
1316 7440 SZA
1317 7240 CLA CMA
1320 0116 AND
1321 3026 DCA
1322 5600 JMP I
/
1323 7604 LAS
1324 0115 AND
1325 7650 SNA CLA
1326 5342 JMP
1327 7604 LAS
1330 0114 AND
1331 7640 SZA CLA
1332 7402 HLT
1333 7604 LAS
1334 0116 AND
1335 7650 SNA CLA
1336 5740 JMP I
1337 5741 JMP I
1340 2000 FCT
1341 0600 SIMR01
1342 4446 JMS I
1343 5525 OK2-1
1344 5327 JMP
/
1345 5327 JMP
/
1346 5327 JMP
/
1347 5327 JMP
/
1348 5327 JMP
/
1349 5327 JMP
/
1350 5327 JMP
/
1351 5327 JMP
/
1352 5327 JMP
/
1353 5327 JMP
/
1354 5327 JMP
/
1355 5327 JMP
/
1356 5327 JMP
/
1357 5327 JMP
/
1358 5327 JMP
/
1359 5327 JMP
/
1360 5327 JMP
/
1361 5327 JMP
/
1362 5327 JMP
/
1363 5327 JMP
/
1364 5327 JMP
/
1365 5327 JMP
/
1366 5327 JMP
/
1367 5327 JMP
/
1368 5327 JMP
/
1369 5327 JMP
/
1370 5327 JMP
/
1371 5327 JMP
/
1372 5327 JMP
/
1373 5327 JMP
/
1374 5327 JMP
/
1375 5327 JMP
/
1376 5327 JMP
/
1377 5327 JMP
/
1378 5327 JMP
/
1379 5327 JMP
/
1380 5327 JMP
/
1381 5327 JMP
/
1382 5327 JMP
/
1383 5327 JMP
/
1384 5327 JMP
/
1385 5327 JMP
/
1386 5327 JMP
/
1387 5327 JMP
/
1388 5327 JMP
/
1389 5327 JMP
/
1390 5327 JMP
/
1391 5327 JMP
/
1392 5327 JMP
/
1393 5327 JMP
/
1394 5327 JMP
/
1395 5327 JMP
/
1396 5327 JMP
/
1397 5327 JMP
/
1398 5327 JMP
/
1399 5327 JMP
/
1400 5327 JMP
/
1401 5327 JMP
/
1402 5327 JMP
/
1403 5327 JMP
/
1404 5327 JMP
/
1405 5327 JMP
/
1406 5327 JMP
/
1407 5327 JMP
/
1410 0000 R2,
1411 7340 CLA CLL
1412 0251 AND
1413 3450 DCA I
1414 3035 DCA
1415 3024 DCA
1416 3023 DCA
/
1417 5327 JMP
/
1418 5327 JMP
/
1419 5327 JMP
/
1420 5327 JMP
/
1421 5327 JMP
/
1422 5327 JMP
/
1423 5327 JMP
/
1424 5327 JMP
/
1425 5327 JMP
/
1426 5327 JMP
/
1427 5327 JMP
/
1428 5327 JMP
/
1429 5327 JMP
/
1430 5327 JMP
/
1431 5327 JMP
/
1432 5327 JMP
/
1433 5327 JMP
/
1434 5327 JMP
/
1435 5327 JMP
/
1436 5327 JMP
/
1437 5327 JMP
/
1438 5327 JMP
/
1439 5327 JMP
/
1440 5327 JMP
/
1441 5327 JMP
/
1442 5327 JMP
/
1443 5327 JMP
/
1444 5327 JMP
/
1445 5327 JMP
/
1446 5327 JMP
/
1447 5327 JMP
/
1448 5327 JMP
/
1449 5327 JMP
/
1450 5327 JMP
/
1451 5327 JMP
/
1452 5327 JMP
/
1453 5327 JMP
/
1454 5327 JMP
/
1455 5327 JMP
/
1456 5327 JMP
/
1457 5327 JMP
/
1458 5327 JMP
/
1459 5327 JMP
/
1460 5327 JMP
/
1461 5327 JMP
/
1462 5327 JMP
/
1463 5327 JMP
/
1464 5327 JMP
/
1465 5327 JMP
/
1466 5327 JMP
/
1467 5327 JMP
/
1468 5327 JMP
/
1469 5327 JMP
/
1470 5327 JMP
/
1471 5327 JMP
/
1472 5327 JMP
/
1473 5327 JMP
/
1474 5327 JMP
/
1475 5327 JMP
/
1476 5327 JMP
/
1477 5327 JMP
/
1478 5327 JMP
/
1479 5327 JMP
/
1480 5327 JMP
/
1481 5327 JMP
/
1482 5327 JMP
/
1483 5327 JMP
/
1484 5327 JMP
/
1485 5327 JMP
/
1486 5327 JMP
/
1487 5327 JMP
/
1488 5327 JMP
/
1489 5327 JMP
/
1490 5327 JMP
/
1491 5327 JMP
/
1492 5327 JMP
/
1493 5327 JMP
/
1494 5327 JMP
/
1495 5327 JMP
/
1496 5327 JMP
/
1497 5327 JMP
/
1498 5327 JMP
/
1499 5327 JMP
/
1500 5327 JMP
/

```

```

/GET MASK FOR LINK
/TEST LINK
/IS LINK 0
/PLACE LINK IN NEW POSITION
/GET MASK FOR BIT TO BE ROTATED INTO LINK
/TEST BIT IN WORD TO BE ROTATED INTO LINK
/IS IT 0
/NO, SET LINK=1
/TEST SR10
/IS SR10=1
/NO, TYPE "SIMROT"
/TEST SR09
/IS SR09=1
/YES, HALT AT END OF ROTATE TESTS
/TEST SR11
/IS SR11=1
/NO, GO TO NEXT TEST
/YES, REPEAT ROTATE TESTS
/SET UP FOR ROTATE TESTS
/SET UP HEADER
/FOR RAL TEST ERROR MESSAGE
/CLEAR ROTATE HEADER FLAG
/SET UP HEADER
/FOR RAR TEST ERROR MESSAGE

```

```

PAGE
R1,
1400
0000
7340 CLA CLL
1402 0250 AND
1403 3450 DCA I
1404 3035 DCA
1405 3024 DCA
1406 3023 DCA
1407 5600 JMP I
1410 0000 R2,
1411 7340 CLA CLL
1412 0251 AND
1413 3450 DCA I
1414 3035 DCA
1415 3024 DCA
1416 3023 DCA
/
1417 5600 JMP I
/
1418 5600 JMP I
/
1419 5600 JMP I
/
1420 5600 JMP I
/
1421 5600 JMP I
/
1422 5600 JMP I
/
1423 5600 JMP I
/
1424 5600 JMP I
/
1425 5600 JMP I
/
1426 5600 JMP I
/
1427 5600 JMP I
/
1428 5600 JMP I
/
1429 5600 JMP I
/
1430 5600 JMP I
/
1431 5600 JMP I
/
1432 5600 JMP I
/
1433 5600 JMP I
/
1434 5600 JMP I
/
1435 5600 JMP I
/
1436 5600 JMP I
/
1437 5600 JMP I
/
1438 5600 JMP I
/
1439 5600 JMP I
/
1440 5600 JMP I
/
1441 5600 JMP I
/
1442 5600 JMP I
/
1443 5600 JMP I
/
1444 5600 JMP I
/
1445 5600 JMP I
/
1446 5600 JMP I
/
1447 5600 JMP I
/
1448 5600 JMP I
/
1449 5600 JMP I
/
1450 5600 JMP I
/
1451 5600 JMP I
/
1452 5600 JMP I
/
1453 5600 JMP I
/
1454 5600 JMP I
/
1455 5600 JMP I
/
1456 5600 JMP I
/
1457 5600 JMP I
/
1458 5600 JMP I
/
1459 5600 JMP I
/
1460 5600 JMP I
/
1461 5600 JMP I
/
1462 5600 JMP I
/
1463 5600 JMP I
/
1464 5600 JMP I
/
1465 5600 JMP I
/
1466 5600 JMP I
/
1467 5600 JMP I
/
1468 5600 JMP I
/
1469 5600 JMP I
/
1470 5600 JMP I
/
1471 5600 JMP I
/
1472 5600 JMP I
/
1473 5600 JMP I
/
1474 5600 JMP I
/
1475 5600 JMP I
/
1476 5600 JMP I
/
1477 5600 JMP I
/
1478 5600 JMP I
/
1479 5600 JMP I
/
1480 5600 JMP I
/
1481 5600 JMP I
/
1482 5600 JMP I
/
1483 5600 JMP I
/
1484 5600 JMP I
/
1485 5600 JMP I
/
1486 5600 JMP I
/
1487 5600 JMP I
/
1488 5600 JMP I
/
1489 5600 JMP I
/
1490 5600 JMP I
/
1491 5600 JMP I
/
1492 5600 JMP I
/
1493 5600 JMP I
/
1494 5600 JMP I
/
1495 5600 JMP I
/
1496 5600 JMP I
/
1497 5600 JMP I
/
1498 5600 JMP I
/
1499 5600 JMP I
/
1500 5600 JMP I
/

```

```

1417 JMP I R2
1420 0
1421 CLA CLL CMA
1422 AND XM4
1423 DCA I XRHD
1424 DCA RHFLG
1425 DCA RLNK
1426 DCA RAC
1427 JMP I R3
1430 0
1431 CLA CLL CMA
1432 AND XM5
1433 DCA I XRHD
1434 DCA RHFLG
1435 DCA RLNK
1436 DCA RAC
1437 JMP I R4
1440 0
1441 CLA CLL CMA
1442 AND XM6
1443 DCA I XRHD
1444 DCA RHFLG
1445 DCA RLNK
1446 DCA RAC
1447 JMP I R5
1450 EM2-1
1451 EM3-1
1452 EM4-1
1453 EM5-1
1454 EM6-1

```

/SET UP HEADER  
 /FOR RTR TEST ERROR MESSAGE

/SET UP HEADER  
 /FOR RTL TEST ERROR MESSAGE

/SET UP HEADER  
 /FOR BSW TEST ERROR MESSAGE

```

/ /CHARACTER STRING TYPE ROUTINE
/ * RETURN, * =LINE FEED

```

```

1600 PAGE
1601 PRINT,
1602 0
1603 CLA
1604 TAD I PRINT
1605 DCA POINT1
1606 ISZ PRINT
1607 TAD I POINT1
1608 DCA CHAR
1609 TAD CHAR
1610 RTR
1611 RTR
1612 RTR
1613 JMS TYPSET
1614 TAD CHAR
1615 JMS TYPSET
1616 JMP PRINT+5
1617 0
1620 AND K0077
1621 SNA
1622 JMP I PRINT

```

V141 11-NOV-78

PAL18

1623	1246	TAD	M48
1624	7510	SPA	
1625	5230	JMP	*3
1626	1076	TAD	K240
1627	5243	JMP	MTP
1630	7001	IAC	
1631	7440	SEA	
1632	5235	JMP	*3
1633	1251	TAD	K215
1634	5243	JMP	MTP
1635	7001	IAC	
1636	7440	SEA	
1637	5242	JMP	*3
1640	1250	TAD	K212
1641	5243	JMP	MTP
1642	1247	TAD	K336
1643	4447	JMS I	XTYPE
1644	5617	JMP I	TYPSET
1645	0077	0077	
1646	7740	M40,	
1647	0336	K336,	
1650	0212	K212,	
1651	0215	K215,	
1652	0000	TYPE,	
1653	6046	0	
1654	6041	TLS	
1655	5254	TSP	
1656	7200	JMP	.-1
1657	5652	CLA	
		JMP I	TYPE

1660	0001	BSWTAB, 1	
1661	0100	1	
1662	0002	2	
1663	0200	200	
1664	0004	4	
1665	0400	400	
1666	0010	10	
1667	1000	1000	
1670	0020	20	
1671	2000	2000	
1672	0040	40	
1673	4000	4000	
1674	0000	0	

  

2000	2000	PAGE	
2001	7300	FCT,	
2002	1122	CLA CLL	
2002	3154	TAD	SEQ1
2003	3020	DCA	SEQ
		DCA	CNTR1
		/	/
		/	/FALSE CARRY TEST#1

```

PAL10 V141 2004 7300 /
      FCT1, / CLA CLL
      / PLACE INSTRUCTIONS AND DATA IN TEST ADDRESSES
      /
      FCS1, / CLA CLL /DATA=0000
      DCA I XSTA1 /LOC.=7776
      TAD INS1 /INSTRUCTION=TAD ,-1
      DCA I XSTA2 /LOC.=7777
      TAD INS2 /INSTRUCTION=TAD ,+3
      DCA TSTA3 /LOC.=0000
      TAD INS3 /INSTRUCTION=IAC
      DCA TSTA4 /LOC.=0001
      TAD INS4 /INSTRUCTION=JMP I ,+2
      DCA TSTA5 /LOC.=0002
      CLA CMA /DATA=7777
      DCA TSTA6 /LOC.=0003
      TAD AD1 /ADDRESS=RET1
      DCA TSTA7 /LOC.=0004
      /EXECUTE INSTRUCTIONS PREVIOUSLY ASSEMBLED IN TEST
      /ADDRESSES
  
```

```

      FCL1, / CLA CLL
      JMP I XSTA2 /PROVIDED FOR PROGRAM MODIFICATION
      NOP /SAVE LINK AND AC
      NOP /EXPECTED RESULTS ARE AC=0, LINK=1
      JMS I XAVREG /COMPUTATION ERROR HAS OCCURED
      /TEST FOR HALT
      /TEST FOR LOOP
      JMP FCL1 /ADDRESS OF NEXT TEST
      CLA SEQ2 /GO TO NEXT TEST
      TAD SEC
      DCA SEC
      JMP I SEQ
      /FALSE CARRY TEST#2
      /
      FCT2, / CLA CLL
      / PLACE INSTRUCTIONS AND DATA IN TEST ADDRESSES
      /
      FCS2, / CLA CLL CMA /DATA=7777
      DCA I XSTA1 /LOC.=7776
  
```

```

      2030 7430 /
      2031 7440 /
      2032 4465 /
      2033 7410 /
      2034 4466 /
      2035 4467 /
      2036 5223 /
      2037 7200 /
      2040 1123 /
      2041 3154 /
      2042 5554 /
  
```

```

      2043 7300 /
      2044 7340 /
      2045 3471 /
  
```

2046 1136 TAD INS1 /INSTRUCTION=TAD .-1  
 2047 3472 DCA I XSTA2 /LOC.=7777  
 2050 1137 TAD INS3 /INSTRUCTION=IAC  
 2051 3000 DCA TSTA3 /LOC.=0000  
 2052 1141 TAD INS5 /INSTRUCTION=JMP I .+1  
 2053 3001 DCA TSTA4 /LOC.=0001  
 2054 1330 TAD AD2 /ADDRESS=RET2  
 2055 3002 DCA TSTA5 /LOC.=0002

/EXECUTE INSTRUCTIONS PREVIOUSLY ASSEMBLED IN TEST  
 /ADDRESSES

2056 7300 FCL2, CLA CLL XSTA2  
 2057 5472 JMP I XSTA2  
 2060 7000 NOP  
 2061 7000 NOP  
 2062 4464 JMS I XAVREG /SAVE AC AND LINK

/EXPECTED RESULTS ARE AC=0, LINK=1

2063 7430 SZL  
 2064 7440 SZA  
 2065 4465 JMS I XDATER  
 2066 7410 SKP  
 2067 4466 JMS I XHALT2  
 2070 4467 JMS I XLOOP  
 2071 5256 JMP FCL2  
 2072 7200 CLA SEQ3  
 2073 1124 TAD SEQ  
 2074 3154 DCA SEQ  
 2075 5554 JMP I SEQ

/FALSE CARRY TEST #3

2076 7300 FCT3, CLA CLL  
 2077 1137 FCS3, TAD INS3 /INSTRUCTION=IAC  
 2100 3471 DCA I XSTA1 /LOC.=7776  
 2101 1333 TAD INS16 /INSTRUCTION=TAD I 21  
 2102 3472 DCA I XSTA2 /LOC.=7777  
 2103 1152 TAD INS14 /INSTRUCTION=JMP I .+1  
 2104 3000 DCA TSTA3 /LOC.=0000  
 2105 1331 TAD AD3 /ADDRESS=RET3  
 2106 3001 DCA TSTA4 /LOC.=0001

2107 7300 FCL3, CLA CLL XSTA1  
 2110 5471 JMP I XSTA1







2265 1127 TAD SEQ6  
2266 3154 DCA SEQ  
2267 5554 JMP I SEQ

/  
/  
/FALSE CARRY TEST #6

2270 7300 FCT6, CLA CLL  
/ CLA CLL  
2271 7300 FCS6, CLA CLL  
2272 1144 TAD INS8  
2273 3472 DCA I XSTA2  
2274 1137 TAD INS3  
2275 3001 DCA TSTA4  
2276 1151 TAD INS13  
2277 3002 DCA TSTA5  
2300 1326 TAD AD6  
2301 3003 DCA TSTA6  
/ INSTRUCTION=ISZ ,\*1  
/LOC.=7777  
/INSTRUCTION=IAC  
/LOC.=0001  
/INSTRUCTION=JMP I ,\*1  
/LOC.=0002  
/ADDRESS=RET6  
/LOC.=0003

2302 7340 FCL6, CLA CLL CMA  
2303 3000 DCA TSTA3  
2304 7240 CLA CMA  
2305 5472 JMP I XSTA2  
2306 7000 NOP  
2307 7000 NOP  
2310 4464 JMS I XAVREG  
/

2311 7430 SZL  
2312 7440 SZA  
2313 4465 JMS I XDATER  
2314 7410 SKP  
2315 4466 JMS I XHALT2  
2316 4467 JMS I XLOOP  
2317 5302 JMP FCL6  
2320 7200 CLA  
2321 1130 TAD SEQ7  
2322 3154 DCA SEQ  
2323 5554 JMP I SEQ  
2324 2215 RET4  
2325 2252 RET5  
2326 2306 RET6

2400 PAGE  
/ FALSE CARRY TEST#7  
/

2400

FCT7,

CLA CLL

2401

FCS7,

CLA CLL

2402 7300  
2403 1145  
2404 3472  
2405 1137  
2406 3001  
2407 1151  
2408 3002  
2409 1326  
2410 3003

INS9  
XSTAR  
INS3  
TSTA4  
INS13  
TSTA3  
AD7  
TSTA6  
/INSTRUCTION=ISE I TSTIND  
/LOC.=0777  
/INSTRUCTION=IAC  
/LOC.=0001  
/INSTRUCTION=JMP I .+1  
/LOC.=0002  
/ADDRESS=RET7  
/LOC.=0003

2412

FCL7,

CLA CLL

2413 7340  
2414 3010  
2415 7040  
2416 3000  
2417 7040  
2418 5472  
2419 7000  
2420 7000  
2421 7000  
2422 4464

CMA  
TSTIND  
DCA  
CMA  
TSTA3  
CMA  
XSTAR  
NOP  
JMS I  
XAVREG

2423

SEL

SEA

2424 7430  
2425 7440  
2426 4465  
2427 7410  
2428 4466  
2429 4467  
2430 5212  
2431 7200  
2432 1131  
2433 3154  
2434 5554

XDATE  
JMS I  
XDATE  
JMS I  
XHALT2  
XLOOP  
FCL7  
CLA  
SEQ8  
TAD  
DCA  
SEQ  
JMS I  
SEQ

2436

FCT8,

CLA CLL

/FALSE CARRY TEST #8

2437

FCS8,

CLA CLL

2438 7300  
2439 1137  
2440 3000  
2441 1137  
2442 3001  
2443 1140  
2444 3002

INS3  
TSTA3  
INS3  
TSTA4  
INS4  
TSTA5  
/INSTRUCTION=IAC  
/LOC.=0000  
/INSTRUCTION=IAC  
/LOC.=0001  
/INSTRUCTION=JMP I .+2  
/LOC.=0002

2446 1327 TAD ADB /ADDRESS=RET0  
2447 1004 DCA YSTA7 /LOC.=0004

2450 7320 FCL9, CLA CLL /INSTRUCTION=JMS ,01  
2451 1146 TAD I IN910 /LOC.=7777  
2452 3472 DCA I XSTA2

2453 7240 CLA CMA  
2454 5472 JMP I XSTA2  
2455 7000 NOP  
2456 7000 NOP

2457 7430 SZL  
2460 7440 SEA  
2461 4465 JMS I XDATER  
2462 7410 SKP  
2463 4466 JMS I XHALT2  
2464 4467 JMS I XLOOP  
2465 5250 JMP FCL8  
2466 7200 CLA  
2467 1132 TAD SEQ9  
2470 3154 DCA SEQ  
2471 5554 JMP I SEQ

2472 7300 FCT9, /FALSE CARRY TEST #9  
CLA CLL

2473 7340 FCS9, /DATA=7777  
2474 3471 DCA I XSTA1 /LOC.=7776  
2475 1137 TAD INS3 /INSTRUCTION=IAC  
2476 3000 DCA YSTA3 /LOC.=0000  
2477 1141 TAD INS5 /INSTRUCTION=JMP I ,01  
2500 3001 DCA TSTA4 /LOC.=0001  
2501 4330 TAD AD9 /ADDRESS=RET9  
2502 3002 DCA TSTA5 /LOC.=0002

2503 7320 FCL9, CLA CLL /INSTRUCTION=JMS I ,01  
2504 1147 TAD I IN910  
2505 3472 DCA I XSTA2  
2506 7240 CLA CMA  
2507 5472 JMP I XSTA2  
2510 7200 NOP  
-11

2512

4464

JMS I XAVREG

2513

7430

SZL

2514

7440

SEA

2515

4465

JMS I XDATE

2516

7410

SKP

2517

4466

JMS I XHALT2

2520

4467

JMS I XLOOP

2521

5303

JMP FCL9

2522

7200

CLA

2523

1133

TAD SEQ10

2524

3154

DCA SEQ

2525

5554

JMP I SEQ

2526

2420

RET7

2527

2455

RET8

2530

2510

RET9

PAGE

2600

JMS I XAVREG

/FALSE CARRY TEST #10

2600

7300

CLA CLL

2601

7300

CLA CLL

2602

1150

TAD INS12

2603

3472

DCA I XSTA2

2604

1137

TAD INS3

2605

3001

DCA TSTA4

2606

1151

TAD INS13

2607

3002

DCA TSTA5

2610

1315

TAD AD10

2611

3003

DCA TSTA6

/INSTRUCTION=JMS I TSTIND  
/LOC.=7777  
/INSTRUCTION=IAC  
/LOC.=0001  
/INSTRUCTION=JMP I .+1  
/LOC.=0002  
/ADDRESS=RET10  
/LOC.=0003

2612

7340

CLA CLL CMA

2613

3010

DCA TSTIND

2614

7040

CMA

2615

5472

JMP I XSTA2

2616

7000

NOP

2617

7000

NOP

2620

4464

JMS I XAVREG

2621

7430

SZL

2622 7440  
 2623 4455  
 2624 7410  
 2625 4456  
 2626 4467  
 2627 5212  
 2630 7200  
 2631 1134  
 2632 3154  
 2633 5554

/  
 /  
 /FALSE CARRY TEST #11  
 /  
 /

FCT11, CLA CLL

2634 7300

2635 7300  
 2636 8137  
 2637 9000  
 2640 8141  
 2641 3001  
 2642 1316  
 2643 3002

FCS11, CLA CLL

TAD INS3  
 DCA TSTAS  
 TAD INS5  
 DCA TSTA4  
 TAD AD11  
 DCA TSTAS  
 /  
 /  
 /INSTRUCTION=IAC  
 /LOC.=0000  
 /INSTRUCTION=JMP I ,01  
 /ADDRESS=0001  
 /ADDRESS=RET11  
 /LOC.=0002

2644 7300  
 2645 1153  
 2646 3472  
 2647 7240  
 2650 5472  
 2651 7000  
 2652 7000  
 2653 4464

FCL11, CLA CLL

TAD INS10  
 DCA I XSTA2  
 CLA CMA XSTA2  
 JMP I XSTA2  
 NOP  
 NOP  
 JMS I XAVREG  
 /  
 /  
 /INSTRUCTION=JMS  
 /LOC.=7777

RET11,

2654 7430  
 2655 7440  
 2656 4465  
 2657 7410  
 2660 4466  
 2661 4467  
 2662 5244  
 2663 7200  
 2664 1135  
 2665 3154  
 2666 5554

SZL  
 SZA  
 JMS I XDATAER  
 SKP  
 JMS I XHALT2  
 JMS I XLOOP  
 JMP FCL11  
 CLA  
 TAD SEQ12  
 DCA SEQ  
 JMP I SEQ

/FALSE CARRY TEST #12

CLA CLL

FCT12,

2667 7300

CLA CLL  
TAD I  
DCA I  
TAD I  
DCA I  
TAD I  
DCA I

FCS12,

2670 7300  
2671 1137  
2672 3472  
2673 1152  
2674 3000  
2675 1317  
2676 3001

/INSTRUCTION=IAC  
/LOC.=7777  
/INSTRUCTION=JMP I .+1  
/LOC.=0000  
/ADDRESS=RET12  
/LOC.=0001

CLA CLL CMA  
JMP I XSTAR  
NOP  
NOP  
JMS I XAVREG

FCL12,  
RET12,

2677 7340  
2700 5472  
2701 7000  
2702 7000  
2703 4464

SZL  
SZA I XDATE  
SKP I XDATE  
JMS I XHALT2  
JMS I XLOOP  
JMP I FCL12  
JMP I .+1  
ENDFCT  
RET10  
RET11  
RET12

AD10,  
AD11,  
AD12,

2704 7430  
2705 7440  
2706 4465  
2707 7410  
2710 4466  
2711 4467  
2712 5277  
2713 5714  
2714 3200  
2715 2616  
2716 2651  
2717 2701

PAGE

3000

/(TAPE 3)  
/COMPARE TWO NUMBERS: W1\*NOT(W2)+W2\*NOT(W1)=0, W1=W2

SAMEAS,  
CLA CLL CMA  
AND W2  
CMA  
AND W1  
SZA CLA  
JMP I SAMEAS  
CMA  
AND W1

3000  
3001 7340  
3002 0040  
3003 7040  
3004 0037  
3005 7640  
3006 5600  
3007 7040  
3010 0037

/W1\*NOT(W2)=0  
/W1\*NOT(W2)NOT 0, ERROR





3055 5646 JMP I LOOP

/TYPE DATA ERROR MESSAGE

TYP52, 0 JMS I XPRINT /TYPE "DATA ERROR"
DATE-1 W1 TAD XADOUT /TYPE TEST ADDRESS
CLA CLL CMA AND TEMPAC
DCA WD1 AND TEMPL
DCA WD2 JMS I XLNKOU /OUTPUT RECEIVED LINK
JMS I XMDOUT /OUTPUT RECEIVED AC
JMP I TYP52
XADOUT, ADOUT

/END OF PASS

PAGE ENDFCT, CLA CLL CNTR1 /INCREMENT PASS COUNT
ISE OUT /PASS NOT COMPLETE
JMP LAS SR10 /TEST SR10
AND SNA CLA FCTOK /IS SR10#1
JMP LAS SR09 /NO, TYPE FCT
AND SZA CLA SR09 /TEST SR09
HLT LAS SR11 /IS SR09#1
AND SZA CLA SR11 /YES, HALT
JMP I XPRINT /TEST SR11
RNAD1 OK3-1 /IS SR11#1
JMS I XPRINT /YES, LOOP ON FCT
OK3-1 FCTHLT /NO, GO TO NEXT TEST
JMP TAD SEQ1
DCA SEQ
JMP I SEQ

FCTOK, OUT, DCA

ADOUT, 0 DCA TEMPI
3030 3037 TAD TEMPI
3231 1037

/CONVERT ADDRESS TO ASCII AND OUTPUT

```

3232 0266 AND K0007
3233 3264 A2
3234 1037 TAD TEMP1
3235 7006 RTL
3236 7004 RAL
3237 0267 AND K0700
3240 1264 TAD A2
3241 1270 TAD K6060
3242 3264 DCA A2
3243 1037 TAD TEMP1
3244 7012 RTR
3245 7012 RTR
3246 7012 RTR
3247 0266 AND K0007
3250 3263 DCA A1
3251 1037 TAD TEMP1
3252 7012 RTR
3253 7010 RAR
3254 0267 AND K0700
3255 1263 TAD A1
3256 1270 TAD K6060
3257 3263 DCA A1
3260 4446 JMS I XPRINT
3261 3262 A1-1
3262 5627 JMP I ADOUT
3263 0000 A1,
3264 0000 A2,
3265 4000 4000
3266 0007 0007
3267 0700 0700
3270 6060 6060

```

/MULTIPLE ADDITIONS OF RANDOM NUMBER AND ITS TWO'S COMPLEMENT

```

PAGE PAGE /GENERATE RANDOM NUMBERS
RNAD1,
3400 3400 CLA CLL
3401 7300 JMS I XRAND
3402 7300 CLA CLL
3403 1041 TAD RANDB
3404 1043 TAD RANDC
3405 1043 TAD RANDE
3406 1041 TAD RANDA
3407 1041 TAD RANDB
3410 1041 TAD RANDC
3411 1043 TAD RANDE
3412 1043 TAD RANDB
3413 1041 TAD RANDC
3414 1041 TAD RANDE
3415 1043 TAD RANDB
3416 1041 TAD RANDC
3417 1043 TAD RANDE
3420 1043 TAD RANDB
3421 1041 TAD RANDC
3422 1041 TAD RANDE
3423 1043 TAD RANDB
3424 1043 TAD RANDC
3425 1043 TAD RANDE

```

```

3425 1043 TAD RANDC
3426 1041 TAD RANDB
3427 1043 TAD RANDC
3430 1041 TAD RANDB
3431 1041 TAD RANDB
3432 1041 TAD RANDB
3433 1043 TAD RANDC
3434 1043 TAD RANDC
3435 7000 NOP
3436 4464 JMS I XAVREG
3437 7430 SEL
3440 7440 SZA
3441 4646 JMS I XRN1ER
3442 4467 JMS I NERROP
3443 5202 JMP RNAD1+2
3444 5645 JMP I .+1
3445 3600 RNAD2

```

```

3446 3447 XRN1ER, RN1ER

```

```

/ /RANDOM ADD TEST 1 ERROR HANDLER
/
RN1ER,
0000 LAS
3450 7604 AND SR01
3451 0104 SZA CLA SR01
3452 7640 JMP SKHLT
3453 5302 JMS I XPRINT
3454 4446 EM10-1
3455 5365 JMS I XPRINT
3456 4446 DH4-1
3457 5116 CLA CLL CMA
3460 7340 AND RANDB
3461 0041 DCA WD1
3462 3037 JMS I XWDOUT
3463 4461 CLA CLL CMA
3464 7340 AND RANDB
3465 0043 DCA WD1
3466 3037 JMS I XWDOUT
3467 4461 CLA CLL CMA
3470 7340 AND TEMPAC
3471 0025 DCA WD1
3472 3037 CMA
3473 7040 AND
3474 0026 DCA WD1
3475 3040 JMS I XLNKOU
3476 4460 JMS I XWDOUT
3477 4461 JMS I XPRINT
3500 4446 CRLF-1
3501 5542 LAS
3502 7604 AND SR00
3503 0103 SZA CLA SR00
3504 7640 JMP I RN1ER
3505 5647

```

```

/TEST SR01
/IS SR01=1
/YES, SUPPRESS ERROR TYPEOUT
/TYPE "RANDOM ADD TEST1 FAILED"
/TYPE "RANDB, RANDC, RESULT"
/OUTPUT RANDB
/OUTPUT RANDB
/OUTPUT RESULTANT LINK
/OUTPUT RESULTANT AC
/TEST SR00
/IS SR00=1
/YES, SUPPRESS ERROR HALT

```

3506 7300 CLA CLL RN1ER  
3507 1247 TAD RN1ER  
3510 7402 HLT  
3511 5647 JMP I RN1ER

/RANDOM NUMBER GENERATOR  
/

RAND0H, 0  
3512 0000  
3513 7300  
3514 1041 YAD RANDA  
3515 7004 RAL  
3516 7430 SZL  
3517 1342 YAD K0003 RANDA  
3520 3041 DCA RANDA  
3521 1041 YAD RANDC  
3522 7041 CIA R2A  
3523 3043 DCA  
3524 7100 CLL  
3525 1341 TAD R2A  
3526 7004 RAL  
3527 7430 SZL  
3530 1342 TAD K0003  
3531 3341 DCA R2A  
3532 7430 SZL  
3533 7040 CMA  
3534 3044 DCA LINKR  
3535 1044 TAD LINKR  
3536 7040 CMA  
3537 3045 DCA LINKRC  
3540 5712 JMP I RANDOM  
3541 0001 R2A,  
3542 0003 K0003,

/ADDITION OF RANDOM NUMBER AND MODIFIED  
/COMPLEMENT TO PRODUCE ONE KNOWN BIT  
/SET IN AC  
/

PAGE 3600  
3601 0041 RNAD2,  
3602 3346 CLA CLL CMA  
3623 7040 AND RANDA  
3624 2041 APOS  
3625 7040 DCA  
3626 3347 AND RANDA  
3627 7040 CMA ANEG  
3610 2103 AND K4000  
3611 3352 DCA MASK  
3612 7040 CMA

/ONE'S COMPLIMENT OF RANDOM NUMBER  
/GET MASK  
/GET RANDOM NUMBER  
/STORE IT

```

3613 0352      AND      MASK
3614 7240      CMA
3615 3353      DCA      VMASK
3616 7040      CMA
3617 2346      AND      APOS
3620 0352      AND      MASK
3621 7440      SZA
3622 5232      JMP      MODNEG
3623 7040      CMA
3624 0346      AND      APOS
3625 4301      JMS      XOR1
3626 7040      CMA
3627 0347      AND      ANEG
3630 3351      DCA      BNEG
3631 5240      JMP      CBTST1
3632 7240      CMA      CLA
3633 0347      AND      ANEG
3634 4315      JMS      XOR2
3635 7040      CMA
3636 0346      AND      APOS
3637 3351      DCA      BNEG
3640 7340      CLA      CLL
3641 0350      AND      BPOS
3642 1351      TAD      BNEG
3643 7430      SZL
3644 7001      IAC
3645 4464      JMS      XAVREG
3646 4463      JMS      XAMEA
3647 7410      SKP
3650 4756      JMS      XRN2ER
3651 4467      JMS      NERROP
3652 5240      JMP      CBTST1
3653 5254      JMP      CBTST2

3654 7340      CBTST2,  CLL  CLA  CMA
3655 0351      AND      BNEG
3656 1350      TAD      BPOS
3657 7430      SZL
3660 7001      IAC
3661 4464      JMS      XAVREG
3662 4463      JMS      XAMEA
3663 7410      SKP
3664 4756      JMS      XRN2ER
3665 4467      JMS      NERROP
3666 5254      JMP      CBTST2
      /SHIFT MASK ONE PLACE TO RIGHT

3667 7340      MOVMSK,  CLA  CLL  CMA
3670 0352      AND      MASK
3671 7010      RAR
3672 3352      DCA      MASK
3673 7420      SNL
3674 5212      JMP      NXTBT
3675 4467      JMS      NERROP
3676 5200      JMP      RNAD2

      /COMPLIMENT MASK
      /GET RANDOM NUMBER
      /TEST SIGN BIT
      /IS NUMBER NEGATIVE
      /YES, MODIFY COMPLIMENT OF NUMBER

      /GET RANDOM NUMBER
      /MODIFY WITH MASK

      /GET COMPLIMENT OF RANDOM NUMBER
      /AND USE AS IS

      /MODIFY NEGATIVE NUMBER
      /GET COMPLEMENT OF RANDOM NUMBER
      /MODIFY WITH MASK

      /GET RANDOM NUMBER
      /AND USE AS IS

      /LOAD AC WITH MODIFIED ARGUMENT
      /ADD UNMODIFIED ARGUMENT
      /DID CARRY PROPAGATE INTO LINK
      /NO, INCREMENT NUMBER
      /SAVE AC
      /COMPARE MODIFIED BIT AND MASK

      /AC AND MASK DIFFERENT, ERROR
      /NO ERROR, AC AND MASK THE SAME
      /RETURN HERE FOR LOOPING

      /LOAD AC WITH UNMODIFIED ARGUMENT
      /ADD MODIFIED ARGUMENT
      /DID CARRY PROPAGATE INTO LINK
      /NO, INCREMENT NUMBER
      /SAVE AC
      /COMPARE AC AND MASK

      /AC AND MASK NOT THE SAME, ERROR
      /NOERROR, AC AND MASK THE SAME
      /RETURN HERE FOR LOOPING

      /HAVE ALL BITS BEEN TESTED
      /NO, CONTINUE
      /YES, TEST FOR LOOP ON RNAD2

```

3677	5700	JMP I	.+1
3700	4200	RARR	
3721	0000	Z	
3722	2353	AND	\MASK
3723	7040	CMA	
3724	3354	DCA	ABNOT
3725	7040	CMA	
3726	0347	AND	ANEG
3727	0352	AND	MASK
3710	7040	CMA	
3711	0354	AND	ABNOT
3712	7040	CMA	
3713	3350	DCA	BPOS
3714	5701	JMP I	XOR1
3715	0000	Z	
3716	0352	AND	MASK
3717	7040	CMA	
3720	3354	DCA	ABNOT
3721	7040	CMA	
3722	0346	AND	APCS
3723	0353	AND	NMASK
3724	7040	CMA	
3725	0354	AND	ABNOT
3726	3350	DCA	BPOS
3727	5715	JMP I	XOR2

3730	0000	Z	
3731	7040	CMA	
3732	3355	DCA	NOTAC
3733	7040	CMA	
3734	0025	AND	TEMPAC
3735	0353	AND	NMASK
3736	7440	SZA	
3737	5344	JMP	EROUT1
3740	7040	CMA	
3741	0352	AND	MASK
3742	0355	AND	NOTAC
3743	7440	SZA	
3744	2330	ISE	SAMEA
3745	5730	JMP I	SAMEA
3746	0000	Z	
3747	0000	Z	
3750	0000	Z	
3751	0000	Z	
3752	0000	Z	
3753	0000	Z	
3754	0000	Z	
3755	0000	Z	
3756	4000	XRN2ER,	RA2ER

4000 PAGE

/ERROR HANDLER FOR RANDOM ADD TEST 2.

```

4000 0000 /TEST SR01
4001 7604 AND SR01
4002 0104 SZA CLA /IS SR01 = 1
4003 7640 JMP I SHLT /YES SUPPRESS ERROR TYPEOUT
4004 5233 JMS I XPRINT /NO, TYPE "RANDOM ADD TEST 2 FAILED"
4005 4446 EM11-1
4006 5405 JMS I XPRINT /TYPE ARG1, ARG2, ARG3+ARG2, EXPECTED
4007 4446 JMS I XPRINT
4008 5164 OH6-1
4009 7340 CLA CLL CMA
4010 0777 AND BPOS
4011 3037 DCA WD1
4012 0777 JMS I XWDOUT /OUTPUT ARG1
4013 3037 CMA
4014 4461 AND BNEG
4015 7040 DCA WD1
4016 0776 JMS I XWDOUT /OUTPUT ARG2
4017 3037 CMA
4018 4461 AND MASK
4019 7040 DCA WD1
4020 0775 JMS I XWDOUT /OUTPUT EXPECTED RESULT
4021 7040 CMA
4022 0775 AND TEMPAC
4023 3037 DCA WD1
4024 4461 JMS I XWDOUT /OUTPUT RESULTANT AC
4025 7040 CMA
4026 0025 AND SR00
4027 3037 DCA CLA /IS SR00 = 1
4028 4461 JMS I RN2ER /YES, DO NOT HALT
4029 5542 CLA CLL /NO, HALT WITH ADDRESS IN AC
4030 7604 SHLT,
4031 0103 AND SR00
4032 7640 SZA CLA
4033 5600 JMP I RN2ER
4034 7300 CLA CLL
4035 1200 TAD
4036 7402 HLT
4037 5600 JMP I RN2ER

```

/ROTATE RANDOM NUMBER RIGHT USING RAR

```

4175 3752 PAGE
4176 3751 RARR,
4177 3750 CLA CLL LINKR
4200 4200 /GET LINK TO BE ROTATED
4201 1044 TAD
4202 7440 SZA
4203 7220 CLA CML
4204 1041 TAD RANDA
/GET NUMBER TO BE ROTATED

```

4205	RAR				
4206	RAR				
4207	RAR				
4210	RAR				
4211	RAR				
4212	RAR				
4213	RAR				
4214	RAR				
4215	RAR				
4216	RAR				
4217	RAR				
4220	RAR				
4221	RAR				
4222	RAR				
4223	RAR				
4224	RAR				
4225	RAR				
4226	RAR				
4227	RAR				
4230	RAR				
4231	RAR				
4232	RAR				
4233	RAR				
4234	RAR				
4235	RAR				
4236	RAR				
4237	NOP				
4240	NOP				
4241	JMS	I	XAVREG		/SAVE AC AND LINK
4242	TAD		RANDG		/ADD COMPLEMENT OF NUMBER TO AC
4243	SEA	CLA			/ARE THEY EQUAL
4244	JMP				/NO, ERROR
4245	TAD				
4246	DCA				
4247	TAD				
4250	DCA				
4251	JMS	I	XAMEAS		/ARE LINKS THE SAME
4252	JMS	I	XRARR		/NO, ERROR
4253	JMS	I	NERROP		/TEST FOR LOOPING
4254	JMP		RARR		/LOOP ON RARR

4255	RALR,				
4256	CLA	CLL			/ROTATE RANDOM NUMBER LEFT USING RAL
4257	TAD		LINKR		/GET LINK TO BE ROTATED
4260	SEA				
4261	CLA	CHL	RANDA		/GET NUMBER TO BE ROTATED
4262	TAD				
4263	RAL				
4264	RAL				
4265	RAL				
4266	RAL				
4267	RAL				
4270	RAL				



```

4271 7004 RAL
4272 7004 RAL
4273 7004 RAL
4274 7004 RAL
4275 7004 RAL
4276 7004 RAL
4277 7004 RAL
4300 7004 RAL
4301 7004 RAL
4302 7004 RAL
4303 7004 RAL
4304 7004 RAL
4305 7004 RAL
4306 7004 RAL
4307 7004 RAL
4310 7004 RAL
4311 7004 RAL
4312 7004 RAL
4313 7004 RAL
4314 7000 NOP
4315 7000 NOP
4316 4464 JMS I
4317 1043 TAD
4320 7440 SEA
4321 5325 JMP
4322 1044 TAD
4323 3037 DCA
4324 1026 TAD
4325 3040 DCA
4326 4462 JMS I
4327 4734 JMS I
4330 4467 JMS I
4331 5255 JMP I
4332 5733 JMP I
4333 4400 RTL
4334 4613 XRALR,
4335 4600 XRARR,

XAVREG
RANDC
.04
LINKR
WD1
TEMPL
WD2
XAMEAS
XRALR
NERROP
RALR
.01

/SAVE AC AND LINK
/ADD COMPLIMENT OF ORIGINAL NUMBER TO AC
/ARE THEY THE SAME
/NO, ERROR

/COMPARE ORIGINAL AND ROTATED LINKS
/LINKS NOT THE SAME, ERROR

```

/ROTATE RANDOM NUMBER LEFT USING RTL

```

PAGE,
RTL,
4400 7300 CLA CLL
4401 1044 TAD
4402 7440 SZA
4403 7220 CLA CML
4404 1041 TAD
4405 7006 RTL
4406 7006 RTL
4407 7006 RTL
4410 7006 RTL
4411 7006 RTL
4412 7006 RTL
4413 7006 RTL
4414 7006 RTL

/GET LINK TO BE ROTATED
/GET NUMBER TO BE ROTATED

```

```

4415 7006 RTL
4416 7006 RTL
4417 7006 RTL
4420 7006 RTL
4421 7006 RTL
4422 7006 RTL
4423 7006 RTL
4424 7006 RTL
4425 7006 RTL
4425 7006 RTL
4427 7006 RTL
4430 7006 RTL
4431 7006 RTL
4432 7006 RTL
4433 7006 RTL
4434 7006 RTL
4435 7006 RTL
4436 7006 RTL
4437 7000 NOP
4440 7000 NOP
4441 4464 JMS I
4442 1045 TAD
4443 7440 SZA
4444 5250 JMP
4445 1044 TAD
4446 3037 DCA
4447 1026 TAD
4450 3040 DCA I
4451 4052 JMS I
4452 4760 JMS I
4453 0007 JMS I
4454 5200 JMP

XAVRES
RANDC
*G
LINKR
WD1
TEMPL
WD2
XAMEAS
XRTLRL
NERROR
RTLRL

/SAVE AC AND LINK
/ADD COMPLEMENT OF ORIGINAL NUMBER TO AC
/ARE THEY THE SAME
/NO, ERROR

/COMPARE ORIGINAL AND ROTATED LINKS
/LINKS NOT THE SAME, ERROR

```

```

RTRR,
4455 7300 CLA CLL
4456 1044 TAD LINKR
4457 7440 SZA
4460 7220 CLA CML
4461 1041 TAD RANDA
4462 7012 RTR
4463 7012 RTR
4464 7012 RTR
4465 7012 RTR
4466 7012 RTR
4467 7012 RTR
4470 7012 RTR
4471 7012 RTR
4472 7012 RTR
4473 7012 RTR
4474 7212 RTR
4475 7012 RTR
4476 7012 RTR
4477 7012 RTR

/ROTATE RANDOM NUMBER RIGHT USING RTR
/GET LINK TO BE ROTATED
/GET NUMBER TO BE ROTATED

```

```

4500 RTR
4501 RTR
4502 RTR
4503 RTR
4504 RTR
4505 RTR
4506 RTR
4507 RTR
4510 RTR
4511 RTR
4512 RTR
4513 RTR
4514 NOP
4515 NOP
4516 JMS I
4517 TAD
4520 SEA
4521 JMP
4522 TAD
4523 DCA
4524 TAD
4525 DCA
4526 JMS I
4527 JMS I
4530 JMS I
4531 JMP

XAVREG
RANDC
.+4
LINKR
WD1
TEMPL
WD2
XAMEAS
XRTRR
NERROP
RTRR

/SAVE AC AND LINK
/ADD COMPLEMENT OF ORIGINAL NUMBER TO AC
/ARE THEY THE SAME
/NO, ERROR

/ARE LINKS THE SAME
/NO, ERROR

```

```

4532 ISZ
4533 JMP
4534 LAS
4535 AND
4536 SNA CLA
4537 JMP
4540 LAS
4541 AND
4542 SEA CLA
4543 HLT
4544 LAS
4545 AND
4546 SEA CLA
4547 JMP
4550 JMP I
4551 RSIMAD
4552 JMS I
4553 OK4-1
4554 JMP
4555 JMP I
4556 RNAD1
4557 RTRR
4560 XRTLr,
4600 PAGE
4600 RARER, 0

CNR1
ENRN
SR10
RNDOK
SR09
SR11
ENRN
.+1
XPRINT
RNDHLT
.+1

/INCREMENT PASS COUNTER
/NOT END OF PASS
/TEST SR10
/IS SR10=1
/NO, TYPE RANDOM
/TEST SR09
/IS SR09=1
/YES, HALT AT END OF RANDOM
/TEST SR11
/IS SR11=1
/YES, LOOP ON RANDOM TESTS
/NO, GO TO SIMULATED ADDITION TEST

```

4571 7504 LAS  
 4602 0104 AND SR01  
 4623 7640 SEA CLA  
 4624 5210 JMP \*4  
 4625 4446 JMS I XPRINT  
 4627 5425 EM12-1 ROPRT  
 4610 7300 JMS CLA CLL  
 4611 1200 TAD RALER  
 4612 5253 JMP ROHLT

4613 0000 RALER, 0  
 4614 7604 LAS  
 4615 0104 AND SR01  
 4616 7640 SEA CLA  
 4617 5223 JMP \*4  
 4620 4446 JMS I XPRINT  
 4621 5444 EM13-1 ROPRT  
 4622 4264 JMS CLA CLL  
 4623 7300 TAD RALER  
 4624 1213 JMP ROHLT

4625 5253 RTRER, 0  
 4626 0000 0  
 4627 7604 LAS  
 4630 0104 AND SR01  
 4631 7640 SEA CLA  
 4632 5236 JMP \*4  
 4633 4446 JMS I XPRINT  
 4634 5463 EM14-1 ROPRT  
 4635 4264 JMS CLA CLL  
 4636 7300 TAD RTRER  
 4637 1226 JMP ROHLT

4640 5253 RTLER, 0  
 4641 0000 0  
 4642 7604 LAS  
 4643 0104 AND SR01  
 4644 7640 SEA CLA  
 4645 5251 JMP \*4  
 4646 4446 JMS I XPRINT  
 4647 5502 EM15-1 ROPRT  
 4650 4264 JMS CLA CLL  
 4651 7300 TAD RTLER  
 4652 1241 DCA ROBACK

4653 3263 ROHLT, 0  
 4654 7604 LAS  
 4655 0103 AND SR00  
 4656 7640 SEA CLA  
 4657 5262 JMP \*3  
 4660 1263 TAD ROBACK  
 4661 7402 HLT  
 4662 5663 JMP I ROBACK  
 4663 0000 ROBACK, 0

4664	0000	ROPRT,	0	JMS I	XPRINT
4665	4446			DH5-1	
4666	5147			CLA CLL	CMA
4667	7340			AND	LINKR
4670	0044			DCA	WD2
4671	3040			CMA	
4672	7040			AND	RANDA
4673	0041			DCA	WD1
4674	3037			JMS I	XLNKOU
4675	4460			JMS I	XWDOUT
4676	4461			CMA	
4677	7040			AND	TEMPL
4700	0026			DCA	WD2
4701	3040			JMS I	XLNKOU
4702	4460			CMA	
4703	7040			AND	TEMPAC
4704	0025			DCA	WD1
4705	3037			JMS I	XWDOUT
4706	4461			JMS I	XPRINT
4707	4446			CRLF-1	
4710	9542			JMP I	ROPRT
4711	5664				

PAGE 5000  
 DH1: 3736  
 ARG1+ARG2 5001  
 ARG2 4040  
 SIMULATED 4001  
 ARG1+ARG2 2207  
 ARG2 6140  
 ARG1+ARG2 5004  
 ARG2 4040  
 ARG1+ARG2 5005  
 ARG2 4040  
 ARG1+ARG2 5006  
 ARG2 4040  
 ARG1+ARG2 5007  
 ARG2 4040  
 ARG1+ARG2 5010  
 ARG2 4040  
 ARG1+ARG2 5011  
 ARG2 0122  
 ARG1+ARG2 5012  
 ARG2 0762  
 ARG1+ARG2 5013  
 ARG2 4040  
 ARG1+ARG2 5014  
 ARG2 4040  
 ARG1+ARG2 5015  
 ARG2 4040  
 ARG1+ARG2 5017  
 ARG2 4023  
 ARG1+ARG2 5020  
 ARG2 1115  
 ARG1+ARG2 5021  
 ARG2 2514  
 ARG1+ARG2 5022  
 ARG2 0124  
 ARG1+ARG2 5023  
 ARG2 0504  
 ARG1+ARG2 5024  
 ARG2 4040  
 ARG1+ARG2 5025  
 ARG2 4040  
 ARG1+ARG2 5026  
 ARG2 4040  
 ARG1+ARG2 5027  
 ARG2 4001  
 ARG1+ARG2 5030  
 ARG2 2207  
 ARG1+ARG2 5031  
 ARG2 6153  
 ARG1+ARG2 5032  
 ARG2 0122



5122	0140
5123	4040
5124	4040
5125	4040
5126	4002
5127	2017
5130	2340
5131	4040
5132	4040
5133	4040
5134	4040
5135	0216
5136	0507
5137	4040
5140	4040
5141	4040
5142	4040
5143	4022
5144	0523
5145	2514
5146	2437
5147	3600
5150	3736
5151	1722
5152	1107
5153	1116
5154	0114
5155	4040
5156	4040
5157	4040
5160	0103
5161	2425
5162	0114
5163	3736
5164	0000
5165	3736
5166	4040
5167	4040
5170	0122
5171	0761
5172	4040
5173	4040
5174	4040
5175	4040
5176	4001
5177	2207
5200	6240
5201	4040
5202	4040
5203	4040
5204	0530
5205	2005
5206	0324
5207	0504
5210	4040

DH5, TEXT /% ORIGINAL ACTUAL\*\*/

DH6, TEXT /% ARG1 ARG2 EXPECTED ACTUAL\*\*/

5211	4040			
5212	4040			
5213	0103			
5214	2425			
5215	0114			
5216	3736			
5217	0000			
5220	3736			
5221	4040			
5222	4040			
5223	4023			
5224	1115			
5225	2514			
5226	0124			
5227	0504			
5230	4001			
5231	0404			
5232	4024			
5233	0523			
5234	2440			
5235	0601			
5236	1114			
5237	0504			
5240	0000			
5241	3736			
5242	4040			
5243	4040			
5244	4023			
5245	1115			
5246	2514			
5247	0124			
5250	0504			
5251	4022			
5252	0114			
5253	4024			
5254	0523			
5255	2440			
5256	0601			
5257	1114			
5260	0504			
5261	0000			
5262	3736			
5263	4040			
5264	4040			
5265	4023			
5266	1115			
5267	2514			
5270	0124			
5271	0504			
5272	4022			
5273	0122			
5274	4024			
5275	0523			
5276	2440			
5277	0601			

EM1, TEXT /% SIMULATED ADD TEST FAILED/

EM2, TEXT /% SIMULATED RAL TEST FAILED/

EM3, TEXT /% SIMULATED RAR TEST FAILED/



EM4,	TEXT	/00	SIMULATED RTL TEST FAILED/
5350	1114		
5351	0504		
5352	0000		
5353	3736		
5354	4040		
5355	4040		
5356	4023		
5357	1119		
5358	2514		
5359	0124		
5360	0504		
5361	4022		
5362	2414		
5363	4024		
5364	0523		
5365	2440		
5366	0601		
5367	1114		
5368	0504		
5369	0000		
5370	3736		
5371	4040		
5372	4040		
5373	4023		
5374	1119		
5375	2514		
5376	0124		
5377	0504		
5378	4022		
5379	2422		
5380	4024		
5381	0523		
5382	2440		
5383	0601		
5384	1114		
5385	0504		
5386	0000		
5387	3736		
5388	4040		
5389	4040		
5390	4023		
5391	1119		
5392	2514		
5393	0124		
5394	0504		
5395	4022		
5396	2422		
5397	4024		
5398	0523		
5399	2440		
5400	0601		
5401	1114		
5402	0504		
5403	0000		
5404	3736		
5405	4040		
5406	4040		
5407	4023		
5408	1119		
5409	2514		
5410	0124		
5411	0504		
5412	4022		
5413	2414		
5414	4024		
5415	0523		
5416	2440		
5417	0601		
5418	1114		
5419	0504		
5420	0000		
5421	3736		
5422	4040		
5423	4040		
5424	4023		
5425	1119		
5426	2514		
5427	0124		
5428	0504		
5429	4022		
5430	2422		
5431	4024		
5432	0523		
5433	2440		
5434	0601		
5435	1114		
5436	0504		
5437	0000		
5438	3736		
5439	4040		
5440	4040		
5441	4023		
5442	1119		
5443	2514		
5444	0124		
5445	0504		
5446	4022		
5447	2422		
5448	4024		
5449	0523		
5450	2440		
5451	0601		
5452	1114		
5453	0504		
5454	4022		
5455	2327		
5456	4024		
5457	0523		
5458	2440		
5459	0601		
5460	1114		
5461	0504		
5462	0000		
5463	3736		

EM5, TEXT /00 SIMULATED RTR TEST FAILED/

EM6, TEXT /00 SIMULATED BSW TEST FAILED/

EM10, TEXT /00 RANDOM ADD TEST 1 FAILED/

5367 4040  
5370 4240  
5371 4222  
5372 0116  
5373 0417  
5374 1540  
5375 0104  
5376 0440  
5377 2405  
5400 2324  
5401 4001  
5402 4006  
5403 0111  
5404 1405  
5405 0400  
5406 3736  
5407 4040  
5410 4040  
5411 4022  
5412 0116  
5413 0417  
5414 1540  
5415 0104  
5416 0440  
5417 2405  
5420 2324  
5421 4062  
5422 4006  
5423 0111  
5424 1405  
5425 0400  
5426 3736  
5427 4040  
5430 4040  
5431 4022  
5432 0116  
5433 0417  
5434 1540  
5435 2201  
5436 2240  
5437 2405  
5440 2324  
5441 4006  
5442 0111  
5443 1405  
5444 0400  
5445 3736  
5446 4040  
5447 4040  
5450 4022  
5451 0116  
5452 0417  
5453 1540  
5454 2201  
5455 1440

EM11, TEXT /00 RANDOM ADD TEST 2 FAILED/

EM12, TEXT /00 RANDOM RAR TEST FAILED/

EM13, TEXT /00 RANDOM RAL TEST FAILED/

FAILD

5456 2405  
 5457 2324  
 5458 4006  
 5459 0111  
 5460 1405  
 5461 0400  
 5462 3736  
 5463 4040  
 5464 4040  
 5465 4040  
 5466 0116  
 5467 0417  
 5468 1540  
 5469 2224  
 5470 1440  
 5471 2405  
 5472 2324  
 5473 4006  
 5474 0111  
 5475 1405  
 5476 0400  
 5477 3736  
 5478 4040  
 5479 4040  
 5480 4022  
 5481 0116  
 5482 0417  
 5483 1540  
 5484 2224  
 5485 2405  
 5486 2324  
 5487 4006  
 5488 0111  
 5489 1405  
 5490 0400  
 5491 3736  
 5492 4040  
 5493 4040  
 5494 4022  
 5495 0116  
 5496 0417  
 5497 1540  
 5498 2224  
 5499 2405  
 5500 2324  
 5501 4006  
 5502 0111  
 5503 1405  
 5504 0400  
 5505 3736  
 5506 4040  
 5507 4022  
 5508 0116  
 5509 0417  
 5510 1540  
 5511 2224  
 5512 2405  
 5513 2324  
 5514 4006  
 5515 0111  
 5516 1405  
 5517 0400  
 5518 3736  
 5519 2311  
 5520 1522  
 5521 1724  
 5522 0000  
 5523 3736  
 5524 0603  
 5525 2400  
 5526 3736  
 5527 2201  
 5528 1604  
 5529 1715  
 5530 0000  
 5531 3736  
 5532 0000  
 5533 1604  
 5534 1715  
 5535 0000  
 5536 3736  
 5537 0000  
 5538 1604  
 5539 1715  
 5540 0000  
 5541 3736  
 5542 0000  
 5543 0000

EM14, TEXT / \*\* RANDOM RTL TEST FAILED/

EM15, TEXT / \*\* RANDOM RTR TEST FAILED/

OK1, TEXT / \*\* SIMAD/

OK2, TEXT / \*\* SIMROT/

OK3, TEXT / \*\* FCT/

OK4, TEXT / \*\* RANDOM/

CRLF, TEXT / \*\* /

TEXT /- DATA ERROR-/-

DATE	TIME
5545	3736
5546	4004
5547	0124
5550	0140
5551	0522
5552	2257
5553	2237
5554	3600

```

/
/
/RESTORE BINARY LOADER AND START LOADER
/

```

ADDRESS	OPERATION	OPERAND
7600	CLA	CLL
7601	TAD	BIN
7602	DCA	TSTAR
7603	JMP	TSTAR
		S





A1	3263	EM1	3220	FCT3	2076	K4000	0103
A10	0027	EM10	5366	FCT4	2200	K6000	0101
A2	3264	EM11	5406	FCT5	2232	K6060	3270
ABNOV	3754	EM12	5426	FCT6	2270	KXXXX	0170
AD1	2127	EM13	5445	FCT7	2400	LINK1	0032
AD10	2715	EM14	5464	FCT8	2436	LINK2	0034
AD11	2716	EM15	5503	FCT9	2472	LINKR	0044
AD12	2717	EM2	5241	FCTHLT	3207	LINKRC	0045
AD2	2130	EM3	3262	FCTOK	3221	LINKOUT	0004
AD3	2131	EM4	3303	GOPEST	0177	LOOP	5046
AD4	2324	EM5	3324	HALT2	3027	LOOP1	0552
AD5	2325	EM6	3345	HALTA	0477	H4	1070
AD6	2326	ENCAR	0243	HALTB	1063	H40	1046
AD7	2526	ENCAR1	0252	HLTA	0004	MASK	3732
AD8	2527	ENBSSW	1277	HLTB	1052	MODNEG	3632
AD9	2530	ENDECT	3200	INS1	0136	MOVMSK	3667
ADA1	0021	ENDROT	1303	INS10	0146	MOA	7501
ADA2	0022	ENRN	4555	INS11	0147	HCL	7421
ADB	0273	EROUT1	3744	INS12	0150	MTP	1643
ADDERR	0400	ERROR1	0377	INS13	0151	NIBIT	1244
ADHLT	0556	ERROR2	1046	INS14	0152	NBIT	1207
ADOUT	3227	ERROT	1026	INS15	0153	NERROP	0067
ADPRT	0417	FCL1	2023	INS16	2133	NEHLNK	1044
ADT	0551	FCL10	2612	INS2	2132	NLOOP	3054
AHFLG	0035	FCL11	2644	INS3	0137	NMASK	3753
AHOUT	0467	FCL12	2677	INS4	0140	NOTAG	3755
ALTIBT	3616	FCL3	2056	INS5	0141	NXBIT	0527
ANEC	3747	FCL4	2107	INS6	0142	NXTADD	0364
APOS	3746	FCL4	2213	INS7	0143	NXTBT	3612
ARG1	0023	FCL5	2246	INS8	0144	NXTCAR	0233
ARG2	0024	FCL6	2302	INS9	0145	NXTROT	1031
GIN	0155	FCL7	2412	K0001	0116	OK1	5522
ENEG	3751	FCL8	2450	K0002	0115	OK2	5526
EPOS	3750	FCL9	2503	K0003	3542	OK3	5533
BSW	7002	FCS1	2005	K0004	0114	OK4	5536
BSWTAB	1660	FCS10	2601	K0007	3266	ORI	1225
CAF	6007	FCS11	2635	K0010	0113	OUT	3224
CARRY	0030	FCS12	2670	K0020	0112	OUT1	0520
CBTST1	3640	FCS2	2044	K0040	0111	OUT1A	0542
CBTST2	3654	FCS3	2077	K0077	1645	POINT1	0011
CHAR	0036	FCS4	2201	K0100	0110	POINT2	0012
CNTR1	0020	FCS5	2233	K0200	0107	PRINT	1600
CNTR2	1000	FCS6	2271	K0400	0106	R1	1400
CMCT	5543	FCS7	2401	K0700	3267	R2	1410
CATE	5545	FCS8	2437	K1000	0105	R2A	3541
CATER	3037	FCS9	2473	K2000	0104	R3	1420
CH1	5000	FCT	2000	K212	1650	R4	1430
CH2	5045	FCT1	2004	K215	1651	R5	1440
CH3	5074	FCT10	2600	K240	0076	RAC	0023
CH4	5117	FCT11	2634	K260	0077	RALER	1013
CH5	5150	FCT12	2667	K261	0100	RALR	4255
CH6	5165	FCT2	2043	K336	1647	RALTAB	0757

NAME	ADDRESS	PHONE	DATE	PAGE	NO.
RANDA	0041	1236	7776	7776	0090
RANDB	0242	0154	7777	7777	0091
RANDE	0243	0122	0000	0000	0092
RANDE	3512	0133	0001	0001	0093
RANDE	4620	0134	0002	0002	0094
RANDE	4200	0135	0003	0003	0095
RANDE	0731	0123	0004	0004	0096
RETI	2025	0124	0010	0010	0097
RETI	2616	0125	0540	0540	0098
RETI	2651	0126	0513	0513	0099
RETI	2701	0127	1692	1692	0100
RETI	2060	0130	3096	3096	0101
RETI	2111	0131	1617	1617	0102
RETI	2215	0132	0037	0037	0103
RETI	2252	0033	0040	0040	0104
RETI	2306	0025	0037	0037	0105
RETI	2420	0203	0040	0040	0106
RETI	2455	0725	0523	0523	0107
RETI	2510	0577	0415	0415	0108
RHD	1133	0577	0073	0073	0109
RHFLG	0035	0601	0063	0063	0110
RHOUT	1131	0626	0002	0002	0111
RLNK	0024	0600	0004	0004	0112
RN1R	3447	0625	0055	0055	0113
RN2R	4000	0652	0056	0056	0114
RNAD1	3400	0677	0065	0065	0115
RNAD2	3600	0724	0066	0066	0116
RNDHLT	4540	0653	0060	0060	0117
RNDOK	4552	0700	0067	0067	0118
ROBACK	4663	3562	0075	0075	0119
ROHLT	4653	0545	0074	0074	0120
ROPR	4664	0103	1450	1450	0121
ROTDNE	1323	0104	1451	1451	0122
ROTHLT	1327	0105	1452	1452	0123
ROTPRT	1071	0106	1453	1453	0124
RRAC	0031	0107	1454	1454	0125
RRAL	0605	0110	0416	0416	0126
RRAR	0632	0111	0057	0057	0127
RRLNK	0033	0112	3701	3701	0128
RRTL	0657	0113	3715	3715	0129
RRTR	0704	0114	0257	0257	0130
RSIMAD	0200	0115	0046	0046	0131
RYLER	4641	0116	0752	0752	0132
RTL	4400	1200	0753	0753	0133
RTLTAB	1160	1342	0754	0754	0134
RTTR	4626	0156	0755	0755	0135
RTTR	4455	0031	0756	0756	0136
RTTAB	1141	2033	4334	4334	0137
SADOK	2570	0037	0052	0052	0138
SAMEAS	3730	0025	0073	0073	0139
SAMEAS	3020	0026	4335	4335	0140
SAVREG	3017	7775	0102	0102	0141



ERRORS DETECTED 0

LINKS GENERATED 3

RUN-TIME: 17 SECONDS

3K CORE USED

